Science & Technology Facilities Council

Technical Report

A second derivative SQP method: local convergence

N I M Gould D P Robinson

February 9, 2009

RAL-TR-2009-002

Library and Information Services
SFTC Rutherford Appleton Laboratory
Harwell Science and Innovation Campus
Didcot
OX11 0QX
UK
Tel: +44 (0)1235 445384
Fax: +44(0)1235 446403
Email: library@rl.ac.uk

ISSN 1358-6254

RAL-TR-2009-002

A second derivative SQP method: local convergence[1]

Nicholas I. M. Gould[2,3] and Daniel P. Robinson[3]

ABSTRACT

Gould and Robinson (RAL-TR-2009-001) gave global convergence results for a second-derivative SQP method for minimizing the exact ℓ_1-merit function for a *fixed* value of the penalty parameter. To establish this result, we used the properties of the so-called Cauchy step, which was itself computed from the so-called predictor step. In addition, we allowed for the computation of a variety of (optional) SQP steps that were intended to improve the efficiency of the algorithm.

Although we established global convergence of the algorithm, we did not discuss certain aspects that are critical when developing software capable of solving general optimization problems. In particular, we must have strategies for updating the penalty parameter and better techniques for defining the positive-definite matrix B_k used in computing the predictor step. In this paper we address both of these issues. We consider two techniques for defining the positive-definite matrix B_k—a simple diagonal approximation and a more sophisticated *limited*-memory BFGS update. We also analyze a strategy for updating the penalty parameter based on *approximately* minimizing the ℓ_1-penalty function over a sequence of increasing values of the penalty parameter. Algorithms based on exact penalty functions have certain desirable properties. To be practical, however, these algorithms must be guaranteed to avoid the so-called Maratos effect. We show that a *nonmonotone* variant of our algorithm avoids this phenomenon and, therefore, results in asymptotically superlinear local convergence; this is verified by preliminary numerical results on the Hock and Shittkowski test set.

Keywords: Nonlinear programming, nonlinear inequality constraints, sequential quadratic programming, ℓ_1-penalty function, nonsmooth optimization

AMS classification: 49J52, 49M37, 65F22, 65K05, 90C26, 90C30, 90C55.

---

[1] This work was supported by the EPSRC grants EP/E053351/1 and EP/F005369/1.

[2] Computational Science and Engineering Department, Rutherford Appleton Laboratory, Chilton, Oxfordshire, OX11 0QX, England, EU. Email: nick.gould@stfc.ac.uk .
Current reports available from "http://www.numerical.rl.ac.uk/reports/reports.shtml".

[3] Oxford University Computing Laboratory, Numerical Analysis Group, Wolfson Building, Parks Road, Oxford, OX1 3QD, England, EU.
Email: nick.gould@comlab.ox.ac.uk, daniel.robinson@comlab.ox.ac.uk
Current reports available from "http://web.comlab.ox.ac.uk/oucl/publications/natr/index.html".

Computational Science and Engineering Department
Atlas Centre
Rutherford Appleton Laboratory
Oxfordshire OX11 0QX

February 9, 2009

1. Introduction

In [19], we presented a sequential inequality/equality constrained quadratic programming algorithm (an SIQP/SEQP "hybrid") for solving the problem

$$(\ell_1\text{-}\sigma) \qquad \underset{x\in\mathbb{R}^n}{\text{minimize}} \;\; \phi(x) = f(x) + \sigma\|[c(x)]^-\|_1,$$

where the constraint vector $c(x) : \mathbb{R}^n \to \mathbb{R}^m$ and the objective function $f(x) : \mathbb{R}^n \to \mathbb{R}$ are assumed to be twice continuously differentiable, σ is a positive scalar known as the penalty parameter, and we have used the notation $[v]^- = \min(0, v)$ for a generic vector v (the minimum is understood to be component-wise). The motivation for solving this problem is that solutions of problem $(\ell_1\text{-}\sigma)$ correspond (under certain assumptions) to solutions of the nonlinear programming problem

$$(\text{NP}) \qquad \underset{x\in\mathbb{R}^n}{\text{minimize}} \;\; f(x) \quad \text{subject to} \quad c(x) \geq 0.$$

An outline of this paper is as follows. In Section 2 we provide methods for defining the positive-definite matrix associated with the so-called predictor step subproblem, while in Section 3 we discuss a strategy for updating the penalty parameter. In Section 4 we discuss the local convergence properties of a *nonmonotone* variant of the algorithm described in [19] that culminates with two rate-of-convergence results. The first applies when the so-called SQP step is computed from an *equality* constrained subproblem based on the predictor step, while the second applies when the SQP step is computed from an *inequality* constrained subproblem based on the so-called Cauchy step [19, Sections 2.3.1 and 2.3.2]. Finally, in Section 5, we provide preliminary numerical results for the proposed algorithm.

Before proceeding, we catalogue essential notation and provide an outline of the algorithm presented in [19]. The outline is relatively brief and, therefore, we recommend a careful reading of [19] since this paper is essentially a continuation of that work.

1.1. Notation and definitions

Most of our notation is standard. We let e denote the vector of all ones whose dimension is determined by the context. A local solution of (NP) is denoted by x^*; $g(x)$ is the gradient of $f(x)$, and $H(x)$ its (symmetric) Hessian; the matrix $H_j(x)$ is the Hessian of $c_j(x)$; $J(x)$ is the $m \times n$ Jacobian matrix of the constraints with ith row $\nabla c_i(x)^T$. The Lagrangian function associated with (NP) is $\mathcal{L}(x, y) = f(x) - y^T c(x)$. The Hessian of the Lagrangian with respect to x is $\nabla^2_{xx}\mathcal{L}(x, y) = H(x) - \sum_{j=1}^m y_j H_j(x)$.

For a general vector v, the notation $[v]^- = \min(0, v)$ is used, where the minimum is understood to be component-wise, and $\text{diag}(v)$ represents a diagonal matrix whose ith diagonal entry is v_i; given two general vectors v and w, the notation $v \cdot w$ represents the vector whose ith component is $v_i w_i$; given a general symmetric matrix A the notation $A \succeq \lambda$ means that the smallest eigenvalue of A is bigger than or equal to λ; and given a

set of of matrices A_1, A_2, ..., A_p for some $p \geq 1$, we define $\text{diag}(A_1, A_2, \ldots, A_p)$ to be the block-diagonal matrix whose ith block is A_i.

Given a vector $v \in \mathbb{R}^n$ and scalar ε we define $\mathcal{B}_\varepsilon(v) = \{x \in \mathbb{R}^n : \|x - v\|_2 < \varepsilon\}$ and $\bar{\mathcal{B}}_\varepsilon(v) = \{x \in \mathbb{R}^n : \|x - v\|_2 \leq \varepsilon\}$ to be the open and the closed ball centered at v of radius ε, respectively.

We often consider problem functions evaluated at a specific point x_k. To simplify notation we define the following: $f_k = f(x_k)$, $c_k = c(x_k)$, $g_k = g(x_k)$ and $J_k = J(x_k)$. Given a pair of values (x_k, y_k), we let H_k and B_k denote symmetric approximations to $H(x_k, y_k)$ in which B_k is required additionally to be positive definite. Similar notation is used for a solution x^*; we define $f^* = f(x^*)$, $c^* = c(x^*)$, $g^* = g(x^*)$, and $J^* = J(x^*)$. When a primal-dual solution (x^*, y^*) is given, we define $H^* = H(x^*, y^*)$.

Given the symmetric matrix H_k, we define

$$M_k^H(s) \stackrel{\text{def}}{=} M_k^H(s\,; x_k) = f_k + g_k^T s + \tfrac{1}{2} s^T H_k s + \sigma \|[c_k + J_k s]^-\|_1 \tag{1.1}$$

to be the *faithful* model of ϕ and

$$\Delta M_k^H(s) \stackrel{\text{def}}{=} \Delta M_k^H(s\,; x_k) = M_k^H(0\,; x_k) - M_k^H(s\,; x_k) \tag{1.2}$$

to be the *change* in the faithful model. We mention that this notation does not allude to their dependence on the penalty parameter σ.

Given a solution x^* to problem (NP), we use the indexing sets $\mathcal{A} \stackrel{\text{def}}{=} \{i : c_i(x^*) = 0\}$ and $\mathcal{I} \stackrel{\text{def}}{=} \{i : c_i(x^*) > 0\}$, which are the set of active and inactive constraints, respectively, at x^*. Given a generic vector v, a generic matrix V, and a generic indexing set S, the notation v_S and V_S will denote the rows of v and V that correspond to the indices in S; if v and V are functions of x, then we sometimes write $v_S(x)$ and $V_S(x)$ instead of $[v(x)]_S$ and $[V(x)]_S$.

We use the following definitions related to a solution of problem (NP).

Definition 1.1. (First-order KKT point) *We say that the point (x^*, y^*) is a first-order KKT point for problem* (NP) *if*

$$g^* - J^{*T} y^* = 0, \quad c^* \geq 0, \quad y^* \geq 0, \quad \textit{and} \quad c^* \cdot y^* = 0. \tag{1.3}$$

Definition 1.2. (Second-order sufficient conditions) *A point (x^*, y^*) satisfies the second-order sufficient conditions for problem* (NP) *if (x^*, y^*) is a first-order KKT point and if there exists $\lambda_{\min}^H > 0$ such that $s^T H^* s \geq \lambda_{\min}^H s^T s$ for all s satisfying $J_{\mathcal{A}}^* s = 0$.*

Definition 1.3. (Strict complementarity) *We say that strict complementarity holds at a KKT point (x^*, y^*) for problem* (NP) *if $y_{\mathcal{A}}^* > 0$.*

Definition 1.4. (Linear independent constraint qualification) *We say that the linear independent constraint qualification* (LICQ) *holds at a KKT point (x^*, y^*) for problem* (NP) *if $J_{\mathcal{A}}^*$ has full row rank.*

Definition 1.5. *We say that the* strong *second-order sufficient conditions hold at a point (x^*, y^*) if it satisfies Definitions* 1.1 – 1.4.

1.2. Algorithm Overview

We now give a brief description of the algorithm we proposed in [19] for minimizing problem (ℓ_1-σ); the algorithm has been restated as Algorithm 1.1. First, the user supplies an initial guess (x_0, y_0) of a solution to problem (ℓ_1-σ). Next, "success" parameters $0 < \eta_S \leq \eta_{VS} < 1$, a maximum allowed predictor trust-region radius $\Delta_{\rm U}$, expansion and contraction factors $0 < \eta_c < 1 < \eta_e$, sufficient model decrease and approximate Cauchy point tolerances $0 < \eta \leq \eta_{\rm ACP} < 1$, and SQP trust-region radius factor τ_f are defined. With parameters set, the main "do-while" loop begins. First, the problem functions are evaluated at the current point (x_k, y_k). Next, a symmetric positive-definite matrix B_k is defined and the predictor step $s_k^{\rm P}$ is computed as a solution to

$$\underset{s\in\mathbb{R}^n}{\text{minimize}} \;\; f_k + g_k^T s + \tfrac{1}{2} s^T B_k s + \sigma\|[c_k + J_k s]^-\|_1 \quad \text{subject to} \quad \|s\|_\infty \leq \Delta_k^{\rm P}. \tag{1.4}$$

By introducing elastic variables [15], this problem is equivalent to

$$\underset{s\in\mathbb{R}^n}{\text{minimize}} \;\; f_k + g_k^T s + \tfrac{1}{2} s^T B_k s + \sigma e^T v \quad \text{subject to} \quad c_k + J_k s + v \geq 0,\; v \geq 0,\; \|s\|_\infty \leq \Delta_k^{\rm P}. \tag{1.5}$$

Strategies for defining the positive-define matrix B_k are discussed in Section 2. Next, we define the Cauchy step as $s_k^{\rm CP} = \alpha_k s_k^{\rm P}$, where α_k is the solution to

$$\underset{0\leq\alpha\leq 1}{\text{minimize}} \;\; M_k^H(\alpha s_k^{\rm P}), \tag{1.6}$$

and then compute $\Delta M_k^H(s_k^{\rm CP})$. We then have the option of computing an SQP step $s_k^{\rm S}$ as the solution of any of the subproblems discussed in [19, Section 3.2]. The trial step computation is completed by defining the full step s_k so that the condition

$$\Delta M_k^H(s_k) \geq \eta \Delta M_k^H(s_k^{\rm CP}) \tag{1.7}$$

is satisfied for some constant $0 < \eta < 1$ independent of k (see [19, Section 2.3] for more details). Next, we evaluate $\phi(x_k + s_k)$ and $\Delta M_k^H(s_k)$ and compute the ratio r_k of actual versus predicted decrease in the merit function.

The strategy for updating the trust-region radii and for accepting or rejecting candidate steps is identical to that used by Fletcher [12] and is determined by the ratio r_k (except we have the added responsibility of updating the SQP trust-region radius). More precisely, if the ratio r_k is larger than η_{VS}, then we believe that the model is a very accurate representation of the merit function within the current trust-region; therefore, we increase the predictor trust-region radius with the belief that the current trust-region radius may be overly restrictive. If the ratio is greater than η_S, then we believe the model is sufficiently accurate and we keep the predictor trust-region radius fixed. Otherwise, the ratio indicates that there is poor agreement between the model M_k^H and the merit function and, therefore, we decrease the predictor trust-region radius with the hope that the model will accurately capture the behavior of the merit function over the smaller trust-region. As for step acceptance or rejection, we accept any iterate for which r_k is positive, since this

indicates that the merit function has decreased. We note that the update used for the dual vector y_{k+1} is not important for proving global convergence, so no specific update is provided. However, the update to y_k is crucial when considering local convergence and the multiplier vector from the SQP subproblem is the most obvious candidate. We consider this further in Section 4. Finally, the SQP trust-region radius is defined to be a constant multiple of the predictor trust-region radius, although the condition $\Delta_{k+1}^{\text{S}} \leq \tau_f \cdot \Delta_{k+1}^{\text{P}}$ for some constant τ_f is also sufficient.

Algorithm 1.1. Minimizing the ℓ_1-penalty function
Input: (x_0, y_0)
Set parameters $0 < \eta_s \leq \eta_{vs} < 1$, $\Delta_{\text{U}} > 0$, $0 < \eta \leq \eta_{\text{ACP}} < 1$, and $\tau_f > 0$.
Set expansion and contraction factors $0 < \eta_c < 1 < \eta_e$.
$k \leftarrow 0$
do
  Evaluate f_k, g_k, c_k, J_k, and then compute ϕ_k.
  Define B_k and H_k to be symmetric approximations to $H(x_k, y_k)$ with B_k positive definite.
  Solve problem (1.4) for s_k^{P}.
  Solve problem (1.6) for s_k^{CP} and compute $\Delta M_k^H(s_k^{\text{CP}})$.
  Possibly compute an SQP step s_k^{S}.
  Define a full step s_k that satisfies condition (1.7).
  Evaluate $\phi(x_k + s_k)$ and $\Delta M_k^H(s_k)$.
  Compute $r_k = \big(\phi_k - \phi(x_k + s_k)\big)/\Delta M_k^H(s_k)$.
  if $r_k \geq \eta_{vs}$ [very successful]
    $\Delta_{k+1}^{\text{P}} \leftarrow \min(\, \eta_e \cdot \Delta_k^{\text{P}},\, \Delta_{\text{U}}\,)$ [increase predictor radius]
  else if $r_k \geq \eta_s$ [successful]
    $\Delta_{k+1}^{\text{P}} \leftarrow \Delta_k^{\text{P}}$ [keep predictor radius]
  else [unsuccessful]
    $\Delta_{k+1}^{\text{P}} \leftarrow \eta_c \cdot \Delta_k^{\text{P}}$ [decrease predictor radius]
  end
  if $r_k > 0$ [accept step]
    $x_{k+1} \leftarrow x_k + s_k$
    $y_{k+1} \leftarrow$ arbitrary
  else [reject step]
    $x_{k+1} \leftarrow x_k$
    $y_{k+1} \leftarrow y_k$
  end
  $\Delta_{k+1}^{\text{S}} \leftarrow \tau_f \cdot \Delta_{k+1}^{\text{P}}$ [update SQP radius]
  $k \leftarrow k + 1$
end do

The following global convergence result applies to Algorithm 1.1.

Theorem 1.1. [19, Theorem 4.1] *Let f and c be twice continuously differentiable functions, and let $\{x_k\}$, $\{H_k\}$, $\{B_k\}$, $\{\Delta_k^p\}$, and $\{\Delta_k^s\}$, be sequences generated by* Algorithm 1.1. *Assume that the following conditions hold:*

1. *$\{x_k\}_{k\geq 0} \subset \mathcal{B} \subset \mathbb{R}^n$, where $\mathcal{B}$ is a closed and bounded set; and*
2. *there exists positive constants b_B and b_H such that $\|B_k\|_2 \leq b_B$ and $\|H_k\|_2 \leq b_H$ for all $k \geq 0$.*

Then, either x_K is a first-order point for problem (ℓ_1-σ) for some $K \geq 0$, or there exists a subsequence of $\{x_k\}$ that converges to a first-order solution of problem (ℓ_1-σ).

2. Defining B_k

The definition of the positive-definite matrix B_k is critical in developing an efficient algorithm. In Section 2.1 we consider defining B_k as a diagonal matrix. Although this approach is simple to implement and cheap to compute, it can not be expected to perform well in general. A more promising idea is to *update* B_k using the well-known BFGS formula. This is an appealing alternative, but we must be cautious since the matrix $H(x_k, y_k)$ is generally indefinite and, therefore, the traditional BFGS update may result in an indefinite matrix [2, 33]. In Section 2.2 we consider a *limited-memory* BFGS update since we are interested in solving large-scale problems.

2.1. A diagonal approximation

Given scalars $\nu_i > 0$ for $i = 1 : n$, we define the diagonal matrix $B_{k+1} = \operatorname{diag}(\nu_1, \nu_2, \ldots, \nu_n)$. Possibly the simplest choice is

$$\nu_i = \max\left(\left| \left(s_k^T H(x_k, y_k) s_k\right) / \left(s_k^T s_k\right) \right|, \, \varepsilon \right) \tag{2.1}$$

for all i, where ε is a small pre-defined positive constant (ε has this meaning for the remainder of this section). This strategy approximates the magnitude of the curvature of $H(x_k, y_k)$ in the previous direction s_k.

