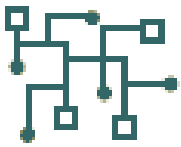

ISE

Industrial and Systems Engineering

How to Characterize the Worst-Case Performance of Algorithms for Nonconvex Optimization

Frank E. Curtis

Department of Industrial and Systems Engineering, Lehigh University

Daniel P. Robinson

Department of Applied Mathematics and Statistics, Johns Hopkins University

COR@L Technical Report 18T-003

LEHIGH UNIVERSITY

COR@L

COMPUTATIONAL OPTIMIZATION RESEARCH AT LEHIGH

How to Characterize the Worst-Case Performance of Algorithms for Nonconvex Optimization

Frank E. Curtis*[1] and Daniel P. Robinson†[2]

[1]Department of Industrial and Systems Engineering, Lehigh University
[2]Department of Applied Mathematics and Statistics, Johns Hopkins University

February 3, 2018

Abstract

A proposal is presented for how to characterize the worst-case performance of algorithms for solving smooth nonconvex optimization problems. Contemporary analyses characterize worst-case performance by providing, under certain broad assumptions on the objective function, an upper bound on the number of iterations (or function or derivative evaluations) required until a pth-order stationarity condition is approximately satisfied. However, this arguably leads to characterizations based on anomalous objective functions rather than on ones that are typically encountered in practice. By contrast, the proposal in this paper recommends to characterize worst-case performance over distinct regions of the search space. These regions are defined according to a partitioning of the search space based on whether or not regularized pth-order derivative models of the objective function predict a decrease proportional to an objective value error. This strategy can be adopted to characterize the behavior of algorithms when minimizing different classes of objective functions.

1 Introduction

Users of optimization algorithms often choose to employ one algorithm instead of another based on its theoretical properties. One such property of broad interest is worst-case complexity, wherein one measures the resources that an algorithm will require, in the worst case, to (approximately) solve a given problem. In the context of convex optimization [23], such worst-case complexity has for many years been stated in terms of an upper bound on the number of iterations (or function or derivative evaluations) required until either

- the distance between an iterate and an element of the set of minimizers is less than $\epsilon_x \in (0, \infty)$, or
- the difference between an iterate's objective value and the optimal objective value is less than $\epsilon_f \in (0, \infty)$.

In the context of nonconvex optimization, a similar strategy has been adopted. However, since one generally cannot guarantee that a method for solving nonconvex optimization problems will produce iterates that converge to a global minimizer, or at least have corresponding objective values that converge to the global minimum, the common approach has been to determine a worst-case upper bound on the number of iterations (or function or derivative evaluations) until a pth-order stationarity measure is satisfied with error below $\epsilon_p \in (0, \infty)$. For example, in a body of literature that has been growing in recent years (e.g., see [1, 2, 6, 8, 11, 12, 13, 14, 17, 24, 25]), the main measure of interest has been the number of iterations required until an

*E-mail: frank.e.curtis@gmail.com
†E-mail: daniel.p.robinson@gmail.com

algorithm is guaranteed to produce an iterate at which the norm of the gradient of the objective function—a first-order stationarity measure—is below $\epsilon_1 \in (0, \infty)$.

Unfortunately, when it comes to minimizing nonconvex objectives satisfying broad assumptions—such as Lipschitz continuity of some low-order derivatives—these types of worst-case complexity guarantees are forced to take into account exceptional objectives on which algorithms experience behavior that might be considered atypical. For example, in [4, 7], the authors show that the worst-case complexity guarantees for various well-known methods are tight due to certain objective functions that one might argue are pedogogical and distinct from those encountered in regular practice.

One might overcome the resulting discrepancy between theory and practice in various ways. Some argue that it would be ideal to be able to characterize *average case* behavior of an algorithm rather than worst case, such as has been studied for the simplex method for linear optimization; see [3, 26, 27]. However, it seems difficult to set forth a useful, valid, and widely accepted definition of an average case when minimizing nonconvex objectives. Alternatively, one might consider separately analyzing the behavior of algorithms when they are employed to minimize functions in a particular class of interest; e.g., a common class of interest is that of strongly convex functions with Lipschitz continuous first-order derivatives. This was a strategy employed in [24] in which various interesting results for a cubic regularization method—e.g., applied to star-convex and gradient-dominated functions—are presented. However, this approach to worst-case performance guarantees naturally limits itself to certain classes of objective functions. It also allows for the possibility that analyses might be performed differently for different classes, which fails to establish a standard for how to compare methods across multiple problem classes at the same time.

The purpose of this paper is to outline a strategy for characterizing the worst-case performance of algorithms for solving nonconvex optimization problems. In particular, in order to offer a characterization of algorithms both (i) within contexts seen in typical practice and (ii) without limiting attention to specific problem classes, we propose that an algorithm should be characterized based on its behavior over specific problem-dependent regions of the search space. We define these regions based on whether or not the reduction in a regularized pth-order derivative model of the objective is proportional to an objective value error. By partitioning the search space according to the lowest order model that offers such a proportional reduction, one can provide a useful characterization of the worst-case behavior of an algorithm.

We believe that there are various benefits of our proposed strategy. For one thing, it leads to a characterization of algorithms wherein each measure is defined with respect to a reference reduction in the objective. This naturally ties the characterization to what could be expected from an algorithm that employs derivative information to compute search directions, rather than on what can be proved for exceptional example functions. Another benefit is that it facilitates the characterization of the behavior of an algorithm in certain important cases. For example, questions that are often asked are: When one takes an algorithm for nonconvex optimization and employs it to minimize a strongly convex function, what is its worst-case behavior? Is it the same as when it is employed to minimize a nonconvex function? Or is it better? The answers to these questions are not readily apparent from contemporary analyses. However, in our approach, one can answer them if one shows that, for a particular class of functions, the search space is represented by the union of a subset of the regions that our strategy defines. The behavior of a method over the class of functions then reduces to the proved behavior over this subset of regions. In this paper, we demonstrate how this is done for a few classes that are of broad interest.

We mention at the outset that our approach might be viewed as offering something between contemporary global complexity and local convergence rate guarantees. Our approach does not merely offer a local convergence rate guarantee since the problem-dependent regions that we define do not only include points in the search space that are within small neighborhoods of local minimizers. Indeed, for certain classes of functions of common interest, our regions include the entire search space. On the other hand, our approach does not offer a global complexity bound in all cases since our partitioning strategy might leave out certain subsets of the search space for some functions. All of this being said, we contend that our approach can be very informative to users of optimization methods. Overall, to have a well-informed understanding of the behavior of an algorithm, one should analyze its global convergence properties, worst-case complexity, behavior in the regions that we define in this paper, and its local convergence rate guarantees, when applicable.

