Math. Program., Ser. A (2010) 122:273–299
DOI 10.1007/s10107-008-0248-3

FULL LENGTH PAPER

An inexact Newton method for nonconvex equality constrained optimization

Richard H. Byrd · Frank E. Curtis · Jorge Nocedal

Received: 15 August 2007 / Accepted: 26 August 2008 / Published online: 28 October 2008

Abstract We present a matrix-free line search algorithm for large-scale equality constrained optimization that allows for inexact step computations. For strictly convex problems, the method reduces to the inexact sequential quadratic programming approach proposed by Byrd et al. [SIAM J. Optim. 19(1) 351–369, 2008]. For nonconvex problems, the methodology developed in this paper allows for the presence of negative curvature without requiring information about the inertia of the primal–dual iteration matrix. Negative curvature may arise from second-order information of the problem functions, but in fact exact second derivatives are not required in the approach. The complete algorithm is characterized by its emphasis on sufficient reductions in a model of an exact penalty function. We analyze the global behavior of the algorithm and present numerical results on a collection of test problems.

Richard H. Byrd was supported by National Science Foundation grants CCR-0219190 and CHE-0205170.
Frank E. Curtis was supported by Department of Energy grant DE-FG02-87ER25047-A004.
Jorge Nocedal was supported by National Science Foundation grant CCF-0514772 and by a grant from the Intel Corporation.

R. H. Byrd
Department of Computer Science, University of Colorado, Boulder, CO 80309, USA
e-mail: richard@cs.colorado.edu

F. E. Curtis (✉)
Department of Industrial Engineering and Management Sciences, Northwestern University,
Evanston, IL 60208, USA
e-mail: f-curtis@northwestern.edu

J. Nocedal
Department of Electrical Engineering and Computer Science, Northwestern University,
Evanston, IL 60208, USA
e-mail: nocedal@eecs.northwestern.edu

Keywords Large-scale optimization · Constrained optimization · Nonconvex programming · Inexact linear system solvers · Krylov subspace methods

Mathematics Subject Classification (2000) 49M37 · 65K05 · 90C06 · 90C26 · 90C30

1 Introduction

In this paper we discuss an algorithm for optimization problems of the form

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

where $f : \mathbb{R}^n \to \mathbb{R}$ and $c : \mathbb{R}^n \to \mathbb{R}^t$ are smooth and possibly nonconvex functions. Our interest is in methods for very large problems that ensure global convergence to first-order optimal points. We focus primarily on equality constrained problems with $t \leq n$, but note that our techniques can also be applied to unconstrained or general nonlinear programming problems.

Our goal is to develop a *matrix-free* technique for solving nonconvex instances of problem (1.1). By matrix-free, we mean that the algorithm employs methods for computing products with derivatives of the objective and the constraint functions, but does not require the explicit formation or factorization of any matrix. For strictly convex problems, a robust and efficient method of this type is the line search inexact Sequential Quadratic Programming (SQP) method proposed in [2]. In this paper we build on the algorithmic framework presented in [2], but now allow for the presence of negative or zero curvature directions during the step computation. Line search methods for nonconvex optimization often insist on convex quadratic models of the objective and enforce this requirement by modifying the (approximate) Hessian matrix whenever the reduced Hessian is known not to be positive definite. This technique is not entirely practical in our framework, however, as we lack inertia information by not factoring the constraint Jacobians or the primal–dual iteration matrices. Our approach may still require perturbations to a given Hessian matrix, but these modifications are now performed within an iterative solution procedure for the primal–dual equations, where the resulting system may or may not correspond to a convex model.

Overall, the methodology we propose is novel in that it places primary importance on attaining a sufficient reduction in a local approximation of a merit function during each iteration. During an iterative step computation procedure, we monitor these reductions in order to dictate when a given step is acceptable, when a perturbation to the Hessian matrix should be made, and when a change in the merit function itself is appropriate in order to guarantee global convergence. The resulting algorithm fully integrates the procedures of step computation and step selection and automatically reduces to an inexact SQP method in convex regions of the search space.

The organization of this paper is as follows. In Sect. 2 we outline our solution methodology and global convergence strategy. Central to our approach is the notion of a sufficient model reduction that we develop in Sect. 3. Here, we motivate our ideas by combining aspects of the inexact SQP algorithm presented in [2] and further observations from the environment of unconstrained optimization. These beginnings lead us, in Sect. 4, to develop an algorithm for constrained optimization, complete with termination criteria for controlling inexactness in the computation of steps. Global convergence of the approach is presented in Sect. 5 and numerical experiments with a collection of test problems are illustrated in Sect. 6. Final remarks are provided in Sect. 7.

2 An outline of the algorithm

Let us first formalize the basic features of our approach so that we may further remark on its similarities and differences with standard approaches. We begin by defining Lagrange multipliers $\lambda \in \mathbb{R}^t$ and the Lagrangian

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

with which we can state the first-order optimality conditions for problem (1.1):

$$\nabla \mathcal{L}(x, \lambda) = \begin{bmatrix} g(x) + A(x)^T \lambda \\ c(x) \end{bmatrix} = 0. \tag{2.1}$$

Here, we have used the fact that f and c are differentiable and define $g(x)$ as the gradient of the objective and $A(x)$ as the Jacobian of the constraint functions. A Newton iteration for this system of nonlinear equations defines a primal–dual step (d_k, δ_k) from a given iterate (x_k, λ_k) as the solution to the linear system

$$\begin{bmatrix} W_k & A(x_k)^T \\ A(x_k) & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \delta_k \end{bmatrix} = - \begin{bmatrix} g(x_k) + A(x_k)^T \lambda_k \\ c(x_k) \end{bmatrix} \tag{2.2}$$

where

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

is the Hessian of the Lagrangian and $c^i(x)$ and λ^i represent the ith constraint function and its corresponding dual variable, respectively. We call the $(n+t) \times (n+t)$ matrix on the left-hand-side of (2.2) the *primal–dual matrix* and note that we can alternatively set W_k to a symmetric approximation of the Hessian $\nabla^2_{xx} \mathcal{L}(x_k, \lambda_k)$. The primal–dual system (2.2) forms the basis of our step computation.

One useful interpretation of a solution (d_k, δ_k) to the primal–dual equations (2.2) is that in the *strictly convex* case, the primal component d_k is the unique solution to the local optimization subproblem

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

Here, we intend the expression *strictly convex* to mean that W_k is positive definite over the null space of $A(x_k)$ so that there exists no direction of negative or zero curvature $u \in \mathbb{R}^n$; i.e., u such that

$$\begin{aligned} A(x_k)u &= 0 \\ \text{and } u^T W_k u &\leq 0. \end{aligned} \tag{2.4}$$

If such a zero or negative curvature direction were present, as would be the case in what we refer to as the *nonconvex* case, then a solution to problem (2.3) is either not unique or does not exist, and the relationship between (2.2) and (2.3) may break down.

Despite this difficulty, a number of methods have been proposed that provide global convergence guarantees even in the presence of negative or zero curvature directions; see, for example, [5,6,8,9,12,16,22] and the literature surveyed in [3]. These methods, however, in some way require a factorization of the constraint Jacobian. In the class of *full space* methods, the primal–dual matrix must be factored and its inertia must be determined (and possibly altered via a modification to the system, as in [22]) to compute a search direction, or a negative curvature direction must be found. In the classes of *step decomposition* and *reduced space* methods, the Jacobian $A(x_k)$ is factored in order to construct a reduced problem that allows for the use of unconstrained optimization techniques; e.g., see [3].

We would like our approach, however, not to rely on factorizations of the Jacobian $A(x_k)$, the (approximate) Hessian W_k, or the primal–dual matrix—and to permit inexact step computations. This has the benefit of avoiding a significant amount of computational cost, particularly when the problem is large and/or factorizations of $A(x_k)$ and W_k are not readily available. *Inexact reduced space* SQP methods that meet these requirements and yield good results have been proposed by Heinkenschloss and Vicente [14], Ulbrich [21], and Heinkenschloss and Ridzal [13]; see Ridzal [19] for a recent review of this topic.

