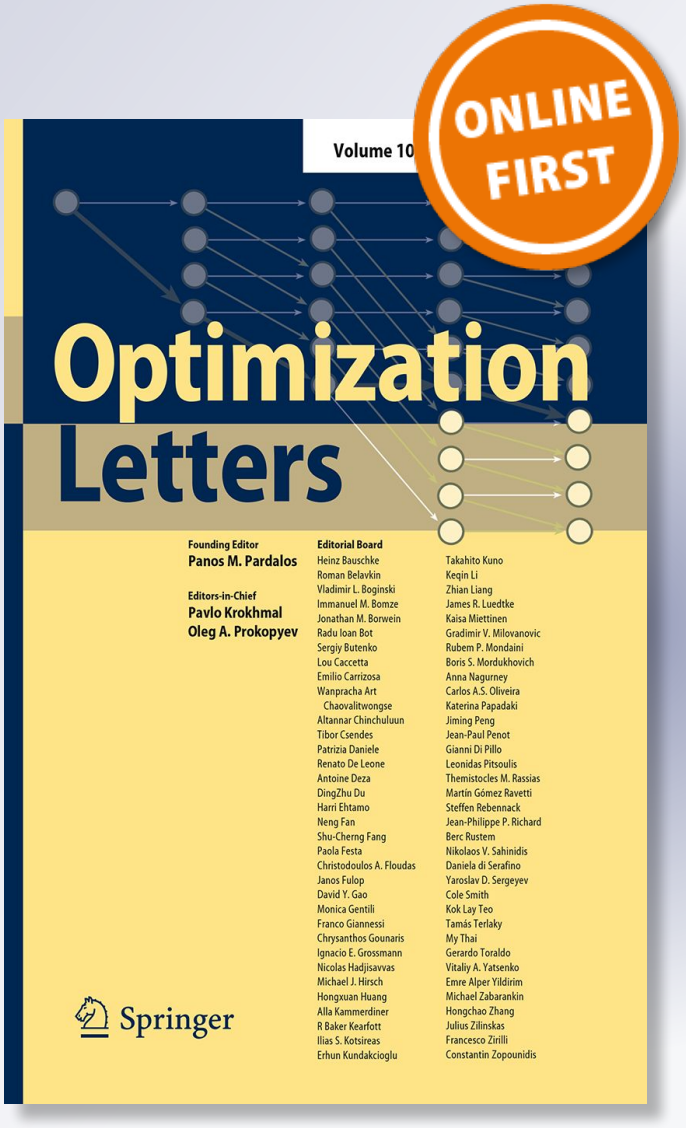
Concise complexity analyses for trust region methods

Frank E. Curtis, Zachary Lubberts & Daniel P. Robinson

Optimization Letters

ISSN 1862-4472

Optim Lett
DOI 10.1007/s11590-018-1286-2



Springer

Optimization Letters
https://doi.org/10.1007/s11590-018-1286-2

ORIGINAL PAPER

Concise complexity analyses for trust region methods

Frank E. Curtis[1] · **Zachary Lubberts**[2] · **Daniel P. Robinson**[2]

Received: 21 February 2018 / Accepted: 14 June 2018

Abstract
Concise complexity analyses are presented for simple trust region algorithms for solving unconstrained optimization problems. In contrast to a traditional trust region algorithm, the algorithms considered in this paper require certain control over the choice of trust region radius after any successful iteration. The analyses highlight the essential algorithm components required to obtain certain complexity bounds. In addition, a new update strategy for the trust region radius is proposed that offers a second-order complexity bound.

Keywords Unconstrained optimization · Nonlinear optimization · Nonconvex optimization · Trust region methods · Global convergence · Worst-case iteration complexity · Worst-case evaluation complexity

1 Introduction

We analyze a trust region framework for solving the smooth optimization problem

$$\min_{x\in\mathbb{R}^n} f(x), \tag{1.1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$. Since trust region methods have been extensively studied and analyzed, it is important that we clarify the contributions of this paper. To this end, in Sect. 1.1 we give the notation and assumptions on the objective function used throughout. Then, in Sect. 1.2, we highlight our contributions.

✉ Daniel P. Robinson
daniel.p.robinson@gmail.com

Frank E. Curtis
frank.e.curtis@gmail.com

Zachary Lubberts
zlubber1@jhu.edu

[1] Department of Industrial and Systems Engineering, Lehigh University, Bethlehem, PA, USA

[2] Department of Applied Mathematics and Statistics, Johns Hopkins University, Baltimore, MD, USA

Published online: 21 June 2018

1.1 Notation and assumption about the objective function

We use $\mathbb{R}$ to denote the set of real numbers, $\mathbb{R}_+$ (respectively $\mathbb{R}_{++}$) to denote the set of nonnegative (respectively positive) real numbers, $\mathbb{R}^n$ to denote the set of n-dimensional real vectors, and $\mathbb{R}^{m\times n}$ to denote the set of m-by-n-dimensional real matrices. The set of natural numbers is denoted as $\mathbb{N} := \{0, 1, 2, \ldots\}$.

We denote $g := \nabla f : \mathbb{R}^n \to \mathbb{R}$ and $H := \nabla^2 f : \mathbb{R}^n \to \mathbb{R}^{n\times n}$. For all $k \in \mathbb{N}$, we let x_k denote the kth iteration computed by the trust region method. For brevity, we append $k \in \mathbb{N}$ as a subscript to a function to denote its value at the kth iterate x_k, e.g., $f_k = f(x_k)$, $g_k = g(x_k)$, and $H_k := H(x_k)$. Given H_k, we let $\lambda_k := \lambda_k(H_k)$ denote the leftmost eigenvalue of H_k, and $(\lambda_k)_- := \min\{0, \lambda_k\}$, i.e., $(\lambda_k)_-$ is the negative part of λ_k. Finally, for $v \in \mathbb{R}^n$, we use $\|v\|$ to denote the two-norm of v.

