SIAM J. OPTIM.
Vol. 16, No. 2, pp. 587–609

A TWO-SIDED RELAXATION SCHEME FOR MATHEMATICAL PROGRAMS WITH EQUILIBRIUM CONSTRAINTS*

VICTOR DEMIGUEL†, MICHAEL P. FRIEDLANDER‡, FRANCISCO J. NOGALES§, AND STEFAN SCHOLTES¶

Abstract. We propose a relaxation scheme for mathematical programs with equilibrium constraints (MPECs). In contrast to previous approaches, our relaxation is two-sided: both the complementarity and the nonnegativity constraints are relaxed. The proposed relaxation update rule guarantees (under certain conditions) that the sequence of relaxed subproblems will maintain a strictly feasible interior—even in the limit. We show how the relaxation scheme can be used in combination with a standard interior-point method to achieve superlinear convergence. Numerical results on the MacMPEC test problem set demonstrate the fast local convergence properties of the approach.

Key words. nonlinear programming, mathematical programs with equilibrium constraints, complementarity constraints, constrained minimization, interior-point methods, primal-dual methods, barrier methods

AMS subject classifications. 49M37, 65K05, 90C30

DOI. 10.1137/04060754x

1. Introduction.

Consider the generic mathematical program with equilibrium constraints (MPEC), expressed as

$$\text{(MPEC)} \qquad \begin{array}{ll} \underset{x}{\text{minimize}} & f(x) \\ \text{subject to} & c(x) = 0, \\ & \min(x_1, x_2) = 0, \\ & x_0 \geq 0, \end{array}$$

where $x = (x_0, x_1, x_2) \in \mathrm{R}^{p \times n \times n}$, $f : \mathrm{R}^{p+2n} \to \mathrm{R}$ is the objective function, and $c : \mathrm{R}^{p+2n} \to \mathrm{R}^m$ is a vector of constraint functions. The complementarity constraint $\min(x_1, x_2) = 0$ requires that either $[x_1]_j$ or $[x_2]_j$ vanishes for each component $j = 1, \ldots, n$ and that the vectors x_1 and x_2 are nonnegative. See the survey paper by Ferris and Pang [5] for examples of complementarity models and the monographs by Luo, Pang, and Ralph [13] and Outrata, Kocvara, and Zowe [17] for details on MPEC theory and applications.

MPECs can be reformulated as standard nonlinear programs (NLPs) by replacing the nonsmooth complementarity constraint by a set of equivalent smooth constraints:

$$\min(x_1, x_2) = 0 \qquad \Longleftrightarrow \qquad X_1 x_2 = 0, \quad x_1, x_2 \geq 0,$$

*Received by the editors April 29, 2004; accepted for publication (in revised form) July 21, 2005; published electronically December 30, 2005.
http://www.siam.org/journals/siopt/16-2/60754.html
†London Business School, Regent's Park, London NW1 4SA, UK (avmiguel@london.edu).
‡Department of Computer Science, University of British Columbia, Vancouver, BC V6T 1Z4, Canada (mpf@cs.ubc.ca). The work of this author was partially supported by the Mathematical, Information, and Computational Sciences Division subprogram of the Office of Advanced Scientific Computing Research, Office of Science, U.S. Department of Energy contract W-31-109-Eng-38.
§Department of Statistics, Universidad Carlos III de Madrid, Avda. de la Universidad 30, 28911 Leganes, Madrid, Spain (FcoJavier.Nogales@uc3m.es).
¶Judge Institute of Management, University of Cambridge, Cambridge CB2 1AG, UK (s.scholtes@jims.cam.ac.uk).

where $X_1 = \text{diag}(x_1)$. However, these constraints do not admit a strictly feasible point, which implies that both the linear independence and the weaker Mangasarian–Fromovitz constraint qualifications are violated at every feasible point. These conditions are key ingredients for standard convergence analyses of NLP methods.

We propose a strategy that forms a sequence of NLP approximations to the MPEC, each with a feasible set that has a strict interior and that will typically satisfy a constraint qualification. In contrast to previous approaches, the relaxation is two-sided: both the complementarity ($X_1 x_2 = 0$) and the nonnegativity ($x_1, x_2 \geq 0$) constraints are relaxed. The proposed relaxation update rules guarantee (under certain conditions) that the sequence of relaxed subproblems will maintain a strictly feasible interior—even in the limit. Consequently, a standard interior method may be applied to the relaxed subproblem, as we show in section 4. The relaxation scheme could, in principle, be used in combination with other Newton-type methods, such as sequential quadratic programming or linearly constrained Lagrangian [8] methods.

1.1. Other work on MPECs. The direct application of off-the-shelf nonlinear optimization codes to MPECs was long neglected following early reports of their poor performance. See, for example, Luo, Pang, and Ralph [13] and, more recently, Anitescu [1], who describes the poor performance of the popular MINOS [15] code on MacMPEC [11] test problems. Interest in the application of standard NLP methods to MPECs has been revitalized for two reasons, however. First, it is now clear that the approach makes sense because strong stationarity implies the existence of standard NLP multipliers for MPECs in their NLP form, albeit an unbounded set (see Fletcher et al. [6]). Second, Fletcher and Leyffer [7] report promising numerical results for sequential quadratic programming (SQP) codes. These favorable numerical results are complemented by the local convergence analysis in [6].

Considerable effort has gone into the specialization of standard nonlinear programming methods in answer to the attendant difficulties of reformulating MPECs as NLPs. The approaches can be roughly divided into two categories: penalization and relaxation strategies. Such a categorization may be viewed as synthetic, however, because both approaches share the same philosophy: to relax the troublesome complementarity constraints. The difference is evident in the methodology.

The first analyses of penalization approaches can be found in Scholtes and Stöhr [21] and in Anitescu [1]. The strategy is to eliminate the explicit complementarity constraints $X_1 x_2 = 0$ by adding an exact penalty function to the objective to account for complementarity violation. The structural ill-posedness is thereby removed from the constraints. Anitescu gives conditions under which SQP methods with an elastic mode, such as SNOPT, will converge locally at a fast rate when applied to MPECs. Hu and Ralph [9] analyze the global convergence of the penalization method with exact solves. Anitescu [2] gives global convergence results with inexact solves. The penalization approach has been applied within the interior-point context by Benson, Shanno, and Vanderbei [4] and Leyffer, Lopez-Calva, and Nocedal [10]. Both papers report very good numerical results. Leyffer, Lopez-Calva, and Nocedal also give a comprehensive global and local convergence analysis of the penalization approach within an interior-point framework.

The relaxation approach (sometimes called regularization) keeps the complementarity constraints explicit but relaxes them to $X_1 x_2 \leq \delta_k$, where δ_k is a positive vector that is driven to zero. This scheme replaces the MPEC by a sequence of relaxed subproblems whose strictly feasible region is nonempty. The approach was extensively analyzed by Scholtes [22]. We call this a one-sided relaxation scheme to contrast it against our approach. The one-sided relaxation strategy has been adopted by Liu and Sun [12] and Raghunathan and Biegler [18]. Liu and Sun propose an interior method that solves each of the relaxed subproblems to within a prescribed tolerance. On the other hand, the method of Raghunathan and Biegler takes only one iteration of an interior method on each of the relaxed subproblems. A difficulty associated with both methods is that the strictly feasible regions of the relaxed problems become empty in the limit, and this may lead to numerical difficulties. Raghunathan and Biegler address this difficulty by using a modified search direction that ensures that their algorithm converges locally at a quadratic rate.

The relaxation scheme that we propose (described in section 3) does not force the strictly feasible regions of the relaxed MPECs to become empty in the limit. As a result, one can apply a standard interior method to the relaxed problems without having to modify the search direction, as in [18]. But like [18], our algorithm (described in section 4) performs only one interior iteration per relaxed problem. We show in section 4.2 that it converges locally at a superlinear rate, and in section 5 we discuss some implementation issues. We illustrate in section 6 the performance of the algorithm on a subset of the MacMPEC test problems. The numerical results seem to reflect our local convergence analysis and give evidence to the algorithm's effectiveness in practice.

1.2. Definitions. Unless otherwise specified, the function $\|x\|$ represents the Euclidean norm of a vector x. With vector arguments, the functions $\min(\cdot,\cdot)$ and $\max(\cdot,\cdot)$ apply componentwise to each element of the arguments. Denote by $[\,\cdot\,]_i$ the ith component of a vector. The uppercase variables X, S, V, and Z denote diagonal matrices formed from the elements of the vectors x, s, v, and z, respectively. Let $g(x)$ denote the gradient of the objective function $f(x)$. Let $A(x)$ denote the Jacobian of $c(x)$, a matrix whose ith row is the gradient of $[c(x)]_i$. Let $H_i(x)$ denote the Hessian of $[c(x)]_i$.

We make frequent use of standard definitions for linear independence constraint qualification (LICQ) and strict complementary slackness (SCS), and the second order sufficiency condition (SOSC). These definitions can be found in [16, Ch. 12].

2. Optimality conditions for MPECs. The standard KKT theory of nonlinear optimization is not directly applicable to MPECs because standard constraint qualifications do not hold. There is a simple way around this problem, however, as observed by Scheel and Scholtes [20]. At every feasible point of the MPEC one can define the relaxed NLP, which is typically well behaved in nonlinear programming terms. It is shown in [20] that the KKT conditions of the relaxed NLP are necessary optimality conditions for (MPEC), provided that the relaxed NLP satisfies LICQ.

2.1. First-order conditions and constraint qualification. Let $\bar{x}$ be feasible with respect to (MPEC). The relaxed NLP at $\bar{x}$ is defined as

$$
\begin{array}{llll}
(\mathrm{RNLP}_{\bar{x}}) & \underset{x}{\text{minimize}} & f(x) & \\
& \text{subject to} & c(x) = 0, \quad x_0 \geq 0 & \\
& & [x_1]_j = 0, \quad [x_2]_j \geq 0 & \text{for all } j \text{ such that } [\bar{x}_1]_j = 0 < [\bar{x}_2]_j, \\
& & [x_1]_j \geq 0, \quad [x_2]_j = 0 & \text{for all } j \text{ such that } [\bar{x}_1]_j > 0 = [\bar{x}_2]_j, \\
& & [x_1]_j \geq 0, \quad [x_2]_j \geq 0 & \text{for all } j \text{ such that } [\bar{x}_1]_j = 0 = [\bar{x}_2]_j.
\end{array}
$$

The feasible region defined by the bound constraints of $(\mathrm{RNLP}_{\bar{x}})$ is larger than that defined by the equilibrium constraints. Hence the term *relaxed* NLP. Most important, the problematic equilibrium constraints of (MPEC) have been substituted by a better-posed system of equality and inequality constraints.

Define

$$\mathcal{L}(x, y) = f(x) - y^T c(x)$$

as the Lagrangian function of $\text{RNLP}_{\bar{x}}$.

Despite a possibly larger feasible set, it can be shown that if LICQ holds for (RNLP_{x^*}), its KKT conditions are also necessary optimality conditions for (MPEC) [20]. This observation leads to the following stationarity concept for MPECs.

