A Sequential NCP Algorithm for Solving Equilibrium Problems with Equilibrium Constraints

Che-Lin Su*

Original: October 30, 2004; Revision: November 28, 2004

Abstract. This paper studies algorithms for equilibrium problems with equilibrium constraints (EPECs). We present a generalization of Scholtes's regularization scheme for MPECs and extend his convergence results to this new relaxation method. We propose a sequential nonlinear complementarity (SNCP) algorithm to solve EPECs and establish the convergence of this algorithm. We present numerical results comparing the SNCP algorithm and diagonalization (nonlinear Gauss-Seidel and nonlinear Jacobi) methods on randomly generated EPEC test problems. The computational experience to date shows that both the SNCP algorithm and the nonlinear Gauss-Seidel method outperform the nonlinear Jacobi method.

1 Introduction

An *Equilibrium Problem with Equilibrium Constraints* (EPEC) is a mathematical program to find an equilibrium point that simultaneously solves several *Mathematical Programs with Equilibrium Constraints* (MPECs), each of which is parameterized by decision variables of other MPECs. The applications of EPECs often arise from noncooperative games, especially multi-leader-follower games [20], where each leader is solving a Stackelberg game formulated as an MPEC [14]. Several EPEC models have been developed to study the strategic behavior of generating firms in deregulated electricity markets [3, 9, 17]. In a companion paper [24], we investigate a spot-forward equilibrium model arising in the field of supply chain management and compute a forward market equilibrium formulated as an EPEC.

This paper is organized as follows. In the next section, we review the recent theoretical studies on MPECs. In Section 3, we propose a generalization of Scholtes's regularization scheme and show that the convergence theorems studied in [23] can be extended to our

*Department of Management Science and Engineering, Stanford University (clsu@stanford.edu). This research was supported by the Electric Power Research Institute (EPRI).

approach. Section 4 introduces EPECs and their definitions of stationary concepts. Diagonalization methods [2, 3, 9, 11, 17] have been widely used by researchers in engineering fields to solve EPECs. We briefly discuss the convergence properties of the diagonalization methods based on an equivalent NLP approach [6] to solve each MPEC subproblem. We also propose a sequential nonlinear complementarity (SNCP) approach for solving EPECs and establish the convergence of this algorithm. Section 5 presents numerical results comparing diagonalization methods and the SNCP algorithm on randomly generated EPEC test problems.

A word about the notation: For $u \in R^n$ and $v \in R^m$, (u, v) denotes the column vector $[u^{\mathrm{T}}, v^{\mathrm{T}}]^{\mathrm{T}} \in R^{n+m}$. If u and v are two vectors in R^n, then $u \circ v$ denotes the Hadamard product, i.e., $u \circ v = [u_1 v_1, \cdots, u_n v_n]^{\mathrm{T}} \in R^n$, and $u \perp v$ indicates orthogonality of vectors u and v. We let e denote the vector of all ones in the appropriate dimension. By $\|\cdot\|$, we mean the Euclidean norm. If a function $f : R^n \to R$ is twice continuously differentiable, then $\nabla f(x) \in R^n$ and $\nabla^2 f(x) \in R^{n\times n}$ denote the gradient and the Hessian matrix of f at x, respectively. If the mapping g is differentiable, then $\nabla g(x) \in R^{m\times n}$ denotes the Jacobian matrix, in which the i-th row is $\nabla g_i(x)^{\mathrm{T}}$. For a matrix $A \in R^{m\times n}$ with linearly independent rows, a *basis* in A will consist of m linearly independent columns of A which together form a $m \times m$ matrix $A_{\bullet\beta}$, where β is the set of indices of those m independent columns.

2 Preliminaries on MPECs

We consider an MPEC formulated as a nonlinear program with complementarity constraints:

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & g(x) \le 0, \quad h(x) = 0, \\ & 0 \le G(x) \perp H(x) \ge 0, \end{array} \tag{1}$$

where $f : R^n \to R,\ g : R^n \to R^p,\ h : R^n \to R^q,\ G : R^n \to R^m$, and $H : R^n \to R^m$ are twice continuously differentiable functions. Given a feasible vector $\bar{x}$ of the MPEC (1), we define the following index sets of active and inactive constraints:

$$\begin{aligned} \mathcal{I}_g(\bar{x}) &:= \{i \,|\, g_i(\bar{x}) = 0\}, \\ \mathcal{I}_G(\bar{x}) &:= \{i \,|\, G_i(\bar{x}) = 0\}, \\ \mathcal{I}_G^c(\bar{x}) &:= \{i \,|\, G_i(\bar{x}) > 0\}, \\ \mathcal{I}_H(\bar{x}) &:= \{i \,|\, H_i(\bar{x}) = 0\}, \\ \mathcal{I}_H^c(\bar{x}) &:= \{i \,|\, H_i(\bar{x}) > 0\}, \\ \mathcal{I}_{GH}(\bar{x}) &:= \{i \,|\, G_i(\bar{x}) = H_i(\bar{x}) = 0\}, \end{aligned} \tag{2}$$

where $\mathcal{I}_{GH}(\bar{x})$ is known as the *degenerate* set. If $\mathcal{I}_{GH}(\bar{x}) = \emptyset$, then the feasible vector $\bar{x}$ is said to fulfill the *strict complementarity* conditions.

Associated with any given feasible vector $\bar{x}$ of MPEC (1), there is a nonlinear program, called the *tightened* NLP (TNLP($\bar{x}$)) [19, 22]:

$$\begin{array}{lll} \text{minimize} & f(x) & \\ \text{subject to} & g(x) \leq 0, & h(x) = 0, \\ & G_i(x) = 0, & i \in \mathcal{I}_G(\bar{x}), \\ & G_i(x) \geq 0, & i \in \mathcal{I}_G^c(\bar{x}), \\ & H_i(x) = 0, & i \in \mathcal{I}_H(\bar{x}), \\ & H_i(x) \geq 0, & i \in \mathcal{I}_H^c(\bar{x}). \end{array} \tag{3}$$

Similarly, there is a *relaxed* NLP (RNLP($\bar{x}$)) [19, 22] defined as follows:

$$\begin{array}{lll} \text{minimize} & f(x) & \\ \text{subject to} & g(x) \leq 0, & h(x) = 0, \\ & G_i(x) = 0, & i \in \mathcal{I}_H^c(\bar{x}), \\ & G_i(x) \geq 0, & i \in \mathcal{I}_H(\bar{x}), \\ & H_i(x) = 0, & i \in \mathcal{I}_G^c(\bar{x}), \\ & H_i(x) \geq 0, & i \in \mathcal{I}_G(\bar{x}). \end{array} \tag{4}$$

It is well known that an MPEC cannot satisfy the standard constraint qualifications, such as *linear independence constraint qualification* (LICQ) or *Mangasarian-Fromovitz constraint qualification* (MFCQ), at any feasible point [4, 22]. This implies that the classical KKT theorem on necessary optimality conditions (with the assumption that LICQ or MFCQ is satisfied at local minimizers) are not appropriate in the context of MPECs. One then needs to develop suitable variants of CQs and concepts of stationarity for MPECs. Specifically, the MPEC-CQs are closely related to those of the RNLP (4).

Definition 2.1. The MPEC (1) is said to satisfy the MPEC-LICQ (MPEC-MFCQ) at a feasible point $\bar{x}$ if the corresponding RNLP($\bar{x}$) (4) satisfies the LICQ (MFCQ) at $\bar{x}$.

In what follows, we define B(ouligand)-stationarity for MPECs. We also summarize various stationarity concepts for MPECs introduced in Scheel and Scholtes [22].

Definition 2.2. Let $\bar{x}$ be a feasible point for the MPEC (1). We say that $\bar{x}$ is a Bouligand- or B-stationary point if $d = 0$ solves the following *linear program with equilibrium constraints*

(LPEC) with the vector $d \in R^n$ being the decision variable:

$$\begin{array}{ll} \text{minimize} & \nabla f(\bar{x})^{\mathrm{T}} d \\ \text{subject to} & g(\bar{x}) + \nabla g(\bar{x})^{\mathrm{T}} d \leq 0, \quad h(\bar{x}) + \nabla h(\bar{x})^{\mathrm{T}} d = 0, \\ & 0 \leq G(\bar{x}) + \nabla G(\bar{x})^{\mathrm{T}} d \perp H(\bar{x}) + \nabla H(\bar{x})^{\mathrm{T}} d \geq 0. \end{array} \tag{5}$$

B-stationary points are good candidates for local minimizers of the MPEC (1). However, checking B-stationarity is difficult because it may require checking the optimality of $2^{|\mathcal{I}_{GH}(\bar{x})|}$ linear programs [14, 22].

Definition 2.3. We define the MPEC Lagrangian with the vector of MPEC multipliers $\lambda = (\lambda^g, \lambda^h, \lambda^G, \lambda^H)$ as in Scholtes [23]:

$$\mathcal{L}(x, \lambda) = f(x) + (\lambda^g)^{\mathrm{T}} g(x) + (\lambda^h)^{\mathrm{T}} h(x) - (\lambda^G)^{\mathrm{T}} G(x) - (\lambda^H)^{\mathrm{T}} H(x). \tag{6}$$

Notice that the complementarity constraint $G(x)^{\mathrm{T}} H(x) = 0$ does not appear in the MPEC Lagrangian function. This special feature distinguishes MPECs from standard nonlinear programming problems.

The following four concepts of MPEC stationarity, stated in increasing strength, are introduced in Scheel and Scholtes [22].

