UNIVERSIDAD CARLOS III DE MADRID working papers

Working Paper 04-24
Statistics and Econometrics Series 08
April 2004

Departamento de Estadística
Universidad Carlos III de Madrid
Calle Madrid, 126
28903 Getafe (Spain)
Fax (34) 91 624-98-49

AN INTERIOR-POINT METHOD FOR MPECs BASED ON STRICTLY FEASIBLE RELAXATIONS.

Angel Víctor de Miguel, Michael P. Friedlander, Francisco J. Nogales and Stefan Scholtes.*

Abstract

An interior-point method for solving mathematical programs with equilibrium constraints (MPECs) is proposed. At each iteration of the algorithm, a single primal-dual step is computed from each subproblem of a sequence. Each subproblem is defined as a relaxation of the MPEC with a nonempty strictly feasible region. In contrast to previous approaches, the proposed relaxation scheme preserves the nonempty strict feasibility of each subproblem even in the limit.

Local and superlinear convergence of the algorithm is proved even with a less restrictive strict complementarity condition than the standard one. Moreover, mechanisms for inducing global convergence in practice are proposed. Numerical results on the MacMPEC test problem set demonstrate the fast-local convergence properties of the algorithm.

Keywords: Nonlinear programming, mathematical programs with equilibrium constraints, constrained minimization, interior-point methods, primal-dual methods, barrier methods.

*de Miguel, A.V. London Business School, U.K., e-mail: avmiguel@london.edu; Friedlander, Mathematics and Computer Science Division, Argonne National Laboratory, IL, USA, e-mail: michael@mcs.anl.gov; Nogales, Department of Statistics, Universidad Carlos III de Madrid, e-mail: FcoJavier.Nogales@uc3m.es; Scholtes, Judge Institute of Management, Cambridge University, U.K., e-mail: s.scholtes@jims.cam.ac.uk.

AN INTERIOR-POINT METHOD FOR MPECS BASED ON STRICTLY FEASIBLE RELAXATIONS

ANGEL–VICTOR DEMIGUEL[*], MICHAEL P. FRIEDLANDER[†], FRANCISCO J. NOGALES[‡], AND STEFAN SCHOLTES[§]

Abstract. An interior-point method for solving mathematical programs with equilibrium constraints (MPECs) is proposed. At each iteration of the algorithm, a single primal-dual step is computed from each subproblem of a sequence. Each subproblem is defined as a relaxation of the MPEC with a nonempty strictly feasible region. In contrast to previous approaches, the proposed relaxation scheme preserves the nonempty strict feasibility of each subproblem even in the limit.

Local and superlinear convergence of the algorithm is proved even with a less restrictive strict complementarity condition than the standard one. Moreover, mechanisms for inducing global convergence in practice are proposed. Numerical results on the MacMPEC test problem set demonstrate the fast-local convergence properties of the algorithm.

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

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

1. Introduction.

Mathematical programs with equilibrium constraints are optimization problems that incorporate responses of equilibrium systems to changes of system design parameters. Often, system equilibria have variational characterizations that, under suitable regularity and convexity assumptions, admit necessary and sufficient first-order conditions and thus characterizations of equilibrium responses as solution sets of equations and inequalities. An optimization problem involving equilibrium response functions is thereby turned into a mathematical program with equilibrium constraints (MPECs). The resulting constrained optimization problems resemble ordinary nonlinear programs (NLPs) with one notable difference: they typically involve complementarity conditions stemming from complementary slackness in the first-order optimality conditions for the equilibrium response.

We consider MPECs of the general form

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

where $x = (x_0, x_1, x_2) \in \mathbb{R}^{p \times n \times n}$, $f : \mathbb{R}^{p+2n} \to \mathbb{R}$ is a linear or nonlinear objective function, and $c : \mathbb{R}^{p+2n} \to \mathbb{R}^m$ is a vector of linear or nonlinear constraint functions;

[*] London Business School, UK (avmiguel@london.edu).

[†] Mathematics and Computer Science Division, Argonne National Laboratory, 9700 South Cass Avenue, Argonne, IL 60439-4844 (michael@mcs.anl.gov). The work of this author was 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, Spain (FcoJavier.Nogales@uc3m.es).

[§] Judge Institute of Management, Cambridge University, UK (s.scholtes@jims.cam.ac.uk).
Version: April 29, 2004

the familiar complementarity constraint $\min(x_1, x_2) = 0$ requires that either $[x_1]_j$ or $[x_2]_j$ vanishes for each component $j = 1, \ldots, n$.

In addition to their prevalence in first-order optimality conditions, complementarity conditions can arise directly in equilibrium models. For example, an economic equilibrium requires that either the price for a product is zero or excess supply is zero; in a mechanical model we may require that either the distance between objects is zero or the force between them vanishes. We refer to the survey paper by Ferris and Pang [5] for further examples of complementarity models and to the monographs by Luo et al. [18] and Outrata et al. [23] for more details on MPEC theory and applications.

1.1. Other work on MPECs.

MPECs can be reformulated as standard NLPs by replacing the nonsmooth equation $\min(x_1, x_2) = 0$ by the equivalent smooth complementarity constraints $x_1, x_2 \geq 0, x_1 \circ x_2 = 0$, where $\circ$ denotes the Hadamard, or componentwise, product of two vectors. However, a key difficulty in solving MPECs as NLPs is the failure of standard constraint qualifications, most notably that of Mangasarian and Fromovitz [19]. The Mangasarian-Fromovitz constraint qualification (MFCQ) implies the existence of feasible points that satisfy all inequalities strictly, and it is thus in contradiction to the smooth complementarity constraints. This is worrying because MFCQ, and often the stronger linear independence constraint qualification (LICQ), is a key ingredient in standard convergence analyses for NLP methods. In fact, because standard constraint qualifications fail, it is not obvious that MPECs in their NLP form should admit Lagrange multipliers at local optima. This seems to make the application of methods that are designed to converge to points where such multipliers exist even more problematic.

Over the past years, considerable effort has gone into the search for computationally useful characterizations of stationarity for MPECs. Scheel and Scholtes [26], for example, characterize stationarity by defining at each feasible point a so-called relaxed NLP, with a locally enlarged feasible set, whose Karush-Kuhn-Tucker (KKT) conditions are necessary optimality conditions for the MPEC under LICQ for the relaxed program. It is quite surprising that, while MFCQ fails at all feasible points of an MPEC, LICQ for the relaxed NLP (sometimes called MPEC-LICQ) holds at all feasible points of a generic MPEC (see Scholtes and Stöhr [29]). The fact that the KKT conditions for the relaxed program are necessary optimality conditions for the MPEC provides a computationally convenient multiplier characterization of MPEC stationarity, called strong stationarity.

In parallel to the theoretical developments, a number of specific MPEC adaptations, as well as smoothing, regularization, or penalization schemes, for the application of standard nonlinear optimization methods were shown to converge to strongly stationary points under suitable assumptions [10, 11, 13, 27, 28]. In contrast, direct applications of off-the-shelf nonlinear optimization codes to MPECs were long neglected following early reports of poor performance; see, for example, Luo et al. [18], and more recently, Anitescu [1], who describes the poor performance of the popular MINOS [22] code on MacMPEC test problems [15].

Recently, however, interest in directly applying NLP methods for MPECs has been revitalized, primarily for two reasons. First, it is now clear that the approach makes sense because strong stationarity, a generic MPEC optimality condition, implies the existence of standard NLP multipliers for MPECs in their NLP form, albeit an unbounded set of them in view of the failure of MFCQ (see Fletcher et al. [8]). Second, in striking contrast to early experiments with augmented Lagrangian-based

NLP methods, Fletcher and Leyffer [7] report promising numerical results for sequential quadratic programming (SQP) codes. Notably filterSQP [6, 7] and SNOPT [12] perform well on the MacMPEC test set. These favorable numerical results are complemented by the local convergence analyses in [1] and [8]. Based on exact penalization arguments, Anitescu [1] gives conditions under which, SQP methods with an elastic mode, such as SNOPT, will converge locally at a fast rate when applied to MPECs. Fletcher et al. [8] show that a standard SQP algorithm, without elastic mode, will converge quadratically under sensible MPEC regularity conditions.

For interior-point methods, the situation is not rosy. The numerical results in [7] indicate that standard interior-point methods such as KNITRO [4] and LOQO [30] perform poorly on MPECs. These results are not surprising because the strictly feasible region of the MPEC in their NLP form is empty. Two different approaches have been proposed to overcome this difficulty.

The first approach, inspired by the analysis in [1], employs an exact penalty function to remove the complementarity constraint $x_1 \circ x_2 = 0$ and thereby remove the structural ill-posedness from the constraints. This approach has proved quite effective when applied to interior-point implementations such as KNITRO [17] and LOQO [2].

The second approach, adopted by Liu and Sun [16] and Raghunathan and Biegler [24], is based on the relaxation scheme analyzed by Scholtes [27]. This scheme replaces the MPEC by a sequence of relaxed subproblems whose strictly feasible region is nonempty. Liu and Sun [16] propose an interior-point method that solves each of the relaxed subproblems to within a prescribed tolerance. Raghunathan and Biegler [24], on the other hand, take only one iteration of an interior-point method on each of the relaxed subproblems. A difficulty associated with both approaches 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.