Definition 2.1. *A point* (x^*, y^*, z^*) *is* strongly stationary *for (MPEC) if it satisfies the KKT conditions for* (RNLP_{x^*})*:*

$$\nabla_x \mathcal{L}(x^*, y^*) = z^*, \tag{2.1a}$$

$$c(x^*) = 0, \tag{2.1b}$$

$$\min(x_0^*, z_0^*) = 0, \tag{2.1c}$$

$$\min(x_1^*, x_2^*) = 0, \tag{2.1d}$$

$$[x_1^*]_j [z_1^*]_j = 0, \tag{2.1e}$$

$$[x_2^*]_j [z_2^*]_j = 0, \tag{2.1f}$$

$$[z_1^*]_j, [z_2^*]_j \geq 0, \quad \text{if} \quad [x_1^*]_j = [x_2^*]_j = 0. \tag{2.1g}$$

With the relaxed NLP we can define a constraint qualification for MPECs analogous to LICQ and deduce a necessary optimality condition for MPECs.

Definition 2.2. *The point* x^* *satisfies* MPEC linear independence constraint qualification *(MPEC-LICQ) for (MPEC) if it is feasible and if LICQ holds at* x^* *for* (RNLP_{x^*}).

Proposition 2.3 (see, for example, Scheel and Scholtes [20]). *If* x^* *is a local minimizer for (MPEC) at which MPEC-LICQ holds, then there exist unique multipliers* y^* *and* z^* *such that* (x^*, y^*, z^*) *is strongly stationary.*

2.2. Strict complementarity and second-order sufficiency. Through the relaxed NLP we can define strict complementarity and second-order conditions for MPECs. These play a crucial role in the development and analysis of the relaxation scheme proposed in this paper.

We define two different strict complementary slackness conditions for MPECs. The first of the two is stronger and is the one assumed in [22, Theorem 4.1]. It requires all multipliers z_0, z_1, and z_2 to be strictly complementary with respect to their associated primal variables. In our analysis, we only assume the second, less restrictive condition, which only requires strict complementarity of z_0.

Definition 2.4. *The triple* (x^*, y^*, z^*) *satisfies the* MPEC strict complementary slackness *(MPEC-SCS) condition for (MPEC) if it is strongly stationary, if* $\max(x_0^*, z_0^*) > 0$*, and if* $[x_i^*]_j + [z_i^*]_j \neq 0$ *for each* $i = 1, 2$ *and* $j = 1, \ldots, n$.

Definition 2.5. *The triple* (x^*, y^*, z^*) *satisfies* MPEC weak strict complementary slackness *(MPEC-WSCS) for (MPEC) if it is strongly stationary and if* $\max(x_0^*, z_0^*) > 0$.

We define two second-order sufficient conditions for MPECs: MPEC-SOSC and MPEC-SSOSC. The first condition is equivalent to the RNLP-SOSC defined by Ralph and Wright [19, Definition 2.7]. The second condition is stronger than the RNLP-SSOSC defined in [19, Definition 2.8].

The tangent cone of the feasible set of (RNLP$_{x^*}$) is given by

$$\begin{aligned}\mathcal{T} = \{&\alpha p \mid \alpha > 0,\ p \in \mathrm{R}^n\} \\ &\cap \{p \mid A(x^*)p = 0\} \\ &\cap \{p \mid [p_0]_j \geq 0 \text{ for all } j \text{ such that } [x_0^*]_j = 0\}.\end{aligned}$$

The second-order sufficient condition for optimality depends on positive curvature of the Lagrangian in a subspace, i.e.,

$$p^T \nabla^2_{xx} \mathcal{L}(x^*, y^*) p > 0, \quad p \neq 0, \tag{2.2}$$

for all p in some subset of the feasible directions $\mathcal{T}$.

DEFINITION 2.6. *The triple* (x^*, y^*, z^*) *satisfies the* MPEC second-order sufficiency condition *(MPEC-SOSC) for (MPEC) if it is strongly stationary and if* (2.2) *holds for all nonzero* $p \in \overline{\mathcal{F}}$*, where*

$$\begin{aligned}\overline{\mathcal{F}} \stackrel{\text{def}}{=} \{p \in \mathcal{T} \mid [p_0]_j = 0 &\quad \text{for all } j \text{ such that } [x_0^*]_j = 0 \text{ (and } [z_0^*]_j > 0), \\ [p_0]_j \geq 0 &\quad \text{for all } j \text{ such that } [x_0^*]_j = 0 \text{ (and } [z_0^*]_j = 0), \\ [p_i]_j = 0 &\quad \text{for all } j \text{ such that } [x_i^*]_j = 0 \text{ (and } [z_i^*]_j \neq 0),\ i = 1, 2, \\ [p_i]_j \geq 0 &\quad \text{for all } j \text{ such that } [x_i^*]_j = 0 \text{ (and } [z_i^*]_j = 0),\ i = 1, 2, \\ [p_i]_j = 0 &\quad \text{for all } j \text{ such that } [x_i^*]_j = 0 < [x_\ell^*]_j,\ i, \ell = 1, 2,\ i \neq \ell\}.\end{aligned}$$

If the last two conditions in the definition of $\overline{\mathcal{F}}$ are dropped, we obtain a stronger second-order condition, which is equivalent to the one assumed in [22, Theorem 4.1].

DEFINITION 2.7. *The triple* (x^*, y^*, z^*) *satisfies* MPEC strong second-order sufficiency condition *(MPEC-SSOSC) for (MPEC) if it is strongly stationary and if* (2.2) *holds for all nonzero* $p \in \mathcal{F}$*, where*

$$\begin{aligned}\mathcal{F} \stackrel{\text{def}}{=} \{p \in \mathcal{T} \mid [p_0]_j = 0 &\quad \text{for all } j \text{ such that } [x_0^*]_j = 0 \text{ (and } [z_0^*]_j > 0), \\ [p_0]_j \geq 0 &\quad \text{for all } j \text{ such that } [x_0^*]_j = 0 \text{ (and } [z_0^*]_j = 0), \\ [p_i]_j = 0 &\quad \text{for all } j \text{ such that } [x_i^*]_j = 0 \text{ (and } [z_i^*]_j \neq 0),\ i = 1, 2\}.\end{aligned}$$

Note that MPEC-SSOSC ensures that the Hessian of the Lagrangian has positive curvature in the range space of all nonnegativity constraints $(x_1, x_2 \geq 0)$ whose multipliers are zero. Note also that MPEC-SOSC and -SSOSC are equivalent when MPEC-SCS holds.

In our analysis, we assume MPEC-WSCS and -SSOSC. However, we note that our results are also valid under MPEC-SCS and -SOSC. To see this, simply note that MPEC-SCS implies MPEC-WSCS and that MPEC-SOSC and -SSOSC are equivalent when MPEC-SCS holds. Thus our analysis holds either under MPEC-SCS and -SOSC, or under a weaker SCS (at the expense of assuming a stronger SOSC).

Raghunathan and Biegler [18] make a strict complementarity assumption that is more restrictive than MPEC-WSCS but less restrictive than MPEC-SCS. In particular, they require $\max(x_0^*, z_0^*) > 0$ and $[z_i^*]_j \neq 0$ for all j such that $[\bar{x}_1]_j = [\bar{x}_2]_j = 0$. This condition is termed upper-level SCS (MPEC-USCS) in [19]. The strength of the second-order condition they assume is also between that of MPEC-SOSC and MPEC-SSOSC. In particular, their second-order condition is obtained from the MPEC-SOSC by dropping the last condition in the definition of $\overline{\mathcal{F}}$.

3. A strictly feasible relaxation scheme.

In this section we propose a relaxation scheme for which the strictly feasible region of the relaxed problems may remain nonempty even in the limit.

A standard relaxation of the complementarity constraint proceeds as follows. The complementarity constraint $\min(x_1, x_2) = 0$ is first reformulated as the system of inequalities $X_1 x_2 \leq 0$ and $x_1, x_2 \geq 0$. A vector $\delta_c \in \mathrm{R}^n$ of strictly positive parameters relaxes the complementarity constraint to arrive at

$$X_1 x_2 \leq \delta_c, \quad x_1, x_2 \geq 0. \tag{3.1}$$

The original complementarity constraint is recovered when $\delta_c = 0$. Note that at all points feasible for (MPEC) the gradients of the active constraints in (3.1) are linearly independent when $\delta_c > 0$. Moreover, the strictly feasible region of the relaxed constraints (3.1) is nonempty when $\delta_c > 0$. Unfortunately, the strictly feasible region of the relaxed MPEC becomes empty as the components of δ_c tend to zero.

3.1. A two-sided relaxation.

In contrast to (3.1), our proposed scheme additionally relaxes each component of the bounds $x_1, x_2 \geq 0$ by the amounts $[\delta_1]_j > 0$ and $[\delta_2]_j > 0$ so that the relaxed complementarity constraints become

$$X_1 x_2 \leq \delta_c, \quad x_1 \geq -\delta_1, \quad x_2 \geq -\delta_2, \tag{3.2}$$

where $\delta_c, \delta_1, \delta_2 \in \mathrm{R}^n$ are vectors of strictly positive relaxation parameters. Note that for any relaxation parameter vectors $(\delta_1, \delta_2, \delta_c)$ that satisfy $\max(\delta_c, \delta_1) > 0$ and $\max(\delta_c, \delta_2) > 0$, the strictly feasible region of (3.2) is nonempty, and the active constraint gradients are linearly independent.

The main advantage of the strictly feasible relaxation scheme (3.2) is that there is no need to drive both relaxation parameters to zero to recover a stationary point of the MPEC. As we show in Theorem 3.1, for any strongly stationary point of (MPEC) that satisfies MPEC-LICQ, -WSCS, and -SSOSC, there exist relaxation parameter vectors $(\delta_1^*, \delta_2^*, \delta_c^*)$ satisfying $\max(\delta_c^*, \delta_1^*) > 0$ and $\max(\delta_c^*, \delta_2^*) > 0$ such that the relaxed MPEC satisfies LICQ, SCS, and SOSC.

3.2. An example.

The intuition for the relaxation scheme proposed in section 3.1 is best appreciated with an example. Consider the MPEC [22]

$$\begin{array}{ll} \underset{x}{\text{minimize}} & \frac{1}{2}\left[(x_1 - a)^2 + (x_2 - b)^2\right] \\ \text{subject to} & \min(x_1, x_2) = 0 \end{array} \tag{3.3}$$

and the associated relaxed MPEC derived by applying the relaxation (3.2) to (3.3):

$$\begin{array}{ll} \underset{x}{\text{minimize}} & \frac{1}{2}\left[(x_1 - a)^2 + (x_2 - b)^2\right] \\ \text{subject to} & \quad x_1 \geq -\delta_1, \\ & \quad x_2 \geq -\delta_2, \\ & X_1 x_2 \leq \delta_c. \end{array} \tag{3.4}$$

For any choice of parameters $a, b > 0$, (3.3) has two local minimizers: $(a, 0)$ and $(0, b)$. Each is strongly stationary and satisfies MPEC-LICQ, -SCS, and -SOSC and thus they also satisfy MPEC -LICQ, -WSCS, and -SSOSC. Evidently, these local minimizers are also minimizers of (3.4) for $\delta_c = 0$ and for any $\delta_1, \delta_2 > 0$. If the data are changed so that $a > 0$ and $b < 0$, then the point $(a, 0)$ is a unique minimizer of (3.3), and also a unique minimizer of (3.4) for any $\delta_c > 0$ and for $\delta_1 = \delta_2 = 0$. Moreover, if $a, b < 0$, then $(0, 0)$ is the unique minimizer of (3.3) and also a unique minimizer of (3.4) for any $\delta_c > 0$ and for $\delta_1 = \delta_2 = 0$. Thus there is no need to drive both δ_c and δ_1, δ_2 to zero to recover a stationary point of (MPEC).