In this paper we focus on full-space primal–dual SQP methods, which have not received much attention in the nonconvex inexact case. Without a factorization of the primal–dual matrix, or at least of the constraint Jacobian, it is difficult to determine if W_k is positive definite on the null space of $A(x_k)$. That is, without the inertia of the primal–dual matrix, we cannot be sure if a solution to the primal–dual equations (2.2) can be interpreted as a solution to (2.3). In order to bypass these obstacles, we choose to shift the focus away from the relationship between the primal–dual system (2.2) and quadratic programming subproblems similar to (2.3), and instead consider the relationship between the primal–dual equations and the exact penalty function

$$\phi(x;\pi) \triangleq f(x) + \pi\|c(x)\|_2$$

with $\pi > 0$. In other words, we intend to forgo the interpretation of our step as an (approximate) solution to a local optimization subproblem and instead jump straight to its effect on our globalization mechanism. We require that the primal component d_k provide a sufficient reduction in the local approximation

$$m_k(d;\pi) \triangleq l_k(d) + \pi\|r_k(d)\|_2$$

of $\phi(x;\pi)$ about x_k, where

$$l_k(d) \triangleq f(x_k) + g(x_k)^T d$$

is a model of the objective and $r_k(d)$ is the model of the constraints defined in (2.3). The reduction in $m_k(d;\pi)$ yielded by d_k can be computed easily as a combination of the reduction in the objective model

$$\begin{aligned} \Delta l_k(d_k) &\triangleq l_k(0) - l_k(d_k) \\ &= -g(x_k)^T d_k \end{aligned} \tag{2.5}$$

and the reduction in the constraint model

$$\begin{aligned} \Delta\|r_k(d_k)\|_2 &\triangleq \|r_k(0)\|_2 - \|r_k(d_k)\|_2 \\ &= \|c(x_k)\|_2 - \|c(x_k) + A(x_k)d_k\|_2; \end{aligned} \tag{2.6}$$

i.e., we have

$$\begin{aligned} \Delta m_k(d_k;\pi) &\triangleq m_k(0;\pi) - m_k(d_k;\pi) \\ &= \Delta l_k(d_k) + \pi\Delta\|r_k(d_k)\|_2 \\ &= -g(x_k)^T d_k + \pi(\|c(x_k)\|_2 - \|c(x_k) + A(x_k)d_k\|_2). \end{aligned} \tag{2.7}$$

We show that, as long as certain conditions hold with respect to the accuracy of the step computations, global convergence can be guaranteed when $\Delta m_k(d_k;\pi)$ is sufficiently positive during each iteration.

Naturally, a significant reduction in the model $m_k(d;\pi_k)$ may not always be attainable for the given value π_k of the penalty parameter during iteration k. As in other methods that employ an exact penalty function as a merit function, we have the option of increasing π_k to produce a larger value for $\Delta m_k(d;\pi_k)$ whenever a reduction in the constraint model has been attained. However, these increases must be done cautiously and should not always be used to satisfy our model reduction criteria. In some circumstances we illustrate that the step computation should in fact be altered to compute a more appropriate step. For this purpose, in this paper we consider perturbations to W_k in the primal–dual equations (2.2). We describe conditions that suggest when such modifications should be applied during the step computation, and note that the procedure may or may not result in a strictly convex subproblem.

It is worth noting here that, although the model $m_k(d;\pi)$ we consider is composed of linear models of the objective and the constraints, our method can achieve a fast rate of local convergence. This is because our method approximates an SQP approach near a solution. We also wish to distinguish our technique from those in the class of *penalty methods* that attempt to compute a step as some minimizer of $\phi(x;\pi)$ over a local region of the search space. Although we appraise a given solution according to its effect on $\phi(x;\pi)$, our steps are in fact computed via (2.2) and so our approach should not be considered a penalty method per se.

Notation. We drop functional notation when values become clear from the context and use subscripts to delimit iteration numbers; i.e., we denote $g_k \triangleq g(x_k)$ and similarly for other quantities. All norms (including those written above) are considered Euclidean (or l_2) norms unless otherwise indicated, though much of the analysis applies for any vector-based norm. The expression $M_1 \succ M_2$ for square symmetric matrices M_1 and M_2 conveys that $M_1 - M_2$ is positive definite.

3 Sufficient model reductions

The main task in the development of our algorithm is to construct a condition that quantifies when the reduction obtained in a local approximation of the merit function is sufficiently large for a given step. We begin by considering the unconstrained case where steps are computed via an exact solution of the Newton equations, as in this setting we can begin to motivate certain algorithmic choices made in more complicated frameworks.

For an unconstrained problem, the Newton equations (2.2) reduce to

$$W_k d_k = -g_k, \tag{3.1}$$

$W_k \approx \nabla^2_{xx} f_k$ is now a symmetric approximation of the Hessian of the objective at x_k, $\phi(x;\pi)$ reduces to $f(x)$, and the reduction in m_k obtained by d_k is simply the reduction in the objective model $\Delta l_k(d_k)$. If W_k is positive definite with its smallest eigenvalue bounded away from zero for all k, then d_k should be considered an acceptable step for the algorithm to follow. This can be argued easily in that, in this case, the directional derivative of f along d_k satisfies [see (3.1)]

$$g_k^T d_k = -d_k^T W_k d_k \leq -\theta \|d_k\|^2 \tag{3.2}$$

for some $\theta > 0$. Thus, d_k is a descent direction.

In terms of reducing the model $l_k(d)$, it is obvious from (2.5) and (3.2) that

$$\Delta l_k(d_k) = -g_k^T d_k \geq \theta \|d_k\|^2.$$

In fact, we claim that this level of reduction is acceptable in more than just the strictly convex case. For instance, even if W_k is not positive definite, we may still have $d_k^T W_k d_k \geq \theta \|d_k\|^2$ for some constant $\theta > 0$ when d_k is given by (3.1), in which case we again have $\Delta l_k(d_k) \geq \theta \|d_k\|^2$ with d_k being a direction of descent.

More generally, in the inexact case [i.e., when a potential step is computed as merely an approximate solution of (3.1)] we can use the above observations to form a guideline for the types of steps that should be considered acceptable. If d_k is computed with $d_k^T W_k d_k$ sufficiently positive, then we can emulate the positive definite case above by saying that d_k is acceptable if

$$\Delta l_k(d_k) \geq \tfrac{1}{2} d_k^T W_k d_k. \tag{3.3}$$

In any case, however, we can always consider d_k acceptable as long as

$$\Delta l_k(d_k) \geq \theta \|d_k\|^2$$

for some $\theta > 0$. Overall, we can both emulate the strictly convex case (and thus attempt for fast local convergence) and provide for sufficient descent in the nonconvex case by defining d_k to be acceptable if and only if

$$\Delta l_k(d_k) \geq \max\left\{\tfrac{1}{2} d_k^T W_k d_k, \theta \|d_k\|^2\right\}. \tag{3.4}$$

Here, the right-hand-side expression is positive for all nonzero d_k and 2θ can be viewed as the smallest Rayleigh quotient for (W_k, d_k) such that the curvature along d_k is considered sufficiently positive.

Now we turn to a constrained environment, and again begin by considering a situation where W_k is positive definite for all k. As before, we know that in this case the step provided by (2.2) may be considered acceptable and so we can use the magnitudes of the resulting objective and constraint reductions to develop a general guideline for appropriate reductions in the full model m_k. The reduction attained in the linear model of the constraints [see (2.6)] is in this case

$$\Delta \|r_k(d_k)\| = \|c_k\| - \|c_k + A_k d_k\| = \|c_k\|. \tag{3.5}$$

Similarly, the reduction in the linear model of the objective becomes

$$\begin{aligned} \Delta l_k(d_k) &= -g_k^T d_k \\ &= d_k^T W_k d_k + d_k^T A_k^T (\lambda_k + \delta_k) \\ &= d_k^T W_k d_k - c_k^T (\lambda_k + \delta_k) \\ &\geq d_k^T W_k d_k - \|c_k\| \|\lambda_k + \delta_k\|, \end{aligned} \tag{3.6}$$

which may be positive or negative depending on the relationship between the objective and constraints in problem (2.3). Thus, after accounting for the fact that there may be conflicts between our dual goals of minimizing the objective and satisfying the constraints, we can conclude from (3.5) and (3.6) that with an appropriate π_k a sufficient model reduction is one satisfying

$$\Delta m_k(d_k; \pi_k) = \Delta l_k(d_k) + \pi_k \Delta \|r_k(d_k)\| \geq \tfrac{1}{2} d_k^T W_k d_k + \sigma \pi_k \|c_k\| \tag{3.7}$$

for some $\sigma \in (0, 1)$. [Notice the parallel between (3.7) and (3.3).]

With only (3.7), however, we can see a difficulty that arises in the face of nonconvexity. Let us consider the decomposition

$$d_k = u_k + v_k, \tag{3.8}$$

where u_k lies in the null space of A_k and v_k lies in the range space of A_k^T. If A_k has full row rank and $\|c_k\| = 0$, then by (2.2) we have $v_k = 0$ and so (3.7) only requires

$$\Delta m_k(d_k; \pi_k) = \Delta m_k(u_k; \pi_k) \geq \tfrac{1}{2} u_k^T W_k u_k.$$

This can be a problem if u_k is a direction of negative or zero curvature [see (2.4)], as in this case (3.7) would allow for an increase in the model $m_k(d; \pi_k)$.

Thus, we can borrow from our experience in the unconstrained case above to form a condition that can provide for both strictly convex and nonconvex environments. We propose that a step (d_k, δ_k) obtained via (2.2) be regarded as acceptable if

$$\Delta m_k(d_k; \pi_k) \geq \max\left\{\tfrac{1}{2} d_k^T W_k d_k, \theta \|u_k\|^2\right\} + \sigma \pi_k \|c_k\| \tag{3.9}$$

for some $\sigma \in (0, 1)$, $\theta > 0$, and an appropriate $\pi_k > 0$. The max in (3.9) can be motivated by (3.4), where in this case the tangential component u_k plays a role similar to the unconstrained step.

Two alterations must be made to transform (3.9) into a practical condition for our framework. First, the tangential component u_k may not be available explicitly [as is indeed the case for steps computed via (2.2)], and so we are not able to compute the norm of this tangential component directly. This measure can be approximated by $\|d_k\|^2$, but in general we desire Υ_k satisfying

$$\|u_k\|^2 \leq \Upsilon_k \leq \|d_k\|^2 \tag{3.10}$$

that is as close to $\|u_k\|^2$ as possible so that (3.9) is not overly restrictive. For the implementation of our algorithm described in Section 6, we provide in equation (6.7) a practical method for computing a value for Υ_k that exploits some properties of the primal–dual system (2.2).

