AN INEXACT SQP METHOD FOR EQUALITY CONSTRAINED OPTIMIZATION

RICHARD H. BYRD*, FRANK E. CURTIS †, AND JORGE NOCEDAL ‡

Abstract. We present an algorithm for large-scale equality constrained optimization. The method is based on a characterization of inexact sequential quadratic programming (SQP) steps that can ensure global convergence. Inexact SQP methods are needed for large-scale applications for which the iteration matrix cannot be explicitly formed or factored and the arising linear systems must be solved using iterative linear algebra techniques. We address how to determine when a given inexact step makes sufficient progress toward a solution of the nonlinear program, as measured by an exact penalty function. The method is globalized by a line search. An analysis of the global convergence properties of the algorithm and numerical results are presented.

Key words. large-scale optimization, constrained optimization, sequential quadratic programming, inexact linear system solvers, Krylov subspace methods

AMS subject classifications. 49M37, 65K05, 90C06, 90C30, 90C55

1. Introduction. In this paper we discuss an algorithm for equality constrained optimization problems of the form

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & f(x) \\ \text{s.t.} \quad & c(x) = 0, \end{aligned} \tag{1.1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ and $c : \mathbb{R}^n \to \mathbb{R}^t$ are smooth nonlinear functions. Our interest is in methods for very large problems with $t \leq n$ for which the exact computation of steps in contemporary methods can be prohibitively expensive. One class of problems of this type that demands algorithmic improvements are those where the constraint functions are defined by systems of partial differential equations (PDEs).

One of the leading methods for solving constrained optimization problems is sequential quadratic programming (SQP). (In fact, modern interior point methods reduce to SQP when inequality constraints are not present in the problem formulation [18].) Algorithms in this class enjoy global convergence guarantees and typically require few iterations and function evaluations to locate a solution point. A drawback of many contemporary SQP algorithms, however, is that they require explicit representations of exact derivative information and the solution of one or more linear systems during every iteration. The acquisition of these quantities is particularly cumbersome in large-scale settings and the factorization of large iteration matrices is often impractical.

One way to overcome these difficulties is to solve the SQP subproblems approximately using iterative linear algebra techniques. The main purpose of this paper is to determine the accuracy with which the SQP subproblems must be solved in order to

*Department of Computer Science, University of Colorado, Boulder, CO, 80309, USA. This author was supported by Army Research Office Grant DAAD19-02-1-0407, and by National Science Foundation grants CCR-0219190 and CHE-0205170.

†Department of Industrial Engineering and Management Sciences, Northwestern University, Evanston, IL, 60208-3118, USA. This author was supported by Department of Energy grant DE-FG02-87ER25047-A004.

‡Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, IL, 60208-3118, USA. This author was supported by National Science Foundation grant CCF-0514772 and by a grant from the Intel Corporation.

ensure global convergence in the context of a practical algorithm for problem (1.1). We both propose such a method and analyze its global behavior.

Our method resembles those in the class of inexact Newton methods for solving nonlinear systems of equations. There are, however, important differences between the two approaches. Inexact Newton methods for systems of equations are controlled by forcing parameters that ensure that the norm of the entire residual of the Newton equations decreases at every iteration [8]. Our approach, on the other hand, is based on a requirement that the step decreases a local approximation of a merit function, while also satisfying bounds on the primal and dual components of the residual. We present sets of easily calculable conditions that handle these two components of the residual as separate quantities when determining if a given inexact solution is appropriate for the algorithm to follow. Such a solution may, for example, allow for an increase in the residual corresponding to primal feasibility provided it yields a substantial decrease in dual feasibility, or vice versa. The behavior of these components also helps determine when it is appropriate to increase the penalty parameter in the merit function.

A variety of methods for constrained optimization with inexactness in step computations have been proposed recently. Jäger and Sachs [14] describe an inexact reduced SQP method in Hilbert space. Lalee, Nocedal, and Plantenga [16], Byrd, Hribar, and Nocedal [5], and Heinkenschloss and Vicente [13] propose composite step trust region approaches where the step is computed as an approximate solution to an SQP subproblem. Similarly, Walther [22] provides a composite step method that allows incomplete constraint Jacobian information. Leibfritz and Sachs [17] analyze an interior point method that benefits from a reformulation of the quadratic programming subproblems as mixed linear complementarity problems. Our approach has some features in common with the algorithms of Biros and Ghattas [1, 2], Haber and Ascher [11], and Prudencio, Byrd and Cai [20] as we follow a full space SQP method and perform a line search to promote convergence. Unlike these papers, however, we present conditions that guarantee the global convergence of inexact SQP steps.

This paper is organized as follows. In §2 we provide an overview of our approach and globalization strategy. Section 3 contains details about the most crucial aspect of our algorithm, namely, the sets of conditions used to determine if a given inexact SQP solution is considered an acceptable step. The well-posedness of our approach is also discussed, the accountability of which allows us to present global convergence guarantees under common conditions in §4. Section 5 provides numerical results to illustrate the robustness of our method. We focus on problems for which overall algorithm performance has been seen to be sensitive to the quality of inexact subproblem solutions. Closing remarks and issues related to extensions of this work are presented in §6.

2. Outline of the Algorithm. Let us formalize a basic SQP approach before clarifying the novelties of our algorithm. The Lagrangian function corresponding to problem (1.1) is

$$\mathcal{L}(x, \lambda) \triangleq f(x) + \lambda^T c(x), \tag{2.1}$$

where $\lambda \in \mathbb{R}^t$ are Lagrange multipliers. If f and c are continuously differentiable, then the first-order optimality conditions for x^* to be an optimal solution to problem (1.1) state that there exist multipliers λ^* such that (x^*, λ^*) is a solution to the nonlinear

system of equations

$$\nabla \mathcal{L}(x, \lambda) = \begin{bmatrix} g(x) + A(x)^T \lambda \\ c(x) \end{bmatrix} = 0, \tag{2.2}$$

where $g(x)$ is the gradient of the objective function and $A(x)$ is the Jacobian of $c(x)$. The components in (x, λ) are referred to as the primal and dual variables, respectively.

An SQP algorithm defines an appropriate displacement d_k in the primal space from the kth iterate x_k as the minimizer of a quadratic model of the objective subject to a linearization of the constraints. The quadratic program can be defined as

$$\begin{aligned} \min_{d \in \mathbb{R}^n} \quad & f(x_k) + g(x_k)^T d + \tfrac{1}{2} d^T W(x_k, \lambda_k) d \\ \text{s.t.} \quad & c(x_k) + A(x_k) d = 0, \end{aligned} \tag{2.3}$$

where

$$W(x, \lambda) \approx \nabla^2_{xx} \mathcal{L}(x, \lambda) = \nabla^2_{xx} f(x) + \sum_{i=1}^{t} \lambda^i \nabla^2_{xx} c^i(x)$$

is equal to, or is a symmetric approximation for, the Hessian of the Lagrangian. Here, $c^i(x)$ and λ^i denote the ith constraint function and its corresponding dual variable, respectively. If the constraint Jacobian $A(x_k)$ has full row rank and $W(x_k, \lambda_k)$ is positive definite on the null space of $A(x_k)$, then a solution to (2.3) is well defined in this context. An alternative characterization of the SQP step d_k is given by the fact that it can equivalently be obtained under similar assumptions as part of the solution to the primal-dual system (see [18])

$$\begin{bmatrix} W(x_k, \lambda_k) & A(x_k)^T \\ A(x_k) & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \delta_k \end{bmatrix} = - \begin{bmatrix} g(x_k) + A(x_k)^T \lambda_k \\ c(x_k) \end{bmatrix} \tag{2.4}$$

constructed by applying Newton's method to (2.2).

