SIAM J. OPTIM.
Vol. 30, No. 3, pp. 2501–2529

A TRUST REGION METHOD FOR FINDING SECOND-ORDER STATIONARITY IN LINEARLY CONSTRAINED NONCONVEX OPTIMIZATION*

MAHER NOUIEHED† AND MEISAM RAZAVIYAYN‡

Abstract. Motivated by the TRACE algorithm [F. E. Curtis, D. P. Robinson, and M. Samadi, *Math. Program.*, 162 (2017), pp. 1–32], we propose a trust region algorithm for finding second-order stationary points of a linearly constrained nonconvex optimization problem. We show the convergence of the proposed algorithm to (ϵ_g, ϵ_H)-second-order stationary points in $\widetilde{\mathcal{O}}(\max\{\epsilon_g^{-3/2}, \epsilon_H^{-3}\})$ iterations. This iteration complexity is achieved for general linearly constrained optimization without cubic regularization of the objective function.

Key words. trust region, nonconvex optimization, linear constraints, second-order stationarity

AMS subject classifications. 90C26, 90C30

DOI. 10.1137/19M1256415

1. Introduction.

Due to its wide application in machine learning, solving nonconvex optimization problems encountered significant attention in recent years [1, 12, 14, 15, 18, 19, 30]. While this topic has been studied for decades, recent applications and modern analytical and computational tools revived this area of research. In particular, a wide variety of numerical methods for solving nonconvex problems have been proposed in recent years [28, 23, 33, 9, 10, 11, 36].

For general nonconvex optimization problems, it is well-known that computing a local optimum is NP-hard [32]. Given this hardness result, recent focus has been shifted toward computing (approximate) first- and second-order stationary points of the objective function. The latter set of points provides stronger guarantees compared to the former as it constitutes a smaller subset of points that includes local and global optima. Therefore, when applied to problems with "nice" geometrical properties, the set of second-order stationary points could even coincide with the set of global optima—see [2, 3, 8, 35, 38, 21, 39, 40] for examples of such objective functions.

Convergence to second-order stationarity in a smooth unconstrained setting has been thoroughly investigated in the optimization literature [22, 17, 10, 11, 33, 18, 19, 21]. As a second-order algorithm, [33] proposed a cubic regularization method that converges to approximate second-order stationarity in a finite number of steps. More recently, [10, 11] proposed the adaptive regularization cubic algorithm (ARC) that computes an approximate solution for a local cubic model at each iteration. They established convergence to first- and second-order stationary points with optimal complexity rates. Motivated by these rates, [19] proposed an adaptive trust region method, entitled TRACE, and established iteration complexity bounds for finding ϵ-first-order stationarity with worst-case iteration complexity $\mathcal{O}(\epsilon^{-3/2})$ and for finding (ϵ_g, ϵ_H)-second-order stationarity with worst-case complexity $\mathcal{O}(\max\{\epsilon_g^{-3/2}, \epsilon_H^{-3}\})$.

*Received by the editors April 15, 2019; accepted for publication (in revised form) June 8, 2020; published electronically September 16, 2020.
https://doi.org/10.1137/19M1256415

†Department of Industrial Engineering and Management, American University of Beirut, Lebanon (mn102@aub.edu.lb).
‡Department of Industrial and Systems Engineering, University of Southern California, Los Angeles, CA 90089 (razaviya@usc.edu).

This method altered the acceptance criteria adopted by traditional trust region methods and implemented a new mechanism for updating the trust region radius. A more recent second-order algorithm that uses a dynamic choice of direction and step-size was proposed in [17]. This method computes first- and second-order descent directions and chooses the direction that predicts a more significant reduction in the objective value. All of the above methods satisfy the set of generic conditions of a general framework proposed in [18].

Recent results show that for smooth unconstrained optimization problems, even first-order methods can converge to second-order stationarity, almost surely. For instance, [21] showed that noisy stochastic gradient descent escapes *strict saddle points* with probability one. Therefore, when applied to problems satisfying the *strict saddle property* this method converges to a local minimum. The property of escaping strict saddle points with probability one was also established for the vanilla gradient descent algorithm in [28]. A negative result provided by [20] shows that vanilla gradient descent can take an exponential number of steps to converge to second-order stationarity. This computational inefficiency can be overcome with high probability by a smart perturbed form of gradient descent proposed in [26].

Most of the above results can be extended to the smooth constrained optimization in the presence of simple manifold constraints. In this case, [27] showed that manifold gradient descent converges to second-order stationarity, almost surely. More recently, [23] established similar results for gradient primal-dual algorithms applied on linearly constrained optimization problems. When the constraints are of nonmanifold type, projected gradient descent is a natural replacement of gradient descent. As a negative result, [34] constructed an example, with a single linear constraint, showing that there is a positive probability that projected gradient descent with random initialization can converge to a strict saddle point. This raises the question of *whether there exists a first-order method that can converge to second-order stationarity in the presence of inequality constraints.* To our knowledge, no affirmative answer has been given to this question to date.

The answer to the question above is obvious when replacing first-order methods with second-order methods. In fact, convergence to second-order stationarity in the presence of convex constraints has been established by adapting many of the aforementioned second-order algorithms [9, 14, 13]. Under an assumption that guarantees a desired sufficient decrease of a cubic subproblem, [9] adapted the ARC algorithm and showed convergence to ϵ_g-first-order stationarity in at most $\mathcal{O}(\epsilon_g^{-3/2})$ iterations. [7] used an active set method and cubic regularization to achieve this rate for special types of constraints. The work [6] used an interior point method to achieve a second-order stationarity in $O\big(\max\{\epsilon_g^{-3/2}, \epsilon_H^{-3}\}\big)$ iterations for box constraints. For general constraints, [15] proposed a *conceptual* trust region algorithm that can compute an ϵ-q^{th} stationary point in at most $\mathcal{O}(\epsilon^{-q-1})$ iterations. More recently, [31] proposed a general framework for computing (ϵ_g, ϵ_H)-second-order stationary points for convex-constrained optimization problem with worst-case complexity $\mathcal{O}\big(\max\{\epsilon_g^{-2}, \epsilon_H^{-3}\}\big)$. In particular, this framework allows for using Frank–Wolfe or projected gradient descent to converge to an approximate first-order method and then computes a second-order descent direction if it exists.

The iteration complexity bounds computed for the methods above hide the per-iteration complexity of solving the quadratic or cubic subproblems. As shown in [34], for linearly constrained nonconvex problems, even checking whether a given point is an *approximate* second-order stationary point is NP-hard. Despite this hardness result, [34] proposed a second-order Frank–Wolfe algorithm that adapts the dynamic method introduced in [17] and identified instances for which solving the constrained quadratic subproblem can be done efficiently. The algorithm converges to approximate first- and second-order stationarity with a worst-case complexity similar to [17]. However, second-order information as utilized in the adapted ARC algorithm yields better iteration complexity rates. Motivated by this result, in this paper, we propose a trust region algorithm, entitled LC-TRACE, that adapts TRACE to linearly constrained nonconvex problems. We establish the convergence of our algorithm to (ϵ_g, ϵ_H)-second-order stationarity in at most $\widetilde{\mathcal{O}}(\epsilon_g^{-3/2}, \epsilon_H^{-3})$ iterations.

The remainder of this paper is organized as follows. In section 2, we first review and define the concepts of first- and second-order stationarity. Then, we review some of our previous results in section 3. Finally, in section 4, we propose and analyze the LC-TRACE algorithm.

2. First- and second-order stationarity definitions.

To understand the definition of first- and second-order stationarity, let us first start by considering the unconstrained optimization problem

$$\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x}), \tag{2.1}$$

where $f : \mathbb{R}^n \mapsto \mathbb{R}$ is a twice continuously differentiable function. We say a point $\bar{\mathbf{x}}$ is a first-order stationary point (FOSP) of (2.1) if $\nabla f(\bar{\mathbf{x}}) = \mathbf{0}$. Similarly, a point $\bar{\mathbf{x}}$ is said to be a second-order stationary point (SOSP) of (2.1) if $\nabla f(\bar{\mathbf{x}}) = \mathbf{0}$ and $\nabla^2 f(\bar{\mathbf{x}}) \succeq 0$. In practice, most of the algorithms used for finding stationary points are iterative. Therefore, we define the concept of approximate first- and second-order stationarity. We say a point $\bar{\mathbf{x}}$ is an ϵ_g-first-order stationary point if

$$\|\nabla f(\bar{\mathbf{x}})\|_2 \leq \epsilon_g. \tag{2.2}$$

Moreover, we say a point $\bar{\mathbf{x}}$ is an (ϵ_g, ϵ_H)-second-order stationary point if

$$\|\nabla f(\bar{\mathbf{x}})\|_2 \leq \epsilon_g \text{ and } \nabla^2 f(\bar{\mathbf{x}}) \succeq -\epsilon_H \mathbf{I}. \tag{2.3}$$

We now extend these definitions to the constrained optimization problem

$$\min_{\mathbf{x}\in\mathcal{P}} f(\mathbf{x}), \tag{2.4}$$

where $\mathcal{P} \subseteq \mathbb{R}^n$ is a closed convex set. As defined in [5], we say $\bar{\mathbf{x}} \in \mathcal{P}$ is a FOSP of (2.4) if

$$\langle \nabla f(\bar{\mathbf{x}}), \mathbf{x} - \bar{\mathbf{x}} \rangle \geq 0 \quad \forall\, \mathbf{x} \in \mathcal{P}. \tag{2.5}$$

Similarly, we say a point $\bar{\mathbf{x}}$ is a SOSP of the optimization problem (2.4) if $\bar{\mathbf{x}} \in \mathcal{P}$ is a first-order stationary point and

$$0 \leq \mathbf{d}^T \nabla^2 f(\bar{\mathbf{x}}) \mathbf{d} \quad \forall\, \mathbf{d} \text{ s.t. } \langle \mathbf{d}, \nabla \mathrm{f}(\bar{\mathbf{x}}) \rangle = 0 \text{ and } \bar{\mathbf{x}} + \mathbf{d} \in \mathcal{P}. \tag{2.6}$$

Notice that when $\mathcal{P} = \mathbb{R}^n$, the definitions above obviously correspond to the definitions in the unconstrained case.

Motivated by (2.5) and (2.6), given a feasible point $\mathbf{x}$, we define the following first- and second-order stationarity measures

$$\begin{aligned} \mathcal{X}(\mathbf{x}) \triangleq - \min_{\mathbf{s}} \quad & \langle \nabla f(\mathbf{x}), \mathbf{s} \rangle \\ \text{s.t.} \quad & \mathbf{x} + \mathbf{s} \in \mathcal{P},\ \|\mathbf{s}\| \leq 1, \end{aligned} \tag{2.7}$$

and

$$\psi(\mathbf{x}) \triangleq -\min_{\mathbf{d}} \quad \mathbf{d}^T \nabla^2 f(\mathbf{x}) \mathbf{d} \quad \text{s.t.} \quad \mathbf{x}+\mathbf{d} \in \mathcal{P},\ \|\mathbf{d}\| \leq 1, \quad \langle \nabla f(\mathbf{x}), \mathbf{d} \rangle \leq 0. \tag{2.8}$$

Notice that since $\mathbf{x}$ is feasible, $\mathcal{X}(\mathbf{x}) \geq 0$ and $\psi(\mathbf{x}) \geq 0$. Moreover, these optimality measures, which are also used in [34], can be linked to the standard definitions in [5] by the following lemma.

Lemma 2.1 (see [34]). *The first- and second-order stationarity measures $\mathcal{X}(\cdot)$ and $\psi(\cdot)$ are continuous in $\mathbf{x}$. Moreover, if $\bar{\mathbf{x}} \in \mathcal{P}$, then the following hold:*
- $\mathcal{X}(\bar{\mathbf{x}}) = 0$ *if and only if $\bar{\mathbf{x}}$ is a first-order stationary point.*
- $\mathcal{X}(\bar{\mathbf{x}}) = \psi(\bar{\mathbf{x}}) = 0$ *if and only if $\bar{\mathbf{x}}$ is a second-order stationary point.*

Using this lemma, we define the approximate first- and second-order stationarity. %def 2.2

Definition 2.2. *Approximate stationary point: For problem* (2.4)*,*
- *A point $\bar{\mathbf{x}} \in \mathcal{P}$ is said to be an ϵ_g-first-order stationary point if $\mathcal{X}(\bar{\mathbf{x}}) \leq \epsilon_g$.*
- *A point $\bar{\mathbf{x}} \in \mathcal{P}$ is said to be an (ϵ_g, ϵ_H)-second-order stationary point if $\mathcal{X}(\bar{\mathbf{x}}) \leq \epsilon_g$ and $\psi(\bar{\mathbf{x}}) \leq \epsilon_H$.*

In the unconstrained scenario, these definitions correspond to the standard definitions (2.3) and (2.2).

Remark 2.3. Notice that our definition of (ϵ_g, ϵ_H)-second-order stationarity is different than the definition in [31]. In particular, there are two major differences:

(1) The definition used for approximate first- and second-order stationarity in [31] does not include the normalization constraints $\|\mathbf{s}\| \leq 1$ and $\|\mathbf{d}\| \leq 1$ in (2.7) and (2.8).

(2) The second-order optimality measure in [31] is defined based on using equality constraint $\langle \nabla f(\mathbf{x}), \mathbf{d} \rangle = 0$ in (2.8) instead of the inequality constraint $\langle \nabla f(\mathbf{x}), \mathbf{d} \rangle \leq 0$.

To understand the necessity of using normalization, consider the optimization problem $\min x^2$ and the point $\bar{x} = \epsilon$ with ϵ being (arbitrary) small. Clearly, $\bar{x}$ is close to optimal, while the optimality measure (2.7) does not reflect this approximate optimality if we do not include the normalization constraint in (2.7).

To understand the importance of using inequality constraint $\langle \nabla f(\mathbf{x}), \mathbf{d} \rangle \leq 0$ instead of equality constraint in (2.8), consider the scalar optimization problem

$$\min_{x} \quad -\frac{1}{2}x^2 \quad \text{s.t.} \quad 0 \leq x \leq 10.$$

Let us look at the point $\bar{x} = \epsilon > 0$. Using second-order information, one can say that $\bar{x}$ is not a reasonable point to terminate your algorithm at. This is because the Hessian provides a descent direction with a large amount of improvement in the second-order approximation of the objective value. This fact is also reflected in the value of $\psi(\bar{x}) = 1$. However, if we had used equality constraint $\langle \nabla f(\mathbf{x}), \mathbf{d} \rangle = 0$ in the definition of $\psi(\cdot)$ in (2.8), then the value of $\psi(\cdot)$ would have been zero.

Remark 2.4. There are other definitions of second-order stationarity in the literature. For example, the works [6, 7] use a scaled version of the Hessian in different directions to define second-order stationarity for box constraints. Recently, [37] carefully revised it to account for the coordinates which are very far from the boundary. Another related definition of second-order stationary, which leads to a practical perturbed gradient descent algorithm, is provided in [29] for general linearly constrained optimization problems.

3. Finding second-order stationary points for constrained optimization. Consider the quadratic co-positivity problem

$$\min_{\mathbf{x}\in\mathbb{R}^n} \quad \frac{1}{2}\mathbf{x}^T\mathbf{Q}\mathbf{x} \qquad \text{s.t.} \quad \mathbf{x} \geq \mathbf{0},\ \|\mathbf{x}\| \leq 1. \tag{3.1}$$

Clearly, checking whether $\bar{\mathbf{x}} = \mathbf{0}$ is a second-order stationary point of (3.1) is equivalent to checking its local optimality, which is an NP-hard problem [32]. This observation shows that checking exact second-order stationarity is hard. The following result, which is borrowed from [34], shows that even checking *approximate* second-order stationarity is NP-hard.