A second alteration to (3.9) is necessary for the algorithm to allow for inexact step computations. Here, we are considering situations such as when an iterative linear system solver is employed to solve the primal–dual equations (2.2), where during each *inner iteration* we are provided with a solution to the perturbed equations

$$\begin{bmatrix} W_k & A_k^T \\ A_k & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \delta_k \end{bmatrix} = -\begin{bmatrix} g_k + A_k^T \lambda_k \\ c_k \end{bmatrix} + \begin{bmatrix} \rho_k \\ r_k \end{bmatrix} \tag{3.11}$$

where (ρ_k, r_k) is a residual vector. [Note that denoting the constraint residual as r_k is consistent with our definition of $r_k(d_k)$ in (2.3) above.] In this setting, the quantity Δm_k can vary considerably and careful attention must be paid to ensure that the model reductions obtained are sufficiently large with respect to appropriate quantities.

The following condition can be seen as the marriage between (3.9) and the notion of a sufficient model reduction condition for an inexact algorithm provided in [2], and is the centerpiece of the algorithm described in detail in the following section.

Model reduction condition. Let $\theta > 0$ be a given constant. An inexact solution (d_k, δ_k) to (2.2) is an acceptable step only if

$$\Delta m_k(d_k; \pi_k) \geq \max\left\{\tfrac{1}{2} d_k^T W_k d_k, \theta \Upsilon_k\right\} + \sigma \pi_k \max\{\|c_k\|, \|r_k\| - \|c_k\|\} \tag{3.12}$$

for some $0 < \sigma < 1$ and appropriate π_k, where the residual r_k is defined in (3.11).

Finally, let us immediately point out an important result that is crucial to our analysis in Sect. 5. If we denote $D\phi(d_k; \pi_k)$ as the directional derivative of the penalty function $\phi(x; \pi_k)$ at x_k along d_k, it is known that

$$\begin{aligned} D\phi(d_k; \pi_k) &\leq g_k^T d_k - \pi_k(\|c_k\| - \|r_k\|) \\ &= -\Delta l_k(d_k) - \pi_k \Delta \|r_k(d_k)\| \\ &= -\Delta m_k(d_k; \pi_k) \end{aligned} \tag{3.13}$$

(e.g., see [2]). Thus, a step satisfying the Model Reduction Condition is a descent direction for the penalty function $\phi(x; \pi_k)$ at x_k.

4 An inexact newton method based on model reductions

In this section we develop a complete algorithm for nonconvex optimization based on sufficient reductions in local approximations of a merit function.

We first briefly outline our methodology in the case where steps are computed exactly. First, we compute a trial step (d_k, δ_k) satisfying the primal–dual equations (2.2). If the Model Reduction Condition is satisfied for the most recent value of the penalty parameter π, call it π_{k-1}, then we accept the step, set $\pi_k \leftarrow \pi_{k-1}$, and continue the iteration. Otherwise, since a reduction in the model of the constraints has been attained [see (3.5)], we have the option of increasing π in order to satisfy the Model Reduction Condition. This, however, should only be done in two circumstances. On the one hand, if $\frac{1}{2} d_k^T W_k d_k \geq \theta \Upsilon_k$, then we find it safe to assume that the problem is sufficiently convex and so (d_k, δ_k) should be considered acceptable and π should be increased. On the other hand, if $\frac{1}{2} d_k^T W_k d_k < \theta \Upsilon_k$, then we only consider increasing π if v_k is a significant component of d_k (i.e., if the step is sufficiently *normal*) as in this case we can be sure that the objective value does not suffer *too much* by us placing a priority on satisfying the constraints. We express this condition as $\Upsilon_k \leq \psi \nu_k$ for some $\psi > 0$ where

$$0 \leq \nu_k \leq \|v_k\|^2 \tag{4.1}$$

is any lower bound for the squared norm of the normal step component. [A practical expression for ν_k used in our implementation is provided as equation (6.6).] If none of the above avenues can be taken to ensure that the Model Reduction Condition is satisfied for some appropriate $\pi_k \geq \pi_{k-1}$, then we have no choice but to consider a modification to our step computation via a perturbation to W_k. Once such a modification is made, we compute a new trial step via the perturbed primal–dual equations and repeat the steps above until an acceptable step is obtained.

In the inexact case we emulate these processes, but with added restrictions for the residuals in (3.11). We define the following as *termination tests* in reference to implementations where an iterative solver is being applied to the primal–dual equations to produce a sequence of values $\{(d^j, \delta^j)\}$ corresponding to some sequence of residuals $\{(\rho^j, r^j)\}$ via (3.11). Both tests below can be seen as extensions to the Sufficient Merit function Approximation Reduction Termination Tests (or *SMART Tests* for short) that are central to the algorithm in [2].

The first test relates to those steps providing a sufficient model reduction with the most recent value of the penalty parameter.

Termination Test I. Let $0 < \sigma, \kappa < 1$ be given constants and let θ be given by the Model Reduction Condition. A step (d_k, δ_k) is acceptable if the Model Reduction Condition (3.12) holds for $\pi_k = \pi_{k-1}$ and

$$\left\| \begin{bmatrix} \rho_k \\ r_k \end{bmatrix} \right\| \leq \kappa \left\| \begin{bmatrix} g_k + A_k^T \lambda_k \\ c_k \end{bmatrix} \right\| \tag{4.2}$$

for (ρ_k, r_k) defined by (3.11).

Steps that satisfy Termination Test I are those that satisfy the Model Reduction Condition directly, without requiring a change in the penalty parameter or any (further) modifications to the step computation. (In Sect. 7 we describe an alternative version of Termination Test I.)

The second termination test relates to those steps yielding a sufficiently large reduction in the model of the constraints. For steps of this type, the penalty parameter may be increased, but, as in the exact case, we are mindful of the relationship between the norms of the tangential and normal components. Consideration of this test during iteration k naturally requires that $\|c_k\| > 0$.

Termination Test II. Let $0 < \epsilon < 1$ and $\beta, \psi > 0$ be given constants and let θ be given by the Model Reduction Condition. A step (d_k, δ_k) is acceptable if

$$\|r_k\| \leq \epsilon \|c_k\|, \tag{4.3a}$$

$$\|\rho_k\| \leq \beta \|c_k\|, \tag{4.3b}$$

and

$$\tfrac{1}{2} d_k^T W_k d_k \geq \theta \Upsilon_k \quad \text{or} \quad \psi \nu_k \geq \Upsilon_k, \tag{4.4}$$

for Υ_k, (ρ_k, r_k), and ν_k defined by (3.10), (3.11), and (4.1), respectively.

Steps satisfying Termination Test II may not satisfy the Model Reduction Condition for $\pi_k = \pi_{k-1}$; thus, for steps of this type we update the penalty parameter by requiring

$$\pi_k \geq \frac{g_k^T d_k + \max\left\{\frac{1}{2} d_k^T W_k d_k, \theta \Upsilon_k\right\}}{(1-\tau)(\|c_k\| - \|r_k\|)} \triangleq \pi_k^{\text{trial}} \tag{4.5}$$

for a given $0 < \tau < 1$. By forcing this inequality, we can see from (2.7) and (4.3a) that

$$\begin{aligned}\Delta m_k(d_k;\pi_k) \geq&\ \max\left\{\tfrac{1}{2}d_k^T W_k d_k, \theta\Upsilon_k\right\} + \tau\pi_k(\|c_k\| - \|r_k\|)\\ \geq&\ \max\left\{\tfrac{1}{2}d_k^T W_k d_k, \theta\Upsilon_k\right\} + \tau(1-\epsilon)\pi_k\|c_k\|,\end{aligned} \tag{4.6}$$

and so the step satisfies the Model Reduction Condition (3.12) for $\sigma = \tau(1-\epsilon)$. Indeed, for consistency between Termination Tests I and II, let us assume that we choose $\sigma = \tau(1-\epsilon)$.

Finally, we are confronted with the issue of how to modify our step computation if it appears that a sufficient model reduction will not be obtained by solving a given primal–dual system (2.2). One technique is to modify W_k to increase some or all of its eigenvalues so that the resulting matrix is *closer* to being positive definite. For example, if the objective function f includes a regularization parameter, this parameter could be increased. Alternatively, W_k could be replaced by a positive definite quasi-Newton approximation, or a positive definite matrix could be added to W_k so that the resulting matrix, call it $\widetilde{W}_k$, satisfies $\widetilde{W}_k - W_k \succ \mu I$ for some $\mu > 0$.

For the sake of concreteness, let us simply assume that the chosen method for modifying W_k is such that after a finite number of modifications we have W_k sufficiently positive definite. Then, regardless of the specific method chosen, the following rule can be used in conjunction with Termination Tests I and II to decide when a modification may be necessary.