A key property of MPECs that we exploit is the fact that the MPEC multipliers provide information about which relaxation parameters need to be driven to zero. To illustrate this, let us suppose $a, b > 0$ and consider the local minimizer $(a, 0)$ of the MPEC. In this simple example the minimizer of the relaxed problem will lie on the curve $X_1 x_2 = \delta_c$ for all sufficiently small δ_c. The MPEC solution will be recovered if we drive δ_c to zero. The values of the other parameters δ_1, δ_2 have no impact as long as they remain positive; the corresponding constraints will remain inactive. Note that this situation occurs precisely if the MPEC multiplier of the active constraint, here $x_2 \geq 0$, is negative, that is, the gradient of the objective function points outside of the positive orthant. If this situation is observed algorithmically, we will reduce δ_c and keep δ_1, δ_2 positive. A similar argument can be made if the gradient points in the interior of the positive orthant, in which case δ_1 or δ_2 need to be driven to zero to recover the MPEC minimizer. The parameter δ_c, however, must remain positive to maintain the strict interior of the feasible set.

The foregoing cases correspond to nondegenerate solutions; that is, there are no biactive constraints. Biactivity occurs in the example if $a, b < 0$. In this case the minimizer is the origin, and both MPEC multipliers are positive. Hence, we need to drive δ_1, δ_2 to zero and keep δ_c positive to avoid a collapsing strictly feasible region.

To see how one can recover an MPEC minimizer that satisfies MPEC-WSCS and -SSOSC, consider the example with $a = 0$ and $b = 1$. In this case $(0, 1)$ is a minimizer satisfying MPEC-WSCS and -SSOSC. To recover this minimizer from the relaxed MPEC (3.4) we do not need to drive any of the three relaxation parameters to zero. In particular, it is easy to see that $(0, 1)$ is a minimizer to the relaxed problem satisfying LICQ, SCS, and SOSC for any $\delta_1, \delta_2, \delta_c > 0$.

Our goal in the remainder of this paper is to turn this intuition into an algorithm and to analyze its convergence behavior for general MPECs.

3.3. The relaxed MPEC. In addition to introducing the relaxation parameter vectors $(\delta_1, \delta_2, \delta_c)$, we introduce slack variables $s \equiv (s_0, s_1, s_2, s_c)$ so that only equality and nonnegativity constraints on s are present. The resulting relaxed MPEC is

$$
\text{(MPEC-}\delta\text{)} \qquad \begin{array}{lrl}
\underset{x,s}{\text{minimize}} & f(x) & \\
\text{subject to} & c(x) = 0 & : y, \\
& s_0 - x_0 = 0 & : v_0, \\
& s_1 - x_1 = \delta_1 & : v_1, \\
& s_2 - x_2 = \delta_2 & : v_2, \\
& s_c + X_1 x_2 = \delta_c & : v_c, \\
& s \geq 0, &
\end{array}
$$

where the dual variables y and $v \equiv (v_0, v_1, v_2, v_c)$ are shown next to their corresponding constraints. We note that the slack variable s_0 is not strictly necessary—the nonnegativity of x_0 could be enforced directly. However, such a device may be useful in practice because an initial value of x can be used without modification, and we need to choose starting values only for s, y, and v. Moreover, this notation greatly simplifies the following discussion.

To formulate the stationarity conditions for the relaxed MPEC, we group the set of equality constraints involving the slack variables s into a single expression by defining

$$h(x,s) = -\begin{bmatrix} s_0 - x_0 \\ s_1 - x_1 \\ s_2 - x_2 \\ s_c + X_1 x_2 \end{bmatrix} \quad \text{and} \quad \delta = \begin{bmatrix} 0 \\ \delta_1 \\ \delta_2 \\ \delta_c \end{bmatrix}. \tag{3.5}$$

The Jacobian of h with respect to the variables x is given by

$$B(x) \equiv \nabla_x h(x,s)^T = \begin{bmatrix} I & & \\ & I & \\ & & I \\ & -X_2 & -X_1 \end{bmatrix}. \tag{3.6}$$

The tuple (x^*, s^*, y^*, v^*) is a KKT point for (MPEC-δ) if it satisfies

$$\nabla_x \mathcal{L}(x,y) - B(x)^T v \equiv r_d = 0, \tag{3.7a}$$

$$\min(s,v) \equiv r_c = 0, \tag{3.7b}$$

$$c(x) \equiv r_f = 0, \tag{3.7c}$$

$$h(x,s) + \delta \equiv r_\delta = 0. \tag{3.7d}$$

Define the vector $w = (x, s, y, v)$ and the vector $r(w;\delta) = (r_d, r_c, r_f, r_\delta)$ as a function of w and δ. With this notation, w^* is a KKT point for (MPEC-δ) if $\|r(w^*;\delta)\| = 0$. The Jacobian of (3.7) is given by

$$K(w) \equiv \begin{bmatrix} H(x) & & -A(x)^T & -B(x)^T \\ & V & & S \\ A(x) & & & \\ B(x) & -I & & \end{bmatrix},$$

where

$$H(x) \equiv \nabla^2_{xx} \mathcal{L}(x,y) + \begin{bmatrix} 0 & & \\ & & V_c \\ & V_c & \end{bmatrix}.$$

3.4. Properties of the relaxed MPEC. Stationary points of (MPEC-δ) are closely related to those of (MPEC) for certain values of the relaxation parameters. The following theorem makes this relationship precise.

THEOREM 3.1. *Let (x^*, y^*, z^*) be a strongly stationary point of (MPEC), and let the vector δ^* satisfy*

$$[\delta_i^*]_j = 0 \quad \text{if} \quad [z_i^*]_j > 0, \tag{3.8a}$$

$$[\delta_i^*]_j > 0 \quad \text{if} \quad [z_i^*]_j \le 0, \tag{3.8b}$$

$$[\delta_c^*]_j = 0 \quad \text{if} \quad [z_1^*]_j < 0 \quad \text{or} \quad [z_2^*]_j < 0, \tag{3.8c}$$

$$[\delta_c^*]_j > 0 \quad \text{if} \quad [z_1^*]_j \ge 0 \quad \text{and} \quad [z_2^*]_j \ge 0 \tag{3.8d}$$

for $i = 1, 2$ *and* $j = 1, \ldots, n$. *Then*

$$\max(\delta_c^*, \delta_1^*) > 0 \quad \textit{and} \quad \max(\delta_c^*, \delta_2^*) > 0, \tag{3.9}$$

and the point (x^*, s^*, y^*, v^*), *with*

$$(s_0^*, s_1^*, s_2^*) = (x_0^*, x_1^* + \delta_1^*, x_2^* + \delta_2^*), \tag{3.10a}$$

$$(v_0^*, v_1^*, v_2^*) = (z_0^*, [z_1^*]^+, [z_2^*]^+), \tag{3.10b}$$

$$s_c^* = \delta_c^*, \tag{3.10c}$$

and

$$[v_c^*]_j = \begin{cases} [-z_1^*]_j^+/[x_2^*]_j & \text{if} \quad [x_2^*]_j > 0 \quad (\text{and } [x_1^*]_j = 0), \\ [-z_2^*]_j^+/[x_1^*]_j & \text{if} \quad [x_1^*]_j > 0 \quad (\text{and } [x_2^*]_j = 0), \\ 0 & \text{if} \quad [x_1^*]_j = [x_2^*]_j = 0 \end{cases} \tag{3.10d}$$

for $j = 1, \ldots, n$, *is a stationary point for (MPEC-*δ^**). Moreover, if* (x^*, y^*, z^*) *satisfies MPEC- LICQ, -WSCS, or -SSOSC for (MPEC), then* (x^*, s^*, y^*, v^*) *satisfies the LICQ, SCS, or SOSC, respectively, for (MPEC-*δ^**).*

Proof. The proof is divided into three parts. The first part demonstrates that (x^*, s^*, y^*, v^*) is a stationary point for (MPEC-δ^*) and that SCS is satisfied. The second and third parts prove that LICQ and SOSC hold for (MPEC-δ^*), respectively, if MPEC-LICQ and MPEC-SOSC hold.

Part 1. *Stationarity and SCS.* We first need to show that (3.9) holds. For $j = 1, \ldots, n$ consider the following cases. If $[z_1^*]_j, [z_2^*]_j \le 0$, then by (3.8b) we have that $[\delta_1^*]_j, [\delta_2^*]_j > 0$, and thus (3.9) holds. Note that the case $[z_1^*]_j > 0$ and $[z_2^*]_j < 0$ (or $[z_1^*]_j < 0$ and $[z_2^*]_j > 0$) cannot take place because otherwise (2.1e)–(2.1f) imply that $[x_1^*]_j, [x_2^*]_j = 0$, and then (2.1g) requires $[z_1^*]_j, [z_2^*]_j \ge 0$, which is a contradiction. Finally, if $[z_1^*]_j, [z_2^*]_j \ge 0$, then by (3.8d) we have that $[\delta_c^*]_j > 0$. Thus (3.9) holds, as required.

Next we verify stationarity of (x^*, s^*, y^*, v^*) for (MPEC-δ^*). The point (x^*, y^*, z^*) is strongly stationary for (MPEC), and so by Definition 2.1, it satisfies conditions (2.1). Then from (3.5), (3.6), and (3.10), (x^*, y^*, s^*, v^*) satisfies (3.7a) and (3.7c)–(3.7d).

We now show that s^* and v^* satisfy (3.7b). First, note from (3.10) that $s^*, v^* \ge 0$ because $x^* \ge 0$ and $\delta_c^*, \delta_1^*, \delta_2^* \ge 0$.

To see that s^* and v^* are componentwise strictly complementary if WSCS holds for the (MPEC), recall that WSCS requires that x_0^* and z_0^* are strictly complementary; hence (3.10a) and (3.10b) imply that s_0^* and v_0^* are also strictly complementary. For x_1^* and x_2^*, consider the indices $i = 1, 2$. If $[z_i^*]_j = 0$, then $[v_i^*]_j = 0$ and $[\delta_i^*]_j > 0$. From (3.10a) it follows that $[s_i^*]_j > 0$, as required. If $[z_i^*]_j > 0$, then (3.10b) implies that $[v_i^*]_j > 0$. Moreover, by (2.1e)–(2.1f), and (3.8a), $[x_i^*]_j = [\delta_i^*]_j = 0$. Hence $[s_i^*]_j = 0$, and $[s_i^*]_j$ and $[v_i^*]_j$ are strictly complementary, as required. If $[z_i^*]_j < 0$, then $[v_i^*]_j = 0$, and by (3.10a) and (3.8b), $[s_i^*]_j > 0$. Hence $[s_i^*]_j$ and $[v_i^*]_j$ are again strictly complementary. It remains to verify that $[s_c^*]_j$ and $[v_c^*]_j$ are strictly complementary. If $[s_c^*]_j = 0$, then (3.10c) and (3.8c) imply that $[z_1^*]_j < 0$ or $[z_2^*]_j < 0$ and $[v_c^*]_j > 0$ by (3.10d), as required. If $[s_c^*]_j > 0$ then (3.10c) implies that $[\delta_c^*]_j > 0$, and by (3.8d) we have that $[z_1^*]_j \ge 0$ and $[z_2^*]_j \ge 0$. Then by (3.10d) we know that $[v_c^*]_j = 0$.