Definition 2.4. A feasible point $\bar{x}$ of the MPEC (1) is called *weakly stationary* if there exists a vector of MPEC multipliers $\bar{\lambda} = (\bar{\lambda}^g, \bar{\lambda}^h, \bar{\lambda}^G, \bar{\lambda}^H)$ such that $(\bar{x}, \bar{\lambda})$ is a KKT stationary point of the TNLP (3), i.e., $(\bar{x}, \bar{\lambda})$ satisfies the following conditions:

$$\nabla_x \mathcal{L}(\bar{x}, \bar{\lambda}) = \nabla f(\bar{x}) + \nabla g(\bar{x})^{\mathrm{T}} \bar{\lambda}^g + \nabla h(\bar{x})^{\mathrm{T}} \bar{\lambda}^h - \nabla G(\bar{x})^{\mathrm{T}} \bar{\lambda}^G - \nabla H(\bar{x})^{\mathrm{T}} \bar{\lambda}^H = 0,$$

$$\begin{array}{lrrr} & h(\bar{x}) = 0; \quad g(\bar{x}) \leq 0, & \bar{\lambda}^g \geq 0, & (\bar{\lambda}^g)^{\mathrm{T}} g(\bar{x}) = 0, \\ i \in \mathcal{I}_G(\bar{x}): & & & G_i(\bar{x}) = 0, \\ i \in \mathcal{I}_G^c(\bar{x}): & G_i(\bar{x}) \geq 0, & \bar{\lambda}_i^G \geq 0, & \bar{\lambda}_i^G G_i(\bar{x}) = 0, \\ i \in \mathcal{I}_H(\bar{x}): & & & H_i(\bar{x}) = 0, \\ i \in \mathcal{I}_H^c(\bar{x}): & H_i(\bar{x}) \geq 0, & \bar{\lambda}_i^H \geq 0, & \bar{\lambda}_i^H H_i(\bar{x}) = 0. \end{array} \tag{7}$$

In addition, the feasible vector $\bar{x}$ is called

(a) a *C(larke)-stationary* point if $\bar{\lambda}_i^G \bar{\lambda}_i^H \geq 0 \quad \forall\, i \in \mathcal{I}_{GH}(\bar{x})$.

(b) a *M(ouduckhovich)-stationary* point if either $(\bar{\lambda}_i^G > 0,\ \bar{\lambda}_i^H > 0)$ or $(\bar{\lambda}_i^G \bar{\lambda}_i^H = 0)$ $\forall\, i \in \mathcal{I}_{GH}(\bar{x})$.

(c) a *strongly stationary* point if $\bar{\lambda}_i^G \geq 0, \bar{\lambda}_i^H \geq 0 \quad \forall\, i \in \mathcal{I}_{GH}(\bar{x})$.

Notice that by Definition 2.4, a point $\bar{x}$ is a strongly stationary point of the MPEC (1) if $(\bar{x}, \bar{\lambda})$ is a KKT pair of the RNLP (4).

Definition 2.5 Upper-level strict complementarity (ULSC). A weakly stationary point $\bar{x}$ is said to satisfy ULSC if there exist MPEC multipliers $\bar{\lambda} = (\bar{\lambda}^g, \bar{\lambda}^h, \bar{\lambda}^G, \bar{\lambda}^H)$ satisfying (7) with $\bar{\lambda}_i^G \bar{\lambda}_i^H \neq 0$ for all $i \in \mathcal{I}_{GH}(\bar{x})$.

See [22] for a discussion of these various stationarity conditions and their relations to others in the literature such as Clarke's generalized stationarity.

The following two theorems relate the strongly stationary point to B-stationary point and local minimizers of MPECs.

Theorem 2.6 ([22]). If a feasible point $\bar{x}$ is a strong stationary point for the MPEC (1), then it is a B-stationary point. Conversely, if $\bar{x}$ is B-stationary of the MPEC (1), and if the MPEC-LICQ holds at $\bar{x}$, then it is a strongly stationary point.

Theorem 2.7 ([19, 22]). If the MPEC-LICQ holds at a local minimizer $\bar{x}$ of the MPEC (1), then $\bar{x}$ is a strongly stationary point with a unique vector of MPEC multipliers $\bar{\lambda} = (\bar{\lambda}^g, \bar{\lambda}^h, \bar{\lambda}^G, \bar{\lambda}^H)$.

Fletcher and Leyffer [6] suggest reformulating the MPEC (1) as the following equivalent nonlinear program:

$$\begin{array}{lll} \text{minimize} & f(x) & \\ \text{subject to} & g(x) \leq 0, \quad h(x) = 0, & \\ & G(x) \geq 0, \quad H(x) \geq 0, & \\ & G(x) \circ H(x) \leq 0. & \end{array} \tag{8}$$

An interesting observation by Anitescu [1] and Fletcher et al. [7] on the equivalent nonlinear program (8) is stated in the following theorem.

Theorem 2.8 ([1, 7, 13]). A vector $\bar{x}$ is a *strongly stationary point* of the MPEC (1) if and only if it is a KKT point of nonlinear program (8), i.e., there exists a vector of Lagrangian multipliers $\hat{\lambda} = (\hat{\lambda}^g, \hat{\lambda}^h, \hat{\lambda}^G, \hat{\lambda}^H, \hat{\lambda}^{GH})$, such that $(\bar{x}, \hat{\lambda})$ satisfies the following conditions:

$$\begin{aligned} \nabla f(\bar{x}) + \nabla g(\bar{x})^{\mathrm{T}} \hat{\lambda}^g + \nabla h(\bar{x})^{\mathrm{T}} \hat{\lambda}^h & \\ -\nabla G(\bar{x})^{\mathrm{T}} [\hat{\lambda}^G - H(\bar{x}) \circ \hat{\lambda}^{GH}] - \nabla H(\bar{x})^{\mathrm{T}} [\hat{\lambda}^H - G(\bar{x}) \circ \hat{\lambda}^{GH}] & = 0, \\ h(\bar{x}) = 0; \quad g(\bar{x}) \leq 0, \quad \hat{\lambda}^g \geq 0, \quad (\hat{\lambda}^g)^{\mathrm{T}} g(\bar{x}) & = 0, \\ G(\bar{x}) \geq 0, \quad \hat{\lambda}^G \geq 0, \quad (\hat{\lambda}^G)^{\mathrm{T}} G(\bar{x}) & = 0, \\ H(\bar{x}) \geq 0, \quad \hat{\lambda}^H \geq 0, \quad (\hat{\lambda}^H)^{\mathrm{T}} G(\bar{x}) & = 0, \\ G(\bar{x}) \circ H(\bar{x}) \leq 0, \quad \hat{\lambda}^{GH} \geq 0, \quad (\hat{\lambda}^{GH})^{\mathrm{T}} [G(\bar{x}) \circ H(\bar{x})] & = 0. \end{aligned} \tag{9}$$

3 A Generalization of Scholtes's Regularization

In this section, we present a generalization of Scholtes's regularization scheme [23]. Our approach suggests relaxing the complementarity constraints and perturbing the coefficients in the objective function and constraints simultaneously. Hence, Scholtes's scheme is a special case of our approach if the objective function and constraints are not perturbed. We show that the convergence analysis studied in [23] can be extended to our method without any difficulty. The convergence results of our method will be applied to establish the convergence of the sequential nonlinear complementarity algorithm in the next section.

For any mapping $F : R^n \times \mathcal{A}^F \to R^m$, where R^n is the space of variables and $\mathcal{A}^F$ is the space of (fixed) parameters, we denote the mapping as $F(x; \bar{a}^F)$ with $x \in R^n$ and $\bar{a}^F \in \mathcal{A}^F$. The order of elements in $\bar{a}^F$ is mapping specific. For any positive sequence $\{t\}$ tending to 0, we perturb the parameters in F and denote the new parameter vector as a_t^F with $a_t^F \to \bar{a}^F$ as $t \to 0$, and $a_t^F = \bar{a}^F$ when $t = 0$. Note that the perturbation on $\bar{a}^F$ does not require the perturbed vector a_t^F to be parameterized by t.

Let $\Omega := \{f, g, h, G, H\}$ be the collection of all the functions in the MPEC (1). With the notation defined above, the MPEC (1) is presented as

$$\begin{array}{ll} \text{minimize} & f(x; \bar{a}^f) \\ \text{subject to} & g(x; \bar{a}^g) \le 0, \quad h(x; \bar{a}^h) = 0, \\ & 0 \le G(x; \bar{a}^G) \perp H(x; \bar{a}^H) \ge 0, \end{array} \tag{10}$$

where $\bar{a}^\omega \in \mathcal{A}^\omega$, for all $\omega \in \Omega$.

For any positive sequence $\{t\}$ tending to 0, we perturb every parameter vector $\bar{a}^\omega$ and denote the perturbed parameter vector as a_t^ω for all $\omega \in \Omega$. The perturbed vector a_t^ω should satisfy the following two conditions for all $\omega \in \Omega$:

$$a_t^\omega \to \bar{a}^\omega, \quad \text{as } t \to 0^+. \tag{11}$$

$$a_t^\omega = \bar{a}^\omega, \quad \text{when } t = 0. \tag{12}$$

As $\{t\} \to 0^+$, we are solving a sequence of perturbed NLPs, denoted by Reg(t):

$$\begin{array}{ll} \text{minimize} & f(x; a_t^f) \\ \text{subject to} & g(x; a_t^g) \le 0, \quad h(x; a_t^h) = 0, \\ & G(x; a_t^G) \ge 0, \quad H(x; a_t^H) \ge 0, \\ & G(x; a_t^G) \circ H(x; a_t^H) \le t\, e. \end{array} \tag{13}$$

In what follows, we extend Theorem 3.1, Theorem 3.3, and Corollary 3.4 in [23] to our relaxation method. The proof closely follows the one given by Scholtes in [23] and is

included here for the completeness. We first state two technical lemmas. Lemma 3.1 states that the NLP-LICQ at a feasible point carries over to all feasible points in a sufficiently small neighborhood. Lemma 3.2 extends similar results to MPECs.

Lemma 3.1. Consider the nonlinear program

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & h(x) = 0, \\ & g(x) \leq 0, \end{array} \tag{14}$$

where $f : R^n \to R^1$, $h : R^n \to R^l$ and $g : R^n \to R^m$ are twice continuously differentiable functions. If the NLP-LICQ holds at a feasible point $\bar{x}$ of (14), then there exists a neighborhood $\mathcal{N}(\bar{x})$ such that the NLP-LICQ holds at every feasible point $x \in \mathcal{N}(\bar{x})$.

