A Proximal-Gradient Method for Constrained Optimization *

YUTONG DAI, XIAOYI QU, AND DANIEL P. ROBINSON†

Abstract. We present a new algorithm for solving optimization problems with objective functions that are the sum of a smooth function and a (potentially) nonsmooth regularization function, and nonlinear equality constraints. The algorithm may be viewed as an extension of the well-known proximal-gradient method that is applicable when constraints are not present. To account for nonlinear equality constraints, we combine a decomposition procedure for computing trial steps with an exact merit function for determining trial step acceptance. Under common assumptions, we show that both the proximal parameter and merit function parameter eventually remain fixed, and then prove a worst-case complexity result for the maximum number of iterations before an iterate satisfying approximate first-order optimality conditions for a given tolerance is computed. Our preliminary numerical results indicate that our approach has great promise, especially in terms of returning approximate solutions that are structured (e.g., sparse solutions when a one-norm regularizer is used).

Key words. nonlinear optimization, nonconvex optimization, worst-case iteration complexity, regularization methods, sequential quadratic programming, sequential quadratic optimization

AMS subject classifications. 49M37, 65K05, 65K10, 65Y20, 68Q25, 90C30, 90C60

1. Introduction.

In this paper we consider the problem

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

where $f : \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable, $r : \mathbb{R}^n \to \mathbb{R}$ is convex (possibly nondifferentiable) and nonnegative valued, and $c : \mathbb{R}^n \to \mathbb{R}^m$ is continuously differentiable with $m \leq n$. The optimization problem (1.1) has applications in model predictive control [2], image processing [35], nonsmooth optimization on a Steifel manifold [34], and low rank matrix completion [6]. In addition, optimization problems such as sparse approximation, empirical risk minimization, and neural network modeling with mixed activations can be reformulated as (1.1); see [37] for additional details.

When the regularizer r is not present, the algorithms most commonly employed to solve problem (1.1) are penalty methods [9, 11, 12, 15, 28, 32, 36] and sequential quadratic optimization (SQO) methods [1, 19, 20, 22, 23, 24, 29, 31]. Penalty methods is to solve problem (1.1) by minimizing a sequence of unconstrained optimization subproblems defined in terms of f, a measure of constraint violation, and various parameters (e.g., Lagrange multiplier estimates and penalty parameters). After each minimization subproblem in the sequence is solved, the parameters are updated in a manner that allows for convergence guarantees. Since computing each subproblem minimizer may be expensive, and the number of subproblems solved may be nontrivial, penalty methods often require a significant amount of computation (e.g., numbers of iterations, function/derivative evaluations, and linear systems solved), which may be prohibitive. On the other hand, during each iteration of a line search SQO method, the main expense is the computation of a search direction, which is achieved by solving a single linear system of equations. Equivalently, the search direction is the minimizer of a certain quadratic approximation of f subject to a linearization of the constraints. SQO methods are generally viewed as the state-of-the-art because

*This work is supported by the U.S. Office of Naval Research under Award Number N00014-21-1-2532 and by the U.S. National Science Foundation, Division of Mathematical Sciences, Computational Mathematics Program under Award Number DMS–2012243.

†All authors are from the Department of Industrial and Systems Engineering, Lehigh University, Bethlehem, PA, USA; E-mail: yud319@lehigh.edu, xiq322@lehigh.edu, daniel.p.robinson@lehigh.edu

of their remarkable practical performance. The superior performance of line search SQO methods over penalty methods can be attributed to two main sources. First, line search SQO methods solve a sequence of linear systems rather than a sequence of general optimization subproblems. Second, the search directions for SQO methods are designed to find a solution of problem (1.1) (again, when r is not present), whereas penalty methods *indirectly* aim to find a solution of problem (1.1) (again, when r is not present) by adjusting its parameters after minimizing each subproblem.

When the constraint $c(x) = 0$ is not present in problem (1.1), the algorithms most commonly employed are variants of the proximal-gradient (PG) method [4, 5, 7, 8, 25, 26]. Each iterate of a basic PG method is the minimizer of a subproblem (i.e., the PG subproblem) formed by replacing f in (2.1) by the sum of its first-order Taylor expansion (expanded at the current point) and a simple quadratic-regularization term. For some commonly used regularizers, the PG subproblem has a closed-form solution, which is an attractive feature of such methods. Moreover, since the regularizer r is explicitly used in the definition of the PG subproblem (i.e., it is not approximated), the solutions generated by a PG method inherit the structure induced by the regularizer (e.g., if $r(x) = \|x\|_1$, then a PG method can produce sparse solution estimates). This *structure preserving* property is an important feature of PG methods when used to solve problem (1.1) (again, when the constraint $c(x) = 0$ is not present).

The work in this paper is motivated by both SQO methods for solving (1.1) when r is not present, and the structure preserving property of PG methods for solving (1.1) when the constraint $c(x) = 0$ is not present. In particular, we design and analyze a method for solving problem (1.1) based on subproblems that linearize the constraints (like SQO methods) and explicitly use the regularizer (like PG methods).

1.1. Literature review. We are aware of four papers, namely [14, 17, 27, 37], that present algorithms for minimizing regularized optimization functions subject to nonlinear constraints. The algorithms in [14, 37] are penalty methods built upon the popular augmented Lagrangian function. Therefore, both approaches have a penalty parameter and a vector of Lagrange multiplier estimates that balance the objective and constraint functions, and must be updated throughout the optimization procedure. We note that [14] can solve regularized optimization problems with both equality and inequality constraints, whereas the algorithm in [37] can only handle special classes of regularized optimization problems with constraints. The algorithms presented in [17, 27] are of the SQO variety. In [17], subgradient information of the nonsmooth function is used to formulate a sequence of min-max subproblems. Since the regularizer is approximated in each subproblem, the structure preserving property of the iterates is lost. In contrast, [27] relies on a smoothing technique that approximates the nonsmooth term in the objective function and, thereafter, sequentially solves a convex quadratic problem with linear constraints. Unfortunately, in general, the smoothing technique ruins the structure of the composite optimization problem, and consequently the structure preserving property is lost.

1.2. Contributions. Our contributions relate to the proposal and analysis of a new algorithm for solving problem (1.1), as we summarize next.

- We propose a PG-based algorithm for solving problem (1.1) that uses subproblems with linearized constraints (like SQO methods) and explicit regularization (like PG methods). The method that results from this combination avoids the previously discussed challenges and weaknesses of augmented Lagrangian approaches, and provides solution estimates that are structure preserving. During each iteration, we compute a trial step as the sum of two orthogonal directions called the normal and tangential steps. First, the normal step is computed from a trust region subproblem designed to reduce the constraint violation. Second, the tangential step is computed from a linearly constrained convex optimization subproblem with objective function reminiscent of PG methods (i.e., r appears explicitly and a proximal term is used). Overall, the tangential step aims to reduce the objective function while maintaining the predicted progress in reducing infeasibility achieved by the normal step. The quality of the trial step, defined as the sum of the normal and tangential steps, is then determined by an ℓ_2 merit function that uses a merit parameter to weight the objective function relative to the two-norm of the constraint violation. The merit parameter and PG parameter (i.e., the weight on the proximal term) are reduced as the iterations proceed, if necessary, to promote convergence of the iterates to a solution of problem (1.1).

- Under minimal assumptions, we prove that a measure of first-order optimality for a feasibility problem converges to zero. Under additional commonly used assumptions, we prove that the merit parameter and PG parameter both remain uniformly bounded away from zero. These results allow us to then prove that our algorithm generates a sequence of iterates such that any limit point is a KKT point (see Theorem 3.17). Moreover, we provide a worst-case complexity result for the maximum number of iterations before a certain criticality measure will be less than a given tolerance (see Theorem 3.15).
- We present numerical experiments that verify our theoretical convergence results, and illustrate that our algorithm is capable of returning solutions that preserve the structure related to r. Specifically, we confirm that our method returns sparse solution estimates when r is chosen as the ℓ_1-norm function, which is known to be a sparsity-inducing regularizer.

1.3. Notation and assumptions. We use $\mathbb{R}$ to denote the set of real numbers (i.e., scalars), $\mathbb{R}_{\geq 0}$ (resp., $\mathbb{R}_{>0}$) to denote the set of nonnegative (resp., positive) real numbers, $\mathbb{R}^n$ to denote the set of n-dimensional real vectors, and $\mathbb{R}^{m\times n}$ to denote the set of m-by-n-dimensional real matrices. The set of natural numbers is denoted as $\mathbb{N} := \{0, 1, 2, \dots\}$. Given a matrix $M \in \mathbb{R}^{m\times n}$, we let $\sigma_{\min}(M)$ (resp., $\sigma_{\max}(M)$) denote the smallest (resp., largest) singular value of M. For $v \in \mathbb{R}^n$, we let $\|v\|_2 := \sqrt{\sum_{i=1}^n v_i^2}$ denote its two norm. For a nonempty compact set $\mathcal{R} \subset \mathbb{R}^n$, we let $\|\mathcal{R}\|_2 := \max\{\|s\|_2 : s \in \mathcal{R}\}$ denote its largest element measured in the two-norm.

The following assumption is used throughout the paper.

ASSUMPTION 1.1. *Let $\mathcal{X} \subset \mathbb{R}^n$ be an open convex set that contains the iterates $\{x_k\} \subset \mathbb{R}^n$ and trial steps $\{x_k + s_k\} \subset \mathbb{R}^n$ generated by Algorithm 2.1. The function $f : \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable and bounded below over $\mathcal{X}$ and its gradient function $\nabla f : \mathbb{R}^n \to \mathbb{R}^n$ is Lipschitz continuous and bounded over $\mathcal{X}$. Similarly, the function $c : \mathbb{R}^n \to \mathbb{R}^m$ is continuously differentiable and bounded over $\mathcal{X}$ and its Jacobian $\nabla c(x)^T$ is Lipschitz continuous and bounded over $\mathcal{X}$. Finally, the function $r : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ is convex and its subdifferential $\partial r : \mathbb{R}^n \to \mathbb{R}^n$ is bounded over $\mathcal{X}$.*

Under Assumption 1.1, there exist constants $(f_{\inf}, \kappa_{\nabla f}, \kappa_{\partial r}, \kappa_c, \kappa_{\nabla c}, L_g, L_J) \in \mathbb{R} \times \mathbb{R}_{>0} \times \mathbb{R}_{>0} \times \mathbb{R}_{>0} \times \mathbb{R}_{>0} \times \mathbb{R}_{>0} \times \mathbb{R}_{>0}$ such that for all $x \in \mathcal{X}$ one has

$$
\begin{aligned}
&f(x) \geq f_{\inf}, \qquad \|\nabla f(x)\|_2 \leq \kappa_{\nabla f}, \qquad \|\partial r(x)\|_2 \leq \kappa_{\partial r}, \\
&\|c(x)\|_2 \leq \kappa_c, \qquad \|\nabla c(x)^T\|_2 \leq \kappa_{\nabla c},
\end{aligned} \tag{1.2}
$$

and for all $(x,\bar{x}) \in \mathcal{X} \times \mathcal{X}$ one has

$$\|\nabla f(x) - \nabla f(\bar{x})\|_2 \le L_g \|x - \bar{x}\|_2 \;\text{ and }\; \|\nabla c(x)^T - \nabla c(\bar{x})^T\|_2 \le L_J \|x - \bar{x}\|_2. \tag{1.3}$$

For convenience, we define $g(x) := \nabla f(x)$ and $J(x) := \nabla c(x)^T$. We append a natural number as a subscript for a quantity to denote its value during an iteration of an algorithm; i.e., we let $f_k := f(x_k)$, $g_k := g(x_k)$, $c_k := c(x_k)$, and $J_k := J(x_k)$.

1.4. Organization. In Section 2, we propose our algorithm for solving problem (1.1), and its convergence properties are presented in Section 3. In Section 4, we discuss our numerical tests. Final conclusions are provided in Section 5.

2. Algorithm.

The algorithm that we propose for solving problem (1.1) is formally stated as Algorithm 2.1. Given the kth iterate x_k, the kth PG parameter α_k, and constant $\kappa_v \in \mathbb{R}_{>0}$, we compute a step v_k that aims to reduce the constraint infeasibility at x_k as an approximate solution to the following problem:

$$\min_{v \in \mathbb{R}^n} \; m_k(v) \;\text{ s.t. }\; \|v\|_2 \le \kappa_v \alpha_k \|J_k^T c_k\|_2, \;\;\text{ with }\;\; m_k(v) := \tfrac{1}{2}\|c_k + J_k v\|_2^2. \tag{2.1}$$

The PG parameter α_k is used to define the trust-region constraint so that $\{v_k\} \to 0$ if $\{\alpha_k\} \to 0$. We consider a vector v_k to be an adequate approximate solution to subproblem (2.1) if, for some $\kappa_v \in \mathbb{R}_{>0}$, it satisfies the following conditions:

$$v_k \in \operatorname{Range}(J_k^T), \tag{2.2a}$$

$$\|v_k\|_2 \le \kappa_v \alpha_k \|J_k^T c_k\|_2, \;\text{ and} \tag{2.2b}$$

$$\|c_k + J_k v_k\|_2 \le \|c_k + J_k v_k^c\|_2 \tag{2.2c}$$

where v_k^c is the Cauchy point given by

$$v_k^c := -\beta_k^c J_k^T c_k \;\;\text{ with }\;\; \beta_k^c := \arg\min_{\beta \in \mathbb{R}} \; m_k(-\beta J_k^T c_k) \;\;\text{ s.t. }\;\; 0 \le \beta \le \kappa_v \alpha_k. \tag{2.3}$$

In other words, the Cauchy point v_k^c minimizes $m_k(v)$ along the direction $-\nabla m_k(0) = -J_k^T c_k$ and within $\{v : \|v\|_2 \le \kappa_v \alpha_k \|J_k^T c_k\|_2\}$. It is known (see [10]) that v_k^c satisfies

$$m_k(0) - m_k(v_k^c) \ge \tfrac{1}{2}\|J_k^T c_k\|_2^2 \min\left\{\tfrac{1}{1+\|J_k^T J_k\|_2}, \kappa_v \alpha_k\right\}. \tag{2.4}$$

We note that the conditions (2.2) are well-posed since they are satisfied by $v_k = v_k^c$.