We propose an interior-point method based on a relaxation scheme (see section 3) that 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-point method to the resulting relaxed problems without having to modify the search direction, as in [24]. But like [24], our algorithm (described in section 4) performs only one interior-point iteration per relaxed problem. We show in section 5 that it converges locally at a superlinear rate, and we propose in section 6 mechanisms to ensure reasonable numerical performance also globally. We illustrate in section 7 the performance of the algorithm on a subset of the MacMPEC test problems. The numerical results confirm our local convergence analysis and demonstrate that in practice the algorithm is also quite effective globally.

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. At places we use the abbreviation $z^+ = \max\{z, 0\}$ to denote the positive part of a vector z. We denote by $[\,\cdot\,]_i$ the ith component of a vector.

Many of the optimality conditions needed for the analysis of MPECs are derived from optimality concepts of standard nonlinear optimization theory. In this section only, we consider the generic nonlinear optimization problem that results from removing the complementarity constraint from (MPEC),

$$\boxed{\begin{array}{lll}\text{(NLP)} \qquad\qquad & \underset{x}{\text{minimize}} & f(x) \\ & \text{subject to} & c(x) = 0 \\ & & x \ge 0,\end{array}}$$

and review the definitions that will be analogously defined for MPECs.

We define the *Lagrangian function* corresponding to (NLP) as

$$\mathcal{L}(x, y) = f(x) - y^T c(x), \tag{1.1}$$

where x and the m-vector y are independent variables. 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 $[c(x)]_i$.

The first and second derivatives of $\mathcal{L}(x, y)$ with respect to x are given by

$$\begin{aligned}\nabla_x \mathcal{L}(x, y) &= g(x) - A(x)^T y \\ \nabla^2_{xx} \mathcal{L}(x, y) &= \nabla^2_{xx} f(x) - \sum_{i=1}^{m} [y]_i H_i(x).\end{aligned}$$

We assume that (NLP) is feasible and has at least one point that satisfies the KKT conditions.

Definition 1.1 (First-order KKT conditions). *A triple (x^*, y^*, z^*) is a first-order KKT point for (NLP) if the following hold:*

$$\nabla_x \mathcal{L}(x^*, y^*) = z^* \tag{1.2a}$$

$$c(x^*) = 0 \tag{1.2b}$$

$$\min(x^*, z^*) = 0. \tag{1.2c}$$

Note that (1.2c) implies a complementarity relationship between x^* and z^* and that $x^*, z^* \ge 0$.

Let $\mathcal{I} = \{j \in 1, \ldots, n \mid [x^*]_j > 0\}$ be the set of indices that corresponds to inactive bounds at x^*. Define $\widehat{A}(x)$ to be the columns of $A(x)$ corresponding to the indices in $\mathcal{I}$.

Definition 1.2 (Linear independence constraint qualification). *The point x^* satisfies the* linear independence constraint qualification *for (NLP) if $\widehat{A}(x^*)$ has full-row rank.*

Definition 1.3 (Strict complementary slackness). *The point (x^*, y^*, z^*) satisfies strict complementary slackness (SCS) for (NLP) if it satisfies* (1.2) *and* $\max(x^*, z^*) > 0$.

We define a second-order sufficient condition for optimality. It depends on positive curvature of the Lagrangian,

$$p^T \nabla^2_{xx} \mathcal{L}(x^*, y^*) p > 0, \quad p \ne 0, \tag{1.3}$$

for all p in some set of feasible directions.

Definition 1.4 (Second-order sufficiency). *The point (x^*, y^*, z^*) satisfies the second-order sufficient conditions (SOSC) for (NLP) if it satisfies* (1.2), *and if* (1.3) *holds for all nonzero p such that $A(x^*)^T p = 0$ and*

$$[p]_j = 0 \textit{ if } [z^*]_j > 0 \quad \textit{and} \quad [p]_j \ge 0 \textit{ if } [z^*]_j = 0$$

for all j such that $[x]_j = 0$.

Further notation follows (including some already introduced):

x, s	primal and slack variables
y, v, z	dual variables
x^*, s^*, y^*, v^*, z^*	optimal primal, slack, and dual variables
(v_0, v_1, v_2), (z_0, z_1, z_2)	partitions of v and z corresponding to (x_0, x_1, x_2)
X, S, V, Z	diagonal matrices formed from the elements of x, s, v, z
$\delta \equiv (0, \delta_1, \delta_2, \delta_c)$	relaxation parameter vectors (see section 3.1)

2. Optimality conditions for MPECs.

2. Optimality conditions for MPECs. Standard KKT theory of nonlinear programming 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 [26]. At every feasible point of the MPEC one can define an associated problem, called the *relaxed NLP*, which is typically well behaved in nonlinear programming terms. It is shown in [26] 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 qualifications. Let $\bar{x}$ be feasible with respect to (MPEC). The *relaxed NLP at* $\bar{x}$ is defined as

$$\boxed{\begin{array}{llllll} \text{(R-NLP)} & \underset{x}{\text{minimize}} & f(x) & & & \\ & \text{subject to} & c(x) = 0, & x_0 \geq 0 & & \\ & & [x_1]_j = 0, & [x_2]_j \geq 0, & j \in \mathcal{X}_1(\bar{x}) \\ & & [x_1]_j \geq 0, & [x_2]_j = 0, & j \in \mathcal{X}_2(\bar{x}) \\ & & [x_1]_j \geq 0, & [x_2]_j \geq 0, & j \in \mathcal{B}(\bar{x}), \end{array}}$$

where we define the index subsets of $j = 1, \ldots, n$ as

$$\begin{aligned} \mathcal{X}_1(x) &= \{j \mid [x_1]_j = 0 < [x_2]_j\} \\ \mathcal{X}_2(x) &= \{j \mid [x_1]_j > 0 = [x_2]_j\} \\ \mathcal{B}(x) &= \{j \mid [x_1]_j = 0 = [x_2]_j\}. \end{aligned}$$

Only a single difference exists between (R-NLP) and (MPEC): in (R-NLP) the problematic equilibrium constraints have been substituted by a better-posed system of equality and inequality constraints. In particular, for components $j \in \mathcal{X}_1(\bar{x})$, we substitute the equilibrium constraints $[\min(x_1, x_2)]_j = 0$ by the constraints $[x_1]_j = 0$ and $[x_2]_j \geq 0$. The gradients of these constraints are linearly independent. We perform an analogous substitution for components $j \in \mathcal{X}_2(\bar{x})$. Note that these substitutions do *not* alter the feasible region around the point $\bar{x}$. For the *biactive* components $j \in \mathcal{B}(\bar{x})$, on the other hand, we substitute the equilibrium constraints by the nonnegativity bounds $[x_1]_j, [x_2]_j \geq 0$, whose gradients are again linearly independent. However, the feasible region defined by these nonnegativity bounds is larger than that defined by the equilibrium constraints. Hence, the terminology *relaxed* NLP. Note that (1.1) is also the Lagrangian function of (R-NLP)

Despite a possibly larger feasible set, one can show that if LICQ holds for (R-NLP) defined at x^*, its KKT conditions are also necessary optimality conditions for (MPEC) [26]. 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 (R-NLP) defined at x^*:*

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

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

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

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

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

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

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

With (R-NLP) we can define a constraint qualification for MPECs analogous to Definition 1.2 and deduce a necessary optimality condition for MPECs.

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

Proposition 2.3 (See, for example, Scheel and Scholtes [26]). *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.*

Note that all relaxed NLPs are contained in the globally relaxed NLP that one obtains from (MPEC) by relaxing the constraint $\min(x_1, x_2) = 0$ to $x_1, x_2 \geq 0$. If LICQ holds for all points of the globally relaxed NLP, then it also holds for all points at all relaxed NLPs. The former is the case for generic data f and c in the sense of Jongen et al. [14]. In this sense, MPEC-LICQ is satisfied at all feasible points of a generic MPEC. The condition remains generic if the equilibrium constraints have the special structure imposed by first-order conditions for variational inequalities [29].

Through the relaxed NLP, we can also define the following 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.

2.2. Strict complementary slackness conditions.

We use the relaxed NLP to define two different strict complementary slackness conditions for MPECs. The first of the two is stronger and is the one assumed in [24] and [27]. In our analysis, we assume only the second, less restrictive, condition.

Definition 2.4. *The point (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 point (x^*, y^*, z^*) satisfies the* MPEC weak strict complementary slackness *(MPEC-WSCS) for (MPEC) if it is strongly stationary and if $\max(x_0^*, z_0^*) > 0$.*

2.3. Second-order sufficiency conditions.

We define two second-order sufficient conditions for MPECs through the relaxed NLP. The first (weaker) condition is equivalent to the RNLP-SOSC defined in [25]. The second (stronger) condition is the one we assume in our analysis.