Hessian Modification Strategy. Let W_k be the current Hessian approximation, let θ, σ, and ψ be given by the Model Reduction Condition and Termination Tests I and II, and let a trial step (d_k, δ_k) be given. If (d_k, δ_k) yields

$$\Delta m(d_k;\pi_{k-1}) < \max\left\{\tfrac{1}{2}d_k^T W_k d_k, \theta\Upsilon_k\right\} + \sigma\pi_{k-1}\max\{\|c_k\|, \|r_k\| - \|c_k\|\} \tag{4.7a}$$

$$\tfrac{1}{2}d_k^T W_k d_k < \theta\Upsilon_k, \tag{4.7b}$$

$$\text{and}\quad \psi\nu_k < \Upsilon_k, \tag{4.7c}$$

then modify W_k; otherwise, maintain the current W_k. (It is assumed that after a finite number of modifications we have $W_k \succ 2\theta I$.)

This update rule can be motivated by observing a case where the primal–dual equations (2.2) are solved exactly, as in this setting the rule states that W_k should be modified if and only if the computed step does not satisfy Termination Test I or II. This is clear as in this case we have $(\rho_k, r_k) = 0$, and so (4.2) and (4.3) are satisfied. Moreover, we note that under the conditions of this strategy only a finite number of modifications will be performed during a given iteration k, as $W_k \succ 2\theta I$ would imply that (4.7b) cannot hold. (In Sect. 7 we provide an extension to this strategy that may be useful along with certain types of modifications.)

Finally, upon the computation of an acceptable step, we use a backtracking line search on the merit function $\phi(x;\pi_k)$ to promote convergence, where the steplength

α_k must satisfy the Armijo condition

$$\phi(x_k + \alpha_k d_k; \pi_k) \leq \phi(x_k; \pi_k) + \eta\alpha_k D\phi(d_k; \pi_k) \tag{4.8}$$

for some $\eta \in (0, 1)$.

The complete algorithm can be summarized as follows.

Algorithm INS*: Inexact Newton with SMART Tests.*

```
Choose parameters $0 < \kappa, \epsilon, \tau, \eta < 1$ and $\theta, \beta, \psi > 0$ and set $\sigma \leftarrow \tau(1-\epsilon)$
Initialize $x_0$, $\lambda_0$, and $\pi_{-1} > 0$
for $k = 0, 1, 2, \ldots,$ until a convergence test for (1.1) is satisfied
    Compute $f_k, g_k, c_k, A_k$, and $W_k$ and initialize $\pi_k \leftarrow \pi_{k-1}$
    repeat
        Compute an approximate solution $(d_k, \delta_k)$ to (2.2) with the current $W_k$
        Compute $\Upsilon_k$ and $\nu_k$ satisfying (3.10) and (4.1) (see (6.6) and (6.7))
        if Termination Test I or II is satisfied, then break
        Run the Hessian Modification Strategy to update $W_k$
    endrepeat
    if Termination Test II is satisfied and (4.5) does not hold, set $\pi_k \leftarrow \pi_k^{trial} + 10^{-4}$
    Backtrack from $\alpha_k \leftarrow 1$ to obtain $\alpha_k$ satisfying (4.8)
    Set $(x_{k+1}, \lambda_{k+1}) \leftarrow (x_k, \lambda_k) + \alpha_k(d_k, \delta_k)$
endfor
```

If an iterative linear system solver is employed in the step computation of Algorithm INS, then an efficient implementation will run the solver unadulterated as long as W_k is not modified by the Hessian Modification Strategy. Once a modification is invoked, however, the solver may need to be reinitialized, where it may be beneficial to use the most recent trial step as an initial estimate of the solution of the newly perturbed system. Further practical issues for implementing Algorithm INS are discussed in Sect. 6.

5 Global analysis

In this section we analyze the global convergence properties of Algorithm INS under the following assumptions about the problem formulation and the set of computed iterates.

Assumption 1 The sequence $\{x_k, \lambda_k\}$ generated by Algorithm INS 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) For all k, the system (2.2) is consistent for each W_k.

These assumptions are similar to those made in [2], except that we do not assume that the value for W_k used to compute the step during iteration k is positive definite on the null space of the constraint Jacobian A_k. Assumption 1(e) is easily satisfiable during a run of Algorithm INS if, for example, the iterative solver for the primal–dual system can detect rank deficiency or breaks down when given an inconsistent system. If such a system is detected, then perturbations to W_k can remedy this and produce a set of consistent linear equations.

Before starting our analysis, it is important to verify that each iteration of Algorithm INS will be successful in that a step satisfying Termination Test I or II will be computed. To do this, suppose that the current iterate (x_k, λ_k) does not satisfy the optimality conditions (2.1). Under Assumption 1 we argue that, since the primal–dual systems (2.2) are consistent and as long as the iterative step computation converges to a solution with (ρ_k, r_k) arbitrarily small, a W_k will eventually be chosen such that a sufficiently accurate solution to (2.2) will satisfy Termination Test I or II. Let us consider two cases. First, if $c_k \neq 0$, then (4.2) and (4.3) will be satisfied for (ρ_k, r_k) sufficiently small. Consequently, the Model Reduction Condition will be satisfied for $\pi_k = \pi_{k-1}$ (in which case Termination Test I will be satisfied), (4.4) will hold (in which case Termination Test II will be satisfied), or the Hessian Modification Strategy will invoke a modification to W_k. Similarly, if $c_k = 0$, then a sufficiently accurate solution will satisfy (4.2) and either the Model Reduction Condition will be satisfied for $\pi_k = \pi_{k-1}$ (and so Termination Test I will be satisfied), or the lack of a positive definite reduced Hessian will trigger a modification via (4.7b). In any case, since after a finite number of modifications we have that $W_k \succ 2\theta I$, a sufficiently accurate solution to (2.2) will satisfy Termination Test I or II.

We are now ready to establish global convergence properties for Algorithm INS. From now on, we consider W_k to be the last value for the matrix used to compute the accepted step; i.e., W_k is the Hessian approximation once all of the modifications via the Hessian Modification Strategy have been made.

We begin by proving a result on the properties of the normal step v_k. It is important to note that v_k and the tangential component u_k may not be computed explicitly; we study them only for purposes of this analysis. Although the proof of this first result is essentially the same as in [2], we include it here for completeness.

Lemma 1 *For all k, the normal component v_k is bounded in norm and for some $\gamma_1 > 0$ satisfies*

$$\|v_k\|^2 \leq \gamma_1 \max\{\|c_k\|, \|r_k\|\}. \tag{5.1}$$

Furthermore, for all k such that Termination Test II is satisfied, there exists $\gamma_2 > 0$ such that

$$\|v_k\| \leq \gamma_2(\|c_k\| - \|r_k\|). \tag{5.2}$$

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

$$\|v_k\| \leq \|A_k^T (A_k A_k^T)^{-1}\|(\|c_k\| + \|r_k\|). \tag{5.3}$$

This, along with (4.2), (4.3), the fact that Assumption 1(a) and (b) imply that $\|c_k\|$ and $\|g_k + A_k^T\lambda_k\|$ are bounded, and the fact that Assumption 1(a) and (c) imply that $\|A_k^T(A_kA_k^T)^{-1}\|$ is bounded, implies v_k is bounded in norm for all k. The inequality (5.3) also yields

$$\begin{aligned}\|v_k\|^2 &\leq \left(\|A_k^T(A_kA_k^T)^{-1}\|(\|c_k\| + \|r_k\|)\right)^2\\ &\leq \left(2\|A_k^T(A_kA_k^T)^{-1}\| \max\{\|c_k\|, \|r_k\|\}\right)^2\\ &= \left[4\|A_k^T(A_kA_k^T)^{-1}\|^2 \max\{\|c_k\|, \|r_k\|\}\right] \max\{\|c_k\|, \|r_k\|\},\end{aligned} \tag{5.4}$$

where (4.2), (4.3), and Assumption 1(a), (b), and (c) also imply that the bracketed expression in (5.4) is bounded. Thus, (5.1) holds. Finally, if Termination Test II is satisfied, then from (4.3a) and (5.3) 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 (5.2) holds. □

The next result provides some important bounds related to the length of the primal component d_k.

Lemma 2 *There exists $\gamma_3 > 0$ such that, for all k,*

$$\|d_k\|^2 \leq \gamma_3\left(\Upsilon_k + \max\{\|c_k\|, \|r_k\|\}\right), \tag{5.5}$$

and hence

$$\|d_k\|^2 + \|c_k\| \leq 2\gamma_3\left(\Upsilon_k + \max\{\|c_k\|, \|r_k\|\}\right). \tag{5.6}$$

Proof Since u_k and v_k are defined in (3.8) to be orthogonal we have $\|d_k\|^2 = \|u_k\|^2 + \|v_k\|^2$. Inequality (5.5) then follows from Lemma 1 and the fact that Υ_k is an upper bound for $\|u_k\|^2$. The inequality (5.6) follows trivially from (5.5). □

The directional derivative of the penalty function can be bounded similarly.

Lemma 3 *There exists $\gamma_4 > 0$ such that, for all k,*

$$D\phi(d_k; \pi_k) \leq -\gamma_4\left(\Upsilon_k + \max\{\|c_k\|, \|r_k\|\}\right).$$

Proof We find by the Model Reduction Condition (3.12), (4.6), and (3.13) that

$$\begin{aligned}D\phi(d_k; \pi_k) &\leq -\max\left\{\tfrac{1}{2}d_k^T W_k d_k, \theta\Upsilon_k\right\} - \sigma\pi_k \max\{\|c_k\|, \|r_k\| - \|c_k\|\}\\ &\leq -\theta\Upsilon_k - \tfrac{\sigma\pi_k}{2}\max\{\|c_k\|, \|r_k\|\}.\end{aligned}$$

Thus, since $\pi_k \geq \pi_{-1}$ for all k, the result holds for $\gamma_4 = \min\{\theta, \frac{\sigma\pi_{-1}}{2}\} > 0$. □

An important feature of Algorithm INS is that the penalty parameter remains bounded. The next result illustrates the importance of the bounds enforced in (4.3) and (4.4).