1.1 Organization

In §2, we present our strategy for partitioning a search space. In §3 and §4, we employ this partitioning strategy to provide theoretical worst-case performance results for various algorithms, showing in particular that the most interesting results are those for when a pth-order optimization method is employed in a region in which a qth-order method (with $q \neq p$) offers a better predicted reduction. In §5, we prove various results about our partitioning strategy for certain classes of functions, such as how for certain classes our strategy leads to a partitioning containing only a subset of regions. Concluding remarks are provided in §6.

1.2 Notation

We use $\mathbb{R}$ to denote the set of real numbers (i.e., scalars), $\mathbb{R}_{\geq 0}$ (resp., $\mathbb{R}_{>0}$) to denote the set of nonnegative (resp., positive) real numbers, $\mathbb{R}^n$ to denote the set of n-dimensional real vectors, and $\mathbb{R}^{m\times n}$ to denote the set of m-by-n-dimensional real matrices. The set of natural numbers is denoted as $\mathbb{N} := \{0, 1, 2, \dots\}$. Given two scalars $a \in \mathbb{R}_{\geq 0}$ and $b \in \mathbb{R}_{\geq 0}$, we write $a \perp b$ to indicate that $ab = 0$.

We append a natural number, often involving $k \in \mathbb{N}$, as a subscript for a quantity to denote its value during the kth iteration of an algorithm; e.g., $x_k \in \mathbb{R}^n$ denotes the kth value of a variable x. We write $\lambda(M)$ to denote the left-most (with respect to the real line) eigenvalue of a real symmetric matrix M.

2 Partitioning the Search Space

In this section, we propose a strategy for partitioning the search space into distinct regions. Our strategy is generic, but the resulting regions depend on properties of a given objective function and starting point. We do not claim that our strategy is always exhaustive in the sense that, for a given objective function and starting point, the union of the regions defined by our strategy recovers the entire search space. Consequently, we must concede that, for a complete analysis, one needs to consider the performance of an algorithm over the complement, with respect to the search space, of the union of the regions that our strategy defines. In this manner, our strategy can be seen to be complementary to contemporary approaches that provide worst-case upper bounds on the number of iterations until a pth-order stationarity condition is approximately satisfied.

For simplicity in our presentation, let us assume that one is interested in analyzing the worst-case behavior of a *descent* algorithm, i.e., one for which, given an objective $f : \mathbb{R}^n \to \mathbb{R}$ and starting point $x_0 \in \mathbb{R}^n$, the search space can be limited to

$$\mathcal{L} := \{x \in \mathbb{R}^n : f(x) \leq f(x_0)\}.$$

This and various other quantities that we define depend on the objective function f and starting point x_0. For simplicity in our presentation, we do not indicate this dependence explicitly, though it should remain clear throughout that these quantities are problem- and starting-point-dependent.

Our strategy requires that the following assumption holds true. (One could extend our strategy to other settings as well, in which case one might wish to replace this assumption with a looser alternative; see §6 for further discussion.) We employ similar notation as used, e.g., in [1]; in particular, the pth-order derivative of a function f at x is given by the pth-order tensor $\nabla^p f(x)$, and the application of this tensor $j \in \mathbb{N}$ times to a vector $s \in \mathbb{R}^n$ is written as $\nabla^p f(x)[s]^j$.

Assumption 2.1. *Over an open set $\mathcal{L}^+$ containing the sublevel set $\mathcal{L}$ and for some integer $\bar{p} \geq 1$, the objective function $f : \mathbb{R}^n \to \mathbb{R}$ is $\bar{p}$-times continuously differentiable and bounded below by a scalar $f_{\inf} := \inf_{x\in\mathbb{R}^n} f(x) > -\infty$. In addition, for each integer $p \in \{1, \dots, \bar{p}\}$, the pth-order derivative of f is Lipschitz continuous with some minimum Lipschitz constant $L_p \in \mathbb{R}_{>0}$ in that*

$$\|\nabla^p f(x) - \nabla^p f(\bar{x})\|_{[p]} \leq (p-1)! L_p \|x - \bar{x}\|_2 \quad \text{for all} \quad (x, \bar{x}) \in \mathbb{R}^n \times \mathbb{R}^n,$$

where $\|\cdot\|_{[p]}$ denotes the tensor norm recursively induced by the Euclidean norm $\|\cdot\|_2$; see [1, eq. (2.2)–(2.3)]. A consequence of Lipschitz continuity of these derivatives is that, for each $p \in \{1, \dots, \bar{p}\}$, with the pth-order

Taylor series approximation of f centered at $x \in \mathcal{L}^+$, namely, the function $t_p(x,\cdot) : \mathbb{R}^n \to \mathbb{R}$ defined by

$$t_p(x,s) = f(x) + \sum_{j=1}^{p} \frac{1}{j!} \nabla^j f(x)[s]^j, \tag{2.1}$$

one has the property (see [1, eq. (2.6)]) that, for all $s \in \mathbb{R}^n$,

$$f(x+s) \leq t_p(x,s) + \frac{L_p}{p} \|s\|_2^{p+1}. \tag{2.2}$$

Under this assumption, for each $p \in \{1,\ldots,\bar{p}\}$, let $m_p(x,\cdot) : \mathbb{R}^n \to \mathbb{R}$ represent the sum of the pth-order term of the Taylor series approximation of f centered at $x \in \mathcal{L}^+$ and a $(p+1)$st-order regularization term with constant $r_p \in \mathbb{R}_{>0}$, i.e.,

$$m_p(x,s) = \frac{1}{p!} \nabla^p f(x)[s]^p + \frac{r_p}{p+1} \|s\|_2^{p+1}.$$

The specific value of r_p, which depends on f and p but is independent of x, will be discussed later in this section. For now, let us simply note that the model function $m_p(x,\cdot)$ is coercive, so it has a minimum norm global minimizer $s_{m_p}(x) \in \mathbb{R}^n$ with which we define the reduction function $\Delta m_p : \mathcal{L}^+ \to \mathbb{R}$ defined by

$$\Delta m_p(x) = m_p(x,0) - m_p(x, s_{m_p}(x)) \geq 0.$$

For brevity, we refer to this quantity as *the reduction in the pth-order term from x*.

