SIAM J. OPTIM.
Vol. 20, No. 4, pp. 2049–2079

A SECOND DERIVATIVE SQP METHOD: LOCAL CONVERGENCE AND PRACTICAL ISSUES*

NICHOLAS I. M. GOULD† AND DANIEL P. ROBINSON‡

Abstract. Gould and Robinson [*SIAM J. Optim.*, 20 (2010), pp. 2023–2048] proved global convergence of a second derivative SQP method for minimizing the exact ℓ_1-merit function for a fixed value of the penalty parameter. This result required the properties of a so-called Cauchy step, which was itself computed from a so-called predictor step. In addition, they allowed for the additional computation of a variety of (optional) accelerator steps that were intended to improve the efficiency of the algorithm. The main purpose of this paper is to prove that a nonmonotone variant of the algorithm is quadratically convergent for two specific realizations of the accelerator step; this is verified with preliminary numerical results on the Hock and Schittkowski test set. Once fast local convergence is established, we consider two specific aspects of the algorithm that are important for an efficient implementation. First, we discuss a strategy for defining the positive-definite matrix B_k used in computing the predictor step that is based on a limited-memory BFGS update. Second, we provide a simple 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.

Key words. nonlinear programming, nonlinear inequality constraints, sequential quadratic programming, ℓ_1-penalty function, nonsmooth optimization

AMS subject classifications. 49J52, 49M37, 65F22, 65K05, 90C26, 90C30, 90C55

DOI. 10.1137/080744554

1. Introduction.

In [17], we presented a sequential inequality/equality constrained quadratic programming algorithm (an SIQP/SEQP "hybrid") for solving the problem

$$\text{minimize}_{x\in\mathbb{R}^n} \ \phi(x) = f(x) + \sigma\|[c(x)]^-\|_1, \tag{ℓ_1-σ}$$

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 componentwise. The motivation for solving this problem is that solutions of problem (ℓ_1-σ) correspond, under certain assumptions, to solutions of the nonlinear programming problem

$$\text{minimize}_{x\in\mathbb{R}^n} \ f(x) \ \text{ subject to } \ c(x) \geq 0. \tag{NP}$$

Our trust-region algorithm is similar to the original work of Fletcher [12]. One novel difference, however, is that we never require the *global* minimizer of a general indefinite quadratic program (QP). This is accomplished by computing trial steps as the sum of two well-defined steps. The first step is computed from the *predictor* step, which is the *unique* solution to a strictly convex QP. This step drives convergence of the algorithm

*Received by the editors December 22, 2008; accepted for publication (in revised form) December 8, 2009; published electronically March 12, 2010. Research supported by the EPSRC grants EP/E053351/1 and EP/F005369/1.

http://www.siam.org/journals/siopt/20-4/74455.html

†Rutherford Appleton Laboratory, Numerical Analysis Group, Chilton, Didcot, Oxfordshire, OX11 0QX, UK (nick.gould@stfc.ac.uk).

‡University of Oxford, Mathematical Institute, 24–29 St Giles', Oxford, OX1 3LB, UK (robinson@maths.ox.ac.uk).

and is, in fact, the only step required to establish *global* convergence. The second step is called the accelerator step, and its purpose is to enhance global efficiency and improve local convergence. The primary purpose of this paper is to establish fast local convergence rates for two particular instances of the accelerator step suggested in [17, section 2.3].

The first instance is an *equality* constrained QP that relies on an estimate of the set of constraints active at a solution to (ℓ_1-σ), i.e., an optimal active set. A local analysis for this case is required for two reasons. First, the requirements on the Hessian matrix of the QP associated with the predictor step are quite minimal. Thus its ability to accurately predict an optimal active set must be established. Second, it must be shown that this particular accelerator step is accepted by the algorithm in a neighborhood of a solution, since this may be disallowed by *global* convergence criterion. The second accelerator step we consider is derived from an *inequality* constrained QP that uses an artificial so-called descent-constraint whose purpose is to ensure that the accelerator step is not an ascent direction. To establish fast local convergence we must again show that this particular accelerator step will be allowed by the algorithm within a neighborhood of a solution. Additionally, we must prove that the artificial descent-constraint does not interfere with the anticipated fast (quadratic) convergence.

The algorithm we propose has been designed with large problems in mind. Nevertheless, we expect the use of the additional, and carefully constructed, accelerator step to offer significant improvement over traditional quasi-Newton SQP methods for problems of any size.

An outline of this paper is as follows. In section 2 we provide the details of a *nonmonotone* variant of the algorithm proposed in [17]. Section 3 then discusses the local convergence properties of that algorithm and culminates with two rate-of-convergence results. The first applies when the accelerator step is computed from an *equality* constrained subproblem [17, section 2.3.2], while the second applies when the accelerator step is computed from an *inequality* constrained subproblem [17, section 2.3.1]. In section 4 we consider other aspects of the algorithm that relate to efficiency. In particular we address strategies for adjusting the penalty parameter and for defining the positive-definite matrix required for computing the predictor step. Finally, in section 5, we provide preliminary numerical results for the proposed algorithm.

Notation. We let e denote the vector of all ones whose dimension is determined by the context. The gradient of $f(x)$ is $g(x)$ and $\nabla_{xx} f(x)$ its (symmetric) Hessian; the matrix $\nabla_{xx} c_j(x)$ 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 $\nabla_{xx}\mathcal{L}(x, y) = \nabla_{xx} f(x) - \sum_{j=1}^{m} y_j \nabla_{xx} c_j(x)$.

For a general vector v, the notation $[v]^- = \min(0, v)$ is used, where the minimum is understood to be componentwise, and $\mathrm{diag}(v)$ 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 $\mathrm{diag}(A_1, A_2, \ldots, A_p)$ to be the block-diagonal matrix whose ith block is A_i.

Given a solution x^* to problem (NP), we define the indexing sets

$$\mathcal{A} \stackrel{\text{def}}{=} \{i : c_i(x^*) = 0\} \quad \text{and} \quad \mathcal{I} \stackrel{\text{def}}{=} \{i : c_i(x^*) > 0\}, \tag{1.1}$$

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 $[V(x)]_S$.

2. The algorithm. We now describe the details of Algorithm 2.1, which is a *nonmonotone* variant of [17, Algorithm 3.1]. First, the user supplies an initial guess (x_0, y_0) of a solution to problem (ℓ_1-σ). Next, "success" parameters $0 < \eta_s \le \eta_{vs} < 1$, a reset value Δ_{R} and maximum allowed value Δ_{U} for the predictor trust-region radius satisfying $0 < \Delta_{\text{R}} \le \Delta_{\text{U}}$, sufficient model decrease tolerance and approximate Cauchy point tolerance $0 < \eta \le \eta_{\text{ACP}} < 1$, accelerator trust-region radius factor τ_f, maximum number of consecutive failures *max_fails*, and expansion and contraction factors $0 < \eta_c < 1 < \eta_e$ are defined. With parameters set, the main iteration loop begins. First, the problem functions are evaluated at the current point (x_k, y_k). Next, a symmetric positive-definite approximation B_k to $\nabla_{xx}\mathcal{L}(x_k, y_k)$ is defined, and the predictor step s_k^{P} is computed as the *unique* minimizer to

$$\underset{s\in\mathbb{R}^n}{\text{minimize}} \;\; f_k + g_k^T s + \tfrac{1}{2} s^T B_k s + \sigma \|[c_k + J_k s]^-\|_1 \quad \text{subject to} \;\; \|s\|_\infty \le \Delta_k^{\text{P}}, \tag{2.1}$$

where $\Delta_k^{\text{P}} > 0$ is the predictor trust-region radius and we have used the notation $f_k = f(x_k)$, $g_k = g(x_k)$, $c_k = c(x_k)$, and $J_k = J(x_k)$. By introducing elastic variables [13], this problem is equivalent to

$$\underset{s\in\mathbb{R}^n}{\text{minimize}} \;\; f_k + g_k^T s + \tfrac{1}{2} s^T B_k s + \sigma e^T v \;\; \text{subject to} \;\; c_k + J_k s + v \ge 0,\; v \ge 0,\; \|s\|_\infty \le \Delta_k^{\text{P}}. \tag{2.2}$$

Strategies for defining the positive-define matrix B_k are discussed in section 4.1. Next, we define y_k^{F} to be any multiplier estimate that ultimately satisfies the conditions $y_k^{\text{F}} - y^* = O(\|x_k - x^*\|_2)$ and $[y_k^{\text{F}}]_{\mathcal{I}} = 0$; Lemma 3.13 shows that we may choose $y_k^{\text{F}} = y_k^{\text{P}}$, where y_k^{P} is the predictor multiplier vector from (2.2), but this is not required. We prefer using this more flexible requirement on y_k^{F}, since it allows for alternatives, such as least-squares multiplier estimates, to be tested in the future. Next, we define H_k to be any symmetric approximation to $\nabla_{xx}\mathcal{L}(x_k, y_k^{\text{F}})$, but for the local convergence results given by Theorems 3.12 and 3.14 we choose $H_k \equiv \nabla_{xx}\mathcal{L}(x_k, y_k^{\text{F}})$. We may then define the Cauchy step as $s_k^{\text{CP}} = \alpha_k s_k^{\text{P}}$, where α_k is the solution to

$$\underset{0\le\alpha\le 1}{\text{minimize}} \;\; M_k^{H}(\alpha s_k^{\text{P}}) \tag{2.3}$$

for the so-called *faithful* model

$$M_k^{H}(s) \overset{\text{def}}{=} M_k^{H}(s\,; x_k) = f_k + g_k^T s + \tfrac{1}{2} s^T H_k s + \sigma \|[c_k + J_k s]^-\|_1. \tag{2.4}$$

The word "faithful" is used, since we are allowed to choose H_k to be the exact Hessian of the Lagrangian $\nabla_{xx}\mathcal{L}(x_k, y_k^{\text{F}})$ (note that this is generally not allowed for the predictor subproblem (2.1), since $\nabla_{xx}\mathcal{L}(x_k, y_k^{\text{F}})$ is unlikely to be positive definite). Once the Cauchy step has been computed, we compute

$$\Delta M_k^{H}(s) \overset{\text{def}}{=} \Delta M_k^{H}(s\,; x_k) = M_k^{H}(0\,; x_k) - M_k^{H}(s\,; x_k), \tag{2.5}$$

which is the *change* in the faithful model of ϕ. We mention that this notation does not allude to the dependence on the penalty parameter σ. Next we have the option of computing an accelerator step s_k^{A} as the solution of any of the subproblems discussed

in [17, section 2.3]. However, in this paper we will restrict our attention to two particular instances. The first accelerator subproblem [17, section 2.3.2] is given by

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

where $\mathcal{A}(s_k^{\text{P}}) = \{i : [c_k + J_k s_k^{\text{P}}]_i \le 0\}$, $\bar{f}_k = f_k + g_k^T s_k^{\text{P}} + \frac{1}{2} {s_k^{\text{P}}}^T H_k s_k^{\text{P}}$, and $\Delta_k^{\text{S}} > 0$ is the accelerator radius. Since this subproblem defines only multipliers for the constraints whose indices are in the set $\mathcal{A}(s_k^{\text{P}})$, we form accelerator multipliers y_k^{A} by "scattering" the multipliers from subproblem (EQP) into the appropriate locations of a zero-vector of length m. The second accelerator subproblem [17, section 2.3.1] that we will consider is given by