A second possibility is to utilize more of the matrix $H(x_k, y_k)$. Given any value $0 \leq r \leq n$ we may define

$$\nu_i = \max\left(\frac{1}{r_u - r_l + 1} \sum_{j=r_l}^{r_u} |[H(x_k, y_k)]_{ij}|, \, \varepsilon \right), \tag{2.2}$$

where $r_l = \max(i - r, 1)$ and $r_u = \min(n, i + r)$. In other words, ν_i is the average of the absolute values of the elements of $H(x_k, y_k)$ in row i within band-width r. We note that if $r = 0$, then the curvature of B_{k+1} and $H(x_k, y_k)$ will agree in those standard coordinate directions for which $H(x_k, y_k)$ is sufficiently positive definite as determined by the parameter ε.

2.2. A limited-memory BFGS update

Symmetric positive-definite approximations based on equations (2.1) and (2.2) are cheap to compute, but can not be expected to approximate $H(x_k, y_k)$ very well. An attractive alternative is to define the matrix B_{k+1} from the positive-definite matrix B_k by using a limited-memory quasi-Newton BFGS update. This approach uses a fixed number of vectors, say l, to define a positive-definite approximation to $H(x_k, y_k)$ based on the most recent l iterations (for more details see [28, 2]). If we define $d_k = \nabla_x \mathcal{L}(x_k + s_k, y_{k+1}) - \nabla_x \mathcal{L}(x_k, y_{k+1})$, then we may write the update as

$$B_k = B_k^0 + \sum_{i=k-l}^{k-1} (q_i q_i^T - p_i p_i^T), \tag{2.3}$$

where B_k^0 denotes *any* initial positive-definite approximation to $H(x_k, y_k)$ and

$$p_i = \frac{B_i s_i}{(s_i^T B_i s_i)^{1/2}}, \quad q_i = \frac{d_i}{(d_i^T s_i)^{1/2}}, \quad \text{and} \quad B_i = B_k^0 + \sum_{j=k-l}^{i-1} (q_j q_j^T - p_j p_j^T). \tag{2.4}$$

Note that in these definitions we have assumed that $k \geq l-1$ so that there are l vectors to use. This formula is relatively simple, but one must be careful. It is tempting to store the vector-pairs (p_i, q_i). However, as equation (2.4) illustrates, the vector p_i is defined from B_i and the matrix B_i changes from iteration to iteration since the "oldest" vector-pair (s_i, d_i) is removed from the set of l vector-pairs. Hence, the vector p_i must be recomputed at each iteration. The relationships given by equation (2.4) suggest how this may be done since

$$B_i s_i = B_k^0 s_i + \sum_{j=k-l}^{i-1} \left[(q_j^T s_i) q_j - (p_j^T s_i) p_j \right]. \tag{2.5}$$

Algorithm 2.1, which is [28, Procedure 7.6], computes the vector-pair (p_i, q_i) recursively.

Algorithm 2.1. Computing the vector-pairs (p_i, q_i)
for $i = k-l, k-l+1, \ldots, k-1$
  $q_i \leftarrow d_i/(d_i^T s_i)^{1/2}$
  $p_i \leftarrow B_k^0 s_i + \sum_{j=k-l}^{i-1} \left[(q_j^T s_i) q_j - (p_j^T s_i) p_j \right]$
  $p_i \leftarrow p_i/(s_i^T p_i)^{1/2}$
end (for)

During the kth iteration, Algorithm 2.1 computes the values q_i for $k-l \leq i \leq k-1$ and $q_j^T s_i$ for all $k-l \leq j \leq i-1$. However, since q_i only depends on the data (d_i, s_i), only the value q_{k-1} and values $q_j^T s_{k-1}$ $(k-l \leq j \leq k-2)$ need to be computed (the other quantities should be stored from previous iterations).

Once the vector-pairs (p_i, q_i) have been computed, we set $B_k = B_k^0 - PP^T + QQ^T$ where we have defined $P = [p_{k-l}\ p_{k-l+1}\ \ldots\ p_{k-1}]$ and $Q = [q_{k-l}\ q_{k-l+1}\ \ldots\ q_{k-1}]$. The predictor

subproblem (1.5) then becomes

$$\begin{array}{ll} \underset{s\in\mathbb{R}^n, v\in\mathbb{R}^m}{\text{minimize}} & f_k + g_k^T s + \frac{1}{2} s^T (B_k^0 - PP^T + QQ^T) s + \sigma e^T v \\ \text{subject to} & c_k + J_k s + v \geq 0, \quad v \geq 0, \quad \|s\|_\infty \leq \Delta_k^{\mathrm{P}}. \end{array} \tag{2.6}$$

If we define the $2l$ extra variables

$$w_a = P^T s \quad \text{and} \quad w_b = Q^T s, \tag{2.7}$$

then problem (2.6) is equivalent to

$$\begin{array}{ll} \underset{s,v,w_a,w_b}{\text{minimize}} & f_k + g_k^T s + \frac{1}{2}(s^T B_k^0 s - w_a^T w_a + w_b^T w_b) + \sigma e^T v \\ \text{subject to} & c_k + J_k s + v \geq 0, \quad P^T s = w_a, \quad Q^T s = w_b, \quad v \geq 0, \quad \|s\|_\infty \leq \Delta_k^{\mathrm{P}}. \end{array} \tag{2.8}$$

As a function of (s, v, w_a, w_b), the Hessian associated with subproblem (2.8) is given by

$$B_k^{\mathrm{A}} = \begin{pmatrix} B_k^0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -I & 0 \\ 0 & 0 & 0 & I \end{pmatrix}, \tag{2.9}$$

which is *not* positive definite. This may seem strange since problem (2.8) is equivalent to the *strictly convex* QP (2.6) (assuming that the updated matrix was positive definite). However, if the current iterate is feasible for subproblem (2.8), then any step that maintains linear feasibility is guaranteed to be a direction of positive curvature even though B_k^{A} is indefinite. To see this, suppose that (s, v, w_a, w_b) is a feasible point so that $w_a = P^T s$ and $w_b = Q^T s$. Furthermore, suppose that $P^T(s+\Delta s) = w_a + \Delta w_a$ and $Q^T(s+\Delta s) = w_b + \Delta w_b$. Simplification yields $P^T \Delta s = \Delta w_a$ and $Q^T \Delta s = \Delta w_b$. It then follows that

$$\begin{aligned} (\Delta s, \Delta v, \Delta w_a, \Delta w_b)^T B_k^{\mathrm{A}} (\Delta s, \Delta v, \Delta w_a, \Delta w_b) &= \Delta s^T B_k^0 \Delta s - \Delta w_a^T \Delta w_a + \Delta w_b^T \Delta w_b \\ &= \Delta s^T B_k^0 \Delta s - \Delta s^T P P^T \Delta s + \Delta s^T Q Q^T \Delta s \\ &= \Delta s^T (B_k^0 - PP^T + QQ^T) \Delta s \\ &= \Delta s^T B_k \Delta s > 0, \end{aligned}$$

since B_k is positive definite by construction. A great advantage in using subproblem (2.8) is that the Hessian matrix has essentially the same sparsity as B_k^0. In contrast, the Hessian matrix associated with subproblem (2.6) is generally dense since it uses a sum of rank-1 updates. Note, however, that the $2l$ extra constraints (2.7) are generally dense; fortunately a limited number of dense constraints can be accommodated easily by modern sparse QP solvers such as `QPA` and `QPB` from the GALAHAD library [18].

Until this point we have assumed that the limited-memory BFGS update results in a positive-definite matrix. However, it is well-known that this is true if and only if the

quantity $d_k^T s_k > 0$ and this is not guaranteed to hold. We say that the vector-pair (s_k, d_k) will result in a sufficiently positive-definite update if

$$s_k^T d_k \geq \eta_{\text{SPD}} s_k^T B_k s_k \tag{2.10}$$

for some positive scalar $0 < \eta_{\text{SPD}} < 1$. Since we want to reserve the notation s_k for the solution of kth iterate of Algorithm 1.1, we use the notation $(\bar{s}_k, \bar{d}_k)$ to denote the (possibly) modified values of (s_k, d_k) that satisfy condition (2.10).

When the estimate $d_k^T s_k \geq \eta_{\text{SPD}} s_k^T B_k s_k$ fails, the simplest course of action is to skip the update. Since we do not advocate this option, we use a *damping* technique introduced by Powell [33], which is guaranteed to produce acceptable values for $\bar{d}_k$ and $\bar{s}_k$ by perturbing d_k if necessary. This is accomplished by defining the "damped" vectors

$$\bar{d}_k = \theta_k d_k + (1 - \theta_k) B_k s_k, \tag{2.11a}$$

$$\bar{s}_k = s_k, \tag{2.11b}$$

where the *damping factor* θ_k is defined as

$$\theta_k = \begin{cases} 1 & \text{if } s_k^T d_k \geq \eta_{\text{SPD}} s_k^T B_k s_k, \\ (1 - \eta_{\text{SPD}}) s_k^T B_k s_k / (s_k^T B_k s_k - s_k^T d_k) & \text{if } s_k^T d_k < \eta_{\text{SPD}} s_k^T B_k s_k. \end{cases} \tag{2.12}$$

If $\theta_k \neq 1$, it can be verified by computation that $\bar{s}_k^T \bar{d}_k = \eta_{\text{SPD}} s_k^T B_k s_k$. Note that if $\theta_k = 0$, then $\bar{d}_k = B_k s_k$, $p_k = q_k$, and $B_{k+1} = B_k$.

We finish this section by briefly mentioning other strategies that could be implemented. First, we could approximate the *reduced* Hessian of the Lagrangian since it is this matrix that is known to be positive definite at a minimizer satisfying the second-order sufficient conditions (see Gill, Murray and Saunders [15]). It will be shown in Section 4 that if x^* is a solution to problem (NP) that satisfies the strong-second order sufficient conditions, then the active constraints at x^* will ultimately be identified by the predictor step s_k^{P}. If one examines the various SQP subproblems discussed in [19, Section 2], then it is reasonable to expect that the solution of the kth SQP subproblem will be "close enough" to the null space of the active constraints to be a direction of positive curvature. This observation suggests that we define

$$\bar{s}_k = s_k^{\text{S}} \quad \text{and} \quad \bar{d}_k = \nabla_x \mathcal{L}(x_{k+1}, y_{k+1}) - \nabla_x \mathcal{L}(x_k^{\text{P}}, y_{k+1}), \tag{2.13}$$

where $x_k^{\text{P}} = x_k + s_k^{\text{P}}$ and s_k^{S} is the solution to any of the SQP subproblems considered in [19, Section 2.3]. The quantity $\bar{s}_k^T \bar{d}_k$ approximates the curvature of the reduced Hessian and is likely to be positive definite in the neighborhood of a solution. Note that this strategy requires an extra evaluation of the first derivatives at x_k^{P}. Second, we could approximate the curvature of the Augmented Lagrangian. This has been studied by Han [21], Tapia [40], Byrd, Tapia, and Zhang [7], but we mention here the approach used in the software package SNOPT [15]. The idea is to use the augmented Lagrangian function to define a perturbation

Δd of d_k such that $(d_k + \Delta d)^T s_k = \eta_{\text{SPD}} s_k^T B_k s_k$. To obtain Δd, we consider the augmented Lagrangian function [32, 22]:

$$\mathcal{L}_{\mathcal{A}}(s, y, \Omega(\omega)) = f(x) - c(x)^T y + \tfrac{1}{2} c(x) \Omega(\omega) c(x), \tag{2.14}$$

where $\omega \equiv (\omega_1, \ldots, \omega_n)^T \in \mathbb{R}^m$ and $\Omega(\omega) = \text{diag}(\omega_1, \ldots, \omega_n)$. The gradient of the augmented Lagrangian with respect to x is given by

$$\nabla_x \mathcal{L}_{\mathcal{A}}\big(x, y, \Omega(\omega)\big) = g(x) - J(x)^T \big(y - \Omega(\omega) c(x)\big), \tag{2.15}$$

so that

$$\nabla_x \mathcal{L}_{\mathcal{A}}\big(x_{k+1}, y_{k+1}, \Omega(\omega)\big) - \nabla_x \mathcal{L}_{\mathcal{A}}\big(x_k, y_{k+1}, \Omega(\omega)\big) = d_k + \Delta d(\omega), \tag{2.16}$$

where $\Delta d(\omega) = J(x_{k+1})^T \Omega(\omega) c(x_{k+1}) - J_k^T \Omega(\omega) c_k$. The authors of SNOPT [15] suggest computing an ω of minimal norm so that $\bar{s}_k \stackrel{\text{def}}{=} s_k$ and $\bar{d}_k \stackrel{\text{def}}{=} d_k + \Delta d(\omega)$ satisfy condition (2.10) by solving the problem

$$\underset{\omega \in \mathbb{R}^n}{\text{minimize}} \ \|\omega\|_2^2 \quad \text{subject to} \quad a^T \omega = b, \ \ \omega \geq 0, \tag{2.17}$$

where $b = \eta_{\text{SPD}} \bar{s}_k^T B_k \bar{s}_k - d_k^T \bar{s}_k$, $a_i = c_i(x_{k+1}) t_i - c_i(x_k) r_i$, $t = J(x_{k+1}) \bar{s}_k$, and $r = J_k \bar{s}_k$. If no solution exists or if the norm of the solution is considered too large, then a different strategy should be used.

3. Updating the penalty parameter

The following theorem clarifies why it is essential to incorporate a strategy for updating the penalty parameter (see [9, 10, 31] for more details).

Theorem 3.1. *[9, Theorem 14.5.1] Suppose that f and c_i are twice continuously differentiable for $1 \leq i \leq m$ and that x^* and y^* are vectors such that $c(x^*) \geq 0$ and $\sigma \geq \|y^*\|_\infty$. Then if (x^*, y^*) satisfies the second-order sufficient conditions for problem* (NP)*, x^* also satisfies the second-order sufficient conditions for problem (ℓ_1-σ). In addition, if $\sigma > \|y^*\|_\infty$ then the second-order sufficient conditions for the two problems are equivalent.*

The updating scheme that we now discuss is based on the simple idea of calculating a sequence of approximate solutions for problem (ℓ_1-σ). After each approximate solution is computed, we check the constraint violation and if sufficient improvement is not obtained, then the penalty parameter is increased with the intent of driving the constraint violation to zero. Since the penalty parameter is now allowed to change over a sequence of iterations, we let σ_k denote the penalty parameter during the kth iterate. We accept the vector-pair (x_k, π_k) as an approximate solution for problem (ℓ_1-σ) if it satisfies

$$\varepsilon_k^{\text{D}} \geq \frac{\|g_k + \sigma_k J_k^T \pi_k\|_\infty}{1 + \|g_k\|_\infty} \tag{3.1a}$$

$$[\pi_k]_i = \begin{cases} [-\frac{\varepsilon_k^{\text{C}}}{\sigma_k}, \frac{\varepsilon_k^{\text{C}}}{\sigma_k}] & \text{if } [c_k]_i > \varepsilon_k^{\text{P}}, \\ [-1 - \varepsilon_k^{\text{C}}, \frac{\varepsilon_k^{\text{C}}}{\sigma_k}] & \text{if } -\varepsilon_k^{\text{P}} \leq [c_k]_i \leq \varepsilon_k^{\text{P}}, \\ [-1 - \varepsilon_k^{\text{C}}, -1 + \varepsilon_k^{\text{C}}] & \text{if } [c_k]_i < -\varepsilon_k^{\text{P}}, \end{cases} \tag{3.1b}$$

where $\varepsilon_k^{\mathrm{P}}$, $\varepsilon_k^{\mathrm{D}}$, and $\varepsilon_k^{\mathrm{C}}$ denote the kth primal, dual, and complementary-slackness tolerances, respectively, for problem (ℓ_1-σ). These conditions are based on the optimality conditions for an exact minimizer (x, π), which are given by $g(x) + \sigma_k J(x)^T \pi = 0$ for $\pi \in \partial \, \|[c(x)]^-\|_1$ (see [12, Section 14.3] for more details). Perhaps the most natural way of generating a vector π_k is to use the multipliers from either subproblem (1.5) or any of the *elastic* SQP subproblems considered in [19, Section 3.3]. If we define y_k to be any of those choices, then the optimality conditions for their respective subproblems (assuming that the TR constraint is inactive) suggest the definition $\pi_k = -y_k/\sigma_k$. Provided the sequence $\{y_k\}$ converges to a Lagrange multiplier vector for the elastic version of problem (ℓ_1-σ), this strategy will eventually produce a vector-pair (x_k, π_k) satisfying equation (3.1). A second way of generating a vector π_k is by defining it to be a solution to the optimization problem

$$\underset{\pi \in \mathbb{R}^m}{\text{minimize}} \;\; \tfrac{1}{2}\|g_k + \sigma_k J_k^T \pi\|_2^2 \quad \text{subject to} \quad \pi \text{ satisfying equation (3.1b).} \tag{3.2}$$

This will also eventually result in a vector-pair (x_k, π_k) satisfying equation (3.1), regardless of the predictor and SQP multipliers. Therefore, for a fixed value of the penalty parameter, we have a guaranteed method for computing a vector pair (x_k, π_k) that is an approximate critical point to problem (ℓ_1-σ).

Algorithm 3.1 provides the pseudo-code for updating the penalty parameter as well as the additional parameter initiations that must be made.

