Infeasibility Detection and SQP Methods for Nonlinear Optimization

Richard H. Byrd* Frank E. Curtis † Jorge Nocedal ‡

October 16, 2008

Abstract

This paper addresses the need for nonlinear programming algorithms that provide fast local convergence guarantees no matter if a problem is feasible or infeasible. We present an active-set sequential quadratic programming method derived from an exact penalty approach that adjusts the penalty parameter appropriately to emphasize optimality over feasibility, or vice versa. Conditions are presented under which superlinear convergence is achieved in the infeasible case. Numerical experiments illustrate the practical behavior of the method.

1 Introduction

Constrained optimization algorithms are confronted with two tasks: the minimization of a function and the satisfaction of constraints. In cases when feasible points exist and one or more are optimal, a number of rapidly convergent methods can be employed. However, when many of these methods are applied to infeasible problems, the progress of the iteration can be very slow and a great number of function evaluations is often required before a declaration of infeasibility can be made.

The primary focus of this paper is to address the need for optimization algorithms that can both efficiently solve feasible problems and rapidly detect when a given problem instance is infeasible. Fast detection of infeasibility has become increasingly important due to the central role it plays in branch and bound methods for mixed integer nonlinear programming

*Department of Computer Science, University of Colorado, Boulder, CO, USA. This author was supported by National Science Foundation grant CMMI 0728190.

†Courant Institute of Mathematical Sciences, New York University, New York, NY, USA. This author was supported by National Science Foundation grant DMS 0602235.

‡Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, IL, USA. This author was supported by National Science Foundation grant CCF-0514772 and by Department of Energy grant DE-FG02-87ER25047-A004.

and in parametric studies of optimization models, but is also a concern in its own right for general nonlinear programming problems that may include constraint incompatibilities.

One way to contend with the possibility that a problem could be infeasible, is to employ a switch in an algorithm to decide whether the current iteration should seek a solution of the nonlinear program or, in constrast, to solely minimize some measure of infeasibility. Such an approach has been advocated by Fletcher and Leyffer [17] and has the benefit that infeasibility can be declared when a minimizer of the feasibility measure violates one or more constraints. The main difficulty faced by this type of approach lies in the design of effective criteria for determining when such a switch should be made, since an inappropriate technique can lead to inefficiencies. In particular, since the objective function is ignored during iterations that care only about minimizing a measure of infeasibility, the iterates may stray from an optimal solution, thus delaying the optimization process.

In this paper we study an alternative approach involving a single optimization strategy, and show that it is effective for finding an optimal feasible solution (when one exists) or finding the minimizer of a feasibility measure (when no feasible point exists). Our algorithm is an active-set exact penalty method that uses the penalty parameter to emphasize optimization over infeasibility detection, or vice versa. It belongs to the class of penalty sequential quadratic programming (SQP) methods proposed by Fletcher [16] that compute steps by minimizing a piecewise quadratic model of a penalty function subject to a linearization of the constraints. An important feature of this type of active-set approach is that with an accurate estimate of the set of constraints satisfied as equalities at a solution point, the asymptotic convergence rate of the iteration is primarily influenced by the choice of penalty parameter. Thus, in this paper we pay careful attention to the procedure for updating the penalty parameter at every iteration, particularly for infeasible problem instances.

The problem of interest is formulated as

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & f(x) \\ \text{s.t.} \quad & g_i(x) \geq 0, \quad i \in \mathcal{I} = \{1,\ldots,t\}, \end{aligned} \tag{1.1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ and the $g_i : \mathbb{R}^n \to \mathbb{R}$ are smooth functions. Since the results and algorithms presented in this paper can be extended without difficulty to the case when the nonlinear program contains both equality and inequality constraints, we restrict our attention here to problem (1.1), for simplicity. When feasible points of (1.1) do not exist, the algorithm should return a solution of the problem

$$\min_x \; v(x) \triangleq \sum_{i\in\mathcal{I}} \max\{-g_i(x), 0\}. \tag{1.2}$$

More specifically, we would like to design the optimization algorithm so that, when the problem is infeasible, the iterates converge quickly to an *infeasible stationary point* $\hat{x}$, which is defined as a stationary point of problem (1.2) such that $v(\hat{x}) > 0$. We say that problem (1.1) is *locally infeasible* if there is an infeasible stationary point for it.

We note at the outset that infeasibility detection is difficult in the non-convex case. In such cases it has all the difficulties inherent in global optimization since, even if an algorithm identifies an infeasible point where constraint violations are locally minimized, there may exist feasible points in other regions of the search space. Nevertheless, the techniques discussed in this paper can be used in conjunction with global optimization methods [32, 23] to determine if a problem is in fact globally infeasible.

The paper is organized in five sections. In Section 2 we motivate and present a general form of our penalty-SQP method. In Section 3 we show that this type of approach yields fast convergence guarantees to certain classes of infeasible stationary points when the penalty parameter is handled appropriately. In Section 4 we describe two practical techniques for updating the penalty parameter and illustrate some numerical tests in Section 5. Throughout the paper $\|\cdot\|$ denotes any vector norm.

2 A Penalty-SQP Framework

The algorithm we propose is a penalty-SQP method, also known as an Sℓ_1QP method [15, 16] or an ellastic SQP method [18, 4]. It is a technique for the minimization of the exact penalty function

$$\phi(x;\rho) = \rho f(x) + v(x), \tag{2.1}$$

where $\rho > 0$ is a penalty parameter updated dynamically within the approach and v is the infeasibility measure defined in (1.2). (The penalty function is often written in the literature as $f(x) + \pi v(x)$, for some parameter π, but the equivalent formulation (2.1) is more convenient for our analysis.) If the penalty parameter ρ is sufficiently small, stationary points of the nonlinear program (1.1) are also stationary points of the penalty function ϕ; see for example [22].

Given a value for ρ_k and an iterate x_k, we define an appropriate step d_k as a solution to the subproblem

$$\min_{d\in\mathbb{R}^n} \quad q_k(d;\rho_k) \tag{2.2}$$

where

$$q_k(d;\rho_k) = \rho_k \nabla f(x_k)^T d + \tfrac{1}{2} d^T W(x_k,\lambda_k;\rho_k) d + \textstyle\sum_{i\in\mathcal{I}} \max\{-g_i(x_k) - \nabla g_i(x_k)^T d, 0\} \tag{2.3}$$

is a local model of the penalty function $\phi(\cdot;\rho)$ about x_k. Here, $W(x_k,\lambda_k;\rho_k)$ is the Hessian matrix

$$W(x_k,\lambda_k;\rho_k) = \rho_k \nabla^2 f(x_k) - \sum_{i\in\mathcal{I}} \lambda_k^i \nabla^2 g_i(x_k). \tag{2.4}$$

As we discuss in Section 4, this Hessian differs from that used in standard penalty methods (see e.g. [28, eq.(18.49)]) in that the penalty parameter multiplies only the Hessian of the objective and not the term involving the Hessian of the constraints. The problem (2.2) is nonsmooth, but as is well known, it has the smooth reformulation

$$\min_{d\in\mathbb{R}^n, s\in\mathbb{R}^t} \quad \rho_k \nabla f(x_k)^T d + \tfrac{1}{2} d^T W(x_k,\lambda_k;\rho_k) d + \textstyle\sum_{i\in\mathcal{I}} s_i \tag{2.5a}$$

$$\text{s.t.} \quad g_i(x_k) + \nabla g_i(x_k)^T d + s_i \geq 0, \;\; i\in\mathcal{I}, \tag{2.5b}$$

$$s_i \geq 0, \quad i\in\mathcal{I}, \tag{2.5c}$$

where s_i are slack variables. This subproblem is the focal point of our approach as a way to transition smoothly between seeking optimality and feasibility with the evolution of the value for ρ.

With a solution d_k to problem (2.5), the iterate is updated as

$$x_{k+1} = x_k + \alpha_k d_k,$$

where α_k is a steplength parameter that ensures sufficient reduction in $\phi(\cdot; \rho_k)$, and the new dual variables are given by

$$\lambda_{k+1} \leftarrow \text{optimal multipliers associated with the constraints (2.5b).} \tag{2.6}$$

A general form of our approach can now be presented as follows. For concreteness, we describe a line search algorithm with a merit function given by ϕ. (In Section 3, we assume in our local analysis that the steplength is always $\alpha_k = 1$, so our results apply equally to trust region and line search approaches.)

Algorithm I: Penalty-SQP Method

Initialize: (x_0, λ_0), $\tau \in (0, 1)$, and $\eta \in (0, 1)$

For $k = 0, 1, 2, ...$, or until x_k solves either (1.1) or (1.2):

1. Determine an appropriate value ρ_k for the penalty parameter.
2. Compute (d_k, λ_{k+1}) by solving the subproblem (2.5) with $\rho = \rho_k$.
3. Let $0 < \alpha_k \leq 1$ be the first member of the sequence $\{1, \tau, \tau^2, \ldots\}$ such that
$$\phi(x_k; \rho_k) - \phi(x_k + \alpha_k d_k; \rho_k) \geq \eta \alpha_k [q_k(0; \rho_k) - q_k(d_k; \rho_k)]. \tag{2.7}$$
4. Let $x_{k+1} \leftarrow x_k + \alpha_k d_k$.

Our remaining task in the presentation of our approach is to design a mechanism for setting the penalty parameter ρ during each iteration. We confront this issue in Section 4 after considering the local behavior of Algorithm I in Section 2.