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

where $\bar{f}_k = f_k + g_k^T s_k^{\text{CP}} + \frac{1}{2} {s_k^{\text{CP}}}^T H_k s_k^{\text{CP}}$, z_k is defined componentwise as

$$[z_k]_i = \begin{cases} -1 & \text{if } i \in \mathcal{V}_k, \\ 0 & \text{if } i \in \mathcal{S}_k, \end{cases} \tag{2.6}$$

where $\mathcal{V}_k = \{i : [c_k + J_k s_k^{\text{CP}}]_i < 0\}$ and $\mathcal{S}_k = \{i : [c_k + J_k s_k^{\text{CP}}]_i \ge 0\}$, and $(g_k + H_k s_k^{\text{CP}} + \sigma_k J_k^T z_k)^T s \le 0$ is the so-called descent-constraint. The descent-constraint guarantees that the directional derivative of $M_k^H(s)$ in the direction s_k^{A} is nonpositive. In this case the accelerator multipliers y_k^{A} are the multipliers for the general constraints of the *smooth* variant of problem (EIQP). The trial step computation is completed by defining the full step s_k as

$$\text{(EQP)} \qquad s_k = \begin{cases} s_k^{\text{P}} + s_k^{\text{A}} & \text{if } \Delta M_k^H(s_k^{\text{P}} + s_k^{\text{A}}) \ge \eta \Delta M_k^H(s_k^{\text{CP}}), \\ s_k^{\text{CP}} & \text{otherwise}, \end{cases} \tag{2.7}$$

$$\text{(EIQP)} \qquad s_k = \begin{cases} s_k^{\text{CP}} + s_k^{\text{A}} & \text{if } \Delta M_k^H(s_k^{\text{CP}} + s_k^{\text{A}}) \ge \eta \Delta M_k^H(s_k^{\text{CP}}), \\ s_k^{\text{CP}} & \text{otherwise} \end{cases} \tag{2.8}$$

for some constant $0 < \eta < 1$ independent of k (see [17, section 2.3] for more details). Note that the bound

$$\Delta M_k^H(s_k) \ge \eta \Delta M_k^H(s_k^{\text{CP}}) \tag{2.9}$$

is always satisfied and is the crucial bound used to prove global convergence in [17]. We note that although the accelerator step is optional for proving global convergence, it is generally required for establishing quadratic convergence. We then evaluate $\phi(x_k + s_k)$ and the change in the faithful model at the full step $\Delta M_k^H(s_k)$.

We now must decide whether to accept the trial step s_k. This decision is based on the value of the ratio r_k, which in turn depends on the history of "successful/unsuccessful" steps. If we assume that every iterate is successful in the sense of traditional trust-region strategies, i.e., there is "good agreement" between the actual and predicted decrease in M_k^H, then our nonmonotone algorithm is identical to traditional updating strategies. However, if a failure occurs (in the traditional sense),

then Algorithm 2.1 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 a traditional trust-region updating strategy until the next failure occurs. We also note that our update to the predictor trust-region radius is slightly different than that used in [17, Algorithm 3.1]. The new update ensures that the radius following every successful/very successful iteration is at least as large as some predefined 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 [10, Chapter 15]. Finally, we update the accelerator trust-region radius as $\Delta^{\text{A}}_{k+1} \leftarrow \tau_f \cdot \Delta^{\text{P}}_{k+1}$.

It may easily be verified that [17, Theorem 4.3] is still true with these changes. Therefore, Algorithm 2.1 is globally convergent. We point out that the maximum allowed predictor trust-region radius Δ_{U} and the approximate Cauchy point tolerance η_{ACP} are not relevant for the local convergence analysis of section 3, but they are required to prove *global* convergence [17].

ALGORITHM 2.1 NONMONOTONE ALGORITHM.

Input: (x_0, y_0)

Set parameters $0 < \eta_s \le \eta_{vs} < 1$, $0 < \Delta_{\text{R}} \le \Delta^{\text{P}}_0 \le \Delta_{\text{U}}$, $0 < \eta \le \eta_{\text{ACP}} < 1$, and $\tau_f \ge 1$.

Set expansion and contraction factors $0 < \eta_c < 1 < \eta_e$ and fail counter $fails = 0$.

Set nonmonotone parameters $0 \le max\_fails \in \mathbb{N}$ and $fails = 0$.

$k \leftarrow 0$

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 $\nabla_{xx}\mathcal{L}(x_k, y_k)$.

Solve problem (2.1) for predictor step and multipliers $(s^{\text{P}}_k, y^{\text{P}}_k)$.

Define multiplier estimate y^{F}_k such that $y^{\text{F}}_k - y^* = O(\|x_k - x^*\|_2)$ and $[y^{\text{F}}_k]_{\mathcal{I}} = 0$.

Define H_k to be a symmetric approximation to $\nabla_{xx}\mathcal{L}(x_k, y^{\text{F}}_k)$.

Solve problem (2.3) for s^{CP}_k and compute $\Delta M^{H}_k(s^{\text{CP}}_k)$.

Compute an accelerator step and multipliers $(s^{\text{A}}_k, y^{\text{A}}_k)$ (optional).

Define a full step s_k that satisfies condition (2.9).

Evaluate $\phi(x_k + s_k)$ and $\Delta M^{H}_k(s_k)$.

if $fails = 0$, **then**

$r_k \leftarrow \big(\phi(x_k) - \phi(x_k + s_k)\big)/\Delta M^{H}_k(s_k)$ [standard definition]

else

$r_k \leftarrow \big(\phi_{\text{R}} - \phi(x_k + s_k)\big)/\Delta^{\text{H}}_{\text{R}}$ [change in ϕ based on point x_{R}]

end if

if $r_k \ge \eta_{vs}$, **then** [very successful]

$x_{k+1} \leftarrow x_k + s_k$

$y_{k+1} \leftarrow y^{\text{A}}_k$ ($y_{k+1} \leftarrow y^{\text{F}}_k$ if accelerator step not computed)

$\Delta^{\text{P}}_{k+1} \leftarrow \min\big(\max(\eta_e \cdot \Delta^{\text{P}}_k, \Delta_{\text{R}}), \Delta_{\text{U}}\big)$ [increase Δ^{P}_k and ensure $\Delta^{\text{P}}_k \ge \Delta_{\text{R}}$]

$fails \leftarrow 0$

else if $r_k \geq \eta_s$, **then** [successful]
  $x_{k+1} \leftarrow x_k + s_k$
  $y_{k+1} \leftarrow y_k^{\mathrm{A}}$ ($y_{k+1} \leftarrow y_k^{\mathrm{F}}$ if accelerator 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}]
  $\mathit{fails} \leftarrow 0$
else [failure]
  if $\mathit{fails} \leq \mathit{max\_fails}$, **then**
    if $\mathit{fails} = 0$, **then** [save current point]
      $x_{\mathrm{R}} \leftarrow x_k, \quad y_{\mathrm{R}} \leftarrow y_k, \quad \phi_{\mathrm{R}} \leftarrow \phi_k, \quad \Delta_{\mathrm{R}}^{\mathrm{H}} \leftarrow \Delta M_k^H(s_k)$
      $\Delta_{\mathrm{R}}^{\mathrm{P}} \leftarrow \Delta_k^{\mathrm{P}}$
      $\Delta_{k+1}^{\mathrm{P}} \leftarrow \eta_c \Delta_k^{\mathrm{P}}$ (optional)
    end if
    $x_{k+1} \leftarrow x_k + s_k$
    $\Delta_{k+1}^{\mathrm{P}} \leftarrow \Delta_k^{\mathrm{P}}$
    $\mathit{fails} \leftarrow \mathit{fails} + 1$
  else [revert to saved point]
    $x_{k+1} \leftarrow x_{\mathrm{R}}, \quad y_{k+1} \leftarrow y_{\mathrm{R}}$
    $\Delta_{k+1}^{\mathrm{P}} \leftarrow \eta_c \Delta_{\mathrm{R}}^{\mathrm{P}}$ [decrease Δ_k^{P}]
    $\mathit{fails} \leftarrow 0$
  end if
end if
$\Delta_{k+1}^{\mathrm{A}} \leftarrow \tau_f \cdot \Delta_{k+1}^{\mathrm{P}}$ [update accelerator radius]
$k \leftarrow k + 1$
end do

3. Local convergence. In this section we show that Algorithm 2.1 is superlinearly convergent if an accelerator step is computed from either subproblem (EQP) or (EIQP). 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 $w^* = (x^*, y^*)$, $w_k^{\mathrm{P}} = (x_k^{\mathrm{P}}, y_k^{\mathrm{P}})$ for the solution to the predictor subproblem (2.2), and $w_k^{\mathrm{A}} = (x_k^{\mathrm{A}}, y_k^{\mathrm{A}})$ for the solution of the accelerator subproblem (the precise definition of y_k^{A} depends on which accelerator subproblem is used).

The main result of this section is that if a successful iterate of Algorithm 2.1 gets close enough to a local minimizer w^* of problem (NP) that satisfies the strong second-order sufficient conditions, and if we compute an accelerator step from either subproblem (EQP) or (EIQP), then the sequence of iterates converges to w^* with the same convergence properties as Newton's method for zero-finding applied to the function

$$F_N(x, y_{\mathcal{A}}) = \begin{pmatrix} g(x) - J_{\mathcal{A}}(x)^T y_{\mathcal{A}} \\ c_{\mathcal{A}}(x) \end{pmatrix}, \tag{3.1}$$

where the indexing set $\mathcal{A}$ is defined in (1.1). 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 accelerator 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 2.1 is a nonmonotone approach, the analysis given by Conn, Gould, and Toint [10, 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.

In the following definitions related to a solution of problem (NP), we use the notation $c^* = c(x^*)$, $g^* = g(x^*)$, and $J^* = J(x^*)$.

DEFINITION 3.1 (first-order KKT point). *We say that the point (x^*, y^*) is a first-order KKT point for problem* (NP) *if*

$$g^* - J^{*T} y^* = 0, \quad c^* \geq 0, \quad y^* \geq 0, \quad \text{and } c^* \cdot y^* = 0. \tag{3.2}$$

DEFINITION 3.2 (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 $\lambda^H_{\min} > 0$ such that $s^T \nabla_{xx} \mathcal{L}(x^*, y^*) s \geq \lambda^H_{\min} s^T s$ for all s satisfying $J^*_{\mathcal{A}} s = 0$.*

DEFINITION 3.3 (strict complementarity). *We say that strict complementarity holds at a KKT point (x^*, y^*) for problem* (NP) *if $y^*_{\mathcal{A}} > 0$.*

DEFINITION 3.4 (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 3.5. *We say that the* strong *second-order sufficient conditions hold at a point (x^*, y^*) if it satisfies Definitions* 3.1–3.4.

In the ensuing analysis we prove results that hold in a neighborhood of a solution to problem (NP). It is thus convenient for us to 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, for a given vector $v \in \mathbb{R}^n$ and scalar ε.

