CONVERGENCE ANALYSIS OF AN INTERIOR-POINT METHOD FOR NONCONVEX NONLINEAR PROGRAMMING

HANDE Y. BENSON, ARUN SEN, AND DAVID F. SHANNO

ABSTRACT. In this paper, we present global and local convergence results for an interior-point method for nonlinear programming. The algorithm uses an ℓ_1 penalty approach to relax all constraints, to provide regularization, and to bound the Lagrange multipliers. The penalty problems are solved using a simplified version of Chen and Goldfarb's strictly feasible interior-point method [6]. The global convergence of the algorithm is proved under mild assumptions, and local analysis shows that it converges Q-quadratically. The proposed approach improves on existing results in several ways: (1) the convergence analysis does not assume boundedness of dual iterates, (2) local convergence does not require the Linear Independence Constraint Qualification, (3) the solution of the penalty problem is shown to locally converge to optima that may not satisfy the Karush-Kuhn-Tucker conditions, and (4) the algorithm is applicable to mathematical programs with equilibrium constraints.

1. INTRODUCTION

There has recently been renewed interest in penalty methods. These methods, as described in [8], have traditionally been an alternative to barrier methods, but the focus of their recent reemergence is primarily due to the fact that they provide bounds for problems with unbounded sets of optimal Lagrange multipliers. One such set of problems is Mathematical Programs with Equilibrium Constraints (MPECs). The theoretical desirability of penalty methods for MPECs is studied in, for instance, [1], while [2] shows that they work quite well in practice when implemented in LOQO, an interior-point code. In subsequent papers [3] and [4], various other theoretical advantages of penalty methods within the interior-point framework are provided, including warmstarting capabilities, regularization, and infeasibility identification for both linear and nonlinear programming problems. Of course, penalty methods have long been used extensively in sequential quadratic programming codes, most notably in SNOPT as the *elastic mode* to aid in infeasibility detection.

Our main goal in this paper is to use a penalty method approach in order to propose an interior-point algorithm whose convergence results use rather mild assumptions on the underlying problem. Traditional proofs of convergence for interior-point methods, both with and without penalty approaches, generally make assumptions on the boundedness of primal and dual iterates and the existence of strictly feasible

---

Date: June 11, 2007.

Key words and phrases. interior-point methods, nonlinear programming, mathematical programs with equilibrium constraints.

Research of the first author is sponsored by ONR grant N00014-04-1-0145. Research of the second author is supported by NSF grant DMS-9870317 and ONR grant N00014-98-1-0036. Research of the third author is supported by NSF grant DMS-0107450.

primal and dual solutions, among others. If used, the penalty approach is generally employed to handle the equality constraints. Two recent works that propose and prove the convergence of interior-point methods with a penalty approach are [9] and [6]. In [9], Liu and Sun show that their interior-point algorithm converges globally under the standard assumptions of bounded primal and dual iterates. However, they also propose using a steepest descent approach whenever the Newton direction fails to be a descent direction. Doing so guarantees the convergence theoretically, but would greatly increase the iteration count within an implementation. In [6], Goldfarb and Chen analyze a Newton-method based interior-point method which is similar to the underlying algorithms in LOQO and IPOPT, two highly efficient interior-point codes. Their approach incorporates an ℓ_2 penalty function for use with nonlinear programming problems (NLPs) that have inequality and equality constraints. The authors prove that the algorithm either (1) finds a first-order point, (2) approaches a minimizer of infeasibility, or (3) finds an optimal solution that does not satisfy the Karush-Kuhn-Tucker (KKT) conditions by letting the penalty parameter tend to infinity. They also assume that the primal and dual iterates remain bounded.

The proposed approach improves on existing results in 4 ways: (1) the convergence analysis does not assume boundedness of dual iterates, (2) local convergence does not require the Linear Independence Constraint Qualification, (3) the solution of the penalty problem is shown to locally converge to optima that may not satisfy the Karush-Kuhn-Tucker conditions, and (4) the algorithm is applicable to mathematical programs with equilibrium constraints. In the next section, we propose an interior-point method that uses an ℓ_1 penalty approach. This approach will relax all the constraints, which makes it trivial to identify a feasible solution. Therefore, an assumption of a nonempty feasible region will not be necessary. It also ensures that the dual iterates remain bounded, removing the need for the assumption of the boundedness of the Hessian of the Lagrangian in [6]. In Section 3, we will start by showing various characteristics of the algorithm that will be useful for developing a convergence proof. With these results in hand, we will provide a global convergence proof using results from [6], showing that our algorithm will either converge to a first-order point, a non-KKT optimum, or an infeasible point that minimizes an ℓ_1 measure of infeasibility. We will discuss further properties of the penalty problem for use in the local convergence proof in Section 4, including a result that shows that the KKT solutions of the penalty problem locally converge to optima for the original problem, even if the said optima do not satisfy the KKT conditions. Finally, in Section 5, we will show that locally, our algorithm has a Q-quadratic rate of convergence under the Mangasarian-Fromowitz Constraint Qualification (MFCQ), instead of the traditional Linear Independence Constraint Qualification (LICQ). The proposed reformulation and algorithm will be applicable to Mathematical Programs with Equilibrium Constraints (MPECs), as well. In short, this approach will strengthen the assumptions mentioned in literature and share characteristics with efficiently-implemented interior-point methods.

2. A Penalty Interior-Point Method for Nonconvex NLP

An NLP in standard form can be written as

$$
\begin{array}{llll}
\min\limits_{y,z} & f(y,z) & & \\
\text{s.t.} & g(y,z) & \geq & 0 \\
 & h(y,z) & = & 0 \\
 & y & \geq & 0,
\end{array}
\tag{1}
$$

where $y \in \mathbb{R}^p$, $z \in \mathbb{R}^l$, $f : \mathbb{R}^{n+l} \to \mathbb{R}$, $g : \mathbb{R}^{n+l} \to \mathbb{R}^q$, and $h : \mathbb{R}^{p+l} \to \mathbb{R}^k$. To simplify our notation, we will convert our problem into

$$
\begin{array}{llll}
\min\limits_{x} & f(x) & & \\
\text{s.t.} & r(x) & \geq & 0,
\end{array}
\tag{2}
$$

where

$$
x = \begin{bmatrix} y \\ z \end{bmatrix}, \text{ and } r(x) = \begin{bmatrix} g(y,z) \\ h(y,z) \\ -h(y,z) \\ y \end{bmatrix}
$$

We will let $n = p + l$ and $m = q + 2k + p$, so we have that $x \in \mathbb{R}^n$ and $r : \mathbb{R}^n \to \mathbb{R}^m$.