An explicit representation of the primal-dual matrix

$$\begin{bmatrix} W(x_k, \lambda_k) & A(x_k)^T \\ A(x_k) & 0 \end{bmatrix} \tag{2.5}$$

and an exact solution of (2.4) can be expensive to obtain, particularly when the factors of (2.5) are not very sparse. We are interested, therefore, in identifying inexact solutions of (2.4) that can also be considered appropriate steps for the algorithm to accept during a given iteration. Such inexact solutions can be obtained in a variety of ways, such as by applying an iterative linear system solver to the primal-dual system. Regardless of the method chosen, for an inexact solution (d_k, δ_k) we define the residual vectors (ρ_k, r_k) by the equation

$$\begin{bmatrix} W(x_k, \lambda_k) & A(x_k)^T \\ A(x_k) & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \delta_k \end{bmatrix} = - \begin{bmatrix} g(x_k) + A(x_k)^T \lambda_k \\ c(x_k) \end{bmatrix} + \begin{bmatrix} \rho_k \\ r_k \end{bmatrix}. \tag{2.6}$$

The step can then be appraised based on properties of the residual vector and other quantities related to the SQP subproblem formulation (2.3). For convex problems, an inexact Newton method intended for nonlinear equations will suffice, provided that $W(x_k, \lambda_k)$ is the exact Hessian of the Lagrangian [8]. That is, the norm of

the right-hand-side vector in (2.4) can serve as a merit function, and convergence can be guaranteed by systematically decreasing this value. For nonconvex problems, however, a step that decreases the first order optimality error may move away from a minimizer, or may be trapped near a stationary point of the Lagrangian. Thus, merit functions more appropriate to constrained optimization should be considered.

We now outline the algorithm and globalization strategy that will be developed in detail in the following sections. An integral part of the approach is the mechanism used to determine if a trial primal-dual solution (d, δ) to (2.4) is acceptable during a given iteration. For this purpose, we make use of the merit function

$$\phi(x;\pi) \triangleq f(x) + \pi\|c(x)\|, \tag{2.7}$$

where $\pi > 0$ is known as the penalty parameter and $\|\cdot\|$ denotes a norm on $\mathbb{R}^t$. We observe that $\phi(x;\pi)$ is not continuously differentiable, but it is exact in the sense that if π is greater than a certain threshold, then a first order optimal point of (1.1) is a stationary point of $\phi(x;\pi)$. That is, the directional derivative of $\phi(x;\pi)$ in a direction d, denoted by $D\phi(d;\pi)$, is nonnegative at x^* for all $d \in \mathbb{R}^n$. The challenge is to compute inexact SQP steps and a value for π that ensure progress in the merit function $\phi(x;\pi)$ during every iteration.

Upon the calculation and acceptance of the search direction d_k for a particular value π_k of the penalty parameter, we perform a backtracking line search to compute a steplength coefficient α_k satisfying the Armijo condition

$$\phi(x_k + \alpha_k d_k; \pi_k) \leq \phi(x_k;\pi_k) + \eta\alpha_k D\phi(d_k;\pi_k) \tag{2.8}$$

for some $0 < \eta < 1$. Accordingly, a primal-dual step will only be accepted if its primal component is a descent direction for the merit function.

In summary, our approach follows a standard line search SQP framework. During each iteration, a step is computed as an inexact solution to the primal-dual system (2.6) satisfying appropriate conditions that deem the step acceptable. The penalty parameter is then set based on properties of the computed step, after which a backtracking line search is performed to compute a steplength coefficient α_k satisfying the Armijo condition (2.8). Finally, the iterate is updated along with function and derivative information at the new point. The novelty of our approach, i.e., the precise definition of what constitutes an acceptable step, and the convergence properties of this algorithm are considered in the remainder of this paper.

Notation. We drop functional notation throughout the rest of the paper when values are clear from the context and delimit iteration number information for functions as with variables; i.e., we denote $g_k \triangleq g(x_k)$ and similarly for other quantities. All norms are considered Euclidean (or l_2) norms unless otherwise indicated, though much of our analysis will apply for any norm.

3. Step Computation and Selection. An intuitive condition that one may impose on an inexact SQP step is that the directional derivative of the merit function along the primal component d_k must be sufficiently negative. Such a condition could be used in the development of a globally convergent SQP approach, but quantifying an appropriate steepness of the directional derivative is a difficult task in practice.

As an alternative, let us borrow from an approach commonly employed in trust region methods that begins by considering a local model of the merit function $\phi(x;\pi)$ around the current iterate x_k and the changes in the merit function it predicts for steps in the primal space. The model has the form

$$m_k(d;\pi) \triangleq \; f_k + g_k^T d + \max\{\tfrac{1}{2} d^T W_k d, 0\} + \pi\|c_k + A_k d\|,$$

where the max term yields a quadratic model of the objective or a linear one depending on the curvature of W_k along d. With this approximation, we can estimate the reduction in the merit function given by a step d_k by evaluating

$$\begin{aligned}\Delta m_k(d_k;\pi_k) &\triangleq m_k(0;\pi_k) - m_k(d_k;\pi_k)\\ &= -g_k^T d_k - \max\{\tfrac{1}{2} d_k^T W_k d_k, 0\} + \pi_k(\|c_k\| - \|c_k + A_k d_k\|)\\ &= -g_k^T d_k - \max\{\tfrac{1}{2} d_k^T W_k d_k, 0\} + \pi_k(\|c_k\| - \|r_k\|),\end{aligned} \tag{3.1}$$

where the residual $r_k = c_k + A_k d_k$ as in (2.6).

At the heart of our approach is the claim that a given primal-dual step is often beneficial for the algorithm to follow provided the following condition is satisfied.

Model Reduction Condition. *A step (d_k, δ_k) computed in an inexact SQP algorithm must satisfy*

$$\Delta m_k(d_k;\pi_k) \geq \sigma \pi_k \max\{\|c_k\|, \|r_k\| - \|c_k\|\} \tag{3.2}$$

for some $0 < \sigma < 1$ and appropriate $\pi_k > 0$.

We will see the effects of this condition below and in §4. In particular, (3.2) will indeed ensure that the directional derivative of the merit function is sufficiently negative along the primal step component d_k while also providing a mechanism for determining appropriate values of the penalty parameter. We note that conditions similar to the Model Reduction Condition (3.2) are presented in the context of the inexact composite-step SQP algorithm proposed by Heinkenschloss and Vicente [13]. However, their conditions are only applicable to a step that has been decomposed into basic and nonbasic components, as accuracy is imposed on the components separately. Their approach also differs from the one treated here in that they use a trust region and assume that an approximate reduced Hessian is available.

3.1. Step Acceptance Conditions. An acceptable step will be required to satisfy one of two sets of conditions. We refer to the conditions as termination tests in reference to algorithms that apply an iterative solver to the primal-dual system (2.4), as in this framework the conditions are used to determine when to terminate the iteration. Each termination test will allow us to ensure that the step satisfies (3.2) for an appropriate value of the penalty parameter and enforces requirements on the residuals (ρ_k, r_k) to ensure convergence to a local solution of (1.1). In addition, the tests impose restrictions on when the algorithm is allowed to increase the penalty parameter in order to satisfy the Model Reduction Condition (3.2).

The first termination test addresses those steps providing a sufficiently large reduction in the model of the merit function for the most recent value of the penalty parameter. We assume that an initial value $\pi_{-1} > 0$ is given.

Termination Test I. *Let $0 < \sigma, \kappa < 1$ be given constants. A step (d_k, δ_k) computed in an inexact SQP algorithm is acceptable if the Model Reduction Condition* (3.2) *holds for $\pi_k = \pi_{k-1}$ and*

$$\left\| \begin{bmatrix} \rho_k \\ r_k \end{bmatrix} \right\| \leq \kappa \left\| \begin{bmatrix} g_k + A_k^T \lambda_k \\ c_k \end{bmatrix} \right\| \tag{3.3}$$

for the residuals (ρ_k, r_k) defined by (2.6).

We claim that Termination Test I allows for productive steps to be taken that may have been computed in a relatively cheap manner, say after only a few iterations

of an iterative solver applied to the primal-dual system (2.4). For steps satisfying this test, given that a sufficient reduction in the model of the merit function has been obtained we need only enforce a generally loose bound on the residual vector. For even greater flexibility one can in fact choose $\kappa \geq 1$ in Termination Test I if the additional condition

$$\|\rho_k\| \leq \max\{\kappa_1 \|g_k + A_k^T \lambda_k\|, \kappa_2 \|c_k\|\}, \quad 0 < \kappa_1 < 1, \ 0 < \kappa_2, \tag{3.4}$$

is enforced. This may be useful, say, when applying our step acceptance criteria when steps are not computed directly via (2.4) or when the use of a left preconditioner for (2.4) produces steps corresponding to residuals larger in norm than the right-hand-side vector $(g_k + A_k^T \lambda_k, c_k)$. All of the results in the following sections hold if Termination Test I has $\kappa < 1$ or if (3.4) is included when $\kappa \geq 1$.