3.1. 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 $\nabla_{xx}\mathcal{L}(x^*, y^*)$; this certainly occurs if $H_k \equiv \nabla_{xx}\mathcal{L}(x_k, y_k)$ or $H_k \equiv \nabla_{xx}\mathcal{L}(x_k, y^{\text{F}}_k)$. 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 cannot expect the positive-definite matrix B_k to converge to $\nabla_{xx}\mathcal{L}(x^*, y^*)$, since $\nabla_{xx}\mathcal{L}(x^*, y^*)$ 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 x^* through the indexing set $\mathcal{A}$ defined by (1.1):

$$S(x\,; x^*) = \left\{ M = M^T \in \mathbb{R}^{n \times n} : \beta_{\max} \geq \frac{s^T M s}{s^T s} \geq \lambda_{\min} \quad \text{for all } s \text{ satisfying } J_{\mathcal{A}}(x) s = 0 \right\} \tag{3.3}$$

and

$$S_\varepsilon = \bigcup_{w \in \bar{\mathcal{B}}_\varepsilon(w^*)} S(x\,; x^*) \tag{3.4}$$

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

LEMMA 3.6. *If w^* is a KKT point for problem* (NP) *that satisfies the* LICQ, *then*

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

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

is compact;

(ii) *if $\nu_1 \le \nu_2$, 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 $s^T M s \ge (\lambda_{\min}/2) s^T s$ 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 $\varepsilon_2 \le \varepsilon_1$ 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*

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

are nonsingular and satisfy

$$\|\bar{K}_M(x)^{-1}\|_2 \le \beta_0 \quad \text{and} \tag{3.5a}$$

$$\|K_M(w)^{-1}\|_2 \le \beta, \tag{3.5b}$$

where the indexing set $\mathcal{I}$ is defined in (1.1);

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

$$s = O(\|x - x^*\|_2), \quad \pi_{\mathcal{A}} - y^*_{\mathcal{A}} = O(\|x - x^*\|_2), \quad \text{and} \quad \pi - y^* = O(\|x - x^*\|_2), \tag{3.6}$$

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

$$\bar{K}_M(x) \begin{pmatrix} s \\ -\pi_{\mathcal{A}} \end{pmatrix} \equiv \begin{pmatrix} M & J_{\mathcal{A}}(x)^T \\ J_{\mathcal{A}}(x) & 0 \end{pmatrix} \begin{pmatrix} s \\ -\pi_{\mathcal{A}} \end{pmatrix} = - \begin{pmatrix} g(x) \\ c_{\mathcal{A}}(x) \end{pmatrix} \tag{3.7}$$

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) of Lemma 3.6. Since it is clear that S_ε is bounded, we show only 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 $v^T M_k v \ge \lambda_{\min} v^T v$ 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, [7, 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 $\|\bar{M}\|_2 \le \beta_{\max}$, we have $\bar{M} \in S(\bar{x}\,; x^*) \subseteq S_\varepsilon$. Thus, S_ε is closed.

Part (ii) of Lemma 3.6 follows immediately from the definitions of S_{ν_1} and S_{ν_2}.

We now prove part (iii) of Lemma 3.6. If part (iii) of Lemma 3.6 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) of Lemma 3.6, 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

(3.8)
$$\lim_{k\in K_2} w_k = w^*, \quad \lim_{k\in K_2} M_k = M^*, \quad \lim_{k\in K_2} s_k = s^*, \quad J_{\mathcal{A}}^* s^* = 0, \quad \textit{and} \quad {s^*}^T M^* s^* \le \lambda_{\min}/2.$$

Since $M_k \in S_{\delta_k}$ and $\{\delta_k\} \to 0$, there also exists a sequence $\{\widehat{x}_k\} \to x^*$ such that $s^T M_k s \ge \lambda_{\min} s^T s$ 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 (3.8), and thus part (iii) of Lemma 3.6 must be true.

To show that part (iv) of Lemma 3.6 holds, we first note that strict complementarity and the LICQ imply that there exists a number ε_s such that $0 < \varepsilon_s \le \varepsilon_1$ and

(3.9)
$$c_i(x) \ge \tfrac{1}{2} c_i^* > 0 \ \text{ for } \ i \in \mathcal{I}, \ \ y_i \ge \tfrac{1}{2} y_i^* > 0 \ \text{ for } \ i \in \mathcal{A}, \quad \text{and} \ \ J_{\mathcal{A}}(x) \ \text{ has full row rank}$$

for all $w \in \mathcal{B}_{\varepsilon_s}(w^*)$. Under the current assumptions, it follows from parts (ii) and (iii) of Lemma 3.6, and [1, Lemma 1.27] that

$$\text{the matrix } \ \bar{K}_M(x) \ \text{ is nonsingular for all } \ w \in \mathcal{B}_{\varepsilon_s}(w^*) \ \text{ and } \ M \in S_{\varepsilon_s}. \tag{3.10}$$

Assume that (3.5a) does not hold for any $\varepsilon_2 \le \varepsilon_s$ so that there exists a monotonically decreasing sequence $\{\delta_k\} \to 0$ such that $0 < \delta_k \le \varepsilon_s$ and associated sequences $\{w_k\} \in \mathcal{B}_{\delta_k}(w^*)$ and $\{M_k\} \in S_{\delta_k} \subseteq S_{\varepsilon_s}$ such that

$$\|\bar{K}_{M_k}(x_k)^{-1}\|_2 \ge k \quad \text{for all} \quad k \ge 0. \tag{3.11}$$

Since $\{\delta_k\} \to 0$ and S_{ε_s} is compact as a result of part (i) of Lemma 3.6, 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 (3.10) that $\bar{K}_{M^*}(x^*)$ is nonsingular. Since $\lim_{k\in K_3} \bar{K}_{M_k}(x_k) = \bar{K}_{M^*}(x^*)$, [14, 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 (3.11). Thus, (3.5a) holds for some $\varepsilon_1 \ge \varepsilon_s \ge \varepsilon_2 > 0$ and $\beta_0 > 0$. It also follows from (3.9) 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 (3.5b) holds for ε_2. Let $w \in \mathcal{B}_{\varepsilon_2}(w^*)$ and $M \in S_{\varepsilon_2}$. Equation (3.9) implies that the matrices

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

and $N_S = \operatorname{diag}(I, -I, I)$ are *nonsingular*; they satisfy $N_F K_M(w) N_S = N_M$ so that $\|K_M(w)^{-1}\|_2 \le \|N_M^{-1}\|_2 \|N_F\|_2$. It is also clear from (3.9) 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 $\|\bar{K}_M(x)^{-1}\|_2$, since (3.9) guarantees that $c_{\mathcal{I}}(x) \ge \frac{1}{2} c_{\mathcal{I}}^* > 0$ (componentwise). The result follows from (3.5a), and, therefore, there exists a number $\beta > 0$ such that $\|K_M(w)^{-1}\|_2 \le \beta$ for all $w \in \mathcal{B}_{\varepsilon_2}(w^*)$ and $M \in S_{\varepsilon_2}$.

Finally, we prove part (v) of Lemma 3.6. Let $w_k \in \mathcal{B}_{\varepsilon_2}(w^*)$ and $M \in S_{\varepsilon_2}$. Since $c_{\mathcal{A}}^* = 0$, it follows that system (3.7) is equivalent to

$$\begin{pmatrix} M & J_{\mathcal{A}}(x)^T \\ J_{\mathcal{A}}(x) & 0 \end{pmatrix} \begin{pmatrix} s \\ y_{\mathcal{A}}^* - \pi_{\mathcal{A}} \end{pmatrix} = - \begin{pmatrix} g(x) - J_{\mathcal{A}}(x)^T y_{\mathcal{A}}^* \\ c_{\mathcal{A}}(x) - c_{\mathcal{A}}^* \end{pmatrix}. \tag{3.12}$$

Inequality (3.5a), 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

$$F_{\text{KKT}}(w) = \begin{pmatrix} g(x) - J(x)^T y \\ c(x) \cdot y \end{pmatrix}. \tag{3.13}$$

LEMMA 3.7. *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*

$$\begin{array}{ll} \underset{x\in\mathbb{R}^n}{\text{minimize}} & \frac{1}{2}(x - x_k)^T M(x - x_k) + g_k^T(x - x_k) \\ \textit{subject to} & c_k + J_k(x - x_k) \ge 0 \end{array} \tag{3.14}$$

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* (3.14) *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 3.6. Given any vector-pair $(w, \bar{w})$ and symmetric matrix M, we define

$$F_M(w\,;\bar{w}) = \begin{pmatrix} M(x - \bar{x}) + g(\bar{x}) - J(\bar{x})^T y \\ \big(c(\bar{x}) + J(\bar{x})(x - \bar{x})\big) \cdot y \end{pmatrix}. \tag{3.15}$$

Differentiating (3.15) 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

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

since optimality conditions at w^* imply $c^*_{\mathcal{A}} = 0$ and $y^*_{\mathcal{I}} = 0$. It follows from (3.5a) with the choice $w = w^*$ that the matrix $F'_M(w^*\,;w^*)$ is nonsingular and satisfies

$$\|F'_M(w^*\,;w^*)^{-1}\| \le \beta \ \text{ for all } \ M \in S_{\varepsilon_2}. \tag{3.16}$$

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 $\mu < \mu/2 \le \varepsilon_2 \le \varepsilon_1$, it follows from parts (ii) and (iii) of Lemma 3.6 that $J_{\mathcal{A}}(x_k)$ has full row rank and that estimate (3.16) holds for M; thus part (iv) of Lemma 3.7 is true. Using the argument by Robinson [29, 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

$$\bar{\mathcal{B}}_{\mu/2}(w_k) \subset \mathcal{B}_{\mu}(w^*), \tag{3.17}$$

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

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

Define the function

$$T_M(w) = w - F'_M(w^*\,;w^*)^{-1} F_M(w\,;w_k) \tag{3.18}$$

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

$$\|T'_M(w)\|_2 \le \beta \|F'_M(w^*\,;w^*) - F'_M(w\,;w_k)\|_2 \le \tfrac{1}{2} \quad \big(\text{use (3.16), (3.17), and C3}\big)$$

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

$$\|T_M(w_k) - w_k\|_2 \le \beta \|F_M(w_k\,;w_k)\|_2 \quad \big(\text{use (3.18) and (3.16)}\big). \tag{3.19}$$

Using the triangle inequality, the fact that $T_M(w)$ is a contraction with contraction factor $1/2$, (3.19), and the assumption that $4\beta\|F_M(w_k\,;w_k)\|_2 \le \mu$, we have that for all $w \in \bar{\mathcal{B}}_{\mu/2}(w_k)$ the estimate

$$\begin{aligned} \|T_M(w) - w_k\|_2 &\le \|T_M(w) - T_M(w_k)\|_2 + \|T_M(w_k) - w_k\|_2 \\ &\le \tfrac{1}{2}\|w - w_k\|_2 + \beta\|F_M(w_k\,;w_k)\|_2 \le \tfrac{\mu}{2}, \end{aligned}$$

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 [31, 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

$$\begin{aligned} \|x_k(M) - x_k\|_\infty \le \|x_k(M) - x_k\|_2 &\le \|w_k(M) - w_k\|_2 \quad \text{(use norm inequalities)} \\ &\le 2\|T_M(w_k) - w_k\|_2 \quad \text{(estimate from fixed-point theorem)} \\ &\le 2\beta\|F_M(w_k\,;w_k)\|_2 \quad \text{(use (3.19))}, \end{aligned}$$

which proves part (i) of Lemma 3.7. Since $w_k(M)$ is a fixed point for $T_M(w)$, (3.18) implies that

$$F_M(w_k(M)\,;w_k) = 0. \tag{3.20}$$

Thus $w_k(M)$ satisfies the *equality* conditions for being a first-order KKT point for problem (3.14). We now show that the point $w_k(M)$ is actually a first-order KKT point for problem (3.14). 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 (3.20) implies $[c_k + J_k\big(x_k(M) - x_k\big)]_i = 0$; if $c_i^* > 0$, then C1 implies $[c_k + J_k\big(x_k(M) - x_k\big)]_i > 0$ and then (3.20) 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 (3.14) 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 3.6. 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 $\bar{\mathcal{B}}_{\mu/2}(w_k)$. ∎

3.2. 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 2.1 has a fast rate-of-convergence. We use the following definition.