Algorithm 3.1. Updating σ based on an approximate critical point to problem (ℓ_1-σ).
begin (additions to preamble of Algorithm 1.1)
  Choose $\sigma_0 > 0$, $\eta_0 > 0$, $0 < \varepsilon_c < 1$, $0 < \varepsilon_0^{\mathrm{P}} < \varepsilon_c \eta_0$, $0 < \eta_c < 1$, and $1 < \sigma_e$.
  Set $\varepsilon_0^{\mathrm{D}} = \varepsilon_0^{\mathrm{P}}$ and $\varepsilon_0^{\mathrm{C}} = \varepsilon_0^{\mathrm{P}}$.
end (additions to preamble of Algorithm 1.1)
if (x_k, π_k) satisfies condition (3.1) **then** [an approximate critical point]
  if $c(x_k) \geq -\eta_k e$ [successful]
    $\eta_{k+1} \leftarrow \eta_c \eta_k$ [decrease η_k]
    $\varepsilon_{k+1}^{\mathrm{P}} \leftarrow \varepsilon_c \eta_{k+1}$ [ensure that $\varepsilon_k^{\mathrm{P}}$ is less than η_k]
    $\varepsilon_{k+1}^{\mathrm{D}} \leftarrow \varepsilon_{k+1}^{\mathrm{P}}, \quad \varepsilon_{k+1}^{\mathrm{C}} \leftarrow \varepsilon_{k+1}^{\mathrm{D}}$
    $\sigma_{k+1} \leftarrow \sigma_k$
  else [unsuccessful]
    $\eta_{k+1} \leftarrow \eta_k$
    $\varepsilon_{k+1}^{\mathrm{P}} \leftarrow \varepsilon_c \varepsilon_k^{\mathrm{P}}$
    $\varepsilon_{k+1}^{\mathrm{D}} \leftarrow \varepsilon_{k+1}^{\mathrm{P}}, \quad \varepsilon_{k+1}^{\mathrm{C}} \leftarrow \varepsilon_{k+1}^{\mathrm{D}}$
    $\sigma_{k+1} \leftarrow \sigma_e \sigma_k$ [increase σ_k]
  end if
else [not an approximate critical point]
  $\eta_{k+1} \leftarrow \eta_k, \quad \varepsilon_{k+1}^{\mathrm{P}} \leftarrow \varepsilon_k^{\mathrm{P}}, \quad \varepsilon_{k+1}^{\mathrm{D}} \leftarrow \varepsilon_k^{\mathrm{D}}, \quad \varepsilon_{k+1}^{\mathrm{C}} \leftarrow \varepsilon_k^{\mathrm{C}}, \quad \sigma_{k+1} \leftarrow \sigma_k$
end if

For simplicity, we have defined $\varepsilon_k^{\rm D} = \varepsilon_k^{\rm C} = \varepsilon_k^{\rm P}$. However, all that is required is that $\lim_{k\to\infty} \varepsilon_k^{\rm P} = \lim_{k\to\infty} \varepsilon_k^{\rm D} = \lim_{k\to\infty} \varepsilon_k^{\rm C} = 0$.

For numerical considerations, it is generally not desirable to let the penalty parameter grow "too large". However, there are two situations in which the penalty parameter should converge to infinity. The first is when the user supplies an optimization problem that is not well-defined. It is possible that the user may formulate a set of nonlinear constraints $c(x) \geq 0$ for which no feasible point exists. Detecting this situation is difficult and is equivalent to showing that the *global* solution of

$$\underset{x\in\mathbb{R}^n}{\text{minimize}} \quad \|[c(x)]^-\|_1 \tag{3.3}$$

is strictly positive. The second situation occurs when the iterates converge to a critical point of problem (3.3) for which $\|[c(x)]^-\|_1 > 0$. This undesirable situation may occur for *all* penalty methods, but it is rarely encountered in practice. Barring these two situations and under reasonable assumptions, Theorem 3.2 below shows that the penalty parameter remains uniformly bounded and that we can expect to generate an approximate solution to problem (NP) in a finite number of iterations. We use the following definition.

Definition 3.1. *A point x is a first-order critical point for problem (3.3) if it satisfies*

$$J(x)^T y = 0 \tag{3.4}$$

for some $y \in \partial\,\|[c(x)]^-\|_1$.

For given primal, dual, and complementary-slackness tolerances τ_p, τ_d, and τ_c, respectively, we say that a vector-pair (x_k, y_k) is an approximate solution to problem (NP) if it satisfies

$$\frac{\|g_k - J_k^T y_k\|_\infty}{1 + \|g_k\|_\infty} \leq \tau_d, \tag{3.5a}$$

$$c_k \geq -\tau_p e, \tag{3.5b}$$

$$y_k \geq -\tau_c e, \tag{3.5c}$$

$$\max(|c_k|, |y_k|) \leq \tau_c e, \tag{3.5d}$$

where condition (3.5d) should be interpreted component-wise.

Theorem 3.2. *Let all the assumptions of Theorem 1.1 hold and let $\{x_k\}$ be the sequence of iterates generated by Algorithm 1.1 with penalty parameter update given by Algorithm 3.1. Assume that at all limit points x_* of $\{x_k\}$, the Jacobian of active constraints has full row rank and if x_* is a first-order critical point for problem (3.3) then $\|[c(x_*)]^-\|_1 = 0$. Then*

(i) *the penalty parameter remains uniformly bounded; and*

(ii) *if τ_p, τ_d, and τ_c denote* positive *primal, dual, and complementary-slackness tolerances, respectively, for problem* (NP)*, then the algorithm described in this theorem terminates in a finite number of iterations with an approximate solution to problem* (NP) *as given by (3.5), where $y_k \stackrel{\rm def}{=} -\sigma_k \pi_k$ and (x_k, π_k) is an approximate solution to (ℓ_1-σ) as given by (3.1) for the value σ_k.*

Proof. We first note that

$$\lim_{k\to\infty} \varepsilon_k^{\mathrm{P}} = \lim_{k\to\infty} \varepsilon_k^{\mathrm{D}} = \lim_{k\to\infty} \varepsilon_k^{\mathrm{C}} = 0, \tag{3.6}$$

since Algorithm 1.1 is guaranteed to generate an infinite sequence $\{(x_k, \pi_k)\}$ of approximate critical points to problem (ℓ_1-σ).

We now prove part (i) by contradiction. Suppose that $\{\sigma_k\} \to \infty$. Examination of Algorithm 3.1 implies the existence of a subsequence $K_0 \subseteq \mathbb{N}$ such that for each $k \in K_0$ the vector-pair (x_k, π_k) is an approximate critical point to problem (ℓ_1-σ) as given by equation (3.1) for which $c_k \not\geq -\eta_k e$. Since $\{\pi_k\}$ is bounded and $\{x_k\}$ belongs to the compact set $\mathcal{B}$, we may pass to a further subsequence K_1 so that $\lim_{k\in K_1}(x_k, \pi_k) = (x_*, \pi_*)$. Equation (3.6) and condition (3.1b) then imply that $\pi_* \in \partial \|[c(x_*)]^-\|_1$, while condition (3.1a) and equation (3.6) imply

$$\lim_{k\in K_1} \frac{\|\frac{g_k}{\sigma_k} + J_k^T \pi_k\|_\infty}{1 + \|g_k\|_\infty} \leq \lim_{k\in K_1} \frac{\varepsilon_k^{\mathrm{D}}}{\sigma_k} = 0. \tag{3.7}$$

Since g is continuous by assumption and $\lim_{k\in K_1} x_k = x_*$, we know that $\lim_{k\in K_1} g_k/\sigma_k = 0$ and we may conclude from equation (3.7) that

$$J(x_*)^T \pi_* = 0. \tag{3.8}$$

Thus, x_* is a first-order critical point for problem (3.3), and it follows from the assumptions of this theorem that $\|[c(x_*)]^-\|_1 = 0$ so that $c(x_*) \geq 0$. Define the index set of active constraints at x_* to be $\mathcal{A}_* \stackrel{\text{def}}{=} \mathcal{A}(x_*)$. Since $\pi_* \in \partial \|[c(x_*)]^-\|_1$ and $c(x_*) \geq 0$, it follows that $[\pi_*]_i = 0$ for all $i \notin \mathcal{A}_*$, and therefore

$$J_{\mathcal{A}_*}(x_*)^T [\pi_*]_{\mathcal{A}_*} = 0. \tag{3.9}$$

Since $J_{\mathcal{A}_*}(x_*)$ has full row rank by assumption, it follows that $[\pi_*]_{\mathcal{A}_*} = 0$ and therefore $\pi_* = 0$. Since $\lim_{k\in K_1} \pi_k = \pi_* = 0$, $\lim_{k\to\infty} \sigma_k = \infty$, and $\lim_{k\to\infty} \varepsilon_k^{\mathrm{C}} = 0$, we conclude from (3.1b) that $c(x_k) \geq -\varepsilon_k^{\mathrm{P}} e$ for all $k \in K_1$ sufficiently large. However, since $k \in K_1$ we also know that $c(x_k) \not\geq -\eta_k e$. Combining these two inequalities, we have $\eta_k < \varepsilon_k^{\mathrm{P}}$ which contradicts how Algorithm 3.1 constructs the sequence $\{\varepsilon_k^{\mathrm{P}}\}$. Therefore, $\{\sigma_k\}$ is uniformly bounded.

We now prove part (ii) by contradiction. Suppose that Algorithm 1.1 does not terminate in a finite number of iterations. We may then define the subsequence $K_2 \subseteq \mathbb{N}$ such that for all $k \in K_2$ the vector-pair (x_k, π_k) is an approximate solution to problem (ℓ_1-σ) as given by equation (3.1). Since $\{\sigma_k\}$ is uniformly bounded from part (i) and since σ_k is increased by a constant factor when it is increased, there exists a number σ_b such that $\sigma_k = \sigma_b$ for all k sufficiently large. Consideration of Algorithm 3.1 then implies that $\lim_{k\to\infty} \eta_k = 0$. These observations and equation (3.6) imply the existence of an integer k_{C} such that the following estimates hold for all $k \geq k_{\mathrm{C}}$:

$$\sigma_k = \sigma_b, \quad \varepsilon_k^{\mathrm{D}} \leq \tau_d, \quad \eta_k \leq \min(\tau_p, \tau_c), \quad \varepsilon_k^{\mathrm{C}} \leq \min(\tau_c, 1) \text{ and } \varepsilon_k^{\mathrm{P}} \leq \tau_c. \tag{3.10}$$

We then define the subsequence K_3 to be the subsequence of K_2 for which $k \geq k_{\rm C}$.

We claim that the vector-pair (x_k, y_k) is an approximate solution to problem (NP) for all $k \in K_3$, where the auxiliary vector sequence $\{y_k\}$ is defined by $y_k = -\sigma_k \pi_k$. Condition (3.5a) follows from equation (3.1a) and equation (3.10) since

$$\frac{\|g_k - J_k^T y_k\|_\infty}{1 + \|g_k\|_\infty} = \frac{\|g_k + \sigma_b J_k^T \pi_k\|_\infty}{1 + \|g_k\|_\infty} \leq \varepsilon_k^{\rm D} \leq \tau_d \tag{3.11}$$

for all $k \in K_3$. Next, since all $k \in K_3$ are successful iterates by definition, we have

$$c_k \geq -\eta_k e \geq -\tau_p e, \tag{3.12}$$

where the second inequality follows from equation (3.10). Thus, condition (3.5b) is satisfied. Condition (3.5c) may be verified from equations (3.1b) and (3.10) since

$$y_k = -\sigma_b \pi_k \geq -\sigma_b \frac{\varepsilon_k^{\rm C}}{\sigma_b} = -\varepsilon_k^{\rm C} \geq -\tau_c \tag{3.13}$$

for all $k \in K_3$. Finally, we verify condition (3.5d). Let $k \in K_3$ and consider the ith component of c_k and π_k. If $|[c_k]_i| \leq \tau_c$, then condition (3.5d) is satisfied. Therefore, we assume that $|[c_k]_i| > \tau_c$. Since $k \in K_3$, it follows from equation (3.10) that

$$[c_k]_i \geq -\eta_k \geq -\tau_c \tag{3.14}$$

and, therefore,

$$[c_k]_i > \tau_c \geq \varepsilon_k^{\rm P}, \tag{3.15}$$

where the second inequality follows from equation (3.10). Condition (3.1b) and equation (3.10) then imply

$$|[y_k]_i| = \sigma_b |[\pi_k]_i| \leq \varepsilon_k^{\rm C} \leq \tau_c. \tag{3.16}$$

This verifies condition (3.5d). We have shown that the vector-pair (x_k, y_k) is an approximate solution to problem (NP) with tolerances τ_p, τ_d, and τ_c for all $k \in K_3$. This is a contradiction and, therefore, the algorithm must terminate with an approximate solution to problem (NP) in a finite number of iterations. □

We close this section by mentioning two potential drawbacks associated with using Algorithm 3.1. First, if the initial penalty parameter is substantially smaller than the threshold value required to guarantee convergence (see Theorem 3.1), then Algorithm 3.1 may be laborious since it is based on computing a *sequence* of approximate minimizers of the merit function. We also note that when the penalty parameter is too small, the merit function may not even have a well-defined minimizer [5, Example 1]. Second, even if the merit function does have a well-defined minimizer, there may not exist a strictly decreasing path that connects a poor initial point x_0 to this minimizer [5, Example 2]. A possible way of avoiding these situations is to *dynamically* update the penalty parameter based on linear infeasibility. The so-called "steering" method is based on this idea and has been

studied by Byrd et al. [4, 5]. Their algorithm is composed of essentially two stages that we now briefly describe using our notation. If we denote the current penalty parameter by σ_{C}, then the first stage is to compute a step s_∞ that locally minimizes the linearized constraint violation; this can be viewed as essentially solving the predictor subproblem with penalty parameter $\sigma = \infty$. The second stage is to compute a predictor step s_k^{P} and a new penalty parameter σ_{N} that satisfy the following conditions: (i) the decrease in the linearized constraint violation obtained from s_k^{P} must be at least a fixed multiple of the decrease obtained from s_∞; and (ii) the decrease in the faithful model must respect the progress made by s_k^{P} on the linearized infeasibility by satisfying

$$\Delta M_k^B(s_k^{\text{P}}) \geq \varepsilon_a \sigma_{\text{N}}(\|[c_k]^-\|_1 - \|[c_k + J_k s_k^{\text{P}}]^-\|_1), \tag{3.17}$$

where the constant ε_a satisfies $0 < \varepsilon_a < 1$ (note that $\Delta M_k^B(s_k)$ depends on σ_{N} although the notation does not make this explicit). The authors present three compelling examples that elucidate the strengths of this approach. For this approach to be beneficial, however, the additional cost must be offset by the "superior" values for the penalty parameter. This *dynamic* strategy is used in an SLQP method that is part of the KNITRO software package [41] and the authors report results that are superior to static penalty updating strategies.

We take the stance that both approaches should be available to the user. If the user has no information about the size of the multipliers, then our experience has been that steering is generally superior to finding a sequence of approximate minimizers. However, if a reasonable estimate for the size of the multipliers is known in advance, then steering is likely to be less efficient because of the potential overhead associated with the method.

4. Local convergence

This section considers the local convergence properties of Algorithm 4.1, which is a non-monotone implementation of Algorithm 1.1. The update to the Lagrange multiplier vector y_k is now critical and we must consider the sequence of vector-pairs (x_k, y_k). To simplify notation, we let w denote the combined x and y vectors, i.e., $w = (x, y)$, and we write $w_k = (x_k, y_k)$ for the current estimate of a solution $w^* = (x^*, y^*)$, $w_k^{\text{P}} = (x_k^{\text{P}}, y_k^{\text{P}})$ for the solution to the predictor subproblem (1.5), and $w_k^{\text{S}} = (x_k^{\text{S}}, y_k^{\text{S}})$ for the solution of the SQP subproblem (the precise definition of y_k^{S} depends on which SQP subproblem is used).

The primary result of this section is that if a successful iterate of Algorithm 4.1 gets close enough to a local minimizer w^* of problem (NP) that satisfies the strong second-order sufficient conditions, then the sequence of iterates converges to w^* with convergence properties derived from Newton's Method for zero-finding applied to the function

$$F_N(x, y_{\mathcal{A}}) = \begin{pmatrix} g(x) - J_{\mathcal{A}}(x)^T y_{\mathcal{A}} \\ c_{\mathcal{A}}(x) \end{pmatrix}. \tag{4.1}$$

We accomplish this by first showing that if w_k is close enough to w^*, then the predictor step accurately predicts the optimal active set and that the trust-region constraint is inactive.

We then show that specific SQP steps also identify the optimal active set and that their associated trust-region constraints are inactive. Since these steps are then equivalent to one step of Newton's Method for zero-finding applied to F_N, we deduce that w_{k+1} is closer to w^* than was w_k. This process is then repeated and results in the value w_{k+2}. Since Algorithm 4.1 is a nonmonotone approach, the analysis given by Conn, Gould and Toint [9, Section 15.3.2] shows that the ℓ_1-merit function will accept the value x_{k+2} and it follows that convergence may be described using classical results for Newton's Method applied to the function F_N.

Algorithm 4.1 generates trial steps in exactly the same way as does Algorithm 1.1. In fact, if every iteration is successful, then the two algorithms are identical. However, if a failure occurs then Algorithm 4.1 still accepts the step with the hope that the next iterate will be successful; we say that a "nonmonotone phase" has been entered. If we enter a nonmonotone phase, the ratio r_k of actual to predicted decrease in the merit function is computed based on the trial point $x_k + s_k$ and the *best-known* point, i.e., the solution estimate directly *before* the nonmonotone phase was entered. If the number of consecutive failures reaches the maximum number allowed (as denoted by the parameter *max_fails*), then the algorithm reverts to the best-known point, reduces the predictor trust-region radius, and then tries again. In less precise terms, the algorithm has "gone back in time" and proceeds as if we were using Algorithm 1.1 until the next failure occurs.

We have also changed the update to the predictor trust-region radius. The new update ensures that the radius following every successful/very successful iteration is at least as large as some pre-defined positive number Δ_{R}. We will see that this strategy allows us to prove that the trust-region constraints are eventually inactive; more complicated alternatives are briefly outlined in [9, Chapter 15].

One final modification to Algorithm 1.1 is the introduction of the vector y_k^{F}. Lemma 4.6 shows that we may choose y_k^{F} to be the multipliers from the predictor step, but any estimate satisfying the conditions $y_k^{\text{F}} - y^* = O(\|x_k - x^*\|_2)$ and $[y_k^{\text{F}}]_{\mathcal{I}} = 0$ may be used. We then define H_k to be any symmetric approximation to $H(x_k, y_k^{\text{F}})$, but for the local convergence results given by Theorems 4.1 and 4.2 we choose $H_k \equiv H(x_k, y_k^{\text{F}})$.

It may easily be verified that [19, Theorems 4.3, 4.4, and 4.7] are still true with these changes. Thus, Algorithm 4.1 is globally convergent.

Algorithm 4.1. Nonmonotone algorithm.
Input: (x_0, y_0)
Set parameters $0 < \eta_S \leq \eta_{VS} < 1$, $0 < \Delta_{\text{R}} \leq \Delta_{\text{U}}$, $0 < \eta \leq \eta_{\text{ACP}} < 1$, $\tau_f \geq 1$, and $0 \leq$ *max_fails* $\in \mathbb{N}$.
Set expansion and contraction factors $0 < \eta_c < 1 < \eta_e$ and fail counter *fails* $= 0$.
$k \leftarrow 0$
do
  Evaluate f_k, g_k, c_k, J_k and then compute ϕ_k.
  Define B_k to be a symmetric positive definite approximation to $H(x_k, y_k)$.
  Solve problem (1.4) for predictor step and multipliers $(s_k^{\text{P}}, y_k^{\text{P}})$.