Proof. Let $A(\bar{x})$ be the Jacobian matrix of active constraints at $\bar{x}$ of (14). Since NLP-LICQ holds at $\bar{x}$, the rows of $A(\bar{x})$ are linearly independent. Then, there exists a basis $A_{\bullet\beta}(\bar{x})$ in $A(\bar{x})$ and the determinant of $A_{\bullet\beta}(\bar{x})$, denoted as $det(A_{\bullet\beta}(\bar{x}))$, is nonzero. Since $det(A_{\bullet\beta}(x))$ is a continuous function of x, it follows that there exists a neighborhood $\mathcal{N}(\bar{x})$ such that for all feasible points x of (14) in $\mathcal{N}(\bar{x})$:

$$\begin{aligned} det(A_{\bullet\beta}(x)) &\neq 0, \\ \mathcal{I}_g(x) &\subseteq \mathcal{I}_g(\bar{x}), \\ \mathcal{I}_h(x) &\subseteq \mathcal{I}_h(\bar{x}). \end{aligned} \tag{15}$$

This further implies that for every such x the gradient vectors of the active constraints in (14) are linearly independent. Hence, NLP-LICQ holds at every feasible point $x \in \mathcal{N}(\bar{x})$. □

Given a feasible point $\bar{x}$ of Reg(t) (13), we define the following index sets of active constraints:

$$\begin{aligned} \mathcal{I}_g(\bar{x}, t) &:= \{i \,|\, g_i(\bar{x}; a_t^g) = 0\}, \\ \mathcal{I}_h(\bar{x}, t) &:= \{i \,|\, h_i(\bar{x}; a_t^h) = 0\}, \\ \mathcal{I}_G(\bar{x}, t) &:= \{i \,|\, G_i(\bar{x}; a_t^G) = 0\}, \\ \mathcal{I}_H(\bar{x}, t) &:= \{i \,|\, H_i(\bar{x}; a_t^H) = 0\}, \\ \mathcal{I}_{GH}(\bar{x}, t) &:= \{i \,|\, G_i(\bar{x}; a_t^H) H_i(\bar{x}; a_t^H) = t\}. \end{aligned} \tag{16}$$

Lemma 3.2. If the MPEC-LICQ holds at the feasible point $\bar{x}$ of the MPEC (10), then there exists a neighborhood $\mathcal{N}(\bar{x})$ and a scalar $\bar{t} > 0$ such that for every $t \in (0, \bar{t}\,)$, the NLP-LICQ holds at every feasible point $x \in \mathcal{N}(\bar{x})$ of Reg(t) (13).

Proof. This follows from lemma 3.1 and the following relations on the index sets of active

constraints:

$$\begin{aligned} \mathcal{I}_g(x,t) &\subseteq \mathcal{I}_g(\bar{x}), \\ \mathcal{I}_h(x,t) &\subseteq \mathcal{I}_h(\bar{x}), \\ \mathcal{I}_G(x,t) \cup \mathcal{I}_H(x,t) \cup \mathcal{I}_{GH}(x,t) &\subseteq \mathcal{I}_G(\bar{x}) \cup \mathcal{I}_H(\bar{x}), \\ \mathcal{I}_G(x,t) \cap \mathcal{I}_{GH}(x,t) &= \emptyset, \\ \mathcal{I}_H(x,t) \cap \mathcal{I}_{GH}(x,t) &= \emptyset, \end{aligned} \tag{17}$$

which hold for all x in a sufficiently small neighborhood $\mathcal{N}(\bar{x})$ and all $t \in (0, \bar{t})$ for sufficiently small $\bar{t} > 0$.

For every feasible x of (13) in $\mathcal{N}(\bar{x})$, by Lemma 3.1 and (17), the system

$$\begin{aligned} &\sum_{i \in \mathcal{I}_g(x,t)} \lambda_i^g \nabla g_i(x) + \sum_{i \in \mathcal{I}_h(x,t)} \lambda_i^h \nabla g_i(x) + \sum_{i \in \mathcal{I}_G(x,t)} \lambda_i^G \nabla G_i(x) + \sum_{i \in \mathcal{I}_H(x,t)} \lambda_i^H \nabla H_i(x) \\ &+ \sum_{i \in \mathcal{I}_{GH}(x,t)} [(\lambda_i^{GH} H_i(x; a_t^H)) \nabla G_i(x) + (\lambda_i^{GH} G_i(x; a_t^G)) \nabla H_i(x)] = 0 \end{aligned} \tag{18}$$

implies that $\lambda_i^g = \lambda_i^h = \lambda_i^G = \lambda_i^H = \lambda_i^{GH} G_i(x; a_t^G) = \lambda_i^{GH} H_i(x; a_t^H) = 0$. Since

$$G_i(x; a_t^G) H_i(x; a_t^H) = t,$$

we have

$$G_i(x; a_t^G) > 0, \quad H_i(x; a_t^H) > 0,$$

and thus, $\lambda_i^{GH} = 0$. This proves that the NLP-LICQ holds at x. □

Theorem 3.3. Let $\{t_\nu\}$ be a sequence of positive scalars tending to zero as $\nu \to \infty$, and let x_ν be a stationary point of $\mathrm{Reg}(t_\nu)$ converging to $\bar{x}$. Suppose the MPEC-LICQ holds at $\bar{x}$. Then

(i) The point $\bar{x}$ is a C-stationary point of the MPEC (1).

(ii) If, for each ν, the point x_ν also satisfies second-order necessary optimality conditions for $\mathrm{Reg}(t_\nu)$, then $\bar{x}$ is an M-stationary point of the MPEC (1).

(iii) Moreover, if the ULSC assumption holds at $\bar{x}$, then $\bar{x}$ is a B-stationary point of the MPEC (1).

Proof. First, by (11) and (12), it is easy to see that $\bar{x}$ is a feasible point of the MPEC (1). Let $\lambda_\nu = (\lambda_\nu^g, \lambda_\nu^h, \lambda_\nu^G, \lambda_\nu^H, \lambda_\nu^{GH})$ be the vector of Lagrangian multipliers of $\mathrm{Reg}(t_\nu)$ (13) at the stationary point x_ν, and let

$$\mathcal{I}_0 = \{i \,|\, i \in \mathcal{I}_{GH}(x_\nu, t_\nu) \text{ for infinitely many } \nu\}.$$

Since MPEC-LICQ holds at $\bar{x}$, the multiplier vector λ_ν is unique in the following KKT system of $\mathrm{Reg}(t_\nu)$ (13) for sufficiently small t_ν:

$$\begin{aligned}
\nabla f(x_\nu; a^f_{t_\nu}) + \nabla g(x_\nu; a^g_{t_\nu})^{\mathrm{T}}\lambda^g_\nu + \nabla h(x_\nu; a^h_{t_\nu})^{\mathrm{T}}\lambda^h_\nu & \\
- \nabla G(x_\nu; a^G_{t_\nu})^{\mathrm{T}}\lambda^G_\nu - \nabla H(x_\nu; a^H_{t_\nu})^{\mathrm{T}}\lambda^H_\nu & \\
+ \nabla G(x_\nu; a^G_{t_\nu})^{\mathrm{T}}[H(x_\nu; a^H_{t_\nu}) \circ \lambda^{GH}_\nu] + \nabla H(x_\nu; a^H_{t_\nu})^{\mathrm{T}}[G(x_\nu; a^G_{t_\nu}) \circ \lambda^{GH}_\nu] &= 0, \\
h(x_\nu; a^h_{t_\nu}) &= 0, \\
g(x_\nu; a^g_{t_\nu}) \le 0,\ \lambda^g_\nu \ge 0,\ g(x_\nu; a^g_{t_\nu})^{\mathrm{T}}\lambda^g_\nu &= 0, \\
G(x_\nu; a^G_{t_\nu}) \le 0,\ \lambda^G_\nu \ge 0,\ G(x_\nu; a^G_{t_\nu})^{\mathrm{T}}\lambda^G_\nu &= 0, \\
H(x_\nu; a^H_{t_\nu}) \le 0,\ \lambda^H_\nu \ge 0,\ H(x_\nu; a^H_{t_\nu})^{\mathrm{T}}\lambda^H_\nu &= 0, \\
G(x_\nu; a^G_{t_\nu}) \circ H(x_\nu; a^H_{t_\nu}) - t_\nu\, e \le 0,\ \lambda^{GH}_\nu &\ge 0, \\
[G(x_\nu; a^G_{t_\nu}) \circ H(x_\nu; a^H_{t_\nu}) - t_\nu\, e]^{\mathrm{T}}\lambda^{GH}_\nu &= 0.
\end{aligned} \tag{19}$$

(i) Define $\tilde{\lambda}^G_{i,\nu} = -\lambda^{GH}_{i,\nu} H_i(x_\nu; a^H_{t_\nu})$, and $\tilde{\lambda}^H_{i,\nu} = -\lambda^{GH}_{i,\nu} G_i(x_\nu; a^G_{t_\nu})$, for $i \in \mathcal{I}_{GH}(x_\nu, t_\nu)$ and rewrite the first equation in (19) as

$$\begin{aligned}
-\nabla f(x_\nu; a^f_{t_\nu}) \;=\; & \sum_{i \in \mathcal{I}_g(x_\nu)} \lambda^g_{i,\nu} \nabla g_i(x_\nu; a^g_{t_\nu}) + \sum_{i \in \mathcal{I}_h(x_\nu)} \lambda^h_{i,\nu} \nabla h_i(x_\nu; a^h_{t_\nu}) \\
& - \sum_{i \in \mathcal{I}_G(x_\nu, t_\nu)} \lambda^G_{i,\nu} \nabla G_i(x_\nu; a^G_{t_\nu}) - \sum_{i \in \mathcal{I}_H(x_\nu, t_\nu)} \lambda^H_{i,\nu} \nabla H_i(x_\nu; a^H_{t_\nu}) \\
& - \sum_{i \in \mathcal{I}_{GH}(x_\nu, t_\nu) \cap \mathcal{I}^c_G(\bar{x})} \tilde{\lambda}^H_{i,\nu} \left[\nabla H_i(x_\nu; a^H_{t_\nu}) + \frac{G_i(x_\nu; a^G_{t_\nu})}{H_i(x_\nu; a^H_{t_\nu})} \nabla G_i(x_\nu; a^G_{t_\nu}) \right] \\
& - \sum_{i \in \mathcal{I}_{GH}(x_\nu, t_\nu) \cap \mathcal{I}^c_H(\bar{x})} \tilde{\lambda}^G_{i,\nu} \left[\nabla G_i(x_\nu; a^G_{t_\nu}) + \frac{H_i(x_\nu; a^H_{t_\nu})}{G_i(x_\nu; a^G_{t_\nu})} \nabla H_i(x_\nu; a^H_{t_\nu}) \right] \\
& - \sum_{i \in \mathcal{I}_{GH}(x_\nu, t_\nu) \cap \mathcal{I}_G(\bar{x}) \cap \mathcal{I}_H(\bar{x})} \left[\tilde{\lambda}^G_{i,\nu} \nabla G_i(x_\nu; a^G_{t_\nu}) + \tilde{\lambda}^H_{i,\nu} \nabla H_i(x_\nu; a^H_{t_\nu}) \right].
\end{aligned} \tag{20}$$

