A Trust Region Method for Finding Second-Order Stationarity in Linearly Constrained Non-Convex Optimization

Maher Nouiehed   Meisam Razaviyayn*

April 16, 2019

Abstract

Motivated by TRACE algorithm [18], we propose a trust region algorithm for finding second order stationary points of a linearly constrained non-convex optimization problem. We show the convergence of the proposed algorithm to (ϵ_g, ϵ_H)-second order stationary points in $\widetilde{\mathcal{O}}\left(\max\left\{\epsilon_g^{-3/2}, \epsilon_H^{-3}\right\}\right)$ iterations. This iteration complexity is achieved for general linearly constrained optimization without cubic regularization of the objective function.

1 Introduction

Due to its wide application in machine learning, solving non-convex optimization problems encountered significant attention in recent years [1, 11, 13, 14, 17, 18, 28]. 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 non-convex problems have been proposed in recent years [26, 22, 31, 8, 9, 10, 34].

For general non-convex optimization problems, it is well-known that computing a local optimum is NP-Hard [30]. 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, 7, 33, 36, 20, 37, 38] for examples of such objective functions.

Convergence to second-order stationarity in smooth unconstrained setting has been thoroughly investigated in the optimization literature [21, 16, 9, 10, 31, 17, 18, 20]. As a second-order algorithm, [31] proposed a cubic regularization method that converges to approximate second-order stationarity in finite number of steps. More recently, [9, 10] 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, [18] 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 $\mathcal{O}(\max\{\epsilon_g^{-3/2}, \epsilon_H^{-3}\})$. This method alters the acceptance criteria adopted by traditional trust region methods, and implements a new mechanism for updating the trust region radius. A more recent second-order algorithm that uses a dynamic

---

*Department of Industrial and Systems Engineering, University of Southern California.

choice of direction and step-size was proposed in [16]. 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 [17].

Recent results show that for smooth unconstrained optimization problems, even first order methods can converge to second-order stationarity, almost surely. For instance, [20] shows 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. A similar result was shown for the vanilla gradient descent algorithm in [26]. A negative result provided by [19] shows that vanilla gradient descent can take exponential number of steps to converge to second-order stationarity. This computational inefficiency can be overcome by a smart perturbed form of gradient descent proposed in [24].

Most of the above results can be extended to the smooth constrained optimization in the presence of simple manifold constraints. In this case, [25] shows that manifold gradient descent converges to second-order stationarity, almost surely. More recently, [22] established similar results for gradient primal-dual algorithms applied on linearly constrained optimization problems. When the constraints are non-manifold type, projected gradient descent is a natural replacement of gradient descent. As a negative result, [32] constructs 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 exist 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 [8, 13, 12]. The work in [8] adapts the ARC algorithm and showed convergence to ϵ_g-first-order stationarity in at most $\mathcal{O}(\epsilon_g^{-3/2})$ iterations. [6] uses active set method and cubic regularization to achieve this rate for special types of constraints. The work[5] uses 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, [14] 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, [29] 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 these methods hide the per-iteration complexity of solving the quadratic or cubic sub-problems. As shown in [32], for linearly constrained non-convex problems, even checking whether a given point is an *approximate* second-order stationary point is NP-Hard. Despite this hardness result, [32] proposed a second-order Frank-Wolfe algorithm that adapts the dynamic method introduced in [16], and identified instances for which solving the constrained quadratic sub-problem can be done efficiently. The algorithm converges to approximate first and second-order stationarity with a worst-case complexity similar to [16]. 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 non-convex problems. We establish the convergence of our algorithm to (ϵ_g, ϵ_H)-second order stationarity in at most $\tilde{\mathcal{O}}(\epsilon_g^{-3/2}, \epsilon_H^{-3})$ iterations.

The remainder of this paper is organized as follow. 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 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

$$\min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x}), \tag{1}$$

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 (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 (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

$$\|\nabla f(\bar{\mathbf{x}})\|_2 \leq \epsilon_g. \tag{2}$$

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

$$\|\nabla f(\bar{\mathbf{x}})\|_2 \leq \epsilon_g \text{ and } \nabla^2 f(\bar{\mathbf{x}}) \succeq -\epsilon_H \mathbf{I}. \tag{3}$$

We now extend these definitions to the constrained optimization problem

$$\min_{\mathbf{x} \in \mathcal{P}} f(\mathbf{x}), \tag{4}$$

where $\mathcal{P} \subseteq \mathbb{R}^n$ is a closed convex set. As defined in [4], we say $\bar{\mathbf{x}} \in \mathcal{P}$ is a FOSP of (4) if

$$\langle \nabla f(\bar{\mathbf{x}}), \mathbf{x} - \bar{\mathbf{x}} \rangle \geq 0 \quad \forall \mathbf{x} \in \mathcal{P}. \tag{5}$$

Similarly, we say a point $\bar{\mathbf{x}}$ is a SOSP of the optimization problem (4) if $\bar{\mathbf{x}} \in \mathcal{P}$ is a first order stationary point and

$$0 \leq \mathbf{d}^T \nabla^2 f(\bar{\mathbf{x}}) \mathbf{d}, \quad \forall \mathbf{d} \text{ s.t. } \langle \mathbf{d}, \nabla f(\bar{\mathbf{x}}) \rangle = 0 \text{ and } \bar{\mathbf{x}} + \mathbf{d} \in \mathcal{P}. \tag{6}$$

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

Motivated by (5) and (6), given a feasible point $\mathbf{x}$, we define the following first and second order stationarity measures

$$\begin{aligned} \mathcal{X}(\mathbf{x}) &\triangleq - \min_{\mathbf{s}} \quad \langle \nabla f(\mathbf{x}), \mathbf{s} \rangle \\ &\text{s.t.} \quad \mathbf{x} + \mathbf{s} \in \mathcal{P}, \|\mathbf{s}\| \leq 1. \end{aligned} \tag{7}$$

and

$$\begin{aligned} \psi(\mathbf{x}) &\triangleq - \min_{\mathbf{d}} \quad \mathbf{d}^T \nabla^2 f(\mathbf{x}) \mathbf{d} \\ &\text{s.t.} \quad \mathbf{x} + \mathbf{d} \in \mathcal{P}, \|\mathbf{d}\| \leq 1 \\ &\qquad \langle \nabla f(\mathbf{x}), \mathbf{d} \rangle \leq 0. \end{aligned} \tag{8}$$

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 [32], can be linked to the standard definitions in [4] by the following Lemma.

Lemma 1 ([32]). *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*

- $\mathcal{X}(\bar{\mathbf{x}}) = 0$ *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.

Definition 2. *Approximate Stationary Point: For problem (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 (3) and (2).

Remark 3. *Notice that our definition of (ϵ_g, ϵ_H)-second order stationarity is different than the definition in [29]. In particular, there are two major differences:*

1) *The definition used for approximate first and second order stationarity in [29] does not include the normalization constraints $\|\mathbf{s}\| \leq 1$ and $\|\mathbf{d}\| \leq 1$ in (7) and (8).*

2) *The second order optimality measure in [29] is defined based on using equality constraint $\langle \nabla f(\mathbf{x}), \mathbf{d} \rangle = 0$ in (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 (7) does not reflect this approximate optimality if we do not include the normalization constraint in (7). To understand the importance of using inequality constraint $\langle \nabla f(\mathbf{x}), \mathbf{d} \rangle \leq 0$ instead of equality constraint in (8), consider the scalar optimization problem

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

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 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 (8), then the value of $\psi(\cdot)$ would have been zero.

Remark 4. *There are other definitions of second order stationarity in the literature. For example, the works [5, 6] use a scaled version of the Hessian in different directions to define second order stationarity for box constraints. Recently, [35] 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 [27] for general linearly constrained optimization problems.*

3 Finding second-order stationary points for constrained optimization

Consider the quadratic co-positivity problem

$$\min_{\mathbf{x} \in \mathbb{R}^n} \quad \frac{1}{2}\mathbf{x}^T\mathbf{Q}\mathbf{x} \qquad \text{s.t.} \quad \mathbf{x} \geq \mathbf{0}, \; \|\mathbf{x}\| \leq 1. \tag{9}$$

Clearly, checking whether $\bar{\mathbf{x}} = \mathbf{0}$ is a second order stationary point of (9) is equivalent to checking its local optimality, which is an NP-Hard problem [30]. This observation shows that checking exact second order stationarity is Hard. The following result, which is borrowed from [32], shows that even checking *approximate* second order stationarity is NP-hard.

Theorem 5 (Theorem 6 in [32]). *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 non-convex problems. However, in this problem, *the source of hardness stems from the number of linear inequality constraints*. In fact, when we have a small constant number of linear constraints (say m fixed constraints), [23] proposed a backtracking approach that efficiently solves this quadratic constrained optimization problem. Although the method proposed is exponential in m as it uses an exhaustive search over the set of active constraints, it can still be used when m 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 non-convex 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 Non-Convex Optimization Problems

Consider the optimization problem

$$\min_{\mathbf{x} \in \mathbb{R}^n} \; f(\mathbf{x}),$$
$$\text{s.t.} \quad \mathbf{A}\mathbf{x} \leq \mathbf{b}, \tag{10}$$

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 [18] to the above linearly-constrained non-convex 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 $\widetilde{\mathcal{O}}\big(\max\{\epsilon_g^{-3/2}, \epsilon_H^{-3}\}\big)$.