Lemma 4 *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 The penalty parameter is increased during iteration k of Algorithm INS only if Termination Test II is satisfied. Thus, we can assume that the inequalities (4.3) hold and that π_k satisfies (4.6). This can be rewritten as

$$\Delta m_k(d_k;\pi_k) - \max\left\{\tfrac{1}{2}d_k^T W_k d_k, \theta\Upsilon_k\right\} \geq \tau\pi_k(\|c_k\| - \|r_k\|), \tag{5.7}$$

where we can use (2.7) to expand

$$\begin{aligned}\Delta m_k(d_k;\pi_k) &- \max\left\{\tfrac{1}{2}d_k^T W_k d_k, \theta\Upsilon_k\right\}\\ &= \pi_k(\|c_k\| - \|r_k\|) + \begin{cases}\left[-g_k^T d_k - \tfrac{1}{2}d_k^T W_k d_k\right] & \text{if} \quad \tfrac{1}{2}d_k^T W_k d_k \geq \theta\Upsilon_k\\ \left[-g_k^T d_k - \theta\Upsilon_k\right] & \text{otherwise.}\end{cases}\end{aligned} \tag{5.8}$$

We establish bounds for the terms inside the brackets with respect to the reduction obtained in the linear model of the constraints.

If $\frac{1}{2}d_k^T W_k d_k < \theta\Upsilon_k$, then (3.10), (4.1), and (4.4) yield $\|u_k\|^2 \leq \Upsilon_k \leq \psi\nu_k \leq \psi\|v_k\|^2$. This implies $\|d_k\| \leq \sqrt{1+\psi}\|v_k\|$, and so we have by Lemma 1 that $\|d_k\|$ is bounded. Thus, by Assumption 1(a), Lemma 1, and (3.10), there exist $\gamma_5, \gamma_5' > 0$ such that

$$\begin{aligned}-g_k^T d_k - \theta\Upsilon_k &\geq -\|g_k\|\|d_k\| - \theta\|d_k\|^2\\ &\geq -\gamma_5\|d_k\|\\ &\geq -\gamma_5(\sqrt{1+\psi})\|v_k\|\\ &\geq -\gamma_5'(\|c_k\| - \|r_k\|).\end{aligned}$$

Now suppose $\frac{1}{2}d_k^T W_k d_k \geq \theta\Upsilon_k$ and note that from Assumption 1(d) there exists some constant $0 < \gamma_6 < \infty$ such that, for all k,

$$\frac{\|W_k\|}{\sqrt{\gamma_6}} + \frac{\|W_k\|}{2\gamma_6} \leq \frac{\theta}{2}. \tag{5.9}$$

If $\|u_k\|^2 < \gamma_6\|v_k\|^2$, then as above, Assumption 1(a) and Lemma 1 imply that $\|d_k\|$ is bounded and for some $\gamma_7, \gamma_7' > 0$ we have

$$\begin{aligned}-g_k^T d_k - \tfrac{1}{2}d_k^T W_k d_k &\geq -\gamma_7\|d_k\|\\ &\geq -\gamma_7\sqrt{(1+\gamma_6)}\|v_k\|\\ &\geq -\gamma_7'(\|c_k\| - \|r_k\|).\end{aligned}$$

Otherwise, $\|u_k\|^2 \geq \gamma_6 \|v_k\|^2$ and so the primal step is composed of a proportionately large tangential component. By our consideration of $\frac{1}{2}d_k^T W_k d_k \geq \theta \Upsilon_k$, (3.10), and (5.9) we have

$$\begin{aligned}\tfrac{1}{2}u_k^T W_k u_k &\geq \theta \Upsilon_k - u_k^T W_k v_k - \tfrac{1}{2}v_k^T W_k v_k \\ &\geq \left(\theta - \tfrac{\|W_k\|}{\sqrt{\gamma_6}} - \tfrac{\|W_k\|}{2\gamma_6}\right)\|u_k\|^2 \\ &\geq \tfrac{\theta}{2}\|u_k\|^2.\end{aligned}$$

From this inequality, the first block equation of (3.11), and the fact that $A_k u_k = 0$ we have

$$\begin{aligned}\theta\|u_k\|^2 &\leq u_k^T W_k u_k \\ &= -g_k^T u_k + \rho_k^T u_k - u_k^T W_k v_k \\ &\leq (\|g_k\| + \|\rho_k\| + \|W_k v_k\|)\|u_k\|,\end{aligned}$$

which implies

$$\|u_k\| \leq (\|g_k\| + \|\rho_k\| + \|W_k v_k\|)/\theta,$$

and so we have by Lemma 1 that $\|u_k\|$ is bounded under Assumption 1 and the bounds in (4.3). Thus, from the above, Assumption 1(a), Lemma 1, (4.3b), and the first block equation of (3.11), we find that for some $\gamma_8, \gamma_8' > 0$ we have

$$\begin{aligned}-g_k^T d_k - \tfrac{1}{2}d_k^T W_k d_k &= -g_k^T v_k - \tfrac{1}{2}v_k^T W_k v_k - u_k^T W_k v_k - g_k^T u_k - \tfrac{1}{2}u_k^T W_k u_k \\ &= -g_k^T v_k - \tfrac{1}{2}v_k^T W_k v_k - \rho_k^T u_k + \tfrac{1}{2}u_k^T W_k u_k \\ &\geq -\gamma_8(\|v_k\| + \|\rho_k\|) \\ &\geq -\gamma_8'(\|c_k\| - \|r_k\|).\end{aligned}$$

All together, starting from (5.8) we have shown

$$\Delta m_k(d_k;\pi_k) - \max\left\{\tfrac{1}{2}d_k^T W_k d_k, \theta\Upsilon_k\right\} \geq (\pi_k - \max\{\gamma_5', \gamma_7', \gamma_8'\})(\|c_k\| - \|r_k\|),$$

and so (5.7) is always satisfied for

$$\pi_k \geq \max\{\gamma_5', \gamma_7', \gamma_8'\}/(1-\tau). \tag{5.10}$$

Thus, if $\pi_{\bar{k}}$ satisfies (5.10) for some $\bar{k} \geq 0$, then $\pi_k = \pi_{\bar{k}}$ for all $k \geq \bar{k}$. This, along with the fact that when Algorithm INS increases π it does so by at least a positive finite amount, proves the result. □

We have shown that at the conclusion of the step computation, a bounded penalty parameter has been computed and that the directional derivative of the resulting penalty function is negative. We can now show that the line search will be successful and result in a sufficient reduction in the merit function.

Lemma 5 *The sequence $\{\alpha_k\}$ is bounded below and away from zero.*

Proof Recall that the line search requires (4.8). Suppose that the line search fails for some $\bar{\alpha} > 0$, so

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

A Taylor expansion of $\phi(x; \pi_k)$ about x_k yields for some $\gamma_9 > 0$

$$\phi(x_k + \bar{\alpha} d_k; \pi_k) - \phi(x_k; \pi_k) \leq \bar{\alpha} D\phi(d_k; \pi_k) + \bar{\alpha}^2 \gamma_9 \pi_k \|d_k\|^2,$$

so

$$(\eta - 1) D\phi(d_k; \pi_k) < \bar{\alpha} \gamma_9 \hat{\pi} \|d_k\|^2.$$

Here, $\hat{\pi}$ is a finite upper bound for the sequence $\{\pi_k\}$ whose existence follows from Lemma 4. Lemmas 2 and 3 then yield

$$(1 - \eta)\gamma_4 \left(\Upsilon_k + \max\{\|c_k\|, \|r_k\|\}\right) < \bar{\alpha} \gamma_3 \gamma_9 \hat{\pi} \left(\Upsilon_k + \max\{\|c_k\|, \|r_k\|\}\right),$$

so

$$\bar{\alpha} > (1 - \eta)\gamma_4 / (\gamma_3 \gamma_9 \hat{\pi}).$$

Thus, α_k need never be set below $(1 - \eta)\gamma_4/(\gamma_3\gamma_9\hat{\pi})$ for (4.8) to be satisfied. □

The next result illustrates the convergence of the iterates of Algorithm `INS` toward the feasible region and that the sequence of primal step components vanishes.