In order to establish convergence results for an interior-point method solving (2), we need to focus on two important features of this problem. The first is the presence of the equality constraints. We chose to split the equality constraint into two inequalities, which are incorporated separately into the penalty approach. Theoretically, it is essential to do this, because relaxing an equality constraint means introducing both lower and upper bounds. As later will be discussed, doing so does not violate linear independence of the constraint gradients since a separate slack variable is added to each constraint. Further, as demonstrated in [11], this does not increase the problem size in practice as one of the two inequalities can be expressed solely in terms of slack variables and reduced. The second important feature of (1) is that we make no assumptions on the boundedness of the optimal set of dual solutions. It is a well-known fact (see, for instance [1]) that the set of optimal Lagrange multipliers for a large class of problems, mathematical programs with equilibrium constraints, is always unbounded. Equivalently, it can be stated that the Mangasarian-Fromovitz Constraint Qualification (MFCQ) cannot hold at a solution, or that the solution is not regular. However, assumptions of regularity, the satisfaction of MFCQ, or the boundedness of the optimal set of Lagrange multipliers must appear in standard proofs of convergence for primal-dual interior-point methods, because such methods approach the analytic center of the face of optimal primal and dual solutions, and unboundedness implies that the algorithm will fail to converge.

In [2], we proposed the use of a penalty method in order to resolve the issue of having unbounded Lagrange multipliers within the context of an interior-point method. In fact, other papers such as [1], also discuss the use of a penalty approach for bounding the Lagrange multipliers. In [2], however, we applied an ℓ_∞ penalty method by relaxing only the complementarity conditions. Here, we will consider an ℓ_1 penalty method that relaxes all of the constraints in the problem.

We first reformulate the problem using an ℓ_1 penalty approach on the constraints. Thus, problem (2) is first converted to

$$
\begin{array}{llllll}
\min\limits_{x,\xi} & f(x) & + & d^T\xi & & \\
\text{s.t.} & r(x) & + & \xi & \geq & 0 \\
 & & & \xi & \geq & 0,
\end{array} \tag{3}
$$

where $\xi \in \mathbb{R}^m$ are the relaxation variables and $d \in \mathbb{R}^m$ are the corresponding penalty parameters. Note that the penalty approach used to form (3) corresponds to a smooth version of the ℓ_1 penalty method due to the introduction of the relaxation variables. Also, the use of vectors of penalty parameters serves to accommodate scale differences in the problem and allows for a more stable implementation.

Now, we can apply either of the interior-point methods presented in [6] to solve (3). The first is a quasi-feasible approach, where the constraints are always satisfied, and the second is an infeasible interior-point method, where all constraints are converted into equalities using slack variables, and the resulting equalities are handled with an ℓ_2 penalty method. While the proofs of convergence differ slightly, the convergence results for the two approaches are almost identical. The differences are: (1) The objective and constraint functions need to be twice continuously differentiable everywhere for the infeasible approach, whereas such differentiability is only needed for the interior of the feasible region for the quasi-feasible approach, and (2) The quasi-feasible approach requires a nonempty interior for the feasible region, whereas the infeasible approach does not. The first is more restrictive for the infeasible approach, while the second is more restrictive for the quasi-feasible approach. While we could also easily use the infeasible approach, in this paper, we focus on the combination of the quasi-feasible approach of [6] with the ℓ_1 penalty method. Doing so simplifies the notation as we will not need slack variables, but it is still easy to readily locate a feasible solution for the problem. We will assume that twice continuous differentiability will be required everywhere, but as we will discuss later, a less restrictive differentiability requirement may suffice.

2.1. **The quasi-feasible interior-point method for** (3)**.** The barrier problem associated with (3) for the quasi-feasible approach is

$$
\begin{array}{ll}
\min\limits_{x,\xi} & \rho(x,\xi;\mu) = f(x) + d^T\xi - \mu \sum\limits_{i=1}^m \log\left(r_i(x)+\xi_i\right) - \mu \sum\limits_{i=1}^m \log\left(\xi_i\right) \\
\text{s.t.} & r(x) + \xi > 0 \\
 & \xi > 0,
\end{array} \tag{4}
$$

where $\mu > 0$ is the barrier parameter. Using this approach, therefore, requires that there exist a strictly feasible solution to (3). Such a solution may be hard to obtain without the ℓ_1 penalty on the inequality constraints, but simply setting $\xi_i > \max\{-r_i(x), 0\}$ will ensure that a strictly feasible solution for any x can be found readily.

The first-order conditions for this system are

$$
\begin{array}{lll}
\nabla f(x) - A(x)^T u & = & 0 \\
d - u - \psi & = & 0 \\
U(r(x)+\xi) - \mu e & = & 0 \\
\Psi\xi - \mu e & = & 0,
\end{array} \tag{5}
$$

where $A(x)$ is the transpose of the Jacobian of $r(x)$ and $u, \psi > 0$ are the Lagrange multipliers. By convention, U and Ψ are diagonal matrices with entries from the vectors u and ψ, respectively. In (5) and throughout the paper, e denotes a vector of ones of appropriate dimension.

Applying Newton's Method, at each iteration k we have the following system to solve:

$$
\begin{array}{lcl}
H(x^k,u^k)\Delta x^k - A(x^k)^T v^k & = & -\nabla f(x^k) \\
v^k + \lambda^k & = & d \\
(R(x^k)+\Xi^k)v^k + U^k(A(x^k)\Delta x^k + \Delta\xi^k) & = & \mu e \\
\Psi^k \Delta\xi^k + \Xi^k\lambda^k & = & \mu e,
\end{array}
\tag{6}
$$

where $R(x^k)$ and Ξ^k are diagonal matrices with entries from the vectors $r(x^k)$ and ξ^k, respectively,

$$H(x^k,u^k) = \nabla^2 f(x^k) - \sum_i u_i^k \nabla^2 r_i(x^k),$$

$v^k = u^k + \Delta u^k$, and $\lambda^k = \psi^k + \Delta\psi^k$. Then, we let

$$
\begin{array}{lcl}
x^{k+1} & = & x^k + \alpha^k \Delta x^k \\
\xi^{k+1} & = & \xi^k + \alpha^k \Delta \xi^k
\end{array}
\tag{7}
$$

where $\alpha^k \in (0,1)$ is the steplength, such that

(8)

$$
\begin{array}{lcl}
r(x^{k+1}) + \xi^{k+1} & > & 0 \\
\xi^{k+1} & > & 0 \\
\rho(x^{k+1},\xi^{k+1};\mu) - \rho(x^k,\xi^k;\mu) & < & \tau(\nabla_x\rho(x^k,\xi^k;\mu)\Delta x^k + \nabla_\xi\rho(x^k,\xi^k;\mu)\Delta\xi^k),
\end{array}
$$

where the last inequality is a standard Armijo condition on the decrease of the barrier objective function of (4). We update the Lagrange multipliers as follows:

(9)

$$
\begin{array}{lcl}
u_i^{k+1} & = & \left\{ \begin{array}{ll}
v_i^k, & \text{if } \min\{\theta u_i^k, \frac{\mu}{r_i(x^k)+\xi^k}\} \le v_i^k \le \frac{\mu\gamma}{r_i(x^k)+\xi^k} \\
\min\{\theta u_i^k, \frac{\mu}{r_i(x^k)+\xi^k}\}, & \text{if } v_i^k < \min\{\theta u_i^k, \frac{\mu}{r_i(x^k)+\xi^k}\} \\
\frac{\mu\gamma}{r_i(x^k)+\xi^k}, & \text{if } v_i^k > \frac{\mu\gamma}{r_i(x^k)+\xi^k}
\end{array}\right. \\
\psi_i^{k+1} & = & \left\{ \begin{array}{ll}
\lambda_i^k, & \text{if } \min\{\theta\psi_i^k, \frac{\mu}{\xi^k}\} \le \lambda_i^k \le \frac{\mu\gamma}{\xi^k} \\
\min\{\theta\psi_i^k, \frac{\mu}{\xi^k}\}, & \text{if } \lambda_i^k < \min\{\theta\psi_i^k, \frac{\mu}{\xi^k}\} \\
\frac{\mu\gamma}{\xi^k}, & \text{if } \lambda_i^k > \frac{\mu\gamma}{\xi^k}
\end{array}\right. ,
\end{array}
$$

where $\theta \in (0,1)$ and $\gamma > 1$. The iterations continue until the first-order conditions (5) are satisfied to within a tolerance ϵ_μ for the current value of the barrier parameter, that is

$$
\text{res}(x,\xi,u,\psi) = \left\| \begin{bmatrix} \nabla f(x) - A(x)^T u \\ d - u - \psi \\ U(r(x)+\xi) - \mu e \\ \Psi\xi - \mu e \end{bmatrix} \right\| < \epsilon_\mu.
\tag{10}
$$

Note that v and λ can be eliminated from the system (6) to obtain

$$
\mathcal{M}^k \begin{pmatrix} \Delta x^k \\ \Delta \xi^k \end{pmatrix} = \begin{pmatrix} -\nabla f(x^k) + \mu A(x^k)^T (R(x^k)+\Xi^k)^{-1} e \\ \mu(R(x^k)+\Xi^k)^{-1}e + \mu(\Xi^k)^{-1}e - d \end{pmatrix},
\tag{11}
$$

Set $\epsilon_\mu > 0$, $\gamma > 1$, $\theta \in (0,1)$.
Initialize $x^{(0)} \in \mathbb{R}^n$, $u^{(0)} > 0$, $\psi^{(0)} > 0$, and
$\xi_i^{(0)} > \max\{-r_i(x), 0\}, i = 1, \ldots, m$.
Let $k = 0$.
while $res(x^k, \xi^k, u^k, \psi^k) > \epsilon_\mu$ **do**
  if (12) *does not hold* **then** modify $H(x^k, u^k)$ such that it is positive definite.
  Compute $(\Delta x^k, \Delta\xi^k, v^k, \lambda^k)$ from (6).
  Compute steplength α such that (8) and an Armijo condition are satisfied.
  Set u^{k+1} and ψ^{k+1} as in (9) and x^{k+1} and ξ^{k+1} as in (7).
  Set $k \leftarrow k + 1$.
end

Algorithm 1: A description of the solution algorithm for (4) for a fixed barrier parameter μ.

where

$$\begin{aligned} \mathcal{M}^k &= \begin{bmatrix} \hat{H}(x^k, \xi^k, u^k) & A(x^k)^T (R(x^k) + \Xi^k)^{-1} U^k \\ (R(x^k) + \Xi^k)^{-1} U^k A(x^k) & (R(x^k) + \Xi^k)^{-1} U^k + (\Xi^k)^{-1} \Psi^k \end{bmatrix} \\ \hat{H}(x^k, \xi^k, u^k) &= H(x^k, u^k) + A(x^k)^T (R(x^k) + \Xi^k)^{-1} U^k A(x^k). \end{aligned}$$

In order to use the convergence results of [6], we need to ensure that, for each iteration, the matrix $\mathcal{M}^k$ is sufficiently positive definite. Since we have that

$$(R(x^k) + \Xi^k)^{-1} U^k + (\Xi^k)^{-1} \Psi^k \succeq 0,$$

we need to have

$$\begin{aligned} & H(x^k, \xi^k, u^k) + A(x)^T (R(x^k) + \Xi^k)^{-1} U^k \\ & \left((R(x^k) + \Xi^k)^{-1} U^k + (\Xi^k)^{-1} \Psi^k\right)^{-1} (\Xi^k)^{-1} \Psi^k A(x) \quad \succeq 0 \end{aligned} \tag{12}$$

at each iteration. However, this may not always be the case. Then, we will modify $H(x^k, u^k)$ by adding a term of the form δI where δ is chosen to ensure that (12) holds. While any value of δ above a certain threshold will work in theory, we would also like to keep it as small as possible in order to make this algorithm work in practice, as larger values of δ will make the algorithm behave as in steepest descent which is not desirable.

The algorithm provided above is a simplified version of Algorithm I presented in [6]. There are several steps that are not needed, such as the penalty parameter update as we no longer have equality constraints and the termination due to MFCQ failure as (3) always satisfies MFCQ. We will give a formal proof of convergence for this algorithm in the next section.

This algorithm solves (4) for a fixed value of the barrier parameter μ. In order to solve (3), we will need to solve a series of problems of the form (4) for decreasing values of μ. In order to do so, we will apply Algorithm II from [6]. A formal statement of this algorithm is provided below as Algorithm 2.

In the next section, we will show that Algorithm 2 will always terminate finitely and find a point that satisfies the first-order optimality conditions (5) of the penalty problem (3) for fixed values of the penalty parameters d. If $\|\xi\|$ is sufficiently close 0 at this point, we will declare it as a first-order point for (2). Otherwise, we will increase the penalty parameters and solve the problem again. The resulting

Set $\mu_0 > 0$, $\epsilon_{\mu_0} > 0$, $\beta \in (0, 1)$ and $\hat{\epsilon} > 0$.
Initialize $x^{(0)} \in \mathbb{R}^n$, $u^{(0)} > 0$, $\psi^{(0)} > 0$, and
$\xi_i^{(0)} > \max\{-r_i(x), 0\}, i = 1, \ldots, m$.
Let $j = 0$.
while $res(x^j, \xi^j, u^j, \psi^j) > \hat{\epsilon}$ **do**
  Starting from $(x^j, \xi^j, u^j, \psi^j)$, apply Algorithm 1 to solve (4) with barrier parameter μ_j and stopping tolerance ϵ_{μ_j}. Let the solution be $(x^{j+1}, \xi^{j+1}, u^{j+1}, \psi^{j+1})$.
  Set $\mu_{j+1} \leftarrow \beta\mu_j$ and $\epsilon_{j+1} \leftarrow \beta\epsilon_j$.
  Set $j \leftarrow j + 1$.
end

Algorithm 2: A description of the solution algorithm for (3) for fixed penalty parameters d.