Next, we compute a direction u_k that maintains the level of linearized infeasibility achieved by v_k while also reducing a model of the objective function. Specifically, we compute u_k as the unique solution to the strongly convex subproblem

$$\begin{aligned} u_k &:= \arg\min_{u \in \mathbb{R}^n} \; g_k^T(v_k + u) + \tfrac{1}{2\alpha_k}\|v_k + u\|_2^2 + r(x_k + v_k + u) \;\text{ s.t. }\; J_k u = 0. \\ &= \arg\min_{u \in \mathbb{R}^n} \; (g_k + \tfrac{1}{\alpha_k} v_k)^T u + \tfrac{1}{2\alpha_k}\|u\|_2^2 + r(x_k + v_k + u) \;\text{ s.t. }\; J_k u = 0 \\ &= \arg\min_{u \in \mathbb{R}^n} \; g_k^T u + \tfrac{1}{2\alpha_k}\|u\|_2^2 + r(x_k + v_k + u) \;\text{ s.t. }\; J_k u = 0 \end{aligned} \tag{2.5}$$

where we used the fact that every u feasible for (2.5) satisfies $v_k^T u = 0$ since $v_k \in \operatorname{Range}(J_k^T)$ (see (2.2a)). The trial step s_k is then defined as

$$s_k := v_k + u_k. \tag{2.6}$$

We adopt the ℓ_2 merit function, which for parameter $\tau \in \mathbb{R}_{>0}$ is defined as

$$\Phi_\tau(x) := \tau\big(f(x) + r(x)\big) + \|c(x)\|_2.$$

During the kth iteration, we want to choose τ_k such that $\tau_k \leq \tau_{k-1}$ and s_k is a direction of sufficient descent for the merit function $\Phi_{\tau_k}(\cdot)$ at x_k. To define an appropriate value for τ_k, let us define the model of the merit function given by

$$q_k(s, \tau) := \tau\big(f_k + g_k^T s + \tfrac{1}{2\alpha_k}\|s\|_2^2 + r(x_k + s)\big) + \|c_k + J_k s\|_2,$$

as well as the change in the model

$$\begin{aligned} \Delta q_k(s, \tau) &:= q_k(0) - q_k(s) \\ &= -\tau\big(g_k^T s + \tfrac{1}{2\alpha_k}\|s\|_2^2 + r(x_k + s) - r_k\big) + \|c_k\|_2 - \|c_k + J_k s\|_2. \end{aligned} \tag{2.7}$$

Then, with parameters $\sigma_c \in (0, 1)$ and $\sigma_u \in (0, \frac{1}{2}]$, we set $\bar{\sigma}_u := \sigma_u + \frac{1}{2} \in (\frac{1}{2}, 1]$ and

$$\tau_{k,\text{trial}} \leftarrow \begin{cases} \infty & \text{if } g_k^T s_k + \frac{\bar{\sigma}_u\|s_k\|_2^2}{\alpha_k} + r(x_k + s_k) - r_k \leq 0, \\ \frac{(1-\sigma_c)(\|c_k\|_2 - \|c_k + J_k v_k\|_2)}{g_k^T s_k + \frac{\bar{\sigma}_u\|s_k\|_2^2}{\alpha_k} + r(x_k+s_k) - r_k} & \text{otherwise,} \end{cases}$$

and then set, with $\epsilon_\tau \in (0, 1)$, the kth merit parameter value as

$$\tau_k \leftarrow \begin{cases} \tau_{k-1} & \text{if } \tau_{k-1} \leq \tau_{k,\text{trial}}, \\ \min\{(1-\epsilon_\tau)\tau_{k-1}, \tau_{k,\text{trial}}\} & \text{otherwise.} \end{cases} \tag{2.8}$$

This update ensures that if the merit parameter is decreased during the kth iteration, it is decreased by at least a fraction of its previous value. Moreover, the value for $\tau_{k,\text{trial}}$ ensures that $\Delta q_k(s_k, \tau_k)$ is an upper bound for quantities related to measures of criticality for problem (1.1) (see Lemma 3.4). Moreover, Lemma 3.4 shows that $-\Delta q_k(s_k, \tau_k)$ is an upper bound for the directional derivative of $\Phi_{\tau_k}(\cdot)$ at x_k in the direction s_k (this result holds regardless of the value of the merit parameter).

The kth iteration is completed by checking whether the merit function achieves sufficient decrease in Line 19, and then defining the next iterate and proximal parameter. Specifically, if sufficient decrease is observed in the merit function, then the trial step s_k is accepted (i.e., $x_{k+1} \leftarrow x_k + s_k$) and the proximal parameter value is unchanged (i.e., $\alpha_{k+1} \leftarrow \alpha_k$); otherwise, the trial step is rejected (i.e., $x_{k+1} \leftarrow x_k$) and the proximal parameter value is decreased (i.e., $\alpha_{k+1} \leftarrow \xi\alpha_k$ for some $\xi \in (0, 1)$). This updating scheme motivates the definition of the following index set:

$$\mathcal{S} := \{k : x_{k+1} = x_k + s_k\}, \tag{2.9}$$

which contains the indices of the successful iterations associated with Algorithm 2.1.

3. Analysis.

In this section, we prove convergence results for Algorithm 2.1. Our first result shows that the normal step v_k is zero if and only if $J_k^T c_k$ is zero.

LEMMA 3.1. *For all $k \in \mathbb{N}$, it holds that $v_k = 0$ if and only if $J_k^T c_k = 0$.*

Proof. If $J_k^T c_k = 0$, it follows from (2.2b) that $v_k = 0$. To prove the reverse implication, suppose that $v_k = 0$. Then it follows from (2.2c) that $m_k(v_k) \leq m_k(v_k^c)$, which combined with (2.4) and $v_k = 0$ shows that $0 = m_k(0) - m_k(v_k) \geq m_k(0) - m_k(v_k^c) \geq \frac{1}{2}\|J_k^T c_k\|_2^2 \min\left\{\frac{1}{1+\|J_k^T J_k\|_2}, \kappa_v\alpha_k\right\}$. Since $\alpha_k > 0$ for all $k \in \mathbb{N}$ and $\kappa_v \in \mathbb{R}_{>0}$, it follows that $J_k^T c_k = 0$, completing the proof. ∎

Algorithm 2.1 A proximal-gradient algorithm for problem (1.1).

1: **input:** $x_0 \in \mathbb{R}^n$, $\alpha_0 \in \mathbb{R}_{>0}$, and $\tau_{-1} \in \mathbb{R}_{>0}$.
2: **constants:** $\kappa_v \in \mathbb{R}_{>0}$, $\{\sigma_c, \epsilon_\tau, \xi, \eta, \} \subset (0,1)$, and $\sigma_u \in (0, 1/2]$.
3: **for** $k = 0, 1, 2, \ldots$ **do**
4:     **if** $J_k^T c_k \neq 0$ **then**
5:         Compute v_k as an approximate solution to (2.1) satisfying (2.2).
6:     **else**
7:         Set $v_k \leftarrow 0$.
8:         **if** $c_k \neq 0$ **then**
9:             **return** x_k (infeasible stationary point)
10:         **end if**
11:     **end if**
12:     Compute u_k as the unique solution to (2.5).
13:     Set $s_k \leftarrow v_k + u_k$.
14:     **if** $s_k = 0$ **then**
15:         **return** x_k (first-order KKT point)
16:     **end if**
17:     Compute τ_k using (2.8).
18:     Compute $\Delta q_k(s_k, \tau_k)$ using (2.7).
19:     **if** $\Phi_{\tau_k}(x_k + s_k) \leq \Phi_{\tau_k}(x_k) - \eta \Delta q_k(s_k, \tau_k)$ **then**
20:         Set $x_{k+1} \leftarrow x_k + s_k$ and $\alpha_{k+1} \leftarrow \alpha_k$.
21:     **else**
22:         Set $x_{k+1} \leftarrow x_k$ and $\alpha_{k+1} \leftarrow \xi \alpha_k$.
23:     **end if**
24: **end for**

Concerning the computation of the tangential step u_k, it follows from the optimality conditions for the convex optimization problem (2.5) that u_k and the resulting $s_k = v_k + u_k$ satisfy, for some $g_{r,k} \in \partial r(x_k + s_k)$ and $y_k \in \mathbb{R}^m$, the equalities

$$g_k + \tfrac{1}{\alpha_k} u_k + g_{r,k} - J_k^T y_k = 0 \quad \text{and} \quad J_k u_k = 0. \tag{3.1}$$

Multiplying the first equality by u_k^T and using the second equality, it follows that

$$(g_k + g_{r,k})^T u_k + \tfrac{1}{\alpha_k} \|u_k\|_2^2 = 0. \tag{3.2}$$

These equations related to the tangential step u_k will be useful in the analysis.

3.1. Finite termination. In this section we justify the finite termination conditions in Algorithm 2.1 given in line 9 and line 15. In particular, we show that if Algorithm 2.1 terminates in line 9 then x_k is an infeasible stationary point, and if termination occurs in line 15 then x_k is a first-order KKT point for problem (1.1).

THEOREM 3.2. *The following finite termination results hold for Algorithm 2.1.*

(i) If termination occurs in line 9 then x_k is an infeasible stationary point, i.e., x_k satisfies $c_k \neq 0$ and $J_k^T c_k = 0$.

(ii) If termination occurs in line 15 then x_k is a first-order KKT point for (1.1).

Proof. To prove part (i), suppose that termination occurs in line 9 so that $c_k \neq 0$ and $v_k = 0$. It follows from $v_k = 0$ and Lemma 3.1 that $J_k^T c_k = 0$, as claimed.

To prove part (ii), suppose that termination occurs in line 15 so that $s_k = 0$. Since by construction $v_k^T u_k = 0$, it also follows that $v_k = u_k = 0$. It follows from $v_k = 0$

and Lemma 3.1 that $J_k^T c_k = 0$. Since termination must not have occurred in line 9, we also know that $c_k = 0$. It follows from $v_k = u_k = 0$ and (3.1) that there exists $g_{r,k} \in \partial r(x_k + s_k) \equiv \partial r(x_k)$ and $y_k \in \mathbb{R}^m$ so that $g_k + g_{r,k} - J_k^T y_k = 0$. Combining this equality with $c_k = 0$ shows that x_k is a first-order KKT point for problem (1.1). ∎

Theorem 3.2 shows that if Algorithm 2.1 finitely terminates, then the vector x_k returned has favorable properties. Admittedly, although finite termination in line 9 is not ideal, the existence of infeasible stationary points is something that every algorithm must contend with unless an appropriate constraint qualification is assumed.

3.2. Non-finite termination. In this section, we study the convergence properties of Algorithm 2.1 when finite termination does not occur. Therefore, given how Algorithm 2.1 is constructed, we know in this section that, for all $k \in \mathbb{N}$, it holds that

$$(i)\ s_k \neq 0 \ \text{ and } \ (ii)\ J_k^T c_k = 0 \text{ if and only if } c_k = 0. \tag{3.3}$$

Our first goal is to prove a bound on the directional derivative of $\Phi_\tau(\cdot)$ at x_k in the direction s_k. Given the Lipschitz constants L_g and L_J in Assumption 1.1, it follows for all $t \in \mathbb{R}_{>0}$ from [13, equation (19)] that

$$\begin{aligned} f(x_k + ts_k) &\le f_k + t g_k^T s_k + \tfrac{L_g}{2} t^2 \|s_k\|_2^2 \ \text{ and} \\ \|c(x_k + ts_k)\|_2 &\le \|c_k + tJ_k s_k\|_2 + \tfrac{L_J}{2} t^2 \|s_k\|_2^2. \end{aligned} \tag{3.4}$$

The next result gives an upper bound on the quantity $D_{\Phi_\tau}(x_k, s_k)$, which we use to denote the directional derivative of $\Phi_\tau(\cdot)$ at x_k in the direction s_k.

LEMMA 3.3. *The directional derivative of the merit function satisfies*

$$D_{\Phi_\tau}(x_k, s_k) \le \tau\big(g_k^T s_k + r(x_k + s_k) - r_k\big) + \|c_k + J_k s_k\|_2 - \|c_k\|_2.$$

Proof. For all $t \in \mathbb{R}_{>0}$, it follows from (3.4) and the triangle inequality that

$$\begin{aligned} \|c(x_k + ts_k)\|_2 - \|c_k\|_2 &\le \|c_k + tJ_k s_k\|_2 - \|c_k\|_2 + \tfrac{L_J}{2} t^2 \|s_k\|_2^2 \\ &\le t\|c_k + J_k s_k\|_2 + (1-t)\|c_k\|_2 - \|c_k\|_2 + \tfrac{L_J}{2} t^2 \|s_k\|_2^2 \\ &= t\|c_k + J_k s_k\|_2 - t\|c_k\|_2 + \tfrac{L_J}{2} t^2 \|s_k\|_2^2. \end{aligned}$$

On the other hand, it follows from [3, Theorem 2.25] that $D_r(x_k, s_k) \le r(x_k + s_k) - r(x_k)$. The conclusion follows from this result, the previous displayed equation after dividing by t and taking the limit $t \searrow 0$, and the fact that f is differentiable. ∎

Combining the previous lemma with how the merit parameter τ_k is defined, allows us to prove that the change in the model $q_k(s_k, \tau_k)$ is an upper bound for quantities used in our ultimate convergence result.