Lemma 6 *Algorithm* `INS` *yields*

$$\lim_{k \to \infty} \|c_k\| = 0 \;\; \textit{and} \;\; \lim_{k \to \infty} \|d_k\| = 0. \tag{5.11}$$

Proof By Lemma 4 the algorithm eventually computes, during some iteration $\bar{k} \geq 0$, a finite value $\pi_{\bar{k}}$ beyond which the penalty parameter will never be increased. Thus, (4.8), Lemma 3, Lemma 5, and (5.6) imply that there exists $\gamma_{10} > 0$ such that for $k \geq \bar{k}$ we have

$$\begin{aligned}
\phi(x_{\bar{k}}; \pi_{\bar{k}}) - \phi(x_k; \pi_{\bar{k}}) &= \sum_{j=\bar{k}}^{k-1} \left(\phi(x_j; \pi_{\bar{k}}) - \phi(x_{j+1}; \pi_{\bar{k}})\right) \\
&\geq -\gamma_{10} \sum_{j=\bar{k}}^{k-1} D\phi(d_j; \pi_{\bar{k}}) \\
&\geq \gamma_4 \gamma_{10} \sum_{j=\bar{k}}^{k-1} \left(\Upsilon_k + \max\{\|c_k\|, \|r_k\|\}\right) \\
&\geq \tfrac{\gamma_4 \gamma_{10}}{2\gamma_3} \sum_{j=\bar{k}}^{k-1} \Big(\|d_k\|^2 + \|c_k\|\Big).
\end{aligned}$$

The result follows from this inequality and the fact that Assumption 1(a) implies $\phi(x; \pi_{\bar{k}})$ is bounded below. □

We note that the limits (5.11) are precisely those established in Lemma 4.9 of [2]. Therefore, under Assumption 1 we can apply the same analysis as in Theorem 4.10 in [2] to obtain the following result.

Theorem 5.1 *Algorithm* INS *yields*

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

Thus, we have shown under Assumption 1 that the optimality conditions (2.1) are satisfied in the limit, without requiring any convexity assumptions related to the arising primal–dual systems.

6 An implementation

A Matlab implementation of Algorithm INS was developed to illustrate the robustness of the approach and examine its practical nature. We begin by discussing a few implementational issues in chronological order of the steps of the algorithm and then describe its performance on a varied set of test problems.

A general guideline for the selection of the parameters σ, ϵ, τ, β, and κ is provided in [2]. In particular, the inexactness parameters ϵ, β, and κ can have a significant impact on the speed with which the algorithm converges and should preferably be tuned for each application; e.g., see the numerical study [4]. As for the remaining inputs to Algorithm INS, we make the following choices. First, we initialize W_k to the exact Hessian of the Lagrangian for all k. Then, as an appropriate value for θ we suggest setting

$$\theta \leftarrow 10^{-8} \max\{\|W_k\|_1, 1\}$$

during each iteration, as our step acceptance conditions are then independent of scalings of f and c. For ψ, an appropriate value may be any value between zero and one, as this would imply that we consider a step to be sufficiently *normal* if $\|u_k\|^2$ is at most equal to $\|v_k\|^2$. However, as we only use approximate values for these quantities, we have implemented $\psi \leftarrow 10$. Finally, we set η to a standard value of 10^{-8} and initialize the penalty parameter to 10^{-1}. A complete listing of the parameters used in our implementation is given in Table 1.

For the stopping condition of the outer loop of Algorithm INS, we require

$$\|g_k + A_k^T \lambda_k\|_\infty \leq 10^{-6} \max\{\|g_0\|_\infty, 1\} \tag{6.1}$$

$$\text{and} \quad \|c_k\|_\infty \leq 10^{-6} \max\{\|c_0\|_\infty, 1\} \tag{6.2}$$

for the final solution to be considered first-order optimal, but terminate the algorithm if these inequalities are not satisfied after 1,000 iterations.

Table 1 Parameter values used for Algorithm INS

Parameter	Value	Parameter	Value
κ	10^{-2}	θ	$10^{-8}\max\{\|W_k\|_1, 1\}$
σ	$\tau(1-\epsilon)$	β	10
ϵ	10^{-2}	ψ	10
τ	2×10^{-1}	π_{-1}	10^{-1}
η	10^{-8}		

Our implementation of the step computation of Algorithm `INS` is summarized in Algorithm `INS-step` below. The iterative solution of system (2.2) was performed with an adapted version of Kelley's implementation [15] of the generalized minimum residual (GMRES) method [20]. [We also performed all of the experiments below with adapted versions of Paige and Saunders' implementation [18] of their minimum residual (MINRES) method [17] and with an implementation of the symmetric quasi-minimum residual (SQMR) method [10] provided by Eldad Haber. All of the results were very similar, and so we only discuss the results for GMRES below.] GMRES does not exploit the symmetry of the primal–dual system, but the numerical stability of the approach was ideal for our tests. Since we are only interested in demonstrating the robustness of our approach, we did not implement a preconditioner.

A few comments are necessary here related to the arising primal–dual equations (2.2). First, we begin the process with W_k set as the exact Hessian of the Lagrangian and the GMRES solution initialized to the zero vector. If a modification to W_k is made, we restart the solver with the solution initialized to the last solution computed for the previous W_k. Overall, for any given W_k, we perform at most $n+t$ GMRES iterations and simply attempt to follow the final step computed if this iteration limit is ever reached. We run the Hessian Modification Strategy for each trial step computed, but note that one could perform some inner iterations (a fixed number or until a sufficiently accurate solution is obtained) before considering a modification to W_k. Finally, in our implementation we set $W_k \leftarrow W_k + \mu I$ with $\mu \leftarrow 10^{-4}$ for the first modification and perform a similar perturbation after increasing $\mu \leftarrow 10\mu$ for each additional modification.

Finally, we present a simple technique for computing upper and lower bounds for the lengths of the tangential component u and the normal component v for a given primal step d during iteration k. Recall that as the two components are orthogonal, we have

$$\|d\|^2 = \|u\|^2 + \|v\|^2. \tag{6.3}$$

In addition, since $A_k u = 0$ we have the inequality

$$\|v\| \geq \|A_k v\|/\|A_k\| = \|A_k d\|/\|A_k\| \tag{6.4}$$

that, along with (6.3), implies

$$\|u\|^2 \leq \|d\|^2 - \|A_k d\|^2/\|A_k\|^2. \tag{6.5}$$

Thus, with (6.4) and (6.5) we can use

$$\nu \leftarrow \|A_k d\|^2 / \|A_k\|^2 \tag{6.6}$$

and

$$\Upsilon \leftarrow \|d\|^2 - \|A_k d\|^2 / \|A_k\|^2 \tag{6.7}$$

as an appropriate lower bound for $\|v\|^2$ and upper bound for $\|u\|^2$, respectively. We have implemented more practical forms of these quantities by replacing $\|A_k\|^2$ by the upper bound $\min\{n\|A_k\|_1^2, t\|A_k\|_\infty^2\}$. The accuracy of these estimates will vary; if they are poor, the algorithm will tend to modify the Hessian W_k in a conservative manner.

We summarize the above process with the following algorithm, which can be viewed as a particular implementation of the **repeat** loop in Algorithm `INS`.

Algorithm `INS-step`*: Step Computation for Algorithm* `INS`.

```
Set $j \leftarrow 0$, $(d^0, \delta^0) \leftarrow 0$, $W_k \leftarrow \nabla_{xx}^2 \mathcal{L}_k$, and $\mu \leftarrow 10^{-4}$
while $j < n + t$
    Increment $j \leftarrow j + 1$
    Perform a GMRES iteration on (2.2) to compute $(d^j, \delta^j)$
    Compute $\Upsilon^j$ and $\nu^j$ via (6.6) and (6.7)
    if Termination Test I or II is satisfied, then break
    if $(d^j, \delta^j)$ satisfies (4.7a)–(4.7c)
        Set $j \leftarrow 0$, $(d^0, \delta^0) \leftarrow (d^j, \delta^j)$, $W_k \leftarrow W_k + \mu I$, and $\mu \leftarrow 10\mu$
    endif
endwhile
return $(d_k, \delta_k) \leftarrow (d^j, \delta^j)$ and $(\Upsilon_k, \nu_k) \leftarrow (\Upsilon^j, \nu^j)$
```

Algorithm `INS` is designed for solving very large problems and its effectiveness is best studied in the context of practical applications. However, such a study is outside the scope of this paper and so we choose to illustrate the performance of the algorithm on problems from the CUTEr [1,11] collection. Problems from this set, for which AMPL models were available, were chosen based on memory limitations (problems with $n + t$ >1,000 were removed) and Assumption 1 (e.g., a problem was removed if the algorithm ever encountered a Jacobian A_k with a singular value, computed via Matlab's `svds` function, below 10^{-4}). The complete set is composed of 44 problems. Detailed results can be found in Table 3, where the headers are described in Table 2.