THEOREM 3.1 (see [34, Theorem 6]). *There is no algorithm which can check whether $x = 0$ is an (ϵ_g, ϵ_H)-second-order stationary point in polynomial time in $(n, 1/\epsilon_H)$, unless $P = NP$.*

This hardness result implies that we should not expect an efficient algorithm for finding second-order stationary points of nonconvex problems. However, several recent results [25, 24] have provided various conditions under which the problem can be solved in polynomial time. For instance, [25] has defined a dimension condition under which the linearly constrained quadratic subproblem admits an exact SDP-relaxation which can be solved in polynomial time. The proposed dimension condition was further improved by [24]. Furthermore, the authors in the latter paper proposed a backtracking approach that efficiently solves this quadratic constrained optimization problem given a fixed number of linear constraints (not dependent on the dimension of the problem). Along the same line, a branch and bound method was proposed in [4] to solve the problem when the number of constraints is small.

Motivated by this observation, in the next section we describe our LC-TRACE algorithm and analyze its iteration complexity for finding second-order stationary points of linearly constrained nonconvex optimization problems. A core assumption in our algorithm is that a certain quadratic objective can be minimized given existing linear constraints (for example, when m is small).

4. A trust region algorithm for solving linearly constrained smooth nonconvex optimization problems. Consider the optimization problem

$$\begin{aligned} &\min_{\mathbf{x}\in\mathbb{R}^n} \ f(\mathbf{x}), \\ &\text{s.t.} \quad \mathbf{A}\mathbf{x} \leq \mathbf{b}, \end{aligned} \tag{4.1}$$

where $\mathbf{A} \in \mathbb{R}^{m\times n}$ and $\mathbf{b} \in \mathbb{R}^m$. In this section, we propose a trust region algorithm, entitled LC-TRACE (linearly constrained TRACE) that adapts TRACE [19] to the above linearly constrained nonconvex problem. We establish its convergence to ϵ_g-first-order stationarity with iteration complexity order $\widetilde{\mathcal{O}}(\epsilon_g^{-3/2})$. This method is then used to develop an algorithm to converge to (ϵ_g, ϵ_H)-second-order stationarity with the iteration complexity $\widetilde{\mathcal{O}}(\max\{\epsilon_g^{-3/2}, \epsilon_H^{-3}\})$.

LC-TRACE is different from the traditional trust region method proposed in [15] for constrained optimization. More specifically, LC-TRACE utilizes the mechanisms used in TRACE [19] to provide a faster convergence rate compared to [15]. The improved convergence rate matches the rates achieved by adapted ARC [9] and TRACE [19], up to logarithmic factors. Since applying TRACE directly to constrained optimization fails (as will be discussed later), we introduced modifications to adapt this method to linearly constrained problems. Our modifications are not the result of a "simple extension" of the unconstrained to constrained scenario. Before explaining LC-TRACE, let us first provide an overview of the classical trust region and TRACE algorithms.

4.1. Background on traditional trust region algorithm and TRACE. In traditional trust region methods, the trial step $\mathbf{s}_k$ at iteration k is computed by solving the standard trust region subproblem

$$\min_{\mathbf{s}\in\mathbb{R}^n} q_k(\mathbf{s}), \quad \text{s.t. } \|\mathbf{s}\|_2 \leq \delta_k, \tag{4.2}$$

where $q_k(\mathbf{s}) : \mathbb{R}^n \mapsto \mathbb{R}$ is the second-order Taylor approximation of f around $\mathbf{x}_k$, i.e.,

$$q_k(\mathbf{s}) \triangleq f_k + \mathbf{g}_k^T\mathbf{s} + \frac{1}{2}\mathbf{s}^T\mathbf{H}_k\mathbf{s}.$$

Here $f_k = f(\mathbf{x}_k)$, $\mathbf{g}_k = \nabla f(\mathbf{x}_k)$, and $\mathbf{H}_k = \nabla^2 f(\mathbf{x}_k)$. Based on the resulting trial step, an acceptance criteria is used to either *accept* or *reject* the step. In particular, if the ratio of actual-to-predicted reduction

$$\frac{f_k - f(\mathbf{x}_k + \mathbf{s}_k)}{f_k - q_k(\mathbf{s}_k)}$$

is greater than a prescribed constant, the step is accepted; otherwise it is rejected. The iterate $\mathbf{x}^{k+1}$ and trust region radius are updated accordingly. Traditional trust region methods use a geometric update rule for the trust region radius δ_k, i.e., δ_{k+1} is some constant factor of δ_k. The TRACE algorithm, on the other hand, modifies the acceptance criteria and this linear update rule for δ_k to match the rate achieved by the ARC algorithm [10, 11]. In particular, the authors in [19] observed that ARC computes a positive sequence of cubic regularization coefficients $\sigma_k \in [\underline{\sigma}, \overline{\sigma}]$ that satisfy

$$f_k - f_{k+1} \geq c_1\sigma_k\|\mathbf{s}_k\|_2^3 \quad \text{and} \quad \|\mathbf{s}_k\|_2 \geq \left(\frac{c_2}{\overline{\sigma} + c_3}\right)^{1/2}\|\mathbf{g}_{k+1}\|_2^{1/2} \tag{4.3}$$

for some given positive constants c_1, c_2, c_3. TRACE designed a modified acceptance criteria and a new mechanism for updating the trust region radius to satisfy the conditions provided in (4.3). Some of these ideas are discussed next.

Sufficient decrease acceptance criteria. TRACE defines the ratio

$$\rho_k \triangleq \frac{f_k - f(\mathbf{x}_k + \mathbf{s}_k)}{\|\mathbf{s}_k\|_2^3} \tag{4.4}$$

as a measure to decide whether to accept or reject a trial step. For some prescribed $\rho \in (0, 1)$, a trial step $\mathbf{s}_k$ can only be accepted if $\rho_k \geq \rho$. By noticing that a small $\|\mathbf{s}_k\|_2$ may satisfy only the first condition in (4.3), the developers of TRACE realize that an acceptance criteria that only involves (4.4) is not sufficient. To avoid such cases, TRACE defines a sequence $\{\sigma_k\}$ to estimate an upper bound for the ratio $\lambda_k/\|\mathbf{s}_k\|_2$ used for acceptance. Here $\{\lambda_k\}$ is the sequence of dual variables corresponding to the constraint $\|\mathbf{s}\|_2 \leq \delta_k$ in subproblem (4.2). In short, TRACE accepts a trial pair $(\mathbf{s}_k, \lambda_k)$ if it satisfies the following conditions:

$$\rho_k \geq \rho \quad \text{and} \quad \lambda_k/\|\mathbf{s}_k\|_2 \leq \sigma_k. \tag{4.5}$$

Trust region radius update procedure. In contrast to the linear update rule utilized in traditional trust region algorithms, TRACE uses a CONTRACT subroutine that allows for sublinear updates. In particular, this subroutine compares the radius obtained by the linear update scheme to the norm of the trial step computed using

$$\min_{\mathbf{s}\in\mathbb{R}^n} \; f_k + \mathbf{g}_k^T\mathbf{s} + \frac{1}{2}\mathbf{s}^T(\mathbf{H}_k + \lambda\mathbf{I})\mathbf{s} \tag{4.6}$$

for a carefully chosen λ. If the norm of this trial step falls within a desired range, then it is chosen to be the new trust region radius. This subroutine is called at iteration k if $\rho_k < \rho$.

TRACE is designed to solve unconstrained smooth optimization problems. A direct implementation of this algorithm fails in the constrained setting. In the next section, we describe *two fundamental difficulties* introduced in the presence of constraints and discuss the necessary modifications.

4.2. Difference between LC-TRACE and TRACE. In the constrained setting, we define the trust region subproblem and its regularized Lagrangian form as

$$Q_k \triangleq \min_{\mathbf{s}\in\mathbb{R}^n} \; q_k(\mathbf{s}), \quad \text{s.t.} \; \begin{cases} \mathbf{A}\mathbf{s} \leq \mathbf{b} - \mathbf{A}\mathbf{x}_k, \\ \|\mathbf{s}\|_2 \leq \delta_k \end{cases} \tag{4.7}$$

and

$$Q_k(\lambda) \triangleq \min_{\mathbf{s}\in\mathbb{R}^n} \; f_k + \mathbf{g}_k^T\mathbf{s} + \frac{1}{2}\mathbf{s}^T(\mathbf{H}_k + \lambda\mathbf{I})\mathbf{s}, \quad \text{s.t.} \; \mathbf{A}\mathbf{s} \leq \mathbf{b} - \mathbf{A}\mathbf{x}_k. \tag{4.8}$$

A major difficulty introduced by the constraints is related to the optimality conditions of the subproblem. In the unconstrained case, it is known that $\mathbf{H}_k + \lambda_k\mathbf{I} \succeq 0$ at every iteration [16, Corollary 7.2.2] for optimal Lagrange multiplier λ_k. Along with the fact that $\lambda > \lambda_k$ in the CONTRACT subroutine of TRACE, we conclude that $Q_k(\lambda)$ is a strongly convex quadratic optimization problem which has a unique global minimizer. Let $\mathbf{s}^*(\lambda)$ be the solution of $Q_k(\lambda)$. It follows that the function $\mathbf{s}^*(\lambda)$ is continuous in λ in the unconstrained scenario. However, in the linearly constrained scenario, the regularized subproblem (4.8) might have multiple optimal solutions. Moreover, $\mathbf{s}^*(\lambda)$ and the ratio $\lambda/\|\mathbf{s}^*(\lambda)\|_2$, which are core quantities in TRACE, might not even be continuous. To clarify this difficulty, consider the following simple example:

$$Q(\lambda) = \min_{s_1\leq 5,\, s_2\geq 0} s_1^2 - s_2^2 + \lambda(s_1^2 + s_2^2) \quad \text{s.t.} \; s_2 - 3s_1 \leq -12. \tag{4.9}$$

It is not hard to see that the optimal solution of (4.9) is given by

$$s^*(\lambda) = \begin{cases} (5,3) & \text{if } \lambda < 0, \\ (5,3);\,(4,0) & \text{if } \lambda = 0, \\ (4,0) & \text{otherwise.} \end{cases}$$

Thus, a small increase in λ may lead to a huge change in the ratio $\lambda/\|s^*(\lambda)\|_2$. Therefore, the luxury of having an arbitrarily choice for the bounds $\underline{\sigma}$ and $\bar{\sigma}$ of the ratio $\lambda/\|\mathbf{s}\|_2$ is not present in the constrained case. In LC-TRACEC, we resolved this issue by defining

$$\underline{\sigma} = \frac{\epsilon}{M} \quad \text{and} \quad \bar{\sigma} = 2\Delta \tag{4.10}$$

and altering the update rule of λ in the CONTRACT subroutine. Here $\epsilon > 0$ is the threshold used for the termination of the algorithm, and M is a positive scalar constant defined in (A.26). While estimating the exact parameters of $\underline{\sigma}$ and $\bar{\sigma}$ may be challenging, we only need to have an upper-bound for $\bar{\sigma}$ and a lower bound for $\underline{\sigma}$ to run the algorithm.

Another major difficulty in the constrained scenario is related to the standard trust region theory on the relationship between subproblem solutions and their corresponding dual variables. In the unconstrained case, $\lambda_1 > \lambda_2$, implies $\|\mathbf{s}^*(\lambda_1)\|_2 < \|\mathbf{s}^*(\lambda_1)\|_2$ (see [16, Chapter 7]). This relationship was used in [19] to show that the CONTRACT subroutine reduces the radius of the trust region. However, it can be seen from example (4.9) that this relation may not hold in the constrained case. To account for this issue, we modified the CONTRACT subroutine to guarantee a reduction in the trust region radius (see Lemma A.2). In summary, the differences between LC-TRACE and TRACE are mainly in the CONTRACT subroutine. Next, we describe the steps of the algorithm.

4.3. Description of LC-TRACE. Our proposed algorithm LC-TRACE has two main building blocks: *First-Order-LC-TRACE* and *Second-Order-LC-TRACE*. We first present First-Order-LC-TRACE, which can converge to ϵ_g-first-order stationarity in $\tilde{\mathcal{O}}(\epsilon_g^{-3/2})$. Then, we use this algorithm in Second-Order-LC-TRACE to find an (ϵ_g, ϵ_H)-Second-Order stationarity in $\tilde{\mathcal{O}}(\max\{\epsilon_g^{-3/2}, \epsilon_H^{-3}\})$ iterations.

The First-Order-LC-TRACE algorithm is Algorithm 4.1. At each iteration $\mathbf{x}_k$, this iterative algorithm computes the values $\mathbf{s}_k$, $\boldsymbol{\lambda}_k$, and ρ_k by solving the optimization problem (4.7) and using (4.4). Depending on the obtained values, it decides to either accept the trial point $\mathbf{s}_k$ or reject it. When rejecting the trial point, it either goes to *contraction* or *expansion* procedures. Thus, the main decisions include acceptance, contraction, or expansion. We distinguish the iterations by partitioning the set of iteration numbers into what we refer to as the sets of accepted ($\mathcal{A}$), contraction ($\mathcal{C}$) and expansion ($\mathcal{E}$) steps:

$$\begin{aligned}
&\mathcal{A} \triangleq \{k \in \mathbb{N} : \rho_k \geq \rho \text{ and either } \lambda_k \leq \sigma_k \|\mathbf{s}_k\|_2 \text{ or } \|\mathbf{s}_k\|_2 = \Delta_k\},\\
&\mathcal{C} \triangleq \{k \in \mathbb{N} : \rho_k < \rho\}, \text{ and}\\
&\mathcal{E} \triangleq \{k \in \mathbb{N} : k \notin \mathcal{A} \cup \mathcal{C}\}.
\end{aligned}$$

Hence, step k is accepted if the computed pair $(\mathbf{s}_k, \lambda_k)$ satisfies the sufficient decrease criteria $\rho_k \geq \rho$ and either the norm of $\mathbf{s}_k$ is large enough ($\|\mathbf{s}_k\|_2 = \Delta_k$) or the ratio $\lambda_k/\|\mathbf{s}_k\|_2$ is smaller than an upper-bound σ_k. We also partition the set of accepted steps into two disjoint subsets:

$$\mathcal{A}_\Delta \triangleq \{k \in \mathcal{A} : \|\mathbf{s}_k\|_2 = \Delta_k\} \text{ and } \mathcal{A}_\sigma \triangleq \{k \in \mathcal{A} : k \notin \mathcal{A}_\Delta\}.$$

The sequence Δ_k is used in the algorithm as an upper bound on the norm of $\|\mathbf{s}_k\|_2$. From steps 7, 12, and 16, we notice that this sequence is nondecreasing. We now describe the update mechanism used in a contraction step of First-Order-LC-TRACE, which is the main difference between TRACE and our proposed algorithm.

Algorithm 4.1 First-Order-LC-TRACE.