The second termination test addresses those steps providing a sufficiently large reduction in the linear model of the constraints.

Termination Test II. *Let $0 < \epsilon < 1$ and $0 < \beta$ be given constants. A step (d_k, δ_k) computed in an inexact SQP algorithm is acceptable if*

$$\|r_k\| \leq \ \epsilon \|c_k\| \tag{3.5a}$$

$$\text{and} \ \ \|\rho_k\| \leq \ \beta \|c_k\|, \tag{3.5b}$$

where the residuals (ρ_k, r_k) are defined by (2.6).

A step satisfying Termination Test II may not satisfy the Model Reduction Condition (3.2) for $\pi_k = \pi_{k-1}$. Thus, for such steps we require that the penalty parameter be increased to satisfy

$$\pi_k \geq \frac{g_k^T d_k + \max\{\frac{1}{2} d_k^T W_k d_k, 0\}}{(1-\tau)(\|c_k\| - \|r_k\|)} \triangleq \pi_k^{trial} \tag{3.6}$$

for a given $0 < \tau < 1$. Notice from (3.5a) and $0 < \epsilon < 1$ that the denominator in the above expression is positive and along with (3.1) the rule (3.6) implies

$$\Delta m_k(d_k; \pi_k) \geq \tau \pi_k (\|c_k\| - \|r_k\|) \geq \tau(1-\epsilon)\pi_k \|c_k\|. \tag{3.7}$$

Therefore, when (3.5a) is satisfied, the Model Reduction Condition (3.2) holds with $\sigma = \tau(1-\epsilon)$.

In summary, a step (d_k, δ_k) will be required to satisfy Termination Test I or II. In each case, the Model Reduction Condition (3.2) will hold; Termination Test I demands it explicitly and the rule (3.6) is used to enforce it when Termination Test II is satisfied. For consistency between Termination Test I and II and (3.6), one should set $\sigma = \tau(1-\epsilon)$ for Termination Test I.

The complete algorithm is the following. We refer to our step acceptance criteria as SMART Tests because they can be characterized as Sufficient Merit function Approximation Reduction Termination Tests.

Algorithm A: Inexact SQP with SMART Tests
Given parameters $0 < \kappa, \epsilon, \tau, \sigma, \eta < 1$ and $\beta > 0$
Initialize x_0, λ_0, and $\pi_{-1} > 0$
for $k = 0, 1, 2, \ldots$, until a convergence test for (1.1) is satisfied
  Compute f_k, g_k, c_k, W_k, and A_k and set $\pi_k \leftarrow \pi_{k-1}$ and $\alpha_k \leftarrow 1$
  Compute a step (d_k, δ_k) satisfying Termination Test I or II

if Termination Test II is satisfied and (3.6) does not hold, set $\pi_k \leftarrow \pi_k^{trial} + 10^{-4}$
Perform a backtracking line search to obtain α_k satisfying (2.8)
Set $(x_{k+1}, \lambda_{k+1}) \leftarrow (x_k, \lambda_k) + \alpha_k(d_k, \delta_k)$
endfor

In practice, the step can be computed by producing a sequence of candidate steps $\{(d, \delta)\}$ via the application of an iterative solver to (2.4). The corresponding residuals $\{(\rho, r)\}$ can then be computed and Termination Tests I and II can be evaluated during each iteration or after a few steps of the iterative solver. The constants $(\kappa, \epsilon, \beta)$ should be tuned for a specific application and can significantly influence the practical performance of the algorithm. In particular, the value for β should be chosen to reflect the relationship between the scales of the primal and dual feasibility measures. The scale dependence of such a parameter is not ideal, but a bound similar to (3.5b) is used to ensure the boundedness of the penalty parameter π_k (as we show in Lemma 4.7) if the rule (3.6) is enforced. Since such a method for setting the penalty parameter has proved to work well in practice [23], we employ this update rule in the algorithms in this paper and define β and (3.5b) as given. The constants (τ, σ, η) can generally be set to default values, or, in the case of σ, to promote consistency between Termination Tests I and II. Further discussion of appropriate values for the constants and an example implementation of Algorithm A are given in §5.

3.2. Well-posedness of the Algorithm. It is important to verify that the iterates specified by Algorithm A can be always be computed in practice.

Suppose that (x_k, λ_k) is an iterate that does not satisfy the optimality conditions (2.2). We argue here that whenever A_k has full row rank and W_k is positive definite on the null space of A_k, a sufficiently accurate solution to (2.4) will satisfy either Termination Test I or II. If $c(x_k) \neq 0$, then for (ρ_k, r_k) sufficiently small we have that (3.5), and so Termination Test II, will be satisfied. Otherwise, if $c(x_k) = 0$, then (3.3) will be satsfied for (ρ_k, r_k) sufficiently small. Then, since W_k is positive definite on the null space of A_k, the solution of (2.4) is the solution to problem (2.3), which means that the solution lies in the null space of A_k and corresponds to a nonpositive objective value of (2.3) (since $d = 0$ is feasible). Therefore, by computing a step with (ρ_k, r_k) sufficiently small, it can easily be seen that (3.2), and thus Termination Test I, will be satisfied.

Once an acceptable step is obtained, we must ensure that a positive steplength parameter α_k can be calculated to satisfy the Armijo condition (2.8). We consider this issue by first presenting the following result.

LEMMA 3.1. *The directional derivative of the merit function $\phi(x; \pi)$ along a step d satisfies*

$$D\phi(d; \pi) \leq g^T d - \pi(\|c\| - \|r\|).$$

Proof. Applying Taylor's theorem, we find for some constant $\gamma_1 > 0$

$$\begin{aligned}
\phi(x + \alpha d; \pi) - \phi(x; \pi) = \; & f(x + \alpha d) - f(x) + \pi(\|c(x + \alpha d)\| - \|c(x)\|) \\
\leq \; & \alpha g^T d + \gamma_1 \pi \alpha^2 \|d\|^2 + \pi(\|c(x) + \alpha A d\| - \|c(x)\|) \\
= \; & \alpha g^T d + \gamma_1 \pi \alpha^2 \|d\|^2 + \pi(\|(1 - \alpha)c(x) + \alpha r\| - \|c(x)\|) \\
\leq \; & \alpha(g^T d - \pi(\|c(x)\| - \|r\|)) + \gamma_1 \pi \alpha^2 \|d\|^2,
\end{aligned}$$

where $r = c(x) + Ad$ as in (2.6). Dividing both sides by α and taking the limit as $\alpha \to 0$ yields the result. ∎

Given this result, we present the following consequence of our Model Reduction Condition. (A stronger result will be given as Lemma 4.6.)

LEMMA 3.2. *If the Model Reduction Condition* (3.2) *holds for a step* (d_k, δ_k) *and penalty parameter* π_k*, then the directional derivative of the merit function satisfies* $D\phi(d_k; \pi_k) \leq 0$.

Proof. Observe from (3.1) that the inequality (3.2) can be rewritten as

$$g_k^T d_k - \pi_k(\|c_k\| - \|r_k\|) \leq -\max\{\tfrac{1}{2} d_k^T W_k d_k, 0\} - \sigma\pi_k \max\{\|c_k\|, \|r_k\| - \|c_k\|\},$$

so, by Lemma 3.1, a step (d_k, δ_k) satisfying (3.2) yields

$$\begin{aligned} D\phi(d_k; \pi_k) \leq\ & g_k^T d_k - \pi_k(\|c_k\| - \|r_k\|) \\ \leq\ & -\max\{\tfrac{1}{2} d_k^T W_k d_k, 0\} - \sigma\pi_k \max\{\|c_k\|, \|r_k\| - \|c_k\|\}, \end{aligned} \tag{3.8}$$

which yields the result. ∎

We have shown under common conditions that an acceptable inexact SQP step (d_k, δ_k) can always be computed by Algorithm A and that steps satisfying the Model Reduction Condition (3.2) correspond to directions of nonincrease for the merit function $\phi(x; \pi_k)$. These results allow us to show that the Armijo condition (2.8) is satisfied by some positive α_k (see Lemma 4.8), and so Algorithm A is well-posed.