LC-TRACE is different from the traditional trust region method proposed in [14] for constrained optimization. More specifically, LC-TRACE utilizes the mechanisms used in TRACE [18] to provide a faster convergence rate compared to [14]. The improved convergence rate matches the rates achieved by adapted ARC [8] and TRACE [18], 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 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 sub-problem

$$\min_{\mathbf{s}\in\mathbb{R}^n} q_k(\mathbf{s}), \quad \text{s.t. } \|\mathbf{s}\|_2 \le \delta_k, \tag{11}$$

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. 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 [9, 10]. In particular, the authors in [18] observed that ARC computes a positive sequence of cubic regularization coefficients $\sigma_k \in [\underline{\sigma}, \overline{\sigma}]$ that satisfy

$$f_k - f_{k+1} \ge c_1\sigma_k\|\mathbf{s}_k\|_2^3 \quad \text{and} \quad \|\mathbf{s}_k\|_2 \ge \left(\frac{c_2}{\overline{\sigma}+c_3}\right)^{1/2}\|\mathbf{g}_{k+1}\|_2^{1/2}, \tag{12}$$

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 (12). Some of these ideas are discussed next.

Sufficient Decrease Acceptance Criteria. TRACE defines the ratio

$$\rho_k \triangleq \frac{f_k - f(\mathbf{x}_k + \mathbf{s}_k)}{\|\mathbf{s}_k\|_2^3}, \tag{13}$$

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 $\rho_k \ge \rho$. By noticing that a small $\|\mathbf{s}_k\|_2$ may satisfy only the first condition in (12), the developers of TRACE realize that an acceptance criteria that only involves (13) 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 $\|\mathbf{s}\|_2 \le \delta_k$ in sub-problem (11). In short, TRACE accepts a trial pair $(\mathbf{s}_k, \lambda_k)$ if it satisfies the following conditions:

$$\rho_k \ge \rho \quad \text{and} \quad \lambda_k/\|\mathbf{s}_k\|_2 \le \sigma_k. \tag{14}$$

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 sub-linear updates. In particular, this subroutine compares the radius obtained by the linear update scheme to the norm of the trial step computed using

$$\min_{\mathbf{s}\in\mathbb{R}^n} f_k + \mathbf{g}_k^T \mathbf{s} + \frac{1}{2}\mathbf{s}^T(\mathbf{H}_k + \lambda\mathbf{I})\mathbf{s}, \tag{15}$$

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.

6

4.2 Difference Between LC-TRACE and TRACE

In the constrained setting, we define the trust region sub-problem and its regularized Lagrangian form as

$$Q_k \triangleq \min_{\mathbf{s} \in \mathbb{R}^n} q_k(\mathbf{s}), \quad \text{s.t.} \quad \begin{cases} \mathbf{A}\mathbf{s} \le \mathbf{b} - \mathbf{A}\mathbf{x}_k \\ \|\mathbf{s}\|_2 \le \delta_k \end{cases} \tag{16}$$

and

$$Q_k(\lambda) \triangleq \min_{\mathbf{s} \in \mathbb{R}^n} f_k + \mathbf{g}_k^T \mathbf{s} + \frac{1}{2}\mathbf{s}^T(\mathbf{H}_k + \lambda \mathbf{I})\mathbf{s}, \quad \text{s.t.} \ \mathbf{A}\mathbf{s} \le \mathbf{b} - \mathbf{A}\mathbf{x}_k. \tag{17}$$

A major difficulty introduced by the constraints is related to the optimality conditions of the sub-problem. In the unconstrained case, it is known that $\mathbf{H}_k + \lambda_k \mathbf{I} \succeq 0$ at every iteration [15, 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 sub-problem (17) 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

$$Q(\lambda) = \min_{s_1 \le 5, \, s_2 \ge 0} s_1^2 - s_2^2 + \lambda(s_1^2 + s_2^2) \quad \text{s.t.} \ s_2 - 3s_1 \le -12. \tag{18}$$

It is not hard to see that the optimal solution of (18) 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 increases 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

$$\underline{\sigma} = \frac{\epsilon}{C_{min} + \max\{\lambda_{max}, \lambda_0\}} \quad \text{and} \quad \bar{\sigma} = 2\Delta, \tag{19}$$

and altering the update rule of λ in the CONTRACT sub-routine. Here $\epsilon > 0$ is the threshold used for the termination of the algorithm, C_{min} is defined in Lemma 20, λ_{max} is defined in Lemma 19, and Δ is defined in Lemma 18.

Another major difficulty in the constrained scenario is related to the standard trust region theory on the relationship between sub-problem 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 [15, Chapter 7]. This relationship was used in [18], to show that CONTRACT subroutine reduces the radius of the trust region. However, it can be seen from example (18), that this relation may not hold in the constrained case. To account for this issue, we modified the CONTRACT sub-routine to guarantee a reduction in the trust region radius (See Lemma 12). In summary, the differences between LC-TRACE and TRACE are mainly in the CONTRACT sub-routine. 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 outlined in Algorithm 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 (16) and using the equation (13). 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:

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

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 non-decreasing. 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.

When CONTRACT subroutine is called, two different cases may occur in Algorithm 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

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

where $\underline{\sigma}$ and $\bar{\sigma}$ are prescribed positive constants defined in (19). The second case is reached whenever the conditions in Step 3 tests false. In that case, we choose $\lambda \in (\lambda_k, C\lambda_k]$ with $C > 1$ is 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 6. In what follows, we first present our results about the convergence of First-Order-LC-TRACE algorithm and its iteration complexity.

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 6. *The objective function f is twice continuously differentiable and bounded below by a scalar f_{min} on $\mathcal{P}$. 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 L and H_{Lip}, respectively. Furthermore, we assume the gradient sequence $\{\mathbf{g}_k\}$ is bounded in norm, that is, there exists a scalar constant $g_{max} > 0$ such that $\|\mathbf{g}_k\|_2 \triangleq \|\nabla f(\mathbf{x}_k)\|_2 \leq g_{max}$ for all $k \in \mathbb{N}$. Moreover, we assume that the Hessian sequence $\{\mathbf{H}_k\}$ is bounded in norm, that is, there exist a scalar constant $H_{max} > 0$, such that $\|H_k\|_2 \triangleq \|\nabla^2 f(\mathbf{x}_k)\|_2 \leq H_{max}$ for all $k \in \mathbb{N}$.*

Algorithm 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 (19).

1: **procedure** FIRST-ORDER-LC-TRACE
2:   Choose a feasible point $\mathbf{x}_0$, a pair $\{\delta_0, \Delta_0\}$ with $0 < \delta_0 \leq \Delta_0$, and σ_0 with $\sigma_0 \geq \underline{\sigma}$.
3:   Compute $(\mathbf{s}_0, \lambda_0)$ by solving Q_0, then compute ρ_0 using the definition in (13) .
4:   **for** $k = 0, 1, 2, \ldots$ **do**
5:     **if** $\rho_k \geq \rho$ and either $\lambda_k / \|\mathbf{s}_k\|_2 \leq \sigma_k$ or $\|\mathbf{s}_k\|_2 = \Delta_k$ **then**   (Acceptance)
6:       set $\mathbf{x}_{k+1} \leftarrow \mathbf{x}_k + \mathbf{s}_k$
7:       set $\Delta_{k+1} \leftarrow \max\{\Delta_k, \gamma_E\|\mathbf{s}_k\|_2\}$
8:       set $\delta_{k+1} \leftarrow \min\{\Delta_{k+1}, \max\{\delta_k, \gamma_E\|\mathbf{s}_k\|_2\}\}$
9:       set $\sigma_{k+1} \leftarrow \max\{\sigma_k, \lambda_k / \|\mathbf{s}_k\|_2\}$
10:    **else if** $\rho_k < \rho$ **then**   (Contraction)
11:      set $\mathbf{x}_{k+1} \leftarrow \mathbf{x}_k$
12:      set $\Delta_{k+1} \leftarrow \Delta_k$
13:      set $\delta_{k+1} \leftarrow \text{CONTRACT}(\mathbf{x}_k, \delta_k, \sigma_k, \mathbf{s}_k, \lambda_k)$ defined in Algorithm (2)
14:    **else if** $\rho_k \geq \rho$, $\lambda_k / \|\mathbf{s}_k\|_2 > \sigma_k$, and $\|\mathbf{s}_k\|_2 < \Delta_k$ **then**   (Expansion)
15:      set $\mathbf{x}_{k+1} \leftarrow \mathbf{x}_k$
16:      set $\Delta_{k+1} \leftarrow \Delta_k$
17:      set $\delta_{k+1} \leftarrow \min\{\Delta_{k+1}, \lambda_k / \sigma_k\}$
18:      set $\sigma_{k+1} \leftarrow \sigma_k$
19:    **end if**
20:    Compute $(\mathbf{s}_{k+1}, \lambda_{k+1})$ by solving Q_{k+1}, then compute ρ_{k+1} using (13)
21:    **if** $\rho_k < \rho$ **then**
22:      set $\sigma_{k+1} \leftarrow \max\{\sigma_k, \lambda_{k+1} / \|\mathbf{s}_{k+1}\|_2\}$
23:    **end if**
24: **end for**
25: **end procedure**

Algorithm 2 CONTRACT Sub-routine

Require: update constant $\gamma_C \in (0,1)$.
Require: ratio bound constants $\underline{\sigma}$ and $\overline{\sigma}$ defined in (19).