Let us take a moment to discuss why standard (non-penalty) SQP methods, or any other methods that impose exact linearizations of the constraints, may not be conducive for handling infeasible problems. One can show that, at an infeasible stationary point $\hat{x}$, the constraints $g(\hat{x}) + \nabla g(\hat{x})^T d \geq 0$ are inconsistent, and in a neighborhood of $\hat{x}$ these constraints are either inconsistent or ill conditioned. Thus if the SQP iterates $\{x_k\}$ approach $\hat{x}$, the steps $\{d_k\}$ may not be defined, and even if they are, they become arbitrarily large in norm and the steplengths α_k tend to zero. The rate of convergence to $\hat{x}$ becomes linear at best, and for many problems a large number of iterations is required before the problem can be declared locally infeasible. In Section 5, we provide illustration of this behavior.

In contrast, the constraints (2.5b)-(2.5c) are always feasible, which was one of the main motivations for the Sℓ_1QP approach proposed by Fletcher in the 1980s. The regularization benefits of exact penalty methods have been recognized for a long time; see for example [34, 29, 12, 16, 14, 20, 11] and the literature on optimization problems with complementarity constraints [26, 33, 1, 24, 25]. For example, the SNOPT software package [18] reverts to an ℓ_1 penalty approach when the Lagrange multipliers are deemed too large or when the quadratic subproblem is inconsistent. However, in contrast to the algorithm proposed in this paper, the algorithm in SNOPT employs a switch as previously described—it starts as a regular SQP method and only invokes a penalty approach if difficulties are encountered. In addition, SNOPT makes no attempt to achieve a fast rate of convergence to stationary points in the infeasible case.

3 Fast Convergence to Infeasible Stationary Points

In this section we illustrate the local convergence properties of Algorithm I under certain common conditions. Since such properties for penalty-SQP algorithms applied to feasible problems have been studied elsewhere [16], we focus our analysis here on the infeasible case. This analysis reveals some properties that the penalty parameter update and the model Hessian must possess in order to achieve superlinear convergence of the algorithm.

Our interest in this section is in asymptotic rate of convergence results. Therefore, it is convenient to assume that the steplength is $\alpha_k = 1$. In doing so, we assume that a mechanism, such as a watchdog technique [10] or a second-order correction [16], is employed to ensure that unit steplengths are accepted by the merit function ϕ.

The problem of minimizing $\phi(\cdot, \rho)$ can be written as

$$\begin{aligned} \min_{x,r} \ & \rho f(x) + \sum_{i \in \mathcal{I}} r_i \\ \text{s.t. } & g_i(x) + r_i \geq 0, \quad r_i \geq 0, \quad i \in \mathcal{I}, \end{aligned} \tag{3.1}$$

where r_i are slack variables. If x^ρ is defined as a first-order optimal solution of problem (3.1) for a given ρ, then there exist slack variables r^ρ and Lagrange multipliers $\lambda^\rho, \sigma^\rho$ such that $(x^\rho, \lambda^\rho, r^\rho, \sigma^\rho)$ satisfies the KKT system

$$\rho \nabla f(x) - \sum_{i \in \mathcal{I}} \lambda_i \nabla g_i(x) = 0, \tag{3.2a}$$

$$1 - \lambda_i - \sigma_i = 0, \quad i \in \mathcal{I}, \tag{3.2b}$$

$$\lambda_i (g_i(x) + r_i) = 0, \quad i \in \mathcal{I}, \tag{3.2c}$$

$$\sigma_i r_i = 0, \quad i \in \mathcal{I}, \tag{3.2d}$$

$$g_i(x) + r_i \geq 0, \quad i \in \mathcal{I}, \tag{3.2e}$$

$$r, \lambda, \sigma \geq 0. \tag{3.2f}$$

We note, in particular, that if for $\rho > 0$ such a solution has $r^\rho = 0$, then x^ρ is a first-order optimal solution of the nonlinear program (1.1).

We provide an alternative way of characterizing solutions of the penalty problem (3.1) in the following result.

Lemma 3.1 *Suppose that* $(x^\rho, \lambda^\rho, r^\rho, \sigma^\rho)$ *is a primal-dual KKT point for problem* (3.1) *and that the strict complementarity conditions*

$$r_i^\rho + \sigma_i^\rho > 0, \quad \lambda_i^\rho + (g_i(x^\rho) + r_i^\rho) > 0 \tag{3.3}$$

hold for all $i \in \mathcal{I}$. *Then* (x^ρ, λ^ρ) *satisfies the system*

$$\rho\nabla f(x) - \sum_{i\in\mathcal{I}} \lambda_i \nabla g_i(x) = 0 \tag{3.4a}$$

$$\text{and either } g_i(x) < 0 \text{ and } \lambda_i = 1, \text{ or} \tag{3.4b}$$

$$g_i(x) > 0 \text{ and } \lambda_i = 0, \text{ or} \tag{3.4c}$$

$$g_i(x) = 0 \text{ and } \lambda_i \in (0,1). \tag{3.4d}$$

Conversely, if (x, λ) *satisfies* (3.4), *it also satisfies* (3.2) *together with* $r_i = \max(0, -g_i(x))$ *and* $\sigma_i = 1 - \lambda_i$.

Proof. If (x, λ, r, σ) is a KKT point for problem (3.1), it satisfies the KKT conditions (3.2). If $r_i > 0$ for some i, then (3.2d) implies $\sigma_i = 0$, which in turn implies $\lambda_i = 1$ by (3.2b), and then $g_i(x) = -r_i < 0$ by (3.2c). On the other hand, if $r_i = 0$ and $g_i(x) > 0$ for some i, we have from (3.2*c*) that $\lambda_i = 0$. Finally, if $r_i = 0$ and $g_i(x) = 0$, then by the strict complementarity assumption (3.3) we have that $\lambda_i > 0$ and $\sigma_i > 0$, and hence $\lambda_i \in (0,1)$. Combining all these observations leads to (3.4).

The second part of the lemma follows directly by substituting the proposed solution into (3.2). □

Throughout this section, it will be useful to distinguish between three distinct sets of constraint indices defined with respect to a given point x. Specifically, at x we define the sets of *active*, *violated*, and *strictly satisfied* constraints, respectively, as:

$$\mathcal{A}(x) = \{i \in \mathcal{I} : g_i(x) = 0\}; \quad \mathcal{V}(x) = \{i \in \mathcal{I} : g_i(x) < 0\}; \quad \mathcal{S}(x) = \{i \in \mathcal{I} : g_i(x) > 0\}. \tag{3.5}$$

Let us now characterize stationary points for the feasibility measure v defined in (1.2). Note that problem (1.2) can be recast as (3.1) for $\rho = 0$. Let us denote a stationary point of problem (3.1), for $\rho = 0$, as $\hat{x}$, i.e.,

$$\hat{x} \triangleq x^{\rho=0}.$$

Such a point satisfies conditions (3.2) (with $\rho = 0$), for some vectors $\hat{\lambda}, \hat{r}, \hat{\sigma}$. If we have that $\hat{r} \neq 0$, then as argued in the proof of Lemma 3.1, $g_i(\hat{x}) < 0$ for some i, and therefore $\hat{x}$ is an infeasible stationary point of the nonlinear program (1.1).

Another convenient way of describing an infeasible stationary point $\hat{x}$ of v is to note that it is a first-order optimal solution of the auxiliary problem

$$\begin{aligned} \min_x \; & \sum_{i\in\hat{\mathcal{V}}} -g_i(x) \\ \text{s.t. } & g_i(x) = 0, \quad i \in \hat{\mathcal{A}}, \end{aligned} \tag{3.6}$$

where from now on $\hat{\mathcal{A}}$ is shortand for $\mathcal{A}(\hat{x})$, and similarly for $\hat{\mathcal{V}}$ and $\hat{\mathcal{S}}$. To verify this claim, we note by Lemma 3.1 that since the conditions (3.2) can be recast as (3.4), then if $(\hat{x}, \hat{\lambda})$ solves (3.2) for $\rho = 0$ we have that $\hat{\lambda}_i \in (0, 1)$ for $i \in \hat{\mathcal{A}}$, $\hat{\lambda}_i = 1$ for $i \in \hat{\mathcal{V}}$, and $\hat{\lambda}_i = 0$ for $i \in \hat{\mathcal{S}}$. As a result, the pair $(\hat{x}, \hat{\lambda})$ satisfies the system

$$F(x, \lambda_{\hat{\mathcal{A}}}, \rho) \triangleq \begin{bmatrix} \rho \nabla f(x) - \sum_{i \in \hat{\mathcal{A}}} \lambda_i \nabla g_i(x) - \sum_{i \in \hat{\mathcal{V}}} \nabla g_i(x) \\ g_{\hat{\mathcal{A}}}(x) \end{bmatrix} = 0, \tag{3.7}$$

for $\rho = 0$, where we have defined the subvectors

$$\lambda_{\hat{\mathcal{A}}} = [\lambda_i]_{i \in \hat{\mathcal{A}}} \quad \text{and} \quad g_{\hat{\mathcal{A}}} = [g_i]_{i \in \hat{\mathcal{A}}}.$$

Since (3.7) (with $\rho = 0$) are the KKT conditions of problem (3.6), we have verified that $\hat{x}$ is a first-order optimal point for (3.6).

There are various types of infeasible stationary points. Some are strict isolated local minimizers of the infeasibility measure $v(x)$, while others may belong to a set of minimizers of v or may simply be stationary points of v. Formulation (3.6) is useful in identifying those stationary points for which we are able to establish a fast rate of convergence. In our analysis, we require that $\hat{x}$ is a point satisfying second-order sufficiency for problem (3.6). The complete set of assumptions is as follows.