Initialize $x^{(0)} \in \mathbb{R}^n$, $u^{(0)} > 0$, $\psi^{(0)} > 0$, and
$\xi_i^{(0)} > \max\{-r_i(x), 0\}, i = 1, \ldots, m$.
Set $d > u^{(0)} + \psi^{(0)}$, $\epsilon_{\mu_0} > 0$, $\nu > 1$ and $\hat{\epsilon} > 0$.
Let $i = 0$.
repeat
  Starting from $(x^i, \xi^i, u^i, \psi^i)$, apply Algorithm 2 to solve (3) with penalty parameters d_i. Let the solution be $(x^{i+1}, \xi^{i+1}, u^{i+1}, \psi^{i+1})$.
  Set $d_{i+1} \leftarrow \nu d_i$.
  Set $i \leftarrow i + 1$.
until $\|\xi^i\| \le \hat{\epsilon}$;

Algorithm 3: A description of the solution algorithm for (2).

algorithm will be shown to either converge to a first-order point of (2) for sufficiently large but finite values of the penalty parameters, or to update the penalty parameters infinitely many times while either approaching a point that minimizes the infeasibility or that is a Fritz John point of (2) that does not satisfy MFCQ. A formal description of the algorithm is given as Algorithm 3.

3. Global Convergence Results

We now present a formal convergence proof for Algorithm 3. We start by stating our assumptions and proving some intermediate results. Lemmas 4 and 5 below will show convergence of Algorithms 1 and 2, respectively, by using the convergence results of [6]. Theorem 1 will establish the convergence of Algorithm 3.

Assumption 1. *f and r are twice continuously differentiable everywhere.*

This assumption also implies that the functions g and h of (1) are twice continuously differentiable. As stated before, this assumption is more restrictive than actually needed. It is sufficient to assume that f and r are twice continuously differentiable in a region that can be defined by bounds on x or on ξ. Another option is to use the primal-dual penalty approach proposed in [4] which naturally provides upper bounds for the primal variables.

Assumption 2. *While applying Algorithm 1, the sequence $\{x^k\}$ lies in bounded set.*

Assumption 2 can be guaranteed by using sufficiently large simple bounds on x or by using the primal-dual penalty approach of [4], as well. We omit these transformations for the sake of simplicity in this paper.

Lemma 1. *Assume that the sequence $\{x\}$ generated by Algorithm 1 remains bounded. Then, the sequence $\{\xi\}$ will also remain bounded.*

Proof. By Assumption 1, if the sequence $\{x\}$ generated by Algorithm 1 remains bounded, then the sequence of constraint values $\{r(x)\}$ will also remain bounded. Assume now that $\xi_i \to \infty$ for some i. Then, we must have that $v, \lambda \to 0$ in order to satisfy the last two conditions of (6) for a fixed value of μ. However, that would violate the second equation in (6). Therefore, $\{\xi\}$ must remain bounded. □

Assumption 1 is a standard assumption for all algorithms using Newton's Method. As stated, Assumption 2, along with Lemma 1, implies that the primal iterates generated by this feasible algorithm remain bounded, which is also a standard assumption for convergence proofs. In fact, they are listed as assumptions A2 and A3 in [6]. These are rather mild assumptions, and, in fact, they are the only assumptions we will need to prove convergence of our approach. The remaining assumptions made in [6] will always be satisfied as a result of using the ℓ_1 penalty as shown below in Lemmas 2 and 3.

Lemma 2. *The feasible region of* (3) *has a nonempty interior.*

Proof. Let $\hat{x} \in \mathbb{R}^n$ and let $\hat{\xi}_i > \max\{-r_i(\hat{x}), 0\}$, $i = 1, \ldots, m$. Doing so ensures that

$$\begin{array}{lcl} r_i(\hat{x}) + \hat{\xi} & > & 0 \\ \hat{\xi} & > & 0. \end{array}$$

Therefore, $(\hat{x}, \hat{\xi})$ lies in the interior of the feasible region of (3). This proves that (3) has a nonempty interior. □

Lemma 3. *The sequence of modified Hessians, $\{H^k\}$, is bounded.*

Proof. The second equation in (6) is

$$v^k + \lambda^k = d.$$

Therefore, the dual iterates will always remain bounded above by the penalty parameter. By this fact, and by Assumption 1, the sequence of Hessians is bounded. Thus, the sequence of modified Hessians is also bounded. □

As stated, Lemmas 2 and 3 correspond to assumptions A1 and A4 of [6], respectively. Therefore, the ℓ_1 penalty ensures that we will need fewer assumptions in order to prove convergence of the interior-point method and strengthens the results of [6].

We will now focus on showing that the proposed Algorithms 1 and 2 are equivalent to Algorithms I and II for the quasi-feasible approach of [6]. The ℓ_1 penalty will, in fact, allow us to simplify the algorithms by eliminating certain steps. We start by presenting some preliminaries and then establish the convergence of Algorithms 1 and 2 in Lemmas 5 and 6.

Definition 1. *Given a point* (x, ξ) *feasible for* (3)*, we say that the Mangasarian-Fromowitz Constraint Qualification (MFCQ) holds at* (x, ξ) *if either* (x, ξ) *is strictly feasible or there exists vectors* $s \in \mathbb{R}^n$ *and* $t \in \mathbb{R}^m$ *such that*

$$\begin{array}{lll} \nabla r_i(x)^T s + t_i > 0, & \text{if } r_i(x) + \xi_i = 0, & i = 1, \ldots, m \\ t_i > 0, & \text{if } \xi_i = 0, & i = 1, \ldots, m. \end{array}$$

Lemma 4. *MFCQ holds on the feasible region of* (3)*.*

Proof. Let (x, ξ) lie in the feasible region of (3). The existence of (x, ξ) is guaranteed by Lemma 2. If (x, ξ) is strictly feasible, then MFCQ holds at (x, ξ). Otherwise, let $s = 0$ and $t = e$. Therefore,

$$\begin{array}{lll} \nabla r_i(x)^T s + t_i = 1, & \text{if } r_i(x) + \xi_i = 0, & i = 1, \ldots, m \\ t_i = 1, & \text{if } \xi_i = 0, & i = 1, \ldots, m, \end{array}$$

and MFCQ holds at (x, ξ). Thus, MFCQ holds on the feasible region of (3). □

Definition 2. *The point* (x, ξ) *is a Karush-Kuhn-Tucker (KKT) point of* (4) *if it is a feasible point of* (3) *and* $res(x, \xi, u, \psi) = 0$.