We now propose to partition the search space into regions based on which, if any, is the lowest order for which the reduction in the pth-order term is proportional to an objective value error. Specifically, given a scalar $\kappa \in (0,1)$ and a reference objective value $f_{\text{ref}} \in [f_{\text{inf}}, f(x_0)]$, we define

$$\mathcal{R}_1 := \{x \in \mathcal{L} : \Delta m_1(x) \geq \kappa(f(x) - f_{\text{ref}})\}$$
$$\text{and} \quad \mathcal{R}_p := \{x \in \mathcal{L} : \Delta m_p(x) \geq \kappa(f(x) - f_{\text{ref}})\} \setminus \bigcup_{j=1}^{p-1} \mathcal{R}_j \quad \text{for all} \quad p \in \{2,\ldots,\bar{p}\}.$$

For some functions and certain (κ, f_{ref}), the union of these regions recovers the search space in that $\mathcal{R}_1 \cup \cdots \cup \mathcal{R}_{\bar{p}} = \mathcal{L}$; see §5. However, for others, one need also consider

$$\overline{\mathcal{R}} := \mathcal{L} \setminus \bigcup_{j=1}^{\bar{p}} \mathcal{R}_j;$$

see Figure 1 for an illustration of these sets corresponding to a couple two-dimensional functions. The particular value of κ employed in these definitions might vary from one analysis to the next depending on the class of objective functions that is considered. For example, in §5.2, we show that a certain choice naturally arises in, e.g., the context of strongly convex functions. Similarly, the value of f_{ref} that is employed might vary as well. If one can ensure for a particular class of functions that the algorithm of interest will converge to global optimality, then one can consider $f_{\text{ref}} = f_{\text{inf}}$; otherwise, one might define it *a posteriori* as the limiting value $f_{\text{ref}} = \lim_{k\to\infty} f(x_k)$; see §6. Overall, while it is true that our partitioning strategy depends on these exogenous constant values, we contend that as long as one employs the same values when analyzing each of a set of algorithms, these definitions for regions establishes a consistent and useful framework for comparing algorithm performance.

We also propose that each region above can be partitioned further into regions according to which order term reduction is the largest. In particular, given a pair of integers $(p,q) \in \{1,\ldots,\bar{p}\} \times \{p,\ldots,\bar{p}\}$, we define

$$\mathcal{R}_{pq} := \{x \in \mathcal{R}_p : q \text{ is the smallest integer in } \{p,\ldots,\bar{p}\}$$
$$\text{such that } \Delta m_q(x) \geq \Delta m_r(x) \text{ for all } r \in \{p,\ldots,\bar{p}\}\} \subseteq \mathcal{R}_p.$$

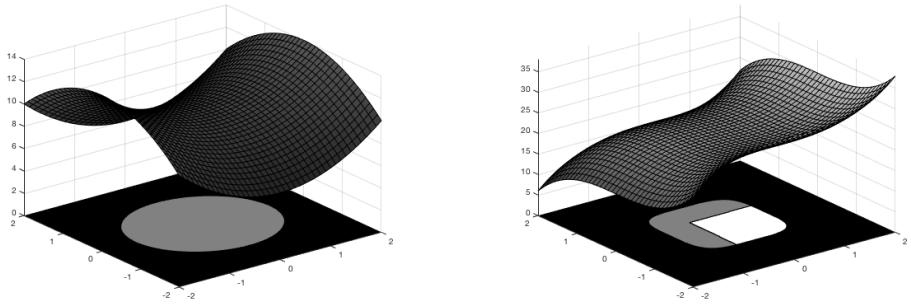

Figure 1: With $\bar{p} = 2$, illustration of $\mathcal{R}_1$ (black), $\mathcal{R}_2$ (gray), and $\overline{\mathcal{R}}$ (white) for the two-dimensional functions $f(x, y) = x^2 - y^2 + 10$ (left) and $f(x) = x^3 - y^3 + 22$ (right).

Here, we restrict attention to $q \geq p$ since the definitions of our regions imply that

$$\text{if} \quad p > q, \quad \text{then} \quad \{x \in \mathcal{R}_p : \Delta m_q(x) \geq \Delta m_p(x)\} = \emptyset.$$

For example, if $\bar{p} = 2$, then the region $\mathcal{R}_1$ is partitioned into

$$\begin{aligned} \mathcal{R}_{11} &:= \{x \in \mathcal{R}_1 : \Delta m_1(x) \geq \Delta m_2(x)\} \\ \text{and} \quad \mathcal{R}_{12} &:= \{x \in \mathcal{R}_1 : \Delta m_1(x) < \Delta m_2(x)\} \end{aligned}$$

whereas one finds $\mathcal{R}_{22} = \mathcal{R}_2$ since the set $\{x \in \mathcal{R}_2 : \Delta m_1(x) \geq \Delta m_2(x)\}$ is empty by the definitions of the sets $\mathcal{R}_1$ and $\mathcal{R}_2$.

Based on this partitioning strategy, we propose that an algorithm for nonconvex optimization should be characterized by upper bounds on the numbers of iterations that it could spend in each region. In terms of the regions in the set $\{\mathcal{R}_1, \ldots, \mathcal{R}_{\bar{p}}\}$, this upper bound could be given as a function of $\epsilon_f \in (0, \infty)$, where one is interested in a (possibly infinite) upper bound for $k \in \mathbb{N}$ until one must have $f(x_k) - f_{\text{ref}} \leq \epsilon_f$.

Let us now return to the value for r_p in the definition of m_p. Toward this end, let us first define the function $w_p(x, \cdot) : \mathbb{R}^n \to \mathbb{R}$ as the sum of the pth-order Taylor series approximation of f centered at $x \in \mathcal{L}^+$ (recall (2.1)) and a $(p+1)$st-order regularization term with a specific constant l_p, namely,

$$w_p(x, s) = t_p(x, s) + \frac{l_p}{p+1}\|s\|_2^{p+1}, \quad \text{where} \quad l_p := \frac{2(p+1)L_p}{p}. \tag{2.3}$$

To motivate the value of l_p in this definition, first consider the following lemma.