For every sufficient large ν, we construct a matrix $A(x_\nu)$ with rows being the transpose

of the following vectors:

$$\begin{aligned}
\nabla g_i(x_\nu; a^g_{t_\nu}), &\quad i \in \mathcal{I}_g(\bar{x}),\\
\nabla h_i(x_\nu; a^h_{t_\nu}), &\quad i \in \mathcal{I}_h(\bar{x}),\\
-\nabla G_i(x_\nu; a^G_{t_\nu}), &\quad i \in \mathcal{I}_G(\bar{x}) \setminus (\mathcal{I}_{GH}(x_\nu, t_\nu) \cap \mathcal{I}^c_H(\bar{x})),\\
-\nabla G_i(x_\nu; a^G_{t_\nu}) - \frac{H_i(x_\nu; a^H_{t_\nu})}{G_i(x_\nu; a^G_{t_\nu})} \nabla H_i(x_\nu; a^H_{t_\nu}), &\quad i \in \mathcal{I}_{GH}(x_\nu, t_\nu) \cap \mathcal{I}^c_H(\bar{x}),\\
-\nabla H_i(x_\nu; a^H_{t_\nu}), &\quad i \in \mathcal{I}_H(\bar{x}) \setminus (\mathcal{I}_{GH}(x_\nu, t_\nu) \cap \mathcal{I}^c_G(\bar{x})),\\
-\nabla H_i(x_\nu; a^H_{t_\nu}) - \frac{G_i(x_\nu; a^G_{t_\nu})}{H_i(x_\nu; a^H_{t_\nu})} \nabla G_i(x_\nu; a^G_{t_\nu}), &\quad i \in \mathcal{I}_H(\bar{x}) \setminus (\mathcal{I}_{GH}(x_\nu, t_\nu) \cap \mathcal{I}^c_G(\bar{x})).
\end{aligned}$$

Then (20) can be represented as an enlarged system of equations $A(x_\nu)^{\mathrm{T}} y_\nu = -\nabla f(x_\nu)$ with some components in y_ν set to 0. The matrix $A(x_\nu)$ converges to the matrix $A(\bar{x})$ with linearly independent rows

$$\begin{aligned}
\nabla g_i(\bar{x}; \bar{a}^g), &\quad i \in \mathcal{I}_g(\bar{x}),\\
\nabla h_i(\bar{x}; \bar{a}^h), &\quad i \in \mathcal{I}_h(\bar{x}),\\
\nabla G_i(\bar{x}; \bar{a}^G), &\quad i \in \mathcal{I}_G(\bar{x}),\\
\nabla H_i(\bar{x}; \bar{a}^H), &\quad i \in \mathcal{I}_H(\bar{x}).
\end{aligned}$$

It follows that y_ν converges to a unique vector, $\bar{\lambda} = (\bar{\lambda}^g, \bar{\lambda}^h, \bar{\lambda}^G, \bar{\lambda}^H)$, with

$$\begin{array}{lrcllrcl}
 & \lim_{\nu\to\infty} \lambda^g_{i,\nu} & = & \bar{\lambda}^g_i \geq 0, & \lim_{\nu\to\infty} \lambda^h_{i,\nu} & = & \bar{\lambda}^h_i,\\
i \notin \mathcal{I}_0: & \lim_{\nu\to\infty} \lambda^G_{i,\nu} & = & \bar{\lambda}^G_i \geq 0, & \lim_{\nu\to\infty} \lambda^H_{i,\nu} & = & \bar{\lambda}^H_i \geq 0,\\
i \in \mathcal{I}_0: & -\lim_{\nu\to\infty} \lambda^{GH}_{i,\nu} H_i(x_\nu; a^H_{t_\nu}) & = & \bar{\lambda}^G_i \leq 0, & -\lim_{\nu\to\infty} \lambda^{GH}_{i,\nu} G_i(x_\nu; a^G_{t_\nu}) & = & \bar{\lambda}^H_i \leq 0.
\end{array} \tag{21}$$

This completes the proof of (i).

(ii) Suppose $\bar{x}$ is not an M-stationary point of the MPEC (1). Then there exists an index $j \in \mathcal{I}_{GH}(\bar{x})$ such that

$$-\lim_{\nu\to\infty} \lambda^{GH}_{j,\nu} H_j(x_\nu; a^H_{t_\nu}) = \bar{\lambda}^G_j < 0, \quad -\lim_{\nu\to\infty} \lambda^{GH}_{j,\nu} G_j(x_\nu; a^G_{t_\nu}) = \bar{\lambda}^H_j \leq 0.$$

From (i), this further implies that $j \in \mathcal{I}_0$ and $G_j(x_\nu; a^G_{t_\nu}) H_j(x_\nu; a^H_{t_\nu}) = t_\nu$ for every sufficiently large ν.

For every ν, we construct a matrix $B(x_\nu)$ with rows being the transpose of the following

vectors

$$\begin{aligned}
\nabla g_i(x_\nu; a^g_{t_\nu}), &\quad i \in \mathcal{I}_g(\bar{x}),\\
\nabla h_i(x_\nu; a^h_{t_\nu}), &\quad i \in \mathcal{I}_h(\bar{x}),\\
\nabla G_i(x_\nu; a^G_{t_\nu}), &\quad i \in \mathcal{I}_G(\bar{x}),\\
\nabla H_i(x_\nu; a^H_{t_\nu}), &\quad i \in \mathcal{I}_H(\bar{x}).
\end{aligned}$$

The matrix $B(x_\nu)$ converges to the matrix $B(\bar{x})$ with linearly independent rows

$$\begin{aligned}
\nabla g_i(\bar{x}; \bar{a}^g), &\quad i \in \mathcal{I}_g(\bar{x}),\\
\nabla h_i(\bar{x}; \bar{a}^h), &\quad i \in \mathcal{I}_h(\bar{x}),\\
\nabla G_i(\bar{x}; \bar{a}^G), &\quad i \in \mathcal{I}_G(\bar{x}),\\
\nabla H_i(\bar{x}; \bar{a}^H), &\quad i \in \mathcal{I}_H(\bar{x}).
\end{aligned}$$

Since the MPEC-LICQ holds at $\bar{x}$, it follows that the rows in the matrix $B(x_\nu)$ are linearly independent for every sufficiently large ν. Consequently, the following system has no solutions for ν large enough:

$$B(x_\nu)^{\mathrm{T}} z_\nu = 0, \quad z_\nu \neq 0. \tag{22}$$

By Gale's theorem of alternatives (page 34 in [15]), the following system has a solution d_ν for every sufficiently large ν:

$$\begin{aligned}
\nabla g_i(x_\nu; a^g_{t_\nu})^{\mathrm{T}} d_\nu &= 0, \qquad i \in \mathcal{I}_g(\bar{x}),\\
\nabla h_i(x_\nu; a^h_{t_\nu})^{\mathrm{T}} d_\nu &= 0, \qquad i \in \mathcal{I}_h(\bar{x}),\\
\nabla G_i(x_\nu; a^G_{t_\nu})^{\mathrm{T}} d_\nu &= 0, \qquad i \in \mathcal{I}_G(\bar{x}),\ i \neq j,\\
\nabla H_i(x_\nu; a^H_{t_\nu})^{\mathrm{T}} d_\nu &= 0, \qquad i \in \mathcal{I}_H(\bar{x}),\ i \neq j,\\
\nabla G_j(x_\nu; a^G_{t_\nu})^{\mathrm{T}} d_\nu &= 1,\\
\nabla H_j(x_\nu; a^H_{t_\nu})^{\mathrm{T}} d_\nu &= -\frac{H_j(x_\nu; a^H_{t_\nu})}{G_j(x_\nu; a^G_{t_\nu})},
\end{aligned}$$

and we represent the system as

$$B(x_\nu)\, d_\nu = b_\nu. \tag{23}$$

Similarly, the following system has a solution $\bar{d}$:

$$\begin{aligned}
\nabla g_i(\bar{x}; \bar{a}^g)^{\mathrm{T}} \bar{d} &= 0, \qquad i \in \mathcal{I}_g(\bar{x}),\\
\nabla h_i(\bar{x}; \bar{a}^h)^{\mathrm{T}} \bar{d} &= 0, \qquad i \in \mathcal{I}_h(\bar{x}),\\
\nabla G_i(\bar{x}; \bar{a}^G)^{\mathrm{T}} \bar{d} &= 0, \qquad i \in \mathcal{I}_G(\bar{x}),\ i \neq j,\\
\nabla H_i(\bar{x}; \bar{a}^H)^{\mathrm{T}} \bar{d} &= 0, \qquad i \in \mathcal{I}_H(\bar{x}),\ i \neq j,\\
\nabla G_j(\bar{x}; \bar{a}^G)^{\mathrm{T}} \bar{d} &= 1,\\
\nabla H_j(\bar{x}; \bar{a}^G)^{\mathrm{T}} \bar{d} &= -\bar{\lambda}^G / \bar{\lambda}^H,
\end{aligned}$$

and we represent the system as

$$B(\bar{x})\,\bar{d} = \bar{b}. \tag{24}$$

Below, we construct a bounded sequence $\{d_\nu\}$ converging to $\bar{d}$. Without loss of generality, we can assume that there exists an index set β such that $B_{\bullet\beta}(\bar{x})$ is a basis in $B(\bar{x})$ and $B_{\bullet\beta}(x_\nu)$ is a basis in $B(x_\nu)$ for every sufficient large ν. Furthermore, the vector $\bar{d}$ is a basic solution of (24) associated with the basis $B_{\bullet\beta}(\bar{x})$ with $\bar{d}_\beta$ satisfying

$$B_{\bullet\beta}(\bar{x})\bar{d}_\beta = \bar{b}$$

and the rest of the components in $\bar{d}$ being 0.