Definition 3. *The point* (x, ξ) *is an approximate KKT point of* (4) *if it is a feasible point of* (3) *and* $res(x, \xi, u, \psi) < \chi$ *for some* $\chi > 0$.

Lemma 5. *Let Assumptions 1 and 2 hold. Then, Algorithm 1 will always find an approximate KKT point of* (4) *satisfying termination criteria* (10) *in a finite number of iterations.*

Proof. Along with Assumptions 1 and 2, Lemmas 2 and 3 show that assumptions A1-A4 of [6] hold. Therefore, we can use Theorem 3.12 of [6]. By Lemma 4, MFCQ always holds, so the second part of Step 1 of Algorithm I of [6] is not needed and, therefore, not included in Algorithm 1. Also, there is no need for an ℓ_2 penalty term because (3) does not have equality constraints. Therefore, there is no ℓ_2 penalty parameter that could be updated indefinitely. As such, a penalty parameter update step is not included in Algorithm 1. Thus, our Algorithm 1 is identical to Algorithm I of [6] for solving (4). By Theorem 3.12 of [6], we will always identify an approximate KKT point of the barrier problem, satisfying the termination criteria (10) in a finite number of iterations. □

Thus, Lemma 5 shows that Algorithm 1 will always succeed in finding an approximate KKT point for the barrier problem for a fixed value of the barrier parameter μ. We need to now show that we can solve (3) for a fixed value of the penalty parameter, which means showing that the sequence of approximate solutions found for the barrier problem for decreasing values of μ converges to a first-order point of the penalty problem. Again, we will do so by showing that our proposed algorithm, Algorithm 2, is equivalent to Algorithm II of [6], allowing us to use its convergence results.

Definition 4. *A point* (x, ξ, u, ψ) *is a first-order point of the penalty problem* (3) *if* (x, ξ) *is feasible for* (3) *and conditions* (5) *are satisfied with* $\mu = 0$.

Lemma 6. *Let Assumptions 1 and 2 hold. Ignoring the termination condition of Algorithm 2, any limit point of the sequence* $(x^j, \xi^j, u^j, \psi^j)$ *generated by Algorithm 2 is a first-order point of the penalty problem* (3)*.*

Proof. Along with Assumptions 1 and 2, Lemmas 2 and 3 show that assumptions A1-A4 of [6] hold. By Lemma 4, we have that MFCQ holds on the feasible region of (3), and by Lemma 2, we have that (3) always has a local feasible solution. Therefore, we satisfy all the assumptions of Theorem 3.14 of [6]. Since there is no ℓ_2 penalty function, there is no penalty parameter to keep bounded. Because our Algorithm 2 is identical to Algorithm II of [6], Theorem 3.14 applies here as well. Therefore, ignoring the termination condition of Algorithm 2, any limit point of the sequence $(x^j, \xi^j, u^j, \psi^j)$ generated by Algorithm 2 is a first-order point of the penalty problem (3). □

Lemma 6 shows that Algorithm 2 will always find a first-order point of the penalty problem (3) for fixed values of the penalty parameters d. We will now analyze Algorithm 3, which solves (3) for increasing values of d in order to find a first-order point for (2) if such a point exists. We begin with some preliminaries and then state our convergence result in Theorem 1.

Definition 5. *A point (x, u) is a first-order point of the problem (2) if*

$$\begin{array}{lcl} r(x) & \geq & 0 \\ \nabla f(x) - A(x)^T u & = & 0 \\ Ur(x) & = & 0 \\ u & \geq & 0. \end{array}$$

Definition 6. *A point x is a Fritz John point of the problem (2) if there exist $u_0 \in \mathbb{R}$ and $u \in \mathbb{R}^m$ with $(u_0, u) \neq 0$ such that*

$$\begin{array}{lcl} r(x) & \geq & 0 \\ u_0 \nabla f(x) - A(x)^T u & = & 0 \\ u & \geq & 0. \end{array}$$

Definition 7. *The infeasibility problem corresponding to (2) is*

$$\begin{array}{lll} \min\limits_{x,\xi} & e^T \xi & \\ s.t. & r(x) + \xi & \geq 0 \\ & \xi & \geq 0. \end{array}$$

Definition 8. *A point (x, ξ, u, ψ) is a first-order point of the infeasibility problem corresponding to (2) if*

$$\begin{array}{lcl} r(x) + \xi & \geq & 0 \\ \xi & \geq & 0 \\ A(x)^T u & = & 0 \\ e - u - \psi & = & 0 \\ U(r(x) + \xi) & = & 0 \\ \Psi \xi & = & 0, \end{array}$$

Theorem 1. *Let Assumptions 1 and 2 hold. Then, one of the following outcomes occurs:*

1. *Algorithm 3 terminates successfully, and the limit point of any subsequence $\{x^i, u^i\}$ is a first-order point of (2).*
2. *The termination criteria of Algorithm 3 are never met, in which case there exists a limit point of iterates $\{x^i\}$ of Algorithm 3 that is either a first-order point of the infeasibility problem or a Fritz John point of (2) that does not satisfy MFCQ.*

Proof. By Lemma 6, Algorithm 2 terminates successfully for any d and finds (approximately) a first-order point of (3).

Case 1: For a sequence of solutions $\{x^i, \xi^i, u^i, \psi^i\}$ to Algorithm 2, assume that we have a finite subsequence such that $\{\|\xi^i\|\}$ approaches 0. The solutions to Algorithm 2 (approximately) satisfy the conditions

$$
\begin{array}{lcl}
r(x^i) + \xi^i & \geq & 0 \\
\xi^i & \geq & 0 \\
\nabla f(x^i) - A(x^i)^T u^i & = & 0 \\
d^i - u^i - \psi^i & = & 0 \\
U^i(r(x^i) + \xi^i) & = & 0 \\
\Psi^i \xi^i & = & 0,
\end{array}
\tag{13}
$$

which reduce to the first-order conditions of (2) as $\|\xi^i\|$ approaches 0.

Case 2: For a sequence of solutions $\{x^i, \xi^i, u^i, \psi^i\}$ to Algorithm 2, assume that we have a subsequence such that $\{\|\xi^i\|\}$ approaches 0 only as $i \to \infty$. This means that $u^i \to \infty$ as $i \to \infty$ by (13). Thus, as $i \to \infty$, the algorithm approaches a solution satisfying (13) that has unbounded Lagrange multipliers. This means that MFCQ for (2) does not hold at this solution. As such, the algorithm approaches a Fritz John point that does not satisfy MFCQ.