Lemma 2.1. *For each $x \in \mathcal{L}^+$ and $p \in \{1, \ldots, \bar{p}\}$, it follows that*

$$t_p(x, 0) - t_p(x, s) \geq \frac{l_p}{p+1}\|s\|_2^{p+1} \text{ for all } s \in \mathbb{R}^n \text{ such that } w_p(x, s) \leq w_p(x, 0).$$

Proof. The inequality follows as in [1, Lem. 2.1]. □

Next, defining $s_{w_p}(x)$ as a minimum norm global minimizer of the model $w_p(x, \cdot)$, notice that (2.2), Lemma 2.1, and the value for l_p as defined in (2.3) yield

$$\begin{aligned} \frac{f(x) - f(x + s_{w_p}(x))}{t_p(x, 0) - t_p(x, s_{w_p}(x))} &= \frac{t_p(x, s_{w_p}(x)) - f(x + s_{w_p}(x))}{t_p(x, 0) - t_p(x, s_{w_p}(x))} + 1 \\ &\geq -\frac{\frac{L_p}{p}\|s_{w_p}(x)\|_2^{p+1}}{\frac{l_p}{p+1}\|s_{w_p}(x)\|_2^{p+1}} + 1 = \frac{1}{2}. \end{aligned} \tag{2.4}$$

This completes the motivation for the value of l_p in the definition of $w_p(x,\cdot)$ in (2.3); in particular, it is with this regularization value that one finds, in (2.4), that the reduction in the objective f obtained from x to $x + s_{w_p}(x)$ is at least half of the reduction in the Taylor model $t_p(x,\cdot)$ obtained from 0 to $s_{w_p}(x)$.

We now define r_p as a scalar greater than $pl_p/2$ such that, over all $x \in \mathcal{L}^+$, the minimum norm global minimizer of $m_p(x,s)$ (namely, $s_{m_p}(x)$) has Euclidean norm less than or equal to that of the minimum norm global minimizer of $w_p(x,s)$ (namely, $s_{w_p}(x)$); i.e., under this assumption, for all $x \in \mathcal{L}^+$, one has

$$\|s_{m_p}(x)\|_2 \leq \|s_{w_p}(x)\|_2. \tag{2.5}$$

The important consequences of this choice, which motivate our interest in it, are captured as part of the following lemma that will be used later on.

Lemma 2.2. *For any $x \in \mathcal{L}^+$, the reduction in the pth-order term from x is*

$$\Delta m_p(x) = \frac{r_p}{p(p+1)}\|s_{m_p}(x)\|_2^{p+1}. \tag{2.6}$$

In addition, since $r_p > pl_p/2$ is defined to yield (2.5), it follows that, for any $x \in \mathcal{L}^+$, the reduction in $t_p(x,\cdot)$ yielded by $s_{w_p}(x)$ satisfies

$$t_p(x,0) - t_p(x, s_{w_p}(x)) \geq \iota_p \Delta m_p(x), \text{ where } \iota_p := \frac{pl_p}{r_p} \in (0,2). \tag{2.7}$$

Proof. First-order stationarity of $s_{m_p}(x)$ with respect to $m_p(x,\cdot)$ implies

$$0 = \frac{1}{p}s_{m_p}(x)^T\nabla_s m_p(x, s_{m_p}(x)) = \frac{1}{p!}\nabla^p f(x)[s_{m_p}(x)]^p + \frac{r_p}{p}\|s_{m_p}(x)\|_2^{p+1},$$

from which it follows that

$$\Delta m_p(x) = \frac{r_p}{p}\|s_{m_p}(x)\|_2^{p+1} - \frac{r_p}{p+1}\|s_{m_p}(x)\|_2^{p+1} = \frac{r_p}{p(p+1)}\|s_{m_p}(x)\|_2^{p+1},$$

as desired. This, Lemma 2.1, and (2.5) imply that

$$\begin{aligned} t_p(x,0) - t_p(x, s_{w_p}(x)) &\geq \frac{l_p}{p+1}\|s_{w_p}(x)\|_2^{p+1} \\ &\geq \frac{l_p}{p+1}\|s_{m_p}(x)\|_2^{p+1} = \frac{pl_p}{r_p}\Delta m_p(x) = \iota_p \Delta m_p(x), \end{aligned}$$

which is the second desired conclusion. □

The previous lemma reveals that, with our definition of r_p, we have guaranteed that the reduction in the pth-order Taylor series model that one obtains with a conservative algorithm, namely, one that obtains a step by minimizing a regularization of this model, is at least proportional to the reduction in the pth-order term model used in the definitions of our regions that partition the search space.

For future reference, let us quantify $\Delta m_p(x)$ for $p \in \{1,2\}$ in another manner.

Theorem 2.1. *Under Assumption 2.1 for $\bar{p} \geq 2$, it follows that, for any $x \in \mathcal{L}^+$,*

$$\Delta m_1(x) = \frac{1}{2r_1}\|\nabla f(x)\|_2^2 \quad \text{and} \quad \Delta m_2(x) = \frac{1}{6r_2^2}\max\{-\lambda(\nabla^2 f(x)), 0\}^3.$$

Proof. Let $x \in \mathcal{L}$ be arbitrary. Since $m_1(x,s) = \nabla f(x)^T s + \frac{1}{2}r_1\|s\|_2^2$, one finds that the global minimizer of $m_1(x,\cdot)$ is $s_{m_1}(x) = -\frac{1}{r_1}\nabla f(x)$, meaning that

$$\begin{aligned} \Delta m_1(x) &= m_1(x,0) - m_1(x, s_{m_1}(x)) \\ &= -\nabla f(x)^T s_{m_1}(x) - \frac{r_1}{2}\|s_{m_1}(x)\|_2^2 = \frac{1}{2r_1}\|\nabla f(x)\|_2^2, \end{aligned}$$

as desired. Now consider $m_2(x,s) = \frac{1}{2}s^T\nabla^2 f(x)s + \frac{1}{3}r_2\|s\|_2^3$. If $\nabla^2 f(x) \succeq 0$, then the minimum norm global minimizer of $m_2(x,\cdot)$ is $s_{m_2}(x) = 0$. Otherwise, the global minimum of $m_2(x,\cdot)$ is achieved at an eigenvector $s_{m_2}(x)$ corresponding to the left-most eigenvalue of $\nabla^2 f(x)$, scaled so that it satisfies the first-order necessary condition

$$(\nabla^2 f(x) + r_2\|s_{m_2}(x)\|_2 I)s_{m_2}(x) = 0,$$

which in particular implies that $\|s_{m_2}(x)\|_2 = -\lambda(\nabla^2 f(x))/r_2$. Thus,

$$\begin{aligned}
\Delta m_2(x) &= m_2(x,0) - m_2(x, s_{m_2}(x)) \\
&= -\frac{1}{2}s_{m_2}(x)^T\nabla^2 f(x)s_{m_2}(x) - \frac{r_2}{3}\|s_{m_2}(x)\|_2^3 \\
&= -\frac{1}{2}\lambda(\nabla^2 f(x))\|s_{m_2}(x)\|_2^2 - \frac{r_2}{3}\|s_{m_2}(x)\|_2^3 \\
&= \frac{1}{2r_2^2}|\lambda(\nabla^2 f(x))|^3 - \frac{1}{3r_2^2}|\lambda(\nabla^2 f(x))|^3 = \frac{1}{6r_2^2}|\lambda(\nabla^2 f(x))|^3.
\end{aligned}$$