Similarly, for every sufficiently large ν, the vector d_ν is a basic solution of (23) associated with the basis $B_{\bullet\beta}(x_\nu)$ with $(d_\nu)_\beta$ satisfying

$$B_{\bullet\beta}(x_\nu)(d_\nu)_\beta = b_\nu$$

and the rest of the components in d_ν being 0.

From (21), it is clear that $H_j(x_\nu; a^H_{t_\nu})/G_j(x_\nu; a^G_{t_\nu}) \to \bar{\lambda}^G/\bar{\lambda}^H$, and hence, $b_\nu \to \bar{b}$ as $\nu \to \infty$. With $B(x_\nu)$ converging to $B(\bar{x})$, it follows that the sequence $\{d_\nu\}$ is bounded and $d_\nu \to \bar{d}$ as $\nu \to \infty$.

It is easy to see that d_ν is a critical direction of $\mathrm{Reg}(t_\nu)$ (13) at x_ν for ν large enough. If the constraint $G_j(x; a^G_{t_\nu})H_j(x; a^H_{t_\nu}) \le t_\nu$ is active at x_ν, we examine the term associated with this constraint in the Lagrangian function of $\mathrm{Reg}(t_\nu)$ for the second-order necessary optimality conditions. In particular,

$$\begin{aligned}
&\lambda^{GH}_{j,\nu}\, d_\nu^{\mathrm{T}}\nabla^2 \Big(G_j(x_\nu; a^G_{t_\nu})H_j(x_\nu; a^H_{t_\nu}) - t_\nu\Big)\, d_\nu \\
&= \lambda^{GH}_{j,\nu} H_j(x_\nu; a^H_{t_\nu})\, d_\nu^{\mathrm{T}}\nabla^2 G_j(x_\nu; a^G_{t_\nu})\, d_\nu + \lambda^{GH}_{j,\nu} G_j(x_\nu; a^G_{t_\nu})\, d_\nu^{\mathrm{T}}\nabla^2 H_j(x_\nu; a^H_{t_\nu})\, d_\nu \\
&\quad - \lambda^{GH}_{j,\nu} H_j(x_\nu; a^H_{t_\nu}) \frac{2}{G_j(x_\nu; a^G_{t_\nu})}.
\end{aligned}$$

While the first two terms in the above equation are bounded, the third term

$$-\lambda^{GH}_{j,\nu} H_j(x_\nu; a^H_{t_\nu}) \frac{2}{G_j(x_\nu; a^G_{t_\nu})} \to -\infty, \quad \text{as } \nu \to \infty,$$

since $\lambda^{GH}_{j,\nu} H_j(x_\nu; a^H_{t_\nu}) \to -\bar{\lambda}_j > 0$ and $G_j(x_\nu; a^G_{t_\nu}) \to 0^+$. It is easy to check that all other terms in $d_\nu^{\mathrm{T}}\nabla^2\mathcal{L}(x_\nu, \lambda_\nu)d_\nu$ are bounded, and hence, the second-order necessary optimality condition of $\mathrm{Reg}(t_\nu)$ (13) fails at x_ν for sufficiently large ν.

(iii) Since, from (ii), $\bar{x}$ is an M-stationary point and the ULSC holds at $\bar{x}$, it follows that $\bar{x}$ is a strongly stationary point, and hence, a B-stationary point. □

4 Equilibrium Problems with Equilibrium Constraints

4.1 Formulation and stationarity conditions

An EPEC is a problem of finding an equilibrium point that solves several MPECs simultaneously. Since practical applications of EPEC models often arise from multi-leader-follower game settings, we consider the EPEC consisting of MPECs with shared decision variables and shared equilibrium constraints. In particular, we assume the EPEC consists of K MPECs, and for each $k = 1, \ldots, K$, the k-th MPEC has the following form with independent decision variables $x^k \in R^{n_k}$ and shared decision variables $y \in R^{n_0}$:

$$\begin{array}{ll} \text{minimize} & f^k(x^k, y; \bar{x}^{-k}) \\ \text{subject to} & g^k(x^k, y; \bar{x}^{-k}) \leq 0, \quad h^k(x^k, y; \bar{x}^{-k}) = 0, \\ & 0 \leq G(x^k, y; \bar{x}^{-k}) \perp H(x^k, y; \bar{x}^{-k}) \geq 0, \end{array} \tag{25}$$

where $x^{-k} = (x^j)_{j=1}^K \setminus x^k$, and $f^k : R^n \to R,\ g^k : R^n \to R^{p_k},\ h^k : R^n \to R^{q_k}$, $G : R^n \to R^m$ and $H : R^n \to R^m$ are twice continuously differentiable functions in both $x = (x^k)_{k=1}^K$ and y, with $n = \sum_{k=0}^K n_k$. The notation $\bar{x}^{-k}$ means that $x^{-k} \in R^{n-n_k-n_0}$ is not a variable but a fixed vector. This implies that we can view the above MPEC, denoted by MPEC($\bar{x}^{-k}$), as being parameterized by $\bar{x}^{-k}$. Given $\bar{x}^{-k}$, we assume the solution set of the k-th MPEC is nonempty and denote it by SOL(MPEC($\bar{x}^{-k}$)). Notice that in the above formulation, each MPEC shares the same equilibrium constraints, represented by the complementarity system

$$0 \leq G(x, y) \perp H(x, y) \geq 0.$$

The EPEC, associated with K MPECs defined as above, is to find a Nash equilibrium $(x^*, y^*) \in R^n$ such that

$$(x^{k*}, y^*) \in \text{SOL}(\text{MPEC}(x^{-k*})) \quad \forall\ k = 1, \ldots, K. \tag{26}$$

In a recent paper, Mordukhovich [16] studies the necessary optimality conditions of EPECs via multiobjective optimization. Following Hu [11], we define stationary conditions for EPECs by applying those for MPECs.

Definition 4.1. We call a vector (x^*, y^*) a *B-stationary* (*strongly stationary*, *M-stationary*, *C-stationary*, *weakly stationary*) point of the EPEC (26) if for each $k = 1, \ldots, K$, (x^{k*}, y^*) is a B-stationary (strongly stationary, M-stationary, C-stationary, weakly stationary) point for the MPEC(x^{-k*}).

Theorem 4.2. Let (x^*, y^*) be a (possibly local) equilibrium point of the EPEC (26). If for each $k = 1, \ldots, K$, the MPEC-LICQ holds at (x^{k*}, y^*) for MPEC(x^{-k*}) (25), then (x^*, y^*)

is an EPEC strongly stationary point. In particular, there exist vectors $\lambda^* = (\lambda^{1*}, \ldots, \lambda^{K*})$ with $\lambda^{k*} = (\lambda^{g,k*}, \lambda^{h,k*}, \lambda^{G,k*}, \lambda^{H,k*}, \lambda^{GH,k*})$ such that (x^*, y^*, λ^*) solves the system

$$\left.\begin{array}{rl}
\nabla_{x^k} f^k(x^k, y; x^{-k}) + \nabla_{x^k} g^k(x^k, y; x^{-k})^{\mathrm{T}} \lambda^{g,k} + \nabla_{x^k} h^k(x^k, y; x^{-k})^{\mathrm{T}} \lambda^{h,k} & \\
-\nabla_{x^k} G(x^k, y; x^{-k})^{\mathrm{T}} \lambda^{G,k} - \nabla_{x^k} H(x^k, y; x^{-k})^{\mathrm{T}} \lambda^{H,k} & \\
+\nabla_{x^k} G(x^k, y; x^{-k})^{\mathrm{T}} [H(x^k, y; x^{-k}) \circ \lambda^{GH,k}] & \\
+\nabla_{x^k} H(x^k, y; x^{-k})^{\mathrm{T}} [G(x^k, y; x^{-k}) \circ \lambda^{GH,k}] & = 0 \\
\nabla_y f^k(x^k, y; x^{-k}) + \nabla_y g^k(x^k, y; x^{-k})^{\mathrm{T}} \lambda^{g,k} + \nabla_y h^k(x^k, y; x^{-k})^{\mathrm{T}} \lambda^{h,k} & \\
-\nabla_y G(x^k, y; x^{-k})^{\mathrm{T}} \lambda^{G,k} - \nabla_y H(x^k, y; x^{-k})^{\mathrm{T}} \lambda^{H,k} & \\
+\nabla_y G(x^k, y; x^{-k})^{\mathrm{T}} [H(x^k, y; x^{-k}) \circ \lambda^{GH,k}] & \\
+\nabla_y H(x^k, y; x^{-k})^{\mathrm{T}} [G(x^k, y; x^{-k}) \circ \lambda^{GH,k}] & = 0 \\
h^k(x^k, y; x^{-k}) = 0 & \\
0 \geq g^k(x^k, y; x^{-k}) \quad \perp \quad \lambda^{g,k} \geq 0, & \\
0 \leq G(x^k, y; x^{-k}) \quad \perp \quad \lambda^{G,k} \geq 0, & \\
0 \leq H(x^k, y; x^{-k}) \quad \perp \quad \lambda^{H,k} \geq 0, & \\
0 \leq -G(x^k, y; x^{-k}) \circ H(x^k, y; x^{-k}) \quad \perp \quad \lambda^{GH,k} \geq 0, &
\end{array}\right\} \quad (27)$$

$$k = 1, \ldots, K.$$

Conversely, if (x^*, y^*, λ^*) is a solution of the above system (27), then (x^*, y^*) is a B-stationary point of the EPEC(26).