Part 2. *LICQ.* Next we prove that (x^*, s^*, y^*, v^*) satisfies LICQ for (MPEC-δ^*) if (x^*, y^*, z^*) satisfies MPEC-LICQ for (MPEC). Note that LICQ holds for (MPEC-δ^*)

if and only if LICQ holds at x^* for the following system of equalities and inequalities:

$$c(x) = 0, \tag{3.11a}$$

$$x_0 \geq 0, \tag{3.11b}$$

$$x_1 \geq -\delta_1^*, \tag{3.11c}$$

$$x_2 \geq -\delta_2^*, \tag{3.11d}$$

$$X_1 x_2 \leq \delta_c^*. \tag{3.11e}$$

But MPEC-LICQ implies that the following system of equalities and inequalities satisfies LICQ at x^*:

$$c(x) = 0, \qquad x \geq 0. \tag{3.12}$$

We now show that the gradients of the active constraints in (3.11) are either a subset or a nonzero linear combination of the gradients of the active constraints in (3.12), and that therefore they must be linearly independent at x^*. To do so, for $j = 1, \dots, n$, we consider the two cases $[\delta_c^*]_j > 0$ and $[\delta_c^*]_j = 0$.

If $[\delta_c^*]_j > 0$, the feasibility of x^* with respect to (MPEC) implies that the inequality $[X_1^* x_2^*]_j \leq [\delta_c^*]_j$ is not active. Moreover, because $\delta_1^*, \delta_2^* \geq 0$ and x^* is feasible with respect to (MPEC), we have that if the constraint $[x_1^*]_j \geq -\delta_1^*$ or $[x_2^*]_j \geq -\delta_2^*$ is active, then the corresponding constraint $[x_1^*]_j \geq 0$ or $[x_2^*]_j \geq 0$ is active. Thus, for the case $[\delta_c^*]_j > 0$, the set of constraints active in (3.11) is a subset of the set of constraints active in (3.12).

Now consider the case $[\delta_c^*]_j = 0$. By (3.9) we have that $[\delta_1^*]_j, [\delta_2^*]_j > 0$, and because x^* is feasible for (MPEC), the jth component (3.11e) is active, but the jth components of (3.11c)–(3.11d) are inactive. In addition, note that the gradient of this constraint has all components equal to zero except $\partial[X_1^* x_2^*]_j / \partial[x_1]_j = [x_2^*]_j$ and $\partial[X_1^* x_2^*]_j / \partial[x_2]_j = [x_1^*]_j$. Moreover, by (3.8c) we know that either $[z_1^*]_j$ or $[z_2^*]_j$ is strictly negative, and thus by (2.1g) we have that $[\max(x_1^*, x_2^*)]_j > 0$. Also, because x^* is feasible for (MPEC), $[\min(x_1^*, x_2^*)]_j = 0$. Thus one, and only one, of $[x_1^*]_j$ and $[x_2^*]_j$ is zero, and thus the gradient of the active constraint $[x_1^*]_j [x_2^*]_j \leq [\delta_c^*]_j$ is a nonzero linear combination of the gradient of whichever of the two constraints $[x_1^*]_j \geq 0$ and $[x_2^*]_j \geq 0$ is active.

Thus, the gradients of the constraints active in system (3.11) are either a subset or a nonzero linear combination of the constraints active in (3.12), and thus LICQ holds for (MPEC-δ^*).

Part 3. *SOSC.* To complete the proof, we need to show that SSOSC at (x^*, y^*, z^*) for (MPEC) implies SOSC at (x^*, s^*, y^*, v^*) for (MPEC-δ^*). Because the slack variables appear linearly in (MPEC-δ^*), we need only to show that (x^*, y^*, v^*) satisfies SOSC for the equivalent problem without slack variables,

$$\begin{array}{ll} \underset{x,s}{\text{minimize}} & f(x) \\ \text{subject to} & c(x) = 0, \\ & x_0 \geq 0, \\ & x_1 \geq -\delta_1^*, \\ & x_2 \geq -\delta_2^*, \\ & X_1 x_2 \leq \delta_c^*, \end{array} \tag{3.13}$$

and with solution (x^*, y^*, v^*). First, we show that the set of critical directions at (x^*, y^*, v^*) for (3.13) is equal to $\mathcal{F}$ (see Definition 2.6). Consider the critical directions for the first two constraints of (3.13). Because the constraints $c(x) = 0$ and $x_0 \geq 0$ and their multipliers are the same for (MPEC) and (3.13), their contribution to the definition of the set of critical directions is the same. In particular, we need to consider critical directions such that $A(x^*)p = 0$, $[p_0]_j \geq 0$ for all j such that $[x_0^*]_j = 0$, and $[p_0]_j = 0$ for all j such that $[z_0^*]_j > 0$. Next, consider the critical direction for the last three constraints of (3.13), $x_1 \geq -\delta_1^*$, $x_2 \geq -\delta_2^*$ and $X_1 x_2 \leq \delta_c^*$. Because we have shown that SCS holds for (3.13) at (x^*, y^*, v^*), we need only to impose the condition $[p_i]_j = 0$ for all j such that $[v_i^*]_j > 0$ for $i = 1, 2$, and $[p_i]_j = 0$ for all i and j such that $[v_c^*]_j > 0$ and $[x_i^*]_j = 0$. But note that because of (3.10) and (3.8), this is equivalent to imposing $[p_i]_j = 0$ for all j such that $[x_i^*]_j = 0$ and $[z_i^*]_j \neq 0$ for $i = 1, 2$, which is the definition of $\mathcal{F}$.

But note that the Hessian of the Lagrangian for (3.13) is different from the Hessian of the Lagrangian for (MPEC). The reason is that in (3.13) the complementarity constraint $X_1 x_2 \leq \delta_c^*$ is included in the Lagrangian, whereas we excluded this constraint from the definition of the Lagrangian for (MPEC). But it is easy to see that this has no impact on the value of $p^T \nabla^2_{xx} \mathcal{L}(x^*, y^*) p > 0$ for all $p \in \mathcal{F}$. To see this, note that the Hessian of $[X_1 x_2]_j$ has only two nonzero elements:

$$\nabla^2_{[x_1]_j [x_2]_j} [X_1 x_2]_j = \begin{bmatrix} & 1 \\ 1 & \end{bmatrix}. \tag{3.14}$$

If $[v_c^*]_j = 0$, then the Hessian of the complementarity constraint $[X_1 x_2]_j \leq [\delta_c^*]_j$ is multiplied by zero, and thus the Hessian of the Lagrangian for (MPEC-δ^*) is the same as the Hessian of the Lagrangian for (MPEC). Now suppose $[v_c^*]_j \neq 0$. Because SCS holds for (3.13), we have that $[v_c^*]_j \neq 0$ implies that the set of critical directions satisfies either $[p_1]_j = 0$ or $[p_2]_j = 0$. This, together with (3.14), implies that $p^T \nabla^2_{xx}([X_1^* x_2^*]_j) p = 0$ for all $p \in \mathcal{F}$. In other words, the second derivative of the complementarity constraint over the axis $[x_1^*]_j = 0$ or $[x_2^*]_j = 0$ is zero. As a result, if MPEC-SOSC holds, then SOSC must hold for (MPEC-δ^*) because all other terms of the Hessians of the Lagrangians of both problems are the same and the sets of critical directions of both problems are the same. ◻

The corollary to Theorem 3.1 is much clearer, but it requires the additional condition that (x^*, s^*, y^*, z^*) is feasible for (MPEC)—in other words, the partitions x_1^* and x_2^* are nonnegative and complementary.

COROLLARY 3.2. *Suppose that δ^* satisfies* (3.9) *and that (x^*, s^*, y^*, v^*) is a solution of (MPEC-δ^*) such that* $\min(x_1^*, x_2^*) = 0$. *Then the point (x^*, y^*, z^*) is strongly stationary for (MPEC), where*

$$z^* = B(x^*)^T v^*. \tag{3.15}$$

Proof. Equation (3.15) is derived by comparing (2.1) with (3.7). ◻

3.5. Relaxation parameter updates. In this section we show how to construct a sequence of relaxation parameters δ_k such that $\lim_{k \to \infty} \delta_k = \delta^*$, where δ^* satisfies (3.8)–(3.9). We are guided by Theorem 3.1 in developing such a parameter update. Under certain conditions (discussed in section 3.6), we can recover the solution of the original MPEC from the solution of (MPEC-δ^*).

Suppose that $w_k = (x_k, s_k, y_k, v_k)$ is an estimate of the solution of (MPEC-δ_k), and let $z_k = B(x_k)^T v_k$ be the corresponding MPEC multipliers given by (3.15). Given an improved estimate w_{k+1}, Algorithm 1 defines a set of rules for updating the relaxation parameter vector δ_k. The algorithm also updates a companion sequence

Algorithm 1. RELAXATION PARAMETER UPDATE.

Input: $\delta_k, \delta_k^*, w_{k+1}, z_{k+1}$

Set fixed parameters $\kappa, \tau \in (0,1)$

1 [**Compute bounds for the KKT residual**]
$\underline{r}_{k+1}^* \leftarrow \|r(w_{k+1};\delta_k^*)\|^{1+\tau}$
$\overline{r}_{k+1}^* \leftarrow \|r(w_{k+1};\delta_k^*)\|^{1-\tau}$

for $i = 1, 2$ and $j = 1, \dots, n$ **do**
2   [**Update bound-constraint relaxations**]
  if $[z_{ik+1}]_j > \overline{r}_{k+1}^*$ **then**
    $[\delta_{ik+1}]_j \leftarrow \min(\kappa[\delta_{ik}]_j, \underline{r}_{k+1}^*)$
    $[\delta_{ik+1}^*]_j \leftarrow 0$
  else
    $[\delta_{ik+1}]_j \leftarrow [\delta_{ik}]_j$
    $[\delta_{ik+1}^*]_j \leftarrow [\delta_{ik}]_j$
3   [**Update complementarity-constraint relaxations**]
  if $[z_{1k+1}]_j < -\overline{r}_{k+1}^*$ or $[z_{2k+1}]_j < -\overline{r}_{k+1}^*$ **then**
    $[\delta_{ck+1}]_j \leftarrow \min(\kappa[\delta_{ck}]_j, \underline{r}_{k+1}^*)$
    $[\delta_{ck+1}^*]_j \leftarrow 0$
  else
    $[\delta_{ck+1}]_j \leftarrow [\delta_{ck}]_j$
    $[\delta_{ck+1}^*]_j \leftarrow [\delta_{ck}]_j$

return $\delta_{k+1}, \delta_{k+1}^*$

$\delta_k^* \equiv (0, \delta_{1k}^*, \delta_{2k}^*, \delta_{ck}^*)$ that defines a nearby relaxed problem (MPEC-δ_k^*). In the vicinity of the minimizer, this nearby relaxed problem gives an estimate of the active constraint set. Also, the residual of (MPEC-δ_k^*) is a better optimality measure than the residual of (MPEC-δ_k) because while all components of the relaxation parameter vector δ_k are strictly positive, some of the components of δ_k^* may be zero. The scalars $\underline{r}_k^*$ and $\overline{r}_k^*$ are lower and upper bounds on the KKT residual of (MPEC-δ_k^*); they provide a measure of nearness to zero of the MPEC multipliers and are used to predict the sign of the optimal MPEC multipliers.

3.6. Active-set identification. Suppose that δ_k^* is a set of relaxation parameters that satisfies (3.8) and that therefore defines a one-sided relaxation. Let $w_k^* = (x_k^*, s_k^*, y_k^*, v_k^*)$ be the minimizer of the associated relaxed problem (MPEC-δ_k^*) defined via (3.10), and let w_k be an estimate of w_k^*. If w_k is close enough to w^* and Algorithm 1 is given an improved estimate w_{k+1}, then it will return the same one-sided relaxation parameter $\delta_{k+1}^* = \delta_k^*$. Therefore, (MPEC-$\delta_k^*$) will remain fixed. Thus, the update rules continue to update (and reduce) the same relaxation parameters at every iteration—this property is used to guarantee that the feasible region remains nonempty even in the limit. In some sense, it implies that the correct active set is identified.