Define y_k^{F} to be any multiplier estimate for which $y_k^{\mathrm{F}} - y^* = O(\|x_k - x^*\|_2)$ and $[y_k^{\mathrm{F}}]_{\mathcal{I}} = 0$.
Define H_k to be a symmetric approximation to $H(x_k, y_k^{\mathrm{F}})$.
Solve problem (1.6) for s_k^{CP} and compute $\Delta M_k^H(s_k^{\mathrm{CP}})$.
Compute an SQP step and multipliers $(s_k^{\mathrm{S}}, y_k^{\mathrm{S}})$ (optional).
Define a full step s_k that satisfies condition (1.7).
Evaluate $\phi(x_k + s_k)$ and $\Delta M_k^H(s_k)$.
if *fails* $= 0$ **then**
  $r_k \leftarrow \big(\phi(x_k) - \phi(x_k + s_k)\big)/\Delta M_k^H(s_k)$ [standard definition]
else
  $r_k \leftarrow \big(\phi_{\mathrm{R}} - \phi(x_k + s_k)\big)/\Delta_{\mathrm{R}}^{\mathrm{H}}$ [change in ϕ based on point x_{R}]
end if
if $r_k \geq \eta_{VS}$ **then** [very successful]
  $x_{k+1} \leftarrow x_k + s_k$
  $y_{k+1} \leftarrow y_k^{\mathrm{S}}$ ($y_{k+1} \leftarrow y_k^{\mathrm{F}}$ if SQP step not computed)
  $\Delta_{k+1}^{\mathrm{P}} \leftarrow \min\big(\max(\eta_e \cdot \Delta_k^{\mathrm{P}}, \Delta_{\mathrm{R}}), \Delta_{\mathrm{U}}\big)$ [increase Δ_k^{P} and ensure $\Delta_k^{\mathrm{P}} \geq \Delta_{\mathrm{R}}$]
  fails $\leftarrow 0$
else if $r_k \geq \eta_S$ **then** [successful]
  $x_{k+1} \leftarrow x_k + s_k$
  $y_{k+1} \leftarrow y_k^{\mathrm{S}}$ ($y_{k+1} \leftarrow y_k^{\mathrm{F}}$ if SQP step not computed)
  $\Delta_{k+1}^{\mathrm{P}} \leftarrow \max(\Delta_k^{\mathrm{P}}, \Delta_{\mathrm{R}})$ [ensure Δ_k^{P} is bigger than Δ_{R}]
  fails $\leftarrow 0$
else [failure]
  if *fails* $\leq$ *max_fails* **then**
    if *fails* $= 0$ **then** [save current point]
      $x_{\mathrm{R}} \leftarrow x_k,\ y_{\mathrm{R}} \leftarrow y_k,\ \phi_{\mathrm{R}} \leftarrow \phi_k,\ \Delta_{\mathrm{R}}^{\mathrm{H}} \leftarrow \Delta M_k^H(s_k)$
      $\Delta_{\mathrm{R}}^{\mathrm{P}} \leftarrow \Delta_k^{\mathrm{P}}$
      $\Delta_{k+1}^{\mathrm{P}} \leftarrow \eta_c \Delta_k^{\mathrm{P}}$ (optional)
    end if
    $x_{k+1} \leftarrow x_k + s_k$
    $\Delta_{k+1}^{\mathrm{P}} \leftarrow \Delta_k^{\mathrm{P}}$
    fails $\leftarrow$ *fails* $+ 1$
  else [revert to saved point]
    $x_{k+1} \leftarrow x_{\mathrm{R}},\ y_{k+1} \leftarrow y_{\mathrm{R}}$
    $\Delta_{k+1}^{\mathrm{P}} \leftarrow \eta_c \Delta_{\mathrm{R}}^{\mathrm{P}}$ [decrease Δ_k^{P}]
    fails $\leftarrow 0$
  end if
end if
$\Delta_{k+1}^{\mathrm{S}} \leftarrow \tau_f \cdot \Delta_{k+1}^{\mathrm{P}}$ [update SQP radius]
$k \leftarrow k + 1$
end do

4.1. Optimal active set identification

The analysis that ensues requires a notion of "uniformity" for the underlying KKT systems within a neighborhood of a solution w^*. This is generally not an an issue for systems involving H_k since it is reasonable to expect that if w_k converges to w^* then H_k will converge to H^*; this certainly occurs if $H_k \equiv H(x_k, y_k)$ or $H_k \equiv H(x_k, y_k^{\rm F})$. A similar statement does not hold for systems involving B_k since B_k is generally not a continuous function of w. Moreover, we certainly can not expect the positive-definite matrix B_k to converge to H^* since H^* is normally indefinite. The optimality conditions for problem (NP) suggest that we need the matrices H_k and B_k to be positive definite when restricted to the null space of the active constraints (note that B_k is positive definite by construction); this is essentially the uniformity that we need. To develop a general framework, we define the following sets that depend on the minimizer w^*:

$$S(x\,;x^*) = \{M \in \mathbb{R}^{n\times n} : M = M^T,\ \|M\|_2 \leq \beta_{\max} \ \text{ and } \ s^TMs \geq \lambda_{\min} s^Ts \ \ \forall\, s \text{ satisfying } J_{\mathcal{A}}(x)s = 0\} \tag{4.2}$$

and

$$S_\varepsilon = \cup_{w\in\bar{\mathcal{B}}_\varepsilon(w^*)} S(x\,;x^*) \tag{4.3}$$

for given real numbers $\beta_{\max}$ and $\lambda_{\min} > 0$. Using this definition, we now state a result that supplies the required uniformity.

Lemma 4.1. *If w^* is a KKT point for problem* (NP) *that satisfies the* LICQ*, then*

(i) *for any $0 \leq \varepsilon < \infty$ the set*

$$S_\varepsilon = \cup_{w\in\bar{\mathcal{B}}_\varepsilon(w^*)} S(x\,;x^*)$$

is compact;

(ii) *if $\nu_1 \leq \nu_2$, then $S_{\nu_1} \subseteq S_{\nu_2}$;*

(iii) *there exists a positive number ε_1 such that if $w \in \mathcal{B}_{\varepsilon_1}(w^*)$ and $M \in S_{\varepsilon_1}$, then $s^TMs \geq (\lambda_{\min}/2)s^Ts$ for all s satisfying $J_{\mathcal{A}}(x)s = 0$.*

If in addition, strict complementarity holds at w^*, then

(iv) *there exists a positive number ε_2 such that $\varepsilon_2 \leq \varepsilon_1$ and numbers $\beta_0 > 0$ and $\beta > 0$ such that if $w \in \mathcal{B}_{\varepsilon_2}(w^*)$ and $M \in S_{\varepsilon_2}$, then $J_{\mathcal{A}}(x)$ has full row rank, $c_{\mathcal{I}}(x) > 0$, $y_{\mathcal{A}} > 0$, and the matrices*

$$\bar{K}_M(x) = \begin{pmatrix} M & J_{\mathcal{A}}(x)^T \\ J_{\mathcal{A}}(x) & 0 \end{pmatrix} \ \text{ and } \ K_M(w) = \begin{pmatrix} M & -J_{\mathcal{A}}(x)^T & -J_{\mathcal{I}}(x)^T \\ \operatorname{diag}(y_{\mathcal{A}})J_{\mathcal{A}}(x) & 0 & 0 \\ 0 & 0 & \operatorname{diag}(c_{\mathcal{I}}) \end{pmatrix}$$

are nonsingular and satisfy

$$\|\bar{K}_M(x)^{-1}\|_2 \leq \beta_0 \ \text{ and} \tag{4.4a}$$

$$\|K_M(w)^{-1}\|_2 \leq \beta; \tag{4.4b}$$

(v) *if* $w \in \mathcal{B}_{\varepsilon_2}(w^*)$ *and* $M \in S_{\varepsilon_2}$, *then it follows that*

$$s = O(\|x - x^*\|_2), \quad \pi_{\mathcal{A}} - y^*_{\mathcal{A}} = O(\|x - x^*\|_2), \quad \text{and} \quad \pi - y^* = O(\|x - x^*\|_2), \tag{4.5}$$

where s *and* $\pi_{\mathcal{A}}$ *satisfy*

$$\bar{K}_M(x) \begin{pmatrix} s \\ -\pi_{\mathcal{A}} \end{pmatrix} \equiv \begin{pmatrix} M & J_{\mathcal{A}}(x)^T \\ J_{\mathcal{A}}(x) & 0 \end{pmatrix} \begin{pmatrix} s \\ -\pi_{\mathcal{A}} \end{pmatrix} = - \begin{pmatrix} g(x) \\ c_{\mathcal{A}}(x) \end{pmatrix}, \tag{4.6}$$

and π *is obtained from* $\pi_{\mathcal{A}}$ *by "scattering" the components of* $\pi_{\mathcal{A}}$ *into a zero-vector of length* m *as indicated by* $\mathcal{A}$.

Proof. We first prove part (i). Since it is clear that S_ε is bounded, we only show that S_ε is closed. Let $\{M_k\}$ be a sequence in S_ε such that $\lim_{k\to\infty} M_k = \bar{M}$. This implies the existence of a sequence $\{w_k\} \in \bar{\mathcal{B}}_\varepsilon(w^*)$ such that $M_k \in S(x_k; x^*)$ and $v^T M_k v \geq \lambda_{\min} v^T v$ for all v such that $J_{\mathcal{A}}(x_k)v = 0$. The set $\bar{\mathcal{B}}_\varepsilon(w^*)$ is compact and, therefore, we can pass to a subsequence K_1 such that $\lim_{k\in K_1} w_k = \bar{w} \in \bar{\mathcal{B}}_\varepsilon(w^*)$. Since J is continuous and $J_{\mathcal{A}}(x^*)$ has full row rank, [6, Theorem 2.3] implies the existence of a locally *continuous* null space basis function $Z(\cdot)$ such that $J_{\mathcal{A}}(x_k)Z(x_k) = 0$, $\lim_{k\in K_1} Z(x_k) = \bar{Z}$, and $J_{\mathcal{A}}(\bar{x})\bar{Z} = 0$. This implies that $Z(x_k)^T M_k Z(x_k) \succeq \lambda_{\min}$ and upon taking limits that $\bar{Z}^T \bar{M} \bar{Z} \succeq \lambda_{\min}$. Since it is clear that $\bar{M}$ is symmetric and satisfies $\|\bar{M}\|_2 \leq \beta_{\max}$, we have $\bar{M} \in S(\bar{x}\,; x^*) \subseteq S_\varepsilon$. Thus, S_ε is closed.

Part (ii) follows immediately from the definitions of S_{ν_1} and S_{ν_2}.

We now prove part (iii). If part (iii) was not true, then there would exist a monotonically decreasing and strictly positive sequence $\{\delta_k\} \to 0$ and associated sequences $\{w_k\}$, $\{s_k\}$, and $\{M_k\}$ such that $w_k \in \mathcal{B}_{\delta_k}(w^*)$, $M_k \in S_{\delta_k} \subseteq S_{\delta_1}$, $J_{\mathcal{A}}(x_k)s_k = 0$, $\|s_k\|_2 = 1$ and $s_k^T M_k s_k < \lambda_{\min}/2$. It follows from these properties, part (i), and the fact that the sequence $\{s_k\}$ belongs to a compact set, that there exists a subsequence K_2, a matrix $M^* \in S_{\delta_1}$ and a unit vector s^* such that

$$\lim_{k\in K_2} w_k = w^*, \quad \lim_{k\in K_2} M_k = M^*, \quad \lim_{k\in K_2} s_k = s^*, \quad J^*_{\mathcal{A}} s^* = 0, \quad \text{and} \quad s^{*T} M^* s^* \leq \lambda_{\min}/2. \tag{4.7}$$

Since $M_k \in S_{\delta_k}$ and $\{\delta_k\} \to 0$, there also exists a sequence $\{\widehat{x}_k\} \to x^*$ such that $s^T M_k s \geq \lambda_{\min} s^T s$ for all s satisfying $J_{\mathcal{A}}(\widehat{x}_k)s = 0$. Using the same argument as in the first paragraph of this proof, we find that $Z^{*T} M^* Z^* \succeq \lambda_{\min}$, where the columns of Z^* form a basis for the null space of $J^*_{\mathcal{A}}$. This contradicts (4.7) and thus (iii) must be true.

To show that part (iv) holds, we first note that strict complementarity and the LICQ imply that there exists a number ε_s such that $0 < \varepsilon_s \leq \varepsilon_1$ and

$$c_i(x) \geq \tfrac{1}{2} c_i^* > 0 \ \text{ for } i \in \mathcal{I} \ , \quad y_i \geq \tfrac{1}{2} y_i^* > 0 \ \text{ for } i \in \mathcal{A} \ , \ \text{ and } J_{\mathcal{A}}(x) \ \text{ has full row rank} \tag{4.8}$$

for all $w \in \mathcal{B}_{\varepsilon_s}(w^*)$. Under the current assumptions, it follows from parts (ii), (iii) and [1, Lemma 1.27] that

$$\text{the matrix } \bar{K}_M(x) \text{ is nonsingular for all } w \in \mathcal{B}_{\varepsilon_s}(w^*) \text{ and } M \in S_{\varepsilon_s}. \tag{4.9}$$

Assume that (4.4a) does not hold for any $\varepsilon_2 \leq \varepsilon_s$ so that there exists a monotonically decreasing sequence $\{\delta_k\} \to 0$ such that $0 < \delta_k \leq \varepsilon_s$, and associated sequences $\{w_k\} \in \mathcal{B}_{\delta_k}(w^*)$ and $\{M_k\} \in S_{\delta_k} \subseteq S_{\varepsilon_s}$ such that

$$\|\bar{K}_{M_k}(x_k)^{-1}\|_2 \geq k \quad \text{for all } k \geq 0. \tag{4.10}$$

Since $\{\delta_k\} \to 0$ and S_{ε_s} is compact as a result of part (i), there exists a subsequence K_3 such that $\lim_{k\in K_3} w_k = w^*$ and $\lim_{k\in K_3} M_k = M^* \in S_{\varepsilon_s}$. It then follows from (4.9) that $\bar{K}_{M^*}(x^*)$ is nonsingular. Since $\lim_{k\in K_3} \bar{K}_{M_k}(x_k) = \bar{K}_{M^*}(x^*)$, [16, Theorem 8.64] implies that the singular values of $\bar{K}_{M_k}(x_k)$ are uniformly bounded away from zero for $k \in K_3$ sufficiently large. Therefore, $\|\bar{K}_{M_k}(x_k)^{-1}\|_2$ must be bounded above for all $k \in K_3$, which contradicts (4.10). Thus, (4.4a) holds for some $\varepsilon_1 \geq \varepsilon_s \geq \varepsilon_2 > 0$ and $\beta_0 > 0$. It also follows from (4.8) that $J_{\mathcal{A}}(x)$ has full row rank, $c_{\mathcal{I}}(x) > 0$, and $y_{\mathcal{A}} > 0$ for all $w \in \mathcal{B}_{\varepsilon_2}(w^*)$.

We now show that equation (4.4b) holds for ε_2. Let $w \in \mathcal{B}_{\varepsilon_2}(w^*)$ and $M \in S_{\varepsilon_2}$. Equation (4.8) implies that the matrices

$$N_F = \begin{pmatrix} I & 0 & J_{\mathcal{I}}(x)^T \operatorname{diag}\left(c_{\mathcal{I}}(x)\right)^{-1} \\ 0 & \operatorname{diag}(y_{\mathcal{A}})^{-1} & 0 \\ 0 & 0 & I \end{pmatrix}, \quad N_M = \begin{pmatrix} M & J_{\mathcal{A}}(x)^T & 0 \\ J_{\mathcal{A}}(x) & 0 & 0 \\ 0 & 0 & \operatorname{diag}\left(c_{\mathcal{I}}(x)\right) \end{pmatrix},$$

and $N_S = \operatorname{diag}(I, -I, I)$ are *nonsingular*; they satisfy $N_F K_M(w) N_S = N_M$ so that $\|K_M(w)^{-1}\|_2 \leq \|N_M^{-1}\|_2 \|N_F\|_2$. It is also clear from equation (4.8) that the quantity $\|N_F\|_2$ is bounded for all $w \in \mathcal{B}_{\varepsilon_2}(w^*)$, so to bound $\|K_M(w)^{-1}\|_2$ we must bound $\|N_M^{-1}\|_2$, but it is sufficient to bound $\|\bar{K}_M(x)^{-1}\|_2$ due to equation (4.8). The result follows from equation (4.4a) and, therefore, there exists a number $\beta > 0$ such that $\|K_M(w)^{-1}\|_2 \leq \beta$ for all $w \in \mathcal{B}_{\varepsilon_2}(w^*)$ and $M \in S_{\varepsilon_2}$.