Case 3: For a sequence of solutions $\{x^i, \xi^i, u^i, \psi^i\}$ to Algorithm 2, assume that there is no subsequence such that $\{\|\xi^i\|\}$ approaches 0. Thus, at each iteration of Algorithm 3, the penalty parameter d^i are updated so that $d^i \to \infty$.

We first show that as $d^i \to \infty$, the sequence of solutions $\{x^i\}$ obtained from Algorithm 2 remain bounded. By the optimality of (x^i, ξ^i) and (x^{i+j}, ξ^{i+j}), $j \geq 1$, for their respective problems, if ν is small enough such that Algorithm 2 converges to a local solution, we have that

$$
\begin{array}{lcl}
f(x^i) + (d^i)^T \xi^i & \leq & f(x^{i+j}) + (d^i)^T \xi^{i+j} \\
f(x^{i+j}) + \nu^j (d^i)^T \xi^{i+j} & \leq & f(x^i) + \nu^j (d^i)^T \xi^i,
\end{array}
$$

which implies that

$$
(d^i)^T \xi^i \geq (d^i)^T \xi^{i+j}, \quad j = 1, \ldots.
$$

Since both d^i and ξ^i are bounded below by 0, in order for this inequality to hold for all $j \geq 1$, we must have that the sequence $\{\xi^i\}$ must remain bounded. This means that the sequence $\{\max_{1\leq k\leq m}\{r_k(x^i), 0\}\}$, which is the sequence of constraint violations for (1), remains bounded. By Assumption 2, therefore, the sequence $\{x^i\}$ must also remain bounded.

Let us now examine the behavior of the sequence $\{x^i, \xi^i\}$ as $d \to \infty$. We start by defining the set

$$
J(x, \xi) = \{j : -r_j(x) = \xi_j, j = 1, \ldots, m\}
$$

for some (x, ξ). The set $J(x, \xi)$ corresponds to those constraints of the original problem (1) that are either violated or active at the point (x, ξ). Then, (x^i, ξ^i) is a local minimum found by Algorithm 2 for the penalty problem only if the following condition holds for all directions $s \in \mathbb{R}^n$ such that $\|s\| = 1$:

$$
\nabla f(x^i)^T s + \sum_{j \in J(x^i, \xi^i)} d^i_j \nabla r_j(x^i)^T s \geq 0. \tag{14}
$$

We also know that (x^i, ξ^i) is a first-order point of the infeasibility problem only if

$$\sum_{j \in J(x^i, \xi^i)} \nabla r_j(x^i)^T s \geq 0$$

for all such directions.

Now assume that (x^i, ξ^i) is not a first-order point of the infeasibility problem for any finite i. Then, for all i, there exists a direction $s \in \mathbb{R}^n$ with $\|s\| = 1$ that is a descent direction for the infeasibility, or

$$\sum_{j \in J(x^i, \xi^i)} \nabla r_j(x^i)^T s < 0.$$

By Assumption 1 and the boundedness of $\{x^i\}$, the sequence $\{\nabla f(x^i)\}$ remains bounded as well. Thus, for large enough d^i, the condition (14) cannot hold. This contradicts the optimality of (x^i, ξ^i) for the penalty problems. It, therefore, follows that as $i \to \infty$, $\{x^i, \xi^i\}$ approaches arbitrarily close to a minimizer of the infeasibility problem. □

Therefore, by Theorem 1, we will be able to find a first-order point of (2) if one exists. Otherwise, the problem is either infeasible or has an optimum that does not satisfy the first-order conditions of (2). In these cases, the penalty parameters will be updated infinitely many times and Algorithm 3 will not terminate. In an implementation, it is easy to include an infeasibility detection phase, where Algorithm 3 switches to solving the infeasibility problem of (2) after a finite number of iterations. By Lemma 6, we are guaranteed to find a first-order point of the infeasibility problem, as it is simply the penalty problem (3) with $f(x) = 0$ everywhere and $d = e$. Therefore, with a simple modification, we can show that a certificate of infeasibility can be issued when necessary after a finite number of iterations of Algorithm 3.

4. Local Convergence Results

In the previous section, we showed that our proposed algorithm gets arbitrarily close to a solution of (2), if one exists. In this section, we examine the local convergence properties of the proposed algorithm, including its rate of convergence. Most local convergence proofs such as the one in [7] require the satisfaction of certain constraint qualifications:

Definition 9. *A feasible solution for an NLP satisfies the linear independence constraint qualification (LICQ) for that NLP if the active constraint gradients at the solution are linearly independent.*

However, the constraint Jacobian for (3) is given by

$$\begin{bmatrix} A(x) & I \\ 0 & I \end{bmatrix},$$

and if LICQ does not hold for the original problem, then it will not hold for the penalty problem, either. Thus, requiring LICQ limits the applicability of any algorithm, including ours, to large classes of problems such as MPECs which do not satisfy this qualification. Therefore, it is our goal in this paper to use a less restrictive constraint qualification, namely MFCQ, to obtain local convergence results. As shown in Lemma 4 of the previous section, MFCQ always holds for (3). We will employ the theory developed in [12] to show local convergence under MFCQ.

First, we need to make some modifications to our algorithm, which will allow us to use the results of [12] and to show fast local convergence. In particular, we need to modify the computation of the dual variables when the barrier parameter is sufficiently close to 0. Instead of solving (6) and then using (9) to compute the dual variables, we will solve the system (11) to obtain the primal step directions and set the values of the dual variables as follows:

$$\begin{aligned} u_i^{k+1} &= \frac{\mu}{r_i(x^k) + \xi_i^k} \\ \psi_i^{k+1} &= \frac{\mu}{\xi_i^k}. \end{aligned}$$

Note that doing so converts the algorithm to a purely primal one, but this is not a substantial change in the structure of the algorithm since the dual step calculations were not entirely independent when using the formula (9).

The second modification we will make is the choice of the barrier parameter at each iteration of Algorithm 2. Rather than setting $\mu_{j+1} \leftarrow \beta\mu_j$, we will update the barrier parameter as follows:

$$\mu^{j+1} \leftarrow \min\{\beta\sigma(x^j, \xi^j, u^j, \psi^j), \sigma(x^j, \xi^j, u^j, \psi^j)^2\},$$

where

$$\sigma(x, \xi, u, \psi) = \max\{\| \begin{bmatrix} \nabla f(x) - A(x)^T u \\ d - u - \psi \end{bmatrix} \|, \| \begin{bmatrix} U(r(x) + \xi) \\ \Psi\xi \end{bmatrix} \|\}.$$

Note that doing so ensures that $\mu^{j+1} \leftarrow \beta\mu^j$ still holds away from the solution, as $\sigma(x, \xi, u, \psi)$ will be dominated by the complementarity terms which are all close to μ^j. Close to the solution, however, the barrier parameter will be decreased at a quadratic rate.