The following assumption on the objective function is made throughout.

Assumption 1.1 The function $f : \mathbb{R}^n \to \mathbb{R}$ is twice continuously differentiable with Hessian function H being Lipschitz continuous, i.e., there exists a constant $L \in \mathbb{R}_{++}$ such that $\|H(x) - H(y)\| \leq L\|x - y\|$ for all $(x, y) \in \mathbb{R}^n \times \mathbb{R}^n$. Also, the function f is bounded below, i.e., there exists $f_{\text{inf}} \in \mathbb{R}$ such that $f(x) \geq f_{\text{inf}}$ for all $x \in \mathbb{R}^n$.

Such an assumption is common among second-order complexity analyses [1,3,8].

1.2 Contributions

This paper proves first- and second-order complexity results for a few trust region methods. Given $\epsilon_g \in \mathbb{R}_{++}$, a first-order complexity result is a bound on the number of iterations until $\|g_k\| \leq \epsilon_g$. Also given $\epsilon_H \in \mathbb{R}_{++}$, a second-order complexity result is a bound on the number of iterations until $\|g_k\| \leq \epsilon_g$ and $(\lambda_k)_- \geq -\epsilon_H$.

In Sect. 2.3, we show that explicitly connecting the kth trust region radius to $\|g_k\|$ allows for a concise analysis that proves a first-order complexity of $\mathcal{O}(\epsilon_g^{-2})$. This method turns out to be a special case of the framework considered in [5,6,10]. Our analysis is simpler, however, since we do not consider such a general framework.

In Sect. 2.4, we propose a new updating strategy for the trust region radius that allows one to obtain a second-order complexity of $\mathcal{O}(\max\{\epsilon_g^{-2}, \epsilon_H^{-3}\})$. This algorithm is similar to an instance of the framework in [7], the main difference being that our choice of trust region radius depends on whether $\|g_k\|^2 \geq |(\lambda_k)_-|^3$. By contrast, the radius in [7] depends on whether $\|g_k\| \geq |(\lambda_k)_-|$. As a consequence of their choice, [7] is only able to establish a second-order complexity of $\mathcal{O}(\max\{\epsilon_g^{-3}, \epsilon_H^{-3}\})$. Due to the difference in the first-order bound ($\mathcal{O}(\epsilon_g^{-2})$ vs. $\mathcal{O}(\epsilon_g^{-3})$), our result is preferred. This kind of discrepancy between first- and second-order complexity recently motivated the work in [8]. Similar to our approach, the framework in [8] chooses between first- and second-order steps, and achieves the same complexity result. A key difference, however, is that their algorithm computes two trial steps and requires two function evaluations per iteration. This is of little concern in theory, but, in practical terms, requiring two function evaluations per iteration might be detrimental. Our method, by choosing the trust region radius before a (single) step is computed, requires only a single function evaluation per iteration.

In Sect. 2.5, we contrast the aforementioned strategies with one that offers better complexity bounds, but has some practical disadvantages.

2 A trust region algorithm

In this section, we present and analyze trust region algorithms, the framework for which is stated in Sect. 2.1. The key results needed to perform most of our complexity analyses are stated and proved in Sect. 2.2. Using these key results, we establish a first-order complexity result in Sect. 2.3 and a second-order complexity result in Sect. 2.4. Specific strategies employed to obtain these complexity results are stated in each subsection. We end by considering an instance with a fixed trust region radius in Sect. 2.5 that has advantages and disadvantages vis-à-vis these other strategies.

2.1 The algorithm

We study the trust region method stated as Algorithm 1. During the kth iteration, given a trust region radius $\delta_k \in \mathbb{R}_{++}$, the algorithm computes an approximate solution $s_k \in \mathbb{R}^n$ to the trust region subproblem

$$\min_{s\in\mathbb{R}^n} \left\{ m_k(s) := f_k + g_k^T s + \tfrac{1}{2} s^T H_k s \right\} \quad \text{s.t.} \quad \|s\| \le \delta_k. \tag{2.1}$$

To facilitate a unified first- and second-order complexity analysis, when $\lambda_k < 0$ we allow in Step 5 the radius to be set as either $\delta_k \leftarrow \gamma_k \|g_k\|$ or $\delta_k \leftarrow \gamma_k |(\lambda_k)_-|$. For this reason, it will be convenient for our analyses to refer to the index sets

$$\begin{aligned} \mathcal{K}_g &:= \{k \in \mathbb{N} : \ \delta_k \leftarrow \gamma_k \|g_k\| \text{ in either Step 5 or Step 7}\} \\ \text{and } \mathcal{K}_H &:= \{k \in \mathbb{N} : \ \delta_k \leftarrow \gamma_k |(\lambda_k)_-| \text{ in Step 5}\}. \end{aligned}$$

Note that $\lambda_k < 0$ for all $k \in \mathcal{K}_H$ due to Step 4. We define an approximate solution to (2.1) in terms of a Cauchy point that applies to our setting, i.e., a vector that ensures that the model m_k is sufficiently reduced. Specifically, if u_k denotes a unit eigenvector corresponding to λ_k scaled by ± 1 so that $g_k^T u_k \le 0$, then with

$$v_k := \begin{cases} -g_k & \text{if } k \in \mathcal{K}_g \\ u_k & \text{if } k \in \mathcal{K}_H, \end{cases} \tag{2.2}$$

the Cauchy point s_k^c is defined as

$$s_k^c := t_k v_k, \ \text{where } t_k := \operatorname*{argmin}_{t\ge 0} m_k(t v_k) \ \text{ s.t. } \ \|t v_k\| \le \delta_k. \tag{2.3}$$

We say that any s_k satisfying $m_k(s_k) \le m_k(s_k^c)$ and $\|s_k\| \le \delta_k$ is a valid choice for an approximate solution to problem (2.1).