Combining the results of the two cases yields the desired conclusion. □

3 Analyses of pth-order Regularization Methods

One can employ our proposed characterization strategy to analyze the worst-case behavior of any algorithm for solving nonconvex optimization problems. To demonstrate this, in this section, we provide results on the worst-case behavior of algorithms that, in a straightforward manner, compute steps based on minimizing regularized pth-order Taylor series models of the objective, assuming the relevant constants in Assumption 2.1 are known. By contrast, in §4, we provide results on the worst-case behavior of some second-order trust region methods, which do not fall into this class of regularization algorithms.

It is worthwhile to mention at the outset that, even though the regularization methods that we study in this section were used to motivate the definitions of our regions that partition the search space, they are *not* guaranteed to possess the best performance guarantees within our characterization strategy; see §3.4 for a discussion on why this is the case. That said, these methods do establish a reasonable baseline against which the worst-case behavior of other algorithms can be compared, as we do with the second-order trust region methods studied in the next section.

3.1 Generic Results

An algorithm that computes steps based on minimizing the regularized pth-order Taylor series model $w_p(x,\cdot)$ at any $x \in \mathcal{L}$ satisfies the conclusion of the following theorem. (It is worthwhile to note that we refer to an algorithm that minimizes $w_p(x,\cdot)$, which involves the constant $l_p > L_p$, rather than one with a regularization term involving L_p. This is consistent with [1], in which it is shown that one always obtains an acceptable step for a sufficiently large regularization constant, which in our setting is represented by l_p.)

Theorem 3.1. *Under Assumption 2.1, suppose that an iterate sequence $\{x_k\}$ is generated by an algorithm defined by the iteration*

$$x_{k+1} \leftarrow x_k + s_{w_p}(x_k). \tag{3.1}$$

Then, with $\Delta f_{ref} := f(x_0) - f_{ref}$, given any scalar $\epsilon_f \in (0, \Delta f_{ref})$, the number of iterations in the set $\mathcal{R}_p \cap \{x \in \mathbb{R}^n : f(x) - f_{ref} \geq \epsilon_f\}$ is at most

$$\left\lceil \log\left(\frac{\Delta f_{ref}}{\epsilon_f}\right)\left(\log\left(\frac{1}{1-\frac{\iota_p\kappa}{2}}\right)\right)^{-1} \right\rceil = \mathcal{O}\left(\log\left(\frac{\Delta f_{ref}}{\epsilon_f}\right)\right). \tag{3.2}$$

Proof. Under Assumption 2.1, it follows by the inequalities in (2.4) and (2.7) that, for all $x_k \in \mathcal{R}_p \cap \{x \in \mathbb{R}^n : f(x) - f_{\text{ref}} \geq \epsilon_f\}$, one finds

$$\begin{aligned} f(x_k) - f(x_{k+1}) &\geq \frac{1}{2}(t_p(x_k, 0) - t_p(x_k, s_{w_p}(x_k))) \\ &\geq \frac{\iota_p}{2}\Delta m_p(x_k) \\ &\geq \frac{\iota_p \kappa}{2}(f(x_k) - f_{\text{ref}}). \end{aligned} \tag{3.3}$$

Adding and subtracting f_{ref} on the left-hand side and rearranging yields

$$\left(1 - \frac{\iota_p \kappa}{2}\right)(f(x_k) - f_{\text{ref}}) \geq f(x_{k+1}) - f_{\text{ref}}.$$

Applying this fact repeatedly, one finds that, for any $\{k_1, \ldots, k_j\} \subseteq \mathcal{R}_p$, it holds that

$$\left(1 - \frac{\iota_p \kappa}{2}\right)^j \Delta f_{\text{ref}} \geq f(x_{k_j}) - f_{\text{ref}}.$$

Since the right-hand side of this inequality is guaranteed to be less than or equal to ϵ_f if the left-hand side is less than or equal to this value, one can set the left-hand side less than or equal to ϵ_f, take the logarithm of both sides of the resulting inequality, and rearrange to conclude that the number of iterations that the algorithm spends in $\mathcal{R}_p \cap \{x \in \mathbb{R}^n : f(x) - f_{\text{ref}} \geq \epsilon_f\}$ is bounded above by (3.2), as desired. □

One also has the following result stating a limitation of a straightforward pth-order regularization method in terms of its worst-case behavior.

Theorem 3.2. *For any $k \in \mathbb{N}$ and $q > p$, there exists a function f satisfying Assumption 2.1 with $\bar{p} \geq q$ such that, for any $\epsilon_f \in (0, \Delta f_{ref})$, the number of iterations in the set $\mathcal{R}_q \cap \{x \in \mathbb{R}^n : f(x) - f_{ref} \geq \epsilon_f\}$ is at least k.*

Proof. The result follows from the fact that, if for some $k \in \mathbb{N}$ and any $q > p$ the iterate x_k is a qth-order stationary point with $f(x_k) - f_{\text{ref}} \geq \epsilon_f$, then the iteration (3.1) yields $s_{w_p}(x_k) = 0$, meaning that $x_{\bar{k}+1} \leftarrow x_{\bar{k}}$ for all $\bar{k} \in \mathbb{N}$ with $\bar{k} \geq k$. □

Theorem 3.2 reveals that one cannot establish a *deterministic* upper bound on the number of iterations that a regularized pth-order method spends in regions surrounding qth-order stationary points for any $q > p$. That said, for certain methods it might be possible to establish *probabilistic* upper bounds, either due to randomization of the starting point or by introducing randomization into an algorithm itself; e.g., see [18, 21]. Such probabilistic bounds are valid to introduce into our characterization approach, though we do not discuss such results further in this paper.