Proof. Since (x^*, y^*) is a (possibly local) equilibrium point of the EPEC (26), it follows that for each $k = 1, \ldots, K$, the point (x^{k*}, y^*) is a (local) minimizer of the MPEC(x^{-k*}) (25). By applying Theorem 2.7 and Theorem 2.8 to the MPEC(x^{-k*}) (25) for $k = 1, \ldots, K$, we can show that there exists a vector $\lambda^{k*} = (\lambda^{g,k*}, \lambda^{h,k*}, \lambda^{G,k*}, \lambda^{H,k*}, \lambda^{GH,k*})$ such that $(x^{k*}, y^*, \lambda^{k*})$ satisfies the conditions in the system (27) for each $k = 1, \ldots, K$. Let $\lambda^* = (\lambda^{1*}, \ldots, \lambda^{K*})$. Then, the vector (x^*, y^*, λ^*) is a solution of the system (27). Conversely, by Theorem 2.8, it is easy to check that for each $k = 1, \ldots, K$, the vector (x^{k*}, y^*) is a strongly stationary point, and hence, B-stationary point (by Theorem 2.6) for the MPEC(x^{-k*}) (25). As a result, the vector (x^*, y^*) is a B-stationary point of the EPEC (26). □

4.2 Algorithms for Solving EPECs

To date, algorithms specifically designed for solving EPECs have not been developed in the literature. The approaches used by researchers in engineering fields to solve EPECs fall into the category of *Diagonalization methods* [2, 3, 9, 11, 17], which mainly rely on NLP solvers, or more appropriately, MPEC algorithms to solve one MPEC at a time and cyclicly repeat the same procedure for every MPEC until an equilibrium point is found. In the remainder of this section, we first describe two types of diagonalization methods: nonlinear Jacobi and nonlinear Gauss-Seidel, and briefly discuss the convergence of the diagonalization methods. We then present a new method called *sequential nonlinear complementarity (SNCP) algorithm* for solving EPECs. This new method is based on simultaneously relaxing the complementarity constraints in each MPEC, and solves EPECs by solving a sequence of nonlinear complementarity problems. We also establish the convergence of the SNCP algorithm.

Diagonalization methods

Diagonalization methods [5, 18] were originally proposed to solve variational inequality problems. In [8], Harker applied a diagonalization (or *nonlinear Jacobi*) algorithm, to find a solution to a variational inequality formulation of the *Nash equilibrium* problem in an oligopolistic market.

Because of their conceptual simplicity and ease of implementation, diagonalization methods using NLP solvers have been natural choices for engineers and applied economists to solve EPEC models [2, 3] arising in deregulated electricity markets. In [9] and [17], MPEC algorithms (penalty interior point algorithm in the former reference and smoothing algorithm in the latter) are used in diagonalization methods to solve EPEC models. Below, we describe two diagonalization methods: nonlinear Jacobi and nonlinear Gauss-Seidel. The framework of the diagonalization methods presented here follows the one given in Hu [11].

The nonlinear Jacobi method for the EPEC (26) is described as follows:

Step 0. *Initialization.* We are given a starting point $(x^{(0)}, y^{(0)}) = (x^{1,(0)}, \ldots, x^{K,(0)}, y^{(0)})$, the maximum number of outer iterations J, and an accuracy tolerance $\varepsilon > 0$.

Step 1. *Loop over every MPEC.* Suppose the current iteration point of (x, y) is $(x^{(j)}, y^{(j)})$. For each $k = 1, \ldots, K$, the MPEC($\bar{x}^{-k,(j)}$) is solved (using NLP solvers or MPEC algorithms) while fixing $\bar{x}^{-k,(j)} = (x^{1,(j)}, \ldots, x^{k-1,(j)}, x^{k+1,(j)}, \ldots, x^{K,(j)})$. Denote x-part of the optimal solution of MPEC($\bar{x}^{-k}$) by $x^{k,(j+1)}$.

Step 2. *Check convergence.* Let $(x^{(j+1)}) = (x^{1,(j+1)}, \ldots, x^{K,(j+1)})$. If $j < J$, then increase j by one and repeat (Step 1.). Otherwise, *stop* and *check the accuracy tolerance*: if $\|x^{k,(j+1)} - x^{k,(j)}\| < \varepsilon$ for $k = 1, \ldots, K$, then accept and report the solution (x^J, y^J) ; otherwise, output "*No equilibrium point found*".

Note that the nonlinear Jacobi method does not use the most recently available information when computing $x^{k,(j+1)}$. For example, $x^{1,(j)}$ is used in the calculation of $x^{2,(j+1)}$, even though the vector, $x^{1,(j+1)}$ is known. If we revise the nonlinear Jacobi method so that we always use the new information, then we have another diagonalization method, the Gauss–Seidel method. Hence, the framework of the nonlinear Gauss–Seidel method for the EPEC (26) is the same as nonlinear Jacobi, except that in Step 1., we have

$$\bar{x}^{-k,(j)} = (x^{1,(j+1)}, \ldots, x^{k-1,(j+1)}, x^{k+1,(j)}, \ldots, x^{K,(j)}).$$

The multi-firm algorithm proposed in [9] belongs to this category.

To solve each MPEC in Step 1 of diagonalization methods, one can solve the equivalent NLP (8) suggested by Fletcher and Leyffer [6] using off-the-shelf NLP solvers. For each $k = 1, \ldots, K$, the equivalent NLP formulation of MPEC($\bar{x}^{-k,(j)}$) is

$$\begin{array}{rrcl} \text{minimize} & f^k(x^k, y; \bar{x}^{-k,(j)}) & & \\ \text{subject to} & g^k(x^k, y; \bar{x}^{-k,(j)}) & \leq & 0, \\ & h^k(x^k, y; \bar{x}^{-k,(j)}) & = & 0, \\ & G(x^k, y; \bar{x}^{-k,(j)}) & \geq & 0, \\ & H(x^k, y; \bar{x}^{-k,(j)}) & \geq & 0, \\ G(x^k, y; \bar{x}^{-k,(j)}) \circ & H(x^k, y; \bar{x}^{-k,(j)}) & \leq & 0. \end{array} \tag{28}$$

We denote the above equivalent NLP of the k-th MPEC by NLP^k($\bar{x}^{-k,(j)}$).

The following theorem states the convergence of diagonalization methods based on solving equivalent NLPs.

Theorem 4.3. Let $\{(x^{(j)}, y^{(j)})\}$ be a sequence of solutions generated by a diagonalization (nonlinear Jacobi or nonlinear Gauss-Seidel) method, in which each MPEC is reformulated and solved as an equivalent NLP (28). Suppose the sequence $\{(x^{(j)}, y^{(j)})\}$ converges to (x^*, y^*) as $j \to \infty$. If, for each $k = 1, \ldots, K$, the MPEC-LICQ holds at (x^{k*}, y^*) for MPEC(x^{-k*}), then (x^*, y^*) is B-stationary for the EPEC (26).

Proof. From Theorem 2.6 applied to the MPEC(x^{-k*}) for each $k = 1, \ldots, K$, the point (x^{k*}, y^*) is a B-stationary point, and hence, the point (x^*, y^*) is a B-stationary point for the EPEC (26). □

Sequential NCP method

We propose a new method for solving EPECs. Instead of solving an EPEC by cyclicly using an MPEC-based approach, our approach simultaneously relaxes the complementarity system in each MPEC($\bar{x}^{-k}$) to

$$\begin{array}{l} G(x^k, y; \bar{x}^{-k}) \geq 0, \quad H(x^k, y; \bar{x}^{-k}) \geq 0, \\ G(x^k, y; \bar{x}^{-k}) \circ H(x^k, y; \bar{x}^{-k}) \leq te, \end{array} \tag{29}$$

and finds an equilibrium solution $(x^*(t), y^*(t)) = (x^{1*}(t), \ldots, x^{K*}(t), y^*(t))$ of the following regularized NLPs, denoted as $\mathrm{Reg}^k(\bar{x}^{-k}; t)$ for $t > 0$:

$$\left.\begin{array}{rrcll} \text{minimize} & f^k(x^k, y; \bar{x}^{-k}) & & & \\ \text{subject to} & g^k(x^k, y; \bar{x}^{-k}) & \leq & 0, & (\lambda^{g,k}) \\ & h^k(x^k, y; \bar{x}^{-k}) & = & 0, & (\lambda^{h,k}) \\ & G(x^k, y; \bar{x}^{-k}) & \geq & 0, & (\lambda^{G,k}) \\ & H(x^k, y; \bar{x}^{-k}) & \geq & 0, & (\lambda^{H,k}) \\ G(x^k, y; \bar{x}^{-k}) \circ H(x^k, y; \bar{x}^{-k}) & & \leq & te, & (\lambda^{GH,k}) \end{array}\right\} \quad k = 1, \ldots, K, \tag{30}$$

where, given $x^{-k*}(t)$ as the input parameter, $(x^{k*}(t), y^*(t))$ is a stationary point of the k-th regularized NLP, $\mathrm{Reg}^k(x^{-k*}(t); t)$.

Let $\mathcal{L}^k(x^k, y, \lambda^k; t)$ denote the Lagrangian function for the $\mathrm{Reg}^k(\bar{x}^{-k}; t)$. If $(x^*(t), y^*(t))$ is an equilibrium solution of (30) and LICQ holds at $(x^*(t), y^*(t))$ for each $\mathrm{Reg}^k(x^{-k*}(t); t)$, $k = 1, \ldots, K$, then $(x^*(t), y^*(t))$ is a solution of the following mixed nonlinear complementarity problem, obtained by combining the first-order KKT system of each $\mathrm{Reg}^k(x^{-k*}(t); t)$ in (30):

$$\left.\begin{array}{rcl} \nabla_{x^k} \mathcal{L}^k(x^k, y, \lambda^k; t) = 0 & & \\ \nabla_y \mathcal{L}^k(x^k, y, \lambda^k; t) = 0 & & \\ h^k(x^k, y; x^{-k}) = 0, & & \\ 0 \geq g^k(x^k, y; x^{-k}) & \perp & \lambda^{g,k} \geq 0, \\ 0 \leq G(x^k, y; x^{-k}) & \perp & \lambda^{G,k} \geq 0, \\ 0 \leq H(x^k, y; x^{-k}) & \perp & \lambda^{H,k} \geq 0, \\ 0 \leq te - G(x^k, y; x^{-k}) \circ H(x^k, y; x^{-k}) & \perp & \lambda^{GH,k} \geq 0, \end{array}\right\} \quad k = 1, \ldots, K. \tag{31}$$

For convenience, we denote the above system by $\mathrm{NCP}(t)$.