DEFINITION 3.8. *Given a vector $v \in \mathbb{R}^n$ and a subspace $\mathcal{V} \subseteq \mathbb{R}^n$, we define*

$$\theta(v,\mathcal{V}) = \begin{cases} \tan^{-1}\left(\|v_{\scriptscriptstyle R}\|_2/\|v_{\scriptscriptstyle N}\|_2\right) & \text{if } \|v_{\scriptscriptstyle N}\|_2 \neq 0, \\ \pi/2 & \text{otherwise,} \end{cases} \qquad 0 \leq \theta \leq \pi/2, \tag{3.21}$$

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 3.9. *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*

$$\theta\Big(s, \operatorname{null}\big(J_{\mathcal{A}}(x)\big)\Big) \leq \bar{\theta} \stackrel{\text{def}}{=} \min\left(\frac{\pi}{4}, \tan^{-1}\left(\frac{\lambda_{\min}}{24\beta_{\max}}\right)\right), \tag{3.22}$$

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

Proof. Let ε_2 be defined as in part (iv) of Lemma 3.6 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 (3.22). If we write $s = s_{\rm N} + s_{\rm R}$ for $s_{\rm N} \in \operatorname{null}(J_{\mathcal{A}}(x))$ and $s_{\rm R} \in \operatorname{Range}(J_{\mathcal{A}}(x)^T)$, it follows from (3.21) and (3.22) that $\theta = \theta(s, \operatorname{null}\big(J_{\mathcal{A}}(x)\big))$ satisfies

$$\frac{\|s_{\rm R}\|_2}{\|s_{\rm N}\|_2} = \tan(\theta) \leq 1. \tag{3.23}$$

Using the orthogonal decomposition of s, parts (ii) and (iii) of Lemma 3.6, the Cauchy–Schwarz inequality, the definition of $\beta_{\max}$, and (3.23) and (3.22), we have

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

which completes the proof. ∎

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

$$\underset{s\in\mathbb{R}^n}{\text{minimize}}\ \ g(x)^T s + \tfrac{1}{2} s^T M s \quad \text{subject to} \quad c_{\mathcal{A}}(x) + J_{\mathcal{A}}(x)s = 0 \tag{3.24}$$

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

LEMMA 3.10. *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* (3.24) *is well defined and the solution s_T satisfies*

$$\big(g(x) + \sigma J(x)^T z\big)^T s_T < -c_2 \|s_T\|_2^2 \quad \textit{for} \quad z = \begin{cases} 0 & \textit{if } c(x) \geq 0, \\ -1 & \textit{otherwise.} \end{cases} \tag{3.25}$$

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

$$y^*_{\mathcal{A}} \geq \kappa_{\text{S}} e > 0. \tag{3.26}$$

We define $\bar{\theta}$ as in Lemma 3.9 and choose positive scalars κ_{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 3.9;
2. $\|J(x)^T\|_2 \|(J(x)J(x)^T)^{-1}\|_2 \leq \kappa_{\text{J}}$;
3. the system

$$\begin{pmatrix} M & J_{\mathcal{A}}(x)^T \\ J_{\mathcal{A}}(x) & 0 \end{pmatrix} \begin{pmatrix} s \\ -q \end{pmatrix} = - \begin{pmatrix} g(x) - J_{\mathcal{A}}(x)^T y_{\mathcal{A}} \\ c_{\mathcal{A}}(x) \end{pmatrix} \tag{3.27}$$

has a unique solution (s, q) that satisfies

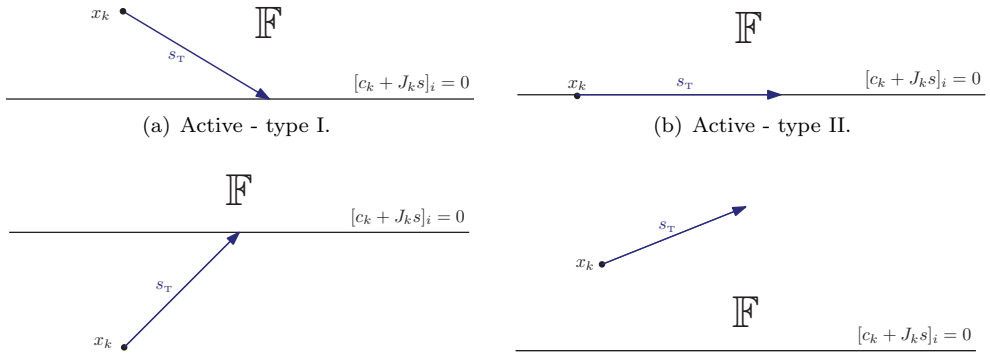
a. $(\kappa_{\text{S}}/2)e \leq y_{\mathcal{A}} + q \leq \sigma(1 - \kappa_\sigma)e$ for some $\kappa_\sigma > 0$;
b. $\|s\|_2 \leq \min(1, c_1)$, where

$$c_1 = \frac{\kappa \sin(\bar{\theta})}{2\kappa_{\text{J}} \beta_{\max}} > 0 \quad \text{and} \quad \kappa = \min\Big(\frac{\kappa_{\text{S}}}{2}, \sigma\kappa_\sigma\Big) > 0; \text{ and} \tag{3.28}$$

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 3.9 hold. Since $\varepsilon_3 \leq \varepsilon_2$, parts (ii), (iii), and (iv) of Lemma 3.6 combined with [1, Lemma 1.27] guarantee that problem (3.24) has a unique solution, say s_{T}, and the optimality conditions show that $(s_{\text{T}}, q_{\text{T}})$ satisfies system (3.27), where q_{T} is the *step* from y to the Lagrange multiplier vector for problem (3.24). Note that we can make the solution $(s_{\text{T}}, q_{\text{T}})$ arbitrarily small in norm, since the target vector in system (3.27) converges to zero as w converges to w^*. This observation, (3.26), 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 $(s_{\text{T}}, q_{\text{T}})$ denote the solution to problem (3.24) so that it satisfies system (3.27). For convenience we "scatter" the vector q_{T}, which has length equal to the size of the indexing set $\mathcal{A}$, into a vector $q_{\text{TF}} \in \mathbb{R}^m$ so that $[q_{\text{TF}}]_i = 0$ if $i \notin \mathcal{A}$. We also partition the constraints up into four types: I, II, III, and IV (see Figure 1); condition 3c and the properties of s_{T} guarantee that these are the only possibilities. Note that $\nabla c_i(x)^T s_{\text{T}} < 0$ for $i \in$ I, $\nabla c_i(x)^T s_{\text{T}} = 0$ for $i \in$ II, and $\nabla c_i(x)^T s_{\text{T}} > 0$ for $i \in$ III. It then follows from system (3.27), the definitions of q_{TF}



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

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

FIG. 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 $\nabla c_i(x_k)^T s_T < 0$. (b) For type* II*, we have $c_i(x_k) = 0$, $c_i^* = 0$, and $\nabla c_i(x_k)^T s_T = 0$. (c) For type* III*, we have $c_i(x_k) < 0$, $c_i^* = 0$, and $\nabla c_i(x_k)^T s_T > 0$. (d) For type* IV*, we have $c_i^* > 0$.*

and z, condition 3a, and the definition of κ that

$$\begin{aligned}
&\big(g(x)+ \sigma J(x)^T z\big)^T s_{\mathrm{T}} \\
&\quad = -{s_{\mathrm{T}}}^T M s_{\mathrm{T}} + (J_{\mathcal{A}}(x)s_{\mathrm{T}})^T[y + q_{\mathrm{TF}}]_{\mathcal{A}} + \sigma z^T(J(x)s_{\mathrm{T}}) \\
&\quad = -{s_{\mathrm{T}}}^T M s_{\mathrm{T}} + \sum_{i\in\mathrm{I}}(\nabla c_i(x)^T s_{\mathrm{T}})[y + q_{\mathrm{TF}}]_i + \sum_{i\in\mathrm{III}}(\nabla c_i(x)^T s_{\mathrm{T}})[y + q_{\mathrm{TF}} - \sigma e]_i \\
&\quad \le -{s_{\mathrm{T}}}^T M s_{\mathrm{T}} + \frac{\kappa_{\mathrm{S}}}{2}\sum_{i\in\mathrm{I}}(\nabla c_i(x)^T s_{\mathrm{T}}) - \sigma\kappa_\sigma \sum_{i\in\mathrm{III}}(\nabla c_i(x)^T s_{\mathrm{T}}) \\
&\quad \le -{s_{\mathrm{T}}}^T M s_{\mathrm{T}} - \kappa \sum_{i\in\mathrm{I}\cup\mathrm{II}\cup\mathrm{III}} |\nabla c_i(x)^T s_{\mathrm{T}}| \\
&\quad = -{s_{\mathrm{T}}}^T M s_{\mathrm{T}} - \kappa\|J_{\mathcal{A}}(x)s_{\mathrm{T}}\|_1. \qquad (3.29)
\end{aligned}$$

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

If we let $s_{\mathrm{T}} = s_{\mathrm{T}}^{\mathrm{R}} + s_{\mathrm{T}}^{\mathrm{N}}$ be the orthogonal decomposition of s_{T} such that $s_{\mathrm{T}}^{\mathrm{R}} \in \mathrm{Range}(J_{\mathcal{A}}(x)^T)$ and $s_{\mathrm{T}}^{\mathrm{N}} \in \mathrm{null}(J_{\mathcal{A}}(x))$, then it follows that there exists a vector r such that $J_{\mathcal{A}}(x)^T r = s_{\mathrm{T}}^{\mathrm{R}}$, and, therefore,

$$\|s_{\mathrm{T}}^{\mathrm{R}}\|_2 \le \big\|J_{\mathcal{A}}(x)^T\big\|_2 \|r\|_2 \quad \text{and} \quad J_{\mathcal{A}}(x)s_{\mathrm{T}} = J_{\mathcal{A}}(x)s_{\mathrm{T}}^{\mathrm{R}} = J_{\mathcal{A}}(x)J_{\mathcal{A}}(x)^T r. \qquad (3.30)$$

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

$$\|r\|_2 \le \Big\|\big(J_{\mathcal{A}}(x)J_{\mathcal{A}}(x)^T\big)^{-1}\Big\|_2 \|J_{\mathcal{A}}(x)s_{\mathrm{T}}\|_2. \qquad (3.31)$$

This inequality, (3.30), and condition 2 imply

$$\|J_{\mathcal{A}}(x)s_{\mathrm{T}}\|_2 \ge \frac{\|r\|_2}{\Big\|\big(J_{\mathcal{A}}(x)J_{\mathcal{A}}(x)^T\big)^{-1}\Big\|_2} \ge \frac{\|s_{\mathrm{T}}^{\mathrm{R}}\|_2}{\Big\|\big(J_{\mathcal{A}}(x)J_{\mathcal{A}}(x)^T\big)^{-1}\Big\|_2 \|J_{\mathcal{A}}(x)^T\|_2} \ge \frac{\|s_{\mathrm{T}}^{\mathrm{R}}\|_2}{\kappa_{\mathrm{J}}}. \qquad (3.32)$$

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

$$\|J_{\mathcal{A}}(x)s_{\mathrm{T}}\|_1 \ge \|J_{\mathcal{A}}(x)s_{\mathrm{T}}\|_2 \ge \big(\sin(\theta)\|s_{\mathrm{T}}\|_2\big)/\kappa_{\mathrm{J}}. \qquad (3.33)$$

Combining this with (3.29) we have

$$\big(g(x)+\sigma J(x)^T z\big)^T s_{\mathrm{T}} \le -{s_{\mathrm{T}}}^T M s_{\mathrm{T}} - \big(\kappa\sin(\theta)\|s_{\mathrm{T}}\|_2\big)/\kappa_{\mathrm{J}}. \tag{3.34}$$