LEMMA 3.4. *The choice of τ_k in (2.8) ensures that the direction s_k satisfies*

$$\begin{aligned} \Delta q_k(s_k, \tau_k) &\ge \tfrac{\sigma_u \tau_k}{\alpha_k} \|s_k\|_2^2 + \sigma_c\big(\|c_k\|_2 - \|c_k + J_k v_k\|_2\big) > 0 \ \text{ and} \\ D_{\Phi_{\tau_k}}(x_k, s_k) &\le -\tfrac{\sigma_u \tau_k}{\alpha_k} \|s_k\|_2^2 - \sigma_c\big(\|c_k\|_2 - \|c_k + J_k v_k\|_2\big) < 0. \end{aligned}$$

Proof. The first result follows from (2.8), definition of $\tau_{k,\text{trial}}$, and $J_k s_k = J_k v_k$ (recall that $J_k u_k = 0$ because of the constraint in (2.5)). The second result follows from Lemma 3.3, $\frac{1}{\alpha_k}\|s_k\|_2^2 \ge 0$, and the first result of this lemma. ∎

We now give a sufficient condition for a successful iteration (see (2.9)) to occur.

LEMMA 3.5. *If* $(1-\eta)\Delta q_k(s_k,\tau_k) \geq \frac{1}{2}(-\frac{\tau_k}{\alpha_k} + \tau_k L_g + L_J)\|s_k\|_2^2$*, then* $k \in \mathcal{S}$*.*

Proof. It follows from (3.4), (2.7), and the assumed inequality in this lemma that

$$\begin{aligned}
&\phi_{\tau_k}(x_k+s_k) - \phi_{\tau_k}(x_k) \\
&= \tau_k\big(f(x_k+s_k) + r(x_k+s_k)\big) + \|c(x_k+s_k)\|_2 - \tau_k\big(f_k + r_k\big) - \|c_k\|_2. \\
&\leq \tau_k g_k^T s_k + \tau_k\big(r(x_k+s_k) - r_k\big) + \|c_k + J_k s_k\|_2 - \|c_k\|_2 + \tfrac{1}{2}(\tau_k L_g + L_J)\|s_k\|_2^2 \\
&= -\Delta q_k(s_k,\tau_k) - \tfrac{\tau_k}{2\alpha_k}\|s_k\|_2^2 + \tfrac{1}{2}(\tau_k L_g + L_J)\|s_k\|_2^2 \\
&= -\Delta q_k(s_k,\tau_k) + \tfrac{1}{2}\big(-\tfrac{\tau_k}{\alpha_k} + \tau_k L_g + L_J\big)\|s_k\|_2^2 \leq -\eta\Delta q_k(s_k,\tau_k).
\end{aligned}$$

Therefore, it follows from Line 19 of Algorithm 2.1 that $k \in \mathcal{S}$, as claimed. ∎

The following result gives a bound on the decrease in linearized feasibility achieved by s_k that is similar to that achieved by the Cauchy point in (2.4).

LEMMA 3.6. *The step* $s_k = v_k + u_k$ *satisfies*

$$\|c_k\|_2 - \|c_k + J_k s_k\|_2 = \|c_k\|_2 - \|c_k + J_k v_k\|_2 \geq \tfrac{1}{2\kappa_c}\|J_k^T c_k\|_2^2 \min\left\{\tfrac{1}{1+\kappa_{\nabla c}^2}, \kappa_v\alpha_k\right\}.$$

Proof. From (1.2), (2.4), (2.2c), and the constraint in (2.5), we have

$$\begin{aligned}
&\tfrac{1}{2}\|J_k^T c_k\|_2^2 \min\left\{\tfrac{1}{1+\kappa_{\nabla c}^2}, \kappa_v\alpha_k\right\} \\
&\qquad \leq \tfrac{1}{2}\|J_k^T c_k\|_2^2 \min\left\{\tfrac{1}{1+\|J_k^T J_k\|_2}, \kappa_v\alpha_k\right\} \\
&\qquad \leq m_k(0) - m_k(v_k^c) = \tfrac{1}{2}\big(\|c_k\|_2^2 - \|c_k + J_k v_k^c\|_2^2\big) \\
&\qquad = \tfrac{1}{2}\big(\|c_k\|_2 + \|c_k + J_k v_k^c\|_2\big)\big(\|c_k\|_2 - \|c_k + J_k v_k^c\|_2\big) \\
&\qquad \leq \|c_k\|_2\big(\|c_k\|_2 - \|c_k + J_k v_k^c\|_2\big) \\
&\qquad \leq \kappa_c\big(\|c_k\|_2 - \|c_k + J_k v_k\|_2\big) = \kappa_c\big(\|c_k\|_2 - \|c_k + J_k s_k\|_2\big),
\end{aligned}$$

from which the desired result follows. ∎

We now begin investigating quantities related to the merit parameter. The following result bounds the denominator in the definition of $\tau_{k,\text{trial}}$.

LEMMA 3.7. *For all* $k \in \mathbb{N}$*, it follows that*

$$g_k^T s_k + \tfrac{\bar\sigma_u\|s_k\|_2^2}{\alpha_k} + r(x_k+s_k) - r_k \leq (\kappa_{\nabla f} + \kappa_{\partial r})\|v_k\|_2 + \tfrac{\bar\sigma_u\|v_k\|_2^2}{\alpha_k}.$$

Proof. With $g_{r,k}$ defined as in (3.1), it follows from convexity of r that $r_k \geq r(x_k+s_k) + g_{r,k}^T(-s_k)$. Combining this inequality with $s_k = v_k + u_k$, $v_k^T u_k = 0$, $\bar\sigma_u \in (\frac{1}{2}, 1]$, (3.2), the Cauchy-Schwartz inequality, and (1.2) it follows that

$$\begin{aligned}
&g_k^T s_k + \tfrac{\bar\sigma_u\|s_k\|_2^2}{\alpha_k} + r(x_k+s_k) - r_k \\
&\leq (g_k + g_{r,k})^T s_k + \tfrac{\bar\sigma_u\|s_k\|_2^2}{\alpha_k} \\
&= (g_k + g_{r,k})^T v_k + \tfrac{\bar\sigma_u\|v_k\|_2^2}{\alpha_k} + (g_k + g_{r,k})^T u_k + \tfrac{\bar\sigma_u\|u_k\|_2^2}{\alpha_k} \\
&\leq (g_k + g_{r,k})^T v_k + \tfrac{\bar\sigma_u\|v_k\|_2^2}{\alpha_k} + (g_k + g_{r,k})^T u_k + \tfrac{\|u_k\|_2^2}{\alpha_k} \\
&\leq \|g_k + g_{r,k}\|_2\|v_k\|_2 + \tfrac{\bar\sigma_u\|v_k\|_2^2}{\alpha_k}
\end{aligned}$$

$$\leq (\kappa_{\nabla f} + \kappa_{\partial r})\|v_k\|_2 + \tfrac{\bar{\sigma}_u \|v_k\|_2^2}{\alpha_k},$$

which completes the proof. ∎

We next show that the merit sequence is positive and monotonically decreasing.

LEMMA 3.8. *For all $k \geq 1$, it holds that $0 < \tau_k \leq \tau_{k-1}$.*

Proof. It is clear from $\tau_0 > 0$ and the update (2.8) that $\{\tau_k\}$ is monotonically decreasing, and therefore all that remains is to prove that $\tau_k > 0$ for all $k \in \mathbb{N}$. It follows from Lemma 3.7 and the definition of $\tau_{k,\text{trial}}$ that $\tau_{k,\text{trial}} = \infty$ if $v_k = 0$, and so for such k we have $\tau_k \leftarrow \tau_{k-1}$. Therefore, for the remainder we only need to consider $k \in \mathbb{N}$ such that $v_k \neq 0$. For such $k \in \mathbb{N}$, we know from Lemma 3.1 that $J_k^T c_k \neq 0$. The result $\tau_k > 0$ follows from this observation, (2.8), $\alpha_k > 0$, and Lemma 3.6. ∎

The first part of the next lemma shows that the merit parameter never needs to be decreased for iterations $k \in \mathbb{N}$ such that $J_k^T c_k = 0$. On the other hand, for all $k \in \mathbb{N}$ such that $J_k^T c_k \neq 0$, the second part of the result gives a lower bound on how small the previous merit parameter could have been.

LEMMA 3.9. *The following merit parameter update holds for each $k \in \mathbb{N} \setminus \{0\}$.*

(i) If $J_k^T c_k = 0$, then $\tau_{k,trial} = \infty$ and $\tau_k \leftarrow \tau_{k-1}$.

(ii) There exists a constant $\epsilon_\tau > 0$ such that, for all $k \in \mathbb{N}$ satisfying $J_k^T c_k \neq 0$ and $\tau_k < \tau_{k-1}$, it holds that $\tau_{k-1} \geq \epsilon_\tau \|J_k^T c_k\|_2$.

Proof. For part (i), it follows from $J_k^T c_k = 0$ and Lemma 3.1 that $v_k = 0$. This fact, Lemma 3.7, and the definition of $\tau_{k,\text{trial}}$ show that $\tau_{k,\text{trial}} = \infty$, so that $\tau_k \leftarrow \tau_{k-1}$.

For part (ii), it follows from (2.8), Lemma 3.6, Lemma 3.7, the trust-region constraint in problem (2.1), and (1.2) that if $\tau_k < \tau_{k-1}$, then

$$\begin{aligned}
\tau_{k-1} &> \frac{(1-\sigma_c)(\|c_k\|_2 - \|c_k + J_k v_k\|_2)}{g_k^T s_k + \frac{\bar{\sigma}_u \|s_k\|_2^2}{\alpha_k} + r(x_k + s_k) - r_k} \\
&\geq \frac{\frac{(1-\sigma_c)}{2\kappa_c}\|J_k^T c_k\|_2^2 \min\{\frac{1}{1+\kappa_{\nabla c}^2}, \kappa_v \alpha_k\}}{(\kappa_{\nabla f} + \kappa_{\partial r})\|v_k\|_2 + \frac{\bar{\sigma}_u \|v_k\|_2^2}{\alpha_k}} \\
&\geq \frac{(1-\sigma_c)\|J_k^T c_k\|_2^2 \min\{\frac{1}{1+\kappa_{\nabla c}^2}, \kappa_v \alpha_k\}}{2\kappa_c(\kappa_{\nabla f} + \kappa_{\partial r})\kappa_v \alpha_k \|J_k^T c_k\|_2 + \frac{\bar{\sigma}_u \kappa_v^2 \alpha_k^2 \|J_k^T c_k\|_2^2}{\alpha_k}} \\
&= \frac{(1-\sigma_c)\|J_k^T c_k\|_2 \min\{\frac{1}{1+\kappa_{\nabla c}^2}, \kappa_v \alpha_k\}}{2\kappa_c(\kappa_{\nabla f} + \kappa_{\partial r})\kappa_v \alpha_k + \bar{\sigma}_u \kappa_v^2 \alpha_k \|J_k^T c_k\|_2} \\
&\geq \frac{(1-\sigma_c)\|J_k^T c_k\|_2 \min\{\frac{1}{1+\kappa_{\nabla c}^2}, \kappa_v \alpha_k\}}{2\kappa_c(\kappa_{\nabla f} + \kappa_{\partial r})\kappa_v \alpha_k + \bar{\sigma}_u \kappa_v^2 \alpha_k \kappa_{\nabla c} \kappa_c}.
\end{aligned} \tag{3.5}$$

It follows from (3.5) and the fact that $\{\alpha_k\}$ is monotonically nonincreasing that

$$\tau_{k-1} \geq \begin{cases} \frac{(1-\sigma_c)\|J_k^T c_k\|_2}{2\kappa_c(\kappa_{\nabla f} + \kappa_{\partial r}) + \bar{\sigma}_u \kappa_v \kappa_{\nabla c} \kappa_c} & \text{if } \kappa_v \alpha_k \leq 1/(1+\kappa_{\nabla c}^2), \\ \frac{(1-\sigma_c)\|J_k^T c_k\|_2}{2\kappa_c(1+\kappa_{\nabla c}^2)(\kappa_{\nabla f} + \kappa_{\partial r})\kappa_v \alpha_0 + \bar{\sigma}_u \kappa_v^2 \alpha_0 \kappa_{\nabla c} \kappa_c} & \text{otherwise,} \end{cases}$$

which completes the proof. ∎

We now prove our first key convergence result. In particular, we prove that there must exist a subsequence of the set of successful iterations over which $\{J_k^T c_k\}$

converges to zero. This conclusion is relevant to our setting because, under a suitable constraint qualification, if $\bar{x}$ is a local minimizer of $\frac{1}{2}\|c(x)\|_2^2$, then $J(\bar{x})^T c(\bar{x}) = 0$.