1: **procedure** CONTRACT($\mathbf{x}_k, \delta_k, \sigma_k, \mathbf{s}_k, \lambda_k$)
2:      set $\bar{\lambda} \leftarrow \lambda_k + \underline{\sigma}\Delta_k$ and set $\bar{\mathbf{s}}$ as the solution of $Q_k(\bar{\lambda})$.
3:      **if** $\|\bar{\mathbf{s}}\|_2 < \|\mathbf{s}_k\|_2$ and $\lambda_k < \underline{\sigma}\|\mathbf{s}_k\|_2$ **then**
4:          set $\lambda \leftarrow \bar{\lambda} + H_{max} + \left(\underline{\sigma}\mathcal{X}_k\right)^{1/2}$ and set $\mathbf{s}$ as the solution of $Q_k(\lambda)$.
5:          **if** $\lambda/\|\mathbf{s}\|_2 \leq \overline{\sigma}$ **then**
6:              **return** $\delta_{k+1} \leftarrow \|\mathbf{s}\|_2$
7:          **else**
8:              set $\lambda \leftarrow \bar{\lambda}$
9:              **return** $\delta_{k+1} \leftarrow \|\bar{\mathbf{s}}\|_2$
10:          **end if**
11:      **else**
12:          **if** $\|\bar{\mathbf{s}}\|_2 = \|\mathbf{s}_k\|_2$ **then**
13:              set $\lambda \leftarrow \gamma_\lambda \bar{\lambda}$ and set $\mathbf{s}$ as the solution of $Q_k(\lambda)$
14:          **else**
15:              set $\lambda \leftarrow \gamma_\lambda \lambda$ and set $\mathbf{s}$ as the solution of $Q_k(\lambda)$
16:          **end if**
17:          **while** $\|\mathbf{s}\|_2 = \|\mathbf{s}_k\|_2$ **do**
18:              $\lambda \leftarrow \gamma_\lambda \lambda$ and set $\mathbf{s}$ as the solution of $Q_k(\lambda)$
19:          **end while**
20:          **if** $\|\mathbf{s}\|_2 \geq \gamma_C \|\mathbf{s}_k\|_2$ **then**
21:              **return** $\delta_{k+1} \leftarrow \|\mathbf{s}\|_2$
22:          **else**
23:              **return** $\delta_{k+1} \leftarrow \gamma_C \|\mathbf{s}_k\|_2$
24:          **end if**
25:      **end if**
26: **end procedure**

10

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

Theorem 7. *Under Assumption 6, any limit point of the iterates generated by First-Order-LC-TRACE algorithm is a first-order stationary point .*

Proof. The proof of the Theorem is relegated to Appendix A.1. $\qquad\square$

Unfortunately, Assumption 6 is not sufficient to obtain the desired rate of convergence in the presence of constraints; in particular, Assumption 6 may not ensure a model decrease of the form

$$f_k - q_k(\mathbf{s}_k) = -\mathbf{g}_k^T\mathbf{s}_k - \frac{1}{2}\mathbf{s}_k^T\mathbf{H}_k\mathbf{s}_k \geq \kappa\lambda_k\|\mathbf{s}_k\|_2^2, \tag{20}$$

for some constant $\kappa \in (0,1)$. To understand this, let us first review the same result for the unconstrained scenario: it is known that $\mathbf{H}_k + \lambda_k\mathbf{I} \succeq 0$ at every iteration [15, Corollary 7.2.2]. Thus, by Lemma 11, we get

$$f_k - q_k(\mathbf{s}_k) = -\mathbf{g}_k^T\mathbf{s}_k - \frac{1}{2}\mathbf{s}_k^T\mathbf{H}_k\mathbf{s}_k \geq \frac{1}{2}\lambda_k\|\mathbf{s}_k\|_2^2. \tag{21}$$

However, in contrast to the unconstrained case, there is no guarantee that the step $\mathbf{s}_k$ satisfies (20) in the constrained scenario. More specifically, in the presence of constraints, the condition is not guaranteed when the step $\mathbf{s}_k$ provides ascent first-order direction with negative curvature. To account for this case, we assume the following assumption holds.

Assumption 8. *If $\mathbf{g}_k^T\mathbf{s}_k \geq 0$ and $\mathbf{s}_k^T\mathbf{H}_k\mathbf{s}_k \leq 0$, there exists a sequence of feasible points $\{\mathbf{x}_{k,i}\}_{i=0}^{l_k}$ with $0 \leq l_k \leq \bar{l}$, $\mathbf{x}_{k,0} = \mathbf{x}_k$, $\mathbf{s}_{k,i} = \mathbf{x}_{k,i} - \mathbf{x}_k$ and $\mathbf{x}_{k,l_k} = \mathbf{x}_k + \mathbf{s}_k$ such that for $i = 1,\dots,l_k$,*

$$q_k(\mathbf{s}_{k,i}) \leq q_k(\mathbf{s}_{k,i-1});$$
$$\mathbf{g}_k^T(\mathbf{x}_{k,i} - \mathbf{x}_{k,i-1}) + \mathbf{s}_{k,i}^T\mathbf{H}_k(\mathbf{x}_{k,i} - \mathbf{x}_{k,i-1}) \leq -\lambda_k\mathbf{s}_{k,i}^T(\mathbf{x}_{k,i} - \mathbf{x}_{k,i-1});$$
$$\mathbf{g}_k^T(\mathbf{x}_{k,i} - \mathbf{x}_{k,i-1}) + \mathbf{s}_{k,i-1}^T\mathbf{H}_k(\mathbf{x}_{k,i} - \mathbf{x}_{k,i-1}) \leq -\lambda_k\mathbf{s}_{k,i-1}^T(\mathbf{x}_{k,i} - \mathbf{x}_{k,i-1}).$$

This assumption was also used in [8] to show that the number of iterations required to reach an ϵ-first order stationary point when adaptive ARC algorithm is used is $\mathcal{O}(\epsilon^{-3/2})$. As mentioned in [8], this assumptions holds if $\mathbf{x}_{k,l_k}$ is the first minimizer of the model q_{λ_k} along the piecewise linear path $\mathcal{P}_k \triangleq \bigcup_{i=1}^{l_k}[\mathbf{x}_{k,i-1},\mathbf{x}_i]$.
Using Assumption 8, we obtain the desired model decrease (20) and we have the following Theorem.

Theorem 9. *Under Assumptions 6 and 8, for any given scalar $\epsilon \in (0,\infty)$, the total number of sub-problem routines of First-Order-LC-TRACE required to reach an ϵ-first order stationary point of* (10) *is $\mathcal{O}(\epsilon^{-3/2}\log^3(1/\epsilon))$.*

Proof. The proof of the Theorem is relegated to Appendix A.2. $\qquad\square$

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 3 for converging to second order stationary points.

Algorithm 3 Second-Order-LC-TRACE

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

1: **procedure**
2:     Choose a feasible point $\mathbf{x}_0$.
3:     Compute $\mathcal{X}_0$ and ψ_0 by solving (7) and (8), respectively.
4:     **for** $k = 0, 1, 2, \ldots$ **do**
5:         **if** $\mathcal{X}_k > \epsilon_g$ **then**
6:             Compute $\mathbf{x}_{k+1}$ by running one iteration of First-Order-LC-TRACE starting with $\mathbf{x}_k$.
7:         **else**
8:             Compute $\widehat{\mathbf{d}}_k$ and ψ_k by solving (8).
9:             set $\mathbf{x}_{k+1} \leftarrow \mathbf{x}_k + \dfrac{2\psi_k}{\tilde{H}} \widehat{\mathbf{d}}_k.$
10:        **end if**
11:    **end for**
12: **end procedure**

We now show that this algorithm can find an (ϵ_g, ϵ_H)-second-order stationary point of problem (10).

Theorem 10. *Under Assumptions 6 and 8, 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 (10) when running Algorithm 3 is $\mathcal{O}\left(\log^3(\epsilon_g^{-1}) \max\left\{ \epsilon_g^{-3/2}, \epsilon_H^{-3} \right\} \right)$.*

Proof. The proof of the Theorem is relegated to Appendix A.3. $\square$

References

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

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

[3] B. Barazandeh and M. Razaviyayn. On the behavior of the expectation-maximization algorithm for mixture models. *arXiv preprint arXiv:1809.08705*, 2018.

[4] D. P. Bertsekas. *Nonlinear programming*. Athena scientific Belmont, 1999.