We consider two cases. First suppose that ${s_{\mathrm{T}}}^T M s_{\mathrm{T}} \ge (\lambda_{\min}/8){s_{\mathrm{T}}}^T s_{\mathrm{T}}$. Then it immediately follows from (3.34) that

$$\big(g(x)+\sigma J(x)^T z\big)^T s_{\mathrm{T}} \le -(\lambda_{\min}/8)\|s_{\mathrm{T}}\|_2^2. \tag{3.35}$$

Next, suppose that ${s_{\mathrm{T}}}^T M s_{\mathrm{T}} < (\lambda_{\min}/8){s_{\mathrm{T}}}^T s_{\mathrm{T}}$. Lemma 3.9 then implies that $0<\bar\theta<\theta$, and, therefore, $0<\sin(\bar\theta)<\sin(\theta)$. We can then use this fact, (3.34), the Cauchy–Schwarz inequality, the definition of $\beta_{\max}$, and condition 3b to conclude that

$$\begin{aligned}\big(g(x)+\sigma J(x)^T z\big)^T s_{\mathrm{T}} &\le \|s_{\mathrm{T}}\|_2^2\|M\|_2 - (\kappa\sin(\bar\theta)\|s_{\mathrm{T}}\|_2)/\kappa_{\mathrm{J}}\\ &\le \|s_{\mathrm{T}}\|_2\left(\beta_{\max}\|s_{\mathrm{T}}\|_2 - (\kappa\sin(\bar\theta))/\kappa_{\mathrm{J}}\right)\\ &\le -(\kappa\sin(\bar\theta)/2\kappa_{\mathrm{J}})\|s_{\mathrm{T}}\|_2 \le -(\kappa\sin(\bar\theta)/2\kappa_{\mathrm{J}})\|s_{\mathrm{T}}\|_2^2.\end{aligned} \tag{3.36}$$

If we define

$$c_2 = \min\left(\frac{\lambda_{\min}}{8},\ \frac{\kappa\sin(\bar\theta)}{2\kappa_{\mathrm{J}}}\right) > 0, \tag{3.37}$$

then it follows from (3.35) and (3.36) that

$$\big(g(x)+\sigma J(x)^T z\big)^T s_{\mathrm{T}} \le -c_2\|s_{\mathrm{T}}\|_2^2, \tag{3.38}$$

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 (3.24) 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 3.1 suggest that we make the following assumption.

Assumption 3.1. There exists a number $\lambda_{\min}^B > 0$ such that the sequence of positive-definite matrices $\{B_k\}$ defined in Algorithm 2.1 satisfies

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

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

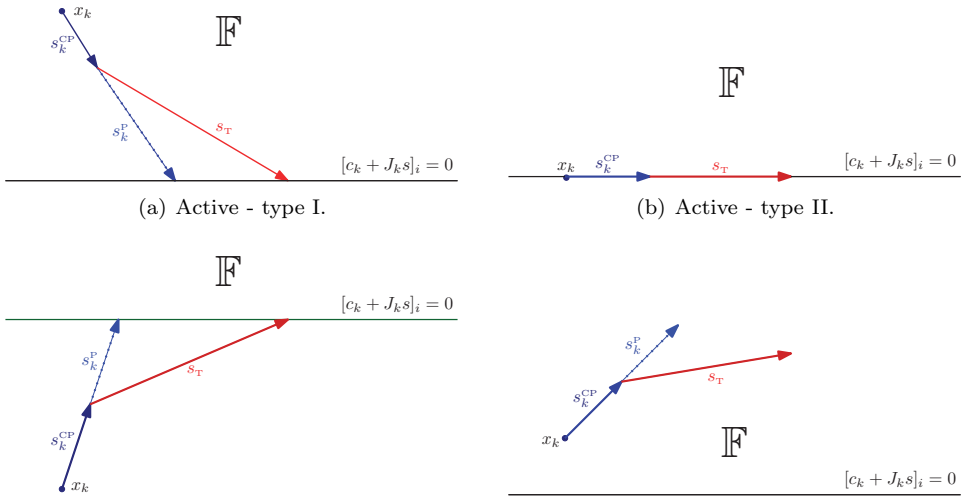
$$\underset{s\in\mathbb{R}^n}{\text{minimize}}\ \ \big(g_k + M s_k^{\mathrm{CP}}\big)^T s + \tfrac{1}{2}s^T M s \quad \text{subject to} \quad c_{\mathcal{A}}(x_k) + J_{\mathcal{A}}(x_k)(s_k^{\mathrm{CP}}+s) = 0 \tag{3.39}$$

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

LEMMA 3.11. *Let w^* be a minimizer for problem* (NP) *that satisfies the* LICQ *and strict complementarity, and assume that $\sigma_k > \|y^*\|_\infty$, that Assumption* 3.1 *holds, and that $\|B_k\|_2 \le b_B$ 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* (3.39) *is well defined and the solution s_T satisfies*

$$(g_k + M s_k^{CP} + \sigma_k J_k^T z_k)^T s_T < -c_2\|s_T\|_2^2 \quad \textit{for} \quad [z_k]_i = \begin{cases} 0 & \textit{if } i\in\mathcal{V}_k,\\ -1 & \textit{if } i\in\mathcal{S}_k,\end{cases} \tag{3.40}$$

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



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

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

FIG. 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* $\nabla c_i(x_k)^T s_T < 0$. (b) *For type* II, *we have* $c_i(x_k) = 0$, $c_i^* = 0$, *and* $\nabla c_i(x_k)^T s_T = 0$. (c) *For type* III, *we have* $c_i(x_k) < 0$, $c_i^* = 0$, *and* $\nabla c_i(x_k)^T s_T > 0$. (d) *For type* IV, *we have* $c_i^* > 0$.

Proof. Since the proof is very similar to Lemma 3.10, we point out only 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 (2.1) is equivalent to problem (3.14) for the choice $M = B_k$, provided that the trust-region constraint is inactive. Third, Lemma 3.7 shows that the solution to problem (3.14) with $M = B_k$ correctly identifies the optimal active set if w_k is sufficiently close to w^*, so that the solution satisfies system (3.7). Equation (3.6) then shows that we can make the solution to problem (3.14) 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 2.1). Combining all of this together, we know that there exists a positive number $\varepsilon_4 < \mu/2$ (μ is defined in Lemma 3.7) 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 2), and $\|s_k^{\rm P}\|_2$ is as small as we wish. The system that arises in place of (3.27) is

$$\begin{pmatrix} M & J_{\mathcal{A}}(x)^T \\ J_{\mathcal{A}}(x) & 0 \end{pmatrix} \begin{pmatrix} s \\ -q \end{pmatrix} = - \begin{pmatrix} g(x) - J_{\mathcal{A}}(x)^T y_{\mathcal{A}} + M s_k^{\rm CP} \\ c_{\mathcal{A}}(x) + J_{\mathcal{A}}(x_k) s_k^{\rm CP} \end{pmatrix}, \tag{3.41}$$

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 3.10 are once again satisfied. The rest of the proof is identical to Lemma 3.10. ∎

3.3. Local convergence with an (EQP) step. Our first rate-of-convergence result for Algorithm 2.1 assumes that the accelerator step is computed from subproblem (EQP) as discussed in [17, section 2.3.2] and restated on page 2052.

THEOREM 3.12 ((EQP) local convergence result). *Let* $w^* = (x^*, y^*)$ *be a minimizer for problem* (NP) *that satisfies the strong second-order sufficient conditions as given by Definition* 3.5. *Let Assumption* 3.1 *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 accelerator step is computed from subproblem* (EQP) *with the choice* $H_k \equiv \nabla_{xx}\mathcal{L}(x_k, y_k^{\text{F}})$*, and* max_fails ≥ 1 *in Algorithm* 2.1*. It follows that there exists a positive number* δ *such that if the accelerator step is computed for every iteration once the first successful iterate of Algorithm* 2.1 *is contained in* $\mathcal{B}_\delta(w^*)$*, then the sequences of iterates* $\{x_k\}$ *and* $\{y_k\}$ *generated by Algorithm* 2.1 *converge to* x^* *and* y^* *at a Q-superlinear and R-superlinear rate, respectively. Moreover, if* $\nabla_{xx}\mathcal{L}(x, y)$ *is Lipschitz continuous in a neighborhood of* (x^*, y^*)*, then they converge at a Q-quadratic and R-quadratic rate, respectively.*

Proof. Set $\lambda_{\min} = \min(\lambda_{\min}^H/2, \lambda_{\min}^B)$ and $\beta_{\max} = \max(b_B, \|\nabla_{xx}\mathcal{L}(x^*, y^*)\|_2 + 1)$ in the definition of $S(x\,; x^*)$ as given by (3.3), and let β, ε_1, ε_2, and μ be the positive constants guaranteed by Lemmas 3.6 and 3.7; note that they satisfy $0 < \mu \leq \varepsilon_2 \leq \varepsilon_1$ by construction, so that part (ii) of Lemma 3.6 implies

$$\mathcal{B}_{\mu/2}(w^*) \subseteq \mathcal{B}_{\mu}(w^*) \subseteq \mathcal{B}_{\varepsilon_2}(w^*) \subseteq \mathcal{B}_{\varepsilon_1}(w^*) \text{ and } S_{\mu/2} \subseteq S_{\mu} \subseteq S_{\varepsilon_2} \subseteq S_{\varepsilon_1}, \tag{3.42}$$

where S_ε is defined by (3.3) and (3.4). 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. $\|\nabla_{xx}\mathcal{L}\big(x, y^{\text{F}}(x)\big)\|_2 \leq \|\nabla_{xx}\mathcal{L}(x^*, y^*)\|_2 + 1$, where $y^{\text{F}}(x)$ is any estimate satisfying $y^{\text{F}}(x) - y^* = O(\|x - x^*\|_2)$;

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

C4. Newton's method applied to the function F_N in (3.1) 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 3.11; and

C6. $\delta_{\Delta} \leq \eta_c\Delta_{\text{R}}/2$, where $0 < \Delta_{\text{R}} \leq \Delta_{\text{U}}$ and η_c are used in Algorithm 2.1.