In the next two subsections, we provide a few additional results for regularized pth-order methods, specifically for $p = 1$ and $p = 2$. In each case, we present results related to our characterization strategy, contrasting the results with those one obtains with a contemporary analysis that bounds the number of iterations until approximate first-order and/or second-order stationarity is satisfied.

3.2 Regularized Gradient Method

Consider the behavior of a first-order regularization method defined by the iteration

$$x_{k+1} \leftarrow x_k + s_{w_1}(x_k). \tag{RG}$$

Since, as is easily shown, $s_{w_1}(x) = -\frac{1}{l_1}\nabla f(x)$, this is nothing more than the standard gradient descent method with constant stepsize $1/l_1$; we refer to it as the *regularized gradient* (RG) method. In commonly

accepted contemporary analyses, the worst-case behavior of this approach would be characterized in the following manner. First, by Assumption 2.1 and the fact that $l_1 > 2L_1$, it follows that

$$\begin{aligned} f(x_{k+1}) &\leq f(x_k) + \nabla f(x_k)^T(x_{k+1} - x_k) + \frac{l_1}{2}\|x_{k+1} - x_k\|_2^2 \\ &= f(x_k) - \frac{1}{l_1}\|\nabla f(x_k)\|_2^2 + \frac{1}{2l_1}\|\nabla f(x_k)\|_2^2 \\ &= f(x_k) - \frac{1}{2l_1}\|\nabla f(x_k)\|_2^2. \end{aligned} \tag{3.4}$$

This, with $\mathcal{K}(\epsilon_g) := \{k \in \mathbb{N} : \|\nabla f(x_k)\|_2 > \epsilon_g\}$ and $\Delta f_{\inf} := f(x_0) - f_{\inf}$, where $f_{\inf}$ is defined in Assumption 2.1, implies that

$$\Delta f_{\inf} \geq \sum_{k=0}^{\infty}(f(x_k) - f(x_{k+1})) \geq \sum_{k \in \mathcal{K}(\epsilon_g)}(f(x_k) - f(x_{k+1})) \geq \frac{1}{2l_1}|\mathcal{K}(\epsilon_g)|\epsilon_g^2.$$

Hence, by rearrangement, the cardinality of the set $\mathcal{K}(\epsilon_g)$ is bounded by

$$\lceil 2l_1 \Delta f_{\inf} \epsilon_g^{-2} \rceil = \mathcal{O}(l_1 \Delta f_{\inf} \epsilon_g^{-2}). \tag{3.5}$$

While this analysis is valid, providing a characterization of the worst-case behavior of the method under loose assumptions on f, it only provides a conservative bound. After all, notice that the analysis is based on a reduction in f that is proportional to ϵ_g^2 for *every* step. This might be extremely small compared to the true reduction achieved, especially when far from a minimizer! By contrast, Theorem 3.1 reveals that the algorithm actually possesses a linear rate of convergence in $\mathcal{R}_1$, a region that might cover a large portion of the search space, e.g., in the case of functions satisfying the Polyak-Łojasiewicz condition; see §5.2.

We summarize the worst-case behavior of the iteration (RG) along with that of the next method later in this section; see Theorem 3.3 in §3.4.

3.3 Regularized Newton Method

Let us now consider the behavior of a regularized second-order method defined by the iteration

$$x_{k+1} \leftarrow x_k + s_{w_2}(x_k). \tag{RN}$$

This iteration is commonly known as a cubic regularization method; for our purposes, we refer to it as the *regularized Newton* (RN) method. A contemporary analysis of this approach would proceed in the following manner. First, from [5, 6, 24], it is known that for any $x \in \mathcal{L}^+$ the step $s_{w_2}(x)$ yields

$$\nabla f(x)^T s_{w_2}(x) \leq 0, \tag{3.6a}$$

$$(\nabla^2 f(x) + l_2\|s_{w_2}(x)\|_2 I) \succeq 0, \tag{3.6b}$$

$$\text{and} \quad (\nabla^2 f(x) + l_2\|s_{w_2}(x)\|_2 I)s_{w_2}(x) = -\nabla f(x), \tag{3.6c}$$

from which it follows that

$$\begin{aligned} & w_2(x,0) - w_2(x, s_{w_2}(x)) \\ = & -\nabla f(x)^T s_{w_2}(x) - \frac{1}{2}s_{w_2}(x)^T \nabla^2 f(x) s_{w_2}(x) - \frac{l_2}{3}\|s_{w_2}(x)\|_2^3 \\ = & -\frac{1}{2}\nabla f(x)^T s_{w_2}(x) + \frac{l_2}{6}\|s_{w_2}(x)\|_2^3 \\ \geq & \frac{l_2}{6}\|s_{w_2}(x)\|_2^3. \end{aligned} \tag{3.7}$$

From this, and since Assumption 2.1, (2.3), and the fact that $l_2 > L_2$ imply that $f(x_k) = w_2(x_k, 0)$ and $f(x_{k+1}) \leq w_2(x_k, s_{w_2}(x_k))$, it follows that

$$f(x_k) - f(x_{k+1}) \geq w_2(x_k, 0) - w_2(x_k, s_{w_2}(x_k)) \geq \frac{l_2}{6}\|s_{w_2}(x_k)\|_2^3. \tag{3.8}$$

On the other hand, with (3.6c) and since $l_2 > L_2$, it follows for any $k \in \mathbb{N}$ that

$$\begin{aligned}
&\|\nabla f(x_k + s_{w_2}(x_k))\|_2 \\
&= \|\nabla f(x_k + s_{w_2}(x_k)) - (\nabla f(x_k) + (\nabla^2 f(x_k) + l_2\|s_{w_2}(x_k)\|_2 I)s_{w_2}(x_k))\|_2 \\
&\leq \|\nabla f(x_k + s_{w_2}(x_k)) - (\nabla f(x_k) + \nabla^2 f(x_k)s_{w_2}(x_k))\|_2 + l_2\|s_{w_2}(x_k)\|_2^2 \\
&\leq \left\|\int_0^1 (\nabla^2 f(x_k + \tau s_{w_2}(x_k)) - \nabla^2 f(x_k))s_{w_2}(x_k)d\tau\right\|_2 + l_2\|s_{w_2}(x_k)\|_2^2 \\
&\leq \int_0^1 \left\|\nabla^2 f(x_k + \tau s_{w_2}(x_k)) - \nabla^2 f(x_k)\right\|_2 d\tau \cdot \|s_{w_2}(x_k)\|_2 + l_2\|s_{w_2}(x_k)\|_2^2 \\
&\leq \int_0^1 \tau d\tau \cdot l_2\|s_{w_2}(x_k)\|_2^2 + l_2\|s_{w_2}(x_k)\|_2^2 \\
&\leq \frac{3}{2}l_2\|s_{w_2}(x_k)\|_2^2,
\end{aligned}$$

from which it follows that

$$\|s_{w_2}(x_k)\|_2 \geq \left(\frac{2}{3l_2}\right)^{1/2} \|\nabla f(x_k + s_{w_2}(x_k))\|_2^{1/2}.$$