The tangent cone to the feasible set at x^* is given by

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

Definition 2.6. *The point* (x^*, y^*, z^*) *satisfies the* MPEC second-order sufficiency condition *(MPEC-SOSC) for (MPEC) if* (1.3) *holds for all nonzero* $p \in \overline{\mathcal{F}}$*, where*

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

Definition 2.7. *The point* (x^*, y^*, z^*) *satisfies the* MPEC strong second-order sufficiency condition *(MPEC-SSOSC) for (MPEC) if* (1.3) *holds for all nonzero* $p \in \underline{\mathcal{F}}$*, where*

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

Note that $\overline{\mathcal{F}} \subseteq \underline{\mathcal{F}}$. Roughly speaking, 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 Lagrange multipliers are zero.

3. A relaxation scheme for MPECs.

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

$$\begin{aligned} x_1 \circ x_2 &\leq 0\\ x_1, x_2 &\geq 0. \end{aligned} \tag{3.1}$$

A vector $\delta_c \in \mathbb{R}^n$ of strictly positive relaxation parameters relaxes (3.1) as follows:

$$\begin{aligned} x_1 \circ x_2 &\leq \delta_c\\ x_1, x_2 &\geq 0, \end{aligned} \tag{3.2}$$

so that the original complementarity constraint (3.2) is recovered when $\delta_c = 0$. Note that at all points feasible for (MPEC) the gradients of the active constraints in (3.2) are linearly independent when $\delta_c > 0$. Moreover, the strictly feasible region of the relaxed constraints (3.2) is nonempty when $\delta_c > 0$.

Scholtes [27] shows that for every strongly stationary point of the MPEC that satisfies MPEC-LICQ, MPEC-SCS, and MPEC-SSOSC, there is a locally unique piecewise smooth trajectory of local minimizers of the relaxed MPEC for each $[\delta_c]_j \in (0, \epsilon)$ with ϵ small enough. Unfortunately, the strictly feasible region of the relaxed MPEC becomes empty as the components of δ_c tend to zero.

3.1. Strictly feasible relaxations.

In contrast to (3.2), our proposed scheme relaxes each component of the bounds $x_1, x_2 \geq 0$ by the amounts $[\delta_1]_j$ and $[\delta_2]_j$ so that the relaxed complementarity constraints become

$$\begin{aligned} x_1 \circ x_2 &\leq \delta_c\\ x_1 &\geq -\delta_1\\ x_2 &\geq -\delta_2, \end{aligned} \tag{3.3}$$

where $\delta_1, \delta_2 \in \mathbb{R}^n$ 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 (3.3) is nonempty, and the active constraint gradients are linearly independent.

The main advantage of the strictly feasible relaxation scheme (3.3) is that there is no need to drive *both* relaxation parameters to zero in order 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, MPEC-WSCS, and MPEC-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 following MPEC [27]:

$$\begin{array}{ll} \underset{x}{\text{minimize}} & \frac{1}{2}\left[(x_1 - a)^2 + (x_2 - b)^2\right] \\ \text{subject to} & \min(x_1, x_2) = 0, \end{array} \tag{3.4}$$

and the associated relaxed MPEC derived by applying the relaxation (3.3) to (3.4):

$$\begin{array}{ll} \underset{x}{\text{minimize}} & \frac{1}{2}\left[(x_1 - a)^2 + (x_2 - b)^2\right] \\ \text{subject to} & x_1 \geq -\delta_1 \\ & x_2 \geq -\delta_2 \\ & x_1 \circ x_2 \leq \delta_c. \end{array} \tag{3.5}$$

For any choice of parameters $a, b > 0$, (3.4) has two local minimizers: $(a, 0)$ and $(0, b)$. Each is strongly stationary and satisfies MPEC-LICQ, MPEC-SCS, and MPEC-SOSC. Evidently, these local minimizers are also minimizers of (3.5) for $\delta_c = 0$ and for any $\delta_1, \delta_2 > 0$. If the data is changed so that $a > 0$ and $b < 0$, then the point $(a, 0)$ is a unique minimizer of (3.4), and also a unique minimizer of (3.5) 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 (3.4) and satisfies MPEC-LICQ, MPEC-SCS, and MPEC-SOSC. This point is also a unique minimizer of (3.5) 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 in order to recover a stationary point of (MPEC).

A key property of MPECs that we will exploit in this paper 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 obviously will be (a, b) or will lie on the curve $x_1 \circ x_2 = \delta_c$, the latter being the case 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 function, here $x_1 \geq 0$, is negative, that is, the gradient of the objective function points outside of the positive orthant. Note also that the multiplier of the constraint $x_1 \circ x_2 \leq \delta_c$ of the relaxed problem, which is active for sufficiently small δ_c, will converge to the value of the MPEC multiplier z_1 divided by $x_2 > 0$ as δ_c tends to zero and will thus indicate the negativity of the MPEC multiplier if δ_c is small enough. 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, for example if $b < 0, a > 0$, which is indicated by a positive MPEC multiplier or, for sufficiently small δ_1, corresponding positive multiplier of one of the bound constraints $x_2 \geq -\delta_2$. In this case we need to drive δ_2 to zero to recover the MPEC minimizer. We can also drive δ_1 to zero without causing an empty feasible set. The parameter δ_c, however, must remain positive.

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 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 MPEC-SSOSC, consider the example with $a = 0$ and $b = 1$. In this case $(0, 1)$ is a minimizer satisfying MPEC-WSCS and MPEC-SSOSC. In order to recover this minimizer from the relaxed MPEC (3.5) 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 $\delta_1, \delta_2, \delta_c > 0$.

Also, for the case $a = -1$ and $b = 0$, the point $(0, 0)$ is a minimizer satisfying the MPEC-WSCS and the MPEC-SSOSC. In this case, we need only to drive δ_1 to zero in order to recover the minimizer. In particular, for $\delta_1 = 0$ and $\delta_2, \delta_c > 0$, the point $(0, 0)$ is a minimizer satisfying LICQ, SCS, and SOSC for the corresponding relaxed MPEC.

Obviously, the arguments above apply only to the example. 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. Reformulation. 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 constraints and nonnegativity bounds on s are present. The resulting *relaxed* MPEC is

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

where the dual variables

$$y \qquad \text{and} \qquad v \equiv (v_0, v_1, v_2, v_c)$$

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

$$h(x,s) = -\begin{bmatrix} s_0 - x_0 \\ s_1 - x_1 \\ s_2 - x_2 \\ s_c + x_1 \circ x_2 \end{bmatrix} \qquad \text{and} \qquad \delta = \begin{bmatrix} 0 \\ \delta_1 \\ \delta_2 \\ \delta_c \end{bmatrix}. \tag{3.6}$$

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

$$B(x) \equiv \nabla_x h(x,s)^T = \begin{bmatrix} I & & \\ & I & \\ & & I \\ & -X_2 & -X_1 \end{bmatrix}. \tag{3.7}$$

Following Definition 1.1, a point (x^*, s^*, y^*, v^*) is a KKT point for (MPEC-δ) if it satisfies

$$\nabla_x \mathcal{L}(x,y) - B(x)^T v = 0 \tag{3.8a}$$
$$\min(s,v) = 0 \tag{3.8b}$$
$$c(x) = 0 \tag{3.8c}$$
$$h(x,s) + \delta = 0. \tag{3.8d}$$

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 δ^* be such that*

$$[\delta_i^*]_j = 0 \quad \textit{if} \quad [z_i^*]_j > 0 \tag{3.9a}$$
$$[\delta_i^*]_j > 0 \quad \textit{if} \quad [z_i^*]_j \le 0 \tag{3.9b}$$
$$[\delta_c^*]_j = 0 \quad \textit{if} \quad [z_1^*]_j < 0 \quad \textit{or} \quad [z_2^*]_j < 0 \tag{3.9c}$$
$$[\delta_c^*]_j > 0 \quad \textit{if} \quad [z_1^*]_j \ge 0 \quad \textit{and} \quad [z_2^*]_j \ge 0, \tag{3.9d}$$

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

$$\max(\delta_c^*, \delta_1^*) > 0 \quad \textit{and} \quad \max(\delta_c^*, \delta_2^*) > 0, \tag{3.10}$$

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

$$(s_0^*, s_1^*, s_2^*) = (x_0^*, x_1^* + \delta_1^*, x_2^* + \delta_2^*) \tag{3.11a}$$
$$(v_0^*, v_1^*, v_2^*) = (z_0^*, [z_1^*]^+, [z_2^*]^+) \tag{3.11b}$$
$$s_c^* = \delta_c^*, \tag{3.11c}$$

and

$$[v_c^*]_j = \begin{cases} [-z_1^*]_j^+ / [x_2^*]_j & \text{if} \quad [x_2^*]_j > 0 \quad (\text{and } [x_1^*]_j = 0) \\ [-z_2^*]_j^+ / [x_1^*]_j & \text{if} \quad [x_1^*]_j > 0 \quad (\text{and } [x_2^*]_j = 0) \\ 0 & \text{if} \quad [x_1^*]_j = [x_2^*]_j = 0, \end{cases} \tag{3.11d}$$

for $j = 1, \dots, n$, 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 corresponding condition LICQ, SCS, or SOSC, respectively, for (MPEC-δ^*).