Require: an acceptance constant $\rho \in (0, 1)$.
Require: update constants $\{\gamma_C, \gamma_E, \gamma_\lambda\}$ with $\gamma_C \in (0, 1)$ and $\gamma_\lambda, \gamma_E > 1$.
Require: ratio bound constants $\underline{\sigma}$ and $\overline{\sigma}$ defined in (4.10).

procedure FIRST-ORDER-LC-TRACE
  Choose a feasible point $\mathbf{x}_0$, a pair $\{\delta_0, \Delta_0\}$ with $0 < \delta_0 \le \Delta_0$, and σ_0 with $\sigma_0 \ge \underline{\sigma}$.
  Compute $(\mathbf{s}_0, \lambda_0)$ by solving Q_0, then compute ρ_0 using the definition in (4.4).
  for $k = 0, 1, 2, \ldots$ **do**
    if $\rho_k \ge \rho$ and either $\lambda_k/\|\mathbf{s}_k\|_2 \le \sigma_k$ or $\|\mathbf{s}_k\|_2 = \Delta_k$ **then** (Acceptance)
      set $\mathbf{x}_{k+1} \leftarrow \mathbf{x}_k + \mathbf{s}_k$
      set $\Delta_{k+1} \leftarrow \max\{\Delta_k,\ \gamma_E\|\mathbf{s}_k\|_2\}$
      set $\delta_{k+1} \leftarrow \min\{\Delta_{k+1}, \max\{\delta_k,\ \gamma_E\|\mathbf{s}_k\|_2\}\}$
      set $\sigma_{k+1} \leftarrow \max\{\sigma_k,\ \lambda_k/\|\mathbf{s}_k\|_2\}$
    else if $\rho_k < \rho$ **then** (Contraction)
      set $\mathbf{x}_{k+1} \leftarrow \mathbf{x}_k$
      set $\Delta_{k+1} \leftarrow \Delta_k$
      set $\delta_{k+1} \leftarrow \text{CONTRACT}(\mathbf{x}_k, \delta_k, \sigma_k, \mathbf{s}_k, \lambda_k)$ defined in Algorithm (4.2)
    else if $\rho_k \ge \rho$, $\lambda_k/\|\mathbf{s}_k\|_2 > \sigma_k$, and $\|\mathbf{s}_k\|_2 < \Delta_k$ **then** (Expansion)
      set $\mathbf{x}_{k+1} \leftarrow \mathbf{x}_k$
      set $\Delta_{k+1} \leftarrow \Delta_k$
      set $\delta_{k+1} \leftarrow \min\{\Delta_{k+1},\ \lambda_k/\sigma_k\}$
      set $\sigma_{k+1} \leftarrow \sigma_k$
    end if
    Compute $(\mathbf{s}_{k+1}, \lambda_{k+1})$ by solving Q_{k+1}, then compute ρ_{k+1} using (4.4).
    if $\rho_k < \rho$ **then**
      set $\sigma_{k+1} \leftarrow \max\{\sigma_k,\ \lambda_{k+1}/\|\mathbf{s}_{k+1}\|_2\}$
    end if
  end for
end procedure

When the CONTRACT subroutine is called, two different cases may occur in Algorithm 4.2. The first case is reached whenever conditions in step 3 in the CONTRACT subroutine tests true. In that case, we carefully choose choose $\lambda > \lambda_k$ to ensure that the pair $(\mathbf{s}, \lambda)$ with $\mathbf{s}$ being the solution of $Q_k(\lambda)$ satisfies

$$\underline{\sigma} \le \lambda/\|\mathbf{s}\|_2 \le \overline{\sigma},$$

where $\underline{\sigma}$ and $\bar{\sigma}$ are prescribed positive constants defined in (4.10). The second case is reached whenever the conditions in step 3 test false. In that case, we choose $\lambda \in (\lambda_k, C\lambda_k]$ with $C > 1$ a constant scalar to ensure that the pair $(\mathbf{s}, \lambda)$ with $\mathbf{s}$ being the solution of $Q_k(\lambda)$ satisfies the following:

$$\frac{\lambda}{\|\mathbf{s}\|_2} < \max\left\{\bar{\sigma}, \left(\frac{\gamma_\lambda}{\gamma_C}\right)\frac{H_{Lip} + 2\rho}{2\kappa}\right\},$$

where $\kappa \in (0, 1]$ is a constant scalars, and H_{Lip} is defined in assumption 4.2. In what follows, we first present our results about the convergence of the First-Order-LC-TRACE algorithm and its iteration complexity.

Algorithm 4.2 CONTRACT Subroutine.

Require: update constant $\gamma_C \in (0,1)$.
Require: ratio bound constants $\underline{\sigma}$ and $\overline{\sigma}$ defined in (4.10).

procedure CONTRACT$(\mathbf{x}_k, \delta_k, \sigma_k, \mathbf{s}_k, \lambda_k)$
  set $\bar{\lambda} \leftarrow \lambda_k + \underline{\sigma}\Delta_k$ and set $\bar{\mathbf{s}}$ as the solution of $Q_k(\bar{\lambda})$.
  if $\|\bar{\mathbf{s}}\|_2 < \|\mathbf{s}_k\|_2$ and $\lambda_k < \underline{\sigma}\|\mathbf{s}_k\|_2$ **then**
    set $\lambda \leftarrow \bar{\lambda} + H_{max} + \left(\underline{\sigma}\mathcal{X}_k\right)^{1/2}$ and set $\mathbf{s}$ as the solution of $Q_k(\lambda)$.
    if $\lambda/\|\mathbf{s}\|_2 \leq \overline{\sigma}$ **then**
      return $\delta_{k+1} \leftarrow \|\mathbf{s}\|_2$
    else
      set $\lambda \leftarrow \bar{\lambda}$.
      return $\delta_{k+1} \leftarrow \|\bar{\mathbf{s}}\|_2$
    end if
  else
    if $\|\bar{\mathbf{s}}\|_2 = \|\mathbf{s}_k\|_2$ **then**
      set $\lambda \leftarrow \gamma_\lambda \bar{\lambda}$ and set $\mathbf{s}$ as the solution of $Q_k(\lambda)$.
    else
      set $\lambda \leftarrow \gamma_\lambda \lambda$ and set $\mathbf{s}$ as the solution of $Q_k(\lambda)$.
    end if
    while $\|\mathbf{s}\|_2 = \|\mathbf{s}_k\|_2$ **do**
      $\lambda \leftarrow \gamma_\lambda \lambda$ and set $\mathbf{s}$ as the solution of $Q_k(\lambda)$.
    end while
    if $\|\mathbf{s}\|_2 \geq \gamma_C\|\mathbf{s}_k\|_2$ **then**
      return $\delta_{k+1} \leftarrow \|\mathbf{s}\|_2$
    else
      return $\delta_{k+1} \leftarrow \gamma_C\|\mathbf{s}_k\|_2$
    end if
  end if
end procedure

4.4. Convergence of First-Order-LC-TRACE to first-order stationarity. Throughout this section, we make the following assumptions that are standard for global convergence theory of trust region methods.

ASSUMPTION 4.1. *The objective function f is twice continuously differentiable and bounded below by a scalar f_{min} on $\mathcal{P}$.*

ASSUMPTION 4.2. *We assume that the functions $\mathbf{g}(\cdot) \triangleq \nabla f(\cdot)$ and $\mathbf{H}(\cdot) \triangleq \nabla^2 f(\cdot)$ are Lipschitz continuous on the path defined by the iterates computed in the algorithm with Lipschitz constants g_{Lip} and H_{Lip}, respectively.*

ASSUMPTION 4.3. *We assume that there exist scalar constants g_{max}, $H_{max} > 0$ such that $\|\mathbf{g}_k\|_2 \triangleq \|\nabla f(\mathbf{x}_k)\|_2 \leq g_{max}$ and $\|H_k\|_2 \triangleq \|\nabla^2 f(\mathbf{x}_k)\|_2 \leq H_{max}$ for all $k \in \mathbb{N}$.*

We next state the main results for convergence of First-Order-LC-TRACE.

THEOREM 4.4. *Under Assumptions* 4.1, 4.2, *and* 4.3, *any limit point of the iterates generated by the First-Order-LC-TRACE algorithm is a first-order stationary point.*

Proof. The proof of the theorem is relegated to Appendix A.1. ∎

Our next theorem establishes the desired rate of convergence.

THEOREM 4.5. *Under Assumptions* 4.1, 4.2, *and* 4.3 *for any given scalar* $\epsilon \in (0, \infty)$, *the total number of subproblem routines of First-Order-LC-TRACE required to reach an* ϵ*-first-order stationary point of* (4.1) *is* $\mathcal{O}(\epsilon^{-3/2} \log^3(1/\epsilon))$.

Proof. The proof of the theorem is relegated to Appendix A.2. ∎

In the next section, we use this first-order result to develop an algorithm for finding second-order stationary points.

5. Second-Order-LC-TRACE algorithm.

Leveraging the convergence result of First-Order-LC-TRACE, we propose Algorithm 5.1 for converging to second-order stationary points.

Algorithm 5.1 Second-Order-LC-TRACE.

Require: The constants $\tilde{L} \triangleq \max\{L, g_{max}\}$, $\tilde{H} \triangleq \max\{H_{Lip}, H_{max}\}$, $\epsilon_g > 0$, $\epsilon_H > 0$.

---

procedure
  Choose a feasible point $\mathbf{x}_0$.
  Compute $\mathcal{X}_0$ and ψ_0 by solving (2.7) and (2.8), respectively.
  for $k = 0, 1, 2, \ldots$ **do**
    if $\mathcal{X}_k > \epsilon_g$ **then**
      Compute $\mathbf{x}_{k+1}$ by running one iteration of First-Order-LC-TRACE starting with $\mathbf{x}_k$.
    else
      Compute $\widehat{\mathbf{d}}_k$ and ψ_k by solving (2.8).
      set $\mathbf{x}_{k+1} \leftarrow \mathbf{x}_k + \dfrac{2\psi_k}{\tilde{H}} \widehat{\mathbf{d}}_k$.
    end if
  end for
end procedure

We now show that this algorithm can find an (ϵ_g, ϵ_H)-second-order stationary point of problem (4.1).

THEOREM 5.1. *Under Assumptions* 4.1, 4.2, *and* 4.3, *for any given scalars* $\epsilon_g > 0$ *and* $\epsilon_H > 0$, *the total number of iterations required to reach an* (ϵ_g, ϵ_H)*-second-order stationary point of* (4.1) *when running Algorithm* 5.1 *is* $\mathcal{O}(\log^3(\epsilon_g^{-1}) \max\{\epsilon_g^{-3/2}, \epsilon_H^{-3}\})$.

Proof. The proof of the theorem is relegated to Appendix A.3. ∎

Appendix A. Proofs for section 4.

Consider the following optimization problem:

$$\underset{\mathbf{x} \in \mathcal{P}}{\text{minimize}} \; f(\mathbf{x}), \tag{A.1}$$

where $\mathcal{P} \triangleq \{\mathbf{x} \in \mathbb{R}^n \,|\, \mathbf{A}\mathbf{x} \leq \mathbf{b}\}$ is a polyhedron with a finite number of linear constraints. In this section we generalize results from [19] to adapt for the linear constraints. For the sake of completeness of the manuscript, some lemmas and proofs are restated from [19].

Recall the subproblem Q_k with trust region δ_k,

$$Q_k \triangleq \min_{\mathbf{s}} q_k(\mathbf{s}) \triangleq f_k + \mathbf{g}_k^T\mathbf{s} + \frac{1}{2}\mathbf{s}^T\mathbf{H}_k\mathbf{s}, \quad \text{subject to } \begin{cases} \mathbf{A}\mathbf{s} \leq \mathbf{b} - \mathbf{A}\mathbf{x}_k, \\ \|\mathbf{s}\|_2 \leq \delta_k. \end{cases}$$

Let $\boldsymbol{\lambda}_k^C$ be the multiplier corresponding to the linear constraint $\mathbf{A}\mathbf{s} \leq \mathbf{b} - \mathbf{A}\mathbf{x}_k$ and λ_k be the multiplier for the trust region constraint $\|\mathbf{s}\|_2 \leq \delta_k$. The first-order KKT optimality conditions for the above problem are stated below [5]:

$$\mathbf{g}_k + (\mathbf{H}_k + \lambda_k\mathbf{I})\mathbf{s}_k + \mathbf{A}^T\boldsymbol{\lambda}_k^C = \mathbf{0}, \tag{A.2}$$

$$\mathbf{0} \leq \boldsymbol{\lambda}_k^C \perp \mathbf{b} - \mathbf{A}\mathbf{x}_k - \mathbf{A}\mathbf{s}_k \geq \mathbf{0}, \tag{A.3}$$

$$0 \leq \lambda_k \perp \delta_k - \|\mathbf{s}_k\|_2^2 \geq 0, \tag{A.4}$$

where $\perp$ denotes orthogonality of vectors.

A.1. Proof of Theorem 4.4. To show convergence to first-order stationarity, we first provide in Lemma A.1 a sufficient decrease condition. Then, in Lemma A.7 we show that the number of accepted steps $|\mathcal{A}|$ is infinite. Combining these two results with the assumption that f is lower bounded, we get the desired convergence result.

LEMMA A.1. *For any $k \in \mathbb{N}$, the trial step $\mathbf{s}_k$ and dual variable λ_k satisfy*

$$f_k - q_k(\mathbf{s}_k) \geq \frac{1}{2}\mathbf{s}_k^T(\mathbf{H}_k + \lambda_k\mathbf{I})\mathbf{s}_k + \frac{1}{2}\lambda_k\|\mathbf{s}_k\|_2^2. \tag{A.5}$$

In addition, for any $k \in \mathbb{N}_+$, the trial step $\mathbf{s}_k$ satisfies

$$f_k - q_k(\mathbf{s}_k) \geq C\mathcal{X}_k \min\Big\{\delta_k, \frac{\mathcal{X}_k}{\|\mathbf{H}_k\|_2}, 1\Big\}, \tag{A.6}$$

where C is a constant positive scalar.

Proof. By definition of q_k,

$$\begin{aligned} f_k - q_k(\mathbf{s}_k) &= -\mathbf{g}_k^T\mathbf{s}_k - \frac{1}{2}\mathbf{s}_k^T\mathbf{H}_k\mathbf{s}_k \\ &= \mathbf{s}_k^T\mathbf{H}_k\mathbf{s}_k + \lambda_k\|\mathbf{s}_k\|_2^2 + \mathbf{s}_k^T\mathbf{A}^T(\boldsymbol{\lambda}_k^C) - \frac{1}{2}\mathbf{s}_k^T\mathbf{H}_k\mathbf{s}_k \\ &= \frac{1}{2}\mathbf{s}_k^T(\mathbf{H}_k + \lambda_k\mathbf{I})\mathbf{s}_k + \frac{1}{2}\lambda_k\|\mathbf{s}_k\|^2 + \mathbf{s}_k^T\mathbf{A}^T(\boldsymbol{\lambda}_k^C) \\ &\geq \frac{1}{2}\mathbf{s}_k^T(\mathbf{H}_k + \lambda_k\mathbf{I})\mathbf{s}_k + \frac{1}{2}\lambda_k\|\mathbf{s}_k\|^2, \end{aligned} \tag{A.7}$$

where the second equality follows by KKT condition (A.2), and the last inequality follows from the feasibility of $\mathbf{x}_k$ and the complementary slackness (A.3).

Also, using [16, Theorem 12.2.2], we obtain

$$f_k - q_k(\mathbf{s}_k) \geq C\mathcal{X}_k \min\Big\{\delta_k, \frac{\mathcal{X}_k}{\|\mathbf{H}_k\|_2}, 1\Big\}. \qquad \square$$

To prove the infinite cardinality of the set $\mathcal{A}$, we need some intermediate lemmas. The next result shows that the trust region radius is reduced when the CONTRACT subroutine is called.