While Scholtes's regularized scheme for MPECs can be described as solving an MPEC by solving a sequence of nonlinear programs (NLPs), our method is to solve an EPEC by solving a sequence of nonlinear complementarity problems (NCPs).

The following theorem states the convergence of the sequential NCP algorithm.

Theorem 4.4. Let $\{t_\nu\}$ be a sequence of positive scalars tending to zero, and let (x_ν, y_ν) be a sequence of solutions to $\mathrm{NCP}(t_\nu)$ (31) converging to (x^*, y^*) as $t_\nu \to 0$. Furthermore, for each $k = 1, \ldots, K$ and for every ν, the point (x_ν^k, y_ν) satisfies the second-order optimality conditions of $\mathrm{Reg}^k(x_\nu^{-k}; t_\nu)$. If, for each $k = 1, \ldots, K$, the MPEC-LICQ and the ULSC hold at (x^{k*}, y^*) for $\mathrm{MPEC}(x^{-k*})$, then (x^*, y^*) is B-stationary for the EPEC (26).

Proof. For each $k = 1, \ldots, K$, the point (x^k_ν, y_ν) satisfies the second-order optimality condition of $\mathrm{Reg}^k(x^{-k}_\nu; t_\nu)$:

$$\begin{array}{ll} \text{minimize} & f^k(x^k, y; x^{-k}_\nu) \\ \text{subject to} & g^k(x^k, y; x^{-k}_\nu) \le 0, \\ & h^k(x^k, y; x^{-k}_\nu) = 0, \\ & G(x^k, y; x^{-k}_\nu) \ge 0, \\ & H(x^k, y; x^{-k}_\nu) \ge 0, \\ & G(x^k, y; x^{-k}_\nu) \circ H(x^k, y; x^{-k}_\nu) \le t_\nu\, e, \end{array}$$

and as $t_\nu \to 0^+$, we have $x^{-k}_\nu \to x^{-k*}$, and $(x^k_\nu, y_\nu) \to (x^{k*}, y^*)$. Since the MPEC-LICQ and the ULSC hold at (x^{k*}, y^*) for MPEC($\bar{x}^{-k}$), by Theorem 3.3, the point (x^{k*}, y^*) is B-stationary for each $k = 1, \ldots, K$. It follows that (x^*, y^*) is a B-stationary point for the EPEC (26). □

5 Implementation and Numerical Comparison

We have implemented the diagonalization methods and the sequential NCP algorithm on randomly generated EPEC test problems with known solutions. In this set of test problems, each EPEC consists of two MPECs in the following form:

$$\left.\begin{array}{ll} \text{minimize}_{(x^k, y)} & \frac{1}{2} \begin{bmatrix} x^k \\ y \end{bmatrix}^{\mathrm{T}} P^k \begin{bmatrix} x^k \\ y \end{bmatrix} + (c^k)^{\mathrm{T}} x^k + (d^k)^{\mathrm{T}} y \\ \text{subject to} & G^k x^k + H^k y + a^k \le 0, \\ & x^k \ge 0, \\ & 0 \le y \ \perp\ N^k x^k + \displaystyle\sum_{i=1, i\ne k}^{K} N^i \bar{x^i} + My + q \ge 0. \end{array}\right\} k = 1, 2, \qquad (32)$$

where $P^k \in R^{(n_k+m)\times(n_k+m)}, c^k \in R^{n_k}, d^k \in R^m, G^k \in R^{l_k \times n_k}, H^k \in R^{l_k \times m}, a^k \in R^{l_k}, N^k \in R^{m \times n_k}, M \in R^{m\times m}$, and $q \in R^m$.

For diagonalization methods, each MPEC is reformulated as an equivalent nonlinear program (8) and solved with TOMLAB–SNOPT [10]. For the SNCP method, one can solve the complementarity system NCP(t) (31) as a set of constraints in an optimization problem with a constant objective function such as 0. However, such a naive implementation will result in numerical instabilities when t is small, because the set of the Lagrange multipliers is unbounded for each MPEC. To stabilize the SNCP method, we minimize the sum of the components in λ^{GH} and use TOMLAB-SNOPT to solve the sequence of optimization problems

with $t = 1, 10^{-1}, \ldots, 10^{-15}$:

$$\begin{array}{ll} \text{minimize} & e^{\mathrm{T}}\lambda^{GH} \\ \text{subject to} & \mathrm{NCP}(t)\,(31). \end{array} \tag{33}$$

Table 5.1 summarizes the parameters used to generate the EPEC test problems. For the definition of these input parameters for each MPEC, see [12].

TABLE 5.1. Input parameters

(n_1, n_2)	(8, 10)	(first_deg1, first_deg2)	(1, 1)
m	15	second_deg	3
(l_1, l_2)	(8, 8)	(mix_deg1, mix_deg2)	(1, 1)

Table 5.3 gives the random seeds used to generate each test problem and the objective function values of MPECs at generated solutions for each test problem. Numerical results for the methods SNCP, nonlinear Gauss-Seidel (with $\varepsilon = 1.0\text{e}{-6}$), and nonlinear Jacobi (with $\varepsilon = 1.0\text{e}{-6}$) on generated test problems are shown in Tables 5.4 – 5.6. To investigate the impact of the accuracy tolerance ε on the performance of diagonalization methods, we ran the nonlinear Gauss-Seidel and nonlinear Jacobi methods on the same test problems with lower tolerance $\varepsilon = 1.0\text{e}{-4}$, and give the numerical results in Table 5.7 – 5.8. The notation used in these tables is explained in Table 5.2.

TABLE 5.2. Notation used for numerical results

Prob	Problem number.
random_seed	The random seed used to generate each EPEC problem.
Time	Total time (in seconds) needed by the termination of algorithms.
Out Iter	# of outer iterations required by diagonalization methods.
Maj Iter	# of major iterations.
f1*	The objective function value of MPEC1 at the found solution (x^{1*}, y^*).
f2*	The objective function value of MPEC2 at the found solution (x^{2*}, y^*).
$\mathrm{f1}_{gen}$	The objective function value of MPEC1 at the generated solution (x^1_{gen}, y_{gen}).
$\mathrm{f2}_{gen}$	The objective function value of MPEC2 at the generated solution (x^2_{gen}, y_{gen}).
Norm	The 2-norm of the difference vector between the found solution (x^{1*}, x^{2*}, y^*) and the generated solution $(x^1_{gen}, x^2_{gen}, y_{gen})$.
flag1	flag1 = 0 if the algorithm finds an equilibrium point; flag1 = 1 if the algorithm is terminated by reaching the iteration limit; flag1 = 2 if the SNCP is terminated by the infeasibility message.
flag2	flag2 = 0 if cycling behavior is not observed for diagonalization methods; flag2 = 1 if cycling behavior is observed for diagonalization methods.

TABLE 5.3. Information on test problems

Prob	random_seed	$f1_{gen}$	$f2_{gen}$
1	2.0e+5	−23.4844	22.8737
2	3.0e+5	−9.7748	−10.7219
3	4.0e+5	−16.7669	−4.6330
4	5.0e+5	−9.6054	−0.8600
5	6.0e+5	−46.9213	−11.1220
6	7.0e+5	−1.8838	−6.1389
7	8.0e+5	−14.9793	−12.1478
8	9.0e+5	−5.7299	−19.3843
9	1.0e+6	7.0672	−19.1931
10	1.1e+6	−3.2355	−17.3311

TABLE 5.4. Numerical results for SNCP method (Major Iteration limit = 120)

Prob	Maj Iter	Time	$f1^*$	$f2^*$	Norm	flag1
1	120	127.9	−23.4844	22.8739	2.70e−4	1
2	75	69.0	−9.7749	−10.7219	7.06e−5	0
3	45	48.4	−16.7669	−4.6330	6.48e−5	0
4	68	73.1	−9.6052	−0.8600	1.57e−3	0
5	80	72.9	−46.9213	−11.1220	4.70e−13	0
6	120	83.9	−1.8839	−6.1393	1.15e−3	1
7	75	71.8	−14.9790	−12.1477	4.03e−4	0
8	120	126.7	−5.7300	−19.3844	1.08e−4	1
9	54	52.6	7.0672	−19.1930	1.68e−4	0
10	72	68.0	−3.2363	−17.3301	2.04e−3	0

TABLE 5.5. Numerical results for nonlinear Gauss-Seidel method ($J = 30$, $\varepsilon = 1.0\text{e}{-6}$)

Prob	Out Iter	Maj Iter	Time	$f1^*$	$f2^*$	Norm	flag1	flag2
1	30	619	96.3	−23.4844	22.8737	1.23e−4	1	0
2	26	821	114.8	−9.7805	−10.7263	6.25e−3	0	0
3	30	903	133.7	−16.7672	−4.6327	6.06e−4	1	0
4	19	1340	232.3	−9.6044	−0.8601	4.67e−3	0	0
5	7	118	17.8	−46.9213	−11.1221	1.02e−4	0	0
6	30	508	73.8	−1.7661	−6.1783	1.18e−1	1	0
7	11	1076	191.0	−14.9807	−12.1489	1.80e−3	0	0
8	30	320	61.8	−5.7228	−19.3929	1.00e−2	1	0
9	9	189	29.4	7.0672	−19.1930	7.24e−5	0	0
10	15	170	30.6	−3.2265	−17.3179	1.50e−2	0	0

TABLE 5.6. Numerical results for nonlinear Jacobi method ($J = 30$, $\varepsilon = 1.0\text{e}{-6}$)

Prob	Out Iter	Maj Iter	Time	f1*	f2*	Norm	flag1	flag2
1	30	695	104.8	−23.4844	22.8738	1.25e−4	1	0
2	30	1036	138.2	−9.7756	−10.7262	4.34e−3	1	1
3	30	807	104.6	−16.7668	−4.6327	4.24e−4	1	1
4	30	703	94.6	−9.6031	−0.8601	6.21e−3	1	1
5	30	375	69.2	−46.9213	−11.1221	6.06e−5	1	1
6	30	819	103.6	−1.8837	−6.1672	3.91e−2	1	1
7	30	667	94.0	−14.9790	−12.1494	2.30e−3	1	1
8	30	847	108.2	−5.7314	−19.3929	6.85e−3	1	1
9	30	624	97.6	7.0672	−19.1930	5.56e−5	1	1
10	30	766	98.7	−3.2819	−17.3179	4.76e−2	1	1