The data in Table 3 illustrate the practical behavior of our approach. First, we note that the run for each problem terminated successfully. It is particularly comforting to view the results for the strictly convex problems in the set [i.e., those labeled with a (*)], as one can see how Algorithm `INS` approximates a fast SQP approach in such a setting. Another interesting observation relates to the average numbers of positive eigenvalues of the iteration matrices (obtained via Matlab's `eigs` function) used to compute the step during each iteration. For some of the nonconvex problems, this value is (slightly) less than the number of variables n, which means that the Hessian Modification Strategy does not fully *convexify* the subproblem during every

Table 2 Key for Tables 3 and 4

Symbol	Meaning
Name	Name of problem
n	Number of variables
t	Number of constraints
Iter.	Total number of outer iterations
Inner iter.: tot.	Total number of inner iterations
Inner iter.: avg.	Avg. number of inner iterations per outer iteration
Pos. eig.: avg.	Avg. number of positive eigenvalues of modified primal–dual matrix
%TT-I	Percentage of times Termination Test I satisfied
%TT-II	Percentage of times Termination Test II satisfied
–	Algorithm failure

Table 3 Results for Algorithm `INS`

Name	n	t	Iter.	Inner iter.		Pos. eig.	%TT-I	%TT-II
				Tot.	Avg.	Avg.		
bt2(*)	3	1	18	53	2.94	3.00	66.67	33.33
bt3(*)	5	3	1	8	8.00	5.00	100.00	0.00
bt4	3	2	15	113	7.53	2.93	80.00	20.00
bt5	3	2	7	33	4.71	2.86	71.43	28.57
bt6(*)	5	2	12	58	4.83	5.00	66.67	33.33
bt7	5	3	45	322	7.16	4.96	33.33	66.67
bt9	4	2	17	74	4.35	4.00	41.18	58.82
bt10(*)	2	2	13	43	3.31	2.00	38.46	61.54
bt11(*)	5	3	9	69	7.67	5.00	44.44	55.56
bt12(*)	5	3	9	54	6.00	5.00	44.44	55.56
catena	32	11	48	1,186	24.71	31.94	10.42	89.58
dtoc1nd	735	490	49	4,217	86.06	734.73	83.67	16.33
eigena2	110	55	148	19,030	128.58	96.37	69.59	30.41
eigenaco	110	55	28	417	14.89	109.96	89.29	10.71
eigenb2	110	55	21	542	25.81	107.14	90.48	9.52
eigenbco	110	55	218	14,730	67.57	109.63	99.08	0.92
eigenc2	462	231	130	57,714	443.95	441.80	93.08	6.92
eigencco	30	15	172	3,361	19.54	29.58	98.26	1.74
fccu(*)	19	8	12	150	12.50	19.00	33.33	66.67
genhs28(*)	10	8	4	32	8.00	10.00	75.00	25.00
gilbert	1,000	1	22	55	2.50	833.91	18.18	81.82
hs006	2	1	17	60	3.53	1.76	64.71	35.29
hs007	2	1	8	28	3.50	1.75	12.50	87.50
hs008(*)	2	2	5	16	3.20	2.00	100.00	0.00
hs026	3	1	203	697	3.43	3.00	99.51	0.49

Table 3 continued

Name	n	t	Iter.	Inner iter.		Pos. eig.	%TT-I	%TT-II
				Tot.	Avg.	Avg.		
hs027(*)	3	1	14	45	3.21	3.00	21.43	78.57
hs028(*)	3	1	1	4	4.00	3.00	100.00	0.00
hs039	4	2	17	74	4.35	4.00	41.18	58.82
hs040(*)	4	3	6	73	12.17	4.00	66.67	33.33
hs046(*)	5	2	110	452	4.11	5.00	100.00	0.00
hs047	5	3	24	156	6.50	5.00	100.00	0.00
hs048(*)	5	2	1	7	7.00	5.00	100.00	0.00
hs049(*)	5	2	41	159	3.88	5.00	100.00	0.00
hs050(*)	5	3	8	58	7.25	5.00	100.00	0.00
hs051(*)	5	3	2	8	4.00	5.00	100.00	0.00
hs052(*)	5	3	1	8	8.00	5.00	100.00	0.00
hs077(*)	5	2	11	58	5.27	5.00	63.64	36.36
hs078(*)	5	3	5	63	12.60	5.00	40.00	60.00
hs079(*)	5	3	12	50	4.17	5.00	91.67	8.33
hs100lnp	7	2	9	60	6.67	6.89	22.22	77.78
hs111lnp	10	3	25	296	11.84	9.96	76.00	24.00
maratos(*)	2	1	4	11	2.75	2.00	25.00	75.00
mwright(*)	5	3	7	52	7.43	5.00	71.43	28.57
orthregb	27	6	21	89	4.24	24.10	95.24	4.76

A (*) indicates that the problem was strictly convex over the set of computed iterates

iteration. We can expect this difference to be more pronounced for larger values of the inexactness parameters $(\epsilon, \beta, \kappa)$. Finally, we find that for most of the problems the Model Reduction Condition is satisfied for the most recent value of the penalty parameter (i.e., Termination Test I is satisfied), which is comforting in view of the fact that the bound (4.2) can often be less demanding than those in (4.3).

We next compare the performance of Algorithm `INS` in relation to other approaches. As we are unaware of another globally convergent line search algorithm that does not require inertia information of primal–dual iteration matrices, we investigate the performance of Algorithm `INS` as compared to two inexact SQP approaches derived from the algorithm proposed in [2]. The first algorithm, which we call `ISQP-naive`, is a straightforward implementation of the inexact SQP algorithm in [2] (with parameter choices equal to those in Table 1) that simply ignores the presence of negative or zero curvature. This algorithm has similar SMART Tests for the step computation and is proved to converge for strictly convex problems, but its performance when confronted with a nonconvex problem is unpredictable. The second algorithm, which we call `ISQP-ideal`, is also based on the algorithm in [2] with SMART Tests, but computes the inertia of primal–dual matrices. With this information, modifications are made prior to the step computation (also by adding multiples of the identity matrix to

Table 4 Iteration and inner iteration statistics for Algorithm `INS` and two inexact SQP algorithms derived from the algorithm in [2]: `ISQP-naive` and `ISQP-ideal`

	Algorithm `INS`			`ISQP-naive`			`ISQP-ideal`		
Name	Iter.	Inner iter.		Iter.	Inner iter.		Iter.	Inner iter.	
		Tot.	Avg.		Tot.	Avg.		Tot.	Avg.
bt2	18	53	2.94	18	53	2.94	18	53	2.94
bt3	1	8	8.00	1	8	8.00	1	8	8.00
bt4	15	113	7.53	–	–	–	7	31	4.43
bt5	7	33	4.71	8	35	4.38	7	31	4.43
bt6	12	58	4.83	12	58	4.83	12	58	4.83
bt7	45	322	7.16	27	200	7.41	37	285	7.70
bt9	17	74	4.35	16	68	4.25	12	51	4.25
bt10	13	43	3.31	13	43	3.31	13	43	3.31
bt11	9	69	7.67	9	69	7.67	9	69	7.67
bt12	9	54	6.00	9	54	6.00	9	54	6.00
catena	48	1,186	24.71	–	–	–	48	1,159	24.15
dtoc1nd	49	4,217	86.06	–	–	–	71	3,801	53.54
eigena2	148	19,030	128.58	222	26, 110	117.61	44	2,924	66.45
eigenaco	28	417	14.89	29	400	13.79	34	499	14.68
eigenb2	21	542	25.81	21	542	25.81	16	213	13.31
eigenbco	218	14,730	67.57	–	–	–	268	16,607	61.97
eigenc2	130	57,714	443.95	47	8, 687	184.83	39	3,625	92.95
eigencco	172	3,361	19.54	99	1, 737	17.55	38	532	14.00
fccu	12	150	12.50	12	150	12.50	12	150	12.50
genhs28	4	32	8.00	4	32	8.00	4	32	8.00
gilbert	22	55	2.50	22	55	2.50	22	55	2.50
hs006	17	60	3.53	–	–	–	16	34	2.12
hs007	8	28	3.50	–	–	–	8	20	2.50
hs008	5	16	3.20	5	16	3.20	5	16	3.20
hs026	203	697	3.43	203	697	3.43	205	692	3.38
hs027	14	45	3.21	14	45	3.21	14	45	3.21
hs028	1	4	4.00	1	4	4.00	1	4	4.00
hs039	17	74	4.35	16	68	4.25	12	51	4.25
hs040	6	73	12.17	5	32	6.40	5	32	6.40
hs046	110	452	4.11	110	452	4.11	110	452	4.11
hs047	24	156	6.50	–	–	–	31	168	5.42
hs048	1	7	7.00	1	7	7.00	1	7	7.00
hs049	41	159	3.88	41	159	3.88	41	159	3.88
hs050	8	58	7.25	8	58	7.25	8	58	7.25
hs051	2	8	4.00	2	8	4.00	2	8	4.00
hs052	1	8	8.00	1	8	8.00	1	8	8.00
hs077	11	58	5.27	11	58	5.27	11	58	5.27

Table 4 continued

Name	Algorithm INS			ISQP-naive			ISQP-ideal		
	Iter.	Inner iter.		Iter.	Inner iter.		Iter.	Inner iter.	
		Tot.	Avg.		Tot.	Avg.		Tot.	Avg.
hs078	5	63	12.60	4	28	7.00	4	28	7.00
hs079	12	50	4.17	12	50	4.17	12	50	4.17
hs100lnp	9	60	6.67	9	60	6.67	9	60	6.67
hs111lnp	25	296	11.84	–	–	–	20	199	9.95
maratos	4	11	2.75	4	11	2.75	4	11	2.75
mwright	7	52	7.43	7	52	7.43	7	52	7.43
orthregb	21	89	4.24	21	89	4.24	11	63	5.73

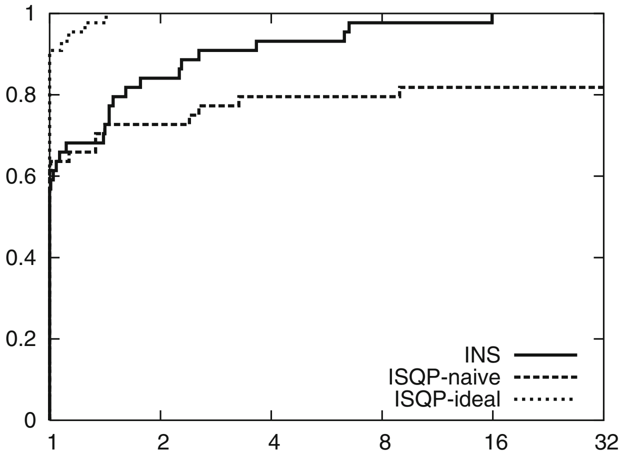

Fig. 1 Performance profile for iteration counts

W_k) in order to create a strictly convex subproblem during each iteration. Note that this algorithm does in fact require explicit representations and factorizations of primal–dual matrices, and thus should not be considered a practical competitor of our approach on problems for which Algorithm INS was designed; we simply implemented this method to gauge how Algorithm INS compares to an idealized approach.