We make the following nondegeneracy assumptions about the MPEC minimizer (x^*, y^*, z^*). These assumptions hold throughout the remainder of the paper.

ASSUMPTION 3.3. *There exist strictly positive relaxation parameters δ such that the second derivatives of f and c are Lipschitz continuous over the set*

$$X_1 x_2 \le \delta_c, \qquad x_1 \ge -\delta_1, \qquad x_2 \ge -\delta_2.$$

ASSUMPTION 3.4. *The point* (x^*, y^*, z^*) *satisfies MPEC-LICQ for (MPEC).*

ASSUMPTION 3.5. *The point* (x^*, y^*, z^*) *satisfies MPEC-WSCS and -SSOSC for (MPEC).*

Theorem 3.6 proves that Algorithm 1 will leave the one-sided relaxation parameter unchanged if w_{k+1} improves the estimate w_k of w_k^*. Applied iteratively, the algorithm continues to update the same relaxation parameters δ_{ik} or δ_{ck}.

Note from (3.10) that the influence of δ_k^* on $w_k^* = (x_k^*, s_k^*, y_k^*, v_k^*)$ is relegated to only s_k^*. We may therefore write $w_k^* \equiv (x^*, s_k^*, y^*, v^*)$. We assume that δ_k^* satisfies (3.8). This implies that δ_k^* reveals the sign of the MPEC multipliers at the solution w^*. We also assume that the $(k+1)$th iterate w_{k+1} is closer to the minimizer than the kth iterate so that $\|w_{k+1} - w_k^*\| < \|w_k - w_k^*\|$. This assumption will hold whenever we apply a linearly convergent algorithm to compute w_{k+1} starting from w_k.

THEOREM 3.6. *Let* (x^*, y^*, z^*) *be a strongly stationary point of (MPEC) and suppose that Assumptions* 3.3–3.5 *hold. Moreover, assume that* δ_k^* *satisfies* (3.8) *and that* $[\delta_k^*]_j = [\delta_k]_j > 0$ *for all* j *such that* $[\delta_k^*]_j \neq 0$. *Let* $w_k^* = (x^*, s_k^*, y^*, v^*)$ *be the solution of the corresponding relaxation (MPEC-*δ_k^**) given by* (3.10)*, and assume that* $\|w_{k+1} - w_k^*\| < \|w_k - w_k^*\|$. *Then if* w_k *is close enough to* w^**, the parameter* δ_{k+1}^* *generated by Algorithm* 1 *satisfies* $\delta_{k+1}^* = \delta_k^*$.

Proof. We first show that $\bar{r}_{k+1}^*$ is bounded above and below by a finite multiple of $\|w_{k+1} - w_k^*\|^{1-\tau}$. By definition of w_k^* and δ_k^*, $r(w_k^*; \delta_k^*) = 0$. Also, by the hypothesis of this theorem, both w_k and w_{k+1} are close to w^*. Moreover, Assumption 3.3 implies that the KKT residual $r(w; \delta)$ is differentiable. This by Taylor's theorem implies that

$$r(w_{k+1}; \delta_k^*) = K(w_k^*)(w_{k+1} - w_k^*) + O(\|w_{k+1} - w_k^*\|^2), \tag{3.16}$$

where $K(w_k^*)$ is the Jacobian of the KKT residual $r(w; \delta)$ with respect to w evaluated at w_k^*. Note that this Jacobian does not depend on δ_k^*. In addition, as a consequence of Theorem 3.1, $K(w_k^*)$ is nonsingular. This together with (3.16), imply that there exist positive constants $\beta_2 > \beta_1$ such that for w_{k+1} in the vicinity of w_k^*

$$\beta_1 \|w_{k+1} - w_k^*\| \leq \|r(w_{k+1}; \delta_k^*)\| \leq \beta_2 \|w_{k+1} - w_k^*\|.$$

Then, by the definition of $\bar{r}_{k+1}^*$ (Step 1 of Algorithm 1) we have that

$$\beta_3 \|w_{k+1} - w_k^*\|^{1-\tau} \leq \bar{r}_{k+1}^* \leq \beta_4 \|w_{k+1} - w_k^*\|^{1-\tau}, \tag{3.17}$$

where $\beta_3 = \beta_1^{1-\tau}$ and $\beta_4 = \beta_2^{1-\tau}$.

Let $\epsilon \equiv \frac{1}{2} \min(|[z^*]_j| \mid \text{for all } j \text{ such that } [z^*]_j \neq 0)$. Then, condition (3.17) and the assumptions that w_k is close enough to w^* and $\|w_{k+1} - w_k^*\| < \|w_k - w_k^*\|$ imply that

$$\bar{r}_{k+1}^* < \epsilon. \tag{3.18}$$

Moreover, because $z_{k+1} = B(x_{k+1})^T v_{k+1}$, $z^* = B(x^*)^T v^*$, and $B(x)$ is Lipschitz continuous by Assumption 3.3, we have that for w_k close enough to w^* and $\|w_{k+1} - w_k^*\| < \|w_k - w_k^*\|$ the following holds:

$$\|z_{k+1} - z^*\| < \epsilon. \tag{3.19}$$

Consider the indices $i = 1, 2$ and $j = 1, \ldots, n$. Suppose that $[z^*]_j > 0$. Then (3.18) and (3.19) imply that

$$[z_{ik+1}]_j = [z^*]_j + ([z_{ik+1}]_j - [z^*]_j) > [z^*]_j - \epsilon \geq \epsilon > \bar{r}_{k+1}^*. \tag{3.20}$$

Suppose instead that $[z^*]_j < 0$. Then (3.18) and (3.19) imply that

$$-\bar{r}^*_{k+1} > -\epsilon > [z^*]_j + \epsilon = [z_{ik+1}]_j - ([z_{ik+1}]_j - [z^*]_j) + \epsilon > [z_{ik+1}]_j.$$

Finally, suppose that $[z^*]_j = 0$. Then because $\tau > 0$, we have that for w_k close enough to w^* and $\|w_{k+1} - w^*_k\| < \|w_k - w^*_k\|$

$$|[z_{ik+1}]_j| = |[z_{ik+1}]_j - [z^*]_j| \leq \|w_{k+1} - w^*_k\| \leq \beta_3 \|w_{k+1} - w^*_k\|^{1-\tau} \leq \bar{r}^*_{k+1}. \tag{3.21}$$

Because $\delta_k > 0$, the updates in Algorithm 1 imply that $\delta_{k+1} > 0$, and together with (3.20)–(3.21), we have that δ^*_{k+1} satisfies (3.8). This in turn implies that the set of indices j for which $[\delta^*_{k+1}]_j \neq 0$ coincides with the set of indices j for which $[\delta^*_k]_j \neq 0$. For this same set of indices, moreover, the parameter updates imply that $[\delta_{k+1}]_j = [\delta_k]_j$. Then because $[\delta^*_k]_j = [\delta_k]_j$ for such j, the update rules imply that $\delta^*_{k+1} = \delta^*_k$, as required. ∎

Note that $\delta^*_{k+1} = \delta^*_k$ implies that $w^*_{k+1} = w^*_k$; that is, w^*_k is also a local minimizer for the relaxed problem for the $(k+1)$th iterate.

4. An interior-point algorithm. The discussion thus far has not made use of a specific optimization algorithm. Theorem 3.6 makes use of an improved estimate of (MPEC-δ^*_k) but does not specify the manner in which it is computed. In this section we show how to construct a primal-dual interior-point algorithm that at each iteration will satisfy the conditions of Theorem 3.6. The parameter update rule in Algorithm 1 is invoked at each iteration of the interior method. The barrier parameter is updated simultaneously. This iteration scheme is repeated until certain convergence criteria are satisfied.

4.1. Algorithm summary. For the remainder of this section, we omit the dependence of each variable on the iteration counter k when the meaning of a variable is clear from its context. The search direction is computed by means of Newton's method on the KKT conditions of the barrier subproblem corresponding to (MPEC-δ). These are given by (3.7), where (3.7b) is replaced by

$$Sv - \mu e \equiv r_\mu = 0 \tag{4.1}$$

and $\mu > 0$ is the barrier parameter. An iteration of Newton's method based on (3.7) (where (4.1) replaces (3.7b)) computes a step direction by solving the system

$$K(w)\Delta w = -r(w; \mu, \delta), \tag{4.2}$$

where $\Delta w \equiv (\Delta x, \Delta s, \Delta y, \Delta v)$ and $r(w; \mu, \delta) \equiv (r_d, r_\mu, r_f, r_\delta)$ is the KKT residual of the barrier problem. (Note the identity $r(w; 0, \delta) \equiv r(w; \delta)$.) The Jacobian K is independent of the barrier *and* relaxation parameters—these appear only in the right-hand side of (4.2). This is a useful property because it considerably simplifies the convergence analysis in section 4.2.

To ensure that s and v remain strictly positive (as required by interior-point methods), each computed Newton step Δw may need to be truncated. Let γ be a steplength parameter such that $0 < \gamma < 1$. At each iteration we choose a steplength α so that

$$\alpha = \min(\alpha_s, \alpha_v), \tag{4.3}$$

where

$$\alpha_d = \min\left(1, \gamma \min_{[\Delta d]_j < 0} -[d]_j/[\Delta d]_j\right), \qquad d = \{s, v\}.$$

Algorithm 2. INTERIOR-POINT RELAXATION FOR MPECs.

Input: x_0, y_0, z_0
Output: x^*, y^*, z^*

[Initialize variables and parameters]
Choose starting vectors $s_0, v_0 > 0$. Set $w_0 = (x_0, s_0, y_0, v_0)$. Set the relaxation and barrier parameters $\delta_0, \mu_0 > 0$. Set parameters $0 < \kappa, \tau, \bar{\gamma} < 1$. Set the starting steplength parameter $\bar{\gamma} \le \gamma_0 < 1$. Set the convergence tolerance $\epsilon > 0$.

$k \leftarrow 0$
repeat
  [Compute the Newton step]
    Solve (4.2) for Δw_k
1 **[Truncate the Newton Step]**
    Determine the maximum steplength α_k, given by (4.3);
    $w_{k+1} \leftarrow w_k + \alpha_k \Delta w_k$
  [Compute MPEC multipliers]
    $z_{k+1} \leftarrow B(x_{k+1})^T v_{k+1}$
  [Update relaxation parameters]
    Compute $\delta_{k+1}, \delta^*_{k+1}$ with Algorithm 1
2 **[Update barrier and step parameters]**
    $\mu_{k+1} = \min(\kappa\mu_k, \underline{r}^*_{k+1})$ [Ensures that $\lim_{k\to\infty} \mu_k = 0$]
    $\gamma_{k+1} = \max(\bar{\gamma}, 1 - \mu_{k+1})$ [Ensures that $\lim_{k\to\infty} \gamma_k = 1$]
  $k \leftarrow k + 1$
until (4.4) holds;
$x^* \leftarrow x_k$; $y^* \leftarrow y_k$; $z^* \leftarrow z_k$
return x^*, y^*, z^*

Because our analysis focuses on the local convergence properties of the proposed algorithm, the $(k+1)$th iterate is computed as $w_{k+1} = w_k + \alpha \Delta w_k$. (A globalization scheme that can choose shorter steps is discussed in section 5.)