Combining this with (3.8), one obtains that

$$f(x_k) - f(x_{k+1}) \geq \frac{1}{6}\left(\frac{2}{3}\right)^{3/2}\left(\frac{1}{l_2}\right)^{1/2} \|\nabla f(x_{k+1})\|_2^{3/2}. \tag{3.9}$$

Hence, defining $\mathcal{K}(\epsilon_g) := \{k \in \mathbb{N} : \|\nabla f(x_{k+1})\|_2 > \epsilon_g\}$, one finds that

$$\begin{aligned}
\Delta f_{\inf} &\geq \sum_{k=0}^{\infty}(f(x_k) - f(x_{k+1})) \\
&\geq \sum_{k\in\mathcal{K}(\epsilon_g)}(f(x_k) - f(x_{k+1})) \geq \frac{1}{6}\left(\frac{2}{3}\right)^{3/2}\left(\frac{1}{l_2}\right)^{1/2} |\mathcal{K}(\epsilon_g)|\epsilon_g^{3/2},
\end{aligned}$$

from which it follows, using a similar argument as for the regularized gradient method in §3.2, that the cardinality of the set $\mathcal{K}(\epsilon_g)$ is bounded above by

$$6\left(\frac{2}{3}\right)^{-3/2} l_2^{1/2}\Delta f_{\inf}\epsilon_g^{-3/2} = \mathcal{O}(l_2^{1/2}\Delta f_{\inf}\epsilon_g^{-3/2}). \tag{3.10}$$

Further, one can combine (3.6b) and (3.8) to conclude that

$$f(x_k) - f(x_{k+1}) \geq \frac{l_2}{6}\|s_{w_2}(x_k)\|_2^3 \geq \frac{1}{6l_2^2}(-\lambda(\nabla^2 f(x_k)))^3.$$

Hence, the cardinality of $\mathcal{K}_2(\epsilon_H) := \{k \in \mathbb{N} : \lambda(\nabla^2 f(x_k)) < -\epsilon_H\}$ is bounded by

$$\lceil 6l_2^2\Delta f_{\inf}\epsilon_H^{-3}\rceil = \mathcal{O}(l_2^2\Delta f_{\inf}\epsilon_H^{-3}). \tag{3.11}$$

As in the case of the contemporary analysis for the regularized gradient method shown in §3.2, both upper bounds (3.10) and (3.11) are quite conservative in that they are based on reductions of f for each step on the order of $\epsilon_g^{3/2}$ and ϵ_H^3, respectively. These can be small compared to the true reductions achieved throughout much of the optimization process. An analysis based on our strategy, on the other hand, can be more revealing in important cases of interest. We present such an analysis now.

Theorem 3.1 reveals that the algorithm possesses a linear rate of convergence in $\mathcal{R}_2$. Let us now analyze the behavior of the approach over $\mathcal{R}_1 = \mathcal{R}_{11} \cup \mathcal{R}_{12}$. For $x_k \in \mathcal{R}_{12}$ with $f_{\text{ref}} = f_{\text{inf}}$, it follows with (3.3) that

$$f(x_k) - f(x_{k+1}) \geq \frac{\iota}{2}\Delta m_2(x_k) \geq \frac{\iota}{2}\Delta m_1(x_k) \geq \frac{\iota\kappa}{2}(f(x_k) - f_{\text{inf}}).$$

As in the proof of Theorem 3.1, this shows that the convergence rate is linear over this region with an upper bound on the number of iterations in $\mathcal{R}_{12} \cap \{x \in \mathbb{R}^n : f(x) - f_{\text{inf}} \geq \epsilon_f\}$ being bounded above by (3.2).

Now consider $\mathcal{R}_{11}$. As in the contemporary analysis above, it will be convenient to work with $x_{k+1} \in \mathcal{R}_{11}$. Corresponding to such a point, one finds that

$$s_{m_1}(x_{k+1}) = -\frac{1}{r_1}\nabla f(x_{k+1}), \quad \text{meaning} \quad \Delta m_1(x_{k+1}) = \frac{1}{2r_1}\|\nabla f(x_{k+1})\|_2^2.$$

Hence, the definition of $\mathcal{R}_{11} \subseteq \mathcal{R}_1$ with $f_{\text{ref}} = f_{\text{inf}}$ implies that

$$\|\nabla f(x_{k+1})\|_2^2 = 2r_1\Delta m_1(x_{k+1}) \geq 2r_1\kappa(f(x_{k+1}) - f_{\text{inf}}).$$

Combining this with (3.9), it follows that

$$f(x_k) - f(x_{k+1}) \geq \frac{1}{6}\left(\frac{2}{3}\right)^{3/2}\left(\frac{1}{l_2}\right)^{1/2}(2r_1\kappa)^{3/4}(f(x_{k+1}) - f_{\text{inf}})^{3/4}.$$

Adding and subtracting f_{inf} from the left-hand side, one obtains that

$$\begin{aligned}&(f(x_k) - f_{\text{inf}}) - (f(x_{k+1}) - f_{\text{inf}})\\ &\geq \frac{1}{6}\left(\frac{2}{3}\right)^{3/2}\left(\frac{1}{l_2}\right)^{1/2}(2r_1\kappa)^{3/4}(f(x_{k+1}) - f_{\text{inf}})^{3/4}.\end{aligned} \tag{3.12}$$

Let $c := 6^{-1}2^{3/2}3^{-3/2}l_2^{-1/2}(2r_1\kappa)^{3/4}$ and consider any $\xi \in \mathbb{R}_{>0}$. There are two cases.