Proof. We first need to show that (3.10) holds. For $j = 1, \ldots, n$ consider the following cases. If $[z_1^*]_j, [z_2^*]_j \leq 0$, then by (3.9b) we have that $[\delta_1^*]_j, [\delta_2^*]_j > 0$, and thus (3.10) 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 $[z_1^*]_j, [z_2^*]_j \geq 0$, which is a contradiction. Finally, if $[z_1^*]_j, [z_2^*]_j \geq 0$, then by (3.9d) we have that $[\delta_c^*]_j > 0$. Thus (3.10) 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 (3.6), (3.7), and (3.11), (x^*, y^*, s^*, v^*) satisfies (3.8a) and (3.8c)–(3.8d).

We now show that s^* and v^* satisfy (3.8b). First, note from (3.11) that $s^*, v^* \geq 0$ because $x^* \geq 0$ and $\delta_c^*, \delta_1^*, \delta_2^* \geq 0$.

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 (3.11a) and (3.11b) imply that s_0^* and v_0^* are also strictly complementary. To verify this, consider first the indices $i = 1, 2$. If $[z_i^*]_j = 0$, then $[v_i^*]_j = 0$ and $[\delta_i^*]_j > 0$. From (3.11a) it follows that $[s_i^*]_j > 0$, as required. If $[z_i^*]_j > 0$, then (3.11b) implies that $[v_i^*]_j > 0$. Moreover, by (2.1e)–(2.1f), and (3.9a), $[x_i^*]_j = [\delta_i^*]_j = 0$. 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 (3.11a) and (3.9b), $[s_i^*]_j > 0$. 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 (3.11c) and (3.9c) imply that $[z_1^*]_j < 0$ or $[z_2^*]_j < 0$ and $[v_c^*]_j > 0$ by (3.11d), as required. If $[s_c^*]_j > 0$ then (3.11c) implies that $[\delta_c^*]_j > 0$, and by (3.9d) we have that $[z_1^*]_j \geq 0$ and $[z_2^*]_j \geq 0$. Then by (3.11d) we know that $[v_c^*]_j = 0$.

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:

$$c(x) = 0 \tag{3.12a}$$
$$x_0 \geq 0 \tag{3.12b}$$
$$x_1 \geq -\delta_1^* \tag{3.12c}$$
$$x_2 \geq -\delta_2^* \tag{3.12d}$$
$$x_1 \circ x_2 \leq \delta_c^*. \tag{3.12e}$$

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

$$\begin{aligned} c(x) &= 0 \\ x &\geq 0. \end{aligned} \tag{3.13}$$

We now show that the gradients of the active constraints in (3.12) are either a subset or a nonzero linear combination of the gradients of the active constraints in (3.13), and that therefore they must be linearly independent at x^*. To do so, for $j = 1, \ldots, n$, 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 $[x_1^* \circ x_2^*]_j \leq [\delta_c^*]_j$ 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 (3.12) is a subset of the set of constraints active in (3.13).

Now consider the case $[\delta_c^*]_j = 0$. By (3.10) we have that $[\delta_1^*]_j, [\delta_2^*]_j > 0$, and because x^* is feasible for (MPEC), the jth component (3.12e) is active, but the jth components of (3.12c)–(3.12d) are inactive. In addition, note that the gradient of this constraint has all components equal to zero except $\partial[x_1^* \circ x_2^*]_j / \partial[x_1]_j = [x_2^*]_j$ and $\partial[x_1^* \circ x_2^*]_j / \partial[x_2]_j = [x_1^*]_j$. Moreover, by (3.9c) 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 $[x_1^*]_j[x_2^*]_j \le [\delta_c^*]_j$ is a nonzero linear combination of the gradient of whichever of the two constraints $[x_1^*]_j \ge 0$ and $[x_2^*]_j \ge 0$ is active.

Thus, the gradients of the constraints active in system (3.12) are either a subset or a nonzero linear combination of the constraints active in (3.13), and thus LICQ holds for (MPEC-δ^*).

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

$$\begin{array}{ll} \underset{x,s}{\text{minimize}} & f(x) \\ \text{subject to} & c(x) = 0 \\ & \quad x_0 \ge 0 \\ & \quad x_1 \ge -\delta_1^* \\ & \quad x_2 \ge -\delta_2^* \\ & x_1 \circ x_2 \le \delta_c^*, \end{array} \tag{3.14}$$

and with solution (x^*, y^*, v^*). First, we show that the set of critical directions at (x^*, y^*, v^*) for (3.14) is equal to $\underline{\mathcal{F}}$ (see Definition 2.7). Consider the critical directions for the first two constraints of (3.14). Because the constraints $c(x) = 0$ and $x_0 \ge 0$ and their multipliers are the same for (MPEC) and (3.14), their contribution to the definition of the set of critical directions is the same. In particular, we need to consider critical directions such that $A(x^*)p = 0$, $[p_0]_j \ge 0$ 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 (3.14), $x_1 \ge -\delta_1^*$, $x_2 \ge -\delta_2^*$ and $x_1 \circ x_2 \le \delta_c^*$. Because we have shown that SCS holds for (3.14) 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 (3.11) and (3.9), this is equivalent to imposing $[p_i]_j = 0$ for all j such that $[x_i^*]_j = 0$ and $[z_i^*]_j \ne 0$ for $i = 1, 2$, which is the definition of $\underline{\mathcal{F}}$.

But note that the Hessian of the Lagrangian for (3.14) is different from the Hessian of the Lagrangian for (MPEC). The reason is that in (3.14) the complementarity constraint $x_1 \circ x_2 \le \delta_c^*$ 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 $p^T \nabla_{xx}^2 \mathcal{L}(x^*, y^*)p > 0$ for all $p \in \underline{\mathcal{F}}$. To see this, note that the Hessian of $[x_1 \circ x_2]_j$ has only two nonzero elements:

$$\nabla^2_{[x_1]_j [x_2]_j} [x_1 \circ x_2]_j = \begin{bmatrix} & 1 \\ 1 & \end{bmatrix}. \tag{3.15}$$

If $[v_c^*]_j = 0$, then the Hessian of the complementarity constraint $[x_1 \circ x_2]_j \le [\delta_c^*]_j$ 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 (3.14), 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 (3.15), implies that $p^T \nabla^2_{xx}([x_1^* \circ x_2^*]_j)p = 0$ for all $p \in \underline{\mathcal{F}}$. 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-SSOSC 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* (3.10), *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*

$$z^* = B(x^*)^T v^*. \tag{3.16}$$

Proof. Equation (3.16) is derived by comparing (2.1) with (3.8). □

4. The algorithm.

Our algorithm tries to find a minimizer to the MPEC by considering a sequence of relaxed problems (MPEC-δ_k) corresponding to a sequence of relaxation parameters δ_k. The algorithm performs only a single iteration of a primal-dual interior-point method on each relaxed MPEC, and then the relaxation parameter vectors and the barrier parameter are updated simultaneously. This iteration scheme is repeated until certain convergence criteria are satisfied. We first describe the interior-point iteration taken on each relaxed subproblem and then state an update rule for the relaxation parameters and the convergence criterion used.

4.1. Computing a search direction.

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),

$$\nabla_x \mathcal{L}(x, y) - B(x)^T v \equiv r_d = 0 \tag{4.1a}$$

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

$$c(x) \equiv r_f = 0 \tag{4.1c}$$

$$h(x, s) + \delta \equiv r_b = 0, \tag{4.1d}$$

where μ is the barrier parameter, and nonnegativity of s and z are enforced at all iterations. An iteration of Newton's method based on (4.1) computes a step direction by solving the system

$$\begin{bmatrix} H(x) & & -A(x)^T & -B(x)^T \\ & V & & S \\ A(x) & & & \\ B(x) & -I & & \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta s \\ \Delta y \\ \Delta v \end{bmatrix} = - \begin{bmatrix} r_d \\ r_c \\ r_f \\ r_b \end{bmatrix}, \tag{4.2}$$

where

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

Define $w = (x, s, y, v)$ and $r = (r_d, r_c, r_f, r_b)$. The Newton system (4.2) may be written as

$$K(w)\mathit{\Delta} w = -r(w; \mu, \delta). \tag{4.3}$$

Note that the Jacobian K is independent of the barrier *and* relaxation parameters—these appear only in the right-hand side of (4.3). This is a useful property because it considerably simplifies the convergence analysis in section 5.

In order to ensure that s and v remain strictly positive (as required by interior-point methods), each computed Newton step $\mathit{\Delta} w$ 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

$$\alpha = \min(\alpha_s, \alpha_v), \tag{4.4}$$

where

$$\begin{aligned}\alpha_s &= \min\left(1, \gamma \min_{[\mathit{\Delta} s]_j<0} -[s]_j/[\mathit{\Delta} s]_j\right)\\ \alpha_v &= \min\left(1, \gamma \min_{[\mathit{\Delta} v]_j<0} -[v]_j/[\mathit{\Delta} v]_j\right).\end{aligned}$$

The $(k+1)$th iterate is computed as $w_{k+1} = w_k + \alpha \mathit{\Delta} w_k$.

4.2. Relaxation update. A sequence of relaxation parameters is constructed so that locally,

$$\lim_{k\to\infty} \delta_k = \delta^*, \tag{4.6}$$

where δ^* satisfies (3.10) and (3.9). Under certain conditions (discussed in section 5), we can recover the solution of the original MPEC from the solution of (MPEC-δ^*).