[5] W. Bian, X. Chen, and Y. Ye. Complexity analysis of interior point algorithms for non-lipschitz and nonconvex minimization. *Mathematical Programming*, 149(1-2):301–327, 2015.

[6] E. Birgin and J. Martínez. On regularization and active-set methods with complexity for constrained optimization. *SIAM Journal on Optimization*, 28(2):1367–1395, 2018.

[7] N. Boumal. Nonconvex phase synchronization. *SIAM Journal on Optimization*, 26(4):2355–2377, 2016.

[8] 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 Journal of Numerical Analysis*, 32(4):1662–1695, 2012.

[9] C. Cartis, N. I. Gould, and P. L. Toint. Adaptive cubic regularisation methods for unconstrained optimization. part i: motivation, convergence and numerical results. *Mathematical Programming*, 127(2):245–295, 2011.

[10] 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. *Mathematical programming*, 130(2):295–319, 2011.

[11] 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 Journal on Optimization*, 23(3):1553–1574, 2013.

[12] C. Cartis, N. I. Gould, and P. L. Toint. On the complexity of finding first-order critical points in constrained nonlinear optimization. *Mathematical Programming*, 144(1-2):93–106, 2014.

[13] 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 Journal on Numerical Analysis*, 53(2):836–851, 2015.

[14] C. Cartis, N. I. Gould, and P. L. Toint. Second-order optimality and beyond: Characterization and evaluation complexity in convexly constrained nonlinear optimization. *Foundations of Computational Mathematics*, pages 1–35, 2017.

[15] A. R. Conn, N. I. Gould, and P. L. Toint. *Trust region methods*, volume 1. Siam, 2000.

[16] F. E. Curtis and D. P. Robinson. Exploiting negative curvature in deterministic and stochastic optimization. *arXiv preprint arXiv:1703.00412*, 2017.

[17] F. E. Curtis, D. P. Robinson, and M. Samadi. An inexact regularized newton framework with a worst-case iteration complexity of o($\epsilon^{-3/2}$) for nonconvex optimization. *arXiv preprint arXiv:1708.00475*, 2017.

[18] F. E. Curtis, D. P. Robinson, and M. Samadi. A trust region algorithm with a worst-case iteration complexity of o($\epsilon^{-3/2}$) for nonconvex optimization. *Mathematical Programming*, 162(1-2):1–32, 2017.

[19] 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 *Advances in Neural Information Processing Systems*, pages 1067–1077, 2017.

[20] R. Ge, F. Huang, C. Jin, and Y. Yuan. Escaping from saddle pointsonline stochastic gradient for tensor decomposition. In *Conference on Learning Theory*, pages 797–842, 2015.

[21] R. Ge, J. D. Lee, and T. Ma. Matrix completion has no spurious local minimum. In *Advances in Neural Information Processing Systems*, pages 2973–2981, 2016.

[22] M. Hong, J. D. Lee, and M. Razaviyayn. Gradient primal-dual algorithm converges to second-order stationary solutions for nonconvex distributed optimization. *arXiv preprint arXiv:1802.08941*, 2018.

[23] Y. Hsia and R.-L. Sheu. Trust region subproblem with a fixed number of additional linear inequality constraints has polynomial complexity. *arXiv preprint arXiv:1312.1398*, 2013.

[24] C. Jin, R. Ge, P. Netrapalli, S. M. Kakade, and M. I. Jordan. How to escape saddle points efficiently. *arXiv preprint arXiv:1703.00887*, 2017.

[25] J. D. Lee, I. Panageas, G. Piliouras, M. Simchowitz, M. I. Jordan, and B. Recht. First-order methods almost always avoid saddle points. *arXiv preprint arXiv:1710.07406*, 2017.

[26] J. D. Lee, M. Simchowitz, M. I. Jordan, and B. Recht. Gradient descent converges to minimizers. *arXiv preprint arXiv:1602.04915*, 2016.

[27] 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. *in prep.*, 2019.

[28] S. Lu, Z. Wei, and L. Li. A trust region algorithm with adaptive cubic regularization methods for nonsmooth convex minimization. *Computational Optimization and Applications*, 51(2):551–573, 2012.

[29] A. Mokhtari, A. Ozdaglar, and A. Jadbabaie. Escaping saddle points in constrained optimization. *arXiv preprint arXiv:1809.02162*, 2018.

[30] K. G. Murty and S. N. Kabadi. Some np-complete problems in quadratic and nonlinear programming. *Mathematical programming*, 39(2):117–129, 1987.

[31] Y. Nesterov and B. T. Polyak. Cubic regularization of newton method and its global performance. *Mathematical Programming*, 108(1):177–205, 2006.

[32] M. Nouiehed, J. D. Lee, and M. Razaviyayn. Convergence to second-order stationarity for constrained non-convex optimization. *arXiv preprint arXiv:1810.02024*, 2018.

[33] M. Nouiehed and M. Razaviyayn. Learning deep models: Critical points and local openness. *arXiv preprint arXiv:1803.02968*, 2018.

[34] M. Nouiehed, M. Sanjabi, J. D. Lee, and M. Razaviyayn. Solving a class of non-convex min-max games using iterative first order methods. *arXiv preprint arXiv:1902.08297*, 2019.

[35] M. O'Neill and S. J. Wright. A log-barrier newton-cg method for bound constrained optimization with complexity guarantees. *arXiv preprint arXiv:1904.03563*, 2019.

[36] J. Sun, Q. Qu, and J. Wright. A geometric analysis of phase retrieval. In *Information Theory (ISIT), 2016 IEEE International Symposium on*, pages 2379–2383. IEEE, 2016.

[37] J. Sun, Q. Qu, and J. Wright. Complete dictionary recovery over the sphere i: Overview and the geometric picture. *IEEE Transactions on Information Theory*, 63(2):853–884, 2017.

[38] J. Sun, Q. Qu, and J. Wright. Complete dictionary recovery over the sphere ii: Recovery by riemannian trust-region method. *IEEE Transactions on Information Theory*, 63(2):885–914, 2017.

A   Proofs for Section 4

Consider the following optimization problem

$$\underset{\mathbf{x} \in \mathcal{P}}{\text{minimize}} \; f(\mathbf{x}), \tag{22}$$

where $\mathcal{P} \triangleq \{\mathbf{x} \in \mathbb{R}^n \,|\, \mathbf{A}\mathbf{x} \le \mathbf{b}\}$ is a polyhedron with finite number of linear constraints. In this section we generalize results from [18] to adapt for the linear constraints. For the sake of completeness of the manuscript, some Lemmas and proofs are restated from [18].

Recall the sub-problem Q_k with trust region δ_k,

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

Let $\boldsymbol{\lambda}_k^C$ be the multiplier corresponding to the linear constraint $\mathbf{A}\mathbf{s} \le \mathbf{b} - \mathbf{A}\mathbf{x}_k$, and λ_k be the multiplier for the trust region constraint $\|\mathbf{s}\|_2 \le \delta_k$. The first order K.K.T optimality conditions for the above problem are state below [4]

$$\mathbf{g}_k + (\mathbf{H}_k + \lambda_k \mathbf{I})\mathbf{s}_k + \mathbf{A}^T \boldsymbol{\lambda}_k^C = \mathbf{0}, \tag{23}$$

$$\mathbf{0} \le \boldsymbol{\lambda}_k^C \perp \mathbf{b} - \mathbf{A}\mathbf{x}_k - \mathbf{A}\mathbf{s}_k \ge \mathbf{0}, \tag{24}$$

$$0 \le \lambda_k \perp \delta_k - \|\mathbf{s}_k\|_2^2 \ge 0. \tag{25}$$

A.1   Proof of Theorem 7

To show convergence to first-order stationarity, we first provide in Lemma 11 a sufficient decrease condition. Then, in Lemma 17 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. In practice, the algorithm terminates when $\mathcal{X}_k$ is below a prescribed positive threshold $\epsilon > 0$. Hence, we assume, without loss of generality that $\mathcal{X}_k \ge \epsilon$ for all $k \in \mathbb{N}$.

Lemma 11. *For any $k \in \mathbb{N}$, the trial step $\mathbf{s}_k$ and dual variable λ_k satisfy*

$$f_k - q_k(\mathbf{s}_k) \ge \frac{1}{2}\mathbf{s}_k^T(\mathbf{H}_k + \lambda_k \mathbf{I})\mathbf{s}_k + \frac{1}{2}\lambda_k \|\mathbf{s}_k\|_2^2. \tag{26}$$

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

$$f_k - q_k(\mathbf{s}_k) \ge C\mathcal{X}_k \min\left\{\delta_k, \frac{\mathcal{X}_k}{\|\mathbf{H}_k\|_2}, 1\right\}. \tag{27}$$

Proof. By definition of q_k,

$$
\begin{aligned}
f_k - q_k(\mathbf{s}_k) &= -\mathbf{g}_k^T \mathbf{s}_k - \frac{1}{2}\mathbf{s}_k^T \mathbf{H}_k \mathbf{s}_k \\
&= \mathbf{s}_k^T \mathbf{H}_k \mathbf{s}_k + \lambda_k \|\mathbf{s}_k\|_2^2 + \mathbf{s}_k^T \mathbf{A}^T(\boldsymbol{\lambda}_k^C) - \frac{1}{2}\mathbf{s}_k^T \mathbf{H}_k \mathbf{s}_k \\
&= \frac{1}{2}\mathbf{s}_k^T(\mathbf{H}_k + \lambda_k \mathbf{I})\mathbf{s}_k + \frac{1}{2}\lambda_k \|\mathbf{s}_k\|^2 + \mathbf{s}_k^T \mathbf{A}^T(\boldsymbol{\lambda}_k^C) \\
&\ge \frac{1}{2}\mathbf{s}_k^T(\mathbf{H}_k + \lambda_k \mathbf{I})\mathbf{s}_k + \frac{1}{2}\lambda_k \|\mathbf{s}_k\|^2, \tag{28}
\end{aligned}
$$

where the second equality follows by KKT condition (23), and the last inequality follows from the feasibility of $\mathbf{x}_k$ and the complementary slackness (24).