Algorithm 1 Trust region algorithm

1: Input an initial estimate $x_0 \in \mathbb{R}^n$ of a solution to (1.1).
2: Choose parameters $(\gamma_c, \eta) \in (0, 1) \times (0, 1)$, $0 < \underline{\gamma} \leq \bar{\gamma} < \infty$, and $\gamma_0 \in [\underline{\gamma}, \bar{\gamma}]$.
3: **for** $k = 0, 1, 2, \ldots$ **do**
4: **if** $\lambda_k < 0$ **then**
5: Set either $\delta_k \leftarrow \gamma_k \|g_k\|$ or $\delta_k \leftarrow \gamma_k |(\lambda_k)_-|$.
6: **else**
7: Set $\delta_k \leftarrow \gamma_k \|g_k\|$.
8: **end if**
9: Find any trial step s_k that satisfies $\|s_k\| \leq \delta_k$ and $m_k(s_k) \leq m_k(s_k^c)$.
10: Set

$$\rho_k \leftarrow \frac{f_k - f(x_k + s_k)}{m_k(0) - m_k(s_k)}.$$

11: **if** $\rho_k \geq \eta$, **then**
12: Set $x_{k+1} \leftarrow x_k + s_k$ and choose any $\gamma_{k+1} \in [\underline{\gamma}, \bar{\gamma}]$. (▷) successful
13: **else**
14: Set $x_{k+1} \leftarrow x_k$ and $\gamma_{k+1} \leftarrow \gamma_c \gamma_k$. (▷) unsuccessful
15: **end if**
16: **end for**

The updates for setting x_{k+1} and γ_{k+1} depend on the ratio of the decrease in f to the decrease predicted by the model m_k, as denoted by ρ_k in Step 10. If ρ_k is larger than a pre-specified value $\eta \in (0, 1)$, then $x_{k+1} \leftarrow x_k + s_k$ and any value for γ_{k+1} satisfying $\gamma_{k+1} \in [\underline{\gamma}, \bar{\gamma}]$ with $0 < \underline{\gamma} \leq \bar{\gamma} < \infty$ may be used. In this event, the iteration is said to be *successful*. On the other hand, if $\rho_k < \eta$, then $x_{k+1} \leftarrow x_k$ and $\gamma_{k+1} \leftarrow \gamma_c \gamma_k$ with $\gamma_c \in (0, 1)$; the iteration is said to be *unsuccessful*. It will be helpful for the analysis to define the sets of successful and unsuccessful iterations:

$$\mathcal{S} := \{k \in \mathbb{N} : \rho_k \geq \eta\} \text{ and } \mathcal{U} := \{k \in \mathbb{N} : \rho_k < \eta\}. \tag{2.4}$$

We now turn to the key results that are needed in our analyses.

2.2 Key results needed for complexity analyses

We start by making the following assumption on the iterates.

Assumption 2.1 The Hessian function H is uniformly bounded over the sequence of iterates, i.e., for some $\kappa \in \mathbb{R}_+$ and all $k \in \mathbb{N}$, it holds that $\|H_k\| \leq \kappa$.

Note that Assumption 2.1 is implied by Assumption 1.1 any time $\{x_k\}$ is contained in a bounded set, which is a common assumption used in some analyses.

Our first result gives the decrease in m_k guaranteed by the Cauchy point.

Lemma 2.2 *For all $k \in \mathbb{N}$, the trial step s_k satisfies*

$$m_k(0) - m_k(s_k) \geq \begin{cases} \frac{1}{2} \min\left\{(1+\kappa)^{-1}, \gamma_k\right\} \|g_k\|^2 & \text{if } k \in \mathcal{K}_g, \\ \frac{1}{2} \gamma_k^2 |(\lambda_k)_-|^3 & \text{if } k \in \mathcal{K}_H. \end{cases} \tag{2.5}$$

Proof First, suppose that $k \in \mathcal{K}_g$, which implies that $v_k = -g_k$ in the definition of the Cauchy point in (2.3). In this case, it follows from the decrease guaranteed by the Cauchy point [2, Theorem 6.3.1], $\delta_k \leftarrow \gamma_k \|g_k\|$, and Assumption 2.1 that

$$m_k(0) - m_k(s_k^c) \geq \tfrac{1}{2}\|g_k\| \min\left\{\frac{\|g_k\|}{1+\|H_k\|}, \delta_k\right\} \geq \tfrac{1}{2}\min\left\{\frac{1}{1+\kappa}, \gamma_k\right\}\|g_k\|^2.$$

The result (2.5) follows from this fact and $m_k(s_k) \leq m_k(s_k^c)$, as required in Step 9.