We are guided by Theorem 3.1 in developing a parameter update that will satisfy (4.6). First, we define two scalars that provide in the proximity of a minimizer lower and upper bounds on the norm of the current KKT residual $\|r(w_{k+1}; 0, \delta_k)\|$. Fix the parameter $0 < \tau < 1$. Given w_{k+1} and a vector δ^*_k (defined in section 4.4), define the lower and upper bounds, respectively, as

$$\begin{aligned}\underline{r}_{k+1} &\equiv \|r(w_{k+1}; 0, \delta^*_k)\|^{1+\tau}\\ \overline{r}_{k+1} &\equiv \|r(w_{k+1}; 0, \delta^*_k)\|^{1-\tau}.\end{aligned} \tag{4.7}$$

The lower bound is used in the relaxation parameter update to ensure the relaxation parameters δ_k converge to δ^* at a fast rate, while the upper bound is used as a threshold to determine whether the current estimate of a multiplier is close to zero.

Set $0 < \kappa < 1$. Given the current relaxation parameters δ_k, we derive δ_{k+1} using the following rule:

$$\begin{array}{llll} [\delta_{ik+1}]_j = \min(\kappa[\delta_{ik}]_j, \underline{r}_{k+1}) & \text{if} & [z_{ik+1}]_j > \overline{r}_{k+1} & \\ [\delta_{ik+1}]_j = [\delta_{ik}]_j & \text{if} & [z_{ik+1}]_j \le \overline{r}_{k+1} & \\ [\delta_{ck+1}]_j = \min(\kappa[\delta_{ck}]_j, \underline{r}_{k+1}) & \text{if} & [z_{1k+1}]_j < -\overline{r}_{k+1} \quad \text{or} & [z_{2k+1}]_j < -\overline{r}_{k+1} \\ [\delta_{ck+1}]_j = [\delta_{ck}]_j & \text{if} & [z_{1k+1}]_j \ge -\overline{r}_{k+1} \quad \text{and} & [z_{2k+1}]_j \ge -\overline{r}_{k+1}, \end{array} \tag{4.8}$$

for $i = 1, 2$ and $j = 1, \ldots, n$, where $z_{k+1} = B(x_{k+1})^T v_{k+1}$.

4.3. Barrier and steplength updates. The barrier parameter is updated as

$$\mu_{k+1} = \min(\kappa\mu_k, \underline{r}_{k+1}), \tag{4.9}$$

where $0 < \kappa < 1$, and the steplength parameter is updated as

$$\gamma_{k+1} = \max(\bar{\gamma}, 1 - \mu_{k+1}), \tag{4.10}$$

where $\bar{\gamma}$ is a fixed parameter such that $0 < \bar{\gamma} < 1$. Note that updates (4.9)–(4.10) imply that

$$\lim_{k\to\infty} \mu_k = 0 \quad \text{and} \quad \lim_{k\to\infty} \gamma_k = 1.$$

4.4. Convergence criterion. Note that because the algorithm updates the sequence of relaxation parameters δ_k so that they remain strictly positive, the current estimate x_k may not in general be feasible for (MPEC), that is, the pair x_{1k} and x_{2k} may not be nonnegative and complementary. For this reason, it may not be a good idea to use the norm of the KKT conditions for the current relaxed problem $\|r(w_{k+1}; 0, \delta_{k+1})\|$ to define our convergence criterion. Instead, we use the norm of the KKT conditions of a nearby relaxed MPEC whose relaxation parameters satisfy the requirements of Theorem 3.1, and thus more closely corresponds to the original MPEC. This nearby relaxed problem, (MPEC-δ_k^*), is defined by the relaxation parameters $\delta_k^* \equiv (0, \delta_{1k}^*, \delta_{2k}^*, \delta_{ck}^*)$, which are in turn defined recursively from the following update rule (recall from (4.7) that $\underline{r}_{k+1}$ and $\overline{r}_{k+1}$ depend on δ_k^*):

$$\begin{array}{llll} [\delta_{ik+1}^*]_j = 0 & \text{if} \quad [z_{ik+1}]_j > \overline{r}_{k+1} & & \\ [\delta_{ik+1}^*]_j = [\delta_{ik+1}]_j & \text{if} \quad [z_{ik+1}]_j \le \overline{r}_{k+1} & & \\ [\delta_{ck+1}^*]_j = 0 & \text{if} \quad [z_{1k+1}]_j < -\overline{r}_{k+1} & \text{or} & [z_{2k+1}]_j < -\overline{r}_{k+1} \\ [\delta_{ck+1}^*]_j = [\delta_{ck+1}]_j & \text{if} \quad [z_{1k+1}]_j \ge -\overline{r}_{k+1} & \text{and} & [z_{2k+1}]_j \ge -\overline{r}_{k+1}, \end{array} \tag{4.11}$$

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

The algorithm terminates when the optimality conditions for (MPEC-δ_k^*) are satisfied, that is, when

$$\|r(w_k; 0, \delta_k^*)\| < \epsilon \tag{4.12}$$

for some small and positive ϵ. We also define $w_k^* = (x_k^*, s_k^*, y_k^*, v_k^*)$ as a solution to this nearby relaxed problem, so that $\|r(w_k^*; 0, \delta_k^*)\| = 0$. Note that we never compute w_k^*—it is used only as an analytical device.

4.5. Algorithm summary. Algorithm 1 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).

5. Local convergence analysis. 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, however, 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