THEOREM 3.10. *Let Assumption 1.1 hold. Then, there exists a subsequence of the iterations $\mathcal{K} \subseteq \mathbb{N}$ such that $\lim_{k\in\mathcal{K}} J_k^T c_k = 0$.*

Proof. For a proof by contradiction, suppose that there exists a $\bar{k}_1 \in \mathbb{N}$ and $\epsilon > 0$ such that $\|J_k^T c_k\|_2 \geq \epsilon$ for all $k \geq \bar{k}_1$. Then, it follows from Lemma 3.9 and $\tau_0 \in \mathbb{R}_{>0}$ that there exits $\bar{\tau}_1 > 0$ such that, for all $k \in \mathbb{N}$, it holds that $\tau_k \geq \bar{\tau}_1$. Moreover, since $\{\tau_k\}$ is monotonically nonincreasing and when $\tau_k < \tau_{k-1}$ the reduction is by at least a constant factor (see (2.8)), we know that there exists $\bar{k}_2 \geq \bar{k}_1$ and $\bar{\tau}_2 \geq \bar{\tau}_1$ such that $\tau_k = \bar{\tau}_2$ for all $k \geq \bar{k}_2$. Combining this with $\Delta q_k(s_k, \tau_k) > 0$ (see Lemma 3.4) and Lemma 3.5 it follows that for all $k \geq \bar{k}_2$ such that $\alpha_k \leq \bar{\tau}_2/(\bar{\tau}_2 L_g + L_J)$ it must also hold that $k \in \mathcal{S}$. Since $\alpha_{k+1} < \alpha_k$ only when $k \notin \mathcal{S}$, it follows that there must exist $\bar{\alpha} \in \mathbb{R}_{>0}$ and $\bar{k}_3 \geq \bar{k}_2$ such that $\alpha_k = \bar{\alpha}$ for all $k \geq \bar{k}_3$. To summarize, we have proved that for all $k \geq \bar{k}_3$ it holds that $\alpha_k = \bar{\alpha}$, $\tau_k = \bar{\tau}_2$, and $k \in \mathcal{S}$. It now follows from line 19 of Algorithm 2.1 that $\Phi_{\bar{\tau}_2}(x_{k+1}) \leq \Phi_{\bar{\tau}_2}(x_k) - \eta\Delta q_k(s_k, \bar{\tau}_2)$ for all $k \geq \bar{k}_3$. Summing over all $k \geq \bar{k}_3$ and using (1.2) and Lemma 3.4 we have

$$
\begin{aligned}
\Phi_{\bar{\tau}_2}(x_{\bar{k}_3}) - \bar{\tau}_2 f_{\inf} &\geq \sum_{k\geq\bar{k}_3} \big(\Phi_{\bar{\tau}_2}(x_k) - \Phi_{\bar{\tau}_2}(x_{k+1})\big) \\
&\geq \eta \sum_{k\geq\bar{k}_3} \Delta q_k(s_k, \bar{\tau}_2) \\
&\geq \eta \sum_{k\geq\bar{k}_3} \tfrac{\sigma_u \tau_k}{\alpha_k}\|s_k\|_2^2 + \sigma_c\big(\|c_k\|_2 - \|c_k + J_k v_k\|_2\big) \\
&> \eta\sigma_c \sum_{k\geq\bar{k}_3} \big(\|c_k\|_2 - \|c_k + J_k v_k\|_2\big).
\end{aligned}
$$

Since the summation of nonnegative terms is finite, we know that

$$\lim_{k\to\infty} \big(\|c_k\|_2 - \|c_k + J_k v_k\|_2\big) = 0.$$

This fact, Lemma 3.6, and $\alpha_k = \bar{\alpha}$ for all $k \geq \bar{k}_3$ imply that $\lim_{k\to\infty} J_k^T c_k = 0$, which contradicts our earlier assumption that $\|J_k^T c_k\|_2 \geq \epsilon$ for all $k \geq \bar{k}_1$. ∎

The remainder of the analysis considers two settings that are characterized by whether a certain constraint qualification holds or not.

3.2.1. Strong LICQ. In this section we make the following assumption, which is closely related to the linear independence constraint qualification (LICQ).

ASSUMPTION 3.1. *The smallest singular values of $\{J_k\}$ are uniformly bounded away from zero, i.e., there exists $\sigma_{\min} > 0$ such that, for all $k \in \mathbb{N}$, $\sigma_{\min}(J_k) \geq \sigma_{\min}$.*

We can now prove a nontrivial bound on the improvement in linearized infeasibility achieved by the trial step s_k relative to the actual infeasibility. This result is critical when we prove a uniform lower bound on the sequence of merit parameters.

LEMMA 3.11. *If $J_k^T c_k \neq 0$, then s_k satisfies $\|c_k + J_k s_k\|_2 \leq \rho_k \|c_k\|_2$ where*

$$\rho_k := \sqrt{\max\left\{1 - \kappa_v \alpha_k \sigma_{\min}^2, 1 - \sigma_{\min}^2/\kappa_{\nabla c}^2\right\}} \in [0, 1).$$

Proof. It follows from [30, Section 4.1] that the Cauchy step v_k^c in (2.3) satisfies

$$v_k^c = -\beta_k^c J_k^T c_k \text{ with } \beta_k^c = \min\left\{\frac{\|J_k^T c_k\|_2^2}{\|J_k J_k^T c_k\|_2^2}, \kappa_v \alpha_k\right\}. \tag{3.6}$$

We now consider two cases.
Case 1: $\|J_k^T c_k\|_2^2 \le \kappa_v \alpha_k \|J_k J_k^T c_k\|_2^2$. In this case, the minimum in (3.6) is the first term, and $J_k J_k^T c_k \neq 0$ since $J_k^T c_k \neq 0$. These facts, the inequality that defines this case, the Cauchy-Schwartz inequality, definition of $m_k(0)$, and Assumption 3.1 give

$$m_k(v_k^c) = m_k(0) - \tfrac{1}{2}\tfrac{\|J_k^T c_k\|_2^4}{\|J_k J_k^T c_k\|_2^2} \le \tfrac{1}{2}\|c_k\|_2^2 - \tfrac{1}{2}\tfrac{\|J_k^T c_k\|_2^2}{\kappa_{\nabla c}^2} \tag{3.7}$$

$$\le \tfrac{1}{2}\|c_k\|_2^2 - \tfrac{1}{2}\tfrac{\sigma_{\min}^2(J_k)}{\kappa_{\nabla c}^2}\|c_k\|_2^2 \le \tfrac{1}{2}\left(1 - \tfrac{\sigma_{\min}^2}{\kappa_{\nabla c}^2}\right)\|c_k\|_2^2. \tag{3.8}$$

Case 2: $\|J_k^T c_k\|_2^2 > \kappa_v \alpha_k \|J_k J_k^T c_k\|_2^2$. In this case, the minimum in (3.6) is the second term. This fact, the previous inequality, definition of $m_k(0)$, and Assumption 3.1 give

$$\begin{aligned}
m_k(v_k^c) &= m_k(0) - \kappa_v \alpha_k \|J_k^T c_k\|_2^2 + \tfrac{1}{2}\kappa_v^2 \alpha_k^2 \|J_k J_k^T c_k\|_2^2 \\
&\le m_k(0) - \kappa_v \alpha_k \|J_k^T c_k\|_2^2 + \tfrac{1}{2}\kappa_v \alpha_k \|J_k^T c_k\|_2^2 \\
&= m_k(0) - \tfrac{1}{2}\kappa_v \alpha_k \|J_k^T c_k\|_2^2 \le \tfrac{1}{2}\left(1 - \kappa_v \alpha_k \sigma_{\min}^2\right)\|c_k\|_2^2.
\end{aligned}$$

By combining the final result for the two cases, we find that $m_k(v_k^c) \le \frac{1}{2}\rho_k^2 \|c_k\|_2^2$. Multiplying both sides of this inequality by two, taking the square root, and using (2.2c) and the fact that $c_k + J_k s_k = c_k + J_k v_k$ since $J_k u_k = 0$, it follows that

$$\|c_k + J_k s_k\|_2 = \|c_k + J_k v_k\|_2 \le \|c_k + J_k v_k^c\|_2 \le \rho_k \|c_k\|_2,$$

which completes the proof. ∎

We may now prove that $\{\tau_k\}$ is bounded away from zero.

LEMMA 3.12. *For all $k \in \mathbb{N}$, it holds that $\tau_{k,trial} \ge \tau_{min,trial}$ with*

$$\tau_{min,trial} := \min\left\{\frac{(1-\sigma_c)\kappa_v \sigma_{\min}^2}{2\kappa_v \kappa_{\nabla c}(\kappa_{\nabla f} + \kappa_{\partial r} + \bar\sigma_u \kappa_c \kappa_v \kappa_{\nabla c})}, \frac{(1-\sigma_c)(\sigma_{\min}/\kappa_{\nabla c})^2}{2\kappa_v \kappa_{\nabla c}(\kappa_{\nabla f} + \kappa_{\partial r} + \bar\sigma_u \kappa_c \kappa_v \kappa_{\nabla c})\alpha_0}\right\},$$

which when combined with (3.9) gives $\tau_k \ge \tau_{\min} := \min\{\tau_0, (1-\epsilon_\tau)\tau_{min,trial}\}$.

Proof. We first prove a lower bound on $\tau_{k,\text{trial}}$. Since it follows that $\tau_{k,\text{trial}} = \infty$ for all $k \in \mathbb{N}$ satisfying $J_k^T c_k = 0$ (see Lemma 3.9(i)), we may assume without loss of generality that each $k \in \mathbb{N}$ satisfies $J_k^T c_k \neq 0$. Next, we see from Lemma 3.7, the trust-region constraint, and (1.2) that

$$\begin{aligned}
&g_k^T s_k + \tfrac{\bar\sigma_u \|s_k\|_2^2}{\alpha_k} + r(x_k + s_k) - r_k \\
&\le (\kappa_{\nabla f} + \kappa_{\partial r})\|v_k\|_2 + \tfrac{\bar\sigma_u \|v_k\|_2^2}{\alpha_k} \\
&= (\kappa_{\nabla f} + \kappa_{\partial r})\kappa_v \alpha_k \|J_k^T c_k\|_2 + \bar\sigma_u \kappa_v^2 \alpha_k \|J_k^T c_k\|_2^2 \\
&\le (\kappa_{\nabla f} + \kappa_{\partial r})\kappa_v \alpha_k \kappa_{\nabla c}\|c_k\|_2 + \bar\sigma_u \kappa_v^2 \alpha_k \kappa_{\nabla c}^2 \|c_k\|_2^2 \\
&\le (\kappa_{\nabla f} + \kappa_{\partial r})\kappa_v \alpha_k \kappa_{\nabla c}\|c_k\|_2 + \bar\sigma_u \kappa_c \kappa_v^2 \alpha_k \kappa_{\nabla c}^2 \|c_k\|_2 \\
&= \kappa_v \kappa_{\nabla c}(\kappa_{\nabla f} + \kappa_{\partial r} + \bar\sigma_u \kappa_c \kappa_v \kappa_{\nabla c})\alpha_k \|c_k\|_2 \quad \text{for all } k \in \mathbb{N}.
\end{aligned}$$

On the other hand, we may use Lemma 3.11 to obtain

$$\|c_k\|_2 - \|c_k + J_k v_k\|_2 \geq \|c_k\|_2 - \rho_k \|c_k\|_2 = (1-\rho_k)\|c_k\|_2 \quad \text{for all } k \in \mathbb{N}.$$

Using the above two bounds and the definition of $\tau_{k,\text{trial}}$, it follows that

$$\begin{aligned}\tau_{k,\text{trial}} &\geq \frac{(1-\sigma_c)(1-\rho_k)\|c_k\|_2}{\kappa_v \kappa_{\nabla c}\left(\kappa_{\nabla f} + \kappa_{\partial r} + \bar{\sigma}_u \kappa_c \kappa_v \kappa_{\nabla c}\right)\alpha_k \|c_k\|_2} \\ &= \frac{(1-\sigma_c)(1-\rho_k)}{\kappa_v \kappa_{\nabla c}\left(\kappa_{\nabla f} + \kappa_{\partial r} + \bar{\sigma}_u \kappa_c \kappa_v \kappa_{\nabla c}\right)\alpha_k} \quad \text{for all } k \in \mathbb{N}.\end{aligned}$$

Next, notice that it follows from the definition of ρ_k that

$$\begin{aligned}1 - \rho_k = \tfrac{1-\rho_k^2}{1+\rho_k} &\geq \tfrac{1-\max\{1-\kappa_v\alpha_k\sigma_{\min}^2, 1-\sigma_{\min}^2/\kappa_{\nabla c}^2\}}{2} \\ &= \tfrac{1-\left(1-\min\{\kappa_v\alpha_k\sigma_{\min}^2, \sigma_{\min}^2/\kappa_{\nabla c}^2\}\right)}{2} \\ &= \tfrac{\min\{\kappa_v\alpha_k\sigma_{\min}^2, \sigma_{\min}^2/\kappa_{\nabla c}^2\}}{2} \quad \text{for all } k \in \mathbb{N}.\end{aligned}$$

Combining this result with the previous displayed equation shows that

$$\tau_{k,\text{trial}} \geq \frac{(1-\sigma_c)\min\{\kappa_v\alpha_k\sigma_{\min}^2, (\sigma_{\min}/\kappa_{\nabla c})^2\}}{2\kappa_v\kappa_{\nabla c}\left(\kappa_{\nabla f} + \kappa_{\partial r} + \bar{\sigma}_u\kappa_c\kappa_v\kappa_{\nabla c}\right)\alpha_k} \quad \text{for all } k \in \mathbb{N}.$$

It follows from this inequality and the fact that $\alpha_k \leq \alpha_0$ for all $k \in \mathbb{N}$ that

$$\tau_{k,\text{trial}} \geq \begin{cases} \frac{(1-\sigma_c)\kappa_v\sigma_{\min}^2}{2\kappa_v\kappa_{\nabla c}(\kappa_{\nabla f}+\kappa_{\partial r}+\bar{\sigma}_u\kappa_c\kappa_v\kappa_{\nabla c})} & \text{if } \kappa_v\alpha_k\sigma_{\min}^2 \leq (\sigma_{\min}/\kappa_{\nabla c})^2, \\ \frac{(1-\sigma_c)(\sigma_{\min}/\kappa_{\nabla c})^2}{2\kappa_v\kappa_{\nabla c}(\kappa_{\nabla f}+\kappa_{\partial r}+\bar{\sigma}_u\kappa_c\kappa_v\kappa_{\nabla c})\alpha_0} & \text{otherwise,} \end{cases}$$

for all $k \in \mathbb{N}$, which proves our first result.