16

Also, using [15, Theorem 12.2.2], we obtain

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

$\square$

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 12. *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 $\|\mathbf{s}\|$. $\square$

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

Lemma 13. *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 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 1. Assume the induction hypothesis holds for iteration k. By the computations in Steps 7, 8, 16, 17 and by Lemma 12, 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 (25) 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 $\delta_k \leq \Delta_k \leq \Delta_{k+1}$. $\square$

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

Lemma 14. *For any $k \in \mathbb{N}$, if $k \in \mathcal{C} \cup \mathcal{E}$, then $k + 1 \notin \mathcal{E}$.*

17

Proof. Observe that if $\lambda_{k+1} = 0$, then conditions in Steps 14 of Algorithm 1 ensure that $(k + 1) \notin \mathcal{E}$. Thus, by (25), 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

$$\lambda_k > \sigma_k \|\mathbf{s}_k\|_2, \quad \delta_{k+1} = \min\{\Delta_k, \lambda_k/\sigma_k\}, \quad \text{and} \quad \sigma_{k+1} = \sigma_k. \tag{29}$$

Combined with (25), 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 (29) that

$$\delta_{k+1} = \lambda_k/\sigma_k > \|\mathbf{s}_k\|_2 = \delta_k. \tag{30}$$

   Therefore, by the relationship between the trust region radius and its corresponding multiplier, we get $\lambda_{k+1} \leq \lambda_k$. Combined with (29) and (30), 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 (29)

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

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

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

$$\lambda_k \geq g_{Lip} + H_{max} + \rho \|\mathbf{s}_k\|_2, \tag{31}$$

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

$$
\begin{aligned}
q_k(\mathbf{s}_k) - f(\mathbf{x}_k + \mathbf{s}_k) = & \left(\mathbf{g}_k - \mathbf{g}(\bar{\mathbf{x}}_k)\right)^T \mathbf{s}_k + \frac{1}{2}\mathbf{s}_k^T \mathbf{H}_k \mathbf{s}_k \\
& \geq -\|\mathbf{g}_k - \mathbf{g}(\bar{\mathbf{x}}_k)\|_2 \|\mathbf{s}_k\|_2 - \frac{1}{2}\|\mathbf{H}_k\|_2 \|\mathbf{s}_k\|_2^2.
\end{aligned}
\tag{32}
$$

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 11 and expression (32). The result $\|\mathbf{s}_k\| = \delta_k$ follows directly from (31) and (25). $\qquad\square$

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 16. *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 12 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

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

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 $\|\mathbf{s}_k\|_2 = \delta_k \to 0$. $\square$

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

Lemma 17. *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 14. Thus, $|\mathcal{C}| = \infty$. Combining this with the result of Lemma 14, 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 16 that $\{\|\mathbf{s}_k\|_2\} \leq \{\delta_k\} \to 0$ and $\{\lambda_k\} \to \infty$. In combination with Lemma 15, 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. $\square$

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 18. *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 1 that

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

Combining this with Lemma 17 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

$$\gamma_E\|\mathbf{s}_k\|_2 \leq \Delta_0 \leq \Delta_k, \tag{33}$$

where the latter inequality follows from Lemma 13. Combined with the update in Steps 7, 12 and 16 of LC-TRACE, we get

$$\Delta_{k+1} = \Delta_k \text{ for all } k \geq k_\mathcal{A}.$$

This proves the first part of the lemma. The second part also follows from (33) 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 13 ensures $\|\mathbf{s}_k\|_2 \leq \delta_k \leq \Delta_k = \Delta$ for all sufficiently large $k \in \mathbb{N}$. $\square$

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 19. *For all $k \in \mathbb{N}$, $\|\mathbf{g}_k + \mathbf{A}^T\boldsymbol{\lambda}_k^C\|_2 \leq G_{max}$, where $G_{max} > 0$ is a constant scalar. Moreover,*

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

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

Proof. By (23) and Lemma 13,

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

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 (25), $\lambda_{k+1} = 0$. Therefore, we may proceed under the assumption that $\|\mathbf{s}_{k+1}\|_2 = \delta_{k+1}$. Suppose $k \in \mathcal{C}$, then by Lemma 15, $\lambda_k < g_{Lip} + H_{max} + \rho\Delta$.

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

$$\begin{aligned}
\lambda_{k+1} &\leq \lambda_k + H_{max} + \underline{\sigma}\Delta_k + (\underline{\sigma}\mathcal{X}_k)^{1/2} \\
&\leq g_{Lip} + 2H_{max} + (\rho + \underline{\sigma})\Delta + (\underline{\sigma}g_{max})^{1/2}.
\end{aligned} \tag{34}$$

Otherwise, if Step 3 tests false, we claim that

$$\lambda_{k+1} \leq \gamma_\lambda(g_{Lip} + H_{max} + \rho\Delta). \tag{35}$$

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

$$q_k(\hat{\mathbf{s}}) - f(\hat{\mathbf{x}} + \hat{\mathbf{s}}) = \left(\mathbf{g}_k - \mathbf{g}(\hat{\mathbf{x}}_k)\right)^T \mathbf{s}_k + \frac{1}{2}\hat{\mathbf{s}}^T\mathbf{H}_k\hat{\mathbf{s}} \geq -g_{Lip}\|\hat{\mathbf{s}}\|_2^2 - \frac{1}{2}H_{max}\|\hat{\mathbf{s}}\|_2^2. \tag{36}$$

Therefore,

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

where the first inequality holds by Lemma 11 and (36). 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 (34) and (35), we get that for all $k \in \mathcal{C}$

$$\lambda_{k+1} \leq \lambda_{max}, \tag{37}$$

20

where $\lambda_{max} \triangleq \max\{g_{Lip} + 2H_{max} + (\rho + \underline{\sigma})\Delta + (\underline{\sigma}g_{max})^{1/2}, \gamma_\lambda(g_{Lip} + H_{max} + \rho\Delta)\}$. Now, suppose that $k \in \mathcal{A} \cup \mathcal{E}$. By Lemma 13, we have $\|\mathbf{s}_{k+1}\|_2 = \delta_{k+1} \geq \delta_k \geq \|\mathbf{s}_k\|_2$. Hence, by the relationship between the trust region radius and its corresponding multiplier, we obtain

$$\lambda_{k+1} \leq \lambda_k. \tag{38}$$

Let $k_\mathcal{C} \triangleq \min\{k \in \mathbb{N} \mid k \in \mathcal{C}\}$ be the first contract step. By (38), $\lambda_k \leq \lambda_0$ for all $k \leq k_\mathcal{C}$. Moreover, using (37) and (38),

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

Combining these results yield

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

which completes the proof. $\qquad\square$

Notice that in the proof of Lemma 19, we have shown that there exists a uniform upper bound for the dual variables λ_k. Our next result shows that the ratio $\dfrac{\mathcal{X}_k}{\|\mathbf{s}\|_2}$ is upper bounded by $C_{min} + \lambda_k$, where C_{min} is a scalar constant.