We mention in passing that, as a corollary to Lemma 3.1, we may avoid the exact computation of the directional derivative of the merit function along a step d by defining the estimate

$$\tilde{D}\phi(d; \pi) \triangleq g^T d - \pi(\|c\| - \|r\|). \tag{3.9}$$

As such, the Armijo condition (2.8) can be substituted by

$$\phi(x_k + \alpha_k d_k; \pi_k) \leq \phi(x_k; \pi_k) + \eta\alpha_k \tilde{D}\phi(d_k; \pi_k). \tag{3.10}$$

All of the analysis in this paper holds when either (2.8) or (3.10) is observed in the line search procedure of Algorithm A. For convenience, we choose to use (3.10).

4. Global Analysis. Let us begin our investigation of the global behavior of Algorithm A by making the following assumptions about the problem and the set of computed iterates.

ASSUMPTIONS 4.1. *The sequence* $\{x_k, \lambda_k\}$ *generated by Algorithm A is contained in a convex set* Ω *and the following properties hold:*

(a) The functions f *and* c *and their first and second derivatives are bounded on* Ω.

(b) The sequence $\{\lambda_k\}$ *is bounded.*

(c) The constraint Jacobians A_k *have full row rank and their smallest singular values are bounded below by a positive constant.*

(d) The sequence $\{W_k\}$ *is bounded.*

(e) There exists a positive constant $\mu > 0$ *such that for any* $u \in \mathbb{R}^n$ *with* $u \neq 0$ *and* $A_k u = 0$ *we have* $u^T W_k u \geq \mu \|u\|^2$.

These assumptions are fairly standard for a line search method [12, 19]. Assumption 4.1(a) is a little weaker than the common assumption that the iterates are contained in a compact set. Assumptions 4.1(b) and (c) are strong; we use them to simplify the analysis in order to focus on the issues related to inexactness. It would

be of interest in future studies of inexact SQP methods to relax these assumptions. Assuming that W_k is positive definite on the null space of the constraints is natural for line search algorithms, for otherwise there would be no guarantee of descent. We comment further on the validity of Assumption 4.1(b) in §6.

We now assume that during iteration k we have obtained an acceptable step (d_k, δ_k) with residuals (ρ_k, r_k) defined by (2.6). We consider the decomposition

$$d_k = u_k + v_k, \tag{4.1}$$

where u_k lies in the null space of the constraint Jacobian A_k and v_k lies in the range space of A_k^T. We do not intend to compute the components explicitly; the decomposition is only for analytical purposes [4, 6]. We refer to u_k, which by definition satisfies $A_k u_k = 0$, as the tangential component and v_k as the normal component of the step.

Our analysis hinges on our ability to classify the effects of two types of steps: those lying sufficiently in the null space of the constraints and those sufficiently orthogonal to the linearized feasible region. We show that such a distinction can be made by observing the relative magnitudes of the normal and tangential components of the primal component d_k.

We first present a result related to the magnitude of the normal step.

LEMMA 4.2. *For all k, the normal component v_k is bounded in norm and for some $\gamma_2 > 0$ satisfies*

$$\|v_k\|^2 \leq \gamma_2 \max\{\|c_k\|, \|r_k\|\}. \tag{4.2}$$

Furthermore, for all k such that Termination Test II is satisfied, there exists $\gamma_3 > 0$ such that

$$\|v_k\| \leq \gamma_3(\|c_k\| - \|r_k\|). \tag{4.3}$$

Proof. From $A_k v_k = -c_k + r_k$ and the fact that v_k lies in the range space of A_k^T, it follows that

$$v_k = A_k^T(A_k A_k^T)^{-1}(-c_k + r_k),$$

and so

$$\|v_k\| \leq \|A_k^T(A_k A_k^T)^{-1}\|(\|c_k\| + \|r_k\|). \tag{4.4}$$

This, along with (3.3), the fact that Assumptions 4.1(a) and (b) imply that $\|c_k\|$ and $\|g_k + A_k^T\lambda_k\|$ are bounded, and the fact that Assumptions 4.1(a) and (c) imply that $\|A_k^T(A_k A_k^T)^{-1}\|$ is bounded, implies v_k is bounded in norm for all k. The inequality (4.4) also yields

$$\begin{aligned}
\|v_k\|^2 \leq & \left(\|A_k^T(A_k A_k^T)^{-1}\|(\|c_k\| + \|r_k\|)\right)^2 \\
\leq & \left(2\|A_k^T(A_k A_k^T)^{-1}\| \max\{\|c_k\|, \|r_k\|\}\right)^2 \\
= & \left[4\|A_k^T(A_k A_k^T)^{-1}\|^2 \max\{\|c_k\|, \|r_k\|\}\right] \max\{\|c_k\|, \|r_k\|\},
\end{aligned} \tag{4.5}$$

where (3.3) and Assumptions 4.1(a), (b), and (c) also imply that the bracketed expression in (4.5) is bounded. Thus, (4.2) holds. Finally, if Termination Test II is

satisfied, then from (3.5a) and (4.4) we have

$$\begin{aligned}\|v_k\| &\leq \|A_k^T(A_kA_k^T)^{-1}\|(1+\epsilon)\|c_k\| \\ &\leq \|A_k^T(A_kA_k^T)^{-1}\|\left(\tfrac{1+\epsilon}{1-\epsilon}\right)(\|c_k\|-\|r_k\|),\end{aligned}$$

and so (4.3) holds. □

A similar result can be proved for the tangential component.

LEMMA 4.3. *The tangential components $\{u_k\}$ are bounded in norm.*

Proof. Assumption 4.1(e), the fact that u_k lies in the null space of the constraint Jacobian A_k, and the first block equation of (2.6) yield

$$\begin{aligned}\mu\|u_k\|^2 &\leq u_k^TW_ku_k \\ &= -g_k^Tu_k+\rho_k^Tu_k-u_k^TW_kv_k \\ &\leq (\|g_k\|+\|\rho_k\|+\|W_kv_k\|)\|u_k\|,\end{aligned}$$

and so

$$\|u_k\| \leq (\|g_k\|+\|\rho_k\|+\|W_kv_k\|)/\mu.$$

The result follows from the facts that Assumptions 4.1, Lemma 4.2, and the bounds (3.3) and (3.5b) imply that all terms in the right-hand-side of this inequality are bounded. □

We now turn to the following result addressing the relative magnitudes of the normal and tangential components of a given step.

LEMMA 4.4. *There exists a constant $\gamma_4 > 0$ such that if $\|u_k\|^2 \geq \gamma_4\|v_k\|^2$ then $\frac{1}{2}d_k^TW_kd_k \geq \frac{\mu}{4}\|u_k\|^2$.*

Proof. Assumption 4.1(e) implies that for any γ_4 such that $\|u_k\|^2 \geq \gamma_4\|v_k\|^2$ we have

$$\begin{aligned}\tfrac{1}{2}d_k^TW_kd_k &= \tfrac{1}{2}u_k^TW_ku_k+u_k^TW_kv_k+\tfrac{1}{2}v_k^TW_kv_k \\ &\geq \tfrac{\mu}{2}\|u_k\|^2-\|u_k\|\|W_k\|\|v_k\|-\tfrac{1}{2}\|W_k\|\|v_k\|^2 \\ &\geq \left(\frac{\mu}{2}-\frac{\|W_k\|}{\sqrt{\gamma_4}}-\frac{\|W_k\|}{2\gamma_4}\right)\|u_k\|^2.\end{aligned}$$

Thus, Assumption 4.1(d) implies the result holds for a sufficiently large $\gamma_4 > 0$. □

With the above results, we can now formalize a distinction between two types of steps. Let $\gamma_4 > 0$ be chosen large enough as described in Lemma 4.4 and consider the sets of indices

$$\begin{aligned}K_1 &\triangleq \{k : \|u_k\|^2 \geq \gamma_4\|v_k\|^2\} \\ \text{and}\quad K_2 &\triangleq \{k : \|u_k\|^2 < \gamma_4\|v_k\|^2\}.\end{aligned}$$

Most of the remainder of our analysis will be dependent on these sets and the corresponding quantity

$$\Theta_k \triangleq \begin{cases}\|u_k\|^2+\|c_k\|, & k \in K_1, \\ \max\{\|c_k\|,\|r_k\|\}, & k \in K_2.\end{cases}$$

The relevance of Θ_k will be seen in the following three lemmas as a quantity that can be used for bounding the length of the primal step and the directional derivative of the merit function, which will then provide a lower bound for the sequence of steplength coefficients $\{\alpha_k\}$.