The second result, namely the positive lower bound on $\{\tau_k\}$, follows from the first result, $\tau_0 \in \mathbb{R}_{>0}$, and (2.8), which completes the proof. ∎

The positive lower bound on $\{\tau_k\}$ lets us prove a positive lower bound on $\{\alpha_k\}$.

LEMMA 3.13. *If $\alpha_k \leq \tau_{\min}/(\tau_{\min}L_g + L_J)$, then $k \in \mathcal{S}$. Therefore,*

$$\alpha_k \geq \alpha_{\min} := \min\{\alpha_0, \xi\tau_{\min}/(\tau_{\min}L_g + L_J)\} > 0 \quad \textit{for all } k \in \mathbb{N}, \tag{3.9}$$

and a bound on the number of unsuccessful iterations is given by

$$|\{k : x_k \notin \mathcal{S}\}| \leq \max\left(0, \left\lceil \frac{\log\left(\frac{\tau_{\min}}{\alpha_0(\tau_{\min}L_g+L_J)}\right)}{\log(\xi)} \right\rceil\right). \tag{3.10}$$

Proof. Suppose that $k \in \mathbb{N}$ satisfies $\alpha_k \leq \tau_{\min}/(\tau_{\min}L_g + L_J)$. Then it follows from the definition of $\alpha_{\min}$, Lemma 3.12, and the fact that $\tau/(\tau L_g + L_J)$ is a monotonically increasing function on the nonnegative real line as a function of τ that

$$\alpha_k \leq \tau_{\min}/(\tau_{\min}L_g + L_J) \leq \tau_k/(\tau_k L_g + L_J),$$

which after rearrangement shows that $-\tau_k/\alpha_k + \tau_k L_g + L_J \leq 0$. It follows from this inequality, Lemma 3.4, and $\eta \in (0,1)$ that

$$(1-\eta)\Delta q_k(s_k, \tau_k) > 0 \geq \tfrac{1}{2}(-\tfrac{\tau_k}{\alpha_k} + \tau_k L_g + L_J)\|s_k\|_2^2,$$

which together with Lemma 3.5 shows that $k \in \mathcal{S}$, as claimed. We know from the result we just proved and the update strategy for $\{\alpha_k\}$ that the bound in (3.9) holds.

Finally, the first result we proved in this lemma and the updating strategy for $\{\alpha_k\}$ shows that the maximum number of unsuccessful iterations is the smallest nonnegative integer n_u such that $\xi^{n_u}\alpha_0 \leq \tau_{\min}/(\tau_{\min}L_g + L_J)$, which gives the final result. ∎

Our worst-case complexity result uses the KKT-residual measure

$$\chi_k := \max\{\|g_k + g_{r,k} - J_k^T y_k\|_2, \|c_k\|_2\}, \tag{3.11}$$

where we remind the reader that $g_{r,k}$ is given in (3.1). In proving our complexity result, it will be convenient to define the shifted merit function

$$\bar{\phi}_\tau(x) := \tau\big(f(x) - f_{\inf} + r(x)\big) + \|c(x)\|_2,$$

where $f_{\inf}$ is defined in Assumption 1.2. We stress that the (typically) unknown value $f_{\inf}$ is never used in the algorithm statement or its implementation, only in our analysis. The following results pertain to the shifted merit function.

Lemma 3.14. *The following properties hold for the shifted merit function $\bar{\phi}_\tau$:*
(i) For all $\{x, y\} \subset \mathbb{R}^n$ and $\tau \in \mathbb{R}_{>0}$, it holds that $\bar{\phi}_\tau(x) - \bar{\phi}_\tau(y) = \phi_\tau(x) - \phi_\tau(y)$.
(ii) For all $x \in \mathbb{R}^n$ and $0 < \tau_2 \leq \tau_1$, it holds that $\bar{\phi}_{\tau_2}(x) \leq \bar{\phi}_{\tau_1}(x)$.
(iii) The sequence $\{\bar{\phi}_{\tau_k}(x_k)\}$ is monotonically decreasing.

Proof. For part (i), it follows from the definitions of $\bar{\phi}_\tau$ and ϕ_τ that

$$\begin{aligned}
\bar{\phi}_\tau(x) - \bar{\phi}_\tau(y) &= \tau\big(f(x) - f_{\inf} + r(x)\big) + \|c(x)\|_2 - \tau\big(f(y) - f_{\inf} + r(y)\big) - \|c(y)\|_2 \\
&= \tau\big(f(x) + r(x)\big) + \|c(x)\|_2 - \tau\big(f(y) + r(y)\big) - \|c(y)\|_2 \\
&= \phi_\tau(x) - \phi_\tau(y),
\end{aligned}$$

which proves part (i). For (ii), the definition of $f_{\inf}$ and nonnegativity of r imply that

$$\bar{\phi}_{\tau_2}(x) = \tau_2\big(f(x) - f_{\inf} + r(x)\big) + \|c(x)\|_2 \leq \tau_1\big(f(x) - f_{\inf} + r(x)\big) + \|c(x)\|_2 = \bar{\phi}_{\tau_1}(x),$$

which proves (ii). Finally, for each $k \in \mathbb{N}$, it follows from Lemma 3.8, parts (i) and (ii) of the current lemma, and how x_{k+1} is updated in Algorithm 2.1 that

$$\bar{\phi}_{\tau_k}(x_k) - \bar{\phi}_{\tau_{k+1}}(x_{k+1}) \geq \bar{\phi}_{\tau_k}(x_k) - \bar{\phi}_{\tau_k}(x_{k+1}) = \phi_{\tau_k}(x_k) - \phi_{\tau_k}(x_{k+1}) \geq 0,$$

which completes the proof of this theorem. ∎

We may now state our worst-case complexity result for Algorithm 2.1.

Theorem 3.15. *Suppose that Assumption 1.1 and Assumption 3.1 hold, and let $\epsilon \in \mathbb{R}_{>0}$ be given. If $\{k_1, k_2\} \subset \mathbb{N}$ are two iterations with $k_1 < k_2$ such that $k \in \mathcal{S}$ and $\chi_k > \epsilon$ for all iterations $k_1 \leq k < k_2$, then it follows that*

$$k_2 - k_1 \leq \left\lfloor \frac{\tau_0\big(f(x_0) - f_{inf} + r(x_0)\big) + \|c(x_0)\|_2}{\kappa_\Phi \epsilon^2} \right\rfloor \tag{3.12}$$

with $\kappa_\Phi := \eta \min\{\sigma_u \tau_{\min}\alpha_{\min}, \frac{\sigma_c\sigma_{\min}^2}{2\kappa_c(1+\kappa_{\nabla c}^2)}, \frac{\sigma_c\sigma_{\min}^2\kappa_v\alpha_{\min}}{2\kappa_c}\}$. Moreover, the maximum number of iterations before $\chi_k \leq \epsilon$ for some iteration $k \in \mathbb{N}$ is

$$\left(\max\left\{0, \left\lceil \frac{\log\left(\frac{\tau_{\min}}{\alpha_0(\tau_{\min}L_g + L_J)}\right)}{\log(\xi)} \right\rceil\right\} + 1\right) \left\lfloor \frac{\tau_0\big(f(x_0) - f_{inf} + r(x_0)\big) + \|c(x_0)\|_2}{\kappa_\Phi \epsilon^2} \right\rfloor. \tag{3.13}$$

Proof. Let $\{k_1, k_2\} \subset \mathbb{N}$ be as described in the theorem statement. Then, it follows from Lemma 3.8, Lemma 3.14(i–ii), Line 19 of Algorithm 2.1, Lemma 3.4, Lemma 3.12, and (3.9) that the following inequalities hold for all $k_1 \le k < k_2$:

$$\begin{aligned}\bar\phi_{\tau_k}(x_k) - \bar\phi_{\tau_{k+1}}(x_{k+1}) &\ge \bar\phi_{\tau_k}(x_k) - \bar\phi_{\tau_k}(x_{k+1})\\ &= \Phi_{\tau_k}(x_k) - \Phi_{\tau_k}(x_{k+1})\\ &\ge \eta\Delta q_k(s_k,\tau_k)\\ &\ge \eta\tfrac{\sigma_u\tau_k}{\alpha_k}\|s_k\|_2^2 + \eta\sigma_c\big(\|c_k\|_2 - \|c_k + J_k v_k\|_2\big)\\ &\ge \eta\sigma_u\tau_{\min}\alpha_{\min}\big(\tfrac{\|s_k\|_2}{\alpha_k}\big)^2 + \eta\sigma_c\big(\|c_k\|_2 - \|c_k + J_k v_k\|_2\big).\end{aligned}$$

Combining this inequality with $s_k = v_k + u_k$ and $v_k^T u_k = 0$ for all $k \in \mathbb{N}$, Lemma 3.6, Lemma 3.13, (3.1), and Assumption 3.1 it follows, for all $k_1 \le k < k_2$, that

$$\begin{aligned}&\bar\phi_{\tau_k}(x_k) - \bar\phi_{\tau_{k+1}}(x_{k+1})\\ &\ge \eta\sigma_u\tau_{\min}\alpha_{\min}\big(\tfrac{\|u_k\|_2}{\alpha_k}\big)^2 + \eta\sigma_c\tfrac{1}{2\kappa_c}\|J_k^T c_k\|_2^2\min\{(1/(1+\kappa_{\nabla c}^2),\kappa_v\alpha_{\min}\}\\ &\ge \eta\sigma_u\tau_{\min}\alpha_{\min}\|g_k + g_{r,k} - J_k^T y_k\|_2^2 + \eta\sigma_c\tfrac{\sigma_{\min}^2}{2\kappa_c}\|c_k\|_2^2\min\{(1/(1+\kappa_{\nabla c}^2),\kappa_v\alpha_{\min}\}\\ &\ge \kappa_\Phi\chi_k^2,\end{aligned}$$

where κ_Φ is defined in the theorem statement. Using this inequality, Lemma 3.14(iii), and nonnegativity of $\bar\phi_\tau$ for all $\tau \in \mathbb{R}_{>0}$, we find that

$$\begin{aligned}\bar\phi_{\tau_0}(x_0) &\ge \bar\phi_{\tau_{k_1}}(x_{k_1}) \ge \bar\phi_{\tau_{k_1}}(x_{k_1}) - \bar\phi_{\tau_{k_2}}(x_{k_2})\\ &\ge \sum_{k=k_1}^{k_2-1}\big(\bar\phi_{\tau_k}(x_k) - \bar\phi_{\tau_{k+1}}(x_{k+1})\big) \ge \sum_{k=k_1}^{k_2-1}\kappa_\Phi\chi_k^2,\end{aligned}$$

which may then be combined with the fact that $\chi_k > \epsilon$ for all iterations $k_1 \le k \le k_2$ (see the assumptions of the current theorem) to conclude that

$$\bar\phi_{\tau_0}(x_0) \ge (k_2 - k_1)\kappa_\Phi\epsilon^2,$$

from which (3.12) follows.

The final result in the theorem, namely the claimed upper bound on the maximum iterations before $\chi_k \le \epsilon$, follows from what we just proved and the fact that maximum number of unsuccessful iterations is bounded as in (3.10). ∎

Before proving a result concerning convergence to a KKT point, we need to prove that the Lagrange multiplier estimates generated by subproblem (2.5) are bounded.

LEMMA 3.16. *The Lagrange multiplier estimate sequence $\{y_k\}$ is bounded.*

Proof. Note from (3.2) and the Cauchy-Schwarz and triangle inequalities that

$$\tfrac{1}{\alpha_k}\|u_k\|_2^2 = -(g_k + g_{r,k})^T u_k \le \|g_k + g_{r_k}\|_2\|u_k\|_2 \le (\|g_k\|_2 + \|g_{r_k}\|_2)\|u_k\|_2,$$

which when combined with (1.2) shows that

$$\tfrac{1}{\alpha_k}\|u_k\|_2 \le \kappa_{\nabla f} + \kappa_{\partial r}. \tag{3.14}$$

Also observe that it follows from (3.1) and Assumption 3.1 that

$$J_k^T y_k = g_k + \tfrac{1}{\alpha_k}u_k + g_{r,k} \iff y_k = (J_k J_k^T)^{-1}J_k\big(g_k + \tfrac{1}{\alpha_k}u_k + g_{r,k}\big). \tag{3.15}$$

Combining (3.15), Assumption 3.1, the triangle inequality, and (3.14) it follows that

$$
\begin{aligned}
\|y_k\|_2 &\leq \tfrac{1}{\sigma_{\min}} \|g_k + \tfrac{1}{\alpha_k} u_k + g_{r,k}\|_2 \\
&\leq \tfrac{1}{\sigma_{\min}} \Big(\kappa_{\nabla f} + \kappa_{\partial r} + \tfrac{1}{\alpha_k} \|u_k\|_2 \Big) \\
&\leq \tfrac{2}{\sigma_{\min}} \Big(\kappa_{\nabla f} + \kappa_{\partial r} \Big).
\end{aligned}
$$

Since this result holds for arbitrary $k \in \mathbb{N}$, we have proved the result. ∎

We can now prove that limit points of the primal sequence are KKT points.