Assumptions 3.2 *The point $\hat{x}$ is an infeasible stationary point of the feasibility problem* (1.2)*, $\hat{\lambda}$ is a vector of Lagrange multipliers such that $(\hat{x}, \hat{\lambda})$ solves* (3.4)*, and the following conditions hold:*

(a) (Smoothness) The functions f and g are twice continuously differentiable in an open convex set containing $\hat{x}$.

(b) (Regularity) The Jacobian of active constraints $\nabla g_{\hat{\mathcal{A}}}(\hat{x})^T$ has full row rank, where

$$\nabla g_{\hat{\mathcal{A}}}(x)^T = [\nabla g_i(x)]_{i \in \hat{\mathcal{A}}}. \tag{3.8}$$

(c) (Strict Complementarity) The multipliers satisfy $\hat{\lambda}_i \in (0, 1)$ for $i \in \hat{\mathcal{A}}$.

(d) (Second-Order Sufficiency) For $\rho = 0$, the Hessian matrix defined in (2.4) *satisfies*

$$d^T W(\hat{x}, \hat{\lambda}; 0) d = -d^T \left[\sum_{i \in \hat{\mathcal{A}}} \hat{\lambda}_i \nabla^2 g_i(\hat{x}) + \sum_{i \in \hat{\mathcal{V}}} \nabla^2 g_i(\hat{x}) \right] d > 0,$$

for all $d \neq 0$ such that $\nabla g_{\hat{\mathcal{A}}}(\hat{x})^T d = 0$.

Our approach for establishing superlinear convergence to an infeasible stationary point $\hat{x}$ is as follows. Let z_k denote the current primal-dual pair (x_k, λ_k) of Algorithm I and $z_{k+1} = (x_{k+1}, \lambda_{k+1})$ the next iterate; recall that z^ρ is a first-order optimal solution of the penalty problem (3.1) for a given value of the penalty parameter ρ. In a series of lemmas we show that, if z_k is close to $\hat{z}$ and ρ is small,

$$\begin{aligned}\|z_{k+1} - \hat{z}\| &\le \|z_{k+1} - z^\rho\| + \|z^\rho - \hat{z}\| \\ &\le C_1\|z_k - z^\rho\|^2 + O(\rho) \\ &\le C_2\|z_k - \hat{z}\|^2 + O(\rho),\end{aligned} \tag{3.9}$$

for some constants $C_1, C_2 > 0$. Thus, to achieve superlinear convergence we only need to ensure that, as the iterates approach a stationary point $\hat{x}$, Algorithm I decreases ρ fast enough.

Our first result quantifies the distance between x^ρ and $\hat{x}$, as a function of ρ.

Lemma 3.3 *Suppose that Assumptions 3.2 are satisfied. Then, for all ρ sufficiently small the penalty problem* (3.1) *has a solution x^ρ with the same sets of active, violated, and strictly satisfied constraints as $\hat{x}$; i.e., $\mathcal{A}(x^\rho) = \hat{\mathcal{A}}$, $\mathcal{V}(x^\rho) = \hat{\mathcal{V}}$, and $\mathcal{S}(x^\rho) = \hat{\mathcal{S}}$. Moreover, for all ρ sufficiently small,*

$$\lambda_i^\rho \in (0,1),\ i \in \hat{\mathcal{A}}, \quad \lambda_i^\rho = 1,\ i \in \hat{\mathcal{V}}, \quad \lambda_i^\rho = 0,\ i \in \hat{\mathcal{S}}, \tag{3.10}$$

and

$$\left\| \begin{bmatrix} x^\rho - \hat{x} \\ \lambda^\rho - \hat{\lambda} \end{bmatrix} \right\| \le C\rho \tag{3.11}$$

for some constant $C > 0$ independent of ρ.

Proof. We have shown above that if $(\hat{x}, \hat{\lambda})$ is a KKT point for problem (3.1) (with $\rho = 0$), it satisfies the system (3.7) (with $\rho = 0$). Under Assumptions 3.2, F is continuously differentiable mapping about the point $(\hat{x}, \hat{\lambda}_{\hat{\mathcal{A}}}, 0)$ and satisfies $F(\hat{x}, \hat{\lambda}_{\hat{\mathcal{A}}}, 0) = 0$. Differentiating F we find

$$F'(x, \lambda_{\hat{\mathcal{A}}}, \rho) \triangleq \frac{\partial F(x, \lambda_{\hat{\mathcal{A}}}, \rho)}{\partial(x, \lambda_{\hat{\mathcal{A}}})} = \begin{bmatrix} G(x, \lambda_{\hat{\mathcal{A}}}, \rho) & -\nabla g_{\hat{\mathcal{A}}}(x) \\ \nabla g_{\hat{\mathcal{A}}}(x)^T & 0 \end{bmatrix}, \tag{3.12}$$

where

$$G(x, \lambda_{\hat{\mathcal{A}}}, \rho) = \rho \nabla^2 f(x) - \sum_{i \in \hat{\mathcal{A}}} \lambda_i \nabla^2 g_i(x) - \sum_{i \in \hat{\mathcal{V}}} \nabla^2 g_i(x). \tag{3.13}$$

By Assumptions 3.2 (b) and (d), the matrix (3.12) is nonsingular at $(\hat{x}, \hat{\lambda}_{\hat{\mathcal{A}}}, 0)$. We can thus apply the implicit function theorem and state that there exist open sets $\mathcal{N}_x \in \mathbb{R}^n$, $\mathcal{N}_\lambda \in \mathbb{R}^{|\hat{\mathcal{A}}|}$ and $\mathcal{N}_\rho \in \mathbb{R}$ containing $\hat{x}, \hat{\lambda}_{\hat{\mathcal{A}}}$ and 0, respectively, and continuously differentiable functions $\bar{x} : \mathcal{N}_\rho \to \mathcal{N}_x$ and $\bar{\lambda}_{\hat{\mathcal{A}}} : \mathcal{N}_\rho \to \mathcal{N}_\lambda$ such that

$$\bar{x}(0) = \hat{x}, \quad \bar{\lambda}_{\hat{\mathcal{A}}}(0) = \hat{\lambda}_{\hat{\mathcal{A}}}, \quad \text{and} \quad F(\bar{x}(\rho), \bar{\lambda}_{\hat{\mathcal{A}}}(\rho), \rho) = 0, \quad \text{for all} \quad \rho \in \mathcal{N}_\rho. \tag{3.14}$$

By the second equation in (3.7), and since the inequalities in the definitions (3.5) of $\hat{\mathcal{V}}$ and $\hat{\mathcal{S}}$ are strict, we have that for ρ sufficiently small

$$g_i(\bar{x}(\rho)) = 0,\ i \in \hat{\mathcal{A}}, \quad g_i(\bar{x}(\rho)) > 0,\ i \in \hat{\mathcal{S}}, \quad g_i(\bar{x}(\rho)) < 0,\ i \in \hat{\mathcal{V}}. \tag{3.15}$$

Also, since $\hat{\lambda}_{\hat{\mathcal{A}}} \in (0,1)$ we have that $\bar{\lambda}_{\hat{\mathcal{A}}}(\rho) \in (0,1)$ for small ρ. If we define

$$\bar{\lambda}_i(\rho) = 1, \; i \in \hat{\mathcal{V}}, \quad \text{and} \quad \bar{\lambda}_i(\rho) = 0, \; i \in \hat{\mathcal{S}}, \tag{3.16}$$

it follows that $(\bar{x}(\rho), \bar{\lambda}(\rho))$ satisfies (3.4). Lemma 3.1 then implies that $(\bar{x}(\rho), \bar{\lambda}(\rho))$ satisfies (3.2) (together with $\bar{r}(\rho) = \max(0, -g(\bar{x}(\rho))$ and $\bar{\sigma}(\rho) = 1 - \bar{\lambda}(\rho))$ and is therefore a first-order optimal point for the penalty problem (3.1) for small ρ. Thus we can write $x^\rho = \bar{x}(\rho)$ for small ρ, and by (3.15), we have that x^ρ has the same sets of active, violated, and strictly satisfied constraints as $\hat{x}$. This proves the first part of the lemma.

We now establish the bound (3.11). Differentiating the equation $F(\bar{x}(\rho), \bar{\lambda}_{\hat{\mathcal{A}}}(\rho), \rho) = 0$ with respect to ρ, we have

$$\frac{\partial F}{\partial(\bar{x}, \bar{\lambda}_{\hat{\mathcal{A}}})} \frac{\partial(\bar{x}, \bar{\lambda}_{\hat{\mathcal{A}}})}{\partial \rho} + \frac{\partial F}{\partial \rho} = 0. \tag{3.17}$$

As the matrix (3.12) is nonsingular in a neighborhood of $(\bar{x}(0), \bar{\lambda}_{\hat{\mathcal{A}}}(0), 0)$, and since

$$\frac{\partial F}{\partial \rho} = \begin{bmatrix} \nabla f(x) \\ 0 \end{bmatrix},$$

we have from (3.17) that

$$\frac{\partial(\bar{x}, \bar{\lambda}_{\hat{\mathcal{A}}})}{\partial \rho} = - \begin{bmatrix} G(\bar{x}, \bar{\lambda}_{\hat{\mathcal{A}}}, \rho) & -\nabla g_{\hat{\mathcal{A}}}(\bar{x}) \\ \nabla g_{\hat{\mathcal{A}}}(\bar{x})^T & 0 \end{bmatrix}^{-1} \begin{bmatrix} \nabla f(\bar{x}) \\ 0 \end{bmatrix}. \tag{3.18}$$

Thus, defining

$$C = \max_{x \in \mathcal{N}_x, \lambda_{\hat{\mathcal{A}}} \in \mathcal{N}_\lambda, \rho \in \mathcal{N}_\rho} \left\| \begin{bmatrix} G(x, \lambda_{\hat{\mathcal{A}}}, \rho) & -\nabla g_{\hat{\mathcal{A}}}(x) \\ \nabla g_{\hat{\mathcal{A}}}(x)^T & 0 \end{bmatrix}^{-1} \begin{bmatrix} \nabla f(x) \\ 0 \end{bmatrix} \right\|,$$

we have by the Mean Value Theorem

$$\|(\bar{x}(\rho), \bar{\lambda}_{\hat{\mathcal{A}}}(\rho)) - (\bar{x}(0), \bar{\lambda}_{\hat{\mathcal{A}}}(0))\| \le C\rho,$$

and so by (3.14), (3.10) and (3.16)

$$\|(x^\rho, \lambda^\rho) - (\hat{x}, \hat{\lambda})\| \le C\rho.$$

This proves the second part of the lemma. □

We now describe an asymptotic property of the steps generated by the penalty-SQP method. Note that we do not assume that $W(x_k, \lambda_k; \rho_k)$ is positive definite; Assumption 3.2 (d) only states that $W(\hat{x}, \hat{\lambda}; 0)$ is positive definite on the tangent space of the active constraints. Therefore, the quadratic program (2.5) could have several solutions. In the following result, we show that at least one of these solutions has certain desirable properties. (This non-uniqueness issue is discussed further in Section 3.2.)