We state that an algorithm fails in one of three ways. First, the outer iteration limit of 1,000 can be reached before an iterate satisfies (6.1) and (6.2). Second, the final step computed by GMRES can be an ascent direction for the penalty function for all allowable values of the penalty parameter; i.e., $D\phi(d_k; \pi) > 0$ for all $\pi \geq \pi_{k-1}$. Finally, the line search can return a steplength α_k that is practically zero. We say that this is the case if $\alpha_k \leq 10^{-6}$.

Results for the three algorithms based on two performance measures, the number of iterations and the total number of inner iterations required to find a solution, are presented in Table 4. We also summarize the results in Figs. 1 and 2 in terms of the

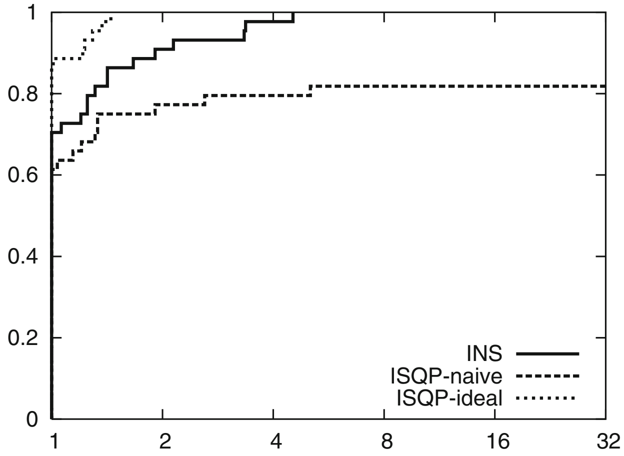

Fig. 2 Performance profile for inner iteration counts

logarithmic performance profiles of Dolan and Moré [7]. Function values on the leftmost side of each figure represent the percentage of times that solver achieved the lowest value of a given performance measure and the rightmost values illustrate the robustness of the approach.

At first glance it appears that `ISQP-naive` is surprisingly robust despite the fact that nonconvexity was ignored throughout the solution process. However, the main reason for this is that many of the problems were strictly convex over the set of computed iterates. Overall this approach failed on a large portion of the nonconvex problems.

7 Final remarks

We have presented a matrix-free inexact Newton algorithm for solving equality constrained optimization problems. The method is based on an extension of the Sufficient Merit function Approximation Reduction Termination (SMART) Tests presented in [2] to cases where negative or zero curvature is present. In addition, we have proposed a mechanism for determining when a modification to the Hessian matrix should be made within an iterative step computation, and have proved (under common assumptions) that the algorithm is globally convergent to first-order optimal points. The robustness of the method has been illustrated on a set of standard test problems. Our approach can be used to solve unconstrained problems and may be extended to generally constrained problems if, for example, our methodology is applied to the equality constrained barrier subproblems arising in an interior point method.

We close by remarking on two practical extensions to Algorithm `INS` as it is presented in Sect. 4. First, as in [2], we mention that all of our analysis holds if one chooses $\kappa \geq 1$ in Termination Test I, provided the additional condition

$$\|\rho_k\| \leq \max\left\{\kappa_1\|g_k + A_k^T\lambda_k\|, \kappa_2\|c_k\|\right\}$$

with $0 < \kappa_1 < 1$ and $0 < \kappa_2$ is also enforced. This type of flexibility may be useful if one uses a left preconditioner for (2.2), as this may yield steps corresponding to residuals larger in norm than the right-hand-side vector $(g_k + A_k^T \lambda_k, c_k)$, or if one applies Termination Test I in an implementation that does not compute steps directly via (2.2). Similarly, a more flexible version of the Hessian Modification Strategy can be implemented along with certain types of modification techniques. If it is known that after a certain modification we have $y^T W_k y \geq \mu \|y\|^2$ for some $\mu > 0$ for all $y \in \mathbb{R}^n$ such that $A_k y = 0$, then no more perturbations are necessary if the algorithm reverts to the version of SMART Tests presented in [2]. All of the results in this paper hold if a subsequence of iterations are performed in this manner.

Acknowledgments The authors would like to thank the referees for careful readings of this paper and for their many useful comments and suggestions.

References

1. Bongartz, I., Conn, A.R., Gould, N.I.M., Toint, Ph.L.: CUTE: constrained and unconstrained testing environment. ACM Trans. Math. Softw. **21**(1), 123–160 (1995)
2. Byrd, R.H., Curtis, F.E., Nocedal, J.: An inexact SQP method for equality constrained optimization. SIAM J. Optim. **19**(1), 351–369 (2008)
3. Conn, A.R., Gould, N.I.M., Toint, Ph.: Trust-Region Methods. MPS-SIAM Series on Optimization. SIAM Publications, Philadelphia (2000)
4. Curtis, F.E., Haber, E.: Numerical experience with an inexact SQP method for PDE-constrained optimization. (In preparation)
5. Dennis, J.E., Vicente, L.N.: On the convergence theory of trust-region based algorithms for equality-constrained optimization. SIAM J. Optim. **7**(4), 927–950 (1997)
6. Di-Pillo, J., Lucidi, S., Palagi, L.: Convergence to second-order stationary points of a primal–dual algorithm model for nonlinear programming. Math. Oper. Res. **30**(4), 879–915 (2005)
7. Dolan, E.D., Moré, J.J.: Benchmarking optimization software with performance profiles. Math. Program. Ser. A **91**, 201–213 (2002)
8. El-Alem, M.: Convergence to a 2nd order point of a trust-region algorithm with nonmonotonic penalty parameter for constrained optimization. J. Optim. Theory Appl. **91**(1), 61–79 (1996)
9. Facchinei, F., Lucidi, S.: Convergence to second order stationary points in inequality constrained optimization. Math. Oper. Res. **23**, 746–766 (1998)
10. Freund, R.W., Nachtigal, N.M.: A new Krylov-subspace method for symmetric indefinite linear systems. In: Ames, W.F. (ed.) Proceedings of the 14th IMACS World Congress on Computational and Applied Mathematics, pp. 1253–1256. IMACS (1994)
11. Gould, N.I.M., Orban, D., Toint, Ph.L.: CUTEr and sifdec: a constrained and unconstrained testing environment, revisited. ACM Trans. Math. Softw. **29**(4), 373–394 (2003)
12. Gould, N.I.M., Toint, Ph.L.: A note on the second-order convergence of optimization algorithms using barrier functions. Math. Program. **85**(2), 433–438 (1999)
13. Heinkenschloss, M., Ridzal, D.: An inexact trust-region SQP method with applications to PDE-constrained optimization. In: Steinbach, O., Of, G. (eds.) Numerical Mathematics and Advance Applications: Proceedings of Enumath 2007, the 7th European Conference on Numerical Mathematics and Advanced Applications, Graz, Austria, September 2007, Springer, Heidelberg (2008, submitted). doi:10.1007/978-3-540-69777-0_73
14. Heinkenschloss, M., Vicente, L.N.: Analysis of inexact trust-region SQP algorithms. SIAM J. Optim. **12**, 283–302 (2001)
15. Kelley, C.T.: Iterative methods for linear and nonlinear equations: Matlab codes (1994). http://www4.ncsu.edu/~ctk/matlab_roots.html
16. Moguerza, J.M., Prieto, F.J.: An augmented Lagrangian interior point method using directions of negative curvature. Math. Program. **95**(3), 573–616 (2003)

17. Paige, C.C., Saunders, M.A.: Solution of sparse indefinite systems of linear equations. SIAM J. Numer. Anal. **12**(4), 617–629 (1975)
18. Paige, C.C., Saunders, M.A.: MINRES: sparse symmetric equations (2003). http://www.stanford.edu/group/SOL/software/minres.html
19. Ridzal, D.: Trust region SQP methods with inexact linear system solves for large-scale optimization. Ph.D. thesis, Rice University (2006)
20. Saad, Y., Schultz, M.H.: GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems. SIAM J. Sci. Stat. Comput. **7**, 856–869 (1986)
21. Ulbrich, S.: Generalized SQP-methods with "parareal" time-domain decomposition for time-dependent PDE-constrained optimization. In: Biegler, L., Ghattas, O., Heinkenschloss, M., Keyes, D., van Bloemen Waanders, B. (eds.) Real-Time PDE-Constrained Optimization, pp. 145–168. SIAM, Philadelphia (2008)
22. Vanderbei, R.J., Shanno, D.F.: An interior point algorithm for nonconvex nonlinear programming. Comput. Optim. Appl. **13**, 231–252 (1999)