Lemma 20. *For any $k \in \mathbb{N}$, it holds*

$$\mathcal{X}_k \leq (C_{min} + \lambda_k)\|\mathbf{s}_k\|_2, \tag{39}$$

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 $\mathbf{Ad} \leq \mathbf{b} - \mathbf{Ax}_k$, we have

$$\begin{aligned}
\mathbf{g}_k^T \mathbf{d} &= -\mathbf{d}^T(\mathbf{H}_k + \lambda_k \mathbf{I})\mathbf{s}_k^T - (\boldsymbol{\lambda}_k^C)^T \mathbf{Ad} \\
&\geq -\mathbf{d}^T(\mathbf{H}_k + \lambda_k \mathbf{I})\mathbf{s}_k^T - (\boldsymbol{\lambda}_k^C)^T \mathbf{As}_k \\
&\geq -(\xi_{k,1} + \lambda_k)\|\mathbf{d}\|_2 \|\mathbf{s}_k\|_2 - (\boldsymbol{\lambda}_k^C)^T \mathbf{As}_k,
\end{aligned} \tag{40}$$

where the first equality holds by (23), and the first inequality holds by complementary slackness (24). Minimizing over all such $\mathbf{d}$, we obtain

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

where the last inequality uses Lemma 19. Then definition of $\mathcal{X}_k$ yields

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

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

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

Lemma 21. *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 19 and 20, 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 17 that insures infinite cardinality of set $\mathcal{A}$, we have $\{\mathbf{s}_k\}_{k \in \mathcal{A}} \to 0$. This contradicts the existence of $s_{min} > 0$. $\square$

Theorem 22. *Under Assumption 6, it holds that*

$$\lim_{k \in \mathbb{N},\, k \to \infty} \mathcal{X}_k = 0. \tag{41}$$

Proof. Suppose the contrary that (41) does not hold. Combined with Lemmas 17 and 21, it implies that there exist an infinite sub-sequence $\{t_i\} \subseteq \mathcal{A}$ (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 17 and 21 imply that there exist an infinite subsequence $\{l_i\} \subseteq \mathcal{A}$ such that

$$\mathcal{X}_k \geq \epsilon_{\mathcal{X}} \text{ and } \mathcal{X}_{l_i} < \epsilon_{\mathcal{X}} \quad \forall\, i \in \mathbb{N},\ k \in \mathbb{N},\ t_i \leq k < l_i. \tag{42}$$

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

$$\|\mathbf{s}_k\|_2 \geq \min\left\{\delta_k, \frac{\mathcal{X}_k}{C_{min}}\right\}. \tag{43}$$

The proof of this claim follows directly from Lemma 20. If $\|\mathbf{s}_k\|_2 = \delta_k$, the result trivially holds. Otherwise, using KKT condition (25), $\lambda_k = 0$ which proves our claim when combined with Lemma 20. 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 (42) and (43) that

$$f_k - f_{k+1} \geq \rho\|\mathbf{s}_k\|_2^3 \geq \rho\left(\min\left\{\delta_k, \frac{\epsilon_{\mathcal{X}}}{C_{min}}\right\}\right)^3. \tag{44}$$

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 (44), we obtain

$$\lim_{k \in \mathcal{K},\, k \to \infty} \delta_k = 0. \tag{45}$$

Using this fact and Lemma 11, we have for all sufficiently large $k \in \mathcal{K}$ that

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

22

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

$$\|\mathbf{x}_{t_i} - \mathbf{x}_{l_i}\|_2 \leq \sum_{k \in \mathcal{K}, k=t_i}^{l_i-1} \|\mathbf{x}_k - \mathbf{x}_{k+1}\|_2$$

$$\leq \sum_{k \in \mathcal{K}, k=t_i}^{l_i-1} \delta_k \leq \sum_{k \in \mathcal{K}, k=t_i}^{l_i-1} \frac{2}{C\epsilon_{\mathcal{X}}}(f_k - f_{k+1}) = \frac{2}{C\epsilon_{\mathcal{X}}}(f_{t_i} - f_{l_i}).$$

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 (42). $\qquad \square$

A.2 Proof of Theorem 9

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 showing the desired model decrease using Assumption 8.

Lemma 23. *Consider the directions* $\mathbf{s}$, $\mathbf{s}^+$ *and points* $\mathbf{x} = \mathbf{x}_k + \mathbf{s}$, $\mathbf{x}^+ = \mathbf{x}_k + \mathbf{s}^+$. *If for some* $\bar{\kappa} \in (0,1]$

$$f_k - q_k(\mathbf{s}) \geq \bar{\kappa}\lambda_k\|\mathbf{s}\|_2^2, \tag{46}$$

$$q_k(\mathbf{s}^+) \leq q_k(\mathbf{s}), \tag{47}$$

$$\mathbf{g}_k^T(\mathbf{s}^+ - \mathbf{s}) + (\mathbf{s}^+)^T\mathbf{H}_k(\mathbf{s}^+ - \mathbf{s}) \leq -\lambda_k(\mathbf{s}^+)^T(\mathbf{s}^+ - \mathbf{s}), \tag{48}$$

$$\mathbf{g}_k^T(\mathbf{s}^+ - \mathbf{s}) + \mathbf{s}^T\mathbf{H}_k(\mathbf{s}^+ - \mathbf{s}) \leq -\lambda_k\mathbf{s}^T(\mathbf{s}^+ - \mathbf{s}), \tag{49}$$

then

$$f_k - q_k(\mathbf{s}^+) \geq \frac{1}{3}\bar{\kappa}\lambda_k\|\mathbf{s}^+\|_2^2.$$

Proof. Suppose that for a given constant scalar $\alpha \in (0,1)$, $\|\mathbf{s}\|_2 \geq \alpha\|\mathbf{s}^+\|_2$. Then, it directly follows by (46) and (47) that

$$f_k - q_k(\mathbf{s}^+) = f_k - q_k(\mathbf{s}) + q_k(\mathbf{s}) - q_k(\mathbf{s}^+) \geq \bar{\kappa}\lambda_k\|\mathbf{s}\|_2^2 \geq \bar{\kappa}\alpha^2\lambda_k\|\mathbf{s}^+\|_2^2. \tag{50}$$

Now consider the case that $\|\mathbf{s}\|_2 < \alpha\|\mathbf{s}^+\|_2$. First note that by (48),

$$\begin{aligned} 0 &\geq (\mathbf{g}_k + \mathbf{H}_k\mathbf{s}^+)^T(\mathbf{s}^+ - \mathbf{s}) + \lambda_k(\mathbf{s}^+)^T(\mathbf{s}^+ - \mathbf{s}) \\ &= (\mathbf{g}_k + \mathbf{H}_k\mathbf{s})^T(\mathbf{s}^+ - \mathbf{s}) + (\mathbf{s}^+ - \mathbf{s})^T\mathbf{H}_k(\mathbf{s}^+ - \mathbf{s}) + \lambda_k(\mathbf{s}^+)^T(\mathbf{s}^+ - \mathbf{s}). \end{aligned} \tag{51}$$

Also, by (49)

$$(\mathbf{g}_k + \mathbf{H}_k\mathbf{s})^T(\mathbf{s}^+ - \mathbf{s}) + \lambda_k\mathbf{s}^T(\mathbf{s}^+ - \mathbf{s}) \leq 0. \tag{52}$$

Adding (51) and (52), we get

$$\begin{aligned} q_k(\mathbf{s}^+) - q_k(\mathbf{s}) &= (\mathbf{g}_k + \mathbf{H}_k\mathbf{s})^T(\mathbf{s}^+ - \mathbf{s}) + \frac{1}{2}(\mathbf{s}^+ - \mathbf{s})^T\mathbf{H}_k(\mathbf{s}^+ - \mathbf{s}) \\ &\leq -\frac{1}{2}\lambda_k(\|\mathbf{s}^+\|_2^2 - \|\mathbf{s}\|_2^2). \end{aligned} \tag{53}$$

Since $\|\mathbf{s}\|_2 < \alpha\|\mathbf{s}^+\|_2$, it follows that

$$f_k - q_k(\mathbf{s}^+) \geq q_k(\mathbf{s}) - q_k(\mathbf{s}^+) \geq \frac{1}{2}\lambda_k(\|\mathbf{s}^+\|_2^2 - \|\mathbf{s}\|_2^2) \geq \frac{1}{2}\lambda_k\bar{\kappa}\|\mathbf{s}^+\|_2^2(1 - \alpha^2), \tag{54}$$

where the first inequality holds by (46), the second inequality holds by (53), and the last inequality holds because $\|\mathbf{s}\|_2 < \alpha\|\mathbf{s}^+\|_2$. We now choose the value of α for which the lower bounds (50) and (54) are equal; i.e. $\alpha^2 = \frac{1}{2}(1 - \alpha^2)$, equivalently $\alpha = \sqrt{\frac{1}{3}}$. $\qquad \square$

We next show that sufficient model decrease is satisfied when either $\mathbf{g}_k^T \mathbf{s}_k \leq 0$ or $\mathbf{s}_k^T \mathbf{H}_k \mathbf{s}_k \geq 0$.