LEMMA 4.5. *There exists $\gamma_5 > 1$ such that, for all k,*

$$\|d_k\|^2 \leq \gamma_5 \Theta_k,$$

and hence

$$\|d_k\|^2 + \|c_k\| \leq 2\gamma_5 \Theta_k. \tag{4.6}$$

Proof. For $k \in K_1$, we find

$$\begin{aligned} \|d_k\|^2 = & \ \|u_k\|^2 + \|v_k\|^2 \\ \leq & \ \left(1 + \tfrac{1}{\gamma_4}\right) \|u_k\|^2 \\ \leq & \ \left(1 + \tfrac{1}{\gamma_4}\right) (\|u_k\|^2 + \|c_k\|). \end{aligned}$$

Similarly, Lemma 4.2 implies that for $k \in K_2$

$$\begin{aligned} \|d_k\|^2 = & \ \|u_k\|^2 + \|v_k\|^2 \\ < & \ (\gamma_4 + 1)\|v_k\|^2 \\ \leq & \ (\gamma_4 + 1)\gamma_2 \max\{\|c_k\|, \|r_k\|\}. \end{aligned}$$

To establish (4.6) we note that $\Theta_k + \|c_k\| \leq 2\Theta_k$ for all k. ∎

The directional derivative of the merit function can be bounded in a similar manner.

LEMMA 4.6. *There exists $\gamma_6 > 0$ such that, for all k,*

$$\tilde{D}\phi(d_k; \pi_k) \leq -\gamma_6 \Theta_k.$$

Proof. Recalling (3.8) and (3.9) we have

$$\tilde{D}\phi(d_k; \pi_k) \leq -\max\{\tfrac{1}{2} d_k^T W_k d_k, 0\} - \sigma\pi_k \max\{\|c_k\|, \|r_k\| - \|c_k\|\}. \tag{4.7}$$

By Lemma 4.4, we have that $\frac{1}{2} d_k^T W_k d_k \geq \frac{\mu}{4}\|u_k\|^2$ for $k \in K_1$ and thus

$$\tilde{D}\phi(d_k; \pi_k) \leq -\tfrac{\mu}{4}\|u_k\|^2 - \sigma\pi_k \|c_k\|.$$

Similarly, for $k \in K_2$ we have from (4.7) that

$$\begin{aligned} \tilde{D}\phi(d_k; \pi_k) \leq & \ -\sigma\pi_k \max\{\|c_k\|, \|r_k\| - \|c_k\|\} \\ \leq & \ -\tfrac{1}{2}\sigma\pi_k \max\{\|c_k\|, \|r_k\|\}. \end{aligned}$$

The result holds for $\gamma_6 = \min\{\frac{\mu}{4}, \frac{1}{2}\sigma\pi_k\}$, which is bounded away from zero as $\{\pi_k\}$ is nondecreasing. ∎

Another important property of Algorithm A is that under Assumptions 4.1 the penalty parameter remains bounded. We prove this result in the following lemma, illustrating the importance of the bound (3.5b).

LEMMA 4.7. *The sequence of penalty parameters* $\{\pi_k\}$ *is bounded above and* $\pi_k = \pi_{\bar{k}}$ *for all* $k \geq \bar{k}$ *for some* $\bar{k} \geq 0$.

Proof. Recall that the penalty parameter is increased during iteration k of Algorithm A only if Termination Test II is satisfied. Therefore, for the remainder of this proof let us assume that Termination Test II is satisfied and so the inequalities in (3.5) hold. By (3.6) the parameter π_k is chosen to satisfy the first inequality in (3.7), namely

$$\Delta m_k(d_k; \pi_k) \geq \tau \pi_k (\|c_k\| - \|r_k\|), \tag{4.8}$$

where, according to the first block equation of (2.6), we can rewrite the model reduction as

$$\begin{aligned}
\Delta m_k(d_k; \pi_k) = \; & \pi_k(\|c_k\| - \|r_k\|) - g_k^T d_k - \max\{\tfrac{1}{2} d_k^T W_k d_k, 0\} \\
= \; & \pi_k(\|c_k\| - \|r_k\|) \\
& + \begin{cases} -g_k^T v_k - \frac{1}{2} v_k^T W_k v_k - \rho_k^T u_k + \frac{1}{2} u_k^T W_k u_k & \text{if } \frac{1}{2} d_k^T W_k d_k \geq 0 \\ -g_k^T v_k - (\rho_k - W_k v_k)^T u_k + u_k^T W_k u_k & \text{otherwise.} \end{cases}
\end{aligned}$$

The result follows from our ability to bound the terms in the second line of this expression with respect to the constraint reduction.

If $\frac{1}{2} d_k^T W_k d_k \geq 0$, then under Assumptions 4.1 we have that Lemmas 4.2 and 4.3 and the bounds (3.5) on the residuals (ρ_k, r_k) imply that there exists $\gamma_7, \gamma_7' > 0$ such that

$$\begin{aligned}
-g_k^T v_k - \tfrac{1}{2} v_k^T W_k v_k - \rho_k^T u_k + \tfrac{1}{2} u_k^T W_k u_k \geq \; & -\|g_k\|\|v_k\| - \tfrac{1}{2}\|W_k\|\|v_k\|^2 - \|\rho_k\|\|u_k\| \\
\geq \; & -\gamma_7(\|v_k\| + \|\rho_k\|) \\
\geq \; & -\gamma_7 \left(\gamma_3 + \tfrac{\beta}{1-\epsilon}\right)(\|c_k\| - \|r_k\|) \\
= \; & -\gamma_7'(\|c_k\| - \|r_k\|).
\end{aligned}$$

Similarly, if $\frac{1}{2} d_k^T W_k d_k < 0$, then under Assumptions 4.1 we again find that Lemmas 4.2 and 4.3 and the bounds (3.5) on the residuals (ρ_k, r_k) imply that there exists $\gamma_8, \gamma_8' > 0$ such that

$$\begin{aligned}
-g_k^T v_k - (\rho_k - W_k v_k)^T u_k + u_k^T W_k u_k \geq \; & -\|g_k\|\|v_k\| - \|\rho_k\|\|u_k\| - \|v_k\|\|W_k\|\|u_k\| \\
\geq \; & -\gamma_8(\|v_k\| + \|\rho_k\|) \\
\geq \; & -\gamma_8 \left(\gamma_3 + \tfrac{\beta}{1-\epsilon}\right)(\|c_k\| - \|r_k\|) \\
= \; & -\gamma_8'(\|c_k\| - \|r_k\|).
\end{aligned}$$

These results together imply

$$\Delta m_k(d_k; \pi_k) \geq (\pi_k - \max\{\gamma_7', \gamma_8'\})(\|c_k\| - \|r_k\|),$$

and so (4.8) is always satisfied for

$$\pi_k \geq \max\{\gamma_7', \gamma_8'\}/(1 - \tau).$$

Therefore, if $\pi_{\bar{k}} \geq \max\{\gamma_7', \gamma_8'\}/(1-\tau)$ for some $\bar{k} \geq 0$, then $\pi_k = \pi_{\bar{k}}$ for $k \geq \bar{k}$. This, along with the fact that whenever Algorithm A increases the penalty parameter it does so by at least a positive finite amount, proves the result. ∎

The previous three lemmas can be used to bound the sequence of steplength coefficients.