Lemma 3.4 *Suppose that Assumptions 3.2 hold. Then, for (x_k, λ_k) sufficiently close to $(\hat{x}, \hat{\lambda})$ and for all ρ sufficiently small, there is a local solution d_k of the SQP subproblem* (2.5) *that has the same sets of active, violated, and strictly satisfied constraints as $\hat{x}$, and can be obtained via a solution to the linear system*

$$\begin{bmatrix} W(x_k, \lambda_k; \rho_k) & -\nabla g_{\hat{\mathcal{A}}}(x_k) \\ \nabla g_{\hat{\mathcal{A}}}(x_k)^T & 0 \end{bmatrix} \begin{bmatrix} d \\ [\lambda_{k+1}]_{\hat{\mathcal{A}}} \end{bmatrix} = - \begin{bmatrix} \rho \nabla f(x_k) - \sum_{i \in \hat{\mathcal{V}}} \nabla g_i(x_k) \\ g_{\hat{\mathcal{A}}}(x_k) \end{bmatrix}. \tag{3.19}$$

Moreover the multipliers in (2.6) *satisfy*

$$[\lambda_{k+1}]_i \in (0,1), \ i \in \hat{\mathcal{A}}, \quad [\lambda_{k+1}]_i = 1, \ i \in \hat{\mathcal{V}}, \quad [\lambda_{k+1}]_i = 0, \ i \in \hat{\mathcal{S}}. \tag{3.20}$$

Proof. The first-order optimality conditions for the SQP subproblem (2.5), at some point (x, λ) and some value ρ, are

$$\begin{aligned} W(x, \lambda; \rho) d - \sum_{i \in \mathcal{I}} \gamma_i \nabla g_i(x) &= -\rho \nabla f(x), \\ 1 - \gamma_i - \omega_i &= 0, \quad i \in \mathcal{I}, \\ \gamma_i (g_i(x) + \nabla g_i(x)^T d + s_i) &= 0, \quad i \in \mathcal{I}, \\ \omega_i s_i &= 0, \quad i \in \mathcal{I}, \\ g_i(x) + \nabla g_i(x)^T d + s_i &\geq 0, \quad i \in \mathcal{I}, \\ s, \gamma, \omega &\geq 0, \end{aligned} \tag{3.21}$$

where (γ, ω) are Lagrange multipliers. Note that if we set $d = 0$, this system is identical to (3.2), and since $(\hat{x}, \hat{\lambda})$ satisfies (3.2) for $\rho = 0$, it follows that the system (3.21) is solved at $(x, \lambda) = (\hat{x}, \hat{\lambda})$ and for $\rho = 0$ by $(d, \hat{\gamma}) = (0, \hat{\lambda})$ and $(\hat{\omega}, \hat{s}) = (\hat{\sigma}, \hat{r})$. By Lemma 3.1, we have that

$$\hat{\lambda}_i \in (0,1), \ i \in \hat{\mathcal{A}}, \quad \hat{\lambda}_i = 1, \ i \in \hat{\mathcal{V}}, \quad \hat{\lambda}_i = 0, \ i \in \hat{\mathcal{S}}, \tag{3.22}$$

and hence by (3.2a) (with $\rho = 0$)

$$\sum_{i \in \hat{\mathcal{A}}} \hat{\gamma}_i \nabla g_i(\hat{x}) + \sum_{i \in \hat{\mathcal{V}}} \nabla g_i(\hat{x}) = 0.$$

Therefore, for $\rho = 0$, the linear system

$$\begin{bmatrix} W(x, \lambda; \rho) & -\nabla g_{\hat{\mathcal{A}}}(x) \\ \nabla g_{\hat{\mathcal{A}}}(x)^T & 0 \end{bmatrix} \begin{bmatrix} d \\ \gamma_{\hat{\mathcal{A}}} \end{bmatrix} = - \begin{bmatrix} \rho \nabla f(x) - \sum_{i \in \hat{\mathcal{V}}} \nabla g_i(x) \\ g_{\hat{\mathcal{A}}}(x) \end{bmatrix}, \tag{3.23}$$

is satisfied at $(x, \lambda) = (\hat{x}, \hat{\lambda})$ by $(d, \gamma_{\hat{\mathcal{A}}}) = (0, \hat{\lambda}_{\hat{\mathcal{A}}})$. Moreover, under Assumptions 3.2, the matrix in (3.23) is nonsingular at $(\hat{x}, \hat{\lambda})$ for $\rho = 0$, and hence the solution of (3.23) varies continuously in a neighborhood of $(\hat{x}, \hat{\lambda}, 0)$. We also have that $W(x, \lambda; \rho)$ is positive definite on the null subspace of $\nabla g_{\hat{\mathcal{A}}}(x)^T$, in a neighborhood of $(\hat{x}, \hat{\lambda}, 0)$.

It follows from these observations that for all (x, λ) sufficiently close to $(\hat{x}, \hat{\lambda})$ and for ρ sufficiently small, the solution $(d, \gamma_{\hat{\mathcal{A}}})$ to (3.23) is close to $(0, \hat{\lambda}_{\hat{\mathcal{A}}})$ and therefore satisfies

$$\begin{aligned} \gamma_i \in (0,1) & \text{ for } i \in \hat{\mathcal{A}}, \\ g_i(x) + \nabla g_i(x)^T d < 0 & \text{ for } i \in \hat{\mathcal{V}}, \\ g_i(x) + \nabla g_i(x)^T d > 0 & \text{ for } i \in \hat{\mathcal{S}}. \end{aligned} \tag{3.24}$$

By construction, such a solution $(d, \gamma_{\hat{\mathcal{A}}})$ satisfies (3.23) and therefore satisfies (3.21) together with $\gamma_i = 1$, for $i \in \hat{\mathcal{V}}$, $\gamma_i = 0$, for $i \in \hat{\mathcal{S}}$, $w_i = 1 - \gamma_i$, and $s_i = \max(0, -g_i(x) - \nabla g_i(x)^T d)$. Therefore, (d, γ) is a KKT point of the SQP subproblem (2.5) (in fact, by Assumption 3.2 (d) it is a minimizer of that problem) and by (3.24) it has the same sets of active, violated, and strictly satisfied constraints as $\hat{x}$. □

Lemma 3.4 shows that, near a stationary point, the SQP step d_k is given by the system (3.19), and by conditions (3.20) we have that for all large k the multiplier estimates generated by Algorithm I satisfy

$$[\lambda_k]_i \in (0,1),\ i \in \hat{\mathcal{A}}, \quad [\lambda_k]_i = 1,\ i \in \hat{\mathcal{V}}, \quad [\lambda_k]_i = 0,\ i \in \hat{\mathcal{S}}. \tag{3.25}$$

Based on these observations, we study the effect of one step of the algorithm on the penalty problem (3.1), for small ρ. We show that the quadratic convergence of the pure Newton iteration $x_{k+1} = x_k + d_k$ to the solution of each penalty problem is uniform with respect to ρ.

Lemma 3.5 *Let (x^ρ, λ^ρ) be a primal-dual pair satisfying the KKT conditions* (3.2) *for the penalty problem* (3.1)*. Then, for (x_k, λ_k) sufficiently close to $(\hat{x}, \hat{\lambda})$, with λ_k satisfying* (3.25)*, and for ρ sufficiently small, there is a constant C' independent of ρ, such that the iterates $x_{k+1} = x_k + d_k$ generated by the SQP algorithm satisfy*

$$\left\| \begin{bmatrix} x_{k+1} - x^\rho \\ \lambda_{k+1} - \lambda^\rho \end{bmatrix} \right\| \le C' \left\| \begin{bmatrix} x_k - x^\rho \\ \lambda_k - \lambda^\rho \end{bmatrix} \right\|^2. \tag{3.26}$$

Proof. By Lemma 3.4, if (x_k, λ_k) is sufficiently close to $(\hat{x}, \hat{\lambda})$ and ρ is sufficiently small, the step d_k generated by Algorithm I is obtained via a solution of the system (3.19). Moreover, since $W(x_k, \lambda_k; \rho_k)$ is given by (2.4), and since conditions (3.25) are satisfied, we have from (3.13) that $W(x_k, \lambda_k; \rho_k) = G(x_k, \lambda_{\hat{\mathcal{A}}}, \rho_k)$. Therefore system (3.19) constitutes the Newton iteration applied to the nonlinear system $F(x, \lambda_{\hat{\mathcal{A}}}, \rho)$, for fixed ρ, where F is defined in (3.7).