Second, suppose that $k \in \mathcal{K}_H$, which implies that $v_k = u_k$ in the definition of the Cauchy point in (2.3); recall that u_k denotes a unit eigenvector of H_k corresponding to λ_k scaled by ± 1 so that $g_k^T u_k \leq 0$. By combining this with $k \in \mathcal{K}_H$ so that $\delta_k \leftarrow \gamma_k |(\lambda_k)_-|$ with $\lambda_k < 0$, (2.2), and (2.3) we obtain

$$\begin{aligned}
&\min_{t\geq 0} \; m_k(tu_k) \;\text{ s.t. }\; \|tu_k\|_2 \leq \delta_k = \gamma_k|(\lambda_k)_-| \\
&= \min_{t\geq 0} \; f_k + g_k^T(tu_k) + \tfrac{1}{2}(tu_k)^T H_k(tu_k) \;\text{ s.t. }\; t \leq \gamma_k|(\lambda_k)_-| \\
&= \min_{t\geq 0} \; f_k + tg_k^T u_k + \tfrac{1}{2}t^2\lambda_k \;\text{ s.t. }\; t \leq \gamma_k|(\lambda_k)_-|.
\end{aligned}$$

Since $g_k^T u_k \leq 0$ and $\lambda_k < 0$, the minimum occurs at $t_k = \gamma_k|(\lambda_k)_-|$, which combined with $m_k(x_k) \leq m_k(s_k^c)$ in Step 9 of Algorithm 1 yields

$$\begin{aligned}
m_k(s_k) \leq m_k(s_k^c) &= \min_{t\geq 0} \; m_k(tu_k) \;\text{ s.t. }\; \|tu_k\|_2 \leq \delta_k = \gamma_k|(\lambda_k)_-| \\
&= m_k(t_k u_k) = f_k - \gamma_k\lambda_k g_k^T u_k - \tfrac{1}{2}\gamma_k^2|(\lambda_k)_-|^3 \\
&\leq f_k - \tfrac{1}{2}\gamma_k^2|(\lambda_k)_-|^3 = m_k(0) - \tfrac{1}{2}\gamma_k^2|(\lambda_k)_-|^3.
\end{aligned}$$

Hence, the reduction in m_k obtained by s_k is bounded as in (2.5). □

We now bound the difference between the objective function and its model.

Lemma 2.3 *For all $k \in \mathbb{N}$, the error in the model m_k at s_k can be bounded as*

$$|f(x_k + s_k) - m_k(s_k)| \leq \begin{cases} \kappa\gamma_k^2\|g_k\|^2 & \text{if } k \in \mathcal{K}_g, \\ \frac{1}{6}L\gamma_k^3|(\lambda_k)_-|^3 & \text{if } k \in \mathcal{K}_H. \end{cases}$$

Proof If $k \in \mathcal{K}_g$, the result follows from [2, Theorem 6.4.1], Assumption 2.1, and the fact that $\delta_k \leftarrow \gamma_k\|g_k\|$ for $k \in \mathcal{K}_g$. If $k \in \mathcal{K}_H$, the result follows from Assumption 1.1, [2, Theorem 3.1.5], and the fact that $\delta_k \leftarrow \gamma_k|(\lambda_k)_-|$ for $k \in \mathcal{K}_H$. □

We can now give a uniform lower bound on $\{\gamma_k\}$ that is independent of k.

Lemma 2.4 *For all $k \in \mathbb{N}$, it holds that*

$$\gamma_k \geq \gamma_{\min} := \min\left\{\underline{\gamma}, \frac{\gamma_c}{1+\kappa}, \frac{\gamma_c(1-\eta)}{2\kappa}, \frac{3\gamma_c(1-\eta)}{L}\right\} \in (0, 1). \tag{2.6}$$

Proof For a proof by induction, we first note that $\gamma_0 \geq \underline{\gamma} \geq \gamma_{\min}$ by the choice for γ_0 in Algorithm 1, so that (2.6) holds when $k = 0$. Next, supposing that (2.6) holds for k, we proceed to show that it also holds with k replaced by $k + 1$.

Case 1: $\gamma_k > \min\{1/(1+\kappa), (1-\eta)/(2\kappa), 3(1-\eta)/L\}$. In this case, it holds that

$$\begin{aligned}\gamma_{k+1} &\geq \min\{\underline{\gamma}, \gamma_c\gamma_k\} \geq \min\left\{\underline{\gamma}, \gamma_c/(1+\kappa), \gamma_c(1-\eta)/(2\kappa), 3\gamma_c(1-\eta)/L\right\} \\ &\equiv \gamma_{\min},\end{aligned}$$

meaning that (2.6) holds with k replaced by $k + 1$ as claimed.

Case 2: $\gamma_k \leq \min\{1/(1+\kappa), (1-\eta)/(2\kappa), 3(1-\eta)/L\}$. First, suppose that $k \in \mathcal{K}_g$. It follows from the definition of ρ_k, Lemmas 2.3, and 2.2 that

$$\begin{aligned}|\rho_k - 1| &= \frac{|f(x_k + s_k) - m_k(s_k)|}{m_k(0) - m_k(s_k)} \leq \frac{2\kappa\gamma_k^2\|g_k\|^2}{\min\left\{(1+\kappa)^{-1}, \gamma_k\right\}\|g_k\|^2} \\ &= 2\kappa\gamma_k \leq 1 - \eta,\end{aligned}$$

which implies that $\rho_k \geq \eta$, i.e., that $k \in \mathcal{S}$. Second, suppose that $k \in \mathcal{K}_H$. It follows from Lemmas 2.3 and 2.2 that

$$|\rho_k - 1| = \frac{|f(x_k + s_k) - m_k(s_k)|}{m_k(0) - m_k(s_k)} \leq \frac{L\gamma_k^3|(\lambda_k)_-|^3}{3\gamma_k^2|(\lambda_k)_-|^3} = \frac{L\gamma_k}{3} \leq 1 - \eta,$$

which implies $\rho_k \geq \eta$, i.e., that $k \in \mathcal{S}$. In either subcase, it follows that $k \in \mathcal{S}$. Using $k \in \mathcal{S}$, we have from Algorithm 1 that $\gamma_{k+1} \geq \underline{\gamma} \geq \gamma_{\min}$, so that (2.6) holds with k replaced by $k + 1$ as claimed.