LEMMA 4.8. *The sequence* $\{\alpha_k\}$ *is bounded below and away from zero.*

Proof. Recall that the line search requires (3.10), which we rewrite for convenience as

$$\phi(x_k + \alpha_k d_k; \pi_k) - \phi(x_k; \pi_k) \leq \eta \alpha_k \tilde{D}\phi(d_k; \pi_k).$$

Suppose that the line search fails for some $\bar{\alpha} > 0$, so

$$\phi(x_k + \bar{\alpha} d_k; \pi_k) - \phi(x_k; \pi_k) > \eta \bar{\alpha} \tilde{D}\phi(d_k; \pi_k).$$

From the proof of Lemma 3.1 and (3.9) we have

$$\phi(x_k + \bar{\alpha} d_k; \pi_k) - \phi(x_k; \pi_k) \leq \bar{\alpha} \tilde{D}\phi(d_k; \pi_k) + \bar{\alpha}^2 \gamma_1 \pi_k \|d_k\|^2,$$

so

$$(\eta - 1)\tilde{D}\phi(d_k; \pi_k) \leq \bar{\alpha} \gamma_1 \hat{\pi} \|d_k\|^2.$$

Here, $\hat{\pi}$ is a finite upper bound for the sequence $\{\pi_k\}$ whose existence follows from Lemma 4.7. Lemmas 4.5 and 4.6 then yield

$$(1 - \eta)\gamma_6 \Theta_k < \bar{\alpha} \gamma_1 \gamma_5 \hat{\pi} \Theta_k,$$

so

$$\bar{\alpha} > (1 - \eta)\gamma_6 / (\gamma_1 \gamma_5 \hat{\pi}).$$

Thus, α_k need never be set below $(1 - \eta)\gamma_6/(\gamma_1 \gamma_5 \hat{\pi})$ for (3.10) to be satisfied. ∎

We can now present the following result related to the lengths of the primal components of the steps computed in Algorithm A and the convergence of the iterates toward the feasible region of problem (1.1).

LEMMA 4.9. *Algorithm A yields*

$$\lim_{k \to \infty} \|c_k\| = 0 \quad \textit{and} \quad \lim_{k \to \infty} \|d_k\| = 0.$$

Proof. For all k, it can easily be seen that

$$\phi(x_k; \pi_k) - \phi(x_k + \alpha_k d_k; \pi_k) \geq \gamma_9 \Theta_k$$

for some $\gamma_9 > 0$ follows from (3.10) and Lemmas 4.6 and 4.8. By Lemma 4.7 the algorithm eventually computes, during iteration $\bar{k} \geq 0$, a finite value $\pi_{\bar{k}}$ beyond which the penalty parameter will never be increased. Therefore, the penalty parameter remains unchanged for $k \geq \bar{k}$ and for all $k > \bar{k}$ (4.6) implies

$$\begin{aligned}
\phi(x_{\bar{k}}; \pi_{\bar{k}}) - \phi(x_k; \pi_{\bar{k}}) &= \sum_{j=\bar{k}}^{k-1} (\phi(x_j; \pi_{\bar{k}}) - \phi(x_{j+1}; \pi_{\bar{k}})) \\
&\geq \gamma_9 \sum_{j=\bar{k}}^{k-1} \Theta_j \\
&\geq \frac{\gamma_9}{2\gamma_5} \sum_{j=\bar{k}}^{k-1} (\|d_j\|^2 + \|c_j\|).
\end{aligned}$$

The result follows from the above and the fact that Assumption 4.1(a) implies $\phi(x;\pi_{\bar{k}})$ is bounded below. ∎

We are now ready to present the main result of this section.

THEOREM 4.10. *Algorithm A yields*

$$\lim_{k\to\infty} \left\| \begin{bmatrix} g_k + A_k^T\lambda_k \\ c_k \end{bmatrix} \right\| = 0.$$

Proof. Recall that $\alpha_k \leq 1$ for all k and from Lemma 4.8 we have that $\{\alpha_k\}$ is bounded below and away from zero. An expansion of the first block of the optimality conditions (2.2) yields

$$\|g_{k+1} + A_{k+1}^T\lambda_{k+1}\| \leq \|g_k + A_k^T\lambda_k + \alpha_k(\nabla_{xx}^2\mathcal{L}_k d_k + A_k^T\delta_k)\| + \alpha_k^2 E(d_k,\delta_k),$$

where

$$E(d_k,\delta_k) = O(\|d_k\|^2 + \|d_k\cdot\delta_k\|).$$

This, along with the first block equation in (2.6) and Assumptions 4.1, implies

$$\begin{aligned}
&\|g_{k+1} + A_{k+1}^T\lambda_{k+1}\| \\
&\quad\leq \|g_k + A_k^T\lambda_k + \alpha_k(W_k d_k + A_k^T\delta_k) + \alpha_k(\nabla_{xx}^2\mathcal{L}_k - W_k)d_k\| + \alpha_k^2 E(d_k,\delta_k) \\
&\quad\leq \|g_k + A_k^T\lambda_k + \alpha_k(\rho_k - g_k - A_k^T\lambda_k)\| + \alpha_k\|(\nabla_{xx}^2\mathcal{L}_k - W_k)d_k\| + \alpha_k^2 E(d_k,\delta_k) \\
&\quad\leq (1-\alpha_k)\|g_k + A_k^T\lambda_k\| + \alpha_k\|\rho_k\| + \alpha_k E'(d_k,\delta_k) \qquad (4.9)
\end{aligned}$$

where

$$E'(d_k,\delta_k) = O(\|d_k\| + \|d_k\|^2 + \|d_k\cdot\delta_k\|). \qquad (4.10)$$

The bounds (3.3) and (3.5b) and the triangle inequality imply

$$\|\rho_k\| \leq \max\{\kappa(\|g_k + A_k^T\lambda_k\| + \|c_k\|), \beta\|c_k\|\}$$

which, along with (4.9) and the boundedness of $\{\alpha_k\}$, implies that for some $0 < \gamma_{10} < 1$ and $\gamma_{11} > 0$ we have

$$\|g_{k+1} + A_{k+1}^T\lambda_{k+1}\| \leq \max\{(1-\gamma_{10})\|g_k + A_k^T\lambda_k\|, \gamma_{11}\|c_k\|\} + \alpha_k E'(d_k,\delta_k). \quad (4.11)$$

The boundedness of $\{\alpha_k\}$, Lemma 4.9, and the fact that Assumption 4.1(b) implies δ_k is bounded in norm imply, along with (4.10), that

$$\lim_{k\to\infty} \alpha_k E'(d_k,\delta_k) = 0. \qquad (4.12)$$

Consider an arbitrary $\hat{\gamma} > 0$. Lemma 4.9 and the limit (4.12) imply that there exists $k' \geq 0$ such that for all $k \geq k'$ we have

$$\gamma_{11}\|c_k\| < (1-\gamma_{10})\hat{\gamma} \quad\text{and}\quad \alpha_k E'(d_k,\delta_k) < \tfrac{1}{2}\gamma_{10}\hat{\gamma}. \qquad (4.13)$$

Suppose $k \geq k'$ and $\|g_k + A_k^T\lambda_k\| > \hat{\gamma}$. We find from (4.11) that

$$\begin{aligned}
\|g_{k+1} + A_{k+1}^T\lambda_{k+1}\| &\leq (1-\gamma_{10})\|g_k + A_k^T\lambda_k\| + \tfrac{1}{2}\gamma_{10}\hat{\gamma} \\
&\leq \|g_k + A_k^T\lambda_k\| - \tfrac{1}{2}\gamma_{10}\hat{\gamma}.
\end{aligned}$$

Therefore, $\{\|g_k + A_k^T\lambda_k\|\}$ decreases monotonically by at least a constant amount for $k \geq k'$ while $\{\|g_k + A_k^T\lambda_k\|\} > \hat{\gamma}$, so we eventually find $\|g_k + A_k^T\lambda_k\| \leq \hat{\gamma}$ for some $k = k'' \geq k'$. Then, for $k \geq k''$ we find from (4.11) and (4.13) that

$$\begin{aligned}\|g_{k+1} + A_{k+1}^T\lambda_{k+1}\| \leq\ & (1-\gamma_{10})\hat{\gamma} + \tfrac{1}{2}\gamma_{10}\hat{\gamma} \\ \leq\ & (1-\tfrac{1}{2}\gamma_{10})\hat{\gamma},\end{aligned}$$

so $\|g_k + A_k^T\lambda_k\| \leq \hat{\gamma}$ for all $k \geq k''$. Since the above holds for any $\hat{\gamma} > 0$, we have

$$\lim_{k\to\infty} \|g_k + A_k^T\lambda_k\| = 0,$$

and so the result follows with the above and the result of Lemma 4.9. ∎

5. An Implementation. This section contains a description of a particular implementation of Algorithm A and corresponding numerical results to illustrate the robustness of our approach. Note that, for the greatest level of generality within our framework, we implemented Termination Test I with $\kappa \geq 1$ and (3.4) included. A study of the efficiency of the new algorithm in realistic applications is devoted to a separate study [7].