We can now apply standard Newton analysis (see for example [13]). By Assumption 3.2 (a) we have that F', given in (3.12), is continuously differentiable, and hence F' is Lipschitz continuous in a neighborhood of $(\hat{x}, \hat{\lambda})$, for each ρ. Moreover, since ρ is bounded, this Lipschitz constant, which we denote by κ_1, is independent of ρ. Next, by Assumptions 3.2 (b) and (d), the matrix F' is nonsingular at $(\hat{x}, \hat{\lambda}, 0)$ and hence its inverse exists and is bounded in norm by a constant κ_2 in a neighborhood of that point. By

Theorem 5.2.1 of [13], if (x_k, λ_k) satisfies

$$\left\| \begin{bmatrix} x_k - x^\rho \\ \lambda_k - \lambda^\rho \end{bmatrix} \right\| \le \frac{1}{2\kappa_1\kappa_2}, \tag{3.27}$$

we have

$$\left\| \begin{bmatrix} x_{k+1} - x^\rho \\ \lambda_{k+1} - \lambda^\rho \end{bmatrix} \right\| \le \kappa_1\kappa_2 \left\| \begin{bmatrix} x_k - x^\rho \\ \lambda_k - \lambda^\rho \end{bmatrix} \right\|^2. \tag{3.28}$$

(This inequality contains all components of x and λ, not just those in the active set $\hat{\mathcal{A}}$, because from (3.10) and (3.25) we have that $[x_k]_i = x_i^\rho$ and $[\lambda_k]_i = \lambda_i^\rho$, for $i \in \hat{\mathcal{V}} \cup \hat{\mathcal{S}}$.)

Finally, if ρ is sufficiently small that (x^ρ, λ^ρ) satisfies

$$\left\| \begin{bmatrix} x^\rho - \hat{x} \\ \lambda^\rho - \hat{\lambda} \end{bmatrix} \right\| \le \frac{1}{4\kappa_1\kappa_2},$$

and (x_k, λ_k) is chosen so that

$$\left\| \begin{bmatrix} x_k - \hat{x} \\ \lambda_k - \hat{\lambda} \end{bmatrix} \right\| \le \frac{1}{4\kappa_1\kappa_2},$$

then (3.27) is satisfied. □

We are now ready to prove the main result of this section, namely quadratic (or superlinear) convergence to an infeasible stationary point. We recall that $z = (x, \lambda)$ denotes a primal-dual pair.

Theorem 3.6 *Suppose that $\hat{x}$ is an infeasible stationary point for the nonlinear program* (1.1) *and that Assumptions 3.2 hold. Suppose also that the penalty parameter is selected so that $\rho_k = O(||z_k - \hat{z}||^2)$ for all large k. Then, if an iterate $z_k = (x_k, \lambda_k)$ is sufficiently close to $\hat{z} = (\hat{x}, \hat{\lambda})$, the sequence $\{z_k\}$ converges to $\hat{z}$ quadratically. If, instead, $\rho_k = o(||z_k - \hat{z}||)$, then the convergence rate is superlinear.*

Proof. Defining $z_{k+1} = (x_{k+1}, \lambda_{k+1})$ and $z^{\rho_k} = (x^{\rho_k}, \lambda^{\rho_k})$, we may apply Lemmas 3.3 and 3.5 to the inequality

$$\|z_{k+1} - \hat{z}\| \le \|z_{k+1} - z^{\rho_k}\| + \|z^{\rho_k} - \hat{z}\|$$

to obtain

$$||z_{k+1} - \hat{z}|| \le C' ||z_k - z^{\rho_k}||^2 + C\rho_k. \tag{3.29}$$

Then, we have that

$$\begin{aligned} ||z_k - z^{\rho_k}||^2 &\le (||z_k - \hat{z}|| + ||\hat{z} - z^{\rho_k}||)^2 \\ &\le (||z_k - \hat{z}|| + C\rho_k)^2 \\ &= ||z_k - \hat{z}||^2 + C^2\rho_k^2 + 2C||z_k - \hat{z}||\rho_k. \end{aligned} \tag{3.30}$$

Therefore, if $\rho = O(||z_k - \hat{z}||^2)$, we have from this relation and (3.29) that the iteration is locally and quadratically convergent, while if $\rho = o(||z_k - \hat{z}||)$, the rate of convergence is superlinear. □

In summary, by using the structured Hessian approximation (2.4), a penalty-SQP method will achieve a fast rate of convergence simply by driving the penalty parameter to zero fast enough in a neighborhood of an infeasible stationary point. Of course, a practical algorithm must also be efficient in the feasible case and a strategy for controling ρ in both cases is an essential part of a general purpose algorithm. In Section 4 we present one such strategy. Before doing so, we discuss two issues related to the convergence analysis just presented.

3.1 A Special Case: n Active Constraints

Let us consider how our analysis specializes to the case in which n constraints $g_i(x)$ are active at the infeasible stationary point $\hat{x}$, where n is the number of variables. That is, we consider situations when $|\hat{\mathcal{A}}| = n$ and the block equations $g_{\hat{\mathcal{A}}}(x) = 0$ in (3.7) constitute a system of n equations in n unknowns with $g_{\mathcal{A}}(\hat{x}) = 0$.

Let us observe how the proof of Lemma 3.3 specializes in this case. As in Assumption 3.2 (b), we assume that the active constraint gradients are linearly independent at $\hat{x}$, i.e., that $\nabla g_{\hat{\mathcal{A}}}(\hat{x})$ is nonsingular. It is easy to verify that in this case (3.18) implies

$$\begin{bmatrix} \partial \bar{x}/\partial \rho \\ \partial \bar{\lambda}_{\hat{\mathcal{A}}}/\partial \rho \end{bmatrix} = \begin{bmatrix} 0 \\ \nabla g_{\mathcal{A}}(\hat{x})^{-1} \nabla f(\hat{x}) \end{bmatrix}, \tag{3.31}$$

for $(\bar{x}, \bar{\lambda}_{\hat{\mathcal{A}}})$ near $(\hat{x}, \hat{\lambda}_{\hat{\mathcal{A}}})$. Let us consider values of ρ in the neighborhood $\mathcal{N}_\rho$ defined in (3.14). By (3.31) we see that as ρ increases from 0, we have that $x(\rho)$ stays at $\hat{x}$ and only the multipliers change; i.e., for ρ smaller than some finite value, the minimizer of the penalty function is always $\hat{x}$.

Therefore, if there are n active constraints at $\hat{x}$, and if these constraint gradients are linearly independent, there is a threshold value $\rho_{min} > 0$ such that for all $\rho < \rho_{min}$ the solution x^ρ of the penalty problem (3.1) satisfies $x^\rho = \hat{x}$. Thus, in this case, the algorithm can declare that the problem is locally infeasible without driving the penalty parameter ρ to zero.

3.2 Positive Definiteness of the Hessian

The analysis we have presented assumes that the exact Hessian of the Lagrangian $W_k \triangleq W(x_k, \lambda_k; \rho_k)$ is used to form the quadratic model (2.5). In general, however, the Hessian (2.4) will not be positive definite in $\mathbb{R}^n$. We can only expect that, in a neighborhood of an infeasible stationary point, the Hessian $W(x_k, \lambda_k; 0)$ is positive definite on the null space of $\nabla g_{\mathcal{A}}(\hat{x})^T$.

Thus, if W_k defined by (2.4) is indefinite, the algorithm requires the solution of an indefinite quadratic program (QP) that may have several local minimizers. The results presented above apply provided the QP solver chooses the minimizer satisfying Lemma 3.4. One way to achieve this near the stationary point $\hat{x}$ is to choose the QP minimizer that is closest to the current iterate x_k.

The situation becomes simpler if W_k is defined to be positive definite over all of $\mathbb{R}^n$, for in this case the quadratic program (2.5) has a unique solution. Our analysis then applies directly without giving further consideration to the solution obtained by the QP solver. Note that the Hessian (2.4) will be positive definite in $\mathbb{R}^n$ if the problem is convex. Otherwise, we could modify or redefine the Hessian to ensure this positive definiteness property, but this must be done carefully to avoid interfering with fast convergence.

Suppose that a modified positive definite approximate Hessian W_k is employed. Since our approach is equivalent to applying an SQP method to the equality constrained problem

$$\begin{aligned} \min\ & \rho f(x) + \sum_{i\in\hat{\mathcal{V}}} -g_i(x) \\ \text{s.t. } & g_i(x) = 0, \quad i \in \hat{\mathcal{A}}, \end{aligned} \tag{3.32}$$

whose Lagrangian Hessian is G (see (3.13)), we can apply the characterization result for superlinear convergence given in [3], [30]. It states that this iteration is superlinear if and only if the modified Hessian W_k satisfies the condition

$$\frac{\|P_k(W_k - G(\hat{x}, \hat{\lambda}_{\hat{\mathcal{A}}}, \rho))\|}{\|x_{k+1} - x_k\|} \to 0, \tag{3.33}$$

where P_k denotes the $n \times n$ orthogonal projection onto the null space of $\nabla g_{\mathcal{A}}^T(x)$. One way to achieve this limit is to define W_k as the augmented Lagrangian Hessian,

$$\rho_k \nabla^2 f(x_k) - \sum_{i\in\mathcal{I}} \lambda_k^i \nabla^2 g_i(x_k) + \sigma \nabla g_{\mathcal{A}}(x_k) \nabla g_{\mathcal{A}}(x_k)^T,$$

for a sufficiently large parameter $\sigma > 0$. In addition, quasi-Newton updating methods have been developed that generate positive definite approximations to the Hessian of the augmented Lagrangian that yield R-superlinear convergence [31] and [9].