Lemma 24. *Suppose that $\mathbf{g}_k^T \mathbf{s}_k \leq 0$ or $\mathbf{s}_k^T \mathbf{H}_k \mathbf{s}_k \geq 0$. Then,*

$$f_k - q_k(\mathbf{s}_k) = -\mathbf{g}_k^T \mathbf{s}_k - \frac{1}{2}\mathbf{s}_k^T \mathbf{H}_k \mathbf{s}_k \geq \frac{1}{2}\lambda_k \|\mathbf{s}_k\|_2^2. \tag{55}$$

Proof. First notice that since the origin is feasible in Q_k,

$$\mathbf{g}_k^T \mathbf{s}_k + \frac{1}{2}\mathbf{s}_k^T \mathbf{H}_k \mathbf{s}_k \leq 0.$$

Hence,

$$\mathbf{s}_k^T \mathbf{H}_k \mathbf{s}_k \geq 0 \Rightarrow \mathbf{g}_k^T \mathbf{s}_k \leq 0.$$

On the other hand, if $\mathbf{g}_k^T \mathbf{s}_k \leq 0$, by (23),

$$2\big(\mathbf{g}_k^T \mathbf{s}_k + \frac{1}{2}\mathbf{s}_k^T \mathbf{H}_k \mathbf{s}_k + \frac{1}{2}\lambda_k \|\mathbf{s}_k\|_2^2\big) = \mathbf{g}_k^T \mathbf{s}_k - \mathbf{s}_k^T \mathbf{A}^T \boldsymbol{\lambda}_k^C \leq \mathbf{g}_k^T \mathbf{s}_k \leq 0, \tag{56}$$

where the first inequality holds due to the complementary slackness condition (24). $\qquad\square$

Lemma 25. *Suppose Assumption 8 holds at iteration k. Then there exist a constant $\kappa > 0$ independent of k such that*

$$f_k - q_k(\mathbf{s}_k) \geq \kappa \lambda_k \|\mathbf{s}_k\|_2^2.$$

Proof. If $\mathbf{g}_k^T \mathbf{s}_k \leq 0$ or $\mathbf{s}_k^T \mathbf{H}_k \mathbf{s}_k \geq 0$, the result follows by Lemma 24. Thus, we may proceed under the assumption that $\mathbf{g}_k^T \mathbf{s}_k \geq 0$ and $\mathbf{s}_k^T \mathbf{H}_k \mathbf{s}_k \leq 0$. Using Assumption 8, we proceed with a proof by induction on l_k. If $l_k = 1$, the last condition in Assumption 8 implies that $\mathbf{g}_k^T \mathbf{s}_k \leq 0$, thus Lemma 24 implies the desired result. Now assume the result holds for $l_k = i < \bar{l}$, we next show that the result holds for $l_k = i + 1$. By the induction step and Assumption 8, we have

$$f_k - q_k(\mathbf{s}_{k,i}) \geq \bar{\kappa}\lambda_k \|\mathbf{s}_{k,i}\|_2^2,$$

$$q_k(\mathbf{s}_{k,i+1}) \leq q_k(\mathbf{s}_{k,i}),$$

$$\big\langle \mathbf{g}_k + \mathbf{H}_k \mathbf{s}_{k,i+1}, \mathbf{x}_{k,i+1} - \mathbf{x}_{k,i} \big\rangle \leq -\lambda_k \mathbf{s}_{k,i+1}^T(\mathbf{x}_{k,i+1} - \mathbf{x}_{k,i+1}),$$

$$\big\langle \mathbf{g}_k + \mathbf{H}_k \mathbf{s}_{k,i}, \mathbf{x}_{k,i+1} - \mathbf{x}_{k,i} \big\rangle \leq -\lambda_k \mathbf{s}_{k,i}^T(\mathbf{x}_{k,i+1} - \mathbf{x}_{k,i+1}).$$

Then using Lemma 23, with $\mathbf{x} = \mathbf{x}_{k,i}$ and $\mathbf{x}^+ = \mathbf{x}_{k,i+1}$ we obtain

$$f_k - q_k(\mathbf{s}_{k,i+1}) \geq \bar{\kappa}^+ \lambda_k \|\mathbf{s}_{k,i+1}\|_2^2,$$

for some $\bar{\kappa}^+ \in (0, 1)$ independent of k. $\qquad\square$

Our next result provides a bound on the ratio $\lambda_{k+1}/\|\mathbf{s}_{k+1}\|_2$ when $k \in \mathcal{C}$.

Lemma 26. *Assume Assumption 8 holds at iteration $k \in \mathcal{C}$. Then,*

- *If Step 6, 9, or 21 of Algorithm 2 is reached, then*

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

- *If Step 23 of Algorithm 2 is reached, then*

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

24

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 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

$$\underline{\sigma} \leq \frac{\lambda_k + \underline{\sigma}\Delta_k}{\Delta_k} \leq \frac{\lambda_{k+1}}{\|\mathbf{s}_{k+1}\|_2} = \frac{\lambda}{\|\mathbf{s}\|_2} \leq \overline{\sigma}, \tag{57}$$

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 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 20 and the fact that $\mathcal{X}_{k+1} = \mathcal{X}_k$ for all $k \in \mathcal{C}$,

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

which contradicts our assumption on $\mathcal{X}_k$. Here the last inequality uses the definition (19) of $\underline{\sigma}$. Combined with Lemma 13 we obtain

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

Therefore,

$$\underline{\sigma} \leq \frac{\lambda_k + \underline{\sigma}\Delta_k}{\Delta_k} \leq \frac{\bar{\lambda}}{\|\bar{\mathbf{s}}\|_2} \leq \frac{\lambda_k + \underline{\sigma}\Delta_k}{\underline{\sigma}} \leq \frac{\underline{\sigma}(\|\mathbf{s}_k\|_2 + \Delta_k)}{\underline{\sigma}} \leq 2\Delta_k \leq \overline{\sigma}, \tag{58}$$

where the fourth inequality holds by the condition of Step 3 and the last inequality holds by Lemmas 13, 18 and the definition of $\overline{\sigma}$ in 19.

The other case correspond to situations in which the condition in Step 3 tests false. It follows by Steps 2 and 3 that

$$\underline{\sigma} \leq \frac{\lambda}{\|\mathbf{s}\|_2}, \tag{59}$$

Finally, using the argument of Lemma 19, we claim that

$$\lambda_{k+1} \leq \gamma_\lambda \left(\frac{H_{Lip} + 2\rho}{2\kappa}\right)\|\mathbf{s}_k\|_2. \tag{60}$$

To show our claim, we assume the contrary, i.e. $\lambda_{k+1} > \gamma_\lambda \left(\frac{H_{Lip} + 2\rho}{2\kappa}\right)\|\mathbf{s}_k\|_2$. 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 $\hat{\lambda} \geq (\frac{H_{Lip} + 2\rho}{2\kappa})\|\hat{\mathbf{s}}\|_2$. There exists $\hat{\mathbf{x}}$ on the line segment $[\mathbf{x}_k, \mathbf{x}_k + \hat{\mathbf{s}}]$ such that

$$q_k(\hat{\mathbf{s}}) - f(\hat{\mathbf{x}} + \hat{\mathbf{s}}) = \frac{1}{2}\hat{\mathbf{s}}^T(\mathbf{H}_k - \mathbf{H}(\mathbf{x}_k))\hat{\mathbf{s}} \geq -\frac{1}{2}H_{Lip}\|\hat{\mathbf{s}}\|_2^3. \tag{61}$$

Therefore,

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

where the first inequality holds by Lemma 25 and (61), and the last inequality holds since $\|\mathbf{s}_k\|_2 \geq \|\hat{\mathbf{s}}\|_2$. However $\rho_k = \dfrac{f_k - f(\mathbf{x}_k + \mathbf{s}_k)}{\|\mathbf{s}_k\|_2^3} < \rho$. 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 22 in Algorithm 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 (59), (60), and the condition in Step 20 of the CONTRACT sub-routine, we observe that

$$\underline{\sigma} \leq \frac{\lambda_{k+1}}{\|\mathbf{s}_{k+1}\|_2} = \frac{\lambda}{\|\mathbf{s}\|_2} \leq \left(\frac{\gamma_\lambda}{\gamma_C}\right)\frac{H_{Lip} + 2\rho}{2\kappa} \tag{62}$$

- Suppose that Step 23 is reached. Then, $\delta_{k+1} = \gamma_C\|\mathbf{s}_k\|_2$. It follows that Step 22 in Algorithm 1 will produce the primal-dual pair $(\mathbf{s}_{k+1}, \lambda_{k+1}) = (\mathbf{s}, \lambda)$. If $\|\mathbf{s}\|_2 < \delta_{k+1} = \gamma_C\|\mathbf{s}_k\|_2$, then $\lambda_{k+1} = 0$. Otherwise, $\|\mathbf{s}\|_2 = \delta_{k+1} = \gamma_C\|\mathbf{s}_k\|_2$ and $\lambda_{k+1} > 0$. Combined with (59) and (60), we obtain

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

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 27. *Assume Assumption 8 holds. 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 18, 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 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 1 and Lemma 26, it follows that

$$\sigma_{k+1} = \max\left\{\sigma_k, \frac{\lambda_{k+1}}{\|\mathbf{s}_{k+1}\|_2}\right\} \leq \max\left\{\sigma_k, \overline{\sigma}, \left(\frac{\gamma_\lambda}{\gamma_C}\right)\frac{H_{Lip} + 2\rho}{2\kappa}\right\}. \tag{63}$$

- If $k \in \mathcal{E}$, then Step 18 of Algorithm 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 28. *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

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

where the first equation follows from (23) and the last inequality follows since $\lambda_k \leq \sigma_k \|\mathbf{s}_k\|_2$ for all $k \in \mathcal{A}_\sigma$. Thus

$$
\min_{\text{s.t. } d \in \mathcal{D}_{k+1}} \mathbf{g}_{k+1}^T \mathbf{d} \geq -H_{Lip} \|\mathbf{s}_k\|_2^2 - \sigma_k \|\mathbf{s}_k\|_2^2 - \max_{\text{s.t. } d \in \mathcal{D}_{k+1}} \mathbf{d}^T \mathbf{A}^T \boldsymbol{\lambda}_k^C, \tag{64}
$$

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 $\mathcal{I}_k \triangleq \{i \,|\, \mathbf{a}_i^T \mathbf{s}_k = b_i - \mathbf{a}_i^T \mathbf{x}_k\}$, 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 (64), 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 27, 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 29. *Assume Assumption 8 holds. 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

$$
\left\lceil \left(\frac{f_0 - f_{min}}{\rho (H_{Lip} + \sigma_{max})^{-3/2}} \right) \epsilon^{-3/2} \right\rceil \triangleq K_\sigma(\epsilon) \geq 0. \tag{65}
$$

Proof. By Lemma 28, 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 22 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 30 and 33.