We developed a Matlab implementation of Algorithm A in which the generalized minimum residual (GMRES) method [21] was used for the step computation, for which we adapted the implementation by Kelley [15]. The GMRES method does not exploit the symmetry of the matrix (2.5) in the primal-dual system (2.6), but the stability of the approach is ideal for illustrating the robustness of Algorithm A.

In terms of the input parameters defined throughout the paper, we make the following general comments on their practical effects. First, the values $(\kappa, \kappa_1, \kappa_2)$ and (ϵ, β) should receive special attention as they may greatly affect the ease with which Termination Tests I and II, and therefore the Model Reduction Condition (3.2), are satisfied; larger values for these constants allow for more steps to satisfy at least one of the tests at a given point. In general, looser bounds in Termination Tests I and II will result in cheaper step computations, but these savings must be balanced against possible increases in the number of outer iterations required to find a solution. These parameters and (σ, τ) may also affect the number of iterations required until the penalty parameter stabilizes, an important phenomenon in the analysis of §4; e.g., larger values of (ϵ, β) may lead to more increases and/or larger values of π_k. In general, however, we claim that the parameters (σ, τ) can be set to default values or to promote consistency between the two termination tests, as we do in (5.3) below.

The stopping condition for the overall nonlinear program is given by

$$\|g_k + A_k^T\lambda_k\|_\infty \leq \max\{\|g_k\|_\infty, 1\}\epsilon_{opt}, \tag{5.1}$$

$$\|c_k\|_\infty \leq \max\{\|c_0\|_\infty, 1\}\epsilon_{feas}, \tag{5.2}$$

where $0 < \epsilon_{opt}, \epsilon_{feas} < 1$ and x_0 is the starting point (e.g., see [23]).

The following algorithm was implemented in Matlab and will be referred to as `isqp`. The `termination` variable is used to indicate the successful or unsuccessful termination of the solver near a local solution of problem (1.1).

Algorithm B: Inexact SQP with GMRES and SMART Tests
Given parameters $0 < \epsilon_{feas}, \epsilon_{opt}, \kappa_1, \epsilon, \tau, \sigma, \eta, \alpha_{\min} < 1$ and $0 < k_{\max}, \beta, \kappa, \kappa_2$
Initialize x_0, λ_0, and $\pi_{-1} > 0$
Set `termination` ← `success`

for $k = 0, 1, 2, \ldots, k_{max}$, or until (5.1) and (5.2) are satisfied
  Compute f_k, g_k, c_k, W_k, and A_k and set $\pi_k \leftarrow \pi_{k-1}$ and $\alpha_k \leftarrow 1$
  for $j = 0, 1, 2, \ldots, n + t$, or until Termination Test I or II is satisfied
    Set (d_k, δ_k) as the jth GMRES solution
  endfor
  if $\tilde{D}\phi(d_k; \pi) > 0$ for all $\pi \geq \pi_k$, set `termination` $\leftarrow$ `failure` and **break**
  if Termination Test II is satisfied and (3.6) does not hold, set $\pi_k \leftarrow \pi_k^{trial} + 10^{-4}$
  while (3.10) is not satisfied and $\alpha_k \geq \alpha_{\min}$, set $\alpha_k \leftarrow \alpha_k/2$
  if $\alpha_k < \alpha_{\min}$, set `termination` $\leftarrow$ `failure` and **break**
  Set $(x_{k+1}, \lambda_{k+1}) \leftarrow (x_k, \lambda_k) + \alpha_k(d_k, \delta_k)$
endfor
if (5.1) or (5.2) is not satisfied, set `termination` $\leftarrow$ `failure`
return `termination`

We recognize three types of failures in the above approach. First, due to the iteration limit $(n + t)$ imposed on the inner **for** loop, or if the positive definiteness of Assumption 4.1(e) is violated, GMRES may not provide a solution satisfying Termination Test I or II. In this case, we will try to use the step d_k anyway, and, if necessary, we will try increasing π_k to yield a positive value for the directional derivative $D\phi(d_k; \pi_k)$. However, if the directional derivative is nonnegative for any value $\pi \geq \pi_{k-1}$ of the penalty parameter, then the step is an ascent direction for the merit function and the algorithm terminates. Second, if the steplength coefficient must be cut below a given $\alpha_{\min}$ in order to obtain a step satisfying the Armijo condition, then the search direction is deemed unsuitable and the algorithm fails. Since we have a descent direction this failure can only occur due to finite precision arithmetic errors, or if $\alpha_{\min}$ is too large relative to the curvature of the functions. Finally, if the algorithm terminates without satisfying the nonlinear program stopping conditions (5.1) and (5.2), then the maximum number of iterations has been reached. Though there exist techniques for continuing a stagnated run of the algorithm when an ascent direction for the merit function or a short steplength coefficient is computed, we implement naïve failure tests in Algorithm B to aggressively challenge the robustness of our approach.

Table 5.1 contains a listing of the input parameters implemented in our code. For

Parameter	Value	Parameter	Value
ϵ_{feas}	10^{-6}	η	10^{-8}
ϵ_{opt}	10^{-6}	$\alpha_{\min}$	10^{-8}
κ_1	0.1	$k_{\max}$	1000
ϵ	0.1	κ	1
τ	0.1	π_{-1}	1

TABLE 5.1
Input parameter values used for Algorithm B

the remaining parameters, we set, as is generally appropriate,

$$\sigma \leftarrow \tau(1 - \epsilon) \tag{5.3}$$

$$\text{and} \quad \kappa_2 \leftarrow \beta \leftarrow \max\left\{ \frac{\|g_0 + A_0^T \lambda_0\|}{\|c_0\| + 1}, 1 \right\}.$$

As previously mentioned, this value for σ promotes consistency between Termination Tests I and II and (3.6). Such a value for κ_2 and β aims to reflect the relationship in scale between the primal and dual feasibility measures.

We compare Algorithm B with an inexact method that only enforces a reduction in the entire primal-dual residual. Our implementation of this approach, also done in Matlab, is identical to Algorithm B except that the GMRES stopping test

for $j = 0, 1, 2, \ldots, n + t$, *or until Termination Test I or II is satisfied*

is replaced by

for $j = 0, 1, 2, \ldots, n + t$, *or until* (3.3) *is satisfied*

where $0 < \kappa < 1$ for (3.3) is a given constant. We performed multiple runs of this algorithm, which we call `ires`, for each problem in the test set and will refer to each run by the particular value of κ used.

The algorithms described above were run for 44 equality constrained problems from the CUTEr [3, 10] and COPS [9] collections. Problems from the CUTEr set for which AMPL models were available were selected based on size – fewer than 10,000 variables – and two moderately sized COPS problems were chosen. We note that W_k was set to the exact Hessian of the Lagrangian and that a multiple of the identity matrix was added to W_k, when necessary, to satisfy the positive definiteness of Assumption 4.1(e). Also, as the results provided in this section are intended only as a simple illustration of the robustness of our approach, we did not implement a preconditioner for the primal-dual system for our numerical experiments and, in fact, this was not an issue as many of the problems are relatively small in size. We stress, however, that preconditioning is an essential part of any implementation for many large-scale problems.

Table 5.2 provides the percentage of problems successfully solved for each of the solvers. All of the failures for the `ires` algorithm occurred due to the fact that

Algorithm	`ires`										`isqp`
κ	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}	2^{-8}	2^{-9}	2^{-10}	-
% Solved	45%	66%	68%	80%	80%	77%	82%	82%	86%	86%	100%

TABLE 5.2

Algorithm success rates; comparison between an inexact SQP method based on the entire residual of the Newton equations and `isqp`*, the algorithm proposed in this paper*

either the directional derivative $D\phi(d_k; \pi)$ of the merit function was nonnegative for all allowable values of the penalty parameter $\pi \geq \pi_{k-1}$ or the backtracking line search reduced the steplength coefficient α_k below the given tolerance $\alpha_{\min}$. Thus, we find that even for relatively small values of the tolerance parameter κ, the primal component d_k provided by GMRES can yield a value for the directional derivative $D\phi(d_k; \pi)$ of the merit function that is not sufficiently negative for any $\pi \geq \pi_{k-1}$. In other words, `ires` runs the risk of computing nearly-exact solutions of the primal-dual system (2.4) that correspond to directions of insufficient decrease for the merit function $\phi(x; \pi_k)$.