The result follows since we proved the inductive step in each case. □

The next result gives a refined bound on the decrease in the model m_k for all $k \in \mathbb{N}$, as well as gives a bound on the decrease in f when $k \in \mathcal{S}$.

Lemma 2.5 *For all $k \in \mathbb{N}$, the trial step s_k satisfies*

$$m_k(0) - m_k(s_k) \geq \begin{cases} \frac{1}{2}\gamma_{\min}\|g_k\|^2 & \text{if } k \in \mathcal{K}_g, \\ \frac{1}{2}\gamma_{\min}^2|(\lambda_k)_-|^3 & \text{if } k \in \mathcal{K}_H. \end{cases} \tag{2.7}$$

In addition, with $\kappa_{\min} := \frac{1}{2}\eta\gamma_{\min}^2$, we have for all $k \in \mathbb{N}$ that

$$f_k - f_{k+1} \geq \begin{cases} \kappa_{\min}\|g_k\|^2 & \text{if } k \in \mathcal{K}_g \cap \mathcal{S}, \\ \kappa_{\min}|(\lambda_k)_-|^3 & \text{if } k \in \mathcal{K}_H \cap \mathcal{S}. \end{cases} \tag{2.8}$$

Proof The lower bound (2.7) follows from Lemmas 2.2 and 2.4, after observing that $\gamma_{\min} \leq \gamma_c(1+\kappa)^{-1} \leq (1+\kappa)^{-1}$. The result (2.8) follows from (2.7) and the fact that $\rho_k \geq \eta$ when $k \in \mathcal{S}$. □

We now show that the maximum number of consecutive unsuccessful iterations can be bounded and is independent of how δ_k is chosen in Step 5 or Step 7.

Lemma 2.6 *The number of consecutive iterations in $\mathcal{U}$ is at most $\lceil \log_{\gamma_c}(\frac{\gamma_{\min}}{\bar{\gamma}}) \rceil > 0$.*

Proof The update strategy for $\{\gamma_k\}$ ensures that $\gamma_k \leq \bar{\gamma}$ for all $k \in \mathbb{N}$. Also, it follows from Lemma 2.4 that $\gamma_k \geq \gamma_{\min}$ for all $k \in \mathbb{N}$. Since the update $\gamma_{k+1} \leftarrow \gamma_c \gamma_k$ is used when $k \in \mathcal{U}$, we must conclude that the maximum number of consecutive iterations in $\mathcal{U}$ can be no larger than $\lceil \log_{\gamma_c}(\gamma_{\min}/\bar{\gamma}) \rceil > 0$, as claimed. □

2.3 A strategy with a concise first-order complexity analysis

Our aim is to provide a bound on the number of iterations until the norm of the gradient falls below a threshold value, say $\epsilon_g \in \mathbb{R}_{++}$. We define the sets

$$\mathcal{S}_1(\epsilon_g) := \{k \in \mathcal{S} : \|g_k\| > \epsilon_g\} \text{ and } \mathcal{K}_1(\epsilon_g) := \{k \in \mathbb{N} : \|g_k\| > \epsilon_g\}.$$

Also, since we are currently interested in approximate *first-order* stationarity, it is reasonable to use the following trust region radius update strategy.

Update 2.7 *For any $k \in \mathbb{N}$ such that Step 5 is reached, we set $\delta_k \leftarrow \gamma_k \|g_k\|$.*

Combining Update 2.7 with Step 7 shows that $\delta_k \leftarrow \gamma_k \|g_k\|$ for all $k \in \mathbb{N}$ so that $\mathcal{K}_g = \mathbb{N}$ and $\mathcal{K}_H = \emptyset$. The results of this section assume that Update 2.7 is used.

We start by proving an upper bound on the size of the index set $\mathcal{S}_1(\epsilon_g)$.

Lemma 2.8 *For any $\epsilon_g \in \mathbb{R}_{++}$, the size of $\mathcal{S}_1(\epsilon_g)$ satisfies*

$$|\mathcal{S}_1(\epsilon_g)| \leq \left\lfloor \left(\frac{f_0 - f_{\inf}}{\kappa_{\min}} \right) \epsilon_g^{-2} \right\rfloor. \tag{2.9}$$

Proof We start by noting that $\mathcal{S}_1(\epsilon_g) \subseteq \mathbb{N} = \mathcal{K}_g$. Combining this with Assumption 1.1, monotonicity of $\{f_k\}$, (2.8), and the definition of $\mathcal{S}_1(\epsilon_g)$ gives

$$f_0 - f_{\inf} \geq \sum_{k \in \mathcal{S}_1(\epsilon_g)} \left(f_k - f_{k+1} \right) \geq \kappa_{\min} \sum_{k \in \mathcal{S}_1(\epsilon_g)} \|g_k\|^2 \geq \kappa_{\min} \epsilon_g^2 |\mathcal{S}_1(\epsilon_g)|.$$

The bound in (2.9) follows from this inequality. □

We obtain the complexity result by combining the last with Lemma 2.6.