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

C7. $\delta \leq \min(\mu/2, \varepsilon_4)$, where ε_4 is defined in Lemma 3.11; and

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

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

Now let $k-1$ be the first successful iterate generated by Algorithm 2.1 such that $w_k \in \mathcal{B}_\delta(w^*)$. By construction of Algorithm 2.1 and the fact that the accelerator trust-region scale factor satisfies $\tau_f \geq 1$, we have

$$\Delta_k^{\text{P}} \geq \Delta_{\text{R}} > 0 \text{ and } \Delta_k^{\text{S}} \geq \tau_f\Delta_{\text{R}} \geq \Delta_{\text{R}} > 0. \tag{3.43}$$

Since (3.42) and C7 imply that $w_k \in \mathcal{B}_{\mu/2}(w^*)$, it follows from C8, Lemma 3.7, and (3.43) 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

$$\|x_k(M) - x_k\|_\infty \leq 2\beta\|F_M(w_k\,; w_k)\|_2 \leq \tfrac{1}{2}\min(\delta_{\Delta}, \eta_c\Delta_{\text{R}}) \leq \tfrac{1}{2}\eta_c \min(\Delta_k^{\text{P}}, \Delta_k^{\text{S}}). \tag{3.44}$$

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 (3.44) implies $\|x_k(B_k) - x_k\|_\infty \leq (\eta_c/2)\Delta_k^{\text{P}} < \Delta_k^{\text{P}}$, we must have $w_k^{\text{P}} = w_k(B_k)$. Thus the solution to the predictor subproblem satisfies $s_k^{\text{P}} = x_k(B_k) - x_k$, correctly identifies the optimal

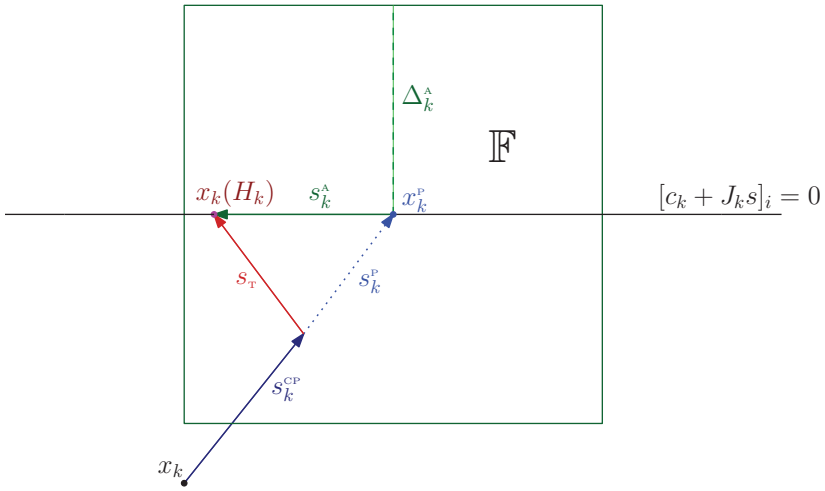


FIG. 3. *A depiction of the scenario in Theorem* 3.12. *The following quantities are displayed:* x_k *is the current iterate,* s_k^P *is the predictor step,* x_k^P *is the predictor point,* s_k^{CP} *is the Cauchy step,* s_k^A *is the accelerator step as computed from problem (EQP),* Δ_k^S *is the accelerator trust-region radius,* s_T *is the solution to problem* (3.39), $x_k(H_k)$ *is the first* n *components of* $w_k(H_k)$, *which is the closest minimizer to* w_k *for problem* (3.14) *with the choice* $M = H_k$, *and* $\mathbb{F}$ *denotes the feasible side of the constraint* $[c_k + J_k s]_i \geq 0$.

active set, and is not restricted by the trust-region constraint; i.e., $s_k^{\rm P}$ is the solution to (3.24) 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 (3.14) in a neighborhood of w_k for $M = H_k$. Since $J_{\mathcal{A}}(x_k)$ has full row rank, it follows from (3.42), part (iii) of Lemma 3.6, and [1, Lemma 1.27] that subproblem (EQP) has $s_k^{\rm A}$ 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 $s_k^{\rm A} = x_k(H_k) - (x_k + s_k^{\rm P})$ (see Figure 3). Using the triangle inequality, the definition of $w_k(B_k)$, and (3.44), we have

$$\begin{aligned}\|x_k(H_k) - (x_k + s_k^{\rm P})\|_2 &\leq \|x_k(H_k) - x_k\|_2 + \|s_k^{\rm P}\|_2 \\ &= \|x_k(H_k) - x_k\|_2 + \|x_k(B_k) - x_k\|_2 \leq \eta_c \Delta_k^{\rm S} \leq \Delta_k^{\rm S}.\end{aligned}$$

Thus, if $s_k^{\rm P} + s_k^{\rm A}$ satisfies condition (2.9), then $s_k = s_k^{\rm P} + s_k^{\rm A}$, and it follows that $x_k + s_k = x_k(H_k)$ and $y_k^{\rm A} = y_k(H_k)$. We now show that this is the case. If $s_{\rm T} \neq 0$, then C5 and Lemma 3.11 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, $M_k^H(s_k^{\rm P} + s_k^{\rm A}) < M_k^H(s_k^{\rm CP})$ so that condition (2.9) is satisfied by $s_k^{\rm P} + s_k^{\rm A}$. On the other hand, if $s_{\rm T} = 0$, then it follows that $s_k^{\rm P} = s_k^{\rm CP}$ and $s_k^{\rm A} = 0$ so that $s_k^{\rm P} + s_k^{\rm A} = s_k^{\rm CP}$ trivially satisfies condition (2.9).

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 $max\_fails \geq 1$, 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 (3.1)) from the point $(x_k, y_k^{\rm F})$ with the understanding that y_{k+1} is formed by "scattering" $y_k^{\rm A}$ into a zero-vector of length m. Since Algorithm 2.1 makes the assignment $w_{k+1} \leftarrow w_k^{\rm A}$, 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 $(x_{k+1}, y^{\mathrm{F}}_{k+1})$. The only difference in the argument is that the predictor and accelerator trust-region radii are only guaranteed to be bigger than $\eta_c \Delta_{\mathrm{R}}$, 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 [10, 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 2.1; 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 [24, Theorem 11.2] and C2 that

$$\begin{pmatrix} x_{k+1} - x^* \\ y_{k+1} - y^* \end{pmatrix} = o\left(\left\| \begin{pmatrix} x_k - x^* \\ y^{\mathrm{F}}_k - y^* \end{pmatrix} \right\|_2 \right) = o\left(\|x_k - x^*\|_2 \right) \tag{3.45}$$

so that $\{x^*_k\}$ and $\{y_k\}$ converge to x^* and y^* Q-superlinearly and R-superlinearly, respectively (see [25, Chapter 9] for a description of Q and R convergence); C2 also shows that $\{y^{\mathrm{F}}_k\}$ converges to y^* R-superlinearly. If $\nabla_{xx}\mathcal{L}(x, y)$ is locally Lipschitz continuous, then a similar argument shows that $\{x_k\}$ converges to x^* Q-quadratically and that $\{y_k\}$ and $\{y^{\mathrm{F}}_k\}$ converge to y^* R-quadratically. ∎

LEMMA 3.13. *Let $w^* = (x^*, y^*)$ be a minimizer for problem* (NP) *that satisfies the* LICQ *and strict complementarity, and suppose that $\sigma_k > \|y^*\|_\infty$, that Assumption* 3.1 *holds, and that $\|B_k\|_2 \le b_B$ for some $b_B > 0$ and all $k \ge 0$. Then there exists a scalar $\delta_P > 0$ such that if iterate $k-1$ is successful and $w_k \in \mathcal{B}_{\delta_P}(w^*)$, then*

$$y^P_k - y^* = O(\|x_k - x^*\|_2) \ \text{ and } \ [y^P_k]_{\mathcal{I}} = 0, \tag{3.46}$$

where y^P_k are the multipliers for the predictor subproblem (2.2).

Proof. Let δ_{P} be defined to satisfy conditions C5–C8 of Theorem 3.12. It follows, just as in the proof of Theorem 3.12, that $s^{\mathrm{P}}_k = x_k(B_k) - x_k$ and that s^{P}_k is the unique solution to problem (3.24) with the choice $M = B_k$. This implies that $(s^{\mathrm{P}}_k, y^{\mathrm{P}}_k)$ satisfies system (3.7) ($\pi = y^{\mathrm{P}}_k$) so that (3.46) follows from (3.6). ∎

3.4. Local convergence with an (EIQP) step. We now consider the rate-of-convergence for Algorithm 2.1 when the accelerator step is computed from subproblem (EIQP) as described in [17, section 2.3.1] and restated on page 2052.

We begin by making two observations. First, since problem (EIQP) is generally a nonconvex *inequality* constrained QP, we will need to assume that the solution s^{A}_k is one of minimal norm; a similar assumption is made by Robinson in [30, 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 (EIQP) will ultimately be the same as the solution to subproblem (EQP). Theorem 3.12 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 accelerator step is chosen to be one of minimal norm, then the proof of Theorem 3.12 carries over, since (1) the Cauchy step s^{CP}_k satisfies $\|s^{\mathrm{CP}}_k\|_\infty \le \|s^{\mathrm{P}}_k\|_\infty$; (2) the vector $x_k(H_k) - s^{\mathrm{CP}}_k$ is a solution to subproblem (EIQP); and (3) Lemma 3.11 guarantees that the descent-constraint does not interfere with the step from s^{CP}_k to $x_k(H_k)$.

THEOREM 3.14 ((EIQP) local convergence result). *Let $w^* = (x^*, y^*)$ be a minimizer for problem* (NP) *that satisfies the strong second-order sufficient conditions as given by Definition* 3.5. *Let Assumption* 3.1 *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 accelerator step is computed from subproblem* (EIQP) *with the choice* $H_k \equiv \nabla_{xx}\mathcal{L}(x_k, y_k^F)$*, and max_fails* ≥ 1 *in Algorithm* 2.1*. It follows that there exists a positive number* δ *such that if the accelerator step is a solution of minimal-norm and is computed for every iteration once the first successful iterate of Algorithm* 2.1 *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* $\nabla_{xx}\mathcal{L}(x, y)$ *is Lipschitz continuous in a neighborhood of* (x^*, y^*)*, then they converge at a Q-quadratic and R-quadratic rate, respectively.*

Proof. The result follows from the proof of Theorem 3.12, the discussion above, and Lemma 3.10. ∎

4. Practical issues concerning B_k and σ. In this section we address two components of Algorithm 2.1 that are important for an efficient implementation. First, we describe a strategy for defining the positive-definite matrix B_k via a limited-memory BFGS update. Second, we briefly consider a simple strategy for updating the penalty parameter.

4.1. A limited-memory BFGS update. In this section we describe a method for defining the positive-define matrix needed in the computation of the predictor step (2.1) that is based on the limited-memory BFGS update. We must be cautious, however, since the matrix $\nabla_{xx}\mathcal{L}(x_k, y_k)$ is generally indefinite, and, therefore, the traditional update may result in an indefinite matrix [2, 27]. We also note that if the problem dimension is small, then a full BFGS update is practical.

The limited-memory BFGS update uses a fixed number of vectors, say l, to define a positive-definite approximation to $\nabla_{xx}\mathcal{L}(x_k, y_k)$ based on the most recent l iterations (for more details see [24, 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

$$B_k = B_k^0 + \sum_{i=k-l}^{k-1} (q_i q_i^T - p_i p_i^T), \tag{4.1}$$

where B_k^0 denotes *any* initial positive-definite approximation to $\nabla_{xx}\mathcal{L}(x_k, y_k)$ and

$$p_i = \frac{B_i s_i}{(s_i^T B_i s_i)^{1/2}}, \quad q_i = \frac{d_i}{(d_i^T s_i)^{1/2}}, \quad \text{and} \quad B_i = B_k^0 + \sum_{j=k-l}^{i-1} (q_j q_j^T - p_j p_j^T). \tag{4.2}$$

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 (4.2) 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 (4.2) suggest how this may be done, since

$$B_i s_i = B_k^0 s_i + \sum_{j=k-l}^{i-1} \left[(q_j^T s_i) q_j - (p_j^T s_i) p_j \right]. \tag{4.3}$$

Algorithm 4.1, which is [24, Procedure 7.6], computes the vector-pair (p_i, q_i) recursively.