Finally, we prove part (v). Let $w_k \in \mathcal{B}_{\varepsilon_2}(w^*)$, and $M \in S_{\varepsilon_2}$. Since $c_{\mathcal{A}}^* = 0$, it follows that system (4.6) is equivalent to

$$\begin{pmatrix} M & J_{\mathcal{A}}(x)^T \\ J_{\mathcal{A}}(x) & 0 \end{pmatrix} \begin{pmatrix} s \\ y_{\mathcal{A}}^* - \pi_{\mathcal{A}} \end{pmatrix} = - \begin{pmatrix} g(x) - J_{\mathcal{A}}(x)^T y_{\mathcal{A}}^* \\ c_{\mathcal{A}}(x) - c_{\mathcal{A}}^* \end{pmatrix}. \tag{4.11}$$

Equation (4.4a), norm inequalities, and Taylor expansions for $g(x)$, $c_{\mathcal{A}}(x)$, and $J_{\mathcal{A}}(x)$ at the point x^* yield $s = O(\|x - x^*\|_2)$ and $\pi_{\mathcal{A}} - y_{\mathcal{A}}^* = O(\|x - x^*\|_2)$. The fact that $\pi - y^* = O(\|x - x^*\|_2)$ follows since $\pi_{\mathcal{I}} = 0$ by construction and $y_{\mathcal{I}}^* = 0$ from the optimality conditions for problem (NP). □

Our next aim is to prove a result concerning active set identification. Given a vector w, we define the function

$$F_{\text{KKT}}(w) = \begin{pmatrix} g(x) - J(x)^T y \\ c(x) \cdot y \end{pmatrix}. \tag{4.12}$$

Lemma 4.2. *Let w^* be a solution to problem* (NP) *that satisfies strict complementarity and the* LICQ. *Then there exist numbers $\mu > 0$ and $\beta > 0$ such that if $w_k \in \mathcal{B}_{\mu/2}(w^*)$, $M \in$*

$S_{\mu/2}$, and $4\beta\|F_{KKT}(w_k)\|_2 \le \mu$ then there exists a unique closest minimizer $\big(x_k(M), y_k(M)\big) = w_k(M)$ to the point w_k for the problem

$$\begin{array}{ll} \underset{x\in\mathbb{R}^n}{\text{minimize}} & \frac{1}{2}(x-x_k)^T M(x-x_k) + g_k^T(x-x_k) \\ \text{subject to} & c_k + J_k(x-x_k) \ge 0 \end{array} \tag{4.13}$$

with the following properties:

(i) $\|x_k(M) - x_k\|_\infty \le \|w_k(M) - w_k\|_2 \le 2\beta\|F_{KKT}(w_k)\|_2$;

(ii) *the set of constraints active at $x_k(M)$ for problem (4.13) are the same as the indices in $\mathcal{A}$;*

(iii) *the solution $w_k(M)$ satisfies strict complementarity; and*

(iv) *$J_{\mathcal{A}}(x_k)$ has full row rank.*

Proof. We begin by letting $\varepsilon_1 \ge \varepsilon_2 > 0$ and $\beta > 0$ be the constants guaranteed by Lemma 4.1. Given any vector-pair $(w, \bar{w})$ and symmetric matrix M, we define

$$F_M(w\,;\bar{w}) = \begin{pmatrix} M(x-\bar{x}) + g(\bar{x}) - J(\bar{x})^T y \\ \big(c(\bar{x}) + J(\bar{x})(x-\bar{x})\big) \cdot y \end{pmatrix}. \tag{4.14}$$

Differentiating (4.14) we have

$$F'_M(w\,;\bar{w}) = \begin{pmatrix} M & -J(\bar{x})^T \\ \operatorname{diag}(y)J(\bar{x}) & \operatorname{diag}\big(c(\bar{x}) + J(\bar{x})(x-\bar{x})\big) \end{pmatrix}.$$

Choosing $(w, \bar{w}) = (w^*, w^*)$ we have

$$F'_M(w^*\,;w^*) = \begin{pmatrix} M & -J_{\mathcal{A}}^{*T} & -J_{\mathcal{I}}^{*T} \\ \operatorname{diag}(y_{\mathcal{A}}^*)J_{\mathcal{A}}^* & 0 & 0 \\ 0 & 0 & \operatorname{diag}(c_{\mathcal{I}}^*) \end{pmatrix}$$

since optimality conditions at w^* imply $c_{\mathcal{A}}^* = 0$ and $y_{\mathcal{I}}^* = 0$. It follows from (4.4a) with the choice $w = w^*$ that the matrix $F'_M(w^*\,;w^*)$ is nonsingular and satisfies

$$\|F'_M(w^*\,;w^*)^{-1}\| \le \beta \ \text{ for all } M \in S_{\varepsilon_2}. \tag{4.15}$$

Next, choose a number μ such that $0 < \mu \le \varepsilon_2$ and if w and $\bar{w}$ are contained in $B_\mu(w^*)$, then the following conditions are satisfied:

C1. if $c_i^* > 0$ then $[c(\bar{x}) + J_k(x-\bar{x})]_i > 0$;

C2. if $y_i^* > 0$ then $y_i > 0$;

C3. $\|F'_M(w\,;\bar{w}) - F'_M(w^*\,;w^*)\|_2 \le 1/(2\beta)$ (this estimate holds for all M);

Let $w_k \in \mathcal{B}_{\mu/2}(w^*)$ and $M \in S_{\mu/2}$. Since $\mu < \mu/2 \leq \varepsilon_2 \leq \varepsilon_1$, it follows from parts (ii) and (iii) of Lemma 4.1 that $J_{\mathcal{A}}(x_k)$ has full row rank and that estimate (4.15) holds for M; thus part (iv) is true. Using the argument by Robinson [35, Lemma 1], we now show that $F_M(w\,;w_k)$ has a unique zero in $\bar{\mathcal{B}}_{\mu/2}(w_k)$. Note that

$$\bar{\mathcal{B}}_{\mu/2}(w_k) \subset \mathcal{B}_{\mu}(w^*) \tag{4.16}$$

since if $w \in \bar{\mathcal{B}}_{\mu/2}(w_k)$ then

$$\|w - w^*\|_2 \leq \|w - w_k\|_2 + \|w_k - w^*\|_2 < \mu/2 + \mu/2 \leq \mu.$$

Define the function

$$T_M(w) = w - F_M'(w^*\,;w^*)^{-1} F_M(w\,;w_k) \tag{4.17}$$

so that

$$T_M'(w) = I - F_M'(w^*\,;w^*)^{-1} F_M'(w\,;w_k) = F_M'(w^*\,;w^*)^{-1}\big(F_M'(w^*\,;w^*) - F_M'(w\,;w_k)\big).$$

It follows that

$$\|T_M'(w)\|_2 \leq \beta \|F_M'(w^*\,;w^*) - F_M'(w\,;w_k)\|_2 \leq \tfrac{1}{2} \quad \big(\text{use (4.15), (4.16), and C3}\big)$$

for all $w \in \bar{\mathcal{B}}_{\mu/2}(w_k)$, which implies that T_M is a contraction. It also follows that

$$\|T_M(w_k) - w_k\|_2 \leq \beta \|F_M(w_k\,;w_k)\|_2 \quad \big(\text{use (4.17) and (4.15)}\big). \tag{4.18}$$

Using the triangle inequality, the fact that $T_M(w)$ is a contraction with contraction factor $1/2$, equation (4.18), and the assumption that $4\beta\|F_M(w_k\,;w_k)\|_2 \leq \mu$, we have that for all $w \in \bar{\mathcal{B}}_{\mu/2}(w_k)$ the estimate

$$\|T_M(w) - w_k\|_2 \leq \|T_M(w) - T_M(w_k)\|_2 + \|T_M(w_k) - w_k\|_2 \leq \tfrac{1}{2}\|w - w_k\|_2 + \beta\|F_M(w_k\,;w_k)\|_2 \leq \mu/2,$$

which implies $T_M : \bar{\mathcal{B}}_{\mu/2}(w_k) \to \bar{\mathcal{B}}_{\mu/2}(w_k)$. We may now apply the well-known fixed point result [37, Theorem 9.23]), which states that T_M has a unique fixed point $w_k(M)$ in $\bar{\mathcal{B}}_{\mu/2}(w_k)$ and that

$$\begin{aligned} \|x_k(M) - x_k\|_\infty \leq \|x_k(M) - x_k\|_2 &\leq \|w_k(M) - w_k\|_2 \quad \text{(use norm inequalities)} \\ &\leq 2\|T_M(w_k) - w_k\|_2 \quad \text{(estimate from fixed-point theorem)} \\ &\leq 2\beta\|F_M(w_k\,;w_k)\|_2 \quad \text{(use (4.18))}, \end{aligned}$$

which proves part (i). Since $w_k(M)$ is a fixed point for $T_M(w)$, equation (4.17) implies that

$$F_M(w_k(M)\,;w_k) = 0. \tag{4.19}$$

Thus $w_k(M)$ satisfies the *equality* conditions for being a first-order KKT point for problem (4.13). We now show that the point $w_k(M)$ is actually a first-order KKT point for problem (4.13). Since $w_k(M) \in \bar{\mathcal{B}}_{\mu/2}(w_k) \subset \mathcal{B}_{\mu}(w^*)$, we may deduce the following: if $y_i^* > 0$

then C2 implies $[y_k(M)]_i > 0$ and then (4.19) implies $[c_k + J_k\big(x_k(M) - x_k\big)]_i = 0$; and if $c_i^* > 0$ then C1 implies $[c_k + J_k\big(x_k(M) - x_k\big)]_i > 0$ and then (4.19) implies $[y_k(M)]_i = 0$. Since strict complementarity holds at w^* by assumption, one of these two cases must hold and, therefore, $w_k(M)$ is a first-order KKT point for the problem (4.13) that satisfies strict complementarity and correctly identifies the optimal active set; this establishes parts (ii) and (iii). The fact that $x_k(M)$ is a minimizer follows from parts (ii) and (iii) of Lemma 4.1. Finally, $w_k(M)$ is the unique closest solution since any other solution would be a KKT point and, therefore, a zero of the function $F_M(w\,; w_k)$. However, $w_k(M)$ is the unique zero inside $\bar{\mathcal{B}}_{\mu/2}(w_k)$. □

4.2. Local descent properties

In this section we show that, in a neighborhood of a solution w^*, directions related to the traditional SQP step are descent directions for the underlying model functions; this result is critical for proving that Algorithm 4.1 has a fast rate of convergence. We use the following definition.

Definition 4.1. *Given a vector $v \in \mathbb{R}^n$ and a subspace $\mathcal{V} \subseteq \mathbb{R}^n$, we define*

$$\theta(v, \mathcal{V}) = \begin{cases} \tan^{-1}\left(\|v_{\scriptscriptstyle R}\|_2 / \|v_{\scriptscriptstyle N}\|_2\right) & \text{if } \|v_{\scriptscriptstyle N}\|_2 \neq 0; \\ \pi/2 & \text{otherwise.} \end{cases} \qquad (0 \leq \theta \leq \pi/2) \tag{4.20}$$

to be the angle between v and $\mathcal{V}$, where $v = v_{\scriptscriptstyle N} + v_{\scriptscriptstyle R}$ is the unique *orthogonal decomposition of v such that $v_{\scriptscriptstyle N} \in \mathcal{V}$ and $v_{\scriptscriptstyle R} \perp \mathcal{V}$.*

The next result essentially says how close a vector s must be to the null space of the active constraints to guarantee positive curvature in a neighborhood of a solution.

Lemma 4.3. *Let w^* be a solution to problem* (NP) *that satisfies the* LICQ. *Then, there exists a number $\varepsilon_2 > 0$ such that if w, s, and M satisfy $w \in \mathcal{B}_{\varepsilon_2}(w^*)$, $M \in S_{\varepsilon_2}$, and*

$$\theta\Big(s, \operatorname{null}\big(J_{\mathcal{A}}(x)\big)\Big) \leq \bar{\theta} \stackrel{\text{def}}{=} \min\left(\pi/4, \tan^{-1}\left(\frac{\lambda_{\min}}{24\beta_{max}}\right)\right), \tag{4.21}$$

then $s^T M s \geq (\lambda_{\min}/8) s^T s$.

Proof. Let ε_2 be defined as in part (iv) of Lemma 4.1 so that $J_{\mathcal{A}}(x)$ has full row-rank for all $w \in \mathcal{B}_{\varepsilon_2}(w^*)$. Suppose that $w \in \mathcal{B}_{\varepsilon_2}(w^*)$, $M \in S_{\varepsilon_2}$, and s satisfy (4.21). If we write $s = s_{\scriptscriptstyle N} + s_{\scriptscriptstyle R}$ for $s_{\scriptscriptstyle N} \in \operatorname{null}(J_{\mathcal{A}}(x))$ and $s_{\scriptscriptstyle R} \in \operatorname{range}(J_{\mathcal{A}}(x)^T)$, it follows from (4.20) and (4.21) that $\theta = \theta\Big(s, \operatorname{null}\big(J_{\mathcal{A}}(x)\big)\Big)$ satisfies

$$\frac{\|s_{\scriptscriptstyle R}\|_2}{\|s_{\scriptscriptstyle N}\|_2} = \tan(\theta) \leq 1. \tag{4.22}$$

Using the orthogonal decomposition of s, parts (ii) and (iii) of Lemma 4.1, the Cauchy-Schwarz inequality, definition of $\beta_{\max}$, and equations (4.22) and (4.21), we have

$$\begin{aligned}\frac{s^T M s}{s^T s} &= \frac{{s_{\rm N}}^T M s_{\rm N} + {s_{\rm R}}^T M s_{\rm R} + 2{s_{\rm N}}^T M s_{\rm R}}{\|s_{\rm N}\|_2^2 + \|s_{\rm R}\|_2^2} \\ &\geq \frac{(\lambda_{\min}/2){s_{\rm N}}^T s_{\rm N} - \beta_{\max}\|s_{\rm R}\|_2^2 - 2\beta_{\max}\|s_{\rm R}\|_2\|s_{\rm N}\|_2}{\|s_{\rm N}\|_2^2 + \|s_{\rm R}\|_2^2} \\ &\geq \lambda_{\min}/4 - \beta_{\max}\tan^2(\theta) - 2\beta_{\max}\tan(\theta) \\ &\geq \lambda_{\min}/4 - 3\beta_{\max}\tan(\theta) \geq \lambda_{\min}/8,\end{aligned}$$

which completes the proof. □

We now show that in the neighborhood of a solution w^*, the (unique) solution to

$$\underset{s\in\mathbb{R}^n}{\text{minimize}}\ \ g(x)^T s + \tfrac{1}{2}s^T M s \quad \text{subject to} \quad c_{\mathcal{A}}(x) + J_{\mathcal{A}}(x)s = 0 \tag{4.23}$$

satisfies a certain "descent" property for the underlying models (under certain assumptions).

Lemma 4.4. *Let w^* be a minimizer for problem* (NP) *that satisfies the* LICQ *and strict complementarity and suppose that $\sigma > \|y^*\|_\infty$. It follows that there exist positive numbers c_2 and ε_3 such that if $w \in \mathcal{B}_{\varepsilon_3}(w^*)$ and $M \in S_{\varepsilon_3}$ then problem (4.23) is well-defined and the solution s_T satisfies*

$$\big(g(x) + \sigma J(x)^T z\big)^T s_T < -c_2\|s_T\|_2^2 \quad \textit{for} \quad z = \begin{cases} 0 & \textit{if } c(x) \geq 0; \\ -1 & \textit{otherwise.}\end{cases} \tag{4.24}$$

Proof. Strict complementarity implies the existence of a scalar $\kappa_{\rm S} > 0$ such that

$$y_{\mathcal{A}}^* \geq \kappa_{\rm S} e > 0. \tag{4.25}$$

We define $\bar{\theta}$ as in Lemma 4.3 and choose positive scalars $\kappa_{\rm J}$ and ε_3 so that the following hold for all $w \in \mathcal{B}_{\varepsilon_3}(w^*)$ and $M \in S_{\varepsilon_3}$:

1. $\varepsilon_3 \leq \varepsilon_2$, where ε_2 is defined in Lemma 4.3;
2. $\|J(x)^T\|_2\|(J(x)J(x)^T)^{-1}\|_2 \leq \kappa_{\rm J}$;
3. the system
$$\begin{pmatrix} M & J_{\mathcal{A}}(x)^T \\ J_{\mathcal{A}}(x) & 0 \end{pmatrix}\begin{pmatrix} s \\ -q \end{pmatrix} = -\begin{pmatrix} g(x) - J_{\mathcal{A}}(x)^T y_{\mathcal{A}} \\ c_{\mathcal{A}}(x) \end{pmatrix} \tag{4.26}$$
has a unique solution (s, q) that satisfies
   a. $(\kappa_{\rm S}/2)e \leq y_{\mathcal{A}} + q \leq \sigma(1 - \kappa_\sigma)e$ for some $\kappa_\sigma > 0$;

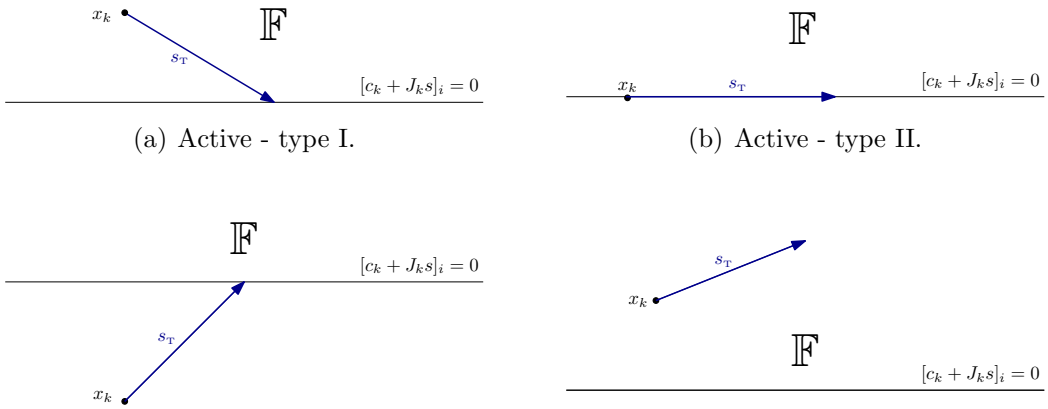
b. $\|s\|_2 \leq \min(1, c_1)$, where

$$c_1 = \frac{\kappa \sin(\bar{\theta})}{2\kappa_{\mathrm{J}}\beta_{\max}} > 0 \quad \text{and} \quad \kappa = \min\left(\frac{\kappa_{\mathrm{S}}}{2}, \sigma\kappa_\sigma\right) > 0; \quad \textit{and} \tag{4.27}$$

c. if $c_i^* > 0$ then $c_i(x) + \nabla c_i(x)^T s > 0$.

Condition 2 can be satisfied since $J_{\mathcal{A}}^*$ has full row rank. Condition 1 is well-defined since the assumptions of this theorem imply that the assumptions of Lemma 4.3 hold. Since $\varepsilon_3 \leq \varepsilon_2$, parts (ii), (iii) and (iv) of Lemma 4.1 combined with [1, Lemma 1.27] guarantee that problem (4.23) has a unique solution, say s_{T}, and the optimality conditions show that $(s_{\mathrm{T}}, q_{\mathrm{T}})$ satisfies system (4.26), where q_{T} is the *step* from y to the Lagrange multiplier vector for problem (4.23). Note that we can make the solution $(s_{\mathrm{T}}, q_{\mathrm{T}})$ arbitrarily small in norm since the target vector in system (4.26) converges to zero as w converges to w^*. This observation, equation (4.25), and the assumption $\sigma > \|y^*\|_\infty$, guarantee that we can satisfy conditions 3a and 3b for some $\kappa_\sigma > 0$.