Thus, although the discussion in this paper has focussed on the exact Hessian, a variety of other practical methods will converge rapidly to minimizers of infeasibiity that satisfy Assumptions 3.2.

4 Penalty Parameter Update

In the previous section, we have given conditions on the sequence of penalty parameters that ensure a fast rate of local convergence in the infeasible case. In order to illustrate the practical behavior of a method that satisfies these conditions, we now present a technique for updating the penalty parameter that promotes convergence from remote starting points and is designed with both the feasible and infeasible cases in mind. This technique is presented as Algorithm II below and can be applied in Step 1 of Algorithm I.

The update follows the spirit of the *steering rules* that are reviewed in [7] and are incorporated in [6] into an active-set line search SQP method. Algorithm II modifies these rules so that they conform with the requirements of the analysis presented in Section 3. For ease of exposition, we define the linear model of the constraints appearing in (2.3) as

$$m_k(d) = \sum_{i\in\mathcal{I}} \max\{-g_i(x_k) - \nabla g_i(x_k)^T d, 0\},$$

so that the quadratic model of the penalty function can be written as

$$q_k(d;\rho) = \rho\nabla f(x_k)^T d + \tfrac{1}{2} d^T W(x_k, \lambda_k; \rho) d + m_k(d). \tag{4.1}$$

In addition, we define a measure for the KKT error, with respect to (3.4), as

$$E_k(\rho) = \|\rho\nabla f(x_k) - \sum_{i\in\mathcal{I}} \lambda_i \nabla g_i(x_k)\|_1 + \sum_{i\in\mathcal{S}_k} |g_i(x_k)\lambda_k^i| + \sum_{i\in\mathcal{V}_k} |g_i(x_k)(1-\lambda_k^i)|. \tag{4.2}$$

Here, $\mathcal{S}_k$ and $\mathcal{V}_k$ are the sets of strictly satisfied and violated constraints evaluated at the iterate x_k, and the $\lambda_k^i \in [0,1]$ are Lagrange multiplier estimates.

Algorithm II: Penalty Parameter Update Strategy

Given $\rho_k = \rho_{k-1}$ and $0 < \epsilon_1, \epsilon_2 < 1$.

a. Solve (2.5) with $\rho = \rho_k$ to obtain d_k.

b. If $m_k(d_k) = 0$, then return ρ_k and d_k.

c. Solve (2.5) with $\rho = 0$ to obtain $\bar{d}_k$.

d. If $m_k(\bar{d}_k) = 0$
   decrease ρ_k until the solution d_k to (2.5) with $\rho = \rho_k$ satisfies $m_k(d_k) = 0$;
   else
   decrease ρ_k until d_k satisfies

$$m_k(0) - m_k(d_k) \geq \epsilon_1(m_k(0) - m_k(\bar{d}_k)). \tag{4.3}$$

e. Decrease ρ_k until d_k satisfies

$$q_k(0;\rho_k) - q_k(d_k;\rho_k) \geq \epsilon_2(q_k(0;0) - q_k(\bar{d}_k;0)). \tag{4.4}$$

f. Set $\rho_k = \min\{\rho_k, [E_k(0)/v_k]^2\}$, where the multipliers λ used in E_k are chosen as the optimal multipliers corresponding to the linearized constraints (2.5b) obtained during the computation of $\bar{d}_k$.

The essential effects of this strategy are the following. First, if the SQP subproblem for the most recent value of the penalty parameter yields a linearly feasible solution (i.e., $m(d_k) = 0$), then we follow this direction. This is to ensure that regularization of the constraints does not mar the progress of the algorithm when it is not needed. Second, if the SQP solution is linearly infeasible, then we decrease the penalty parameter sufficiently so that the new direction provides sufficient progress in linearized feasibility. This is to ensure that the algorithm does not diverge from minimizers of the infeasibility measure. The condition (4.4) ensures that as the algorithm progresses, the penalty parameter is small enough so that minimizers of $\phi(x;\rho)$ correspond to minimizers of the feasibility measure

v. Finally, by reducing ρ_k sufficiently in step f when the optimality error for the feasibility problem $E_k(0)$ is small, Theorem 3.6 states that the algorithm can converge superlinearly.

The steering rules described in [7] were designed to ensure global convergence (even in the infeasible case) and have proved to be effective in practice [5, 6], but were not designed to yield a fast rate of convergence in the infeasible case. The technique in Algorithm II, on the other hand, may be regarded as alternative steering rules that are designed to be efficient for both the feasible and infeasible problems. In particular, an important difference between these two approaches relates to the definition of the quadratic model of the penalty function. The model used in [7], which is commonly employed in penalty-SQP methods [16, 28], is given by

$$Q_k(d;\rho) = \rho\left[\nabla f(x_k)^T d + \tfrac{1}{2} d^T \nabla^2_{xx} L(x_k,\lambda_k) d\right] + m_k(d), \tag{4.5}$$

where

$$\nabla^2_{xx} L(x_k,\lambda_k) = \nabla^2 f(x_k) - \sum_{i\in\mathcal{I}} \lambda_k^i \nabla^2 g_i(x_k) \tag{4.6}$$

is the Hessian of the Lagrangian of the nonlinear problem (1.1). If we compare (2.4) with (4.6) we observe that in q_k the penalty parameter ρ multiplies only the Hessian of the objective, whereas in Q_k it multiplies all of the Hessian terms. Therefore, for $\rho = 0$, Q_k becomes the linear function m_k, whereas the model q_k remains quadratic and includes second order information about the constraints. We have seen in Section 3 that the curvature information of the constraints contained in q_k, which takes on a prominent role as ρ decreases, is crucial in providing a fast rate of convergence to infeasible stationary points. Thus, the strategy described in Algorithm II can be seen as extension of the steering rules that maintains the same Hessian matrix for the constraints even for varying values of ρ.

A possible drawback of Algorithm II is that the computation of the reference direction $\bar{d}_k$ requires the solution to a quadratic program, whereas the rules described in [7] call for the solution of a linear program, which may be less expensive. A numerical exploration of the tradeoffs of these two approaches is outside the scope of this paper as it requires a sophisticated software implementation of the SQP approach and exhaustive testing. For now, we simply present Algorithm II as one that yields good performance on the problems presented in the next section and fits into the framework of Algorithm I for ensuring fast local convergence.

We close this section with mention that other techniques have been proposed in the literature for updating the penalty parameter; see e.g. [19, 2, 27, 11]. In those techniques, the update is based on the behavior of the algorithm over several iterations, while the steering rules proposed here and in [7] explore the properties of the problem at a fixed iterate in order to update ρ.

5 Numerical Tests

We developed a prototype MATLAB implementation of Algorithm I-II in order to observe its performance on some illustrative examples. In the code, we ensure that W_k is positive definite, by adding a multiple of the identity matrix to the matrix (2.4), so that $q_k(d;\rho)$

has a unique minimizer for each $\rho \in [0, \rho_{k-1}]$. The quadratic program (2.5) is solved using MATLAB's `quadprog` routine, and the chosen input parameters are given as follows: $\tau = 0.5, \eta = 10^{-8}, \rho_0 = 1, \epsilon_1 = \epsilon_2 = 0.1$.

Algorithm I terminates when one of the following conditions hold:

$$E_k(\rho_k) \le 10^{-6} \quad \text{and} \quad v_k \le 10^{-6} \quad \text{(optimality)}$$

or

$$E_k(0) \le 10^{-6} \quad \text{and} \quad v_k > 10^{-6} \quad \text{(infeasibility)}.$$

As previously mentioned, during the computation of $E_k(0)$ the Lagrange multipliers are chosen as the optimal multipliers corresponding to the linearized constraints (2.5b) obtained during the computation of $\bar{d}_k$, and so similarly, during the computation of $E_k(\rho_k)$ the multipliers are chosen as the optimal multipliers for (2.5b) obtained during the computation of d_{k-1} in the previous iteration.

Example 1. We refer to the following problem as `unique`:

$$\begin{aligned} &\min\ x_1 + x_2 \\ &\text{s.t. } x_2 - x_1^2 - 1 \ge 0 \\ &\qquad 0.3(1 - e^{x_2}) \ge 0. \end{aligned} \tag{5.1}$$

This problem is infeasible and the point $\hat{x} = (0, 1)$ is a strict minimizer of the infeasibility measure $v(x)$ at which Assumptions 3.2 are satisfied. The second constraint simply imposes the bound $x_2 \le 0$, but we have been written it as a nonlinear expression; the weight 0.3 is chosen so that only the first constraint is active at $\hat{x}$. Running our algorithm on this problem, starting from the point $(3, 2)$, yields the results given in Table 1. We report the iteration number k, the function value f_k, the value of the infeasibility measure v_k, the KKT (optimality) error $E_k(\rho_{k-1})$, the KKT (feasibility) error $E_k(0)$, the penalty parameter ρ_k, the norm of the search direction $\|d_k\|$, and the steplength α_k given in (2.7).