LEMMA A.2. *For any $k \in \mathbb{N}$, if $k \in \mathcal{C}$, then $\delta_{k+1} < \delta_k$.*

Proof. Suppose that $k \in \mathcal{C}$. We prove the result by considering the various cases that may occur within the CONTRACT subroutine. If step 23 is reached, the subroutine returns $\delta_{k+1} = \gamma_C \|\mathbf{s}_k\|_2 < \delta_k$. Otherwise, if step 6 is reached, the subroutine returns $\delta_{k+1} = \|\mathbf{s}\|_2$, where $\mathbf{s}$ solves $Q_k(\lambda)$ for $\lambda \geq \bar{\lambda}$. Hence,

$$\delta_{k+1} = \|\mathbf{s}\|_2 < \|\mathbf{s}_k\|_2 \leq \delta_k,$$

where the strict inequality follows from step 3. Similarly, if step 9 is reached, the subroutine returns $\delta_{k+1} = \|\bar{\mathbf{s}}\|_2$, where $\bar{\mathbf{s}}$ solves $Q_k(\bar{\lambda})$. Hence,

$$\delta_{k+1} = \|\bar{\mathbf{s}}\|_2 < \|\mathbf{s}_k\|_2 \leq \delta_k.$$

Otherwise, step 21 is reached, in which case the subroutine returns $\delta_{k+1} = \|\mathbf{s}\|_2$, where $\mathbf{s}$ solves $Q_k(\lambda)$ for $\lambda > \lambda_k$. The result follows using the while loop condition step 17 along with the inverse relationship of λ and $\|\mathbf{s}\|$. ∎

We now show that for all iterations k, the trust region region radius δ_k is upper bounded by a nondecreasing sequence $\{\Delta_k\}$. Also, if $k \in \mathcal{A} \cup \mathcal{E}$, we show that $\delta_{k+1} \geq \delta_k$.

LEMMA A.3. *For any $k \in \mathbb{N}$, there holds $\delta_k \leq \Delta_k \leq \Delta_{k+1}$. Moreover, $\delta_{k+1} \geq \delta_k$ for all $k \in \mathcal{A} \cup \mathcal{E}$.*

Proof. The fact that $\Delta_k \leq \Delta_{k+1}$ for all $k \in \mathbb{N}$ follows from the computations in steps 7, 12, and 16 of Algorithm 4.1. It remains to show that $\delta_k \leq \Delta_k$ for all $k \in \mathbb{N}$. We prove the result by means of induction.

The inequality holds for $k = 0$ by the initialization of quantities in step 2 of Algorithm 4.1. Assume the induction hypothesis holds for iteration k. By the computations in steps 7, 8, 16, 17 and by Lemma A.2, the result holds for iteration $k+1$. We next show that $\delta_{k+1} \geq \delta_k$ for all $k \in \mathcal{A} \cup \mathcal{E}$.

Suppose $k \in \mathcal{A}$. It follows from steps 7 and 8 that

$$\delta_{k+1} = \min\{\max\{\Delta_k, \gamma_E \|\mathbf{s}_k\|_2\}, \max\{\delta_k, \gamma_E \|\mathbf{s}_k\|_2\}\} \geq \delta_k.$$

Here the inequality follows since $\delta_k \leq \Delta_k \leq \Delta_{k+1}$. Now suppose $k \in \mathcal{E}$. By the conditions indicated in step 14, we have $\lambda_k > \sigma_k \|\mathbf{s}_k\|_2 \geq 0$. It follows by (A.4) that $\|\mathbf{s}_k\|_2 = \delta_k$. We obtain

$$\delta_{k+1} = \min\{\Delta_{k+1}, \lambda_k/\sigma_k\} \geq \min\{\delta_k, \|\mathbf{s}_k\|_2\} = \delta_k,$$

where the inequality follows since $\delta_k \leq \Delta_k \leq \Delta_{k+1}$. ∎

The next result shows that we cannot have two consecutive expansion steps.

LEMMA A.4. *For any $k \in \mathbb{N}$, if $k \in \mathcal{C} \cup \mathcal{E}$, then $k+1 \notin \mathcal{E}$.*

Proof. Observe that if $\lambda_{k+1} = 0$, then conditions in step 14 of Algorithm 4.1 ensure that $(k+1) \notin \mathcal{E}$. Thus, by (A.4), we may proceed under the assumptions that $\|\mathbf{s}_{k+1}\|_2 = \delta_{k+1}$ and $\lambda_{k+1} > 0$.

Suppose that $k \in \mathcal{C}$, i.e., $\rho_k < \rho$. It follows that step 22 sets $\sigma_{k+1} \geq \lambda_{k+1}/\|\mathbf{s}_{k+1}\|_2$. Therefore, if $\rho_{k+1} \geq \rho$, we have $(k+1) \in \mathcal{A}$. Otherwise, $\rho_{k+1} < \rho$, which implies that $(k+1) \in \mathcal{C}$.

Now suppose that $k \in \mathcal{E}$. It follows that

$$\lambda_k > \sigma_k \|\mathbf{s}_k\|_2, \quad \delta_{k+1} = \min\{\Delta_k, \lambda_k/\sigma_k\}, \quad \text{and} \quad \sigma_{k+1} = \sigma_k. \tag{A.8}$$

Combined with (A.4), we get $\|\mathbf{s}_k\|_2 = \delta_k$. We now consider two different cases:

1. Suppose $\Delta_k \geq \lambda_k/\sigma_k$. It follows from (A.8) that

$$\delta_{k+1} = \lambda_k/\sigma_k > \|\mathbf{s}_k\|_2 = \delta_k. \tag{A.9}$$

Therefore, by the relationship between the trust region radius and its corresponding multiplier, we get $\lambda_{k+1} \leq \lambda_k$. Combined with (A.8) and (A.9), we obtain

$$\lambda_{k+1} \leq \lambda_k = \sigma_k \delta_{k+1} = \sigma_{k+1}\|\mathbf{s}_{k+1}\|_2.$$

Hence $(k+1) \notin \mathcal{E}$.

2. Suppose $\Delta_k < \lambda_k/\sigma_k$. Using (A.8)

$$\|\mathbf{s}_{k+1}\|_2 = \delta_{k+1} = \Delta_k = \Delta_{k+1},$$

where the last equality holds by step 16. If $\rho_{k+1} \geq \rho$, then $(k+1) \in \mathcal{A}_\Delta \subseteq \mathcal{A}$. Otherwise, $\rho_{k+1} < \rho$, from which it follows that $(k+1) \in \mathcal{C}$.

Hence, in both cases $(k+1) \notin \mathcal{E}$. ∎

Next, we show that if the dual variable for the trust region constraint λ_k is sufficiently large, then the constraint is active and the sufficient decrease criteria is met.

Lemma A.5. *For any $k \in \mathbb{N}$, if the trial step $\mathbf{s}_k$ and dual variable λ_k satisfy*

$$\lambda_k \geq g_{Lip} + H_{max} + \rho\|\mathbf{s}_k\|_2, \tag{A.10}$$

then $\|\mathbf{s}_k\|_2 = \delta_k$ and $\rho_k \geq \rho$.

Proof. By the definition of the objective function of the model q_k, there exists a point $\bar{\mathbf{x}}_k \in \mathbb{R}^n$ on the line segment $[\mathbf{x}_k, \mathbf{x}_k + \mathbf{s}_k]$ such that

$$\begin{aligned} q_k(\mathbf{s}_k) - f(\mathbf{x}_k + \mathbf{s}_k) &= \big(\mathbf{g}_k - \mathbf{g}(\bar{\mathbf{x}}_k)\big)^T \mathbf{s}_k + \frac{1}{2}\mathbf{s}_k^T \mathbf{H}_k \mathbf{s}_k \\ &\geq -\|\mathbf{g}_k - \mathbf{g}(\bar{\mathbf{x}}_k)\|_2 \|\mathbf{s}_k\|_2 - \frac{1}{2}\|\mathbf{H}_k\|_2 \|\mathbf{s}_k\|_2^2. \end{aligned} \tag{A.11}$$

Therefore,

$$\begin{aligned} f_k - f(\mathbf{x}_k + \mathbf{s}_k) &= f_k - q_k(\mathbf{s}_k) + q_k(\mathbf{s}_k) - f(\mathbf{x}_k + \mathbf{s}_k) \\ &\geq \frac{1}{2}\mathbf{s}_k^T \mathbf{H}_k \mathbf{s}_k + \lambda_k \|\mathbf{s}_k\|_2^2 - \|\mathbf{g}_k - \mathbf{g}(\bar{\mathbf{x}}_k)\|_2 \|\mathbf{s}_k\|_2 - \frac{1}{2}\|\mathbf{H}_k\|_2 \|\mathbf{s}_k\|_2^2 \\ &\geq -\|\mathbf{H}_k\|_2 \|\mathbf{s}_k\|_2^2 + \lambda_k \|\mathbf{s}_k\|_2^2 - g_{Lip}\|\mathbf{s}_k\|_2^2 \\ &\geq (\lambda_k - g_{Lip} - H_{max})\|\mathbf{s}_k\|_2^2 \\ &\geq \rho\|\mathbf{s}_k\|_2^3. \end{aligned}$$

Here the first inequality holds from Lemma A.1 and expression (A.11). The result $\|\mathbf{s}_k\| = \delta_k$ follows directly from (A.10) and (A.4). ∎

We now use the previous results to show that if from some iteration onward, all the steps are contraction steps, then the sequence of trust region radii converge to zero, and the sequence of dual variables converge to infinity.

Lemma A.6. *If $k \in \mathcal{C}$ for all $k \geq k_0$, then $\{\delta_k\} \to 0$ and $\{\lambda_k\} \to \infty$.*

Proof. Assume, without loss of generality, that $k \in \mathcal{C}$ for all $k \in \mathbb{N}$. It follows from Lemma A.2 that $\{\delta_k\}$ is monotonically strictly decreasing. Combined with the fact that $\{\delta_k\}$ is bounded below by zero, we have that $\{\delta_k\}$ converges. We may now observe that if step 23 of the CONTRACT subroutine is reached infinitely often, then clearly, $\{\delta_k\} \to 0$. Hence, it follows by the relationship between the trust region radius and its corresponding multiplier that $\{\lambda_k\} \to \infty$. Therefore, let us assume that step 23 of the CONTRACT subroutine does not occur infinitely often, i.e., that there exists $k_{\mathcal{C}} \in \mathbb{N}$ such that step 6, 9, or 21 is reached for all $k \geq k_{\mathcal{C}}$. Consider iteration $k_{\mathcal{C}}$. Steps 2, 4, 13, 15, 18 in the CONTRACT subroutine will set

$$\lambda_{k+1} = \lambda \geq \min\{\lambda_k + \underline{\sigma}\Delta_k, \gamma_\lambda \lambda_k\} > \lambda_k \quad \forall k \geq k_{\mathcal{C}} + 1.$$

Therefore, since $k \in \mathcal{C}$ for all $k \geq k_{\mathcal{C}}$, we have $\mathbf{x}_k = \mathbf{x}_{k_{\mathcal{C}}}$ (and so $\mathcal{X}_k = \mathcal{X}_{k_{\mathcal{C}}}$) for all $k \geq k_{\mathcal{C}}$, which implies that $\{\lambda_k\} \to \infty$. It follows by the relationship between the trust region radius and its corresponding multiplier that $\|\mathbf{s}_k\|_2 = \delta_k \to 0$. ∎

We now prove that the set of accepted steps is infinite.

LEMMA A.7. *The set $\mathcal{A}$ has infinite cardinality.*

Proof. To derive a contradiction, suppose that $|\mathcal{A}| < \infty$. We claim that this implies $|\mathcal{C}| = \infty$. Indeed, if $|\mathcal{C}| < \infty$, then there exist some $k_{\mathcal{E}} \in \mathbb{N}$ such that $k \in \mathcal{E}$ for all $k \geq k_{\mathcal{E}}$, which contradicts Lemma A.4. Thus, $|\mathcal{C}| = \infty$. Combining this with the result of Lemma A.4, we conclude that there exists some $k_{\mathcal{C}} \in \mathbb{N}_+$ such that $k \in \mathcal{C}$ for all $k \geq k_{\mathcal{C}}$. It follows from Lemma A.6 that $\{\|\mathbf{s}_k\|_2\} \leq \{\delta_k\} \to 0$ and $\{\lambda_k\} \to \infty$. In combination with Lemma A.5, we conclude that there exists some $k \geq k_{\mathcal{C}}$ such that $\rho_k \geq \rho$, which contradicts the fact that $k \in \mathcal{C}$ for all $k \geq k_{\mathcal{C}}$. Having arrived at a contradiction under the supposition that $|\mathcal{A}| < \infty$, the result follows. ∎

We now provide an upper bound for the sequence $\{\Delta_k\}$ and the trial steps $\{\mathbf{s}_k\}$. Moreover, we show that the number of $\mathcal{A}_\Delta$ steps computed by the algorithm is finite.

LEMMA A.8. *There exists a scalar constant $\Delta > 0$ and $k_{\mathcal{A}} \in \mathbb{N}$, such that $\Delta_k = \Delta$ for all $k \geq k_{\mathcal{A}}$. Moreover, the set $\mathcal{A}_\Delta$ has finite cardinality, and there exists a scalar constant $s_{max} > 0$ such that $\|\mathbf{s}_k\|_2 \leq s_{max}$ for all $k \in \mathbb{N}$.*

Proof. For all $k \in \mathcal{A}$, we have $\rho_k \geq \rho$, which implies by step 6 of Algorithm 4.1 that

$$f(\mathbf{x}_k) - f(\mathbf{x}_{k+1}) \geq \rho\|\mathbf{s}_k\|_2^3.$$

Combining this with Lemma A.7 and the fact that f is bounded below, it follows that $\{\mathbf{s}_k\}_{k\in\mathcal{A}} \to 0$. In particular, there exists $k_{\mathcal{A}} \in \mathbb{N}$ such that for all $k \in \mathcal{A}$ with $k \geq k_{\mathcal{A}}$, we have

$$\gamma_E\|\mathbf{s}_k\|_2 \leq \Delta_0 \leq \Delta_k, \tag{A.12}$$

where the latter inequality follows from Lemma A.3. Combined with the update in steps 7, 12, and 16 of LC-TRACE, we get

$$\Delta_{k+1} = \Delta_k \ \forall k \geq k_{\mathcal{A}}.$$

This proves the first part of the lemma. The second part also follows from (A.12), which implies that $\|\mathbf{s}_k\|_2 < \Delta_k$ for all $k \in \mathcal{A}$ with $k \geq k_{\mathcal{A}}$. Finally, the last part of the lemma follows from the first part and the fact that Lemma A.3 ensures $\|\mathbf{s}_k\|_2 \leq \delta_k \leq \Delta_k = \Delta$ for all sufficiently large $k \in \mathbb{N}$. ∎

We now show that there exits a uniform upper bound on the term $\|\mathbf{g}_k + \mathbf{A}^T \boldsymbol{\lambda}_k^C\|_2$.

LEMMA A.9. *For all $k \in \mathbb{N}$,*

$$\lambda_k \leq \max\{\lambda_0, \lambda_{max}\} \quad \forall\, k \in \mathbb{N},$$

where $\lambda_{max} \triangleq \max\{g_{Lip} + 2H_{max} + (\rho+1)\Delta + (g_{max})^{1/2},\ \gamma_\lambda(g_{Lip} + H_{max} + \rho\Delta)\}$. *Moreover,*

$$\|\mathbf{g}_k + \mathbf{A}^T \boldsymbol{\lambda}_k^C\|_2 \leq G_{max},$$

where $G_{max} = (H_{max} + \max\{\lambda_0, \lambda_{max}\})\Delta$ is a constant scalar.