Theorem 2.9 *For any $\epsilon_g \in \mathbb{R}_{++}$, the size of $\mathcal{K}_1(\epsilon_g)$ satisfies*

$$|\mathcal{K}_1(\epsilon_g)| \leq \left\lceil \log_{\gamma_c} \left(\frac{\gamma_{\min}}{\bar{\gamma}} \right) \right\rceil \left\lfloor \left(\frac{f_0 - f_{\inf}}{\kappa_{\min}} \right) \epsilon_g^{-2} \right\rfloor = \mathcal{O}(\epsilon_g^{-2}).$$

Proof The result follows by combining Lemmas 2.8 and 2.6. □

We conclude this section by discussing how the trust region radius update considered in this section compares to a traditional strategy. In fact, the strategies are the same for unsuccessful iterations since they both set $\delta_{k+1} \leftarrow \gamma_c \delta_k$. However, if k is a successful iteration, then a traditional strategy sets $\delta_{k+1}^{trad} \leftarrow \max\{\gamma_e \|s_k\|, \delta_k\}$ for some $\gamma_e \geq 1$. This update is also allowed by Update 2.7 as long as

$$\delta_{k+1}^{trad} \leftarrow \max\{\gamma_e \|s_k\|, \delta_k\} \equiv \max\{\gamma_e \|s_k\|, \gamma_k \|g_k\|\} \in [\underline{\gamma} \|g_{k+1}\|, \bar{\gamma} \|g_{k+1}\|].$$

In particular, this means that the traditional update is *not* allowed in two scenarios. The first is when $\max\{\gamma_e \|s_k\|, \gamma_k \|g_k\|\} < \underline{\gamma} \|g_{k+1}\|$. Since $\gamma_e \geq 1$ and $\underline{\gamma}$ is intended to serve as a lower-bound safeguard (e.g., a typical value might be 10^{-8} or smaller), this scenario indicates that the accepted step s_k and gradient g_k are very small in norm compared to the new gradient g_{k+1}. But since $\|g_{k+1}\|$ being large means that the next reduction in f could also be large with a relatively large step, we argue that Update 2.7 makes sense. The second scenario in which the traditional update is not allowed is when $\max\{\gamma_e \|s_k\|, \gamma_k \|g_k\|\} > \bar{\gamma} \|g_{k+1}\|$. Since γ_e is moderate in size (e.g., a typical value is $\gamma_e = 2$) and $\bar{\gamma}$ is intended to serve as an upper-bound safeguard (e.g., a typical value might be 10^8 or larger), this scenario indicates that the previous radius is significantly larger than the size of the gradient at the new iterate x_{k+1}. Since an additional increase in the trust region radius does not seem warranted, we again believe that Update 2.7 makes sense.

2.4 A strategy with a concise second-order complexity analysis

In this section, our aim is to provide an upper bound on the maximum number of iterations until, given $(\epsilon_g, \epsilon_H) \in \mathbb{R}_{++} \times \mathbb{R}_{++}$, an iterate x_k satisfies $\|g_k\| \leq \epsilon_g$ and $\lambda_k \geq -\epsilon_H$. For this reason, it will be convenient to define the sets

$$\mathcal{S}_2(\epsilon_g, \epsilon_H) := \{k \in \mathcal{S} : \|g_k\| > \epsilon_g \text{ or } |(\lambda_k)_-| > \epsilon_H\}$$
$$\text{and } \mathcal{K}_2(\epsilon_g, \epsilon_H) := \{k \in \mathbb{N} : \|g_k\| > \epsilon_g \text{ or } |(\lambda_k)_-| > \epsilon_H\}.$$

Since we are now interested in approximate second-order optimality, and motivated by the decrease in f guaranteed by (2.8) for successful iterations, in this section we adopt the following trust region radius update strategy.

Update 2.10 *For any $k \in \mathbb{N}$ such that Step 5 is reached, in which case $\lambda_k < 0$, set*

$$\delta_k \leftarrow \begin{cases} \gamma_k \|g_k\| & \text{if } \|g_k\|^2 \geq |(\lambda_k)_-|^3, \\ \gamma_k |(\lambda_k)_-| & \text{if } \|g_k\|^2 < |(\lambda_k)_-|^3. \end{cases}$$

The results of this section assume that Update 2.10 is used.

We first prove an upper bound on the size of the index set $\mathcal{S}_2(\epsilon_g, \epsilon_H)$.

Lemma 2.11 *For any* $(\epsilon_g, \epsilon_H) \in \mathbb{R}_{++} \times \mathbb{R}_{++}$, *the size of* $\mathcal{S}_2(\epsilon_g, \epsilon_H)$ *satisfies*

$$|\mathcal{S}_2(\epsilon_g, \epsilon_H)| \leq \left\lfloor \left(\frac{f_0 - f_{\inf}}{\kappa_{\min}} \right) \max \left\{ \epsilon_g^{-2}, \epsilon_H^{-3} \right\} \right\rfloor. \tag{2.10}$$

Proof Combining Update 2.10 with Step 7, it follows that $k \in \mathcal{K}_g$ if and only if $\|g_k\|^2 \geq |(\lambda_k)_-|^3$ while $k \in \mathcal{K}_H$ if and only if $|(\lambda_k)_-|^3 > \|g_k\|^2$. Consider $k \in \mathcal{S}_2(\epsilon_g, \epsilon_H) \cap \mathcal{K}_H$. By (2.8) and $\mathcal{K} \subseteq \mathcal{K}_H$,

$$f_k - f_{k+1} \geq \kappa_{\min} |(\lambda_k)_-|^3 \geq \kappa_{\min} \min\{\epsilon_g^2, \epsilon_H^3\}. \tag{2.11}$$

Now consider $k \in \mathcal{S}_2(\epsilon_g, \epsilon_H) \cap \mathcal{K}_g$. By (2.8) and $\mathcal{K} \subseteq \mathcal{K}_g$,

$$f_k - f_{k+1} \geq \kappa_{\min} \|g_k\|^2 \geq \kappa_{\min} \min\{\epsilon_g^2, \epsilon_H^3\}. \tag{2.12}$$

Combining (2.11), (2.12), Assumption 1.1, and monotonicity of $\{f_k\}$, one finds

$$f_0 - f_{\inf} \geq \sum_{k \in \mathcal{S}_2(\epsilon_g, \epsilon_H)} (f_k - f_{k+1}) \geq \kappa_{\min} \min\{\epsilon_g^2, \epsilon_H^3\} |\mathcal{S}_2(\epsilon_g, \epsilon_H)|,$$

which, after rearrangement, leads to (2.10). □

Lemmas 2.11 and 2.6 lead to our second-order complexity result.