Algorithm 2 outlines the interior-point relaxation method. The method takes as a starting point the triple (x_0, y_0, z_0) as an estimate of a solution of the relaxed NLP corresponding to (MPEC). The algorithm terminates when the optimality conditions for (MPEC-δ^*_k) are satisfied, that is, when

$$\|r(w_k; \delta^*_k)\| < \epsilon \tag{4.4}$$

for some small and positive ϵ. Recall that $w^*_k = (x^*_k, s^*_k, y^*_k, v^*_k)$ is the solution to the one-sided relaxation (MPEC-δ^*_k); therefore, $\|r(w^*_k; \delta^*_k)\| = 0$. Note that we never compute w^*_k—it is used only as an analytical device.

4.2. Superlinear convergence. In this section we analyze the local convergence properties of the interior-point relaxation algorithm. The distinguishing feature of the proposed algorithm is the relaxation parameters and their associated update rules. If we were to hold the relaxation parameters constant, the relaxation method would reduce to a standard interior-point algorithm applied to a fixed relaxed MPEC; it would converge locally and superlinearly provided that the starting iterate is close to a nondegenerate minimizer of (MPEC-δ_k) (and that standard assumptions held). The main challenge is to show that the interior-point relaxation algorithm continues to converge locally and superlinearly even when the relaxation parameters change at each iteration. We use the shorthand notation $r_k \equiv r(w_k; \mu_k, \delta_k)$ and $r^*_k \equiv r(w_k; \delta^*_k)$.

THEOREM 4.1. *Let (x^*, y^*, z^*) be a strongly stationary point of (MPEC) and suppose that Assumptions* 3.3–3.5 *hold. Assume that δ_k^* satisfies* (3.8), *and let $w_k^* = (x^*, s_k^*, y^*, v^*)$ be the solution of the corresponding relaxation (MPEC-δ_k^*) given by* (3.10). *Then there exists $\epsilon > 0$ and $\beta > 0$ such that if Algorithm* 2 *is started with iterates at $k = 0$ that satisfy*

$$\|w_k - w_k^*\| < \epsilon, \tag{4.5}$$

$$\|\delta_k - \delta_k^*\| < \beta \|w_k - w_k^*\|^{1+\tau}, \tag{4.6}$$

$$\mu_k < \beta \|w_k - w_k^*\|^{1+\tau}, \tag{4.7}$$

$$1 - \gamma_k < \beta \|w_k - w_k^*\|^{1+\tau}, \tag{4.8}$$

and

$$[\delta_k^*]_j = [\delta_k]_j > 0 \quad \text{for all j such that} \quad [\delta_k^*]_j \neq 0, \tag{4.9}$$

then the sequence $\{w_k^*\}$ is constant over all k and $\{w_k\}$ converges Q-superlinearly to $w^* \equiv w_k^*$.

Proof. The proof has three parts. First, we show that there exists a constant $\sigma > 0$ such that $\|w_{k+1} - w_k^*\| \leq \sigma \|w_k - w_k^*\|^{1+\tau}$. Second, we show that $\delta_{k+1}^* = \delta_k^*$, and thus that w_k^* is also a minimizer to the relaxed MPEC corresponding to the $(k+1)$th iterate. Finally, we show that the conditions of the theorem hold also for the $(k+1)$th iterate. The main result therefore follows by induction.

Part 1. *$\|w_{k+1} - w_k^*\| \leq \sigma \|w_k - w_k^*\|^{1+\tau}$.* From Assumptions 3.3–3.5 and Theorem 3.1 we know that $K(w_k)$ is nonsingular for all $\epsilon > 0$ small enough, so that $\|K(w_k)^{-1}\|$ is bounded in the vicinity of w^*. Consider only such ϵ. Define the vector $\eta_k^* = (0, \mu_k e, 0, \delta_k^* - \delta_k)$ with components partitioned as per (3.7). (Note that $r_k = r_k^* - \eta_k^*$.) Then

$$\begin{aligned} w_{k+1} - w_k^* &= w_k - w_k^* - \alpha_k K(w_k)^{-1} r_k \\ &= (1 - \alpha_k)(w_k - w_k^*) + \alpha_k K(w_k)^{-1}(K(w_k)(w_k - w_k^*) - r_k^* + \eta_k^*) \\ &= (1 - \alpha_k)(w_k - w_k^*) \\ &\quad + \alpha_k K(w_k)^{-1} \eta_k^* + \alpha_k K(w_k)^{-1}(K(w_k)(w_k - w_k^*) - r_k^*). \end{aligned} \tag{4.10}$$

Each term on the right-hand side of (4.10) can be bounded as follows. Because (x^*, y^*, z^*) is a strongly stationary point of (MPEC) satisfying assumptions 3.3–3.5, Theorem 3.1 applies. Therefore, w^* satisfies LICQ, SCS, and SOSC for (MPEC-δ^*). Then by Lemma 5 of [24] we know that there exists a positive constant ϵ_1 such that $|1 - \alpha_k| \leq 1 - \gamma_k + \epsilon_1 \|\Delta w_k\|$. Therefore

$$\|(1 - \alpha_k)(w_k - w_k^*)\| \leq \big((1 - \gamma_k) + \epsilon_1 \|\Delta w_k\|)\big) \|w_k - w_k^*\|. \tag{4.11}$$

We now further bound the right-hand side of (4.11). Because $\|K(w_k)^{-1}\|$ is bounded for ϵ small enough, there exists a positive constant ϵ_2 such that

$$\|\Delta w_k\| = \|K(w_k)^{-1}(-r_k^* + \eta_k^*)\| \leq \epsilon_2 (\|r_k^*\| + \|\eta_k^*\|). \tag{4.12}$$

Assumption 3.3 implies that the KKT residual $r(w; \mu, \delta)$, and thus, $r(w; \delta)$, is differentiable. Hence there exists a positive constant ϵ_3 such that

$$\|r_k^*\| = \|r(w_k; \delta_k^*) - r(w_k^*; \delta_k^*)\| \leq \epsilon_3 \|w_k - w_k^*\|. \tag{4.13}$$

Moreover, (4.6) and (4.7) imply that there exists a positive constant ϵ_4 such that

$$\|\eta_k^*\| \leq \epsilon_4 \|w_k - w_k^*\|^{1+\tau}. \tag{4.14}$$

Then substituting (4.12), (4.13), (4.14), and condition (4.8), into (4.11) we have

$$\|(1-\alpha_k)(w_k - w_k^*)\| \leq (\beta + \epsilon_1\epsilon_2\epsilon_4)\|w_k - w_k^*\|^{2+\tau} + \epsilon_1\epsilon_2\epsilon_3\|w_k - w_k^*\|^2. \tag{4.15}$$

From the boundedness of $\|K(w_k)^{-1}\|$ around w_k^* and (4.14), the second term in (4.10) satisfies

$$\|\alpha_k \; K(w_k)^{-1}\eta_k^*\| \leq \alpha_k \|K(w_k)^{-1}\| \; \|\eta_k^*\| \leq \epsilon_5\|w_k - w_k^*\|^{1+\tau}$$

for some positive constant ϵ_5. Finally, the third term in (4.10) satisfies (using Taylor's theorem and again the fact that $\|K(w_k)^{-1}\|$ is bounded around w_k^*)

$$\|\alpha_k \; K(w_k)^{-1}(K(w_k)(w_k - w_k^*) - r_k^*)\| \leq \epsilon_6\|w_k - w_k^*\|^2 \tag{4.16}$$

for some positive constant ϵ_6. Hence, (4.10) and (4.15)–(4.16) yield

$$\|w_{k+1} - w_k^*\| \leq \sigma\|w_k - w_k^*\|^{1+\tau} \tag{4.17}$$

for some positive constant σ, as required.

Part 2. $\delta_{k+1}^* = \delta_k^*$. Note that by (4.17) we know that for ϵ small enough the assumptions of Theorem 3.6 hold and therefore $\delta_{k+1}^* = \delta_k^*$. As a result, w_k^* is also a minimizer of (MPEC-δ_{k+1}^*).

Part 3. *The theorem hypotheses also hold for the* $(k+1)$*th iterate.* As $\delta_{k+1}^* = \delta_k^*$, then δ_{k+1}^* satisfies (3.8). Moreover, (4.17) implies for ϵ small enough that (4.5) holds for w_{k+1}. Because $r(w;\mu,\delta)$ is differentiable, Theorem 3.1 implies that $K(w)$, the Jacobian of $r(w;\mu,\delta)$ with respect to w, is bounded in the vicinity of w_k^*. Together with the definition of $\overline{r}_{k+1}^*$, the fact that $\delta_{k+1}^* = \delta_k^*$, Steps 2 and 3 of Algorithm 1, and Step 2 of Algorithm 2, this implies that (4.6)–(4.9) hold for δ_{k+1}, μ_{k+1}, and γ_{k+1}.

The proof finishes noting that $w_{k+1}^* = w_k^*$ because $\delta_{k+1}^* = \delta_k^*$, so that by induction, $w^* = w_k^*$ for all iterations $k+1, k+2, \ldots$. The superlinear convergence of w_k to w^* then follows by induction from (4.17). ∎

Note that in addition to the assumptions made in Theorem 3.6, we assume that the barrier and steplength parameters satisfy $\mu_k < \beta\|w_k - w_k^*\|^{1+\tau}$ and $1 - \gamma_k < \beta\|w_k - w_k^*\|^{1+\tau}$ for some $\tau \in (0,1)$ and $\beta > 0$. These are standard assumptions used to prove superlinear convergence of interior methods. They imply the barrier and steplength parameters are updated fast enough. In addition, we assume that $\|\delta_k - \delta_k^*\| < \beta\|w_k - w_k^*\|^{1+\tau}$. This assumption implies that the distance between δ_k and δ_k^* is small compared to the distance between the current iterate w_k and the minimizer w^*. Note that in Part 3 of the proof of Theorem 4.1, we show that this assumption will hold when the relaxation parameter update rule in Algorithm 1 is applied for two or more iterations. Finally, the technical Assumption 4.9 simplifies the proof and that is also satisfied whenever Algorithm 1 is applied for two or more consecutive iterations.

5. Implementation details. In this section we discuss two practical aspects of our implementation. First, to globalize the interior-point method, we perform a backtracking linesearch on an augmented Lagrangian merit function (although other globalization schemes could be used). The theoretical properties of this merit function have been analyzed by Moguerza and Prieto [14]. We also modify the Jacobian $K(w)$ as in [23] to ensure a sufficient descent direction for the augmented Lagrangian merit function.