THEOREM 3.17. *Let Assumption 1.1 and Assumption 3.1 hold. Any limit point x_* of the sequence $\{x_k\}$ is a first-order KKT point for problem (1.1), i.e., $c(x_*) = 0$ and there exist vectors $y_* \in \mathbb{R}^m$ and $g_{r,*} \in \partial r(x_*)$ such that $g(x_*) + g_{r,*} - J(x_*)^T y_* = 0$.*

Proof. Let x_* be a limit point of $\{x_k\}$, i.e., there exists $\mathcal{K}_1$ so that $\{x_k\}_{k \in \mathcal{K}_1} \to x_*$. Theorem 3.15 allows us to conclude that there exists a subsequence $\mathcal{K}_2 \subseteq \mathcal{K}_1$ so that

$$
0 = \lim_{k \in \mathcal{K}_2} \chi_k = \lim_{k \in \mathcal{K}_2} \max\{\|g_k + g_{r,k} - J_k^T y_k\|_2, \|c(x_k)\|_2\}. \tag{3.16}
$$

Lemma 3.16 allows us to assert the existence of a vector $y_* \in \mathbb{R}^m$ and subsequence $\mathcal{K}_3 \subseteq \mathcal{K}_2$ such that $\{y_k\}_{k \in \mathcal{K}_3} = y_*$. It follows from this limit, $\{x_k\}_{k \in \mathcal{K}_3} \to x_*$, continuity of g and J, and (3.16) that

$$
\lim_{k \in \mathcal{K}_3} g_{r,k} = \lim_{k \in \mathcal{K}_3} (-g_k + J_k^T y_k) = -g(x_*) + J(x_*)^T y_* =: g_{r,*}.
$$

Finally, combining this equality with $\{x_k\}_{k \in \mathcal{K}_3} \to x_*$, continuity of c, and (3.16) it follows that $g(x_*) + g_{r,*} - J(x_*)^T y_* = 0$ and $c(x_*) = 0$, which completes the proof. ∎

3.2.2. Strong LICQ fails. In this section we prove properties of the iterate sequence $\{x_k\}$ in Algorithm 2.1 when the strong LICQ assumption used in the previous section (see Assumption 3.1) does not hold. In such a setting, we should expect to prove weaker results since, for example, Lagrange multipliers may not even exist.

Our main theorem of this section uses the quantity

$$
\bar{\chi}_k := \max\{\|g_k + g_{r,k} - J_k^T y_k\|_2, \|J_k^T c_k\|_2\}, \tag{3.17}
$$

which is related to the quanity χ_k used in the previous section (see (3.11)).

THEOREM 3.18. *Let Assumption 1.1 hold. One of the following two cases occurs.*

(i) There exists $\bar{\tau}_{\min} > 0$ such that $\tau_k \geq \bar{\tau}_{\min}$ for all $k \in \mathbb{N}$. In this case, it also follows that $\alpha_k \geq \bar{\alpha}_{\min} := \min\{\alpha_0, \xi \bar{\tau}_{\min}/(\bar{\tau}_{\min} L_g + L_J)\}$ for all $k \in \mathbb{N}$ and, for a given $\epsilon > 0$, the maximum number of iterations before $\bar{\chi}_k \leq \epsilon$ is

$$
\left(\max\left\{ 0, \left\lceil \frac{\log\left(\frac{\bar{\tau}_{\min}}{\alpha_0(\bar{\tau}_{\min} L_g + L_J)} \right)}{\log(\xi)} \right\rceil \right\} + 1 \right) \left\lfloor \frac{\tau_0\big(f(x_0) - f_{inf} + r(x_0)\big) + \|c(x_0)\|_2}{\bar{\kappa}_\Phi \epsilon^2} \right\rfloor
$$

where $\bar{\kappa}_\Phi := \eta \min\{\sigma_u \bar{\tau}_{\min} \bar{\alpha}_{\min}, \frac{\sigma_c}{2\kappa_c(1+\kappa_{\nabla c}^2)}, \frac{\sigma_c \kappa_v \bar{\alpha}_{\min}}{2\kappa_c}\}$.

(ii) The merit parameter values converge to zero, i.e., $\lim_{k \to \infty} \tau_k = 0$. In this case, there exists a subsequence $\mathcal{K} \subseteq \mathbb{N}$ such that $\lim_{k \in \mathcal{K}} \|J_k^T c_k\|_2 = 0$.

Proof. Let us start by considering part (i), in which case we know that there exists $\bar{\tau}_{\min} > 0$ such that $\tau_k \geq \bar{\tau}_{\min}$ for all $k \in \mathbb{N}$. Using this lower bound on $\{\tau_k\}$, the proof of Lemma 3.13 still holds (with $\tau_{\min}$ replaced by $\bar{\tau}_{\min}$), so that both (3.9) and (3.10) hold (with $\tau_{\min}$ replaced by $\bar{\tau}_{\min}$), thus proving the first claim on $\bar{\alpha}_{\min}$. Using (3.9) and (3.10) (with $\tau_{\min}$ replaced by $\bar{\tau}_{\min}$), the proof of Theorem 3.15 holds almost exactly as written. In particular, the proof holds as written until the middle of the second displayed equation, where we have (now with $\tau_{\min}$ and $\alpha_{\min}$ replace by $\bar{\tau}_{\min}$ and $\bar{\alpha}_{\min}$, respectively) that

$$\begin{aligned}&\bar{\phi}_{\tau_k}(x_k) - \bar{\phi}_{\tau_{k+1}}(x_{k+1})\\ &\quad\geq \eta\sigma_u\bar{\tau}_{\min}\bar{\alpha}_{\min}\|g_k + g_{r,k} - J_k^T y_k\|_2^2 + \eta\sigma_c\tfrac{1}{2\kappa_c}\|J_k^T c_k\|_2^2 \min\{(1/(1+\kappa_{\nabla c}^2), \kappa_v\bar{\alpha}_{\min}\}.\end{aligned}$$

If we now use the definitions of $\bar{\kappa}_\Phi$ and $\bar{\chi}_k$ we find that

$$\bar{\phi}_{\tau_k}(x_k) - \bar{\phi}_{\tau_{k+1}}(x_{k+1}) \geq \bar{\kappa}_\Phi\bar{\chi}_k^2.$$

The remainder of the proof of Theorem 3.15 now follows exactly as written but with $\bar{\chi}_k$ and $\bar{\kappa}_\Phi$ in place of χ_k and κ_Φ, respectively. This completes the proof of part (i).

Part (ii) follows from Theorem 3.10. ∎

A discussion on Theorem 3.18(i) is of interest. In particular, the result in Theorem 3.18(i) is of the same form as the result Theorem 3.15, with the only difference being the values of the constants $(\tau_{\min}, \alpha_{\min}, \kappa_\Phi)$ versus $(\bar{\tau}_{\min}, \bar{\alpha}_{\min}, \bar{\kappa}_\Phi)$. A consequence of Assumption 3.1 used in Section 3.2.1 is that we have an explicit definition for $\tau_{\min}$ (see Lemma 3.12), which implies an explicit lower bound on $\alpha_{\min}$ and κ_Φ (see Lemma 3.13 and Theorem 3.15). On the other hand, no explicit lower bound on $\bar{\tau}_{\min}$ is possible (in general) when Assumption 3.1 does not hold (in fact, it is even possible that $\{\tau_k\} \to 0$), and therefore the values for the constants $(\bar{\tau}_{\min}, \bar{\alpha}_{\min}, \bar{\kappa}_\Phi)$ in Theorem 3.18(i) will depend on the particular value of $\bar{\tau}_{\min}$ for that given problem. In this respect, the complexity result of Theorem 3.15 is stronger than Theorem 3.18(i), which is not surprising since Theorem 3.15 is proved under Assumption 3.1.

4. Numerical Results.

In this section, we present results of numerical experiments performed with our Python implementation of Algorithm 2.1. The test problems are formulated with an ℓ_1 regularizer, which is a common choice in many applications since it is known to induce sparse solutions. The goal of our numerical tests is to validate the overall performance of our method using standard optimization metrics and to evaluate its ability to correctly identify the zero-nonzero structure of a solution. For comparison purposes, we use the solver Bazinga [14], which is a safeguarded augmented Lagrangian method. The details concerning the test problems, our implementation, and the test results are given in the remainder of this section.

4.1. Test problems.

We considered a special instance of an ℓ_1-regularized objective function with equality constraints that can be written in the form

$$\min_{x\in\mathbb{R}^n, a\in\mathbb{R}^m} \quad f(x) + \lambda\|a\|_1 \quad \text{s.t.} \quad c(x) + a = 0 \tag{4.1}$$

for some chosen regularization parameter $\lambda \in \mathbb{R}_{>0}$. The functions f and c were chosen as a subset of the CUTEst [18] test problems, and we used PyCUTEst [16] to evaluate these functions in our Python code. Our *initial* test problems were chosen as the subset of CUTEst problems that satisfied the following properties: (i) the objective function was non-constant; (ii) the problem had at least one equality constraint, no inequality constraints, and no bound constraints on variables; and (iii) the number of equality constraints and variables satisfied $1 \leq m < n \leq 1000$. The restriction

$m < n$ rules out problems that essentially reduce to finding a feasible point for the constraints, while the restriction $n < 1000$ is used to keep the computational cost to a manageable level. As for the choice of λ, one can show that if $\bar{x}$ is a first-order KKT point with Lagrange multiplier vector $\bar{y}$ to the problem

$$\min_{x \in \mathbb{R}^n} \; f(x) \;\; \text{s.t.} \;\; c(x) = 0, \tag{4.2}$$

then $(\bar{x}, 0)$ is a first-order KKT point to problem (4.1) with Lagrange multiplier $\bar{y}$ as long as $\lambda \geq \|\bar{y}\|_\infty$. Therefore, in our tests, we set $\lambda = \|\bar{y}\|_\infty + 10$ where $\bar{y}$ is computed by solving problem (4.2) using IPOPT [33]. Since problems MSS1, MSS2, and CHAIN were not successfully solved by IPOPT, they were removed from the *initial* test set, thus resulting in the *final* set of 46 test problems found in Table A.1–Table A.2. Although the problem formulation 4.1 is somewhat contrived, this particular formulation allows us to better evaluate the structure identifying properties of the iterates produced by Algorithm 2.1 and Bazinga.

4.2. Implementation details. The parameter and input values used are presented in Table 4.1 (no fine-tuning was performed). As for the starting point (x_0, a_0) for problem (4.1), the vector x_0 is set to the default value supplied by CUTEst and the vector a_0 is set as $-c(x_0)$ so that the initial point (x_0, a_0) is feasible.

TABLE 4.1
Parameters and inputs used by Algorithm 2.1, with x_0 set to the value supplied by CUTEst.

α_0	τ_{-1}	κ_v	σ_c	ϵ_τ	ξ	η	σ_u
10	1	1000	0.1	0.1	0.5	10^{-4}	0.1

To approximately solve the trust-region subproblem (2.1), as needed in Line 5 of Algorithm 2.1, we used a Newton-like method. In particular, assuming for now that J_k had full row-rank, we first computed the minimizer of $m_k(v)$ over all $v \in \text{Range}(J_k^T)$. Using the relationship $v = J_k^T w$, this problem may be written as

$$\min_{w \in \mathbb{R}^m} \; \tfrac{1}{2}\|c_k\|_2^2 + w^T J_k J_k^T c_k + \tfrac{1}{2} w^T J_k J_k^T J_k J_k^T w.$$

It follows from the first-order optimality conditions and the full rank assumption on J_k that the unique solution, call it w_n, satisfies

$$J_k J_k^T J_k J_k^T w_n = -J_k J_k^T c_k \iff J_k J_k^T w_n = -c_k.$$

After solving this linear system for w_n, we have that $v_n = J_k^T w_n$. Next, we project this Newton step v_n onto the trust-region constraint by defining

$$\bar{v}_n := \min\{\|v_n\|_2, \kappa_v \alpha_k \|J_k^T c_k\|_2\} \frac{v_n}{\|v_n\|_2}.$$

Also accounting for the possibility that J_k may be rank deficient, we define v_k as

$$v_k \leftarrow \begin{cases} v_k^c & \text{if } J_k \text{ does not have full rank or } m_k(v_k^c) < m_k(\bar{v}_n) \\ \bar{v}_n & \text{otherwise,} \end{cases}$$

which by construction ensures that v_k satisfies conditions (2.2a)-(2.2c), as needed.

Next, to solve subproblem (2.5) (as needed in Line 12 of Algorithm 2.1) we exploit the structure of the ℓ_1-norm. By introducing variables $(p,q) \in \mathbb{R}^n_{\geq 0} \times \mathbb{R}^n_{\geq 0}$ and using e to denote the vector of all ones, we can consider the equivalent problem

$$\begin{aligned} \min_{u,p,q} \quad & g_k^T u + \tfrac{1}{2\alpha_k}\|u\|_2^2 + \lambda e^T(p+q) \\ \text{s.t.} \quad & J_k u = 0, \quad x_k + v_k + u = p - q, \quad p \geq 0, \quad q \geq 0, \end{aligned} \tag{4.3}$$

which is a convex quadratic program (QP). To solve subproblem (4.3) we use the primal active-set QP solver in the state-of-the-art software Gurobi version 11.0.0 [21]. Note that only a subset of the optimization variables receive ℓ_1 regularization in the test problem formulation (see (4.1)). This setting is handled using the above scheme by introducing p and q variables only for those variables appearing in the ℓ_1 norm.