ALGORITHM 4.1 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 4.1 computes the values q_i for $k-l \le i \le k-1$ and $q_j^T s_i$ for all $k-l \le j \le i-1$. However, since q_i depends only on the data (d_i, s_i), only the value q_{k-1} and values $q_j^T s_{k-1}$ $(k-l \le j \le k-2)$ 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 (2.2) then becomes

$$\begin{array}{ll} \underset{s\in\mathbb{R}^n, v\in\mathbb{R}^m}{\text{minimize}} & f_k + g_k^T s + \frac{1}{2}s^T(B_k^0 - PP^T + QQ^T)s + \sigma e^T v \\ \text{subject to} & c_k + J_k s + v \ge 0,\ \ v \ge 0,\ \ \|s\|_\infty \le \Delta_k^{\text{P}}. \end{array} \tag{4.4}$$

If we define the $2l$ extra variables

$$w_a = P^T s \quad \text{and} \quad w_b = Q^T s, \tag{4.5}$$

then problem (4.4) is equivalent to

$$\begin{array}{ll} \underset{s,v,w_a,w_b}{\text{minimize}} & f_k + g_k^T s + \frac{1}{2}(s^T B_k^0 s - w_a^T w_a + w_b^T w_b) + \sigma e^T v \\ \text{subject to} & c_k + J_k s + v \ge 0,\ \ P^T s = w_a,\ \ Q^T s = w_b,\ \ v \ge 0,\ \ \|s\|_\infty \le \Delta_k^{\text{P}}. \end{array} \tag{4.6}$$

As a function of (s, v, w_a, w_b), the Hessian associated with subproblem (4.6) is given by

$$B_k^{\text{A}} = \begin{pmatrix} B_k^0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -I & 0 \\ 0 & 0 & 0 & I \end{pmatrix}, \tag{4.7}$$

which is *not* positive definite. This may seem strange, since problem (4.6) is equivalent to the *strictly convex* QP (4.4) (assuming that the updated matrix was positive definite). However, if the current iterate is feasible for subproblem (4.6), 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

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

since B_k is positive definite by construction. A great advantage in using subproblem (4.6) is that the Hessian matrix has essentially the same sparsity as B_k^0. In contrast, the Hessian matrix associated with subproblem (4.4) is generally dense, since it uses a sum of rank-1 updates. Note, however, that the $2l$ extra constraints (4.5) 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 [15].

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. When the resulting update is not positive definite, then perhaps the simplest strategy is to use the *damping* technique introduced by Powell [27]. Basically, this approach modifies d_k so that the resulting update is (sufficiently) positive definite.

We have explored other variants of this basic idea in [16, section 2.2].

4.2. Updating the penalty parameter. The updating scheme that we now discuss is based on the simple idea of calculating a sequence of approximate solutions to 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

$$\varepsilon_k^{\mathrm{D}} \geq \frac{\|g_k + \sigma_k J_k^T \pi_k\|_\infty}{1 + \|g_k\|_\infty}, \tag{4.8a}$$

$$[\pi_k]_i = \begin{cases} [-\frac{\varepsilon_k^{\mathrm{C}}}{\sigma_k}, \frac{\varepsilon_k^{\mathrm{C}}}{\sigma_k}] & \text{if } [c_k]_i > \varepsilon_k^{\mathrm{P}}, \\ [-1 - \varepsilon_k^{\mathrm{C}}, \frac{\varepsilon_k^{\mathrm{C}}}{\sigma_k}] & \text{if } -\varepsilon_k^{\mathrm{P}} \leq [c_k]_i \leq \varepsilon_k^{\mathrm{P}}, \\ [-1 - \varepsilon_k^{\mathrm{C}}, -1 + \varepsilon_k^{\mathrm{C}}] & \text{if } [c_k]_i < -\varepsilon_k^{\mathrm{P}}, \end{cases} \tag{4.8b}$$

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).

In practice, we define estimates $\pi_k = -y_k^{\mathrm{P}}/\sigma_k$, where y_k^{P} is the Lagrange multiplier vector for problem (2.2). Provided the sequence $\{y_k^{\mathrm{P}}\}$ 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 (4.8). We note that alternatives, such as defining $\pi_k = -y_k^{\mathrm{A}}/\sigma_k$ for accelerator multipliers y_k^{A} or defining π_k as a solution of the optimization problem

$$\underset{\pi \in \mathbb{R}^m}{\text{minimize}} \ \tfrac{1}{2}\|g_k + \sigma_k J_k^T \pi\|_2^2 \ \text{ subject to } \pi \ \text{ satisfying (4.8b)}, \tag{4.9}$$

may be used.

Algorithm 4.2 provides the pseudocode for updating the penalty parameter as well as the additional parameter initiations that must be made.

ALGORITHM 4.2 UPDATING σ BASED ON APPROXIMATE CRITICAL POINTS TO PROBLEM (ℓ_1-σ).

begin (additions to preamble of Algorithm 2.1)
  Choose $\sigma_0 > 0$, $\eta_0 > 0$, $0 < \varepsilon_c < 1$, $0 < \varepsilon_0^{\mathrm{P}} < \varepsilon_c \eta_0$, 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 2.1)
if (x_k, π_k) satisfies condition (4.8), **then** [an approximate critical point]
  if $c(x_k) \geq -\eta_k e$, **then** [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 $\varepsilon_k^{\mathrm{D}} = \varepsilon_k^{\mathrm{C}} = \varepsilon_k^{\mathrm{P}}$. However, all that is required is that $\lim_{k\to\infty} \varepsilon_k^{\mathrm{P}} = \lim_{k\to\infty} \varepsilon_k^{\mathrm{D}} = \lim_{k\to\infty} \varepsilon_k^{\mathrm{C}} = 0$.

For numerical considerations, it is generally undesirable 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 problem contains infeasible constraints. Detecting this situation is difficult and is equivalent to showing that the *global* solution of

$$\underset{x\in\mathbb{R}^n}{\text{minimize}} \quad \|[c(x)]^-\|_1 \tag{4.10}$$

is strictly positive (see [4] for some recent work on this topic). The second situation occurs when the iterates converge to a critical point of problem (4.10) 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 4.2 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 4.1. *A point x is a first-order critical point for problem* (4.10) *if it satisfies*

$$J(x)^T y = 0 \tag{4.11}$$

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

$$\frac{\|g_k - J_k^T y_k\|_\infty}{1 + \|g_k\|_\infty} \leq \tau_d, \tag{4.12a}$$

$$c_k \geq -\tau_p e, \tag{4.12b}$$

$$y_k \geq -\tau_c e, \tag{4.12c}$$

$$\max(|c_k|, |y_k|) \leq \tau_c e, \tag{4.12d}$$

where condition (4.12d) should be interpreted componentwise.

THEOREM 4.2. *Let the assumptions for global convergence hold* [17, Theorem 4.3]*, and let* $\{x_k\}$ *be the sequence of iterates generated by Algorithm* 2.1 *with penalty parameter update given by Algorithm* 4.2*. 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* (4.10)*, 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* (4.12)*, where* $y_k \stackrel{\text{def}}{=} -\sigma_k \pi_k$ *and* (x_k, π_k) *is an approximate solution to* (ℓ_1-σ) *as given by* (4.8) *for the value* σ_k*.*

Proof. The proof of these statements is a relatively straightforward exercise. For a detailed proof see [16, Theorem 3.3]. ∎

We close this section by mentioning two potential drawbacks associated with using Algorithm 4.2. First, if the initial penalty parameter is substantially smaller than the threshold value required to guarantee convergence [10, Theorem 14.5.1], then Algorithm 4.2 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 [6, 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 [6, 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. [5] and Byrd, Nocedal, and Waltz [6]. 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

$$\Delta M_k^B(s_k^{\text{P}}) \geq \varepsilon_a \sigma_{\text{N}}(\|[c_k]^-\|_1 - \|[c_k + J_k s_k^{\text{P}}]^-\|_1), \tag{4.13}$$

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 a sequential linear

quadratic programming method that is part of the KNITRO software package [33], 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 a reasonable estimate of the size of the entire sequence of computed multiplier estimates is known in advance, then steering is likely to be less efficient because of the potential overhead associated with the method. However, since this is usually not the case, we generally recommend steering.

5. Numerical results. Preliminary testing of Algorithm 2.1 was performed on the Hock–Schittkowski (HS) [19] 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 *HS*87 has been removed from the test set, since the objective function is not continuous.

To be precise, we tested three variants of Algorithm 2.1—they differ in how we compute the accelerator step and update the penalty parameter. In the first variant we computed the accelerator step from the inequality constrained subproblem (EIQP) on page 2052 and updated the penalty parameter by using "steering" as briefly described at the end of section 4.2. In the second variant we computed the accelerator step from the equality constrained subproblem (EQP) on page 2052 and again used steering to update the penalty parameter. Finally, in the third variant we computed the accelerator step from the inequality constrained subproblem (EIQP) and updated the penalty parameter by using Algorithm 4.2.

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 (2.2) 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 damping technique introduced by Powell [27]. For simplicity, we chose $B_0 = I$.

In all cases, we chose $H_k \equiv \nabla_{xx} \mathcal{L}(x_k, y_k^{\mathrm{P}})$ during the computation of the accelerator and Cauchy step, where y_k^{P} is the multiplier vector from the predictor subproblem. We solved both the (convex) quadratic program (2.2) and the (generally indefinite) quadratic program (EIQP) using the GALAHAD [15] package `QPC`, which is a "cross-over" QP solver. In the first phase, `QPC` calls the GALAHAD interior-point QP solver `QPB` [9] 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` [18] to "refine" the approximate solution from the first phase. To solve the equality constrained QP (EQP) 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 [20] 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 $\tau_p = \tau_d = \tau_c = 1.0e^{-5}$, successful/very successful tolerances $\eta_s = 0.01$ and $\eta_{vs} = 0.7$, maximum predictor trust-region radius $\Delta_{\mathrm{U}} = 1000$, trust-region "reset" radius $\Delta_{\mathrm{R}} = 1.0e^{-4}$, accelerator trust-region scale factor $\tau_f = 4.0$, number of nonmonotone steps allowed $max\_fails = 1$, and trust-region contraction and expansion factors $\eta_c = 0.1$ and $\eta_e = 5.0$. We used an initial penalty parameter of $\sigma = 1.0$ for the first and second strategies, and we used an initial penalty parameter of $\sigma = 9.0$ for the third strategy. The larger (seemingly arbitrary) initial penalty

TABLE 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.

	EIQP-steer		EQP-steer		EIQP-seq	
Prob.	#fc	#gJ	#fc	#gJ	#fc	#gJ
*HS*1	46	25	39	24	46	25
*HS*2	11	8	11	8	11	8
*HS*3	3	3	3	3	3	3
*HS*4	3	2	3	2	3	2
*HS*5	10	6	9	6	10	6
*HS*6	3	3	5	5	3	3
*HS*7	10	9	9	8	12	10
*HS*8	6	6	6	6	6	6
*HS*9	3	3	4	4	3	3
*HS*10	10	10	10	10	10	10
*HS*11	6	6	6	6	6	6
*HS*12	7	7	6	6	14	11
*HS*13	15	12	29	27	56	54
*HS*14	5	5	5	5	5	5
*HS*15	7	7	7	7	20	20
*HS*16	4	4	4	4	5	5
*HS*17	7	7	7	7	10	10
*HS*18	9	8	7	7	9	8
*HS*19	6	6	6	6	13	13
*HS*20	8	8	4	4	16	15
*HS*21	2	2	2	2	2	2
*HS*22	2	2	2	2	2	2
*HS*23	6	6	6	6	6	6
*HS*24	3	3	4	4	3	3
*HS*25	1	1	1	1	1	1
*HS*26	17	17	16	16	17	17
*HS*27	12	11	14	13	13	12
*HS*28	2	2	3	3	2	2
*HS*29	6	6	6	6	31	23

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.

Tables 1, 2, 3, and 4 give our preliminary numerical results for these three strategies; column EIQP-steer corresponds to the first strategy, column EQP-steer corresponds to the second strategy, and column EIQP-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.