Theorem 2.12 *For any* $(\epsilon_g, \epsilon_H) \in \mathbb{R}_{++} \times \mathbb{R}_{++}$, *the size of* $|\mathcal{K}_2(\epsilon_g, \epsilon_H)|$ *satisfies*

$$\begin{aligned} |\mathcal{K}_2(\epsilon_g, \epsilon_H)| &\leq \left\lceil \log_{\gamma_c} \left(\frac{\gamma_{\min}}{\bar{\gamma}} \right) \right\rceil \left\lfloor \left(\frac{f_0 - f_{\inf}}{\kappa_{\min}} \right) \max \left\{ \epsilon_g^{-2}, \epsilon_H^{-3} \right\} \right\rfloor \\ &= \mathcal{O}\left(\max \left\{ \epsilon_g^{-2}, \epsilon_H^{-3} \right\} \right). \end{aligned}$$

2.5 A strategy with a fixed trust region radius

The strategy in Sect. 2.3 offers a complexity of $\mathcal{O}(\epsilon_g^{-2})$ for driving the norm of the gradient below $\epsilon_g \in \mathbb{R}_{++}$. This is consistent with the strategy in Sect. 2.4 when $\mathcal{K}_g = \mathbb{N}$ or $\epsilon_H = \epsilon_g^{2/3}$. However, certain methods offer a complexity of $\mathcal{O}(\epsilon_g^{-3/2})$; e.g., see [1]. Is it possible to design a trust region strategy that leads to this complexity, and what are its advantages and disadvantages compared to those in Sects. 2.3 and 2.4 that do not offer the same complexity? This is the subject of this subsection.

A trust region method with an $\mathcal{O}(\epsilon_g^{-3/2})$ complexity for achieving approximate first-order stationarity was proposed and analyzed in [3]. This method can be seen, along with that in [1], as a special case of the general framework in [4] for achieving this order complexity. One can also derive a trust region method with a fixed trust region radius that, with a concise analysis, leads to a $\mathcal{O}(\epsilon_g^{-3/2})$ complexity. Let us present this analysis now, which follows the lecture notes of Yinyu Ye.[1]

[1] http://web.stanford.edu/class/msande311/lecture13.pdf.

Under Assumption 1.1 and with $\beta := \frac{1}{2}L$, it follows from [2, Theorem 3.1.6] and [1, Equation (1.1)] that, for all $(x, s) \in \mathbb{R}^n \times \mathbb{R}^n$,

$$\|g(x+s) - g(x) - H(x)s\| \leq \beta\|s\|^2 \tag{2.13a}$$

$$\text{and } f(x+s) - f(x) \leq g(x)^T s + \tfrac{1}{2}s^T H(x)s + \tfrac{1}{3}\beta\|s\|^3. \tag{2.13b}$$

The algorithm that we consider here is one that, for all $k \in \mathbb{N}$, sets $\delta_k \leftarrow \sqrt{\epsilon}/\beta$ and computes (s_k, ξ_k) as a primal-dual solution of (2.1) satisfying

$$g_k + (H_k + \xi_k I)s_k = 0, \tag{2.14a}$$

$$H_k + \xi_k I \succeq 0, \tag{2.14b}$$

$$\xi_k \geq 0,\ \delta_k - \|s_k\| \geq 0,\ \text{and } \xi_k(\delta_k - \|s_k\|) = 0. \tag{2.14c}$$

This algorithm, unlike traditional methods, accepts all computed steps.

There are two situations to consider.

Case 1: If $\xi_k \leq \sqrt{\epsilon}$, then (2.13a) and (2.14a) imply

$$\begin{aligned}\|g_{k+1}\| &\leq \|g_{k+1} - (g_k + H_k s_k)\| + \|g_k + H_k s_k\| \\ &\leq \beta\|s_k\|^2 + \xi_k\|s_k\| \leq \beta\delta_k^2 + \xi_k\delta_k = \frac{\epsilon}{\beta} + \frac{\xi_k\sqrt{\epsilon}}{\beta} \leq \frac{2\epsilon}{\beta}.\end{aligned}$$

Next, combining [9, page 370] and Assumption 1.1 we have $|\lambda_k - \lambda_{k+1}| \leq \|H_k - H_{k+1}\| \leq L\|s_k\|$, which with (2.14b) implies that $\lambda_{k+1} \geq \lambda_k - L\|s_k\| \geq -\xi_k - L\delta_k \geq -\sqrt{\epsilon} - L\sqrt{\epsilon}/\beta = -3\sqrt{\epsilon}$. Overall, in iteration $k+1$, one finds

$$\|g_{k+1}\| \leq \frac{2\epsilon}{\beta} \quad \text{and} \quad \lambda_{k+1} \geq -3\sqrt{\epsilon}.$$

Case 2: If $\xi_k > \sqrt{\epsilon}$, then we know from (2.14c) that $\|s_k\| = \delta_k$, which may then be combined with (2.14a) and (2.14b) to conclude that

$$g_k^T s_k + \tfrac{1}{2}s_k^T H_k s_k = -\tfrac{1}{2}s_k^T(H_k + \xi_k I)s_k - \tfrac{1}{2}\xi_k\|s_k\|^2 \leq -\tfrac{1}{2}\xi_k\|s_k\|^2 = -\tfrac{1}{2}\xi_k\delta_k^2.$$

This may, in turn, be used with (2.13b) to obtain

$$f_{k+1} - f_k \leq -\tfrac{1}{2}\xi_k\delta_k^2 + \tfrac{1}{3}\beta\delta_k^3 = -\frac{\xi_k\epsilon}{2\beta^2} + \frac{\epsilon^{3/2}}{3\beta^2} \leq -\frac{\epsilon^{3/2}}{6\beta^2}.$$

Letting $\mathcal{K} := \{k \in \mathbb{N} : \xi_k > \sqrt{\epsilon}\}$, we have $f_0 - f_{\inf} \geq \sum_{k\in\mathcal{K}}(f_k - f_{k+1}) \geq \frac{\epsilon^{3/2}}{6\beta^2}|\mathcal{K}|$, which means that $|\mathcal{K}| = \mathcal{O}(\epsilon^{-3/2})$.