Table 1: Output for Example 1 (`unique`)

k	f_k	v_k	$E_k(\rho_{k-1})$	$E_k(0)$	ρ_k	$\|\|d_k\|\|$	α_k
0	+5.00e+00	9.92e+00	8.84e+00	7.51e+00	1.00e+00	1.71e+00	1.00e+00
1	+2.66e+00	2.82e+00	3.51e+00	2.23e+00	1.00e-02	1.16e+00	1.00e+00
2	+1.04e+00	1.06e+00	9.14e-01	9.07e-01	1.00e-02	1.13e+00	5.00e-01
3	+3.20e-01	8.23e-01	4.55e-01	4.53e-01	1.00e-02	7.76e-01	1.00e+00
4	+6.74e-01	6.09e-01	4.33e-01	4.38e-01	1.00e-02	4.66e-01	1.00e+00
5	+9.72e-01	5.81e-01	2.84e-01	2.73e-01	1.00e-02	2.29e-01	1.00e+00
6	+9.39e-01	5.22e-01	4.52e-02	5.49e-02	1.00e-02	3.90e-02	1.00e+00
7	+9.94e-01	5.16e-01	1.66e-03	8.46e-03	2.69e-04	4.93e-03	1.00e+00
8	+1.00e+00	5.15e-01	9.17e-05	3.61e-04	4.91e-07	2.20e-04	1.00e+00
9	+1.00e+00	5.15e-01	1.18e-07	6.08e-07	--------	--------	--------

We mentioned at the end of Section 1 that algorithms that impose strict linearizations of the constraints in the step computation are likely to be inefficient in a neighborhood of infeasible stationary points. An example of such a method is the line search interior point method implemented in the KNITRO/DIRECT solver [8]. For problem `unique`, this method requires 42 iterations and 151 function evaluations to declare the problem locally infeasible. As can be seen from the large ratio of function evalutions to iterations, the step has to be shortened considerably, particularly as the iteration approaches the infeasible stationary point.

Example 2: Problem `robot` is from the CUTEr collection [21]. We ran the version with 14 variables. The original model has three equality constraints (call them c_1, c_2, and c_3), but we added the additional constraint

$$c_1(x) = c_1^2(x) + 1,$$

so that the resulting model is infeasible. The results from our implementation are given in Table 2.

Table 2: Output for Example 2 (`robot`)

k	f_k	v_k	$E_k(\rho_{k-1})$	$E_k(0)$	ρ_k	$\|\|d_k\|\|$	α_k
0	+0.00e+00	1.12e+01	1.00e+00	4.00e+00	1.00e+00	1.25e+00	1.00e+00
1	+1.56e+00	5.06e+00	1.70e+00	2.80e+00	1.00e+00	5.79e-01	1.00e+00
2	+2.12e+00	2.89e+00	3.34e-01	7.62e-01	1.00e-01	7.39e-01	1.00e+00
3	+4.19e+00	1.42e+00	5.13e-01	6.49e-01	1.00e-01	3.25e-01	1.00e+00
4	+4.75e+00	1.19e+00	2.95e-01	5.05e-01	1.00e-01	1.83e-01	1.00e+00
5	+5.20e+00	1.10e+00	1.30e-01	4.49e-01	1.00e-01	1.10e-01	1.00e+00
6	+5.45e+00	1.06e+00	5.96e-02	3.67e-01	1.00e-02	2.04e+00	6.25e-02
7	+5.50e+00	1.05e+00	2.37e-01	2.54e-01	1.00e-02	1.77e-01	1.00e+00
8	+6.01e+00	1.01e+00	1.20e-01	1.42e-01	1.00e-02	2.44e-01	5.00e-01
9	+6.09e+00	1.01e+00	5.65e-02	7.93e-02	1.00e-03	9.37e-02	1.00e+00
10	+6.19e+00	1.01e+00	9.57e-03	6.76e-03	4.57e-05	2.72e-02	1.00e+00
11	+6.15e+00	1.00e+00	2.79e-04	1.59e-04	2.52e-08	2.81e-03	1.00e+00
12	+6.15e+00	1.00e+00	7.53e-06	7.46e-06	5.56e-11	3.17e-05	1.00e+00
13	+6.15e+00	1.00e+00	3.96e-10	2.48e-10	--------	--------	--------

Example 3. We refer to the following problem as `isolated`:

$$\begin{aligned} \min\ & x_1 + x_2 \\ \text{s.t. } & -x_1^2 + x_2 - 1 \geq 0 \\ & -x_1^2 - x_2 - 1 \geq 0 \\ & x_1 - x_2^2 - 1 \geq 0 \\ & -x_1 - x_2^2 - 1 \geq 0. \end{aligned} \tag{5.2}$$

This problem is infeasible and the point $\hat{x} = (0, 0)$ is a strict minimizer of the infeasibility measure $v(x)$ where none of the constraints are active. Running our algorithm with the starting point $(3, 2)$ yields the results in Table 3.

Table 3: Output for Example 3 (`isolated`)

k	f_k	v_k	$E_k(\rho_{k-1})$	$E_k(0)$	ρ_k	$\|\|d_k\|\|$	α_k
0	+5.00e+00	3.00e+01	1.15e+01	1.15e+01	1.00e+00	2.34e+00	1.00e+00
1	+1.70e+00	7.63e+00	4.49e+00	4.15e+00	1.00e+00	2.22e+00	1.00e+00
2	-1.44e+00	7.15e+00	3.09e+00	4.62e+00	1.00e+00	1.05e+00	1.00e+00
3	-1.85e-01	4.07e+00	1.00e+00	7.38e-01	1.00e-01	2.54e-01	1.00e+00
4	+4.32e-02	4.01e+00	3.73e-01	2.73e-01	1.00e-02	7.42e-02	1.00e+00
5	-5.00e-03	4.00e+00	7.69e-16	1.00e-02	1.00e-04	3.50e-03	1.00e+00
6	-5.00e-05	4.00e+00	1.57e-15	1.00e-04	1.00e-08	3.54e-05	1.00e+00
7	-5.00e-09	4.00e+00	6.16e-16	1.00e-08	--------	--------	--------

Example 4. In this example, we simulate a situation that occurs in nonlinear branch-and-bound methods for mixed-integer nonlinear programming. Problem `batch`, from the CUTEr collection, is a problem with 46 variables that has a feasible and optimal solution. Running this problem with our code, we find that our penalty parameter updating strategy does not impede the progress of the algorithm when applied to a feasible problem. In fact, the method performs quite well, quadratic convergence is obtained, and the penalty parameter is kept constant in the last iterations; see Table 4. (The final value of ρ is small is because the optimal objective value is of f order 10^5.)

Table 4: Output for Example 4(a) (`batch`)

k	f_k	v_k	$E_k(\rho_{k-1})$	$E_k(0)$	ρ_k	$\|\|d_k\|\|$	α_k
0	+6.00e+02	4.70e+02	2.43e+02	2.27e+02	1.00e+00	2.48e+01	1.00e+00
1	+2.36e+02	2.45e+02	6.35e+01	8.24e+01	1.00e+00	2.89e+00	1.00e+00
2	+1.04e+02	2.24e+02	1.03e+01	2.76e+01	1.00e-01	7.69e+00	5.00e-01
3	+4.42e+02	1.58e+02	3.21e+00	1.63e+01	1.00e-02	1.10e+01	5.00e-01
4	+2.86e+03	8.21e+01	2.64e+00	1.00e+01	1.00e-03	5.64e+00	1.00e+00
5	+2.97e+04	1.05e+01	3.42e+00	4.83e+00	1.00e-04	3.05e+00	1.00e+00
6	+5.69e+04	3.08e+00	4.90e-01	2.53e+00	1.00e-04	1.66e+00	1.00e+00
7	+6.28e+04	1.89e+00	4.16e-02	1.65e+00	1.00e-05	1.50e+00	1.00e+00
8	+9.33e+04	2.88e-01	2.50e-01	3.44e-01	1.00e-05	9.55e-01	1.00e+00
9	+1.03e+05	2.07e-02	1.86e-02	2.44e-02	1.00e-05	2.17e-01	1.00e+00
10	+1.04e+05	4.24e-04	3.59e-04	4.99e-04	1.00e-05	1.54e-03	1.00e+00
11	+1.04e+05	1.31e-07	7.42e-08	--------	--------	--------	--------

Adding the single bound constraint tl[1]$\geq$ 5 to problem `batch`, however, makes the resulting problem infeasible. We refer to this problem as `batch1`. Running this problem with our implementation, using the same initial conditions as when problem `batch` is solved, yields the results in Table 5.

Table 5: Output for Example 4(b) (`batch1`)

k	f_k	v_k	$E_k(\rho_{k-1})$	$E_k(0)$	ρ_k	$\|\|d_k\|\|$	α_k
0	+6.00e+02	4.75e+02	2.43e+02	2.25e+02	1.00e+00	2.85e+01	1.00e+00
1	+2.36e+02	2.46e+02	6.25e+01	8.13e+01	1.00e+00	3.38e+00	1.00e+00
2	+1.04e+02	2.25e+02	1.03e+01	2.84e+01	1.00e-01	1.03e+01	5.00e-01
3	+4.35e+02	1.61e+02	3.35e+00	1.78e+01	1.00e-02	1.33e+01	5.00e-01
4	+2.85e+03	8.52e+01	2.75e+00	1.09e+01	1.00e-03	6.41e+00	1.00e+00
5	+3.01e+04	1.39e+01	3.55e+00	3.58e+00	1.00e-04	3.30e+00	1.00e+00
6	+5.72e+04	6.13e+00	5.00e-01	1.78e+00	1.00e-04	1.25e+00	1.00e+00
7	+6.35e+04	4.94e+00	3.71e-02	1.29e+00	1.00e-05	1.45e+00	1.00e+00
8	+9.40e+04	3.68e+00	1.70e-01	5.17e-01	1.00e-05	6.01e-01	1.00e+00
9	+1.09e+05	3.45e+00	1.58e-02	4.72e-01	1.00e-06	1.68e+00	1.00e+00
10	+1.63e+05	3.24e+00	1.22e-01	1.55e-01	1.00e-06	1.65e+00	1.00e+00
11	+2.27e+05	3.09e+00	3.92e-02	3.41e-02	1.00e-06	3.68e-01	1.00e+00
12	+2.27e+05	3.07e+00	3.87e-04	4.32e-02	1.00e-07	7.74e-01	1.00e+00
13	+2.66e+05	3.04e+00	6.76e-03	6.79e-03	1.00e-07	2.24e-01	1.00e+00
14	+2.66e+05	3.04e+00	1.19e-05	1.20e-05	1.44e-10	1.30e-01	1.00e+00
15	+2.66e+05	3.04e+00	7.82e-09	6.02e-09	--------	--------	--------

Example 5: We refer to our final example problem as `nactive`:

$$\begin{aligned} &\min\ x_1 \\ &\text{s.t. } \tfrac{1}{2}(-x_1 - x_2^2 - 1) \geq 0 \\ &\quad\ \ x_1 - x_2^2 \geq 0 \\ &\quad\ \ -x_1 + x_2^2 \geq 0. \end{aligned} \tag{5.3}$$

This problem is infeasible with $n = 2$ active constraints at the minimizer of infeasibility $\hat{x} = (0, 0)$. We ran this problem with the starting point $(-20, 10)$, and the results are provided in Table 6.