TABLE 5.7. Numerical results for nonlinear Gauss-Seidel method ($J = 30$, $\varepsilon = 1.0\text{e}{-4}$)

Prob	Out Iter	Maj Iter	Time	f1*	f2*	Norm	flag1	flag2
1	5	134	17.9	−23.4844	22.8738	1.25e−4	0	0
2	4	152	24.5	−9.7805	−10.7263	6.24e−3	0	0
3	5	149	19.1	−16.7672	−4.6327	6.68e−4	0	0
4	6	149	20.2	−9.6044	−0.8601	4.71e−3	0	0
5	5	100	14.4	−46.9213	−11.1220	1.02e−4	0	0
6	30	508	73.8	−1.7661	−6.1783	1.15e−1	1	0
7	6	130	18.0	−14.9807	−12.1489	1.80e−3	0	0
8	17	299	47.0	−5.7228	−19.3929	1.00e−2	0	0
9	7	187	27.3	7.0672	−19.1930	7.42e−5	0	0
10	7	149	20.8	−3.2265	−17.3179	1.49e−2	0	0

TABLE 5.8. Numerical results for nonlinear Jacobi method ($J = 30$, $\varepsilon = 1.0\text{e}{-4}$)

Prob	Out Iter	Maj Iter	Time	f1*	f2*	Norm	flag1	flag2
1	10	257	36.8	−23.4844	22.8738	1.26e−4	0	0
2	30	1036	138.2	−9.7756	−10.7262	4.34e−3	1	1
3	30	807	104.6	−16.7668	−4.6327	4.24e−4	1	1
4	30	703	94.6	−9.6054	−0.8600	6.21e−3	1	1
5	8	155	23.6	−46.9213	−11.1220	6.14e−5	0	0
6	30	819	103.6	−1.8837	−6.1672	3.91e−2	1	1
7	30	667	94.0	−14.9790	−12.1494	2.30e−3	1	1
8	30	847	108.2	−5.7314	−19.3929	6.85e−3	1	1
9	30	624	97.6	7.0672	−19.1930	5.56e−5	1	1
10	30	766	98.7	−3.2819	−17.3179	4.76e−2	1	1

From the numerical results, we have the following observations.

- The SNCP algorithm solves 7 test problems and is terminated for reaching the major iteration limit for 3 test problems. However, the vectors returned by SNCP method

for those three test problems are close to the generated solutions with "Norm" of order 1.0e−4. It takes around one second to perform one major iteration. The number of major iterations for each of the 7 solved problem is consistently between 40 and 80.

- With $\varepsilon = 1.0\text{e}{-6}$, the nonlinear Gauss-Seidel method solves 7 test problems and is terminated for reaching the outer iteration limit for 4 test problems. However, with $\varepsilon = 1.0\text{e}{-4}$, the nonlinear Gauss-Seidel succeed in solving 9 test problems within at most 7 outer iterations. In fact, we observe that the nonlinear Gauss-Seidel method only needs a few (4 or 5) outer iterations to reach an accuracy of 1.0e−3 or 1.0e−4, and then makes slow progress to achieve higher accuracy of 1.0e−5 or 1.0e−6. Note that the cycling behavior is not observed for the nonlinear Gauss-Seidel method on any test problem.

- The results in Table 5.6 show that for each test problem, the nonlinear Jacobi method is terminated for reaching the outer iteration limit, although the solution vector returned by nonlinear Jacobi method is close to the generated solution. Surprisingly, the cycling behavior is observed for 9 out of the 10 problems. This confirms that when cycling occurs, the diagonalization method would fail. Even with lower accuracy ($\varepsilon = 1.0\text{e}{-4}$), nonlinear Jacobi method solves only two test problems. This observation suggests that the nonlinear Jacobi method has difficulty achieving high accuracy and is less reliable.

- The comparison of "Norm" for the SNCP algorithm and the nonlinear Gauss-Seidel method seems to suggest that the SNCP algorithm is able to reach the generated solution $(x^1_{gen}, x^2_{gen}, y_{gen})$ more accurately than the nonlinear Gauss-Seidel method. The fact that all these methods return a solution close to the generated solution $(x^1_{gen}, x^2_{gen}, y_{gen})$ seems to indicate that the generated solution is isolated or locally unique. Further investigation of the properties of the generated solutions is needed.

- With the accuracy tolerance $\varepsilon = 1.0\text{e}{-6}$, it is difficult to say which method, SNCP or nonlinear Gauss-Seidel, is more efficient. However, it is clear that both methods outperform the nonlinear Jacobi method. If a user is willing to accept lower accuracy, e.g., $\varepsilon = 1.0\text{e}{-2}$ or $\varepsilon = 1.0\text{e}{-4}$, the nonlinear Gauss-Seidel method can be very efficient.

6 Conclusions

We have proposed a new relaxation method for MPECs, which can be considered a generalization of Scholtes's regularization scheme. We have extended the convergence results for Scholtes's to our method. We also proposed a new sequential NCP approach for solving EPECs and established the convergence of this method. We presented numerical results for diagonalization methods and sequential nonlinear complementarity algorithms on randomly generated EPEC test problems. The results suggested that the cycling behavior is observed frequently for the nonlinear Jacobi method, which has the worst performance. The SNCP

algorithm consistently performs well in reaching solutions accurately, whereas the nonlinear Gauss-Seidel method can be very efficient if low accuracy is adequate.

Acknowledgements. I am grateful to Professor Richard Cottle for his support and helpful discussions. I thank Professor Michael Saunders for his exceptionally careful readings of this paper and detailed comments, and Dr. Sven Leyffer for his valuable suggestions.

References

[1] Anitescu, M. (2000): 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.

[2] Berry, C.A., Hobbs, B.F., Meroney, W.A., O'Neill, R.P., Stewart Jr., W.R. (1999): Understanding how market power can arise in network competition: A game theoretic approach, *Utility Policy*, **8**, 139–158, .

[3] Cardell, J.B., Hitt, C.C., Hogan, W.W. (1997): Market power and strategic interaction in electricity networks, *Resource and Energy Economics*, **19**, 109–137.

[4] Chen, Y., Florian, M. (1995): The nonlinear bilevel programming problem: Formulations, regularity and optimality conditions, *Optimization*, **32**, 193–209.

[5] Dafemos, S. (1983): An iterative scheme for variational inequalities, *Mathematical Programming*, **26**, 40–47.

[6] Fletcher, R., Leyffer, S. (2004): Solving mathematical programs with complementarity constraints as nonlinear programs, *Optimization Methods and Software*, **19**, 15–40.

[7] Fletcher, R., Leyffer, S., Ralph, D., Scholtes, S. (2002): Local convergence of SQP methods for mathematical programs with equilibrium constraints. Numerical Analysis Report NA/209, Department of Mathematics, University of Dundee, UK.

[8] Harker, P.T. (1984): A variational inequality approach for the determination of oligopolistic market equilibrium, *Mathematical Programming*, **30**, 105–111.

[9] Hobbs, B.F., Metzler, C.B., Pang, J.-S. (2000): Strategic gaming analysis for electric power networks: An MPEC approach, *IEEE Transactions on Power Systems*, **15**, 638–645.

[10] Holmström, K. (1999): TOMLAB Optimization, Webpage: `www.tomlab.biz`.

[11] Hu, X. (2002): *Mathematical Programs with Complementarity Constraints and Game Theory Models in Electricity Markets,* Ph.D. Thesis, Department of Mathematics and Statistics, University of Melbourne.

[12] Jiang, H., Ralph, D. (1999): QPECgen, a MATLAB generator for mathematical programs with quadratic objectives and affine variational inequality constraints, *Computational Optimization and Applications*, **13**, 25–49.

[13] Leyffer, S. (2003): Mathematical programs with equilibrium constraints, Preprint ANL/MCS-P1026-0203, Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL.

[14] Luo, Z.-Q., Pang, J.-S., Ralph, D. (1996): *Mathematical Programs with Equilibrium Constraints*, Cambridge University Press, Cambridge, UK.

[15] Mangasarian, O.L. (1969): *Nonlinear Programming*, McGraw-Hill, New York.

[16] Mordukhovich, B.S. (2004): Equilibrium problems with equilibrium constraints via multiobjective optimization, *Optimization Methods and Software*, **19**, 479–492.

[17] Pieper, H. (2001): *Algorithms for Mathematical Programs with Equilibrium Constraints with Applications to Deregulated Electricity Markets,* Ph.D. Thesis, Department of Management Science and Engineering, Stanford University, Stanford, CA.

[18] Pang, J.-S., Chan, D. (1982): Iterative methods for variational and complementarity problems, *Mathematical Programming*, **24**, 284–313.

[19] Pang, J.-S., Fukushima, M. (1999): Complementarity constraint qualifications and simplified B-stationarity conditions for mathematical programs with equilibrium constraints, *Computational Optimization and Applications*, **13**, 111–136.

[20] Pang, J.-S., Fukushima, M. (2002): Quasi-variational inequalities, generalized Nash equilibrium, and multi-leader-follower games, Submitted to *Computational Management Science*.

[21] Ralph, D., Wright, S.J. (2003): Some properties of regularization schemes for MPECs, Optimization Technical Report 03-04, Computer Science Department, University of Wisconsin, Madison, WI.

[22] Scheel, H., Scholtes, S. (2000): Mathematical programs with equilibrium constraints: stationarity, optimality, and sensitivity, *Mathematics of Operations Research*, **25**, 1–22.

[23] Scholtes, S. (2001): Convergence properties of a regularization scheme for mathematical programs with complementarity constraints, *SIAM J. on Optimization*, **11**, 918–936.

[24] Su, C.-L. (2004): Analysis on the forward market equilibrium model, Working paper, Department of Management Science and Engineering, Stanford University.