Proof. By (A.2) and Lemma A.3,

$$\|\mathbf{g}_k + \mathbf{A}^T \boldsymbol{\lambda}_k^C\|_2 = \|\mathbf{H}_k \mathbf{s}_k + \lambda_k \mathbf{s}_k\|_2 \leq (H_{max} + \lambda_k)\delta_k \leq (H_{max} + \lambda_k)\Delta.$$

Thus, it suffices to find a constant upper bound for λ_k to get the desired result.

If $\|\mathbf{s}_{k+1}\|_2 < \delta_{k+1}$, then by (A.4), $\lambda_{k+1} = 0$. Therefore, we may proceed under the assumption that $\|\mathbf{s}_{k+1}\|_2 = \delta_{k+1}$. Suppose $k \in \mathcal{C}$, then by Lemma A.5, $\lambda_k < g_{Lip} + H_{max} + \rho\Delta$.

If step 3 in the CONTRACT subroutine tests true, we get

$$\begin{aligned} \lambda_{k+1} &\leq \lambda_k + H_{max} + \underline{\sigma}\Delta_k + (\underline{\sigma}\mathcal{X}_k)^{1/2} \\ &\leq g_{Lip} + 2H_{max} + (\rho+1)\Delta + (g_{max})^{1/2}, \end{aligned} \tag{A.13}$$

where the second inequality assumes that $\underline{\sigma} \leq 1$.

Otherwise, if step 3 tests false, we claim that

$$\lambda_{k+1} \leq \gamma_\lambda(g_{Lip} + H_{max} + \rho\Delta). \tag{A.14}$$

To show our claim, we assume the contrary, i.e., $\lambda_{k+1} > \gamma_\lambda(g_{Lip} + H_{max} + \rho\Delta)$. Then the condition of the while loop in step 17 of the CONTRACT subroutine tested true for some $\hat{\mathbf{s}}$ being a solution of $Q_k(\hat{\lambda})$ for $\hat{\lambda} \geq g_{Lip} + H_{max} + \rho\Delta$.

There exist $\hat{\mathbf{x}}$ on the line segment $[\mathbf{x}_k, \mathbf{x}_k + \hat{\mathbf{s}}]$ such that

$$q_k(\hat{\mathbf{s}}) - f(\mathbf{x}_k + \hat{\mathbf{s}}) = \big(\mathbf{g}_k - \mathbf{g}(\hat{\mathbf{x}})\big)^T \mathbf{s}_k + \frac{1}{2}\hat{\mathbf{s}}^T \mathbf{H}_k \hat{\mathbf{s}} \geq -g_{Lip}\|\hat{\mathbf{s}}\|_2^2 - \frac{1}{2}H_{max}\|\hat{\mathbf{s}}\|_2^2. \tag{A.15}$$

Therefore,

$$\begin{aligned} \frac{f_k - f(\mathbf{x}_k + \hat{\mathbf{s}})}{\|\hat{\mathbf{s}}\|_2^3} &= \frac{f_k - q_k(\hat{\mathbf{s}}) + q_k(\hat{\mathbf{s}}) - f(\mathbf{x}_k + \hat{\mathbf{s}})}{\|\hat{\mathbf{s}}\|_2^3} \\ &\geq \frac{-\|\mathbf{H}_k\|_2 + 2\hat{\lambda} - 2g_{Lip} - H_{max}}{2\|\hat{\mathbf{s}}\|_2} \\ &\geq \frac{\hat{\lambda} - g_{Lip} - H_{max}}{\|\hat{\mathbf{s}}\|_2} \\ &\geq \rho, \end{aligned}$$

where the first inequality holds by Lemma A.1 and (A.15). Since

$$\rho_k = \frac{f_k - f(\mathbf{x}_k + \mathbf{s}_k)}{\|\mathbf{s}_k\|_2^3} < \rho,$$

it follows that $\|\hat{\mathbf{s}}\|_2 \neq \|\mathbf{s}_k\|_2$, which contradicts the condition of the while loop in step 17, which tested true for $\hat{\mathbf{s}}$ generated by solving $Q_k(\hat{\lambda})$.

Combining (A.13) and (A.14), we get that for all $k \in \mathcal{C}$

$$\lambda_{k+1} \le \lambda_{max}, \tag{A.16}$$

where $\lambda_{max} \triangleq \max\{g_{Lip} + 2H_{max} + (\rho+1)\Delta + (g_{max})^{1/2},\, \gamma_\lambda(g_{Lip} + H_{max} + \rho\Delta)\}$. Now, suppose that $k \in \mathcal{A} \cup \mathcal{E}$. By Lemma A.3, we have $\|\mathbf{s}_{k+1}\|_2 = \delta_{k+1} \ge \delta_k \ge \|\mathbf{s}_k\|_2$. Hence, by the relationship between the trust region radius and its corresponding multiplier, we obtain

$$\lambda_{k+1} \le \lambda_k. \tag{A.17}$$

Let $k_{\mathcal{C}} \triangleq \min\{k \in \mathbb{N} \,|\, k \in \mathcal{C}\}$ be the first contract step. By (A.17), $\lambda_k \le \lambda_0$ for all $k \le k_{\mathcal{C}}$. Moreover, using (A.16) and (A.17),

$$\lambda_k \le \lambda_{max} \quad \forall\, k > k_{\mathcal{C}}.$$

Combining these results yield

$$\lambda_k \le \max\{\lambda_0, \lambda_{max}\} \quad \forall\, k \in \mathbb{N},$$

which completes the proof. ∎

Notice that in the proof of Lemma A.9, we have shown that there exists a uniform upper bound for the dual variables λ_k. Our next result shows that the ratio $\frac{\mathcal{X}_k}{\|\mathbf{s}\|_2}$ is upper bounded by $C_{min} + \lambda_k$, where C_{min} is a scalar constant.

Lemma A.10. *For any $k \in \mathbb{N}$, it holds that*

$$\mathcal{X}_k \le (C_{min} + \lambda_k)\|\mathbf{s}_k\|_2, \tag{A.18}$$

where $C_{min} \triangleq H_{max} + G_{max} + g_{max}$ is a scalar constant.

Proof. Let $\xi_{k,1}$ be the largest singular value of $\mathbf{H}_k$. For all $\mathbf{d}$ satisfying $\mathbf{A}\mathbf{d} \le \mathbf{b} - \mathbf{A}\mathbf{x}_k$, we have

$$\begin{aligned}\mathbf{g}_k^T\mathbf{d} &= -\mathbf{d}^T(\mathbf{H}_k + \lambda_k\mathbf{I})\mathbf{s}_k^T - (\boldsymbol{\lambda}_k^C)^T\mathbf{A}\mathbf{d}\\ &\ge -\mathbf{d}^T(\mathbf{H}_k + \lambda_k\mathbf{I})\mathbf{s}_k^T - (\boldsymbol{\lambda}_k^C)^T\mathbf{A}\mathbf{s}_k\\ &\ge -(\xi_{k,1} + \lambda_k)\|\mathbf{d}\|_2\|\mathbf{s}_k\|_2 - (\boldsymbol{\lambda}_k^C)^T\mathbf{A}\mathbf{s}_k,\end{aligned} \tag{A.19}$$

where the first equality holds by (A.2), and the first inequality holds by complementary slackness (A.3). Minimizing over all such $\mathbf{d}$, we obtain

$$\begin{aligned}\min_{\mathbf{A}\mathbf{d}\le\mathbf{b}-\mathbf{A}\mathbf{x}_k,\ \|\mathbf{d}\|\le 1} \mathbf{g}_k^T\mathbf{d} &\ge -(\xi_{k,1} + \lambda_k)\|\mathbf{s}_k\|_2 - \|(\boldsymbol{\lambda}_k^C)^T\mathbf{A}\|_2\|\mathbf{s}_k\|_2\\ &\ge -(\xi_{k,1} + \lambda_k)\|\mathbf{s}_k\|_2 - (\|g_k + (\boldsymbol{\lambda}_k^C)^T\mathbf{A}\|_2 + \|g_k\|_2)\|\mathbf{s}_k\|_2\\ &\ge -(H_{max} + \lambda_k)\|\mathbf{s}_k\|_2 - (G_{max} + g_{max})\|\mathbf{s}_k\|_2,\end{aligned}$$

where the last inequality uses Lemma A.9. Then definition of $\mathcal{X}_k$ yields

$$\begin{aligned}\mathcal{X}_k &\le (H_{max} + \lambda_k + G_{max} + g_{max})\|\mathbf{s}_k\|_2\\ &= (C_{min} + \lambda_k)\|\mathbf{s}_k\|_2,\end{aligned}$$

where $C_{min} \triangleq H_{max} + G_{max} + g_{max}$ is a scalar constant. ∎

We now show that the limit inferior of stationarity measure $\mathcal{X}_k$ is equal to zero.

LEMMA A.11. *There holds*

$$\liminf_{k\in\mathbb{N},\,k\to\infty} \mathcal{X}_k = 0.$$

Proof. Suppose the contrary, that there exists a scalar constant $\mathcal{X}_{min} > 0$ such that $\mathcal{X}_k \geq \mathcal{X}_{min}$ for all $k \in \mathbb{N}$. Then by Lemmas A.9 and A.10, for scalar

$$s_{min} = \frac{\mathcal{X}_{min}}{C_{min} + \max\{\lambda_{max}, \lambda_0\}},$$

we have that $\|\mathbf{s}_k\|_2 \geq s_{min} > 0$ for all $k \in \mathbb{N}$. Moreover, for all $k \in \mathcal{A}$ we have $f_k - f_{k+1} \geq \rho\|\mathbf{s}_k\|_2^3 > 0$. Given the lower boundedness of f and Lemma A.7 that ensures infinite cardinality of set $\mathcal{A}$, we have $\{\mathbf{s}_k\}_{k\in\mathcal{A}} \to 0$. This contradicts the existence of $s_{min} > 0$. ∎

THEOREM A.12. *Under Assumptions* 4.1, 4.2, *and* 4.3, *it holds that*

$$\lim_{k\in\mathbb{N},\,k\to\infty} \mathcal{X}_k = 0. \tag{A.20}$$

Proof. Suppose the contrary, that (A.20) does not hold. Combined with Lemmas A.7 and A.11, it implies that there exist an infinite subsequence $\{t_i\} \subseteq \mathcal{A}$ (indexed over $i \in \mathbb{N}$) such that $\mathcal{X}_{t_i} \geq 2\epsilon_{\mathcal{X}}$ for some $\epsilon_{\mathcal{X}} > 0$ and all $i \in \mathbb{N}$. Additionally, Lemmas A.7 and A.11 imply that there exist an infinite subsequence $\{l_i\} \subseteq \mathcal{A}$ such that

$$\mathcal{X}_k \geq \epsilon_{\mathcal{X}} \text{ and } \mathcal{X}_{l_i} < \epsilon_{\mathcal{X}} \quad \forall\, i \in \mathbb{N},\ k \in \mathbb{N},\ t_i \leq k < l_i. \tag{A.21}$$

We claim that for all $k \in \mathbb{N}_+$, the trial step $\mathbf{s}_k$ satisfies the following:

$$\|\mathbf{s}_k\|_2 \geq \min\Big\{\delta_k, \frac{\mathcal{X}_k}{C_{min}}\Big\}. \tag{A.22}$$

The proof of this claim follows directly from Lemma A.10. If $\|\mathbf{s}_k\|_2 = \delta_k$, the result trivially holds. Otherwise, using KKT condition (A.4), $\lambda_k = 0$, which proves our claim when combined with Lemma A.10.

We now restrict our attention to indices in the infinite index set

$$\mathcal{K} \triangleq \{k \in \mathcal{A} : t_i \leq k < l_i \text{ for some } i \in \mathbb{N}\}.$$

Observe from (A.21) and (A.22) that

$$f_k - f_{k+1} \geq \rho\|\mathbf{s}_k\|_2^3 \geq \rho\Big(\min\Big\{\delta_k, \frac{\epsilon_{\mathcal{X}}}{C_{min}}\Big\}\Big)^3. \tag{A.23}$$

Since $\{f_k\}$ is monotonically decreasing and bounded below, we know that $f_k \to \underline{f}$ for some $\underline{f} \in \mathbb{R}$. When combined with (A.23), we obtain

$$\lim_{k\in\mathcal{K},k\to\infty} \delta_k = 0. \tag{A.24}$$

Using this fact and Lemma A.1, we have for all sufficiently large $k \in \mathcal{K}$ that

$$\begin{aligned} f_k - f_{k+1} &= f_k - q_k(\mathbf{s}_k) + q_k(\mathbf{s}_k) - f_{k+1} \\ &\geq C\mathcal{X}_k \min\left\{\delta_k, \frac{\mathcal{X}_k}{\|\mathbf{H}_k\|_2}, 1\right\} - (g_{Lip} + \frac{1}{2}H_{max})\|\mathbf{s}_k\|_2^2 \\ &\geq C\epsilon_{\mathcal{X}} \min\left\{\delta_k, \frac{\epsilon_{\mathcal{X}}}{H_{max}}, 1\right\} - (g_{Lip} + \frac{1}{2}H_{max})\|\mathbf{s}_k\|_2^2 \\ &\geq C\epsilon_{\mathcal{X}}\delta_k - (g_{Lip} + \frac{1}{2}H_{max})\delta_k^2 \\ &\geq \frac{C}{2}\epsilon_{\mathcal{X}}\delta_k. \end{aligned}$$

Consequently, for all sufficiently large $i \in \mathbb{N}$, we have

$$\begin{aligned} \|\mathbf{x}_{t_i} - \mathbf{x}_{l_i}\|_2 &\leq \sum_{k\in\mathcal{K},k=t_i}^{l_i-1} \|\mathbf{x}_k - \mathbf{x}_{k+1}\|_2 \\ &\leq \sum_{k\in\mathcal{K},k=t_i}^{l_i-1} \delta_k \leq \sum_{k\in\mathcal{K},k=t_i}^{l_i-1} \frac{2}{C\epsilon_{\mathcal{X}}}(f_k - f_{k+1}) = \frac{2}{C\epsilon_{\mathcal{X}}}(f_{t_i} - f_{l_i}). \end{aligned}$$

Since $\{f_{t_i} - f_{l_i}\} \to 0$, we get $\{\|\mathbf{x}_{t_i} - \mathbf{x}_{l_i}\|_2\} \to 0$, which, in turn, implies that $\{\mathcal{X}_{t_i} - \mathcal{X}_{l_i}\} \to 0$. This contradicts (A.21). ∎

A.2. Proof of Theorem 4.5. In this section we show that the number of iterations required to reach an ϵ-first-order stationary point is $\mathcal{O}(\epsilon^{-3/2}\log^3\epsilon^{-1})$. To that end, we start by providing a bound on the ratio $\lambda_{k+1}/\|\mathbf{s}_{k+1}\|_2$ when $k \in \mathcal{C}$.

LEMMA A.13. *Suppose that $k \in \mathcal{C}$. Then, the following hold:*

- *If step 6, 9, or 21 of Algorithm 4.2 is reached, then*

$$\underline{\sigma} \leq \frac{\lambda_{k+1}}{\|\mathbf{s}_{k+1}\|_2} \leq \max\left\{\overline{\sigma}, \gamma_\lambda(H_{Lip} + 2\rho)\right\}.$$

- *If step 23 of Algorithm 4.2 is reached, then*

$$\frac{\lambda_{k+1}}{\|\mathbf{s}_{k+1}\|_2} \leq \max\left\{\overline{\sigma}, \gamma_\lambda(H_{Lip} + 2\rho)\right\}.$$

Proof. Let $k \in \mathcal{C}$ and consider the three possible cases. The first two correspond to situations in which the conditions in step 3 in the CONTRACT subroutine tests true.