Second, we make use of a safeguard to the relaxation parameter update that prevents the algorithm from converging to stationary points of the relaxed MPEC that are not feasible with respect to MPEC. To see how this may happen, again consider the example MPEC (3.3). The relaxed MPEC (with slack variables) is given by

$$\begin{array}{ll} \underset{x_1,x_2,s_1,s_2,s_c \in \mathrm{R}}{\text{minimize}} & \frac{1}{2}(x_1-a_1)^2+\frac{1}{2}(x_2-a_2)^2 \\ \text{subject to} & s_1 - x_1 = \delta_1, \\ & s_2 - x_2 = \delta_2, \\ & s_c + X_1 x_2 = \delta_c, \\ & s \geq 0. \end{array} \tag{5.1}$$

For $a_1 = a_2 = 0.01$, $\delta_c = 1$, and $\delta_1 = \delta_2 = 0$, the point

$$(x_1, x_2, s_1, s_2, s_c) = (0.01, 0.01, 0.01, 0.01, 0.9999)$$

with multipliers $(v_1, v_2, v_c) = (0, 0, 0)$ is clearly a stationary point of (5.1), but it is not feasible for (3.3). However, note that a point $(x_0, x_1, x_2, s_0, s_1, s_2, s_c)$ feasible for (MPEC-δ) is feasible for (MPEC) if and only if

$$(s_0, s_1, s_2, s_c) = (x_0, x_1 + \delta_1, x_2 + \delta_2, \delta_c) \tag{5.2}$$

(cf. (3.10a)). To ensure that (5.2) always holds at the limit point, we propose a modification of the bound-constraint relaxations in Steps 2 and 3 of Algorithm 1. The proposed modification is the following:

$$\begin{array}{llll} [\delta_{ik+1}]_j = \min(\kappa[\delta_{ik}]_j, \underline{r}^*_{k+1}) & \text{if} \quad [z_{ik+1}]_j > \overline{r}^*_{k+1}, & & \\ [\delta_{ik+1}]_j = \min([\delta_{ik}]_j, [s_{ik}]_j) & \text{if} \quad [z_{ik+1}]_j \leq \overline{r}^*_{k+1}, & & \\ [\delta_{ck+1}]_j = \min(\kappa[\delta_{ck}]_j, \underline{r}^*_{k+1}) & \text{if} \quad [z_{1k+1}]_j < -\overline{r}^*_{k+1} & \text{or} & [z_{2k+1}]_j < -\overline{r}^*_{k+1}, \\ [\delta_{ck+1}]_j = \min([\delta_{ck}]_j, [s_{ck}]_j) & \text{if} \quad [z_{1k+1}]_j \geq -\overline{r}^*_{k+1} & \text{and} & [z_{2k+1}]_j \geq -\overline{r}^*_{k+1} \end{array}$$

for $i = 1, 2$ and $j = 1, \ldots, n$.

Thus, the above parameter update prevents the algorithm from converging to spurious stationary points for the relaxed MPEC that are not stationary for the MPEC.

Finally, it is possible to show that the local convergence results of previous sections still hold when using both the globalization strategy for the interior point method and the safeguard of the relaxation parameter update. But to simplify the exposition, we have decided to leave these two aspects out of the local convergence analysis of previous sections.

6. Numerical results. We illustrate in this section the numerical performance of the interior-point relaxation algorithm on the MacMPEC test problem set [11]. The results confirm our local convergence analysis and show that our implementation performs well in practice.

The interior-point relaxation algorithm has been implemented as a MATLAB program. Problems from the MacMPEC test suite (coded in AMPL [11]) are accessed via a MATLAB MEX interface. Because the algorithm has been implemented using dense linear algebra, we apply the method to a subset of 87 small- to medium-size problems from the MacMPEC test suite.

We stop the algorithm under three different circumstances: (i) if the iteration limit of 150 is exceeded; (ii) if the current iterate is a stationary point of (MPEC-δ^*), i.e., if $\|r(w_k; 0, \delta_k^*)\| < 10^{-6}(1 + \|\nabla f(x_k)\|)$ (cf. (4.4)); or (iii) if the steplength is too small. We use the following parameter values for the barrier and relaxation updates: $\tau = 0.3$ and $\kappa = 0.9$.

Table 6.1 gives information regarding the performance of our algorithm on each test problem. The first column indicates the name of the problem, the second and third columns indicate the number of iterations and function evaluations, the fourth column shows the final objective function value, the fifth and sixth columns show the norm of the multiplier vector v_c and the norm of the KKT residual of the nearby relaxed MPEC (MPEC-δ^*) at the solution, and the last two columns indicate the exit status of the algorithm. The exit flags are described in Table 6.2. The quantities $(\delta_1^*, \delta_2^*, \delta_c^*)$ are the final values of the relaxation parameters.

The results seem to confirm that the global convergence safeguards proposed in section 5 are effective in practice. In particular, the algorithm converges to a strongly stationary point of (MPEC) for most of the test problems in the collection, that is, flag1 = 1 for most of the problems. Moreover, note that all stationary points of (MPEC-δ^*) found by the algorithm are also strongly stationary for the original MPEC; that is, flag1 is never equal to 2. Finally, some of the problems on which our algorithm fails are ill-posed according to [18, 4, 3]. For instance, *ex9.2.2*, *qpec2*, *ralph1*, *scholtes4*, and *tap-15* do not have a strongly stationary point, the *pack* problems have an empty strictly feasible region, *ralphmod* is unbounded, and *design-cent-3* is infeasible.

TABLE 6.1
Performance of the interior-point relaxation algorithm on the selected MacMPEC test problems.

Problem	iter	nfe	f	$\|v_c^*\|$	$\|r\|$	flag1	flag2
bar-truss-3	36	73	1.017e+04	4.521e+00	4.543e−04	1	1
bard1	13	27	1.700e+01	7.621e−01	4.170e−04	1	1
bard2	66	133	6.163e+03	1.036e+01	5.221e−05	1	1
bard3	16	33	-1.268e+01	3.625e−01	3.225e−06	1	1
bard1m	88	397	1.700e+01	1.504e−03	1.373e−04	1	0
bard2m	66	133	-6.598e+03	1.128e−04	5.444e−05	1	1
bard3m	16	33	-1.268e+01	1.350e+00	4.770e−06	1	1
bilevel1	16	33	5.000e+00	8.700e−02	1.382e−06	1	1
bilevel2	67	135	-6.600e+03	3.848e−01	3.174e−04	1	1
bilevel3	83	277	-8.636e+00	4.587e−03	8.352e−04	1	0
bilin	24	49	-1.215e−04	1.996e+00	1.513e−03	1	0
dempe	17	35	3.125e+01	5.002e+00	3.619e−06	1	1
design-cent-2	150	774	-3.182e−15	2.024e−05	3.749e+02	0	1
design-cent-3	150	2649	3.546e−02	1.930e+00	7.977e+00	0	1
design-cent-4	99	425	1.508e−18	3.616e−04	1.027e−08	1	1
ex9.1.1	19	39	-1.300e+01	1.087e+00	1.343e−03	1	0
ex9.1.2	14	29	-6.250e+00	1.902e+00	1.110e−03	1	0
ex9.1.3	39	80	-2.920e+01	5.357e+00	4.327e−03	1	1
ex9.1.4	33	80	-3.700e+01	1.999e+00	1.389e−07	1	1
ex9.1.5	11	23	-1.000e+00	3.674e+00	6.727e−06	1	1
ex9.1.6	22	47	-1.500e+01	1.000e+00	1.848e−05	1	0

TABLE 6.1
Cont'd.

Problem	iter	nfe	f	$\|v_c^*\|$	$\|r\|$	flag1	flag2
ex9.1.7	87	310	-2.600e+01	2.001e+00	1.497e−03	1	0
ex9.1.8	102	441	-3.250e+00	3.180e+00	1.694e−01	1	0
ex9.1.9	26	63	3.111e+00	2.678e+00	3.081e−03	1	1
ex9.1.10	102	441	-3.250e+00	3.180e+00	1.694e−01	1	0
ex9.2.1	19	39	1.700e+01	2.881e+00	7.365e−04	1	0
ex9.2.2	150	655	1.000e+02	7.374e+03	1.402e−02	0	1
ex9.2.3	16	33	5.000e+00	4.700e−09	2.093e−08	1	1
ex9.2.4	10	21	5.000e−01	1.000e+00	1.778e−08	1	1
ex9.2.5	13	27	9.000e+00	6.185e+00	1.646e−06	1	1
ex9.2.6	66	239	-1.000e+00	7.071e−01	1.981e−02	1	0
ex9.2.7	19	39	1.700e+01	2.881e+00	7.365e−04	1	0
ex9.2.8	12	25	1.500e+00	5.000e−01	1.080e−06	1	1
ex9.2.9	13	27	2.000e+00	1.987e+00	4.019e−08	1	1
flp2	22	49	1.076e−17	1.517e−05	3.595e−04	1	1
flp4-1	35	75	5.411e−07	1.315e−06	2.607e−06	1	1
flp4-2	41	89	7.376e−07	4.076e−06	8.233e−06	1	1
flp4-3	52	126	1.018e−06	1.913e−06	3.905e−06	1	1
flp4-4	56	117	2.456e−06	7.803e−07	8.825e−06	1	1
gauvin	11	23	2.000e+01	2.500e−01	8.152e−07	1	1
hakonsen	150	351	1.113e+01	4.898e−05	2.825e−01	0	1
hs044-i	83	279	3.765e+01	2.271e+00	1.344e−01	1	0
incid-set1-8	54	117	5.016e−06	1.722e−04	3.536e−06	1	1
incid-set1c-8	101	210	4.554e−06	9.816e−04	3.754e−06	1	1
incid-set2-8	149	302	8.929e+00	2.069e+03	2.363e+04	7	1
jr1	8	17	5.000e−01	5.779e−09	1.259e−08	1	1
jr2	8	17	5.000e−01	2.000e+00	2.282e−08	1	1
kth1	9	19	3.950e−07	7.046e−07	5.989e−07	1	1
kth2	8	17	1.432e−09	1.355e−07	2.180e−07	1	1
kth3	7	15	5.000e−01	1.000e+00	9.131e−07	1	1
liswet1-050	36	89	1.399e−02	2.552e−09	5.998e−09	1	1
nash1	26	53	1.339e−07	2.499e−04	3.930e−04	1	1
outrata31	88	184	3.208e+00	3.234e+01	3.653e−07	1	0
outrata32	86	177	3.449e+00	6.586e+01	4.908e−07	1	0
outrata33	83	174	4.604e+00	6.089e+02	2.808e−06	1	0
outrata34	107	218	6.593e+00	8.386e+00	1.549e−06	1	0
pack-comp1-8	97	818	6.240e−01	5.388e+01	5.923e+04	7	1
pack-comp1c-8	126	300	5.741e−01	1.308e+01	2.099e+04	7	0
pack-comp1p-8	135	347	-3.649e+04	3.230e+03	1.383e+05	7	1
pack-comp2-8	38	82	7.724e−01	2.677e+01	1.039e+04	7	1
pack-comp2c-8	150	309	6.537e−01	6.595e+00	2.979e+04	0	1
pack-rig1-8	150	1109	6.623e−01	6.294e+00	1.562e+03	0	1
pack-rig1c-8	61	174	6.013e−01	5.803e+00	4.770e+03	7	1
pack-rig1p-8	150	948	-4.048e+01	1.621e+01	4.220e+03	0	0
pack-rig2-8	150	307	7.804e−01	8.259e−09	9.463e−04	0	1
pack-rig2c-8	75	289	6.046e−01	5.751e+00	4.974e+03	7	0
pack-rig2p-8	147	403	-1.573e+02	1.093e+00	2.086e+02	7	1
portfl-i-1	28	59	2.096e−06	4.971e−04	5.158e−04	1	1
portfl-i-2	30	61	1.099e−06	8.256e−03	2.070e−03	1	1
portfl-i-3	31	64	1.743e−06	3.498e−02	1.864e−04	1	1
portfl-i-4	31	64	2.755e−06	1.418e−02	4.518e−04	1	1
portfl-i-6	28	58	2.394e−06	3.893e−02	4.654e−04	1	1
qpec-100-1	80	163	9.900e−02	1.762e+01	7.324e−06	1	1
qpec1	10	21	8.000e+01	3.044e−07	5.138e−07	1	1
qpec2	150	303	4.500e+01	9.669e+04	2.425e−02	0	1
ralph1	150	303	-1.563e−05	3.191e+04	1.885e−03	0	1
ralph2	15	31	-2.228e−07	2.001e+00	3.071e−07	1	1
ralphmod	75	151	-5.726e+02	8.219e+02	1.167e+02	7	0

TABLE 6.1
Cont'd.