Algorithm 2.1 was terminated when one of the following conditions was satisfied.

- **Approximate KKT point.** Algorithm 2.1 was terminated during the kth iteration with x_k considered an approximate KKT point if $\|c_k\|_2 \leq 10^{-6}$ and $\|g_k + g_{r,k} - J_k^T y_k\|_2 \leq 10^{-6}$, as motivated by (3.11) and Theorem 3.15.
- **Approximate infeasible stationary point.** Algorithm 2.1 was terminated during the kth iteration with x_k considered an approximate infeasible stationary point if $\|c_k\|_2 \geq 10^{-2}$ and $\|J_k^T c_k\|_2 \leq 10^{-12}$.
- **Gurobi error.** Algorithm 2.1 was terminated during the kth iteration if the Gurobi solver failed to solve subproblem (4.3) using its default tolerances.
- **Maximum iterations.** Algorithm 2.1 was terminated if 1000 iterations was completed without terminating for any of the previous reasons.

For comparison purposes, we solve the same test problems using the Bazinga method. Bazinga is a safeguarded augmented Lagrangian framework that uses an inner subproblem solver called PANOC$^+$, which is a linesearch algorithm that combines a forward-backward iteration and a quasi-Newton step over the forward-backward envelop of the objective function; see the Bazinga paper [14] for more details.[1] The Bazinga algorithm was terminated when one of the following conditions was satisfied.

- **Approximate KKT point.** Bazinga was terminated if certain primal feasibility and dual stationarity measures were less than 10^{-6}. Further details on the termination conditions of Bazinga can be found in [14, Section 3.3].
- **Not a number.** Bazinga was terminated if a NaN occurred.
- **Maximum iterations.** Bazinga was terminated if 100 iterations was completed without terminating for any of the previous reasons. Fewer maximum iterations was allowed for Bazinga compared to Algorithm 2.1 because each iteration of Bazinga is significantly more expensive compared to Algorithm 2.1. See the end of Section 4.3 and Appendix A for a discussion and table of results concerning computational times, respectively.

4.3. Test results. In this subsection, we present the results of using our Algorithm 2.1 and Bazinga to solve problems of the form (4.1) with test functions chosen as described in Section 4.1. To see detailed results for each test problem, see Table A.1 and Table A.2 in Appendix A. In the remainder of this section, we discuss the key results and observations summarized in Table 4.2.

We begin by describing the meanings of the columns of Table 4.2, and discuss their corresponding values to compare the performances of Algorithm 2.1 and Bazinga.

- **Method.** The name of the method.

[1] The code package of Bazinga is downloaded from https://github.com/aldma/Bazinga.jl.

- **Feasible.** The number of test problems for which the corresponding method terminated at a point with constraint violation no larger than 10^{-6}. For this metric we see that the two methods behaved similarly, with Bazinga achieving approximate feasibility on one more test problem.
- **Feasible, Better Objective.** To understand the meaning of this column, let $f_{\text{Algorithm 2.1}}$ denote the final objective value returned by Algorithm 2.1 and f_{Bazinga} denote the final objective value returned by Bazinga. We can then define the relative difference in the returned objective function values as

$$f_{\text{diff}} := \frac{f_{\text{Bazinga}} - f_{\text{Algorithm 2.1}}}{\max(1, |\min(f_{\text{Bazinga}}, f_{\text{Algorithm 2.1}})|)}. \tag{4.4}$$

  We indicate that Algorithm 2.1 (resp., Bazinga) had a better relative objective value if $f_{\text{diff}} \geq 10^{-5}$ (resp., $f_{\text{diff}} \leq -10^{-5}$). Using this terminology, column "Feasible, Better Objective" gives the number of test problems for which both algorithms terminated at a point with constraint violation less than 10^{-6} *and* the corresponding method had a better relative objective value. For this metric we see that Algorithm 2.1 significantly outperforms Bazinga in terms of final objective function values when both algorithms return vectors that satisfy the constraint violation tolerance.
- **Performs Better**. The number of test problems for which the corresponding method either (i) met the constraint violation tolerance and the other method did not, or (ii) both methods reached the constraint violation tolerance and the corresponding method had a better relative objective value (see (4.4)). Algorithm 2.1 significantly outperforms Bazinga on this metric.
- a **is Zero** The number of test problems for which the corresponding method returned $a = 0$. Algorithm 2.1 significantly outperformed Bazinga in this metric, with Algorithm 2.1 (resp., Bazinga) returning $a = 0$ on 36 (resp., 13) of the test problems. We conjecture that Bazinga's poor performance on this metric is due to its inner subproblem solver, which sacrifices solution sparsity for faster convergence of its iterates by combining proximal-gradient calculations with quasi-Newton ideas (see [14]). We investigated the test problems that Algorithm 2.1 did not return $a = 0$ and a Gurobi error was not encountered, and found that by increasing the regularization parameter, Algorithm 2.1 would return solutions satisfying $a = 0$.
- a **is Small.** The number of test problems for which the corresponding method returned $\|a\|_\infty \leq 10^{-5}$, thus indicating that a was small (possibly equal to zero). When comparing this column with column "a is Zero", we see that the only difference is that Algorithm 2.1 returned a small (nonzero) value for a on one additional test problem; the results for Bazinga were unchanged.
- **KKT Found.** The number of test problems for which the corresponding method terminated with an approximate KKT point, as discussed in Section 4.2. Algorithm 2.1 computed an approximate KKT point on 33 of the 46 test problems. Algorithm 2.1 encountered Gurobi errors (see Section 4.2) on test problems BT4 and HS56 that were related to large constraint violation values, which were caused by too large of an initial value for the merit parameter. These failures can be avoided by decreasing the initial value for the merit parameter, but we did not do that for the numerical tests presented.

Overall, we are pleased with the results of Table 4.2. We believe that they indicate that there is significant merit to our proposed algorithm, especially in terms

TABLE 4.2
Algorithm 2.1 versus Bazinga on various performance metrics related to solving problem (4.1) with test functions given in Table A.1–Table A.2; see Section 4.3 for the meaning of the columns.

Method	Feasible	Feasible, Better Objective	Performs Better	a is Zero	a is Small	KKT Found
Algorithm 2.1	40	23	23	36	37	33
Bazinga	41	2	7	13	13	21

of computing structured approximate solutions. It is worth noting that we have not discussed computational time since comparing our Python implementation of Algorithm 2.1 with the Julia implementation of Bazinga gives an advantage to Bazinga (purely because of the programming language used). Even still, one can observe from Table A.1 and Table A.2 that Algorithm 2.1 requires less (often significantly less) computing time compared to Bazinga on nearly every test problem.

5. Conclusion.

We have presented one of the first proximal-gradient type methods that can handle nonlinear equality constraints, and effectively return structured solutions where the structure is determined by the choice of regularization function. In the future, it would be interesting to address inequality constraints, establish convergence results under weaker assumptions, and accelerate convergence by incorporating Nesterov acceleration or subspace acceleration.

REFERENCES

[1] M. ANITESCU, *A superlinearly convergent sequential quadratically constrained quadratic programming algorithm for degenerate nonlinear programming*, SIAM J. Optim., 12 (2002), pp. 949–978.
[2] M. ANNERGREN, A. HANSSON, AND B. WAHLBERG, *An admm algorithm for solving l-1 regularized mpc*, in 2012 IEEE 51st IEEE Conference on Decision and Control (CDC), IEEE, 2012, pp. 4486–4491.
[3] A. BAGIROV, N. KARMITSA, AND M. M. MÄKELÄ, *Introduction to Nonsmooth Optimization: theory, practice and software*, vol. 12, Springer, 2014.
[4] A. BECK, *First-order methods in optimization*, SIAM, 2017.
[5] A. BECK AND M. TEBOULLE, *A fast iterative shrinkage-thresholding algorithm for linear inverse problems*, SIAM Journal on Imaging Sciences, 2 (2009), pp. 183–202.
[6] J.-F. CAI, E. J. CANDÈS, AND Z. SHEN, *A singular value thresholding algorithm for matrix completion*, SIAM Journal on optimization, 20 (2010), pp. 1956–1982.
[7] T. CHEN, F. E. CURTIS, AND D. P. ROBINSON, *A reduced-space algorithm for minimizing ℓ_1-regularized convex functions*, SIAM Journal on Optimization, 27 (2017), pp. 1583–1610.
[8] T. CHEN, F. E. CURTIS, AND D. P. ROBINSON, *FaRSA for ℓ_1-regularized convex optimization: local convergence and numerical experience*, Optimization Methods and Software, 33 (2018), pp. 396–415.
[9] A. R. CONN, N. I. M. GOULD, AND PH. L. TOINT, *A globally convergent augmented Lagrangian algorithm for optimization with general constraints and simple bounds*, SIAM J. Numer. Anal., 28 (1991), pp. 545–572.
[10] A. R. CONN, N. I. M. GOULD, AND PH. L. TOINT, *Trust-Region Methods*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2000.
[11] A. R. CONN AND T. PIETRZYKOWSKI, *A penalty function method converging directly to a constrained optimum*, SIAM J. Numer. Anal., 14 (1977), pp. 348–375.
[12] F. E. CURTIS, H. JIANG, AND D. P. ROBINSON, *An adaptive augmented lagrangian method for large-scale constrained optimization*, Math. Program., (2013), pp. 1–45.
[13] F. E. CURTIS, D. P. ROBINSON, AND B. ZHOU, *Inexact sequential quadratic optimization for minimizing a stochastic objective function subject to deterministic nonlinear equality constraints*, arXiv preprint arXiv:2107.03512, (2021).
[14] A. DE MARCHI, X. JIA, C. KANZOW, AND P. MEHLITZ, *Constrained composite optimization and augmented lagrangian methods*, Mathematical Programming, (2023), pp. 1–34.

[15] A. V. Fiacco and G. P. McCormick, *Nonlinear Programming*, Classics in Applied Mathematics, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, second ed., 1990. Reprint of the 1968 original.

[16] J. Fowkes, L. Roberts, and Á. Bűrmen, *Pycutest: an open source python package of optimization test problems*, Journal of Open Source Software, 7 (2022), p. 4377.

[17] M. Fukushima, *A successive quadratic programming method for a class of constrained nonsmooth optimization problems*, Mathematical programming, 49 (1990), pp. 231–251.

[18] N. I. M. Gould, D. Orban, and Ph. L. Toint, *CUTEst: a constrained and unconstrained testing environment with safe threads*, technical report, Rutherford Appleton Laboratory, Chilton, England, 2013, https://doi.org/10.1007/s10589-014-9687-3, http://dx.doi.org/10.1007/s10589-014-9687-3.

[19] N. I. M. Gould and D. P. Robinson, *A second derivative SQP method: Global convergence*, SIAM J. Optim., 20 (2010), pp. 2023–2048.

[20] N. I. M. Gould and D. P. Robinson, *A second derivative SQP method: Local convergence and practical issues*, SIAM J. Optim., 20 (2010), pp. 2049–2079.

[21] Gurobi Optimization, LLC, *Gurobi Optimizer Reference Manual*, 2023, https://www.gurobi.com.

[22] C. B. Gurwitz and M. L. Overton, *Sequential quadratic programming methods based on approximating a projected hessian matrix*, SIAM J. Sci. and Statist. Comput., 10 (1989), pp. 631–653.

[23] S.-P. Han, *A globally convergent method for nonlinear programming*, Journal of optimization theory and applications, 22 (1977), pp. 297–309.

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

[25] H. Karimi, J. Nutini, and M. Schmidt, *Linear convergence of gradient and proximal-gradient methods under the Polyak-Lojasiewicz condition*, in Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Springer, 2016, pp. 795–811.

[26] C.-p. Lee and S. J. Wright, *Inexact successive quadratic approximation for regularized optimization*, Computational Optimization and Applications, 72 (2019), pp. 641–674.

[27] Y.-F. Liu, S. Ma, Y.-H. Dai, and S. Zhang, *A smoothing sqp framework for a class of composite l q minimization over polyhedron*, Mathematical Programming, 158 (2016), pp. 467–500.

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

[29] J. L. Morales, J. Nocedal, and Y. Wu, *A sequential quadratic programming algorithm with an additional equality constrained phase*, IMA J. Numer. Anal., 32 (2012), pp. 553–579.

[30] J. Nocedal and S. Wright, *Numerical optimization*, Springer Science & Business Media, 2006.

[31] M. J. Powell, *A fast algorithm for nonlinearly constrained optimization calculations*, in Numerical Analysis: Proceedings of the Biennial Conference Held at Dundee, June 28–July 1, 1977, Springer, 2006, pp. 144–157.

[32] D. P. Robinson, *Primal-dual active-set methods for large-scale optimization*, Submitted to Journal of Optimization Theory and Applications, (2013).

[33] A. Wächter and L. T. Biegler, *On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming*, Mathematical programming, 106 (2006), pp. 25–57.

[34] N. Xiao, X. Liu, and Y.-x. Yuan, *A penalty-free infeasible approach for a class of nonsmooth optimization problems over the stiefel manifold*, arXiv preprint arXiv:2103.03514, (2021).

[35] G. Yuan and B. Ghanem, *\ell_0 tv: A sparse optimization method for impulse noise image restoration*, IEEE transactions on pattern analysis and machine intelligence, 41 (2017), pp. 352–364.

[36] V. M. Zavala and M. Anitescu, *Scalable nonlinear programming via exact differentiable penalty functions and trust-region newton methods*, SIAM Journal on Optimization, 24 (2014), pp. 528–558.