There are two assumptions required for fast local convergence: strict complementarity and second-order sufficiency. We now define these concepts.

Definition 10. *For a strict local minimum x^* of* (2)*, strict complementarity holds if there exists a corresponding Lagrange multiplier u^* satisfying the first-order conditions for* (2) *with*

$$u_i^* + r_i(x^*) > 0, \quad i = 1, 2, \ldots, m.$$

Definition 11. *For a strict local minimum x^* of* (2)*, the second-order sufficient condition (SOSC) holds if there exists a $\phi > 0$ such that*

$$s^T H(x^*, u^*)s \geq \phi\|s\|^2,$$

for each of its corresponding Lagrange multipliers u^* and for all $s \in \mathbb{R}^n$ with $s \neq 0$ and $\nabla r_i(x)^T s = 0$ for each active constraint i.

Lemma 7. *Let strict complementarity hold for* (2) *at a strict local minimum x^* with Lagrange multiplier u^*. Then, it holds for* (3) *with penalty parameter $d > u^*$ at $(x^*, 0)$.*

Proof. The condition

$$u_i + (r_i(x) + \xi_i) > 0, \quad i = 1, 2, \ldots, m$$

reduces to

$$u_i + r_i(x) > 0, \quad i = 1, 2, \ldots, m$$

when $\xi = 0$. Also, the first-order condition

$$d - u - \psi = 0$$

implies that $\psi^* = d - u* > 0$. Therefore, there exists a Lagrange multiplier (u^*, ψ^*) such that strict complementarity holds at $(x^*, 0)$. □

Lemma 8. *If SOSC holds for* (2) *at a strict local minimum* x^**, then it holds for* (3) *at* $(x^*, 0)$*.*

Proof. At $(x^*, 0)$, we have that

$$\begin{array}{lcl} r(x^*) + \xi^* & = & r(x^*) \\ \xi^* & = & 0. \end{array}$$

Thus, the set of active constraints for (3) at $(x^*, 0)$ is the set of active constraints for (2) at x^* along with the nonnegativity constraints on the relaxation variables.

For each $s \in \mathbb{R}^n$ with $s \neq 0$ and $\nabla r_i(x)^T s = 0$ for each active constraint i, we define $\hat{s} = (s, 0) \in \mathbb{R}^{n+m}$. This means that $\hat{s} \neq 0$, and we have that

$$\begin{bmatrix} \nabla r_i(x) \\ e_i \end{bmatrix}^T \hat{s} = 0$$

for each active inequality constraint i and

$$\begin{bmatrix} 0 \\ e_i \end{bmatrix}^T \hat{s} = 0$$

for each nonnegativity constraints i, with e_i denoting the vector of zeros except for a one in the ith place. Thus, for (3), we have that

$$\hat{s}^T \begin{bmatrix} H(x^*, u^*) & 0 \\ 0 & 0 \end{bmatrix} \hat{s} = s^T H(x^*, u^*) s \geq \phi \|s\|^2.$$

Therefore, SOSC holds for (3) at $(x^*, 0)$. □

Note that since MFCQ does not imply the uniqueness of the Lagrange multipliers corresponding the x^* in the first-order conditions, the above definitions and lemmas accommodate the existence of multiple vectors. Therefore, the definition of strict complementarity only requires one among possibly many optimal vectors of Lagrange multipliers, and SOSC is defined to hold at every vector of optimal Lagrange multipliers.

An additional assumption in [12] is that at least one of the constraints is active at the optimal solution. We now show that this assumption will always hold for (3).

Lemma 9. *For a strict local minimum* (x^*, ξ^*) *of* (3)*,* $r_i(x^*) + \xi_i^* = 0$ *or* $\xi_i^* = 0$ *for some* $i = 1, 2, \ldots, m$*.*

Proof. Suppose that for all $i = 1, 2, \ldots, m$, we have that

$$\begin{array}{lcl} r_i(x^*) + \xi_i^* & > & 0 \\ \xi_i^* & > & 0. \end{array}$$

Let

$$\hat{\xi}_i = \min\{r_i(x^*) + \xi_i^*, \xi_i^*\}.$$

Thus, we have that $\hat{\xi} > 0$, and, therefore, $\xi^* - \hat{\xi} < \xi^*$ componentwise. Note also that by the way we constructed $\xi^* - \hat{\xi}$, $(x^*, \xi^* - \hat{\xi})$ is feasible for (3). Since

$$f(x^*) + d^T(\xi^* - \hat{\xi}) < f(x^*) + d^T \xi^*,$$

(x^*, ξ^*) cannot be the strict local minimum, and we have a contradiction. □

Lemma 10. *Under Assumptions 1, 2, SOSC, and strict complementarity, with the above modifications for fast local convergence, Algorithm 2 will always find a strict local optimum of the penalty problem* (3) *and do so at a Q-quadratic rate.*

Proof. By Lemma 4, (3) always satisfies MFCQ. Therefore, local convergence and its rate of Algorithm 1 for the barrier problem follow from Theorem 8 of [12]. The results of Theorems 11 and 12 of [12] and the updated choice of the barrier parameter guarantee that Algorithm 2 will solve the penalty problem and do so Q-quadratically. □

We now show that with the modifications made above, Algorithm 3 converges locally to a strict local minimum of (2) and does so at a quadratic rate.

Theorem 2. *If Assumptions 1, 2, SOSC, and strict complementarity hold for* (2) *at a first-order point* x^**, then for sufficiently large but finite* d^**, a solution* $(x, \xi, u, \psi) = (x^*, 0, u^*, d^* - u^*)$ *exists for* (3)*. If Algorithm 3 is applied with the above modifications at a point* $(x^i, \xi^i, u^i, \psi^i)$ *sufficiently close to this solution with the penalty parameter set to* d^**, then it converges to* x^* *with no further update of the penalty parameter needed, and the rate of convergence is Q-quadratic.*

Proof. The first-order conditions (5) for (3) reduce to those of (2) at the solution $(x^*, 0, u^*, , d^* - u^*)$ for $d^* > u^*$, so it is a first-order point for (3). Since MFCQ holds for the penalty problem by Lemma 4, there exists a u^* which is finitely-valued, and therefore, it is sufficient to have d^* large and finite. This means that when the penalty parameter is set to d^* and we start sufficiently close, we only need one iteration of Algorithm 2 to find the solution. Thus, by Lemma 7, Algorithm 3 converges to the solution and does so quadratically. □

5. Behavior of the Penalty Problem Near an Isolated Minimizer

Note that Theorem 2 essentially assumes that we have a KKT solution. However, we can still prove some results for a strict local minimum, x^*, that may not necessarily be a KKT solution. We proved a similar result in Theorem 3 of [2], but that was done with some assumptions, which we now drop.