- Suppose that step 6 is reached. Then, $\delta_{k+1} = \|\mathbf{s}\|_2$, where $(\lambda, \mathbf{s})$ is computed in step 4. It follows that step 22 in Algorithm 4.1 will then produce the primal-dual pair $(\mathbf{s}_{k+1}, \lambda_{k+1}) = (\mathbf{s}, \lambda)$ with $\lambda > 0$. Since the condition in step 5 tested true, we have

$$\underline{\sigma} \leq \frac{\lambda_k + \underline{\sigma}\Delta_k}{\Delta_k} \leq \frac{\lambda_{k+1}}{\|\mathbf{s}_{k+1}\|_2} = \frac{\lambda}{\|\mathbf{s}\|_2} \leq \overline{\sigma}, \tag{A.25}$$

  where the second inequality holds since $\|\mathbf{s}_{k+1}\|_2 = \delta_{k+1} \leq \|\mathbf{s}_k\|_2 \leq \Delta_k$.
- Suppose that step 9 is reached. Then, $\delta_{k+1} = \|\bar{\mathbf{s}}\|_2$, where $(\bar{\lambda}, \bar{\mathbf{s}})$ is computed in step 2. Similar to the previous case, it follows that step 22 in Algorithm 4.1

will produce the primal-dual pair $(\mathbf{s}_{k+1}, \lambda_{k+1}) = (\bar{\mathbf{s}}, \bar{\lambda})$ with $\bar{\lambda} = \lambda_k + \underline{\sigma}\Delta_k$. We first show that $\|\bar{\mathbf{s}}\|_2 \geq \underline{\sigma}$. Assume the contrary, then by Lemma A.10 and the fact that $\mathcal{X}_{k+1} = \mathcal{X}_k$ for all $k \in \mathcal{C}$,

$$\begin{aligned}\mathcal{X}_k &\leq (C_{min} + \max\{\lambda_{max}, \lambda_0\})\|\bar{\mathbf{s}}\|_2 \\ &\leq (C_{min} + \max\{\lambda_{max}, \lambda_0\})\underline{\sigma} \\ &= (C_{min} + \max\{\lambda_{max}, \lambda_0\})\frac{\epsilon}{M},\end{aligned}$$

where the last equality uses the definition (4.10) of $\underline{\sigma}$. Now by letting

$$M = C_{min} + \max\{\lambda_0, \lambda_{max}\}, \tag{A.26}$$

we obtain

$$\mathcal{X}_k \leq \epsilon,$$

which contradicts the assumption that the algorithm did not yet terminate at k. Note that by the definition of C_{min} in Lemma A.10 and λ_{max} in Lemma A.9, the constant M can be defined in terms of H_{max}, g_{max}, H_{Lip}, g_{Lip}, and Δ. Here, H_{max}, g_{max}, H_{Lip}, g_{Lip} exist by Assumptions 4.2 and 4.3, and Δ exists by Lemma A.8. Combined with Lemma A.3 we obtain

$$\underline{\sigma} \leq \|\bar{\mathbf{s}}\|_2 \leq \|\mathbf{s}_k\|_2 \leq \Delta_k.$$

Therefore,

$$\underline{\sigma} \leq \frac{\lambda_k + \underline{\sigma}\Delta_k}{\Delta_k} \leq \frac{\bar{\lambda}}{\|\bar{\mathbf{s}}\|_2} \leq \frac{\lambda_k + \underline{\sigma}\Delta_k}{\underline{\sigma}} \leq \frac{\underline{\sigma}(\|\mathbf{s}_k\|_2 + \Delta_k)}{\underline{\sigma}} \leq 2\Delta_k \leq \bar{\sigma}, \tag{A.27}$$

where the fourth inequality holds by the condition of step 3 and the last inequality holds by Lemmas A.3 and A.8 and the definition of $\bar{\sigma}$ in 4.10.

The other case corresponds to situations in which the condition in step 3 tests false. It follows by steps 2 and 3 that

$$\underline{\sigma} \leq \frac{\lambda}{\|\mathbf{s}\|_2}. \tag{A.28}$$

In these cases and using the argument of Lemma A.9, we claim that

$$\lambda \leq \gamma_\lambda \left(H_{Lip} + 2\rho\right) \|\mathbf{s}\|_2. \tag{A.29}$$

To show our claim, we assume the contrary, i.e., $\lambda > \gamma_\lambda \left(H_{Lip} + 2\rho\right) \|\mathbf{s}\|_2$. Then the condition of the while loop in step 17 tested true for some $\hat{\mathbf{s}}$ computed by solving $Q_k(\hat{\lambda})$ for

$$\hat{\lambda} \geq (H_{Lip} + 2\rho)\|\hat{\mathbf{s}}\|_2. \tag{A.30}$$

First note that

$$Q_k(\hat{\lambda}) = f_k + \mathbf{g}_k^T\hat{\mathbf{s}} + \frac{1}{2}\hat{\mathbf{s}}^T\mathbf{H}_k\hat{\mathbf{s}} + \frac{1}{2}\hat{\lambda}\|\hat{\mathbf{s}}\|_2^2 \leq f_k,$$

where the inequality holds since $\mathbf{0}$ is a feasible solution in the optimization problem $Q_k(\hat{\lambda})$. Hence,

$$f_k - q_k(\hat{\mathbf{s}}) = -\mathbf{g}_k^T\hat{\mathbf{s}} - \frac{1}{2}\hat{\mathbf{s}}^T\mathbf{H}_k\hat{\mathbf{s}} \geq \frac{1}{2}\hat{\lambda}\|\hat{\mathbf{s}}\|_2^2. \tag{A.31}$$

There exists $\hat{\mathbf{x}}$ on the line segment $[\mathbf{x}_k, \mathbf{x}_k + \hat{\mathbf{s}}]$ such that

$$\begin{aligned} q_k(\hat{\mathbf{s}}) - f(\mathbf{x}_k + \hat{\mathbf{s}}) &= f_k + \mathbf{g}_k^T\hat{\mathbf{s}} + \frac{1}{2}\hat{\mathbf{s}}^T\mathbf{H}_k\hat{\mathbf{s}} - f_k - \mathbf{g}_k^T\hat{\mathbf{s}} - \frac{1}{2}\hat{\mathbf{s}}^T\mathbf{H}(\hat{\mathbf{x}})\hat{\mathbf{s}} \\ &= \frac{1}{2}\hat{\mathbf{s}}^T\big(\mathbf{H}_k - \mathbf{H}(\hat{\mathbf{x}})\big)\hat{\mathbf{s}} \geq -\frac{1}{2}H_{Lip}\|\hat{\mathbf{s}}\|_2^3. \end{aligned} \tag{A.32}$$

Therefore,

$$\begin{aligned} \frac{f(\mathbf{x}_k) - f(\mathbf{x}_k + \hat{\mathbf{s}})}{\|\hat{\mathbf{s}}\|_2^3} &= \frac{f(\mathbf{x}_k) - q_k(\hat{\mathbf{s}}) + q_k(\hat{\mathbf{s}}) - f(\mathbf{x}_k + \hat{\mathbf{s}})}{\|\hat{\mathbf{s}}\|_2^3} \\ &\geq \frac{0.5\hat{\lambda}\|\hat{\mathbf{s}}\|_2^2 - 0.5H_{Lip}\|\hat{\mathbf{s}}\|_2^3}{\|\hat{\mathbf{s}}\|_2^3} \\ &\geq \rho, \end{aligned}$$

where the first inequality holds due (A.31) and (A.32), and the last inequality holds due to (A.30). However,

$$\rho_k = \frac{f_k - f(\mathbf{x}_k + \mathbf{s}_k)}{\|\mathbf{s}_k\|_2^3} < \rho.$$

It follows that $\|\hat{\mathbf{s}}\|_2 \neq \|\mathbf{s}_k\|_2$, which contradicts the condition of the while loop in step 17 for $\hat{\mathbf{s}}$ computed by solving $Q_k(\hat{\lambda})$.

- Suppose that step 21 is reached. Then, $\delta_{k+1} = \|\mathbf{s}\|_2$. It follows that step 20 in Algorithm 4.1 will produce the primal-dual pair $(\mathbf{s}_{k+1}, \lambda_{k+1})$ solving Q_{k+1} such that $\mathbf{s}_{k+1} = \mathbf{s}$ and $\lambda_{k+1} = \lambda$. In conjunction with (A.28), (A.29), and the condition in step 20 of the CONTRACT subroutine, we observe that

$$\underline{\sigma} \leq \frac{\lambda_{k+1}}{\|\mathbf{s}_{k+1}\|_2} = \frac{\lambda}{\|\mathbf{s}\|_2} \leq \gamma_\lambda(H_{Lip} + 2\rho). \tag{A.33}$$

- Suppose that step 23 is reached. Then, $\delta_{k+1} = \gamma_C\|\mathbf{s}_k\|_2$. It follows that step 20 in Algorithm 4.1 will produce the primal-dual pair $(\mathbf{s}_{k+1}, \lambda_{k+1}) = (\hat{\mathbf{s}}, \hat{\lambda})$. If $\|\hat{\mathbf{s}}\|_2 < \delta_{k+1} = \gamma_C\|\mathbf{s}_k\|_2$, then $\lambda_{k+1} = 0$. Otherwise,

$$\|\hat{\mathbf{s}}\|_2 = \delta_{k+1} = \gamma_C\|\mathbf{s}_k\|_2 \geq \|\mathbf{s}\|_2,$$

where last inequality holds by the condition of step 20, and

$$0 < \hat{\lambda} \leq \lambda,$$

which holds due to the inverse relationship of λ and $\|\mathbf{s}\|_2$. Here λ and $\mathbf{s}$ are computed in step 18, for which (A.29) holds. Hence, combined with (A.29) we obtain

$$\frac{\lambda_{k+1}}{\|\mathbf{s}_{k+1}\|_2} = \frac{\hat{\lambda}}{\|\hat{\mathbf{s}}\|_2} \leq \frac{\lambda}{\|\mathbf{s}\|_2} \leq \gamma_\lambda(H_{Lip} + 2\rho).$$

The result follows since we have obtained the desired inequalities in all cases. ∎

We now provide an upper bound for the sequence $\{\sigma_{max}\}$.

LEMMA A.14. *There exists a scalar constant $\sigma_{max} > 0$ such that*

$$\sigma_k \leq \sigma_{max} \quad \forall\, k \in \mathbb{N}.$$

Proof. First note that by Lemma A.8, the cardinality of the set $\mathcal{A}_\Delta$ is finite. Hence, there exist $k_\mathcal{A} \in \mathbb{N}$ such that $k \notin \mathcal{A}_\Delta$ for all $k \geq k_\mathcal{A}$. We continue by showing that σ_k is upper bounded for all $k \geq k_\mathcal{A}$. Consider the following three cases:

- If $k \in \mathcal{A}_\sigma$, then by definition $\lambda_k \leq \sigma_k \|\mathbf{s}_k\|_2$, which implies by step 9 of Algorithm 4.1 that

$$\sigma_{k+1} = \max\{\sigma_k, \lambda_k/\|\mathbf{s}_k\|_2\} = \sigma_k.$$

- If $k \in \mathcal{C}$, by step 22 of Algorithm 4.1 and Lemma A.13, it follows that

$$\sigma_{k+1} = \max\Big\{\sigma_k, \frac{\lambda_{k+1}}{\|\mathbf{s}_{k+1}\|_2}\Big\} \leq \max\Big\{\sigma_k,\, \overline{\sigma},\, \gamma_\lambda(H_{Lip} + 2\rho)\Big\}. \tag{A.34}$$

- If $k \in \mathcal{E}$, then step 18 of Algorithm 4.1 implies that $\sigma_{k+1} = \sigma_k$.

Combining the results of these three cases, the desired result follows. ∎

We now establish an upper bound on the norm trial steps $\mathbf{s}_k$ when $k \in \mathcal{A}_\sigma$.

LEMMA A.15. *For all $k \in \mathcal{A}_\sigma$, the accepted step $\mathbf{s}_k$ satisfies*

$$\|\mathbf{s}_k\|_2 \geq (H_{Lip} + \sigma_{max})^{-1/2} \mathcal{X}_{k+1}^{1/2}.$$

Proof. For all $k \in \mathcal{A}_\sigma$, there exists $\bar{\mathbf{x}}_k$ on the line segment $[\mathbf{x}_k, \mathbf{x}_k + \mathbf{s}_k]$ such that

$$\begin{aligned}
\mathbf{g}_{k+1}^T\mathbf{d} &= \mathbf{g}_{k+1}^T\mathbf{d} - \mathbf{g}_k^T\mathbf{d} - \mathbf{d}^T(\mathbf{H}_k + \lambda_k\mathbf{I})\mathbf{s}_k - \mathbf{d}^T\mathbf{A}^T\boldsymbol{\lambda}_k^C \\
&= \big(\mathbf{g}(\mathbf{x}_k + \mathbf{s}_k) - \mathbf{g}_k\big)^T\mathbf{d} - \mathbf{d}^T(\mathbf{H}_k + \lambda_k\mathbf{I})\mathbf{s}_k - \mathbf{d}^T\mathbf{A}^T\boldsymbol{\lambda}_k^C \\
&= \mathbf{d}^T(\mathbf{H}(\bar{\mathbf{x}}_k) - \mathbf{H}_k)\mathbf{s}_k - \lambda_k\mathbf{d}^T\mathbf{s}_k - \mathbf{d}^T\mathbf{A}^T\boldsymbol{\lambda}_k^C \\
&\geq -H_{Lip}\|\mathbf{s}_k\|_2^2\|\mathbf{d}\|_2 - \lambda_k\|\mathbf{s}_k\|_2\|\mathbf{d}\|_2 - \mathbf{d}^T\mathbf{A}^T\boldsymbol{\lambda}_k^C \\
&\geq -H_{Lip}\|\mathbf{s}_k\|_2^2 - \sigma_k\|\mathbf{s}_k\|_2^2 - \mathbf{d}^T\mathbf{A}^T\boldsymbol{\lambda}_k^C \quad \forall \mathbf{d} \text{ with } \|\mathbf{d}\|_2 \leq 1,
\end{aligned}$$

where the first equation follows from (A.2) and the last inequality follows since $\lambda_k \leq \sigma_k\|\mathbf{s}_k\|_2$ for all $k \in \mathcal{A}_\sigma$. Thus

$$\min_{\text{s.t. } d \in \mathcal{D}_{k+1}} \mathbf{g}_{k+1}^T\mathbf{d} \geq -H_{Lip}\|\mathbf{s}_k\|_2^2 - \sigma_k\|\mathbf{s}_k\|_2^2 - \max_{\text{s.t. } d \in \mathcal{D}_{k+1}} \mathbf{d}^T\mathbf{A}^T\boldsymbol{\lambda}_k^C, \tag{A.35}$$

where $\mathcal{D}_{k+1} \triangleq \{\mathbf{d} \in \mathbb{R}^n \,|\, \|\mathbf{d}\|_2 \leq 1;\;\; \mathbf{A}\mathbf{d} \leq \mathbf{b} - \mathbf{A}\mathbf{x}_{k+1}\}$. Note that since $k \in \mathcal{A}_\sigma$ the updated step will be $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{s}_k$. Now, let $\mathcal{I}_k \triangleq \{i \,|\, \mathbf{a}_i^T\mathbf{s}_k = \mathbf{b}_i - \mathbf{a}_i^T\mathbf{x}_k\}$, then

$$\mathbf{A}_{\mathcal{I}_k}\mathbf{d} \leq \mathbf{b}_{\mathcal{I}_k} - \mathbf{A}_{\mathcal{I}_k}\mathbf{x}_{k+1} = \mathbf{b}_{\mathcal{I}_k} - \mathbf{A}_{\mathcal{I}_k}\mathbf{x}_k - \mathbf{A}_{\mathcal{I}_k}\mathbf{s}_k = \mathbf{0} \Rightarrow (\boldsymbol{\lambda}_k^C)^T\mathbf{A}\mathbf{d} = (\boldsymbol{\lambda}_k^C)_{\mathcal{I}_k}^T\mathbf{A}_{\mathcal{I}_k}\mathbf{d} \leq 0$$

for all $\mathbf{d} \in \mathcal{D}_{k+1}$. Substituting in (A.35), we obtain

$$\mathcal{X}_{k+1} \leq H_{Lip}\|\mathbf{s}_k\|_2^2 + \sigma_k\|\mathbf{s}_k\|_2^2,$$

which along with Lemma A.14, implies the result. ∎

We are now ready to compute a worst-case upper bound on the number of steps in $\mathcal{A}_\sigma$ for which the first-order criticality measure $\mathcal{X}_k$ is larger than a prescribed $\epsilon > 0$.