Now let $w \in \mathcal{B}_{\varepsilon_3}(w^*)$, $M \in S_{\varepsilon_3}$, and $(s_{\mathrm{T}}, q_{\mathrm{T}})$ denote the solution to problem (4.23) so that it satisfies system (4.26). For convenience we "scatter" the vector q_{T}, which has length equal to the size of the indexing set $\mathcal{A}$, into a vector $q_{\mathrm{TF}} \in \mathbb{R}^m$ so that $[q_{\mathrm{TF}}]_i = 0$ if $i \notin \mathcal{A}$. We also partition the constraints up into four types: I, II, III, and IV (see Figure 4.1); 3a and the properties of s_{T} guarantee that these are the only possibilities.



(a) Active - type I.

(b) Active - type II.

(c) Active - type III.

(d) Inactive - type IV.

Figure 4.1: The only four possibilities in a small enough neighborhood of the solution w^*. (a) For type I, we have $c_i(x_k) > 0$, $c_i^* = 0$, and $\nabla c_i(x_k)^T s_{\mathrm{T}} < 0$. (b) For type II, we have $c_i(x_k) = 0$, $c_i^* = 0$, and $\nabla c_i(x_k)^T s_{\mathrm{T}} = 0$. (c) For type III, we have $c_i(x_k) < 0$, $c_i^* = 0$, and $\nabla c_i(x_k)^T s_{\mathrm{T}} > 0$. (d) For type IV, we have $c_i^* > 0$.

Note that $\nabla c_i(x)^T s_{\mathrm{T}} < 0$ for $i \in$ I, $\nabla c_i(x)^T s_{\mathrm{T}} = 0$ for $i \in$ II and $\nabla c_i(x)^T s_{\mathrm{T}} > 0$ for $i \in$ III. It then follows from system (4.26), the definitions of q_{TF} and z, condition 3a, and the

definition of κ that

$$\begin{aligned}\big(g(x)+\sigma J(x)^T z\big)^T s_{\rm T} &= -{s_{\rm T}}^T M s_{\rm T} + (J_{\mathcal{A}}(x)s_{\rm T})^T[y+q_{\rm TF}]_{\mathcal{A}} + \sigma z^T(J(x)s_{\rm T})\\ &= -{s_{\rm T}}^T M s_{\rm T} + \sum_{i\in{\rm I}}(\nabla c_i(x)^T s_{\rm T})[y+q_{\rm TF}]_i + \sum_{i\in{\rm III}}(\nabla c_i(x)^T s_{\rm T})[y+q_{\rm TF}-\sigma e]_i\\ &\le -{s_{\rm T}}^T M s_{\rm T} + \frac{\kappa_{\rm S}}{2}\sum_{i\in{\rm I}}(\nabla c_i(x)^T s_{\rm T}) - \sigma\kappa_\sigma\sum_{i\in{\rm III}}(\nabla c_i(x)^T s_{\rm T})\\ &\le -{s_{\rm T}}^T M s_{\rm T} - \kappa\sum_{i\in{\rm I}\cup{\rm II}\cup{\rm III}}|\nabla c_i(x)^T s_{\rm T}| = -{s_{\rm T}}^T M s_{\rm T} - \kappa\|J_{\mathcal{A}}(x)s_{\rm T}\|_1.\end{aligned} \tag{4.28}$$

We now develop a lower bound on $\|J_{\mathcal{A}}(x)s_{\rm T}\|_1$.

Let $s_{\rm T} = s_{\rm T}^{\rm R} + s_{\rm T}^{\rm N}$ be the orthogonal decomposition of $s_{\rm T}$ such that $s_{\rm T}^{\rm R} \in \text{range}(J_{\mathcal{A}}(x)^T)$ and $s_{\rm T}^{\rm N} \in \text{null}(J_{\mathcal{A}}(x))$. It follows that there exists a vector r such that $J_{\mathcal{A}}(x)^T r = s_{\rm T}^{\rm R}$ and, therefore,

$$\|s_{\rm T}^{\rm R}\|_2 \le \big\|J_{\mathcal{A}}(x)^T\big\|_2\,\|r\|_2 \quad\text{and}\quad J_{\mathcal{A}}(x)s_{\rm T} = J_{\mathcal{A}}(x)s_{\rm T}^{\rm R} = J_{\mathcal{A}}(x)J_{\mathcal{A}}(x)^T r. \tag{4.29}$$

Using the nonsingularity of $J_{\mathcal{A}}(x)J_{\mathcal{A}}(x)^T$ and norm inequalities, we have

$$\|r\|_2 \le \Big\|\big(J_{\mathcal{A}}(x)J_{\mathcal{A}}(x)^T\big)^{-1}\Big\|_2\,\|J_{\mathcal{A}}(x)s_{\rm T}\|_2. \tag{4.30}$$

This inequality, equation (4.29) and condition 2, imply

$$\|J_{\mathcal{A}}(x)s_{\rm T}\|_2 \ge \frac{\|r\|_2}{\Big\|\big(J_{\mathcal{A}}(x)J_{\mathcal{A}}(x)^T\big)^{-1}\Big\|_2} \ge \frac{\|s_{\rm T}^{\rm R}\|_2}{\Big\|\big(J_{\mathcal{A}}(x)J_{\mathcal{A}}(x)^T\big)^{-1}\Big\|_2\,\|J_{\mathcal{A}}(x)^T\|_2} \ge \frac{\|s_{\rm T}^{\rm R}\|_2}{\kappa_{\rm J}}. \tag{4.31}$$

Using this inequality, norm inequalities, and the fact that $\|s_{\rm T}^{\rm R}\|_2 = \sin(\theta)\|s_{\rm T}\|_2$, we have

$$\|J_{\mathcal{A}}(x)s_{\rm T}\|_1 \ge \|J_{\mathcal{A}}(x)s_{\rm T}\|_2 \ge \big(\sin(\theta)\|s_{\rm T}\|_2\big)/\kappa_{\rm J}. \tag{4.32}$$

Combining this with equation (4.28) we have

$$\big(g(x)+\sigma J(x)^T z\big)^T s_{\rm T} \le -{s_{\rm T}}^T M s_{\rm T} - \big(\kappa\sin(\theta)\|s_{\rm T}\|_2\big)/\kappa_{\rm J}. \tag{4.33}$$

We consider two cases. First suppose that ${s_{\rm T}}^T M s_{\rm T} \ge (\lambda_{\min}/8){s_{\rm T}}^T s_{\rm T}$. Then it immediately follows from equation (4.33) that

$$\big(g(x)+\sigma J(x)^T z\big)^T s_{\rm T} \le -(\lambda_{\min}/8)\|s_{\rm T}\|_2^2. \tag{4.34}$$

Next, suppose that ${s_{\rm T}}^T M s_{\rm T} < (\lambda_{\min}/8){s_{\rm T}}^T s_{\rm T}$. Lemma 4.3 then implies that $0 < \bar\theta < \theta$ and, therefore, $0 < \sin(\bar\theta) < \sin(\theta)$. We can then use this fact, equation (4.33), the Cauchy-Schwarz inequality, the definition of $\beta_{\max}$, and condition 3b to conclude

$$\begin{aligned}\big(g(x)+\sigma J(x)^T z\big)^T s_{\rm T} &\le \|s_{\rm T}\|_2^2\|M\|_2 - (\kappa\sin(\bar\theta)\|s_{\rm T}\|_2)/\kappa_{\rm J}\\ &\le \|s_{\rm T}\|_2\big(\beta_{\max}\|s_{\rm T}\|_2 - (\kappa\sin(\bar\theta))/\kappa_{\rm J}\big)\\ &\le -\frac{\kappa\sin(\bar\theta)}{2\kappa_{\rm J}}\|s_{\rm T}\|_2 \le -\frac{\kappa\sin(\bar\theta)}{2\kappa_{\rm J}}\|s_{\rm T}\|_2^2.\end{aligned} \tag{4.35}$$

If we define

$$c_2 = \min\left(\frac{\lambda_{\min}}{8}, \frac{\kappa \sin(\bar{\theta})}{2\kappa_{\text{J}}}\right) > 0, \tag{4.36}$$

then it follows from equations (4.34) and (4.35) that

$$\big(g(x) + \sigma J(x)^T z\big)^T s_{\text{T}} \leq -c_2 \|s_{\text{T}}\|_2^2, \tag{4.37}$$

which completes the proof. □

With a little more effort, we can show that the step from the Cauchy step s_k^{CP} to the solution of problem (4.23) is a descent direction for the underlying models. Since the Cauchy step is computed from the predictor step, it is imperative that we choose B_k so that s_k^{P} has desirable properties. The results in Section 4.1 suggest that we make the following assumption.

Assumption 4.1. *There exists a number $\lambda_{\min}^B > 0$ such that the sequence of positive-definite matrices $\{B_k\}$ defined in Algorithm 4.1 satisfies*

$$s^T B_k s \geq \lambda_{\min}^B s^T s \quad \textit{for all } s \in \mathbb{R}^n \textit{ and all } k \geq 0.$$

We now show that in the neighborhood of a solution w^*, the (unique) solution to

$$\underset{s\in\mathbb{R}^n}{\text{minimize}} \ \ (g_k + M s_k^{\text{CP}})^T s + \tfrac{1}{2} s^T M s \quad \text{subject to} \quad c_{\mathcal{A}}(x_k) + J_{\mathcal{A}}(x_k)(s_k^{\text{CP}} + s) = 0 \tag{4.38}$$

is a descent direction for the underlying model determined by the matrix M (under certain assumptions).

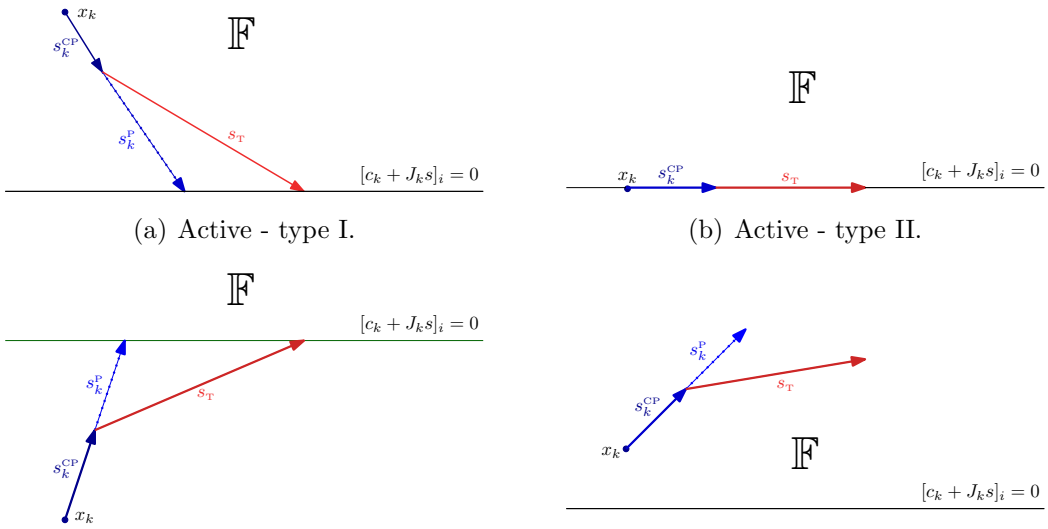
Lemma 4.5. *Let w^* be a minimizer for problem* (NP) *that satisfies the* LICQ *and strict complementarity and assume that $\sigma_k > \|y^*\|_\infty$, that Assumption 4.1 holds, and $\|B_k\|_2 \leq b_B$ for some $b_B > 0$. It follows that there exist positive numbers c_2 and ε_4 such that if iterate $k-1$ is successful, $w_k \in \mathcal{B}_{\varepsilon_4}(w^*)$ and $M \in S_{\varepsilon_4}$, then problem (4.38) is well-defined and the solution s_T satisfies*

$$(g_k + M s_k^{CP} + \sigma_k J_k^T z_k)^T s_T < -c_2 \|s_T\|_2^2 \quad \textit{for} \quad [z_k]_i = \begin{cases} 0 & \textit{if } i \in \mathcal{V}_k, \\ -1 & \textit{if } i \in \mathcal{S}_k, \end{cases} \tag{4.39}$$

where $\mathcal{V}_k = \{i : [c_k + J_k s_k^{CP}]_i < 0\}$ and $\mathcal{S}_k = \{i : [c_k + J_k s_k^{CP}]_i \geq 0\}$.

Proof. Since the proof is very similar to Lemma 4.4, we only point out the differences. First, by choosing $\lambda_{\min} \leq \lambda_{\min}^B$, we have that $B_k \in S_\varepsilon$ for all $\varepsilon > 0$. Second, since $\sigma_k > \|y^*\|_\infty$ the predictor subproblem (1.4) is equivalent to problem (4.13) for the choice $M = B_k$ provided that the trust-region constraint is inactive. Third, Lemma 4.2 shows that the solution to problem (4.13) with $M = B_k$ correctly identifies the optimal active set if w_k is sufficiently close to w^*, so that the solution satisfies system (4.6). Equation (4.5) then shows that we can make the solution to problem (4.13) arbitrarily small by choosing w_k

sufficiently close to w^*. Fourth, since iteration $k-1$ is successful by assumption, we know that the predictor trust-region radius is at least as large as $\Delta_{\rm R}$ for iteration k (see Algorithm 4.1). Combining all of this together, we know that there exists a positive number $\varepsilon_4 < \mu/2$ (μ is defined in Lemma 4.2) such that if $w_k \in \mathcal{B}_{\varepsilon_4}(w^*)$ then the trust-region in the predictor step will be inactive, $s_k^{\rm P}$ correctly identifies the optimal active set (see Figure 4.2), and $\|s_k^{\rm P}\|_2$ is as small as we wish. The system that arises in place of (4.26) is



(a) Active - type I. (b) Active - type II.

(c) Active - type III. (d) Inactive - type IV.

Figure 4.2: The only four possibilities in a small enough neighborhood of the solution w^*. (a) For type I, we have $c_i(x_k) > 0$, $c_i^* = 0$, and $\nabla c_i(x_k)^T s_{\rm T} < 0$. (b) For type II, we have $c_i(x_k) = 0$, $c_i^* = 0$, and $\nabla c_i(x_k)^T s_{\rm T} = 0$. (c) For type III, we have $c_i(x_k) < 0$, $c_i^* = 0$, and $\nabla c_i(x_k)^T s_{\rm T} > 0$. (d) For type IV, we have $c_i^* > 0$.

$$\begin{pmatrix} M & J_{\mathcal{A}}(x)^T \\ J_{\mathcal{A}}(x) & 0 \end{pmatrix} \begin{pmatrix} s \\ -q \end{pmatrix} = - \begin{pmatrix} g(x) - J_{\mathcal{A}}(x)^T y_{\mathcal{A}} + M s_k^{\rm CP} \\ c_{\mathcal{A}}(x) + J_{\mathcal{A}}(x_k) s_k^{\rm CP} \end{pmatrix}, \tag{4.40}$$

but since $\|s_k^{\rm CP}\|_2 \leq \|s_k^{\rm P}\|_2$ we can ensure – by possibly decreasing ε_4 – that parts 3a and 3b of Lemma 4.4 are once again satisfied. The rest of the proof is identical. □

4.3. Local convergence with an SEQP step

Our first rate of convergence result for Algorithm 4.1 assumes that the SQP step is computed from subproblem (SEQP) as discussed in [19, Section 2.3.2]. We restate this subproblem for convenience:

$$\text{(SEQP)} \qquad \begin{array}{ll} \underset{s\in\mathbb{R}^n}{\text{minimize}} & \bar{f}_k + (g_k + H_k s_k^{\rm P})^T s + \frac{1}{2} s^T H_k s \\ \text{subject to} & [J_k s]_{\mathcal{A}(s_k^{\rm P})} = 0, \ \ \|s\|_2 \leq \Delta_k^{\rm S}, \end{array}$$

where $\mathcal{A}(s_k^{\rm P}) = \{i : [c_k + J_k s_k^{\rm P}]_i \leq 0\}$ and $\bar{f}_k = f_k + g_k^T s_k^{\rm P} + \frac{1}{2} s_k^{\rm P T} H_k s_k^{\rm P}$. Since this subproblem only defines multipliers for the constraints whose indices are in the set $\mathcal{A}(s_k^{\rm P})$, we form $y_k^{\rm S}$

by "scattering" the multipliers from subproblem (SEQP) into the appropriate locations of a zero-vector of length m.

Theorem 4.1. (SEQP local convergence result) *Let w^* be a minimizer for problem* (NP) *that satisfies the strong second-order sufficient conditions as given by Definition 1.5. Let Assumption 4.1 hold and suppose that $\sigma_k \equiv \sigma_b > \|y^*\|_\infty$ and $\|B_k\|_2 \leq b_B$ for some $b_B > 0$ and $\sigma_b > 0$ and for all $k \geq 0$, the SQP step is computed from subproblem* (SEQP) *with the choice $H_k \equiv H(x_k, y_k^F)$, and max_fails ≥ 1 in Algorithm 4.1. It follows that there exists a positive number δ, such that if the SQP step is computed for every iteration once the first successful iterate of Algorithm 4.1 is contained in $\mathcal{B}_\delta(w^*)$ then the sequences of iterates $\{x_k\}$ and $\{y_k\}$ generated by Algorithm 4.1 converge to x^* and y^* at a Q-superlinear and R-superlinear rate, respectively. Moreover, if $H(w)$ is Lipschitz continuous in a neighborhood of w^*, then they convergence at a Q-quadratic and R-quadratic rate, respectively.*

Proof. We begin by setting $\lambda_{\min} = \min(\lambda_{\min}^H/2, \lambda_{\min}^B)$ and $\beta_{\max} = \max(b_B, \|H^*\|_2 + 1)$ in the definition of $S(x\,; x^*)$ as given by (4.2), and by letting β, ε_1, ε_2, and μ be the positive constants guaranteed by Lemmas 4.1 and 4.2; note that they satisfy $0 < \mu \leq \varepsilon_2 \leq \varepsilon_1$ by construction, so that part (ii) of Lemma 4.1 implies

$$\mathcal{B}_{\mu/2}(w^*) \subseteq \mathcal{B}_\mu(w^*) \subseteq \mathcal{B}_{\varepsilon_2}(w^*) \subseteq \mathcal{B}_{\varepsilon_1}(w^*) \;\text{ and }\; S_{\mu/2} \subseteq S_\mu \subseteq S_{\varepsilon_2} \subseteq S_{\varepsilon_1}, \tag{4.41}$$

where S_ε is defined by (4.2) and (4.3). By possibly decreasing μ, we can also guarantee that if w and $\bar{w}$ are contained in $B_\mu(w^*)$, then the following conditions are satisfied:

C1. $\|y - y^*\|_\infty < \sigma_b - \|y^*\|_\infty$;

C2. $\|H\big(x, y^{\mathrm{F}}(x)\big)\|_2 \leq \|H^*\|_2 + 1$, where $y^{\mathrm{F}}(x)$ is any estimate satisfying $y^{\mathrm{F}}(x) - y^* = O(\|x - x^*\|_2)$;

C3. $s^T H\big(x, y^{\mathrm{F}}(x)\big)s \geq (\lambda_{\min}^H/2)s^T s$ for all s satisfying $J_{\mathcal{A}}^* s = 0$;

C4. Newton's Method applied to the function F_N in equation (4.1) converges from the point w to w^*; moreover, the Newton update w_+ to w satisfies $\|w_+ - w^*\|_2 \leq \|w - w^*\|_2$ (see Dennis and Schnabel [11, Theorem 5.2.1].