Overall, we may conclude from these cases that the number of iterations until $\|g_k\| \leq \epsilon$ and $\lambda_k \geq -\sqrt{\epsilon}$ is at most $\mathcal{O}(\epsilon^{-3/2})$. An advantage of the strategy employed here is that one might be able to extend this strategy and analysis to situations in which

g_k and H_k cannot be computed exactly in each iteration. However, when g_k and H_k are computable, there are costs to achieving this improved complexity:

1. Knowledge of $\beta = \frac{1}{2}L$ is required, which is often unknown in practice.
2. Exact subproblem solutions are needed. This restriction might be relaxed using ideas such as in [4], but one cannot simply employ Cauchy steps such as those allowed in the strategies in Sects. 2.3 and 2.4.
3. There is a dependency on the choice of ϵ, meaning that the desired accuracy needs to be chosen in advance and early iterations may behave differently depending on the final accuracy desired. In addition, having the trust region radii depend on ϵ, which is likely to be small, means that the algorithm is likely to take very small steps throughout the optimization process. This would likely lead to very poor behavior in practice compared to the strategies in Sects. 2.3 and 2.4, which are much less conservative.

3 Conclusion

We have presented concise complexity analyses of trust region algorithms that tie the trust-region radius to first- and second-order measures of optimality. It is unclear how to establish similar complexity results when a traditional trust region radius update is used. For the first-order case, the reason is that, following a successful iteration, it is possible that $\|g_{k+1}\|$ may become too large relative to the trust region radius δ_{k+1}; traditional updating schemes do not appropriately handle this possibility, whereas Update 2.7 does. For the second-order case, it is similarly possible that $\max\{\|g_{k+1}\|, |(\lambda_{k+1})_-|\}$ may become too large relative to δ_{k+1}; traditional updating schemes do not account for this, but Update 2.10 does.

It follows from both Theorems 2.9 and 2.12 that $\lim_{k\to\infty} \|g_k\| = 0$. This analysis contrasts that of trust region methods that use a traditional radius update strategy, whereby first a liminf result is proved, which is then used to establish the limit result. In the case of Theorem 2.9, in which case Update 2.7 is used, this can be explained by noting that the decrease in the objective function during *all* successful iterations is proportional to $\|g_k\|^2$ (cf. Proof of Lemma 2.8), which is not always true when a traditional radius update is used. In the case of Theorem 2.12, in which case Update 2.10 is used, this can be explained by noting that the decrease in the objective function during *all* successful iterations is proportional to $\max\{\|g_k\|^2, |(\lambda_k)_-|^3\}$ (cf. Proof of Theorem 2.11), which is not always true when a traditional radius update is used.

References

1. Cartis, C., Gould, N.I.M., Toint, Ph.L.: Adaptive cubic regularisation methods for unconstrained optimization. Part II: worst-case function- and derivative-evaluation complexity. Math. Program. **130**, 295–319 (2011)
2. Conn, A.R., Gould, N.I.M., Toint, Ph.L.: Trust-Region Methods. Society for Industrial and Applied Mathematics (SIAM), Philadelphia (2000)
3. Curtis, F.E., Robinson, D.P., Samadi, M.: A trust region algorithm with a worst-case iteration complexity of $\mathcal{O}(\epsilon^{-3/2})$ for nonconvex optimization. Math. Program. **162**(1), 1–32 (2017)

4. Curtis, F.E., Robinson, D.P., Samadi, M.: An Inexact Regularized Newton Framework with a Worst-Case Iteration Complexity of $\mathcal{O}(\epsilon^{-3/2})$ for Nonconvex Optimization. Technical Report 17T-011, COR@L Laboratory, Department of ISE, Lehigh University (2017)
5. Fan, J., Yuan, Y.: A new trust region algorithm with trust region radius converging to zero. In: Li, D. (ed.) 5th International Conference on Optimization: Techniques and Applications (ICOTA 2001, Hong Kong), Proceedings of the 5th International Conference on Optimization: Techniques and Applications, pp. 786–794 (2001)
6. Grapiglia, G.N., Yuan, J., Yuan, Y-x.: On the convergence and worst-case complexity of trust-region and regularization methods for unconstrained optimization. Math. Program. **152**(1–2), 491–520 (2015)
7. Grapiglia, G.N., Yuan, J., Yuan, Y-x.: Nonlinear stepsize control algorithms: complexity bounds for first-and second-order optimality. J. Optim. Theory Appl. **171**(3), 980–997 (2016)
8. Gratton, S., Royer, C.W., Vicente, L.N.: A Decoupled First/Second-Order Steps Technique for Nonconvex Nonlinear Unconstrained Optimization with Improved Complexity Bounds. Technical report, TR 17-21. Department of Mathematics, University of Coimbra, Portugal (2017)
9. Horn, R.A., Johnson, C.R.: Matrix Analysis. Cambridge University Press, Cambridge (1990)
10. Toint, Ph.L.: Nonlinear stepsize control, trust regions and regularizations for unconstrained optimization. Optim. Methods Softw. **28**(1), 82–95 (2013)