TABLE 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.*

	EIQP-steer		EQP-steer		EIQP-seq	
Prob.	#fc	#gJ	#fc	#gJ	#fc	#gJ
*HS*30	10	10	10	10	10	10
*HS*31	8	8	9	7	5	5
*HS*32	3	3	4	4	3	3
*HS*33	5	5	14	11	5	5
*HS*34	9	8	9	8	34	28
*HS*35	2	2	3	3	2	2
*HS*36	3	3	4	4	3	3
*HS*37	5	5	5	5	5	5
*HS*38	84	38	65	46	84	38
*HS*39	12	11	13	11	12	11
*HS*40	4	4	4	4	4	4
*HS*41	2	2	2	2	2	2
*HS*42	6	6	4	4	4	4
*HS*43	7	7	8	8	12	10
*HS*44	2	2	3	3	2	2
*HS*45	3	3	5	4	3	3
*HS*46	16	16	16	16	16	16
*HS*47	15	15	20	18	19	17
*HS*48	2	2	3	3	2	2
*HS*49	16	16	16	16	16	16
*HS*50	9	9	6	6	9	9
*HS*51	2	2	2	2	2	2
*HS*52	2	2	2	2	2	2
*HS*53	2	2	2	2	2	2
*HS*54	9	9	14	14	22	22
*HS*55	2	2	2	2	2	2
*HS*56	110	82	F	F	F	F
*HS*57	8	6	6	6	8	6
*HS*59	10	9	8	8	18	15

The strictly convex predictor subproblem combined with either accelerator subproblem (EIQP) or (EQP) 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 4.2; this agrees with [5, 6]. In particular, steering was essential in solving *HS*93 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 *HS*56. When the update in section 4.2 was used, the merit function again converged to minus infinity; the same occurred when the accelerator step was computed from subproblem (EQP) and steering was used.

TABLE 3

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.

	EIQP-steer		EQP-steer		EIQP-seq	
Prob.	#fc	#gJ	#fc	#gJ	#fc	#gJ
*HS*60	7	7	7	7	7	7
*HS*61	5	5	5	5	14	11
*HS*62	12	6	8	7	12	6
*HS*63	7	7	7	7	7	7
*HS*64	15	15	20	17	23	23
*HS*65	6	6	6	6	22	16
*HS*66	4	4	4	4	4	4
*HS*67	7	7	10	10	14	14
*HS*68	27	20	38	27	20	16
*HS*69	24	19	42	30	45	34
*HS*70	21	17	41	33	21	17
*HS*71	5	5	5	5	5	5
*HS*72	16	15	15	14	46	45
*HS*73	3	3	3	3	3	3
*HS*74	8	8	8	8	8	8
*HS*75	8	8	8	8	8	8
*HS*76	2	2	4	4	2	2
*HS*77	12	12	12	12	12	12
*HS*78	4	4	4	4	4	4
*HS*79	5	5	5	5	5	5
*HS*80	7	7	8	8	7	7
*HS*81	6	6	6	6	6	6
*HS*83	6	6	6	6	16	16
*HS*84	4	4	4	4	5	5
*HS*85	8	8	13	12	21	20
*HS*86	4	4	4	4	4	4
*HS*88	20	18	25	21	43	41
*HS*89	23	21	64	46	45	43

6. Conclusions. In [17], 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. The main purpose of this paper was to study the local convergence properties of a nonmonotone variant of that algorithm. In section 3 we gave two local convergence results—the first applies when the accelerator step is computed from an *equality* constrained subproblem (the so-called SEQP approach), and the second applies when the accelerator 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.

Algorithm 2.1 requires the definition of a positive-definite matrix that approximates the Hessian of the Lagrangian. In section 4.1 we discussed a strategy for defining these matrices based on *limited*-memory BFGS updating. In particular, we showed how the resultant dense predictor step QP could be transformed into an equiv-

TABLE 4

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.*

	EIQP-steer		EQP-steer		EIQP-seq	
Prob.	#fc	#gJ	#fc	#gJ	#fc	#gJ
*HS*90	44	33	40	30	42	34
*HS*91	43	32	32	25	73	61
*HS*92	35	25	34	27	64	37
*HS*93	18	15	6	6	F	F
*HS*95	3	3	2	2	3	3
*HS*96	3	3	2	2	3	3
*HS*97	4	4	5	5	4	4
*HS*98	4	4	5	5	4	4
*HS*99	5	5	FQP	FQP	75	34
*HS*100	10	9	13	10	10	9
*HS*101	34	27	68	40	67	52
*HS*102	28	21	55	34	44	39
*HS*103	26	20	27	20	79	62
*HS*104	17	14	19	14	14	11
*HS*105	21	14	31	23	21	14
*HS*106	103	101	122	66	103	101
*HS*107	6	6	6	6	10	10
*HS*108	12	9	12	10	242	177
*HS*109	9	9	9	9	10	10
*HS*110	9	5	8	6	9	5
*HS*111	28	23	41	31	23	19
*HS*112	11	11	50	50	11	11
*HS*113	5	5	6	6	5	5
*HS*114	142	142	13	13	136	134
*HS*116	F	F	F	F	F	F
*HS*117	10	10	11	11	10	10
*HS*118	3	3	12	12	3	3
*HS*119	7	7	8	8	7	7

alent QP whose sparsity is essentially the same as the initial approximation (which is chosen to be sparse in practice).

In section 4.2 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 [8, 28, 21, 32, 26, 34, 3, 22], the details of our very simple strategy have not been published to our knowledge.

In section 5 we gave preliminary numerical results for the HS test problems. Our first set of results were based on using "steering" [5, 6] to update the penalty parameter and computing the accelerator step from the inequality constrained subproblem (EIQP) on page 2052. Although this requires "solving" a potentially indefinite QP, 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 accelerator step from the equality constrained subproblem (EQP) on page 2052. Our last set of results was based on solving subproblem (EIQP) for the accelerator 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 4.2). 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.

During essentially simultaneous work, Morales, Nocedal, and Wu [23] have developed a similar ℓ_1-SQP *line-search* algorithm. Roughly, they compute a predictor step (without a trust-region constraint) followed by an accelerator step defined as the solution to problem (EQP). They then reduce the ℓ_1-merit function by performing a line search along the "bent" path defined by the steps s_k^{P} and s_k^{A}. Our methods differ in the following ways. First, Algorithm 2.1 is based on trust-region methodology, while their algorithm is based on line-search philosophy. Second, global convergence of our algorithm is guaranteed by the Cauchy step, while convergence of their algorithm is ensured by the predictor step with a suitable line search. Third, we allow and have analyzed an accelerator step computed as the minimizer of an inequality constrained subproblem, which allows for active set refinement; they have not considered such a subproblem, although one could imagine that such an analysis is possible. Finally, our algorithms differ even when subproblem (EQP) is used to compute an accelerator step. Following the rejection of a trial step, Morales, Nocedal, and Wu perform a line search in the direction of the predictor step. We, on the other hand, perform the equivalent of a backtracking line search with each trial point enhanced by a *new* accelerator direction. Since convergence of our method relies on the Cauchy point, we could easily use more sophisticated line search techniques without sacrificing convergence.

Acknowledgments. The authors thank an anonymous referee and the associate editor for their helpful comments on this paper.

REFERENCES

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

[2] P. T. Boggs and J. W. Tolle, *Sequential quadratic programming*, Acta Numer., 4 (1995), pp. 1–51.

[3] J. V. Burke, *A robust trust region method for constrained nonlinear programming problems*, SIAM J. Optim., 2 (1992), pp. 325–347.

[4] R. H. Byrd, F. E. Curtis, and J. Nocedal, *Infeasibility Detection and SQP Methods for Nonlinear Optimization*, Technical report OTC 2008/03, Optimization Technology Center, Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, IL, 2008.

[5] 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.

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

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

[8] L. Chen and D. Goldfarb, *Interior-point l_2-penalty methods for nonlinear programming with strong global convergence properties*, Math. Program., 108 (2006), pp. 1–36.

[9] 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.

[10] A. R. Conn, N. I. M. Gould, and Ph. L. Toint, *Trust-Region Methods*, SIAM, Philadelphia, 2000.

[11] J. E. Dennis, Jr. and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Prentice-Hall, Englewood Cliffs, NJ, 1983.
[12] R. Fletcher, *Practical Methods of Optimization, Vol. 2: Constrained Optimization*, John Wiley and Sons, 1981.
[13] 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.
[14] G. H. Golub and C. F. Van Loan, *Unsymmetric positive definite linear systems*, Linear Algebra Appl., 28 (1979), pp. 85–98.
[15] N. I. M. Gould, D. Orban, and P. L. Toint, *GALAHAD, a library of thread-safe Fortran 90 packages for large-scale nonlinear optimization*, ACM Trans. Math. Software, 29 (2003), pp. 353–372.
[16] N. I. M. Gould and D. P. Robinson, *A Second Derivative SQP Method: Local Convergence*, Numerical analysis report 08/21, Computational Laboratory, University of Oxford, Oxford, UK, 2008.
[17] N. I. M. Gould and D. P. Robinson, *A second derivative SQP method: Global convergence*, SIAM J. Optim., 20 (2010), pp. 2023–2048.
[18] N. I. M. Gould and P. L. Toint, *An iterative working-set method for large-scale non-convex quadratic programming*, Appl. Numer. Math., 43 (2002), pp. 109–128.
[19] W. Hock and K. Schittkowski, *Test Examples for Nonlinear Programming Codes*, Lecture Notes in Econom. and Math. Systems 187, Springer-Verlag, New York, 1981.
[20] HSL, *A Collection of Fortran Codes for Large-scale Scientific Computation*, http://www.cse.clrc.ac.uk/nag/hsl/.
[21] 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.
[22] M. Mongeau and A. Sartenaer, *Automatic decrease of the penalty parameter in exact penalty function methods*, European J. Oper. Res., 83 (1995), pp. 686–699.
[23] J. L. Morales, J. Nocedal, and Y. Wu, *A Sequential Quadratic Programming Algorithm with an Additional Equality Constrained Phase*, Technical report OTC-05, Northwestern University, Evanston, IL, 2008.
[24] J. Nocedal and S. J. Wright, *Numerical Optimization*, Springer-Verlag, New York, 1999.
[25] J. M. Ortega and W. C. Rheinboldt, *Iterative solution of nonlinear equations in several variables*, SIAM, Philadelphia, 2000. Reprint of the 1970 original.
[26] 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.
[27] M. J. D. Powell, *The convergence of variable metric methods for nonlinearly constrained optimization calculations*, in Nonlinear Programming, Vol. 3, Academic Press, New York, 1978, pp. 27–63.
[28] M. J. D. Powell, *A fast algorithm for nonlinearly constrained optimization calculations*, in Numerical Analysis, Dundee 1977, Lecture Notes in Math. 630, G. A. Watson, ed., Springer-Verlag, New York, 1978, pp. 144–157.
[29] S. M. Robinson, *A quadratically-convergent algorithm for general nonlinear programming problems*, Math. Program., 3 (1972), pp. 145–156.
[30] S. M. Robinson, *Perturbed Kuhn-Tucker points and rates of convergence for a class of nonlinear programming algorithms*, Math. Program., 7 (1974), pp. 1–16.
[31] W. Rudin, *Principles of Mathematical Analysis*, McGraw-Hill, New York, 1976.
[32] M. Sahba, *Globally convergent algorithm for nonlinearly constrained optimization problems*, J. Optim. Theory Appl., 52 (1987), pp. 291–309.
[33] 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, 2002.
[34] Y.-X. Yuan, *On the convergence of a new trust region algorithm*, Numer. Math., 70 (1995), pp. 515–539.