With μ defined, we now pick $\delta_\Delta > 0$ so that

C5. $\delta_\Delta \leq \min\big(\mu/2, \varepsilon_4\big)$, where ε_4 is defined in Lemma 4.5; and

C6. $\delta_\Delta \leq \eta_c \Delta_{\mathrm{R}}/2$, where $0 < \Delta_{\mathrm{R}} \leq \Delta_{\mathrm{U}}$ and η_c are used in Algorithm 4.1.

Finally, we choose $\delta > 0$ so that

C7. $\delta \leq \min(\mu/2, \varepsilon_4)$, where ε_4 is defined in Lemma 4.5; and

C8. if $w \in \mathcal{B}_\delta(w^*)$, then the following bound on the KKT equality conditions is satisfied:

$$\|f_{\mathrm{KKT}}(w)\|_2 = \left\| \begin{pmatrix} g(x) - J(x)^T y \\ c(x) \cdot y \end{pmatrix} \right\|_2 < \frac{1}{4\beta} \min\left(\delta_\Delta, \eta_c \Delta_{\mathrm{R}}\right).$$

Now let $k-1$ be the first successful iterate generated by Algorithm 4.1 such that $w_k \in \mathcal{B}_\delta(w^*)$. By construction of Algorithm 4.1 and the fact that the SQP trust-region scale factor satisfies $\tau_f \geq 1$, we have

$$\Delta_k^{\rm P} \geq \Delta_{\rm R} > 0 \;\; \text{and} \;\; \Delta_k^{\rm S} \geq \tau_f \Delta_{\rm R} \geq \Delta_{\rm R} > 0. \tag{4.42}$$

Since equation (4.41) and C7 imply that $w_k \in \mathcal{B}_{\mu/2}(w^*)$, it follows from C8, Lemma 4.2, and (4.42) that $J_{\mathcal{A}}(x_k)$ has full row rank and if $M \in S_{\mu/2}$ then $x_k(M)$ correctly identifies the optimal active set and satisfies

$$\|x_k(M) - x_k\|_\infty \leq 2\beta \|F_M(w_k\,; w_k)\|_2 \leq \tfrac{1}{2}\min(\delta_\Delta, \eta_c \Delta_{\rm R}) \leq \tfrac{1}{2}\eta_c \min(\Delta_k^{\rm P}, \Delta_k^{\rm S}). \tag{4.43}$$

We now observe that $B_k \in S_{\mu/2}$ by construction and is, in fact, positive definite. Furthermore, since C1 implies $\sigma_k = \sigma_b > \|y_k(B_k)\|_\infty$ and equation (4.43) implies $\|x_k(B_k) - x_k\|_\infty \leq (\eta_c/2)\Delta_k^{\rm P} < \Delta_k^{\rm P}$, we must have $w_k^{\rm P} = w_k(B_k)$. Thus the solution to the predictor subproblem satisfies $s_k^{\rm P} = x_k(B_k) - x_k$, correctly identifies the optimal active set, and is not restricted by the trust-region constraint, i.e., $s_k^{\rm P}$ is the solution to (4.23) with $M = B_k$.

Next we observe that C2 and C3 imply that $H_k \in S(x^*; x^*) \subset S_{\mu/2}$. Therefore, the point $w_k(H_k)$ is well-defined, identifies the optimal active set, and is the unique minimizer of problem (4.13) in a neighborhood of w_k for $M = H_k$. Since $J_{\mathcal{A}}(x_k)$ has full row rank, it follows from (4.41), part (iii) of Lemma 4.1, and [1, Lemma 1.27] that subproblem (SEQP) has $s_k^{\rm S}$ as a unique solution. It follows that if $\|x_k(H_k) - (x_k + s_k^{\rm P})\|_2 \leq \Delta_k^{\rm S}$, then $s_k^{\rm S} = x_k(H_k) - (x_k + s_k^{\rm P})$ (see Figure 4.3). Using the triangle inequality, the definition of $w_k(B_k)$, and equation (4.43), we have

$$\|x_k(H_k) - (x_k + s_k^{\rm P})\|_2 \leq \|x_k(H_k) - x_k\|_2 + \|s_k^{\rm P}\|_2 = \|x_k(H_k) - x_k\|_2 + \|x_k(B_k) - x_k\|_2 \leq \eta_c \Delta_k^{\rm S} \leq \Delta_k^{\rm S}. \tag{4.44}$$

Thus, if $s_k^{\rm P} + s_k^{\rm S}$ satisfies condition (1.7) then $s_k = s_k^{\rm P} + s_k^{\rm S}$ and it follows that $x_k + s_k = x_k(H_k)$ and $y_k^{\rm S} = y_k(H_k)$. We now show that this is the case. If $s_{\rm T} \neq 0$ then C5 and Lemma 4.5 show that the vector $s_{\rm T}$, which satisfies $s_k^{\rm CP} + s_{\rm T} = x_k(H_k)$, is a descent direction for the model M_k^H. Therefore, $M_k^H(s_k^{\rm P} + s_k^{\rm S}) < M_k^H(s_k^{\rm CP})$ so that condition (1.7) is satisfied by $s_k^{\rm P} + s_k^{\rm S}$. On the other hand, if $s_{\rm T} = 0$ then it follows that $s_k^{\rm P} = s_k^{\rm CP}$ and $s_k^{\rm S} = 0$, so that $s_k^{\rm P} + s_k^{\rm S} = s_k^{\rm CP}$ trivially satisfies condition (1.7).

If $x_k + s_k$ is a successful step, then $x_{k+1} \leftarrow x_k + s_k$; otherwise, the update $x_{k+1} \leftarrow x_k + s_k$ is still made since *max_fails* ≥ 1, but a nonmonotone phase is entered. In either case, the vector w_{k+1} is the same vector that is obtained by performing one step of Newton's Method on the function F_N (see equation (4.1)) from the point $(x_k, y_k^{\rm F})$ with the understanding that y_{k+1} is formed by "scattering" $y_k^{\rm S}$ into a zero-vector of length m. Since Algorithm 4.1 makes the assignment $w_{k+1} \leftarrow w_k^{\rm S}$, it follows from C4 that $w_{k+1} \in \mathcal{B}_\delta(w^*)$ and so the same argument may be repeated starting from the point w_{k+1}; this results in a vector w_{k+2} that has the same properties as w_{k+1} and is, in fact, equivalent to performing one step of Newton's Method on the function F_N from the point $(x_{k+1}, y_{k+1}^{\rm F})$. The only difference in the argument is that the predictor and SQP trust-region radii are only guaranteed to be

bigger than $\eta_c \Delta_{\text{R}}$ since the predictor trust-region radius may be contracted if the point w_{k+1} was not successful. However, conditions C1–C8 were chosen to ensure that all the previous estimates still hold. It is shown in [9, Section 15.3.2.3] that this process is sufficient for avoiding the Maratos effect provided the ratio r_k of actual to predicted decrease in the merit function is defined using the strategy in Algorithm 4.1; therefore, w_{k+2} will be accepted by the ℓ_1-merit function. This argument can clearly be repeated so that every remaining step will be accepted. As for rate of convergence, we have from [28, Theorem 11.2] and C2 that

$$\begin{pmatrix} x_{k+1} - x^* \\ y_{k+1} - y^* \end{pmatrix} = o\left(\left\| \begin{pmatrix} x_k - x^* \\ y_k^{\text{F}} - y^* \end{pmatrix} \right\|_2 \right) = o\left(\|x_k - x^*\|_2 \right) \tag{4.45}$$

so that $\{x_k^*\}$ and $\{y_k\}$ converge to x^* and y^* Q-superlinearly and R-superlinearly, respectively (see [29, Chapter 9] for a description of Q and R convergence); C2 also shows that $\{y_k^{\text{F}}\}$ converges to y^* R-superlinearly. If $H(w)$ is locally Lipschitz continuous then a similar argument shows that $\{x_k\}$ converges to x^* Q-quadratically and that $\{y_k\}$ and $\{y_k^{\text{F}}\}$ converge to y^* R-quadratically. □

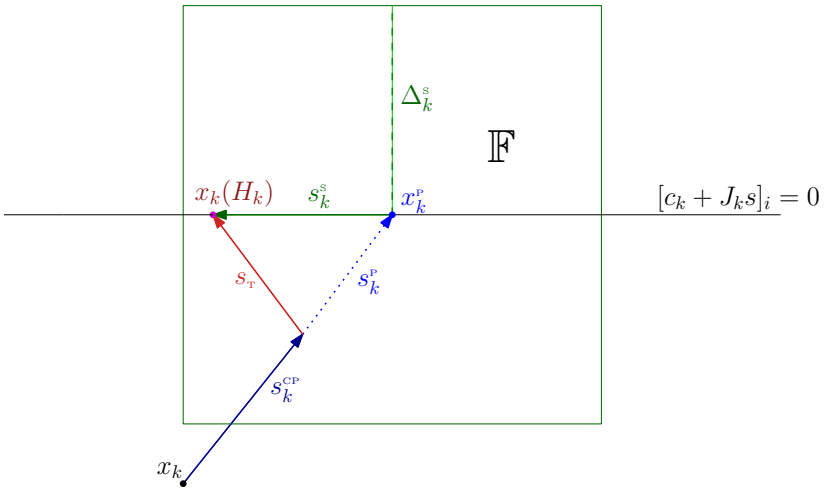


Figure 4.3: A depiction of the scenario in Theorem 4.1. The following quantities are displayed: x_k is the current iterate, s_k^{P} is the predictor step, x_k^{P} is the predictor point, s_k^{CP} is the Cauchy step, s_k^{S} is the SQP step as computed from problem (SEQP), Δ_k^{S} is the SQP trust-region radius, s_{T} is the solution to problem (4.38), $x_k(H_k)$ is the first n components of $w_k(H_k)$, which is the closest minimizer to w_k for problem (4.13) with the choice $M = H_k$, and $\mathbb{F}$ denotes the feasible side of the constraint $[c_k + J_k s]_i \geq 0$.

Lemma 4.6. *Let w^* be a minimizer for problem* (NP) *that satisfies the* LICQ *and strict complementarity and suppose that $\sigma_k > \|y^*\|_\infty$, that Assumption 4.1 holds, and $\|B_k\|_2 \leq b_B$ for some $b_B > 0$ and all $k \geq 0$. Then there exists a scalar $\delta_{\text{P}} > 0$ such that if iterate $k-1$ is successful and $w_k \in \mathcal{B}_{\delta_P}(w^*)$, then*

$$y_k^P - y^* = O(\|x_k - x^*\|_2) \quad \textit{and} \quad [y_k^P]_{\mathcal{I}} = 0, \tag{4.46}$$

where y_k^P are the multipliers for the predictor subproblem (1.5).

Proof. Let $\delta_{\rm P}$ be defined to satisfy conditions C5–C8 of Theorem 4.1. It follows, just as in the proof of Theorem 4.1, that $s_k^{\rm P} = x_k(B_k) - x_k$ and that $s_k^{\rm P}$ is the unique solution to problem (4.23) with the choice $M = B_k$. This implies that $(s_k^{\rm P}, y_k^{\rm P})$ satisfies system (4.6) $(\pi = y_k^{\rm P})$ so that (4.46) follows from (4.5). □

4.4. Local convergence with an SIQP step

We now consider the rate of convergence for Algorithm 4.1 when the SQP step is computed from subproblem (SIQP-E) as described in [19, Section 2.3.1]. We restate this subproblem for convenience:

$$\begin{array}{lll} \text{(SIQP-E)} & \underset{s\in\mathbb{R}^n}{\text{minimize}} & \bar{f}_k + (g_k + H_k s_k^{\rm CP})^T s + \frac{1}{2} s^T H_k s + \sigma_k \|[c_k + J_k(s_k^{\rm CP} + s)]^-_{\mathcal{V}_k}\|_1 \\ & \text{subject to} & [c_k + J_k(s_k^{\rm CP} + s)]_{\mathcal{S}_k} \geq 0, \\ & & (g_k + H_k s_k^{\rm CP} + \sigma_k J_k^T z_k)^T s \leq 0, \quad \|s\|_\infty \leq \Delta_k^{\rm S}, \end{array}$$

where $z_k \in \mathbb{R}^m$ is defined by (4.39), $\bar{f}_k = f_k + g_k^T s_k^{\rm CP} + \frac{1}{2} s_k^{\rm CP\,T} H_k s_k^{\rm CP}$, and $(g_k + H_k s_k^{\rm CP} + \sigma_k J_k^T z_k)^T s \leq 0$ is the so-called "descent-constraint".

We begin by making two observations. First, since problem (SIQP-E) is generally a nonconvex *inequality* constrained QP, we will need to assume that the solution $s_k^{\rm S}$ is one of minimal norm; a similar assumption is made by Robinson in [36, Section 3]. Although this assumption is not ideal, it is not too offensive within our setting; if we use an active-set QP solver with a hot start based on the active set obtained from the predictor step, then the solution to subproblem (SIQP-E) will ultimately be the same as the solution to subproblem (SEQP). Theorem 4.1 validates that this is a good step and, therefore, if this strategy is used then the "minimum-norm solution" assumption is not necessary. The second observation is that if the SQP step is chosen to be one of minimal norm, then the proof of Theorem 4.1 carries over since 1) the Cauchy step $s_k^{\rm CP}$ satisfies $\|s_k^{\rm CP}\|_\infty \leq \|s_k^{\rm P}\|_\infty$; 2) the vector $x_k(H_k) - s_k^{\rm CP}$ is a solution to subproblem (SIQP-E); and 3) Lemma 4.5 guarantees that the descent-constraint does not interfere with the step from $s_k^{\rm CP}$ to $x_k(H_k)$.

Theorem 4.2. (SIQP-E local convergence result) *Let w^* be a minimizer for problem* (NP) *that satisfies the strong second-order sufficient conditions as given by Definition* 1.5. *Let Assumption* 4.1 *hold and assume that $\sigma_k \equiv \sigma_b > \|y^*\|_\infty$ and $\|B_k\|_2 \leq b_B$ for some $b_B > 0$ and $\sigma_b > 0$ and all $k \geq 0$, the SQP step is computed from subproblem* (SIQP-E) *with the choice $H_k \equiv H(x_k, y_k^F)$, and max_fails ≥ 1 in Algorithm* 4.1. *It follows that there exists a positive number δ, such that if the SQP step is a solution of minimal-norm and is computed for every iteration once the first successful iterate of Algorithm* 4.1 *is contained in $\mathcal{B}_\delta(w^*)$, then the sequences of iterates $\{x_k\}$ and $\{y_k\}$ converge to x^* and y^* at a Q-superlinear and R-superlinear rate, respectively. Moreover, if $H(w)$ is Lipschitz continuous in a neighborhood of w^*, then they converge at a Q-quadratic and R-quadratic rate, respectively.*

Proof. Follows from the proof of Theorem 4.1, the discussion above, and Lemma 4.4. □

5. Numerical results

We tested Algorithm 4.1 on the Hock-Schittkowski (HS) [23] test problems. The HS test suite is comprised of generally small and dense problems that are very useful during early stages of code development; the small size of the problems allows for relatively careful inspection of each problem. We note that problem *hs87* has been removed from the test set since the objective function is not continuous.

To be precise, we tested three variants of Algorithm 4.1—they differ in how we compute the SQP step and update the penalty parameter. In the first variant we computed the SQP step from the inequality constrained subproblem (SIQP-E) on page 32 and updated the penalty parameter by using "steering" as briefly described at the end of Section 3. In the second variant we computed the SQP step from the equality constrained subproblem (SEQP) on page 28 and again used steering to update the penalty parameter. Finally, in the third variant we computed the SQP step from the inequality constrained subproblem (SIQP-E) and updated the penalty parameter by using Algorithm 3.1.

Since the problems in the test set are of small dimension, we chose to update the positive-definite matrix B_k in the predictor subproblem (1.5) by using the BFGS update. To perform this update, we used the vectors s_k and $d_k = \nabla_x \mathcal{L}(x_k+s_k, y_{k+1}) - \nabla_x \mathcal{L}(x_k, y_{k+1})$. If these vectors did not result in a sufficiently positive-definite update, then we used the damped vectors given by equations (2.11) and (2.12). For simplicity, we chose $B_0 = I$.

In all cases for the SQP and Cauchy step calculations we chose $H_k \equiv H(x_k, y_k^{\mathrm{P}})$, where y_k^{P} is the multiplier vector from the predictor subproblem. We solved both the (convex) quadratic program (1.5) and the (generally indefinite) quadratic program (SIQP-E) using the GALAHAD [18] package `QPC`, which is a "cross-over" QP solver. In the first phase, `QPC` calls the GALAHAD interior-point QP solver `QPB` [8] to compute an approximate solution and an estimate of the optimal active set. In the second phase, `QPC` calls the GALAHAD active-set QP solver `QPA` [20] to "refine" the approximate solution from the first phase. To solve the equality constrained QP (SEQP) we used the GALAHAD package `EQP`, which has been designed to solve problems of precisely this form. We should mention that most of the GALAHAD packages, including the QP solvers mentioned above, use the sparse solvers `MA48` and `MA57` from [24] to handle the required systems. The modular design of all the GALAHAD packages makes it easy to call these subroutines as needed.