Problem	iter	nfe	f	$\|v_c^*\|$	$\|r\|$	flag1	flag2
scholtes1	10	21	2.000e+00	1.008e−08	2.302e−08	1	1
scholtes2	21	43	1.500e+01	6.894e−06	2.881e−06	1	1
scholtes3	8	18	5.000e−01	1.000e+00	5.044e−07	1	1
scholtes4	150	301	-4.994e−05	3.994e+04	3.895e−03	0	1
scholtes5	8	17	1.000e+00	1.870e+00	1.277e−06	1	1
sl1	30	61	1.003e−04	3.337e−07	1.715e−05	1	1
stackelberg1	12	25	-3.267e+03	8.998e−01	5.536e−06	1	1
tap-09	106	320	1.546e+02	1.964e−01	8.807e−04	1	0
tap-15	136	300	3.131e+02	2.664e−01	3.389e−02	7	1

TABLE 6.2
Exit flags in Table 6.1. *The second exit flag indicates when the final relaxation parameters* $(\delta_c^*, \delta_1^*, \delta_2^*)$ *satisfy the complementarity condition given by* (3.9).

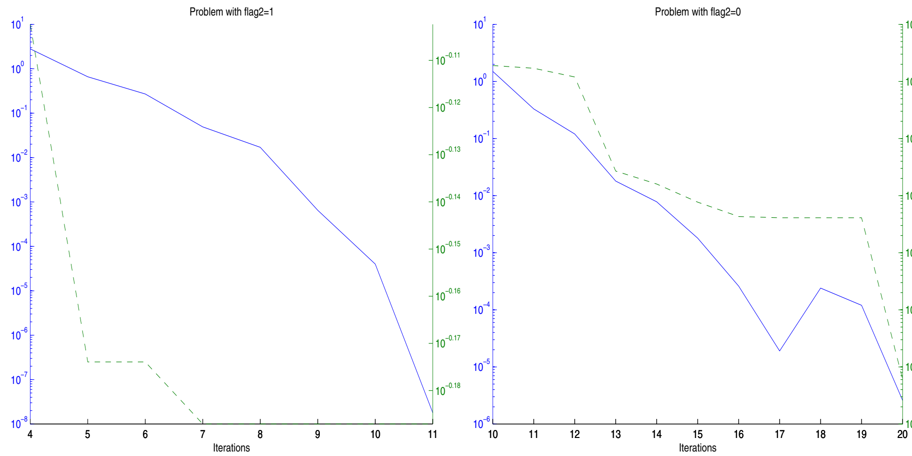
flag1	Status
0	Terminated by iteration limit (150)
1	Found stationary point of (MPEC-δ^*) and strongly stationary point of (MPEC)
2	Found stationary point of (MPEC-δ^*) but not strongly stationary point of (MPEC)
7	Terminated because steplength too small ($\alpha_k < 10^{-12}$) or descent direction not found

flag2	Status
0	$\max(\delta_c^*, \delta_i^*) = 0$
1	$\max(\delta_c^*, \delta_i^*) > 0$

In addition, we have observed that the algorithm is particularly efficient on those problems for which the iterates converge to a strongly stationary point that satisfies the MPEC-WSCS and -SSOSC. For these problems, in particular, the final relaxation parameter satisfy $\max\left(\delta_c^*, \min(\delta_1^*, \delta_2^*)\right) > 0$ and the iterates converge at a superlinear rate. On the other hand, for those problems for which the algorithm converges to a strongly stationary point that does not satisfy the MPEC-WSCS and -SSOSC, there is a zero or very small component of $\max\left(\delta_c^*, \min(\delta_1^*, \delta_2^*)\right)$, and the iterates converge only at a linear rate. In other words, when $\max\left(\delta_c^*, \min(\delta_1^*, \delta_2^*)\right) > 0$ (i.e., flag2 $= 1$), the condition number of the KKT matrix remains bounded and the algorithm converges superlinearly. On the other hand, when $\max\left(\delta_c^*, \min(\delta_1^*, \delta_2^*)\right) = 0$ (i.e., flag2 $= 0$), the condition number of the KKT matrix grows large, and the algorithm converges only linearly.

This behavior can be observed in Figure 6.1, which depicts the evolution of $\|r_k^*\|$ and the minimum value of the vector $\max\left(\delta_c^*, \min(\delta_1^*, \delta_2^*)\right)$ for two problems of the MacMPEC collection. Both vertical axes are in a logarithmic (base 10) scale. The first subfigure shows the last eight iterates generated by the algorithm for problem ex9.2.4 (which confirms $\max\left(\delta_c^*, \min(\delta_1^*, \delta_2^*)\right) > 0$). The second subfigure shows the last 11 iterates generated by the algorithm for problem ex9.2.7 (which confirms a numerically zero component of $\max\left(\delta_c^*, \min(\delta_1^*, \delta_2^*)\right)$.

Moreover, the numerical results confirm the relevance of our relaxing the MPEC-SCS assumption in our analysis. In particular, there are eight problems (approximately 10% of the total) for which the MPEC-SCS does not hold at the minimizer (although MPEC-WSCS and -SSOSC hold) and yet $\max\left(\delta_c^*, \min(\delta_1^*, \delta_2^*)\right) > 0$ in the limit. Likewise, we have confirmed that for all problems for which the minimum value of the vector $\max\left(\delta_c^*, \min(\delta_1^*, \delta_2^*)\right)$ is zero, the algorithm converges to points where the MPEC-WSCS or -SSOSC do not hold.



(a) Problem ex9.2.4

(b) Problem ex9.2.7

Fig. 6.1. *Final iterations of two problems from the MacMPEC test suite. Each graph shows the KKT residual* $\|r_k^*\|$ *(solid line and left axis) and the minimum value of the vector* $\max\left(\delta_c^*, \min(\delta_1^*, \delta_2^*)\right)$ *(dashed line and right axis) against the iteration count.*

Acknowledgments. We are grateful to Sven Leyffer and to three anonymous referees for their detailed feedback and suggestions. We are also grateful to Sven Leyffer for his effort in assembling and maintaining the excellent MacMPEC test-problem set. Many people have given us helpful advice along the way, but we would like to especially thank Mihai Anitescu, Jorge Nocedal, and Danny Ralph for their valuable input and encouragement.

REFERENCES

[1] M. Anitescu, *On Solving Mathematical Programs with Complementarity Constraints as Nonlinear Programs*, Tech. Report ANL/MCS-P864-1200, Argonne National Laboratory, Argonne, IL, 2000.

[2] M. Anitescu, *Global Convergence of an Elastic Mode Approach for a Class of Mathematical Programs with Complementarity Constraints*, Tech. Report ANL/MCS-P1143-0404, Argonne National Laboratory, Argonne, IL, 2004.

[3] H. Y. Benson, A. Sen, D. F. Shanno, and R. J. Vanderbei, *Interior-Point Algorithms, Penalty Methods and Equilibrium Problems*, Tech. Report ORFE-03-02, Operations Research and Financial Engineering, Princeton University, Princeton, NJ, 2003.

[4] H. Y. Benson, D. F. Shanno, and R. J. Vanderbei, *Interior-Point Methods for Nonconvex Nonlinear Programming: Complementarity Constraints*, Tech. Report ORFE-02-02, Operations Research and Financial Engineering, Princeton University, Princeton, NJ, 2002.

[5] M. C. Ferris and J. S. Pang, *Engineering and economic applications of complementarity problems*, SIAM Rev., 39 (1997), pp. 669–713.

[6] R. Fletcher, S. Leyffer, D. Ralph, and S. Scholtes, *Local Convergence of SQP Methods for Mathematical Programs with Equilibrium Constraints*, Tech. Report NA-209, University of Dundee, Dundee, Scotland, UK, 2002.

[7] R. Fletcher and S. Leyffer, *Numerical Experience with Solving MPECs as NLPs*, Tech. Report NA-210, University of Dundee, Dundee, Scotland, UK, August, 2002.

[8] M. P. Friedlander and M. A. Saunders, *A globally convergent linearly constrained Lagrangian method for nonlinear optimization*, SIAM J. Optim., 15 (2004), pp. 863–897.

[9] X. Hu and D. Ralph, *Convergence of a penalty method for mathematical programming with complementarity constraints*, J. Optim. Theory Appl., 123 (2004), pp. 365–390.

[10] S. Leyffer, G. Lopez-Calva, and J. Nocedal, *Interior Methods for Mathematical Programs with Complementarity Constraints*, Tech. Report ANL/MCS-P-1211-1204, Argonne National Laboratory, Argonne, IL, 2004.
[11] S. Leyffer, *MacMPEC: AMPL collection of MPECs*. http://www.mcs.anl.gov/~leyffer/MacMPEC (April 2004).
[12] X. Liu and J. Sun, *Generalized Stationary Points and an Interior Point Method for Mathematical Programs with Equilibrium Constraints*, Tech. Report, National University of Singapore, Singapore, 2001.
[13] Z. Luo, J. Pang, and D. Ralph, *Mathematical Programs with Equilibrium Constraints*, Cambridge University Press, London, 1996.
[14] J. Moguerza and F. Prieto, *An augmented lagrangian interior-point method using directions of negative curvature*, Math. Program. A, 95 (2003), pp. 573–616.
[15] B. A. Murtagh and M. A. Saunders, *MINOS* 5.5 *User's Guide*, Tech. Report 83-20R, Systems Optimization Laboratory, Department of Management Science and Engineering, Stanford University, Stanford, CA, December, 1983.
[16] J. Nocedal and S. J. Wright, *Numerical Optimization*, Springer-Verlag, New York, 1999.
[17] J. Outrata, M. Kocvara, and J. Zowe, *Nonsmooth Approach to Optimization Problems with Equilibrium Constraints: Theory, Applications, and Numerical Results*, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1998.
[18] A. U. Raghunathan and L. T. Biegler, *Interior Point Methods for Mathematical Programs with Complementarity Constraints*, Tech. Report, Department of Chemical Engineering, Carnegie Mellon University, Pittsburgh, PA, 2003.
[19] D. Ralph and S. J. Wright, *Some Properties of Regularization Schemes for MPECs*, Tech. Report 03-04, Computer Sciences, University of Wisconsin, Madison, Wisconsin, December, 2003.
[20] H. Scheel and S. Scholtes, *Mathematical programs with complementarity constraints: Stationarity, optimality, and sensitivity*, Math. Operat. Res., 25 (2000), pp. 1–22.
[21] S. Scholtes and M. Stöhr, *Exact penalization of mathematical programs with equilibrium constraints*, SIAM J. Control and Optim., 37 (1999), pp. 617–652.
[22] S. Scholtes, *Convergence properties of a regularization scheme for mathematical programs with complementarity constraints*, SIAM J. Optim., 11 (2001), pp. 918–936.
[23] D. Shanno and R. Vanderbei, *An interior-point algorithm for nonconvex nonlinear programming*, Comput. Optim. Appl., 13 (1999), pp. 231–252.
[24] H. Yamashita and H. Yabe, *Superlinear and quadratic convergence of some primal-dual interior point methods for constrained optimization*, Math. Program., 75 (1996), pp. 377–397.