Algorithm 1: 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
1   [Compute the Newton step]
        Solve (4.3) for $\Delta w_k$;
2   [Truncate the Newton Step]
        Determine the maximum steplength $\alpha_k$, given by (4.4);
        $w_{k+1} \leftarrow w_k + \alpha_k \Delta w_k$;
3   [Compute MPEC multipliers]
        $z_{k+1} \leftarrow B(x_{k+1})^T v_{k+1}$;
4   [Update parameters]
        Update the relaxation parameters $\delta_{k+1}$ using (4.8);
        Compute $\delta^*_{k+1}$ using (4.11);
        Update the barrier parameter $\mu_{k+1}$ using (4.9);
        Update the steplength parameter $\gamma_{k+1}$ using (4.10);
    $k \leftarrow k + 1$;
until (4.12) holds;
$x^* \leftarrow x_k$; $y^* \leftarrow y_k$; $z^* \leftarrow z_k$;
return $x^*, y^*, z^*$;
```

continues to converge locally and superlinearly even when the relaxation parameters change at each iteration—in other words, only a single interior-point iteration is needed on each relaxed subproblem.

We make the following nondegeneracy assumptions about the MPEC minimizer (x^*, y^*, z^*); these assumptions hold implicitly throughout this section.

Assumption 5.1. *There exist strictly positive relaxation parameters δ such that the second derivatives of f and c are Lipschitz continuous over the set*

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

Assumption 5.2. *The point (x^*, y^*, z^*) satisfies MPEC–LICQ for (MPEC).*

Assumption 5.3. *The point (x^*, y^*, z^*) satisfies MPEC-WSCS and MPEC-SSOSC for (MPEC).*

The following lemma gives a lower bound on the steplength taken by the interior-point relaxation method when applied to a relaxed MPEC.

Lemma 5.4. *Let (x^*, y^*, z^*) be a strongly stationary point of (MPEC) and suppose that Assumptions* 5.1–5.3 *hold. Let δ^* be a vector of relaxation parameters that satisfies* (3.9)*, and let $w^* = (x^*, s^*, y^*, v^*)$ be the solution of (MPEC-δ^*) given by* (3.11)*. Then there exist positive constants ϵ and ϵ_1 such that if $\|w_k - w^*\| \le \epsilon$, Step* 2 *of Algorithm 1 generates a steplength α_k that satisfies*

$$|1 - \alpha_k| \le 1 - \gamma_k + \epsilon_1 \|\Delta w_k\|,$$

where Δw_k is the solution of (4.3)*.*

Proof. Because (x^*, y^*, z^*) is a strongly stationary point of (MPEC), Theorem 3.1 applies. Therefore, w^* satisfies LICQ, SCS, and SOSC for (MPEC-δ^*), and the result follows from Lemma 5 of [31]. □

5.1. Active-set identification. The relaxation parameter update rules (4.8) and (4.11) are central to the development of the relaxation algorithm. If started near enough to a solution, these rules continue to update (and reduce) the same relaxation parameters at every iteration—this property guarantees that the feasible region remains nonempty even in the limit. In some sense, it implies that the correct active set is identified.

Consider a strongly stationary point (x^*, y^*, z^*) of (MPEC). Suppose that δ_k^* is a set of relaxation parameters that satisfies (3.9) and let w_k^* be the minimizer of the associated relaxed problem (MPEC-δ_k^*) defined via (3.11). In our algorithm, we use this relaxed problem to test the optimality of the current iterate w_k (cf. section 4.4). The next lemma proves that the algorithm will continue to use the same subproblem to test the optimality of the next iterate; that is, our parameter updates are such that $\delta_{k+1}^* = \delta_k^*$. Note from (3.11) that the influence of δ_k^* on $w_k^* = (x_k^*, s_k^*, y_k^*, v_k^*)$ is relegated to only s_k^*, so that, in fact, $w_k^* \equiv (x^*, s_k^*, y^*, v^*)$. For the remainder of section 5 only, we use this property.

LEMMA 5.5. *Let (x^*, y^*, z^*) be a strongly stationary point of (MPEC) and suppose that Assumptions* 5.1–5.3 *hold. Moreover, assume that δ_k^* satisfies* (3.9)*, 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* (3.11)*, and assume that $\|w_{k+1} - w_k^*\| \le \epsilon_k \|w_k - w_k^*\|$, where $\epsilon_k < 1$. Then there exists positive constants ϵ and β such that if*

$$\|w_k - w_k^*\| < \epsilon \tag{5.1}$$

$$\|\delta_k - \delta_k^*\| < \beta \|w_k - w_k^*\|^{1+\tau}, \tag{5.2}$$

then the update rules (4.8) *and* (4.11) *imply that $\delta_{k+1}^* = \delta_k^*$.*

Proof. We first show that $\overline{r}_{k+1}$ 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^*, $r(w_k^*; 0, \delta_k^*) = 0$. Moreover, Assumption 5.1 implies that the KKT residual $r(w; \mu, \delta)$ is differentiable. In addition, as a consequence of Theorem 3.1, the Jacobian of the KKT residual $r(w; \mu, \delta)$ with respect to w, $K(w)$, is uniformly bounded and bounded away from zero in the vicinity of w_k^*. Then (5.2) and the hypotheses of this lemma imply that there exist positive constants β_1 and β_2 such that

$$\beta_1 \|w_{k+1} - w_k^*\| < \|r(w_{k+1}; 0, \delta_k^*)\| < \beta_2 \|w_{k+1} - w_k^*\|.$$

Then, by the definition of $\overline{r}_{k+1}$ we have that

$$\beta_1 \|w_{k+1} - w_k^*\|^{1-\tau} < \overline{r}_{k+1} < \beta_2 \|w_{k+1} - w_k^*\|^{1-\tau}. \tag{5.3}$$

Let $\epsilon_4 \equiv \frac{1}{2} \min(|[z^*]_j| \mid \text{for all } j \text{ such that } [z^*]_j \neq 0)$. Conditions (5.1) and (5.3) imply that for ϵ small enough

$$\overline{r}_{k+1} < \epsilon_4 \quad \text{and} \quad \|w_{k+1} - w_k^*\| < \epsilon_4. \tag{5.4}$$

Moreover, because $z_{k+1} = B(x_{k+1})^T v_{k+1}$, $z^* = B(x^*)^T v^*$, and $\|B(x)\|$ is uniformly bounded in a neighborhood of x^*, (5.4) implies that for ϵ small enough

$$\|z_{k+1} - z^*\| < \epsilon_4. \tag{5.5}$$

Consider the indices $i = 1, 2$ and $j = 1, \ldots, n$. Suppose that $[z^*]_j > 0$. Then (5.4) and (5.5) imply that

$$[z_{ik+1}]_j = [z^*]_j + ([z_{ik+1}]_j - [z^*]_j) > [z^*]_j - \epsilon_4 \geq \epsilon_4 > \overline{r}_{k+1}. \tag{5.6}$$

Suppose instead that $[z^*]_j < 0$. Then (5.4) and (5.5) imply that

$$-\overline{r}_{k+1} > -\epsilon_4 > [z^*]_j + \epsilon_4 = [z_{ik+1}]_j - ([z_{ik+1}]_j - [z^*]_j) + \epsilon_4 > [z_{ik+1}]_j. \tag{5.7}$$

Finally, suppose that $[z^*]_j = 0$. Then because $\tau > 0$, there exists an ϵ small enough so that

$$|[z_{ik+1}]_j| = |[z_{ik+1}]_j - [z^*]_j| < \beta_1 \|w_{k+1} - w^*_k\|^{1-\tau} < \overline{r}_{k+1}. \tag{5.8}$$

Because $\delta_k > 0$, (4.8) implies that $\delta_{k+1} > 0$, and from (4.11) and (5.6)–(5.8) we have that δ^*_{k+1} satisfies (3.9). 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, (4.8) implies that $[\delta_{k+1}]_j = [\delta_k]_j$. Then because $[\delta^*_k]_j = [\delta_k]_j$ for such j, the update rules (4.8) and (4.11) 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, the minimizer to the relaxed problem for the $(k+1)$th iterate is the same as for the kth iterate.

5.2. Superlinear convergence. In this section we prove the superlinear convergence property of the algorithm. We use the shorthand notation $r_k \equiv r(w_k; 0, \delta_k)$ and $r^*_k \equiv r(w_k; 0, \delta^*_k)$. Define the vector

$$\eta^*_k = \begin{bmatrix} 0 \\ \mu_k e \\ 0 \\ \delta_k - \delta^*_k \end{bmatrix},$$

and note that $r_k = r^*_k - \eta^*_k$.

Theorem 5.6. *Let (x^*, y^*, z^*) be a strongly stationary point of (MPEC) and suppose that Assumptions* 5.1–5.3 *hold. Assume that δ^*_k satisfies* (3.9)*, and let $w^*_k = (x^*, s^*_k, y^*, v^*)$ be the solution of the corresponding relaxation (MPEC-δ^*_k) given by* (3.11)*. Then there exists positive constants ϵ and β such that if Algorithm 1 is started with iterates at $k = 0$ that satisfy*

$$\|w_k - w^*_k\| < \epsilon \tag{5.9}$$

$$\|\delta_k - \delta^*_k\| < \beta \|w_k - w^*_k\|^{1+\tau} \tag{5.10}$$

$$\mu_k < \beta \|w_k - w^*_k\|^{1+\tau} \tag{5.11}$$

$$1 - \gamma_k < \beta \|w_k - w^*_k\|^{1+\tau} \tag{5.12}$$

and

$$[\delta^*_k]_j = [\delta_k]_j > 0 \quad \text{for all j such that} \quad [\delta^*_k]_j \neq 0, \tag{5.13}$$

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 $\|w_{k+1} - w^*_k\| < \sigma \|w_k - w^*_k\|^{1+\tau}$. Second, we show that $\delta^*_{k+1} = \delta^*_k$, and thus 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 and thus the result follows by induction.

We show now that $\|w_{k+1} - w_k^*\| < \sigma \|w_k - w_k^*\|^{1+\tau}$. From Assumptions 5.1–5.3 it follows from Theorem 3.1 that $K(w_k)$ is nonsingular for all $\epsilon > 0$ small enough, so that $K(w_k)^{-1}$ is uniformly bounded. Consider only such ϵ. Then

$$\begin{aligned} w_{k+1} - w_k^* &= w_k - w_k^* - \alpha_k K(w_k)^{-1} r_k \\ &= (1-\alpha_k)(w_k - w_k^*) + \alpha_k K(w_k)^{-1}(K(w_k)(w_k - w_k^*) - r_k^* + \eta_k^*) \\ &= (1-\alpha_k)(w_k - w_k^*) \\ &\quad + \alpha_k K(w_k)^{-1}\eta_k^* + \alpha_k K(w_k)^{-1}(K(w_k)(w_k - w_k^*) - r_k^*). \end{aligned} \tag{5.14}$$

Each term on the right-hand side of (5.14) can be bounded as follows. By Lemma 5.4 and the Cauchy inequality, there exists a positive constant ϵ_1 such that

$$\|(1-\alpha_k)(w_k - w_k^*)\| \le \big((1-\gamma_k) + \epsilon_1 \|\Delta w_k\|)\big)\|w_k - w_k^*\|. \tag{5.15}$$