LEMMA A.16. *For a scalar $\epsilon \in (0, \infty)$, the total number of elements in the index set*

$$\mathcal{K}_\epsilon \triangleq \{k \in \mathbb{N}_+ : k \geq 1;\ (k-1) \in \mathcal{A}_\sigma;\ \mathcal{X}_k > \epsilon\}$$

is at most

$$\left\lceil \left(\frac{f_0 - f_{min}}{\rho (H_{Lip} + \sigma_{max})^{-3/2}} \right) \epsilon^{-3/2} \right\rceil \triangleq K_\sigma(\epsilon) \geq 0. \tag{A.36}$$

Proof. By Lemma A.15, we have for all $k \in \mathcal{K}_\epsilon$ that

$$f_{k-1} - f_k \geq \rho \|\mathbf{s}_{k-1}\|_2^3 \geq \rho (H_{Lip} + \sigma_{max})^{-3/2} \mathcal{X}_k^{3/2} \geq \rho (H_{Lip} + \sigma_{max})^{-3/2} \epsilon^{3/2}.$$

In addition, we have by Theorem A.12 that $|\mathcal{K}_\epsilon| < \infty$. Hence, we have that

$$f_0 - f_{min} \geq \sum_{k \in \mathcal{K}_\epsilon} (f_{k-1} - f_k) \geq |\mathcal{K}_\epsilon| \rho (H_{Lip} + \sigma_{max})^{-3/2} \epsilon^{3/2}.$$

Rearranging this inequality to yield an upper bound for $|\mathcal{K}_\epsilon|$ we obtain the desired result. ∎

It remains to compute a worst-case upper bound for the number of iterations in $\mathcal{A}_\Delta$ for which $\mathcal{X}_k$ is larger than a prescribed $\epsilon > 0$ and the number of contraction and expansion iterations that may occur between two acceptance steps. We compute these bounds separately in Lemmas A.17 and A.20.

LEMMA A.17. *The cardinality of the set $\mathcal{A}_\Delta$ is upper-bounded by*

$$\left\lceil \frac{f_0 - f_{min}}{\rho \Delta_0^3} \right\rceil \triangleq K_\Delta \geq 0. \tag{A.37}$$

Proof. For all $k \in \mathcal{A}_\Delta$, it follows by Lemma A.3 that

$$f_k - f_{k+1} \geq \rho \|\mathbf{s}_k\|_2^3 = \rho \Delta_k^3 \geq \rho \Delta_0^3.$$

Hence, we have that

$$f_0 - f_{min} \geq \sum_{k=0}^{\infty} (f_k - f_{k+1}) \geq \sum_{k \in \mathcal{A}_\Delta} (f_k - f_{k+1}) \geq |\mathcal{A}_\Delta| \rho \Delta_0^3,$$

from which the desired result follows. ∎

So far, we have obtained upper-bound on the number of acceptance iterations. To obtain upper-bounds on the number of contraction and expansion iterations that may occur until the next accepted step, let us define, for a given $\hat{k} \in \mathcal{A}$,

$$k_{\mathcal{A}}(\hat{k}) \triangleq \min\{k \in \mathcal{A} : k > \hat{k}\},$$

$$\mathcal{I}(\hat{k}) \triangleq \{k \in \mathbb{N}_+ : \hat{k} < k < k_{\mathcal{A}}(\hat{k})\}.$$

Using this notation, the following result shows that the number of expansion iterations between the first iteration and the first accepted step, or between consecutive accepted steps, is never greater than one. Moreover, when such an expansion iteration occurs, it must take place immediately.

LEMMA A.18. *For any $\hat{k} \in \mathbb{N}_+$, if $\hat{k} \in \mathcal{A}$, then $\mathcal{E} \cap \mathcal{I}(\hat{k}) \subseteq \{\hat{k}+1\}$.*

Proof. By the definition of $k_{\mathcal{A}}(\hat{k})$, we have under the conditions of the lemma that $\mathcal{I}(\hat{k}) \cap \mathcal{A} = \emptyset$, which means that $\mathcal{I}(\hat{k}) \subseteq \mathcal{C} \cup \mathcal{E}$. It then follows from Lemma A.4 that $\mathcal{E} \cap \mathcal{I}(\hat{k}) \subseteq \{\hat{k}+1\}$, as desired. ∎

LEMMA A.19. *For any $k \in \mathbb{N}_+$, if $k \in \mathcal{C}$ and step* 21 *in Algorithm* 4.2 *is reached, then*

$$\frac{\lambda_{k+1}}{\|\mathbf{s}_{k+1}\|_2} \geq \gamma_\lambda\Big(\frac{\lambda_k}{\|\mathbf{s}_k\|_2}\Big).$$

Proof. If step 21 is reached, then $\|\mathbf{s}\|_2 \geq \gamma_C\|\mathbf{s}_k\|_2$. It follows that step 22 in Algorithm 4.1 will produce the primal-dual pair $(\mathbf{s}_{k+1}, \lambda_{k+1})$ solving Q_{k+1} such that $\|\mathbf{s}_{k+1}\|_2 = \delta_{k+1} < \|\mathbf{s}_k\|_2 = \delta_k$ and $\lambda_{k+1} \geq \gamma_\lambda \lambda_k$, i.e.,

$$\frac{\lambda_{k+1}}{\|\mathbf{s}_{k+1}\|_2} \geq \frac{\gamma_\lambda \lambda_k}{\|\mathbf{s}_k\|_2}. \tag{A.38}$$ ∎

LEMMA A.20. *Assume that Assumptions* 4.1*,* 4.2*, and* 4.3 *hold. For any $\hat{k} \in \mathbb{N}_+$, if $\hat{k} \in \mathcal{A}$, then*

$$|\mathcal{C} \cap \mathcal{I}(\hat{k})| \leq K_{\mathcal{C}},$$

where

$$K_{\mathcal{C}} \triangleq 1 + \left\lceil \left(2 + \frac{1}{\log(\gamma_\lambda)} \log\left(\frac{\sigma_{max}}{\underline{\sigma}}\right)\right) \frac{\log\left(\epsilon^{-1}\Delta\left(C_{min} + \max\{\lambda_{max}, \lambda_0\}\right)\right)}{\log(1/\gamma_C)} \right\rceil.$$

Proof. The result holds trivially if $|\mathcal{C} \cap \mathcal{I}(\hat{k})| = 0$. Thus, we may assume $|\mathcal{C} \cap \mathcal{I}(\hat{k})| \geq 1$. To proceed with our proof, we first claim that the number of iterations $k \in \mathcal{C} \cap \mathcal{I}(\hat{k})$ with step 23 in the CONTRACT subroutine reached, which we denote by $K_{\mathcal{C},1}$, satisfies

$$K_{\mathcal{C},1} \leq \frac{\log\left(\epsilon^{-1}\Delta(C_{min} + \max\{\lambda_{max}, \lambda_0\})\right)}{\log(1/\gamma_C)}.$$

By Lemma A.10 and the assumption that $\mathcal{X}_{k_{\mathcal{A}}(\hat{k})} \geq \epsilon$ (optimality not reached yet),

$$(C_{min} + \max\{\lambda_{max}, \lambda_0\})^{-1}\epsilon \leq \|\mathbf{s}_{k_{\mathcal{A}}(\hat{k})-1}\|_2 \leq \delta_{k_{\mathcal{A}}(\hat{k})-1} \leq \gamma_C^{K_{\mathcal{C},1}}\Delta,$$

where the last inequality holds by Lemmas A.3 and A.2 and the fact that each time that step 23 is reached the radius of the trust region is multiplied by γ_C. The proof of the claim follows by rearranging the inequality to yield an upper bound for $K_{\mathcal{C},1}$.

It remains to compute the number of iterations between steps in $k \in \mathcal{C} \cap \mathcal{I}(\hat{k})$ for which step 23 is reached. For a given $\hat{k} \in \mathcal{A}$, we define

$$\mathcal{I}_{\mathcal{C}}(\hat{k}) \triangleq \{k \in \mathcal{C} \cap \mathcal{I}(\hat{k})\colon \text{ step 23 is reached in Algorithm 4.2}\},$$

which correspond to indices in $\mathcal{C} \cap \mathcal{I}(\hat{k})$ for which step 23 is reached. Let $k_{\mathcal{C},1}(\hat{k})$ and $k_{\mathcal{C},2}(\hat{k})$ be any two consecutive indices in $\mathcal{I}_{\mathcal{C}}(\hat{k})$ with $k_{\mathcal{C},2}(\hat{k}) - k_{\mathcal{C},1}(\hat{k}) > 2$. By Lemma A.13, we have

$$\frac{\lambda_k}{\|\mathbf{s}_k\|_2} \geq \underline{\sigma} \quad \forall\, k_{\mathcal{C},1}(\hat{k}) + 2 \leq k \leq k_{\mathcal{C},2}(\hat{k}),$$

which implies that step 21 of the CONTRACT subroutine is reached for every $k_{\mathcal{C},1}(\hat{k})+2 < k < k_{\mathcal{C},2}(\hat{k})$. By Lemmas A.14 and A.19, we then get

$$\sigma_{max} \geq \frac{\lambda_{k_{\mathcal{C},2}(\hat{k})}}{\|\mathbf{s}_{k_{\mathcal{C},2}(\hat{k})}\|_2} \geq \underline{\sigma}\gamma_\lambda^{k_{\mathcal{C},2}(\hat{k})-k_{\mathcal{C},1}(\hat{k})-2}.$$

Hence,

$$k_{\mathcal{C},2}(\hat{k}) - k_{\mathcal{C},1}(\hat{k}) \leq \frac{1}{\log(\gamma_\lambda)} \log\left(\frac{\sigma_{max}}{\underline{\sigma}}\right) + 2.$$

Now let $k_{\mathcal{C},last}(\hat{k}) - k_{\mathcal{C},1}(\hat{k})$ be the last element of $\mathcal{I}_{\mathcal{C}}(\hat{k})$. Similarly, we can show that

$$k_{\mathcal{A}}(\hat{k}) - k_{\mathcal{C},last}(\hat{k}) \leq \frac{1}{\log(\gamma_\lambda)} \log\left(\frac{\sigma_{max}}{\underline{\sigma}}\right) + 2.$$

The desired result follows s since

$$|\mathcal{C} \cap \mathcal{I}(\hat{k})| = 1 + K_{C,1}\left(\frac{1}{\log(\gamma_\lambda)} \log\left(\frac{\sigma_{max}}{\underline{\sigma}}\right) + 2\right).$$

∎

Notice that since $\underline{\sigma} = \mathcal{O}(\epsilon^{-1})$, the number of contract steps K_C is of order $\mathcal{O}(\log^2 \epsilon^{-1})$. Due to the while loop in step 17 of the CONTRACT subroutine, completing a contract step may require solving more than one subproblem. Our next result provides an upper bound on the number of subproblem routine calls required in one contract step.

LEMMA A.21. *Assume that Assumptions* 4.1*,* 4.2 *and* 4.3 *hold. For a scalar* $\epsilon \in (0,\infty)$*, the total number of subproblems we are required to solve in a step* $k \in \mathcal{C}$ *with* $\mathcal{X}_k \geq \epsilon$ *is at most*

$$K_{\mathcal{C}}^1 \triangleq \log\left((C_{min} + \max\{\lambda_{max}, \lambda_0\})\frac{H_{max} + g_{Lip} + \rho\Delta}{\underline{\sigma}\epsilon}\right).$$

Proof. We prove the result by contradiction. Assume the contrary,

$$\begin{aligned}\lambda &\geq \lambda_k \gamma_\lambda^{\log\left((C_{min}+\max\{\lambda_{max},\lambda_0\})\frac{H_{max}+g_{Lip}+\rho\Delta}{\underline{\sigma}\epsilon}\right)} \\ &\geq \underline{\sigma}\|\mathbf{s}_k\|_2(C_{min} + \max\{\lambda_{max}, \lambda_0\})\frac{H_{max} + g_{Lip} + \rho\Delta}{\underline{\sigma}\epsilon} \\ &\geq H_{max} + g_{Lip} + \rho\Delta,\end{aligned}$$

where the last inequality holds by Lemma A.10 and the fact that $\mathcal{X}_k \geq \epsilon$. Hence, by Lemma A.5

$$\frac{f(\mathbf{x}_k + \mathbf{s}) - f_k}{\|\mathbf{s}\|_2^3} \geq \rho > \frac{f(\mathbf{x}_k + \mathbf{s}_k) - f_k}{\|\mathbf{s}_k\|_2^3},$$

where $\mathbf{s}$ is computed by solving $Q_k(\lambda)$ and the strict inequality holds since $k \in \mathcal{C}$. We conclude that $\|\mathbf{s}_k\|_2 \neq \|\mathbf{s}\|_2$, which contradicts the condition of the while loop. This completes the proof. ∎

Notice that from the definition of $\underline{\sigma}$ in the algorithm, $K_{\mathcal{C}}^1 = \mathcal{O}(\log \epsilon^{-1})$.

THEOREM A.22. *Under Assumptions* 4.1*,* 4.2*, and* 4.3*, for a scalar* $\epsilon \in (0,\infty)$*, the total number of elements in the index set*

$$\{k \in \mathbb{N}_+ : \mathcal{X}_k > \epsilon\}$$

is at most

$$K(\epsilon) \triangleq 1 + (K_\sigma(\epsilon) + K_\Delta)(1 + K_{\mathcal{C}} K_{\mathcal{C}}^1), \tag{A.39}$$

where $K_\sigma(\epsilon)$, K_Δ, $K_{\mathcal{C}}^1$, *and* $K_{\mathcal{C}}$ *are defined in Lemmas* A.16, A.17, A.20, *and* A.21, *respectively. Hence, for* $\epsilon_g > 0$, *the number of subproblem routines required for First-Order-LC-TRACE to find an* ϵ_g*-first-order stationary point is at most*

$$K(\epsilon_g) = \mathcal{O}\left(\epsilon_g^{-3/2} \log^3 \epsilon_g^{-1}\right).$$

Proof. Without loss of generality, we may assume that at least one iteration is performed. Lemmas A.16 and A.17 guarantee that the total number of elements in the index set $\{k \in \mathcal{A} \,|\, k \geq 1,\, \mathcal{X}_k > \epsilon\}$ is at most $K_\sigma(\epsilon) + K_\Delta$. Also, immediately prior to each of the corresponding accepted steps, Lemmas A.18, A.20, and A.21 guarantee that at most $1 + K_C K_C^1$ subproblem routine calls are required in expansion and contraction. Accounting for the first iteration, the desired result follows. ∎

A.3. Proof of Theorem 5.1.