Lemma 30. *The cardinality of the set $\mathcal{A}_\Delta$ is upper-bounded by*

$$
\left\lceil \frac{f_0 - f_{min}}{\rho \Delta_0^3} \right\rceil \triangleq K_\Delta \geq 0. \tag{66}
$$

27

Proof. For all $k \in \mathcal{A}_\Delta$, it follows by Lemma 13 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 31. *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 14 that that $\mathcal{E} \cap \mathcal{I}(\hat{k}) \subseteq \{\hat{k}+1\}$, as desired. □

Lemma 32. *For any $k \in \mathbb{N}_+$, if $k \in \mathcal{C}$ and Step 21 in Algorithm 2 is reached, then*

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

Proof. If Step 21 is reached, then $\|\mathbf{s}\|_2 \geq \gamma_C \|\mathbf{s}_k\|_2$. It follows that Step 22 in Algorithm 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.,

$$\frac{\lambda_{k+1}}{\|\mathbf{s}_{k+1}\|_2} \geq \frac{\gamma_\lambda \lambda_k}{\|\mathbf{s}_k\|_2}. \tag{67}$$

□

Lemma 33. *Assume Assumptions 6 and 8 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 CONTRACT sub-routine reached, 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 20, 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 Lemma 13, Lemma 12, and the fact that each time 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

$$\mathcal{I}_\mathcal{C}(\hat{k}) \triangleq \{k \in \mathcal{C} \cap \mathcal{I}(\hat{k}) : \text{ Step 23 is reached in Algorithm 2}\}$$

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 26, 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 CONTRACT subroutine is reached for every $k_{\mathcal{C},1}(\hat{k}) + 2 < k < k_{\mathcal{C},2}(\hat{k})$. By Lemmas 27 and 32, 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 $|\mathcal{C} \cap \mathcal{I}(\hat{k})| = 1 + K_{C,1}\left(\frac{1}{\log(\gamma_\lambda)}\log\left(\frac{\sigma_{max}}{\underline{\sigma}}\right) + 2\right).$ $\qquad\square$

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 sub-routine, completing a contract step may require solving more than one sub-problem. Our next result provides an upper bound on the number of subproblem routine calls required in one contract step.

Lemma 34. *Assume Assumptions 6 and 8 hold. For a scalar $\epsilon \in (0, \infty)$, the total number of sub-problems 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,

$$\lambda \geq \lambda_k\gamma_\lambda^{\log\left((C_{min}+\max\{\lambda_{max},\lambda_0\})\frac{H_{max} + g_{Lip} + \rho\Delta}{\underline{\sigma}\epsilon}\right)}$$

$$\geq \underline{\sigma}\|\mathbf{s}_k\|_2\left(C_{min} + \max\{\lambda_{max}, \lambda_0\}\right)\frac{H_{max} + g_{Lip} + \rho\Delta}{\underline{\sigma}\epsilon}$$

$$\geq H_{max} + g_{Lip} + \rho\Delta,$$

where the last inequality holds by Lemma 20 and the fact that $\mathcal{X}_k \geq \epsilon$. Hence, by Lemma 15

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

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 35. *Under Assumptions 6 and 8, 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

$$K(\epsilon) \triangleq 1 + (K_\sigma(\epsilon) + K_\Delta)(1 + K_{\mathcal{C}} K_{\mathcal{C}}^1), \tag{68}$$

where $K_\sigma(\epsilon)$, K_Δ, $K_{\mathcal{C}}^1$, and $K_{\mathcal{C}}$ are defined in Lemmas 29, 30, 33, and 34 respectively. Hence, for $\epsilon_g > 0$, the number of sub-problem 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 29 and 30 guarantee that the total number of elements in the index set $\{k \in \mathcal{A} \mid k \geq 1, \mathcal{X}_k > \epsilon\}$ is at most $K_\sigma(\epsilon) + K_\Delta$. Also, immediately prior to each of the corresponding accepted steps, Lemmas 31, 33, and 34 guarantee that at most $1 + K_C K_C^1$ sub-problem routine calls are required in expansion and contraction. Accounting for the first iteration, the desired result follows. □

A.3 Proof of Theorem 10

.

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

$$\mathcal{V} \triangleq \{k \mid \text{Step 7 in Algorithm 3 is reached at iteration } k\}.$$

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

$$f_{k-1} - f_{k+1} \geq \min\left\{\frac{\rho \mathcal{X}_k^{3/2}}{(H_{Lip} + \sigma_{max})^{3/2}}, \frac{2\psi_k^3}{3\tilde{H}^2}\right\} \tag{69}$$

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

$$\begin{aligned}
f_{k+1} &\leq f_k + \langle \mathbf{g}_k, \mathbf{x}_{k+1} - \mathbf{x}_k \rangle + \frac{1}{2}(\mathbf{x}_{k+1} - \mathbf{x})^T \mathbf{H}_k(\mathbf{x}_{k+1} - \mathbf{x}_k) + \frac{H_{Lip}}{6}\|\mathbf{x}_{k+1} - \mathbf{x}_k\|_2^3 \\
&\leq f_k + \langle \mathbf{g}_k, \mathbf{x}_{k+1} - \mathbf{x}_k \rangle + \frac{1}{2}(\mathbf{x}_{k+1} - \mathbf{x})^T \mathbf{H}_k(\mathbf{x}_{k+1} - \mathbf{x}_k) + \frac{\tilde{H}}{6}\|\mathbf{x}_{k+1} - \mathbf{x}_k\|_2^3 \\
&= f_k + \frac{2\psi_k}{\tilde{H}}\langle \mathbf{g}_k, \widehat{\mathbf{d}}_k \rangle + \frac{2\psi_k^2}{\tilde{H}^2}(\widehat{\mathbf{d}}_k)^T \mathbf{H}_k(\widehat{\mathbf{d}}_k) + \frac{4\psi_k^3}{3\tilde{H}^2}\|\widehat{\mathbf{d}}_k\|_2^3 \\
&\leq f_k - \frac{2\psi_k^3}{\tilde{H}^2} + \frac{4\psi_k^3}{3\tilde{H}^2} \\
&= f_k - \frac{2\psi_k^3}{3\tilde{H}^2}. \tag{70}
\end{aligned}$$

Now if a first-order stationary point was reached at $k-1$, i.e. $k-1 \in \mathcal{V}$, it directly follows from (70) that

$$f_{k-1} - f_{k+1} \geq f_k - f_{k+1} \geq \frac{2\psi_k^3}{3\tilde{H}^2}, \quad \forall\, k \in \mathcal{V},\, k-1 \in \mathcal{V}. \tag{71}$$

Otherwise, Step 5 of Algorithm 3 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 (70) implies that

$$f_{k-1} - f_{k+1} \geq f_k - f_{k+1} \geq \frac{2\psi_k^3}{3\tilde{H}^2}, \quad \forall\, k \in \mathcal{V}, \text{First-Order-LC-TRACE called at } k-1. \tag{72}$$

Combining (71) and (72), we get

$$f_{k-1} - f_{k+1} \geq f_k - f_{k+1} \geq \frac{2\psi_k^3}{3\tilde{H}^2}, \quad \forall\, k \in \mathcal{V}. \tag{73}$$

We next show a lower bound on the reduction of the objective value if $k \in \mathcal{K}_\epsilon$. By Lemma 29, we have

$$f_{k-1} - f_{k+1} \geq f_{k-1} - f_k \geq \frac{\rho \mathcal{X}_k^{3/2}}{(H_{Lip} + \sigma_{max})^{3/2}}, \tag{74}$$

where the first inequality again holds by the monotonicity of First-Order-LC-TRACE and (70). Combining (73) and (74), 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

$$|\mathcal{K}_\epsilon \cup \mathcal{V}| \leq \frac{2(f_0 - f_{min})}{\max\left\{ \dfrac{\rho \mathcal{X}_k^{3/2}}{(H_{Lip} + \sigma_{max})^{3/2}}, \dfrac{2\psi_k^3}{3\tilde{H}^2} \right\}} \tag{75}$$

Let $\mathcal{H}(\epsilon_g, \epsilon_H) \triangleq \{k \mid \mathcal{X}_k > \epsilon_g,\ \psi_k > \epsilon_H\}$. Using Lemmas 31, 33 and 34, the number of sub-problem 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.

$\square$