We now further bound the right-hand side of (5.15). Because $\|K(w_k)^{-1}\|$ is uniformly bounded for ϵ small enough, there exists a positive constant ϵ_2 such that

$$\|\Delta w_k\| = \|K(w_k)^{-1}(-r_k^* + \eta_k^*)\| \le \epsilon_2(\|r_k^*\| + \|\eta_k^*\|). \tag{5.16}$$

The differentiability of r implies that there exists a positive constant ϵ_3 such that

$$\|r_k^*\| = \|r(w_k; 0, \delta_k^*) - r(w_k^*; 0, \delta_k^*)\| \le \epsilon_3 \|w_k - w_k^*\|. \tag{5.17}$$

Moreover, (5.10) and (5.11) imply that there exists a positive constant ϵ_4 such that

$$\|\eta_k^*\| \le \epsilon_4 \|w_k - w_k^*\|^{1+\tau}. \tag{5.18}$$

Then substituting (5.16), (5.17), (5.18), and condition (5.12), into (5.15) we have

$$\|(1-\alpha_k)(w_k - w_k^*)\| \le (\beta + \epsilon_1\epsilon_2\epsilon_4)\|w_k - w_k^*\|^{2+\tau} + \epsilon_1\epsilon_2\epsilon_3\|w_k - w_k^*\|^2. \tag{5.19}$$

From the uniform boundedness of $\|K(w_k)^{-1}\|$ and (5.18), the second term in (5.14) satisfies

$$\|\alpha_k \; K(w_k)^{-1}\eta_k^*\| \le \alpha_k \|K(w_k)^{-1}\| \; \|\eta_k^*\| \le \epsilon_5 \|w_k - w_k^*\|^{1+\tau}, \tag{5.20}$$

for some positive constant ϵ_5. Finally, the third term in (5.14) satisfies (using Taylor's theorem and again the fact that $\|K(w_k)^{-1}\|$ is uniformly bounded)

$$\|\alpha_k \; K(w_k)^{-1}(K(w_k)(w_k - w_k^*) - r_k^*)\| \le \epsilon_6 \|w_k - w_k^*\|^2, \tag{5.21}$$

for some positive constant ϵ_6. Hence, (5.14) and (5.19)–(5.21) yield

$$\|w_{k+1} - w_k^*\| \le \sigma \|w_k - w_k^*\|^{1+\tau}, \tag{5.22}$$

for some positive constant σ, as required.

The second part of the proof is to show that $\delta_{k+1}^* = \delta_k^*$. To see this, note that by (5.22) we know that for ϵ small enough the assumptions of Lemma (5.5) hold and therefore $\delta_{k+1}^* = \delta_k^*$. As a result, w_k^* is also a minimizer of (MPEC-δ_{k+1}^*).

In the third and final part of the proof, we show that the conditions of the theorem also hold for the $(k+1)$th iterate. Moreover, $w_{k+1}^* = w_k^*$ 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 (5.22).

To show this, first note that because $\delta_{k+1}^* = \delta_k^*$, δ_{k+1}^* satisfies (3.9). Moreover, (5.22) implies for ϵ small enough that (5.9) holds for w_{k+1}. Moreover, Assumption 5.1 implies that the KKT residual $r(w;\mu,\delta)$ is differentiable. In addition, as a consequence of Theorem 3.1, the Jacobian of the KKT residual $r(w;\mu,\delta)$ with respect to w, $K(w)$, is uniformly bounded in the vicinity of w_k^*. This, together with the definition of $\overline{r}_{k+1}$, the fact that $\delta_{k+1}^* = \delta_k^*$, and the parameter update rules (4.8), (4.11), (4.9), and (4.10) imply that (5.10)–(5.12) hold for δ_{k+1}, μ_{k+1}, and γ_{k+1}. The result follows by induction from (5.22). □

6. Global convergence discussion.

This section outlines one possible globalization scheme for the proposed method. Rather than providing a detailed global convergence analysis, which is beyond the scope of this paper, our aim is simply to discuss certain mechanisms to ensure a reasonable performance of the algorithm in practice. Moreover, we justify how that the proposed globalization mechanisms do not affect the local convergence properties analyzed in section 5.

We describe in section 6.1 a linesearch procedure that can be used to globalize the algorithm. In section 6.2, we propose a safeguard to the relaxation parameter update that prevents the algorithm from converging to spurious stationary points.

6.1. A linesearch approach.

We propose a backtracking linesearch on the following augmented Lagrangian merit function:

$$\mathcal{L}_A(w;\delta,\rho,\mu) = \mathcal{L}(w;\delta) + \tfrac{1}{2}\rho(\|c(x)\|^2 + \|h(x,s)+\delta\|^2) - \mu\sum_j \log s_j,$$

where ρ and μ are the penalty and barrier parameters and

$$\mathcal{L}(w;\delta) = f(x) - c(x)^T y - (h(x,s)+\delta)^T v$$

is the Lagrangian function corresponding to (MPEC-δ). The theoretical properties of this merit function within an interior-point framework have been analyzed by Moguerza and Prieto [20].

Given a search direction Δw_k and a maximum steplength $\bar{\alpha}_k$, the backtracking linesearch reduces the steplength by a factor $0 < \bar{\beta} < 1$ until the Armijo condition holds, that is, until $\phi_k(\alpha) \le \phi_k(0) + \varphi\alpha\phi_k'(0)$, where $0 < \varphi < 1$ and $\phi_k(\alpha) = \mathcal{L}_A(w_k + \alpha\Delta w_k; \delta_k, \rho_k, \mu_k)$.

The backtracking linesearch is well defined provided that the search direction provides sufficient descent for the augmented Lagrangian merit function. To ensure this, we propose a modification of (4.2) and an update rule for the penalty parameter ρ. These are dicussed below.

Consider a symmetrized version of (4.2):

$$\begin{bmatrix} H(x) & & A(x)^T & B(x)^T \\ & VS^{-1} & & -I \\ A(x) & & & \\ B(x) & -I & & \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta s \\ -\Delta y \\ -\Delta v \end{bmatrix} = - \begin{bmatrix} r_d \\ S^{-1} r_c \\ r_f \\ r_b \end{bmatrix}. \tag{6.1}$$

For the remainder of this section we drop the arguments of the functions in (6.1). We propose modifying this system by introducing a diagonal perturbation of H and solving the following modified system:

$$\bar{K}\Delta\bar{w} = -\bar{r},$$

where

$$\bar{K} = \begin{bmatrix} \bar{H} & \bar{A}^T \\ \bar{A} & \end{bmatrix}, \quad \bar{H} = \begin{bmatrix} H + \lambda I & \\ & VS^{-1} \end{bmatrix}, \quad \text{and} \quad \bar{A} = \begin{bmatrix} A & \\ B & -I \end{bmatrix}.$$

The parameter $\lambda \geq 0$ is chosen to ensure that $\bar{H}$ is positive definite in the null space of $\bar{A}$. This is the modification proposed in [30]. The following proposition shows that this modification of the KKT matrix together with a suitable update rule for the penalty parameter ρ ensures that the search direction is a descent direction for ϕ.

PROPOSITION 6.1. *If the penalty parameter is chosen as*

$$\rho > \frac{\max(\xi, 0)}{\|c(x)\|^2 + \|h(x,s) + \delta\|^2},$$

where

$$\xi \equiv -\Delta x^T H \Delta x - \Delta s^T V S^{-1} \Delta s - 2c(x)^T \Delta y + 2(h(x,s) + \delta)\Delta v,$$

then $\phi'(0) < 0$.

Proof. The gradient of the merit function is given by

$$\nabla_w \mathcal{L}_A(w; \delta, \rho, \mu) = \begin{bmatrix} \nabla_x \mathcal{L}_A \\ \nabla_s \mathcal{L}_A \\ \nabla_y \mathcal{L}_A \\ \nabla_v \mathcal{L}_A \end{bmatrix} = \begin{bmatrix} \nabla_x \mathcal{L}(w; \delta) + \rho A(x)^T c(x) + \rho B(x)^T (h(x,s) + \delta) \\ -\mu S^{-1} e + v + \rho(h(x,s) + \delta) \\ -c(x) \\ -(h(x,s) + \delta) \end{bmatrix}.$$

As a result,

$$\phi'(0) = \Delta w^T \nabla_w \mathcal{L}_A(w; \delta, \rho, \mu) = \xi - \rho\|c(x)\|^2 - \rho\|h(x,s) + \delta\|^2.$$

Note that the directional derivative can be made sufficiently negative by increasing the value of ρ whenever $\|c(x)\| + \|h(x,s) + \delta\| \neq 0$. Otherwise, it is enough that $-\Delta x^T H \Delta x - \Delta s^T V S^{-1} \Delta s$ is sufficiently negative. But this is ensured by the proposed modification of the KKT matrix. □

Note that another positive effect of the proposed KKT matrix modification is that it uses second-order information, which may help to avoid convergence to nonoptimal, first-order stationary points. Moreover, the following proposition shows that the linesearch has no effect on the local convergence analysis of the previous section.

PROPOSITION 6.2. *Let (x^*, y^*, z^*) be a strongly stationary point of (MPEC), and suppose that Assumptions* 5.1–5.3 *hold. Moreover, assume that δ_k^* satisfies* (3.9), *and let $w_k^* = (x^*, s_k^*, y^*, v^*)$ be the solution of the corresponding relaxation (MPEC-δ_k^*) given by* (3.11). *Then there exists positive constants ϵ and β such that if* (5.9)–(5.13) *hold, the Armijo condition*

$$\phi_k(\alpha) \leq \phi_k(0) + \varphi \alpha \phi_k'(0)$$

is satisfied for $\alpha = \bar{\alpha}_k$.

Proof. The result follows from [21, Lemma 6], noting that $\bar{\alpha}_k \geq 1 - \mu_k$ by extending the results in [31, 9]. □

6.2. Safeguard to the relaxation parameter update. In this section, we propose 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 (3.4). The relaxed MPEC (with slacks) is given by

$$\begin{array}{ll} \underset{x_1,x_2,s_1,s_2,s_c\in\mathbb{R}}{\text{minimize}} & \frac{1}{2}(x_1-a_1)^2+\frac{1}{2}(x_2-a_2)^2 \\ \text{subject to} & s_1-x_1=\delta_1 \\ & s_2-x_2=\delta_2 \\ & s_c+x_1x_2=\delta_c \\ & s\geq 0. \end{array} \tag{6.2}$$

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 (6.2), but it is not feasible for (3.4). 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

$$(s_0, s_1, s_2, s_c) = (x_0, x_1 + \delta_1, x_2 + \delta_2, \delta_c) \tag{6.3}$$

(cf. (3.11a)). To ensure that (6.3) always holds at the limit point, we propose the following modification of (4.8):

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