Proof. Let us first define

$$\mathcal{K}_\epsilon \triangleq \{k \in \mathbb{N}_+ : k \geq 1;\, (k-1) \in \mathcal{A}_\sigma;\, \mathcal{X}_k > \epsilon\}$$

and

$$\mathcal{V} \triangleq \{k \,|\, \text{step 7 in Algorithm 5.1 is reached at iteration } k\}.$$

To proceed with our proof, we first show that if $k \in \mathcal{V} \cup \mathcal{K}_\epsilon$, the following reduction bound on the objective value holds:

$$f_{k-1} - f_{k+1} \geq \min\left\{\frac{\rho \mathcal{X}_k^{3/2}}{(H_{Lip} + \sigma_{max})^{3/2}}, \frac{2\psi_k^3}{3\tilde{H}^2}\right\}. \tag{A.40}$$

If $k \in \mathcal{V}$, then using the second-order descent lemma, we obtain

$$\begin{aligned}
f_{k+1} &\leq f_k + \langle \mathbf{g}_k, \mathbf{x}_{k+1} - \mathbf{x}_k\rangle + \frac{1}{2}(\mathbf{x}_{k+1} - \mathbf{x})^T \mathbf{H}_k (\mathbf{x}_{k+1} - \mathbf{x}_k) + \frac{H_{Lip}}{6}\|\mathbf{x}_{k+1} - \mathbf{x}_k\|_2^3 \\
&\leq f_k + \langle \mathbf{g}_k, \mathbf{x}_{k+1} - \mathbf{x}_k\rangle + \frac{1}{2}(\mathbf{x}_{k+1} - \mathbf{x}_k)^T \mathbf{H}_k (\mathbf{x}_{k+1} - \mathbf{x}_k) + \frac{\tilde{H}}{6}\|\mathbf{x}_{k+1} - \mathbf{x}_k\|_2^3 \\
&= f_k + \frac{2\psi_k}{\tilde{H}}\langle \mathbf{g}_k, \widehat{\mathbf{d}}_k\rangle + \frac{2\psi_k^2}{\tilde{H}^2}(\widehat{\mathbf{d}}_k)^T \mathbf{H}_k (\widehat{\mathbf{d}}_k) + \frac{4\psi_k^3}{3\tilde{H}^2}\|\widehat{\mathbf{d}}_k\|_2^3 \\
&\leq f_k - \frac{2\psi_k^3}{\tilde{H}^2} + \frac{4\psi_k^3}{3\tilde{H}^2} \\
&= f_k - \frac{2\psi_k^3}{3\tilde{H}^2}.
\end{aligned} \tag{A.41}$$

Now if a first-order stationary point was reached at $k-1$, i.e., $k-1 \in \mathcal{V}$, it directly follows from (A.41) that

$$f_{k-1} - f_{k+1} \geq f_k - f_{k+1} \geq \frac{2\psi_k^3}{3\tilde{H}^2} \quad \forall\, k \in \mathcal{V},\, k-1 \in \mathcal{V}. \tag{A.42}$$

Otherwise, step 5 of Algorithm 5.1 was reached at iteration $k-1$ and First-Order-LC-TRACE was called. The former algorithm by definition is a monotone algorithm, i.e., it generates a sequence of iterates for which the corresponding sequence of objective values is decreasing. This property combined with (A.41) implies that

$$f_{k-1} - f_{k+1} \geq f_k - f_{k+1} \geq \frac{2\psi_k^3}{3\tilde{H}^2} \quad \forall\, k \in \mathcal{V}, \text{ First-Order-LC-TRACE called at } k-1. \tag{A.43}$$

Combining (A.42) and (A.43), we get

$$f_{k-1} - f_{k+1} \geq f_k - f_{k+1} \geq \frac{2\psi_k^3}{3\tilde{H}^2} \quad \forall\, k \in \mathcal{V}. \tag{A.44}$$

We next show a lower bound on the reduction of the objective value if $k \in \mathcal{K}_\epsilon$. By Lemma A.16, we have

$$f_{k-1} - f_{k+1} \geq f_{k-1} - f_k \geq \frac{\rho \mathcal{X}_k^{3/2}}{(H_{Lip} + \sigma_{max})^{3/2}}, \tag{A.45}$$

where the first inequality again holds by the monotonicity of First-Order-LC-TRACE and (A.41). Combining (A.44) and (A.45), we get

$$f_{k-1} - f_{k+1} \geq \min\left\{\frac{\rho \mathcal{X}_k^{3/2}}{(H_{Lip} + \sigma_{max})^{3/2}}, \frac{2\psi_k^3}{3\tilde{H}^2}\right\} \quad \forall k \in \mathcal{K}_\epsilon \cup \mathcal{V}.$$

By summing over the iterations we get

$$2(f_0 - f_{min}) \geq \sum_{k \in \mathcal{K}_\epsilon \cup \mathcal{V}, k \geq 1} (f_{k-1} - f_{k+1}) \geq |\mathcal{K}_\epsilon \cup \mathcal{V}| \min\left\{\frac{\rho \mathcal{X}_k^{3/2}}{(H_{Lip} + \sigma_{max})^{3/2}}, \frac{2\psi_k^3}{3\tilde{H}^2}\right\}.$$

Rearranging this inequality yields

$$|\mathcal{K}_\epsilon \cup \mathcal{V}| \leq \frac{2(f_0 - f_{min})}{\max\left\{\frac{\rho \mathcal{X}_k^{3/2}}{(H_{Lip} + \sigma_{max})^{3/2}}, \frac{2\psi_k^3}{3\tilde{H}^2}\right\}}. \tag{A.46}$$

Let $\mathcal{H}(\epsilon_g, \epsilon_H) \triangleq \{k \mid \mathcal{X}_k > \epsilon_g,\ \psi_k > \epsilon_H\}$. Using Lemmas A.18, A.20, and A.21, the number of subproblem routine calls required in expansion and contraction between two acceptance steps is upper bounded by $1 + K_{\mathcal{C}} K_{\mathcal{C}}^1$. Hence,

$$|\mathcal{H}(\epsilon_g, \epsilon_H)| \leq (|\mathcal{K}_\epsilon \cup \mathcal{V}| + K_\Delta)(1 + K_{\mathcal{C}} K_{\mathcal{C}}^1),$$

which concludes our proof. ∎

REFERENCES

[1] A. Anandkumar and R. Ge, *Efficient approaches for escaping higher order saddle points in non-convex optimization*, in Proceedings of the Conference on Learning Theory, 2016, pp. 81–102.

[2] A. S. Bandeira, N. Boumal, and V. Voroninski, *On the low-rank approach for semidefinite programs arising in synchronization and community detection*, in Proceedings of the Conference on Learning Theory, 2016, pp. 361–382.

[3] B. Barazandeh and M. Razaviyayn, *On the Behavior of the Expectation-Maximization Algorithm for Mixture Models*, preprint, https://arxiv.org/abs/1809.08705, 2018.

[4] A. Beck and D. Pan, *A branch and bound algorithm for nonconvex quadratic optimization with ball and linear constraints*, J. Global Optim., 69 (2017), pp. 309–342.
[5] D. P. Bertsekas, *Nonlinear Programming*, Athena Scientific, Belmont, MA, 1999.
[6] W. Bian, X. Chen, and Y. Ye, *Complexity analysis of interior point algorithms for non-Lipschitz and nonconvex minimization*, Math. Program., 149 (2015), pp. 301–327.
[7] E. Birgin and J. Martínez, *On regularization and active-set methods with complexity for constrained optimization*, SIAM J. Optim., 28 (2018), pp. 1367–1395.
[8] N. Boumal, *Nonconvex phase synchronization*, SIAM J. Optim., 26 (2016), pp. 2355–2377.
[9] C. Cartis, N. Gould, and P. L. Toint, *An adaptive cubic regularization algorithm for nonconvex optimization with convex constraints and its function-evaluation complexity*, IMA J. Numer. Anal., 32 (2012), pp. 1662–1695.
[10] C. Cartis, N. I. Gould, and P. L. Toint, *Adaptive cubic regularisation methods for unconstrained optimization. Part* I: *Motivation, convergence and numerical results*, Math. Program., 127 (2011), pp. 245–295.
[11] C. Cartis, N. I. Gould, and P. L. Toint, *Adaptive cubic regularisation methods for unconstrained optimization. Part* II: *Worst-case function-and derivative-evaluation complexity*, Math. Program., 130 (2011), pp. 295–319.
[12] C. Cartis, N. I. Gould, and P. L. Toint, *On the evaluation complexity of cubic regularization methods for potentially rank-deficient nonlinear least-squares problems and its relevance to constrained nonlinear optimization*, SIAM J. Optim., 23 (2013), pp. 1553–1574.
[13] C. Cartis, N. I. Gould, and P. L. Toint, *On the complexity of finding first-order critical points in constrained nonlinear optimization*, Math. Program., 144 (2014), pp. 93–106.
[14] C. Cartis, N. I. Gould, and P. L. Toint, *On the evaluation complexity of constrained nonlinear least-squares and general constrained nonlinear optimization using second-order methods*, SIAM J. Numer. Anal., 53 (2015), pp. 836–851.
[15] C. Cartis, N. I. Gould, and P. L. Toint, *Second-order optimality and beyond: Characterization and evaluation complexity in convexly constrained nonlinear optimization*, Found. Comput. Math., 18 (2018), pp. 1073–1107.
[16] A. R. Conn, N. I. Gould, and P. L. Toint, *Trust Region Methods*, Vol. 1, SIAM, Philadelphia, 2000.
[17] F. E. Curtis and D. P. Robinson, *Exploiting Negative Curvature in Deterministic and Stochastic Optimization*, preprint, https://arxiv.org/abs/1703.00412, 2017.
[18] F. E. Curtis, D. P. Robinson, and M. Samadi, *An Inexact Regularized Newton Framework with a Worst-case Iteration Complexity of* $\mathcal{O}(\epsilon^{-3/2})$ *for Nonconvex Optimization*, preprint, https://arxiv.org/abs/1708.00475, 2017.
[19] F. E. Curtis, D. P. Robinson, and M. Samadi, *A trust region algorithm with a worst-case iteration complexity of* $\mathcal{O}(\epsilon^{-3/2})$ *for nonconvex optimization*, Math. Program., 162 (2017), pp. 1–32.
[20] S. S. Du, C. Jin, J. D. Lee, M. I. Jordan, A. Singh, and B. Poczos, *Gradient descent can take exponential time to escape saddle points*, in Proceedings of Advances in Neural Information Processing Systems, 2017, pp. 1067–1077.
[21] R. Ge, F. Huang, C. Jin, and Y. Yuan, *Escaping from saddle points – online stochastic gradient for tensor decomposition*, in Proceedings of the Conference on Learning Theory, 2015, pp. 797–842.
[22] R. Ge, J. D. Lee, and T. Ma, *Matrix completion has no spurious local minimum*, in Proceedings of Advances in Neural Information Processing Systems, 2016, pp. 2973–2981.
[23] M. Hong, J. D. Lee, and M. Razaviyayn, *Gradient Primal-Dual Algorithm Converges to Second-Order Stationary Solutions for Nonconvex Distributed Optimization*, preprint, https://arxiv.org/abs/1802.08941, 2018.
[24] Y. Hsia and R.-L. Sheu, *Trust Region Subproblem with a Fixed Number of Additional Linear Inequality Constraints has Polynomial Complexity*, preprint, https://arxiv.org/abs/1312.1398, 2013.
[25] V. Jeyakumar and G. Li, *Trust-region problems with linear inequality constraints: Exact SDP relaxation, global optimality and robust optimization*, Math. Program., 147 (2014), pp. 171–206.
[26] C. Jin, R. Ge, P. Netrapalli, S. M. Kakade, and M. I. Jordan, *How to Escape Saddle Points Efficiently*, preprint, https://arxiv.org/abs/1703.00887, 2017.
[27] J. D. Lee, I. Panageas, G. Piliouras, M. Simchowitz, M. I. Jordan, and B. Recht, *First-Order Methods Almost Always Avoid Saddle Points*, preprint, https://arxiv.org/abs/1710.07406, 2017.
[28] J. D. Lee, M. Simchowitz, M. I. Jordan, and B. Recht, *Gradient Descent Converges to Minimizers*, preprint, https://arxiv.org/pdf/1602.04915.pdf, 2016.

[29] S. Lu, M. Razaviyayn, B. Yang, K. Huang, and M. Hong, *Finding Second-Order Stationary Solutions Using a Perturbed Projected Gradient Descent Algorithm for Non-Convex Linearly Constrained Problems*, preprint, https://arxiv.org/pdf/1907.04450.pdf, 2019.

[30] S. Lu, Z. Wei, and L. Li, *A trust region algorithm with adaptive cubic regularization methods for nonsmooth convex minimization*, Comput. Optim. Appl., 51 (2012), pp. 551–573.

[31] A. Mokhtari, A. Ozdaglar, and A. Jadbabaie, *Escaping Saddle Points in Constrained Optimization*, preprint, https://arxiv.org/abs/1809.02162, 2018.

[32] K. G. Murty and S. N. Kabadi, *Some NP-complete problems in quadratic and nonlinear programming*, Math. Program., 39 (1987), pp. 117–129.

[33] Y. Nesterov and B. T. Polyak, *Cubic regularization of newton method and its global performance*, Math. Program., 108 (2006), pp. 177–205.

[34] M. Nouiehed, J. D. Lee, and M. Razaviyayn, *Convergence to Second-order Stationarity for Constrained Non-Convex Optimization*, preprint, https://arxiv.org/abs/1810.02024, 2020.

[35] M. Nouiehed and M. Razaviyayn, *Learning Deep Models: Critical Points and Local Openness*, preprint, https://arxiv.org/abs/1803.02968, 2018.

[36] M. Nouiehed, M. Sanjabi, J. D. Lee, and M. Razaviyayn, *Solving a Class of Non-Convex Min-Max Games Using Iterative First Order Methods*, preprint, https://arxiv.org/abs/1902.08297, 2019.

[37] M. O'Neill and S. J. Wright, *A Log-Barrier Newton-CG Method for Bound Bonstrained Optimization with Complexity Guarantees*, preprint, https://arxiv.org/abs/1904.03563, 2019.

[38] J. Sun, Q. Qu, and J. Wright, *A geometric analysis of phase retrieval*, in Proceedings of the IEEE International Symposium on Information Theory (ISIT), 2016, IEEE, 2016, pp. 2379–2383.

[39] J. Sun, Q. Qu, and J. Wright, *Complete dictionary recovery over the sphere* I*: Overview and the geometric picture*, IEEE Trans. Inform. Theory, 63 (2017), pp. 853–884.

[40] J. Sun, Q. Qu, and J. Wright, *Complete dictionary recovery over the sphere* II*: Recovery by riemannian trust-region method*, IEEE Trans. Inform. Theory, 63 (2017), pp. 885–914.