Table 6: Output for Example 5 (`nactive`)

k	f_k	v_k	$E_k(\rho_{k-1})$	$E_k(0)$	ρ_k	$\|\|d_k\|\|$	α_k
0	-2.00e+01	1.60e+02	6.05e+01	1.15e+02	1.00e-01	2.01e+01	1.00e+00
1	-5.00e-01	3.79e+01	2.37e+01	2.51e+01	1.00e-01	1.47e+01	1.00e+00
2	-1.47e+01	1.57e+01	1.43e+01	1.53e+01	1.00e-01	1.37e+01	1.00e+00
3	-1.03e+00	1.03e+00	4.12e-01	5.15e-01	1.00e-01	1.03e+00	1.00e+00
4	-1.61e-10	5.00e-01	6.09e-05	5.54e-05	3.06e-09	2.52e-05	1.00e+00
5	-6.27e-10	5.00e-01	5.03e-06	5.03e-06	2.53e-11	2.52e-06	1.00e+00
6	-6.33e-12	5.00e-01	3.21e-12	3.21e-12	--------	--------	--------

By the commentary at the end of Section 3, the algorithm would have converged to the infeasible stationary point $\hat{x}$ even if the penalty parameter was not driven to zero. However, our method for ensuring fast convergence, namely step f of Algorithm II, always forces ρ to zero on infeasible problems, which is apparent in Table 6.

To verify that it is not necessary to drive ρ to zero in this problem, we disabled step f of Algorithm II and report the output in Table 7. Note that ρ is never changed and that quadratic convergence is obtained. The results are in fact slightly better than those in Table 6. In practice, it may be advantageous to identify the cases when the algorithm is converging to an infeasible stationary point where n or more constraints are active and not impose the quadratic decrease in ρ, but for general purpose implementations the advantage of carefully driving $\rho \to 0$ is illustrated by all of the experiments in this section.

Table 7: Output for Example 5 (`nactive` with modified ρ update)

k	f_k	v_k	$E_k(\rho_{k-1})$	$E_k(0)$	ρ_k	$\|\|d_k\|\|$	α_k
0	-2.00e+01	1.60e+02	6.05e+01	1.15e+02	1.00e-01	2.01e+01	1.00e+00
1	-5.00e-01	3.79e+01	2.37e+01	2.51e+01	1.00e-01	1.47e+01	1.00e+00
2	-1.47e+01	1.57e+01	1.43e+01	1.53e+01	1.00e-01	1.37e+01	1.00e+00
3	-1.03e+00	1.03e+00	4.12e-01	5.15e-01	1.00e-01	1.03e+00	1.00e+00
4	-1.61e-10	5.00e-01	6.09e-05	5.54e-05	1.00e-01	2.77e-05	1.00e+00
5	-7.66e-10	5.00e-01	3.07e-10	3.84e-10	--------	--------	--------

References

[1] M. Anitescu. Global convergence of an elastic mode approach for a class of Mathematical Programs with Complementarity Constraints. *SIAM Journal on Optimization*, 16(1):120–145, 2005.

[2] D. P. Bertsekas. *Nonlinear Programming*. Athena Scientific, Belmont, Massachusetts, 1995.

[3] P. T. Boggs, J. W. Tolle, and P. Wang. On the local convergence of quasi-Newton methods for constrained optimization. *SIAM Journal on Control and Optimization*, 20(2):161–171, 1982.

[4] E. G. Boman. *Infeasibility and negative curvature in optimization*. PhD thesis, Scientific Computing and Computational Mathematics, Stanford University, Palo Alto, California, USA, 1999.

[5] R. H. Byrd, N. I. M. Gould, J. Nocedal, and R. A. Waltz. An algorithm for nonlinear optimization using linear programming and equality constrained subproblems. *Mathematical Programming, Series B*, 100(1):27–48, 2004.

[6] R. H. Byrd, G. López-Calva, and J. Nocedal. A line search penalty method for nonlinear optimization. Technical Report 08/05, Optimization Technology Center, Northwestern University, 2008.

[7] R. H. Byrd, J. Nocedal, and R. A. Waltz. Steering exact penalty methods. *Optimization Methods and Software*, 23(2), 2008.

[8] R. H. Byrd, J. Nocedal, and R.A. Waltz. KNITRO: An integrated package for nonlinear optimization. In G. di Pillo and M. Roma, editors, *Large-Scale Nonlinear Optimization*, pages 35–59. Springer, 2006.

[9] R. H. Byrd, R. A. Tapia, and Y. Zhang. An SQP augmented Lagrangian BFGS algorithm for constrained optimization. *SIAM Journal on Optimization*, 2(2):210–241, 1992.

[10] R. M. Chamberlain, M. J. D. Powell, C. Lemaréchal, and H. C. Pedersen. The watchdog technique for forcing convergence in algorithms for constrained optimization. *Mathematical Programming Studies*, 16(MAR):1–17, 1982.

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

[12] T. F. Coleman and A. R. Conn. Second-order conditions for an exact penalty function. *Mathematical Programming*, 19(2):178–185, 1980.

[13] J. E. Dennis and R. B. Schnabel. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Prentice-Hall, Englewood Cliffs, New Jersey, USA, 1983. Reprinted as *Classics in Applied Mathematics 16*, SIAM, Philadelphia, Pennsylvania, USA, 1996.

[14] J.-P. Dussault. Numerical stability and efficiency of penalty algorithms. *SIAM Journal on Numerical Analysis*, 32(1):296–317, 1995.

[15] R. Fletcher. An ℓ_1 penalty method for nonlinear constraints. In P. T. Boggs, R. H. Byrd, and R. B. Schnabel, editors, *Numerical Optimization 1984*, pages 26–40, Philadelphia, USA, 1985. SIAM.

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

[17] R. Fletcher and S. Leyffer. Nonlinear programming without a penalty function. *Mathematical Programming*, 91:239–269, 2002.

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

[19] P. E. Gill, W. Murray, and M. H. Wright. *Practical Optimization*. Academic Press, London, 1981.

[20] N. I. M. Gould, D. Orban, and Ph. Toint. An interior-point L1-penalty method for nonlinear optimization. Technical Report RAL-TR-2003-022, Rutherford Appleton Laboratory Chilton, Oxfordshire, UK, 2003.

[21] N. I. M. Gould, D. Orban, and Ph. L. Toint. CUTEr and sifdec: A Constrained and Unconstrained Testing Environment, revisited. *ACM Trans. Math. Softw.*, 29(4):373–394, 2003.

[22] S. P. Han and O. L. Mangasarian. Exact penalty functions in nonlinear programming. *Mathematical Programming*, 17(1):251–269, 1979.

[23] R. Horst and H. Tuy. *Global Optimization*. Springer, 3 edition, 1996.

[24] X.M. Hu and D. Ralph. Convergence of a penalty method for mathematical programming with complementarity constraints. *Journal of Optimization Theory and Applications*, 123(2):365–390, 2004.

[25] S. Leyffer, G. López-Calva, and J. Nocedal. Interior methods for mathematical programs with complementarity constraints. *SIAM Journal on Optimization*, 17(1):52–77, 2006.

[26] O. L. Mangasarian and J. S. Pang. Exact penalty functions for mathematical programs with linear complementarity constraints. *Optimization*, 42:1–8, 1997.

[27] M. Mongeau and A. Sartenaer. Automatic decrease of the penalty parameter in exact penalty function methods. *European Journal of Operational Research*, 83(3):686–699, 1995.

[28] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer Series in Operations Research. Springer, second edition, 2006.

[29] T. Pietrzykowski. An exact potential method for constrained maxima. *SIAM Journal on Numerical Analysis*, 6(2):299–304, 1969.

[30] M. J. D. Powell. Variable metric methods for constrained optimization. In Bachem, A., Grötschel, M., and Korte, B., editors, *Mathematical Programming : The State of the Art, Bonn 1982*. Springer-Verlag, 1983.

[31] M.J.D. Powell. The convergence of variable metric methds for nonlinearly constrained optimization calculations. In O. Mangasarian, R. Meyer, and S. Robinson, editors, *Nonlinear Programming 3*, London, 1978. Academic Press.

[32] N. V. Sahinidis and M. Tawarmalani. *BARON 7.2.5: Global Optimization of Mixed-Integer Nonlinear Programs,* User's Manual, 2005.

[33] Scholtes, S. and M. Stöhr. Exact penalization of mathematical programs with equilibrium constraints. *SIAM Journal on Control and Optimization*, 37(2):617–652, 1999.

[34] W.I. Zangwill. Non-linear programming via penalty functions. *Management Science*, 13:344–358, 1967.