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. Moreover, one can easily show that the local convergence results of the previous section still hold when using this relaxation parameter update. The intuition for this is that, in an ϵ-neighborhood of an MPEC minimizer, condition (6.3) holds to order ϵ.

7. Numerical results. We illustrate in this section the numerical performance of the interior-point relaxation algorithm on the MacMPEC test problem set [15]. The results confirm our local convergence analysis and show that the globalization mechanisms proposed in section 6 perform well in practice.

The interior-point relaxation algorithm has been implemented as a Matlab program. Problems from the MacMPEC test suite (coded in AMPL [15]) 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-sized problems from the MacMPEC test suite.

Table 7.2 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 r at the solution, and the last two columns indicate the exit status of the algorithm. The exit flags are described in Table 7.1. The quantities $(\delta_c^*, \delta_1^*, \delta_2^*)$ describe the final values of the relaxation parameters.

Table 7.1
Exit flags in Table 7.2. The second exit flag indicates when the final relaxation parameters $(\delta_c^*, \delta_1^*, \delta_2^*)$ *satisfy the complementarity condition given by* (3.10).

flag1	Status
0	Terminated by iteration limit (150)
1	Stationary point found
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	$\max(\delta_c^*, \delta_i^*) > 0$

The current iterate is considered optimal if $\|r(w_k; 0, \delta_k^*)\| < 10^{-6}(1 + \|g(x_k)\|)$ (cf. (4.12)). The initial penalty parameter is $\rho_0 = 10^2$. The linesearch parameters are $\varphi = 10^{-3}$ and $\bar{\beta} = 0.5$. The parameters for the barrier and relaxation updates are $\tau = 0.3$ and $\kappa = 0.9$.

Table 7.2
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
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

Problem	iter	nfe	f	$\|v_c^*\|$	$\|r\|$	flag1	flag2
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

Problem	iter	nfe	f	$\|v_c^*\|$	$\|r\|$	flag1	flag2
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
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

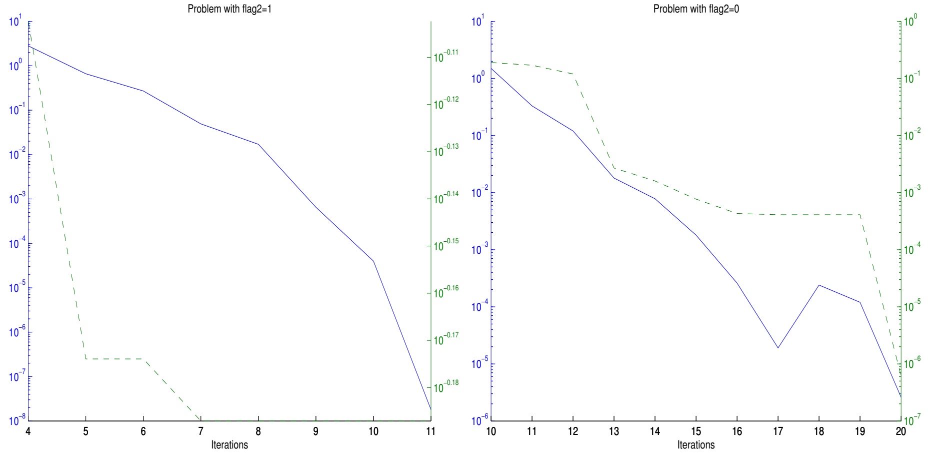
The results seem to confirm that the global convergence safeguards proposed in section 6 are effective in practice. In particular, the algorithm solves most of the test problems in the collection. Moreover, some of the problems on which our algorithm fails are ill posed according to [24, 3, 2]. For instance, *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.

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 MPEC-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 the MPEC-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 $\max\left(\delta_c^*, \min(\delta_1^*, \delta_2^*)\right) > 0$ (flag2=1), the condition number of the KKT matrix remains bounded away from zero and the algorithm converges superlinearly. On the other hand, when $\max\left(\delta_c^*, \min(\delta_1^*, \delta_2^*)\right) = 0$ (flag2=0), the condition number of the KKT matrix grows large, and the algorithm converges only linearly.

This behavior can be observed in Figure 7.1, 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 8 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 8 problems (approximately 10% of the total) for which the MPEC-SCS does not hold at the minimizer (although MPEC-WSCS and MPEC-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 the MPEC-SSOSC do not hold.

8. Conclusions. We propose an interior-point method based on a relaxation scheme that preserves a nonempty strictly feasible region even in the limit. As a consequence, the algorithm has better convergence properties than previous interior-



(a) Problem *ex9.2.4*

(b) Problem *ex9.2.7*

FIG. 7.1. *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.*

point methods for MPECs without the need to use specialized techniques for degenerate problems. Moreover, in our analysis, we assume a less restrictive strict complementarity condition than the standard one. The numerical results confirm the local convergence analysis and demonstrate the practical effectiveness of the mechanisms proposed to induce global convergence.

Acknowledgments. We are grateful to Sven Leyffer for putting together the excellent MacMPEC test problem set, and to Jorge Nocedal and Danny Ralph for many helpful comments. We also acknowledge comments from seminar participants at the 18th International Symposium on Mathematical Programming in Copenhagen.

REFERENCES

[1] M. Anitescu, *On solving mathematical programs with complementarity constraints as nonlinear programs*, Tech. Report ANL/MCS-P864-1200, Argonne National Laboratory, 2000.

[2] 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, 2003.

[3] 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, 2002.

[4] R. Byrd, M. E. Hribar, and J. Nocedal, *An interior point method for large scale nonlinear programming*, SIAM Journal on Optimization, 9 (1999), pp. 877–900.

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

[6] R. Fletcher and S. Leyffer, *User manual for filterSQP*, Tech. Report NA-181, University of Dundee, April 1998.

[7] ———, *Numerical experience with solving MPECs as NLPs*, Tech. Report NA-210, University of Dundee, August 2002.

[8] 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, May 2002.

[9] A. Forsgren, P. Gill, and M. Wright, *Interior methods for nonlinear optimization*, SIAM Review, 44 (2002), pp. 525–597.

[10] M. Fukushima and J. Pang, *Convergence of smoothing continuation method for mathematical programs with complementarity constraints*, in Ill-Posed Variational Problems and Regularization Techniques, M. Thera and R. Tichatschke, eds., Berlin, 1999, Springer-Verlag, pp. 99–110.

[11] M. Fukushima and P. Tseng, *An implementable active-set algorithm for computing a B-stationary point of the mathematical program with linear complementarity constraints*, SIAM Journal on Optimization, 12 (2002), pp. 724–739.

[12] P. E. Gill, W. Murray, and M. A. Saunders, *SNOPT: An SQP algorithm for large-scale constrained optimization*, SIAM Journal on Optimization, 12 (2002), pp. 979–1006.

[13] X. Hu and D. Ralph, *Convergence of a penalty method for mathematical programs with complementarity constraints*, Journal of Optimization Theory and Applications, forthcoming (2004).

[14] H. Jongen, P. Jonker, and F. Twilt, *Nonlinear Optimization in $\mathcal{R}^n$ II: Transversality, Flows, Parametric Aspects*, Peter Lang Verlag, Frankfurt, Germany, 1986.

[15] S. Leyffer, *MacMPEC: AMPL collection of MPECs*, April 2004. http://www.mcs.anl.gov/~leyffer/MacMPEC.

[16] 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, 2001.

[17] G. Lopez-Calva, S. Leyffer, and J. Nocedal, *Interior point methods for MPECs.* Talk at 18th International Symposium on Mathematical Programming, 2003.

[18] Z. Luo, J. Pang, and D. Ralph, *Mathematical Programs with Equilibrium Constraints*, Cambridge University Press, 1996.

[19] O. L. Mangasarian and S. Fromovitz, *The Fritz-John conditions in the presence of equality and inequality constraints*, J. Math. Anal. Appl., 17 (1967), pp. 37–47.

[20] J. Moguerza and F. Prieto, *An augmented lagrangian interior-point method using directions of negative curvature*, Mathematical Programming A, 95 (2003), pp. 573–616.

[21] ———, *Combining search directions using gradient flows*, Mathematical Programming A, 96 (2003), pp. 529–559.

[22] 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.

[23] 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.

[24] A. U. Raghunathan and L. T. Biegler, *Interior point methods for mathematical programs with complementarity constraints*, tech. report, Carnegie Mellon University, Department of Chemical Engineering, 2003.

[25] D. Ralph and S. J. Wright, *Some properties of regularization schemes for MPECs*, Tech. Report 03-04, Computer Sciences, University of Wisconsin, December 2003.

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

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

[28] S. Scholtes and M. Stöhr, *Exact penalization of mathematical programs with equilibrium constraints*, SIAM Journal on Control and Optimization, 37 (1999), pp. 617–652.

[29] S. Scholtes and M. Stöhr, *How stringent is the linear independence condition for mathematical programs with equilibrium constraints?*, Mathematics of Operations Research, 26 (2001), pp. 851–863.

[30] D. Shanno and R. Vanderbei, *An interior-point algorithm for nonconvex nonlinear programming*, Computational Optimization and Applications, 13 (1999), pp. 231–252.

[31] H. Yamashita and H. Yabe, *Superlinear and quadratic convergence of some primal-dual interior point methods for constrained optimization*, Mathematical Programming, 75 (1996), pp. 377–397.