[37] D. Zhu, L. Zhao, and S. Zhang, *A first-order primal-dual method for nonconvex constrained optimization based on the augmented lagrangian*, Mathematics of Operations Research, (2023).

Appendix A. Detailed Results for the Test Problems.

In this appendix we provide the detailed output from our Algorithm 2.1 and Bazinga for the test problems in Table A.1 and Table A.2. See Section 4 for details on the problem formulation, the test functions used, and the implementation details.

The columns of Table A.1 and Table A.2 have the following meanings.

- **Problem.** The name of the test problem. Specifically, the value in this column gives the name of the CUTEst test problem used to obtain the objective function f and constraint function c in the test problem formulation (4.1).
- **Method.** The name of the method used.
- **Obj.** The value of the objective function in problem (4.1) at the final iterate returned by the solver.
- **RE.** The relative error between the objective function value returned by the algorithm and the optimal objective function value. In particular, if we let $(f+r)$ denote the objective function value returned by a solver on a particular problem and let $(f+r)_{\text{opt}}$ denote the optimal objective value for that same problem (as determined by the CUTEst documentation), then we define the relative error for that method on that problem as

$$\mathbf{RE} = \frac{|(f+r) - (f+r)_{\text{opt}}|}{\max(1, |(f+r)_{\text{opt}}|)}.$$

- $\|\boldsymbol{c(x)+a}\|_{\mathbf{2}}$**.** The value of $\|c(x)+a\|_2$ at the point returned by the solver.
- $\|\boldsymbol{a}\|_{\boldsymbol{\infty}}$**.** The value of $\|a\|_\infty$ at the point returned by the solver.
- **Status.** A three letter string that indicates the outcome when the given method was used to solve the given test problem. In particular, the value "Opt" means that the method returned a final iterate that was an **approximate KKT point** as described in Section 4.2. The value "Max" indicates that the method reached its maximum allowed number of iterations as described under **Maximum Iterations** in Section 4.2. The value "Err" only occurred for Algorithm 2.1 and indicates that a Gurobi error occurred as described under **Gurobi error** in Section 4.2. Finally, the value "NaN" only occurred for Bazinga and indicates that the data type not-a-number occurred.
- **Time.** The computational time measured in seconds.

Problem	Method	Obj	RE	$\|c(x)+a\|_2$	$\|a\|_\infty$	Status	Time
BT11	Alg. 2.1	8.25E-01	1.58E-07	1.01E-13	0.00E+00	Opt	1.71E+00
	Bazinga	2.22E+01	2.79E+01	8.23E-07	6.62E-01	Max	3.50E+02
BT12	Alg. 2.1	6.19E+00	3.15E-10	4.26E-10	0.00E+00	Opt	2.99E-01
	Bazinga	3.75E+02	5.95E+01	1.02E-07	1.24E+01	Max	3.33E+02
BT1	Alg. 2.1	-9.00E+01	8.90E+01	4.77E-13	1.00E+00	Opt	3.35E-01
	Bazinga	-5.54E+01	5.44E+01	3.68E-08	6.08E-01	Opt	9.37E+00
BT2	Alg. 2.1	1.02E+05	3.14E+06	2.24E-05	1.02E+04	Max	5.17E+00
	Bazinga	3.26E-02	2.83E-07	8.22E-07	0.00E+00	Opt	1.24E+01
BT3	Alg. 2.1	4.09E+00	3.10E-06	4.71E-15	0.00E+00	Opt	2.58E-01
	Bazinga	3.41E+01	7.33E+00	3.91E-09	5.92E-01	Max	3.39E+02
BT4	Alg. 2.1	-2.67E+31	5.87E+29	2.63E+21	1.84E+10	Err	3.24E-02
	Bazinga	4.00E+01	1.88E+00	1.22E-07	2.69E+00	Max	3.26E+02
BT5	Alg. 2.1	9.62E+02	7.32E-11	4.05E-10	0.00E+00	Opt	1.31E-01
	Bazinga	1.03E+03	7.60E-02	7.06E-08	3.11E+00	Max	3.26E+02
BT6	Alg. 2.1	2.77E-01	4.89E-07	1.82E-12	0.00E+00	Opt	6.57E-01
	Bazinga	2.65E+01	9.45E+01	1.32E-07	1.19E+00	Max	3.42E+02
BT7	Alg. 2.1	3.96E+01	8.71E-01	2.44E-13	4.54E-01	Opt	4.60E+00
	Bazinga	9.26E+02	2.02E+00	9.74E-07	2.50E-01	Opt	1.27E+02
BT8	Alg. 2.1	1.00E+00	2.61E-06	8.67E-13	2.37E-07	Max	7.83E+00
	Bazinga	1.00E+00	9.86E-10	1.31E-09	0.00E+00	Opt	9.57E+00
BT9	Alg. 2.1	-1.00E+00	1.05E-11	1.63E-11	0.00E+00	Opt	1.46E-01
	Bazinga	2.60E+01	2.70E+01	5.87E-07	1.24E+00	Max	3.23E+02
BYRDSPHR	Alg. 2.1	-4.68E+00	7.61E-08	6.55E-09	0.00E+00	Opt	7.89E-02
	Bazinga	6.27E+00	2.34E+00	2.23E-08	4.92E-01	Opt	2.06E+01
DIXCHLNG	Alg. 2.1	1.59E+02	1.59E+02	1.98E-07	9.70E-01	Max	1.55E+01
	Bazinga	NaN	NaN	NaN	NaN	NaN	1.35E+01
ELEC	Alg. 2.1	1.46E+04	2.08E-01	1.12E-06	2.97E+00	Max	2.24E+04
	Bazinga	1.58E+04	1.42E-01	8.22E-08	6.45E-01	Max	3.34E+04
EXTROSNBNE	Alg. 2.1	-2.00E+00	2.00E+00	1.86E-06	0.00E+00	Max	1.38E+05
	Bazinga	4.11E+03	4.11E+03	3.32E-07	4.12E-01	Max	3.65E+04
GENHS28	Alg. 2.1	9.27E-01	9.27E-01	2.47E-15	0.00E+00	Opt	2.07E+00
	Bazinga	9.27E-01	9.27E-01	1.01E-10	0.00E+00	Opt	9.35E+00
HS100LNP	Alg. 2.1	6.81E+02	1.09E-10	2.27E-13	0.00E+00	Opt	6.93E-01
	Bazinga	7.26E+02	1.16E+02	1.17E-07	1.46E+00	Max	3.36E+02
HS111LNP	Alg. 2.1	-4.78E+01	1.10E-03	4.99E-10	0.00E+00	Max	1.15E+01
	Bazinga	-5.22E+01	9.52E-02	7.12E-08	2.60E+00	Max	4.58E+02
HS26	Alg. 2.1	1.09E+00	1.09E+00	5.12E-08	0.00E+00	Max	5.27E+00
	Bazinga	1.05E-13	1.05E-13	2.93E-10	0.00E+00	Opt	9.14E+00
HS27	Alg. 2.1	4.00E-02	5.63E-10	9.48E-15	0.00E+00	Opt	3.48E-01
	Bazinga	4.00E-02	1.46E-07	1.46E-07	0.00E+00	Opt	9.12E+00
HS28	Alg. 2.1	5.05E-11	5.05E-11	8.88E-16	0.00E+00	Opt	2.26E-01
	Bazinga	7.95E-17	7.95E-17	8.01E-10	0.00E+00	Opt	9.05E+00
HS39	Alg. 2.1	-1.00E+00	1.05E-11	1.63E-11	0.00E+00	Opt	1.37E-01
	Bazinga	2.60E+01	2.70E+01	5.87E-07	1.24E+00	Max	3.25E+02

TABLE A.1
Results for test problems BT11-HS39.

Problem	Method	Obj	RE	$\|c(x)+a\|_2$	$\|a\|_\infty$	Status	Time (s)
HS40	Alg. 2.1	-2.50E-01	1.39E-10	5.91E-11	0.00E+00	Opt	1.05E-01
	Bazinga	1.43E+01	5.81E+01	9.97E-10	4.78E-01	Opt	1.25E+02
HS42	Alg. 2.1	1.39E+01	2.68E-10	3.06E-14	0.00E+00	Opt	2.68E-01
	Bazinga	2.76E+01	9.90E-01	5.91E-10	1.13E+00	Opt	9.38E+01
HS46	Alg. 2.1	5.73E-10	5.73E-10	4.09E-12	0.00E+00	Opt	1.84E+00
	Bazinga	NaN	NaN	NaN	NaN	NaN	3.09E+02
HS47	Alg. 2.1	1.33E-05	1.33E-05	4.70E-10	0.00E+00	Max	8.48E+00
	Bazinga	4.21E+01	4.21E+01	1.27E-10	1.36E+00	Opt	3.84E+01
HS48	Alg. 2.1	2.69E-11	2.69E-11	4.44E-16	0.00E+00	Opt	1.84E-01
	Bazinga	2.00E-13	2.00E-13	1.59E-10	0.00E+00	Opt	9.06E+00
HS49	Alg. 2.1	2.20E-04	2.20E-04	1.81E-13	0.00E+00	Max	7.22E+00
	Bazinga	2.90E+01	2.90E+01	2.02E-07	1.43E+00	Max	3.34E+02
HS50	Alg. 2.1	2.86E-11	2.86E-11	1.72E-14	0.00E+00	Opt	6.58E+00
	Bazinga	4.53E+00	4.53E+00	1.66E-09	7.65E-02	Max	3.33E+02
HS51	Alg. 2.1	1.48E-11	1.48E-11	6.28E-16	0.00E+00	Opt	1.71E-01
	Bazinga	1.61E-15	1.61E-15	3.17E-09	0.00E+00	Opt	9.05E+00
HS52	Alg. 2.1	5.33E+00	4.31E-12	7.69E-16	0.00E+00	Opt	1.37E+00
	Bazinga	5.33E+00	6.50E-07	2.98E-07	0.00E+00	Opt	9.15E+00
HS56	Alg. 2.1	-5.81E+83	1.68E+83	4.07E+28	4.94E+27	Err	6.57E-02
	Bazinga	NaN	NaN	NaN	NaN	NaN	7.44E+01
HS61	Alg. 2.1	-1.44E+02	1.55E-11	2.03E-13	0.00E+00	Opt	1.75E-01
	Bazinga	-1.44E+02	4.29E-09	4.48E-07	0.00E+00	Opt	9.10E+00
HS6	Alg. 2.1	1.16E-10	1.16E-10	2.38E-11	0.00E+00	Opt	4.11E-01
	Bazinga	4.46E-14	4.46E-14	2.49E-09	0.00E+00	Opt	1.20E+01
HS77	Alg. 2.1	2.42E-01	6.83E-09	3.16E-12	0.00E+00	Opt	5.32E-01
	Bazinga	NaN	NaN	NaN	NaN	NaN	1.12E+01
HS78	Alg. 2.1	1.10E+01	4.77E+00	4.73E-10	1.00E+00	Max	8.44E+00
	Bazinga	5.44E+01	1.96E+01	3.93E-09	1.74E+00	Max	3.44E+02
HS79	Alg. 2.1	7.88E-02	7.11E-10	6.50E-14	0.00E+00	Opt	3.51E+00
	Bazinga	2.82E+01	3.57E+02	3.74E-08	9.33E-01	Max	3.48E+02
HS7	Alg. 2.1	-1.73E+00	2.44E-10	8.18E-11	0.00E+00	Opt	1.81E+00
	Bazinga	-1.73E+00	2.12E-08	1.29E-07	0.00E+00	Opt	9.11E+00
HS9	Alg. 2.1	-5.00E-01	1.32E-09	1.78E-15	0.00E+00	Opt	7.98E-02
	Bazinga	-5.00E-01	9.02E-10	1.37E-08	0.00E+00	Opt	9.10E+00
LCH	Alg. 2.1	-1.23E+02	2.77E+01	3.42E-06	9.47E+00	Max	1.84E+03
	Bazinga	1.11E+00	1.26E+00	7.89E-09	3.50E-01	Max	1.07E+03
MARATOS	Alg. 2.1	-1.00E+00	2.00E+00	4.71E-10	0.00E+00	Opt	6.19E-02
	Bazinga	5.48E+00	4.48E+00	7.69E-11	6.10E-01	Opt	9.56E+00
MWRIGHT	Alg. 2.1	2.50E+01	2.40E-01	1.09E-13	0.00E+00	Opt	6.92E-01
	Bazinga	2.67E+02	7.11E+00	3.41E-07	3.02E+00	Max	3.46E+02
ORTHREGB	Alg. 2.1	2.09E-12	2.09E-12	9.39E-13	0.00E+00	Opt	5.19E-01
	Bazinga	3.90E+02	3.90E+02	6.22E-08	6.49E+00	Max	5.21E+02
S316-322	Alg. 2.1	3.34E+02	1.29E-06	3.31E-08	0.00E+00	Opt	6.14E-02
	Bazinga	9.12E+02	1.73E+00	3.26E-07	9.80E-01	Opt	1.59E+01
SPIN2OP	Alg. 2.1	1.23E-09	1.23E-09	1.22E-09	0.00E+00	Opt	6.55E+02
	Bazinga	2.09E+03	2.09E+03	2.02E-08	2.09E+00	Max	4.45E+03
STREGNE	Alg. 2.1	4.93E-12	4.93E-12	1.11E-15	0.00E+00	Opt	1.99E-01
	Bazinga	NaN	NaN	NaN	NaN	NaN	1.82E+01

TABLE A.2
Results for test problems HS40-STREGNE.