The following parameters were used in all cases: primal/dual/complementarity slackness tolerances $\tau_p = \tau_d = \tau_c = \texttt{1.0e}^{-5}$, successful/very successful tolerances $\eta_s = \texttt{0.01}$ and $\eta_{VS} = \texttt{0.7}$, maximum predictor trust-region radius $\Delta_{\mathrm{U}} = \texttt{1000}$, trust-region "reset" radius $\Delta_{\mathrm{R}} = \texttt{1.0e}^{-4}$, SQP trust-region scale factor $\tau_f = \texttt{4.0}$, number of nonmonotone steps allowed *max_fails* = 1, and trust-region contraction and expansion factors $\eta_c = \texttt{0.1}$ and $\eta_e = \texttt{5.0}$. We used an initial penalty parameter of $\sigma = \texttt{1.0}$ for the first and second strategies, and an initial penalty parameter of $\sigma = \texttt{9.0}$ for the third strategy. The larger

(seemingly arbitrary) initial penalty parameter for the third variant was chosen based on performance and seems to be related to the less dynamic nature of the update as compared with steering.

Table 5.1: We record the number of function and gradient evaluations required, and note that the difference between these values indicates the number of unsuccessful steps attempted; an F indicates that more than 500 evaluations were required.

	SIQP-steer		SEQP-steer		SIQP-seq	
Prob	#fc	#gJ	#fc	#gJ	#fc	#gJ
hs1	46	25	39	24	46	25
hs2	11	8	11	8	11	8
hs3	3	3	3	3	3	3
hs4	3	2	3	2	3	2
hs5	10	6	9	6	10	6
hs6	3	3	5	5	3	3
hs7	10	9	9	8	12	10
hs8	6	6	6	6	6	6
hs9	3	3	4	4	3	3
hs10	10	10	10	10	10	10
hs11	6	6	6	6	6	6
hs12	7	7	6	6	14	11
hs13	15	12	29	27	56	54
hs14	5	5	5	5	5	5
hs15	7	7	7	7	20	20
hs16	4	4	4	4	5	5
hs17	7	7	7	7	10	10
hs18	9	8	7	7	9	8
hs19	6	6	6	6	13	13
hs20	8	8	4	4	16	15
hs21	2	2	2	2	2	2
hs22	2	2	2	2	2	2
hs23	6	6	6	6	6	6
hs24	3	3	4	4	3	3
hs25	1	1	1	1	1	1
hs26	17	17	16	16	17	17
hs27	12	11	14	13	13	12
hs28	2	2	3	3	2	2
hs29	6	6	6	6	31	23

	SIQP-steer		SEQP-steer		SIQP-seq	
Prob	#fc	#gJ	#fc	#gJ	#fc	#gJ
hs30	10	10	10	10	10	10
hs31	8	8	9	7	5	5
hs32	3	3	4	4	3	3
hs33	5	5	14	11	5	5
hs34	9	8	9	8	34	28
hs35	2	2	3	3	2	2
hs36	3	3	4	4	3	3
hs37	5	5	5	5	5	5
hs38	84	38	65	46	84	38
hs39	12	11	13	11	12	11
hs40	4	4	4	4	4	4
hs41	2	2	2	2	2	2
hs42	6	6	4	4	4	4
hs43	7	7	8	8	12	10
hs44	2	2	3	3	2	2
hs45	3	3	5	4	3	3
hs46	16	16	16	16	16	16
hs47	15	15	20	18	19	17
hs48	2	2	3	3	2	2
hs49	16	16	16	16	16	16
hs50	9	9	6	6	9	9
hs51	2	2	2	2	2	2
hs52	2	2	2	2	2	2
hs53	2	2	2	2	2	2
hs54	9	9	14	14	22	22
hs55	2	2	2	2	2	2
hs56	110	82	F	F	F	F
hs57	8	6	6	6	8	6
hs59	10	9	8	8	18	15

Tables 5.1 and 5.2 on pages 34 and 39 give our preliminary numerical results for these three strategies; column SIQP-steer corresponds to the first strategy, column SEQP-steer corresponds to the second strategy, and column SIQP-seq corresponds to the third strategy. For each strategy we have recorded the number of function evaluations #fc and the number of gradient evaluations #gJ. Note that if the quantity #fc - #gJ is positive, then its value represents the number of unsuccessful iterations, i.e., the number of times that the trust-region radii were necessarily decreased in order to obtain good agreement between the faithful model M_k^H and the merit function ϕ. An F indicates that more than 500 evaluations

were required and an FQP indicates that the QP solver failed.

We believe that these preliminary results show that our method works quite well. The strictly convex predictor subproblem combined with either SQP subproblem (SIQP-E) or (SEQP) typically generates iterates that rapidly converge to a solution; the nonmonotone approach avoids the Maratos effect. The results also indicate that using steering to update the penalty parameter generally performs better than the method discussed in Section 3; this agrees with [4, 5]. In particular, steering was essential in solving *hs93* for otherwise the merit function converged to minus infinity, the constraints blew up, and the method failed. In a less clear manner, the update to the penalty parameter is important in solving *hs56*. When the update in Section 3 was used, the merit function again converged to minus infinity; the same occurred when the SQP step was computed from subproblem (SEQP) and steering was used.

6. Conclusions and future work

In [19], we proved global convergence of a second-derivative SQP method for minimizing the ℓ_1-penalty function for a fixed value of the penalty parameter. This algorithm requires the definition of a positive-definite matrix that approximates the Hessian of the Lagrangian; in Section 2 we considered two possibilities. The first was a simple diagonal approximation that attempted to crudely estimate the size of the Hessian of Lagrangian. The second approach used a *limited*-memory BFGS update. We proceeded to show how the resultant dense QP could be transformed into an equivalent QP whose sparsity is essentially the same as the initial approximation (which will be sparse in practice). In Section 3 we gave details on a simple strategy for updating the penalty parameter based on minimizing the ℓ_1-penalty function over a sequence of increasing values of the penalty parameter. Although the basic idea is certainly not new [34, 25, 38, 30, 42, 3, 26], the details of our very simple strategy have not been published to our knowledge. The main result of that section is that the penalty parameter will stay uniformly bounded and that an *approximate* solution to the nonlinear programming problem will be computed in a finite number of iterations. However, the primary purpose of this paper was to study the local convergence properties of a *nonmonotone* variant of the algorithm given in [19]. In Section 4 we gave two local convergence results—the first applies when the SQP step is computed from an *equality* constrained subproblem (the so-called SEQP approach) and the second applies when the SQP step is computed from an *inequality* constrained subproblem (the so-called SIQP approach). Both results show superlinear convergence of the iterates to a solution satisfying the strong second-order sufficiency conditions; under slightly stronger assumptions on the second-derivatives, the convergence is quadratic. We note that the second convergence result required that the so-called "descent-constraint" be inactive at a "minimum-norm" solution to problem (SIQP-E) on page 32; this result was presented as Lemma 4.5. In Section 5 we gave preliminary numerical results for the Hock-Schittkowski test problems. Our first set of results were based on using "steering" [4, 5] to update the penalty parameter and computing the SQP step from the inequality constrained subproblem (SIQP-E).

Although this requires "solving" a potentially indefinite quadratic program, which is generally perceived as a bad idea, the results are quite good. Our second set of results also used steering to update the penalty parameter, but instead computed the SQP step from the equality constrained subproblem (SEQP) on page 28. Our last set of results were based on solving subproblem (SIQP-E) for the SQP step, but updating the penalty parameter by monitoring the norm of the constraint violation over a sequence of *approximate* minimizers of the merit function (see Section 3). We stress that these results are preliminary and that they are not intended to compare the (SIQP) approach with the (SEQP) approach, but rather to show that *both* approaches have the potential to be successful in practice. We must also mention that in essentially simultaneous work, Morales, Nocedal, and Wu [27] have been developing an ℓ_1-SQP *line-search* algorithm based on subproblem (SEQP) in which the optimal active set is predicted by solving a convex QP. During initial testing, the authors have reported that the use of the convex QP to predict the active set combined with the additional equality constrained subproblem is effective.

There are still many ideas to be explored and options to be added to our evolving and (soon to be) freely available Fortran 95 GALAHAD [18] package `S2QP`. First, we want to explore whether solving an SQP subproblem with perturbed constraints (a "correction step") is sufficient for avoiding the Maratos effect. This deserves further investigation since if this is the case, then fast convergence would be guaranteed without the need for a nonmonotone approach. Second, we plan on experimenting with modern approaches for adjusting/defining the trust-region radius [13, 14, 39] as well as exploring new possibilities. Third, we want to examine the consequences of removing the trust-region constraint from the predictor step computation; the problem is strictly convex and, therefore, well-defined without this constraint. Fourth, we will investigate the notion of "steering", as it pertains to our setting, as a reliable strategy for dynamically updating the penalty parameter. Finally, we plan on extensively testing the package on larger problems from the CUTEr [17] test set.

References

[1] D. P. Bertsekas, *Constrained optimization and Lagrange multiplier methods*, Computer Science and Applied Mathematics, Academic Press Inc. [Harcourt Brace Jovanovich Publishers], New York, 1982.

[2] P. T. Boggs and J. W. Tolle, *Sequential quadratic programming*, in Acta Numerica, 1995, Acta Numer., Cambridge Univ. Press, Cambridge, 1995, pp. 1–51.

[3] J. V. Burke, *A robust trust region method for constrained nonlinear programming problems*, SIAM J. Optim., 2 (1992), pp. 324–347.

[4] R. H. Byrd, N. I. M. Gould, J. Nocedal, and R. A. Waltz, *On the convergence of successive linear-quadratic programming algorithms*, SIAM J. Optim., 16 (2005), pp. 471–489.

[5] R. H. Byrd, J. Nocedal, and R. A. Waltz, *Steering exact penalty methods for nonlinear programming*, Opt. Methods & Software., 23 (2008), pp. 197–213.

[6] R. H. Byrd and R. B. Schnabel, *Continuity of the null space basis and constrained optimization*, Math. Program., 35 (1986), pp. 32–41.

[7] R. H. Byrd, R. A. Tapia, and Y. Zhang, *An SQP augmented Lagrangian BFGS algorithm for constrained optimization*, SIAM J. Optim., 20 (1992), pp. 210–241.

[8] A. R. Conn, N. I. M. Gould, D. Orban, and Ph. L. Toint, *A primal-dual trust-region algorithm for non-convex nonlinear programming*, Math. Program., 87 (2000), pp. 215–249. Studies in algorithmic optimization.

[9] A. R. Conn, N. I. M. Gould, and Ph. Toint, *Trust-Region Methods*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2000.

[10] A. R. Conn and T. Pietrzykowski, *A penalty function method converging directly to a constrained optimum*, SIAM J. Numer. Anal., 14 (1977), pp. 348–375.

[11] J. E. Dennis, Jr. and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1983.

[12] R. Fletcher, *Practical Methods of Optimization*, Volume 2: Constrained Optimization, John Wiley and Sons, Chichester and New York, 1981.

[13] E. M. Gertz, *Combination Trust-Region Line-Search Methods for Unconstrained Optimization*, PhD thesis, Department of Mathematics, University of California San Diego, 1999.

[14] E. M. Gertz, *A quasi-Newton trust-region method*, Math. Program., 100 (2004), pp. 447–470.

[15] P. E. Gill, W. Murray, and M. A. Saunders, *SNOPT: An SQP algorithm for large-scale constrained optimization*, SIAM Rev., 47 (2005), pp. 99–131.

[16] G. H. Golub and C. F. Van Loan, *Unsymmetric positive definite linear systems*, Linear Algebra Appl., 28 (1979), pp. 85–98.

[17] 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 (2003), pp. 373–394.

[18] ——, *GALAHAD, a library of thread-safe Fortran 90 packages for large-scale nonlinear optimization*, ACM Trans. Math. Softw., 29 (2003), pp. 353–372.

[19] N. I. M. Gould and D. P. Robinson, *A second derivative SQP method : theoretical issues*, Numerical Analysis Report 08/18, Computational Laboratory, University of Oxford, Oxford, UK, 2008.

[20] N. I. M. Gould and Ph. L. Toint, *An iterative working-set method for large-scale non-convex quadratic programming*, Applied Numerical Mathematics, 43 (2002), pp. 109–128.

[21] S.-P. Han, *Superlinearly convergent variable metric algorithms for general nonlinear programming problems*, Math. Program., 11 (1976), pp. 263–282.

[22] M. R. Hestenes, *Multiplier and gradient methods*, J. Optim. Theory Appl., 4 (1969), pp. 303–320.

[23] W. Hock and K. Schittkowski, *Test Examples for Nonlinear Programming Codes*, Lecture Notes in Economics and Mathematical Systems 187, Springer Verlag, Berlin, Heidelberg and New York, 1981.

[24] HSL, *A collection of Fortran codes for large-scale scientific computation*, 2007. See http://www.cse.clrc.ac.uk/nag/hsl/.

[25] D. Q. Mayne and N. Maratos, *A first-order, exact penalty function algorithm for equality constrained optimization problems*, Math. Program., 16 (1979), pp. 303–324.

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

[27] J. Morales, J. Nocedal, and Y. Wu, *A sequential quadratic programming algorithm with an additional equality constrained phase*, Tech. Rep. OTC-05, Northwestern University, 2008.

[28] J. Nocedal and S. J. Wright, *Numerical Optimization*, Springer-Verlag, New York, 1999.

[29] J. M. Ortega and W. C. Rheinboldt, *Iterative solution of nonlinear equations in several variables*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2000. Reprint of the 1970 original.

[30] J. F. A. Pantoja and D. Q. Mayne, *Exact penalty functions with simple updating of the penalty parameter*, J. Optim. Theory Appl., 69 (1991), pp. 441–467.

[31] T. Pietrzykowski, *An exact potential method for constrained maxima*, SIAM J. Numer. Anal., 6 (1969), pp. 299–304.

[32] M. J. D. Powell, *A method for nonlinear constraints in minimization problems*, in Optimization, R. Fletcher, ed., London and New York, 1969, Academic Press, pp. 283–298.

[33] ——, *The convergence of variable metric methods for nonlinearly constrained optimization calculations*, in Nonlinear Programming, 3 (Proc. Sympos., Special Interest Group Math. Programming, Univ. Wisconsin, Madison, Wis., 1977), Academic Press, New York, 1978, pp. 27–63.

[34] M. J. D. Powell, *A fast algorithm for nonlinearly constrained optimization calculations*, in Numerical Analysis, Dundee 1977, G. A. Watson, ed., no. 630 in Lecture Notes in Mathematics, Heidelberg, Berlin, New York, 1978, Springer Verlag, pp. 144–157.

[35] S. M. Robinson, *A quadratically-convergent algorithm for general nonlinear programming problems*, Math. Program., 3 (1972), pp. 145–156.

[36] ——, *Perturbed Kuhn-Tucker points and rates of convergence for a class of nonlinear programming algorithms*, Math. Program., 7 (1974), pp. 1–16.

[37] W. Rudin, *Principles of Mathematical Analysis*, McGraw-Hill, Inc., 1976.

[38] M. Sahba, *Globally convergent algorithm for nonlinearly constrained optimization problems*, J. Optim. Theory Appl., 52 (1987), pp. 291–309.

[39] A. Sartenaer, *Automatic determination of an initial trust region in nonlinear programming*, SIAM J. Sci. Comput., 18 (1997), pp. 1788–1803.

[40] R. A. Tapia, *A stable approach to Newton's method for general mathematical programming problems in* $\mathbb{R}^n$, J. Optim. Theory Appl., 14 (1974), pp. 453–476.

[41] R. A. Waltz and J. Nocedal, *KNITRO 2.0 User's Manual*, Report OTC 2002/02, Optimization Technology Center, Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, IL 60208, 2002.

[42] Y. Yuan, *On the convergence of a new trust region algorithm*, Numerische Mathematik, 70 (1995), pp. 515–539.

Table 5.2: We record the number of function and gradient evaluations required, and note that the difference between these values indicates the number of unsuccessful steps attempted; an F indicates that more than 500 evaluations were required and an FQP indicates that the QP solver `QPC` failed.

	SIQP-steer		SEQP-steer		SIQP-seq	
Prob	#fc	#gJ	#fc	#gJ	#fc	#gJ
hs60	7	7	7	7	7	7
hs61	5	5	5	5	14	11
hs62	12	6	8	7	12	6
hs63	7	7	7	7	7	7
hs64	15	15	20	17	23	23
hs65	6	6	6	6	22	16
hs66	4	4	4	4	4	4
hs67	7	7	10	10	14	14
hs68	27	20	38	27	20	16
hs69	24	19	42	30	45	34
hs70	21	17	41	33	21	17
hs71	5	5	5	5	5	5
hs72	16	15	15	14	46	45
hs73	3	3	3	3	3	3
hs74	8	8	8	8	8	8
hs75	8	8	8	8	8	8
hs76	2	2	4	4	2	2
hs77	12	12	12	12	12	12
hs78	4	4	4	4	4	4
hs79	5	5	5	5	5	5
hs80	7	7	8	8	7	7
hs81	6	6	6	6	6	6
hs83	6	6	6	6	16	16
hs84	4	4	4	4	5	5
hs85	8	8	13	12	21	20
hs86	4	4	4	4	4	4
hs88	20	18	25	21	43	41
hs89	23	21	64	46	45	43

	SIQP-steer		SEQP-steer		SIQP-seq	
Prob	#fc	#gJ	#fc	#gJ	#fc	#gJ
hs90	44	33	40	30	42	34
hs91	43	32	32	25	73	61
hs92	35	25	34	27	64	37
hs93	18	15	6	6	F	F
hs95	3	3	2	2	3	3
hs96	3	3	2	2	3	3
hs97	4	4	5	5	4	4
hs98	4	4	5	5	4	4
hs99	5	5	FQP	FQP	75	34
hs100	10	9	13	10	10	9
hs101	34	27	68	40	67	52
hs102	28	21	55	34	44	39
hs103	26	20	27	20	79	62
hs104	17	14	19	14	14	11
hs105	21	14	31	23	21	14
hs106	103	101	122	66	103	101
hs107	6	6	6	6	10	10
hs108	12	9	12	10	242	177
hs109	9	9	9	9	10	10
hs110	9	5	8	6	9	5
hs111	28	23	41	31	23	19
hs112	11	11	50	50	11	11
hs113	5	5	6	6	5	5
hs114	142	142	13	13	136	134
hs116	F	F	F	F	F	F
hs117	10	10	11	11	10	10
hs118	3	3	12	12	3	3
hs119	7	7	8	8	7	7