6. Final Remarks. In this paper we have developed an inexact SQP algorithm for equality constrained optimization that is globally convergent under common conditions. The novelties of the approach are centered around a pair of Sufficient Merit function Approximation Reduction Termination Tests (or SMART Tests for short) for controlling the level of inexactness in the step computation procedure. We close with some remarks about the assumptions used in the paper, the rate of convergence of our approach, and possible extensions of this work.

First, let us recall the boundedness of the multipliers stated in Assumption 4.1(b). Our analysis does not guarantee that the multipliers remain bounded in general; in fact, Algorithm A does not exert direct control over them. We can ensure that $\{\lambda_k\}$ remains bounded, however, by adding to Termination Test I a requirement of the form

$$\|\rho_k\| \leq \kappa' \max\{\|g_k\|, \|A_k\|\}$$

for a constant $\kappa' > 0$. Such a condition ensures that ρ_k is bounded independently of the multipliers λ_k, so then (2.6) and Assumptions 4.1 will imply that $\{\lambda_k\}$ is bounded. An alternative would be to include a safeguard in the algorithm by which the multiplier estimate λ_k is set to a nominal value, say $\lambda_k = 0$, if $\|g_k + A_k^T \lambda_k\|$ is larger than a given constant.

Second, the rate of convergence of Algorithm A may be slow for a given problem. One can ensure a fast convergence rate, however, by imposing at each step a requirement of the form

$$\left\| \begin{bmatrix} \rho_k \\ r_k \end{bmatrix} \right\| \leq \kappa_k \left\| \begin{bmatrix} g_k + A_k^T \lambda_k \\ c_k \end{bmatrix} \right\| \tag{6.1}$$

where $0 < \{\kappa_k\} < 1$ [8]. Then, tightening the values of κ_k during any point of a run of Algorithm A will influence the convergence rate if unit steplengths are taken. For example, if $\kappa_k \leq \hat{\kappa} < 1$ for all large k, then the rate of convergence is linear with rate $\hat{\kappa}$. If, in addition, $\kappa_k \to 0$, then the rate of convergence is superlinear [8]. In practice, the exact penalty function (2.7) can reject unit steps even close to the solution, but this difficulty can be overcome by the use of a second order correction or non-monotone techniques [18]. In this manner, we can be sure that the rate of convergence of Algorithm A will be fast once the penalty parameter is stabilized.

Incidentally, by implementing such an approach, where we require the step provided by the iterative linear system solver to satisfy both (6.1) and Termination Test I or II, one can directly observe the extra cost associated with evolving the `ires` algorithm described in the previous section into a robust method. In our experiments we found this extra cost to be minimal for the problems in our test set. For example, let us define a third algorithm, call it `isqp-ires`, that imposes inequality (6.1) along with our termination tests within the step computation of Algorithm B, where $\kappa = \kappa_k = 2^{-5}$ for all k. Note that the key differences between `isqp-ires` and `isqp` are that we have now implemented $\kappa < 1$ for (3.3) and that an inequality of the form (3.3)/(6.1) is also enforced in Termination Test II. Now, if we compare `isqp-ires` with `ires` (with $\kappa = 2^{-5}$), we can observe the extra cost required to satisfy our termination tests beyond simply attaining an accurate solution to the primal-dual system (2.4). It turns out that for the 35 problems solved by both of these algorithms, an average of only 0.5 extra total GMRES iterations over the entire run of the algorithm were required by `isqp-ires`. Moreover, by observing the termination tests for the iterative solver, the 9 problems left unsolved by `ires` (approximately 20% of the total number of 44 problems) were all solved successfully by `isqp-ires`. Indeed, the extra cost is minimal with respect to the added robustness.

In addition it is worth noting that imposing condition (6.1) with sufficiently small κ_k implies that the bound (3.4) would automatically be satisfied, and the bounds (3.5) of Termination Test II are satisfied in the case where $\|c_k\|$ is greater than some constant times $\|g_k + A_k^T \lambda_k\|$.

Finally, it would be of interest to analyze the behavior of inexact SQP methods in the presence of Jacobian singularities and when $W_k = \nabla^2_{xx} \mathcal{L}_k$ for some k with W_k

not positive definite in the null space of the constraint Jacobian A_k. However, such an analysis can be complex and would have brought the focus away from the intended scope of this paper. Therefore, we chose to discuss the design of inexact SQP methods in the benign context of Assumptions 4.1.

Acknowledgments: The authors are thankful to Eldad Haber and Nick Gould for productive discussions on this work.

REFERENCES

[1] G. Biros and O. Ghattas, *Parallel Lagrange-Newton-Krylov-Schur methods for PDE-constrained optimization. Part I: the Krylov-Schur solver*, SIAM Journal on Scientific Computing, 27 (2005), pp. 687–713.

[2] ———, *Parallel Lagrange-Newton-Krylov-Schur methods for PDE-constrained optimization. Part II: the Lagrange-Newton solver, and its application to optimal control of steady viscous flows*, SIAM Journal on Scientific Computing, 27 (2005), pp. 714–739.

[3] I. Bongartz, A. R. Conn, N. I. M. Gould, and Ph. L. Toint, CUTE*: Constrained and Unconstrained Testing Environment*, ACM Transactions on Mathematical Software, 21 (1995), pp. 123–160.

[4] R. H. Byrd, J.-Ch. Gilbert, and J. Nocedal, *A trust region method based on interior point techniques for nonlinear programming*, Mathematical Programming, 89 (2000), pp. 149–185.

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

[6] R. H. Byrd and J. Nocedal, *An analysis of reduced Hessian methods for constrained optimization*, Mathematical Programming, 49 (1991), pp. 285–323.

[7] F. E. Curtis and E. Haber, *Numerical experience with an inexact SQP method for PDE-constrained optimization*, (2007). In preparation.

[8] R. S. Dembo, S. C. Eisenstat, and T. Steihaug, *Inexact-Newton methods*, SIAM Journal on Numerical Analysis, 19 (1982), pp. 400–408.

[9] E. D. Dolan, J. J. Moré, and T. S. Munson, *Benchmarking optimization software with COPS 3.0*, Tech. Report ANL/MCS-TM-273, Argonne National Laboratory, Argonne, Illinois, USA, 2004.

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

[11] E. Haber and U. M. Ascher, *Preconditioned all-at-once methods for large, sparse parameter estimation problems*, Inverse Problems, 17 (2001), pp. 1847–1864.

[12] S. P. Han, *A globally convergent method for nonlinear programming*, Journal of Optimization Theory and Applications, 22 (1977), pp. 297–309.

[13] M. Heinkenschloss and L. N. Vicente, *Analysis of inexact trust-region SQP algorithms*, SIAM Journal on Optimization, 12 (2001), pp. 283–302.

[14] H. Jäger and E. W. Sachs, *Global convergence of inexact reduced SQP methods*, Optimization Methods and Software, 7 (1996), pp. 83–110.

[15] C. T. Kelley, *Iterative methods for linear and nonlinear equations: Matlab codes*, 1994. `http://www4.ncsu.edu/~ctk/matlab_roots.html`.

[16] M. Lalee, J. Nocedal, and T. D. Plantenga, *On the implementation of an algorithm for large-scale equality constrained optimization*, SIAM Journal on Optimization, 8 (1998), pp. 682–706.

[17] F. Leibfritz and E. W. Sachs, *Inexact SQP interior point methods and large scale optimal control problems*, SIAM Journal on Control and Optimization, 38 (1999), pp. 272–293.

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

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

[20] E. E. Prudencio, R. Byrd, and X. C. Cai, *Parallel full space SQP Lagrange-Newton-Krylov-Schwarz algorithms for PDE-constrained optimization problems*, SIAM Journal on Scientific Computing, 27 (2006), pp. 1305–1328.

[21] Y. Saad and M. H. Schultz, *GMRES: A generalized minimal residual algorithm for solving*

nonsymmetric linear systems, SIAM Journal on Scientific and Statistical Computing, 7 (1986), pp. 856–869.

[22] A. Walther, *A first-order convergence analysis of trust-region methods with inexact Jacobians*, Tech. Report MATH-WR-01-2005, Institute of Scientific Computing, Technische Universitat Dresden, Dresden, Germany, 2005.

[23] R. A. Waltz, J. L. Morales, J. Nocedal, and D. Orban, *An interior algorithm for nonlinear optimization that combines line search and trust region steps*, Mathematical Programming, Series A, 107 (2006), pp. 391–408.