In this section, we will establish that there is a sequence of solutions to the penalized problem (2) which will converge to x^*(even if x^* is not a KKT solution). Again, we are not able to assert that the our algorithm will *necessarily* follow such a sequence. However, absent truly pathological behavior, the algorithm often will be able to track such a path if a starting point is chosen sufficiently close to x^* for sufficiently large d. What we have provided is an explanation for behavior that has in fact been seen in practice (see [2]), not a guarantee that such behavior will always occur.

In the following results, the notation $B_{\kappa>0}(x)$ refers to a ball of positive radius κ centered at the point x.

Lemma 11. *For all* $B_{\kappa>0}(x^*)$ *there exists a* $\bar{d}$ *such that* $x^i \in B_\kappa$ *for all* $d^i > \bar{d}$.

Proof. We choose r to be small enough such that $f(x^*) < f(x)$ for all $x \in \overline{B_\kappa(x^*)}$ with x feasible for (2). Suppose $B_\kappa(x^*)$ does not contain x^i for some d^i. The function $f(x) + d^T\xi$ attains a global minimum over the set of solutions satisfying

$$\begin{array}{lcl} r(x) + \xi & \geq & 0 \\ \xi & \geq & 0 \\ x & \in & \overline{B_\kappa(x^*)}, \end{array}$$

since x is restricted to a closed and bounded set. Call this global minimum $(\hat{x}, \hat{\xi})$. By assumption this global minimum must lie on the boundary of $\overline{B_\kappa(x^*)}$ and must be better than any point in the interior, that is,

$$f(\hat{x}) + d^T \hat{\xi} < f(x) + d^T \xi \tag{15}$$

Furthermore, it must be true that $\|\hat{\xi}\| > 0$, since otherwise $\hat{x}$ would be feasible for (2) with $f(\hat{x}) < f(x^*)$ which cannot happen. Now let S be the set of all such d just described (i.e. those for which the penalty problem has no solutions in $B_\kappa(x^*)$). We assume for the moment that S is unbounded. If we have that

$$\exists \epsilon > 0 \text{ s.t. } \hat{\xi} > \epsilon \, \forall d \in S \ , \tag{16}$$

then it is not hard to see that, since f is bounded on $\overline{B_\kappa(x^*)}$, we must have for sufficiently large d in S,

$$f(\hat{x}) + d^T \hat{\xi} > f(x^*),$$

which of course contradicts (15). So suppose (16) does not hold. Then we have that, for some sequence $\{d^k\}$ in S: $\lim_{k\to\infty} \hat{x^k} = \bar{x}$ which must be feasible for (2) and lie in $\overline{B_\kappa(x^*)}$ (since a closed set must contain its limit points). It must be true that $f(\bar{x}) > f(x^*)$, and yet $\lim_{k\to\infty} f(\hat{x^k}) + (d^k)^T \hat{\xi^k} = f(\bar{x})$, and $f(\hat{x^k}) + (d^k)^T \hat{\xi^k} < f(x^*)$ for all k. Obviously this cannot happen since the limit of a sequence cannot be strictly greater than a number if all the elements of the sequence are strictly less than it. Therefore S must be bounded and the lemma is proved. □

The proof of the claim now follows:

Theorem 3. *There is a sequence of solutions (x^i, ξ^i) which converges to $(x^*, 0)$ as $d^i \to \infty$.*

Proof. We simply apply Lemma 10, and the result is obvious. In fact there are uncountably many such sequences. □

6. Conclusion.

In this paper, we have proved convergence of an interior-point method under the mild assumptions of twice continuous differentiability and the boundedness of the primal iterates. By providing simple bounds on the variables, we can further relax these assumptions. Therefore, incorporating the ℓ_1 penalty approach allowed us to strengthen the convergence results of [6].

Note that we started by attempting to solve a general problem given by (1). As mentioned, this problem had drawbacks, including the use of equality constraints. Again, the use of the ℓ_1 penalty approach allowed us to incorporate all of these constraints into the same framework while allowing for a strictly feasible interior for the feasible region and bounded Lagrange multipliers. The convergence results provided apply not only to NLP but to mathematical programs with equilibrium constraints(MPECs), a wider class of problems whose optimal Lagrange multipliers are always unbounded.

References

[1] Mihai Anitescu. Nonlinear programs with unbounded lagrange multiplier sets. Technical Report ANL/MCS-P793-0200, Argonne National Labs.

[2] H. Y. Benson, A. Sen, D.F. Shanno, and R. J. Vanderbei. Interior point algorithms, penalty methods and equilibrium problems. *Computational Optimization and Applications*, 2005. (to appear).

[3] H.Y. Benson and D.F. Shanno. An exact primal-dual penalty method approach to warmstarting interior-point methods for linear programming. *Computational Optimization and Applications*, (to appear), 2005.

[4] H.Y. Benson and D.F. Shanno. Interior-point methods for nonconvex nonlinear programming: Regularization and warmstarts. *Computational Optimization and Applications*, (to appear), 2005.

[5] F. Bonnans and A. Ioffe. Second-order sufficiency and quadratic growth for non-isolated minima. *Mathematics of Operations Research*, 20(4):801–817, 1995.

[6] L. Chen and D. Goldfarb. Interior-point ℓ_2 penalty methods for nonlinear programming with strong global convergence properties. *Mathematical Programming*, 108:1–36, 2006.

[7] L. Chen and D. Goldfarb. On the fast local convergence of interior-point ℓ_2 penalty methods for nonlinear programming. Technical report, Working Paper, July 2006.

[8] R. Fletcher. *Practical Methods of Optimization*. J. Wiley and Sons, Chichester, England, 1987.

[9] X. Liu and J. Sun. Global convergence analysis of line search interior-point methods for nonlinear programming without regularity assumptions. *Journal of Optimization Theory and Applications*, page (to appear), 2006.

[10] D.F. Shanno and R.J. Vanderbei. Interior-point methods for nonconvex nonlinear programming: Orderings and higher-order methods. *Math. Prog.*, 87(2):303–316, 2000.

[11] R.J. Vanderbei and D.F. Shanno. An interior-point algorithm for nonconvex nonlinear programming. *Computational Optimization and Applications*, 13:231–252, 1999.

[12] S.J. Wright and D. Orban. Properties of the log-barrier function on degenerate nonlinear programs. *Mathematics of Operations Research*, 27(3):585–613, 2002.

Hande Y. Benson, Drexel University, Philadelphia, PA

Arun Sen, Nera Corporation, New York, NY

David F. Shanno, RUTCOR, Rutgers University, New Brunswick, NJ