Case 1: If $c \geq \xi(f(x_{k+1}) - f_{\text{inf}})^{1/4}$, then (3.12) implies that

$$f(x_k) - f_{\text{inf}} \geq (1 + \xi)(f(x_{k+1}) - f_{\text{inf}}).$$

Hence, for any $\{k_1 + 1, \ldots, k_j + 1\} \subseteq \mathcal{R}_{11}$, it holds that

$$\frac{1}{(1+\xi)^j}\Delta f_{\text{inf}} \geq f(x_{k_j+1}) - f_{\text{inf}},$$

which means that the number of iterations in the set $\mathcal{R}_{11} \cap \{x \in \mathbb{R}^n : f(x) - f_{\text{inf}} \geq \epsilon_f\}$ is bounded by

$$\left\lceil \log\left(\frac{\Delta f_{\text{inf}}}{\epsilon_f}\right)\left(\frac{1}{\log(1+\xi)}\right)\right\rceil = \mathcal{O}\left(\log\left(\frac{\Delta f_{\text{inf}}}{\epsilon_f}\right)\right). \tag{3.13}$$

This shows a linear rate of convergence in this case. (We note that, with additional assumptions on f, such as assuming that the Hessian of f is positive definite at a limit point of the iterate sequence, one might show that the algorithm achieves a *superlinear* rate of local convergence. However, such considerations are out of the scope of this paper.)

Case 2: If $c < \xi(f(x_{k+1}) - f_{\text{inf}})^{1/4}$, then it follows from (3.12) that

$$(f(x_k) - f_{\text{inf}}) - (f(x_{k+1}) - f_{\text{inf}}) \geq c^4\xi^{-3} = 2^5 3^{-10} l_2^{-2}(r_1\kappa)^3\xi^{-3}.$$

Hence, the number of iterations until $c \geq \xi(f(x_{k+1}) - f_{\text{inf}})^{1/4}$ is (conservatively) bounded above by

$$\left\lceil (2^{-5}3^{10}l_2^2(r_1\kappa)^{-3}\xi^3\Delta f_{\text{inf}}\right\rceil = \mathcal{O}(l_2^2 r_1^{-3}\Delta f_{\text{inf}}). \tag{3.14}$$

We summarize the worst-case behavior of (RN) along with that of (RG) next.

3.4 Summary of Regularized Gradient and Newton Methods

In the previous subsections, for the regularized gradient iteration (RG) and the regularized Newton iteration (RN), we have presented both contemporary types of analyses that bound the numbers of iterations until a first- and/or second-order stationarity condition is approximately satisfied, as well as analyses that bound the number of iterations spent in our regions $\mathcal{R}_1 = \mathcal{R}_{11} \cup \mathcal{R}_{12}$ and $\mathcal{R}_2$ defined in §2. Combining these analyses, we obtain the following theorem revealing distinct, complementary views of the worst-case behaviors of these methods.

Theorem 3.3. *Under Assumption 2.1 with $\bar{p} \geq 2$, let*

$$\mathcal{K}_1(\epsilon_g) := \{k \in \mathbb{N} : \|\nabla f(x_k)\|_2 > \epsilon_g\}$$
$$\text{and} \quad \mathcal{K}_2(\epsilon_H) := \{k \in \mathbb{N} : \lambda(\nabla^2 f(x_k)) < -\epsilon_H\}.$$

Then, if the Hessian function of f is Lipschitz continuous with constant l_2 over the path of computed iterates, the cardinalities of the index sets $\mathcal{K}_1(\epsilon_g)$ and $\mathcal{K}_2(\epsilon_H)$ are of the order in the following table:

Algorithm	$\lvert\mathcal{K}_1(\epsilon_g)\rvert$	$\lvert\mathcal{K}_2(\epsilon_H)\rvert$
(RG)	$\mathcal{O}\left(\frac{l_1 \Delta f_{inf}}{\epsilon_g^2}\right)$	∞
(RN)	$\mathcal{O}\left(\frac{l_2^{1/2} \Delta f_{inf}}{\epsilon_g^{3/2}}\right)$	$\mathcal{O}\left(\frac{l_2^2 \Delta f_{inf}}{\epsilon_H^3}\right)$

On the other hand, the numbers of iterations in the regions $\mathcal{R}_1 = \mathcal{R}_{11} \cup \mathcal{R}_{12}$ and $\mathcal{R}_2$ as defined in §2 with $f_{ref} = f_{inf}$, intersected with the set $\{x \in \mathbb{R}^n : f(x) - f_{inf} \geq \epsilon_f\}$ for $\epsilon_f \in (0, \Delta f_{inf})$, are of the order in the following table:

	$\mathcal{R}_1$		$\mathcal{R}_2$
Algorithm	$\mathcal{R}_{11}$	$\mathcal{R}_{12}$	
(RG)	$\mathcal{O}\left(\log\left(\frac{\Delta f_{inf}}{\epsilon_f}\right)\right)$	$\mathcal{O}\left(\log\left(\frac{\Delta f_{inf}}{\epsilon_f}\right)\right)$	∞
(RN)	$\mathcal{O}\left(\frac{l_2^2 \Delta f_{inf}}{r_1^3}\right) + \mathcal{O}\left(\log\left(\frac{\Delta f_{inf}}{\epsilon_f}\right)\right)$	$\mathcal{O}\left(\log\left(\frac{\Delta f_{inf}}{\epsilon_f}\right)\right)$	$\mathcal{O}\left(\log\left(\frac{\Delta f_{inf}}{\epsilon_f}\right)\right)$

Proof. The bound on the cardinality of the index set $\mathcal{K}_1(\epsilon_g)$ has been proved for (RG) in (3.5), whereas the lack of a bound for this method with respect to $\mathcal{K}(\epsilon_H)$ follows for similar reasons as the proof of Theorem 3.2. As for (RN), the bounds on the cardinalities of the index sets $\mathcal{K}_1(\epsilon_g)$ and $\mathcal{K}(\epsilon_H)$ have been proved in (3.10) and (3.11), respectively. The behavior of (RG) with respect to $\mathcal{R}_1 \cap \{x \in \mathbb{R}^n : f(x) - f_{\inf} \geq \epsilon_f\}$ and that of (RN) with respect to $\mathcal{R}_2 \cap \{x \in \mathbb{R}^n : f(x) - f_{\inf} \geq \epsilon_f\}$ follow as a consequence of Theorem 3.1. On the other hand, the lack of a bound for (RG) with respect to $\mathcal{R}_2 \cap \{x \in \mathbb{R}^n : f(x) - f_{\inf} \geq \epsilon_f\}$ follows as a consequence of Theorem 3.2, and the bound for (RN) with respect to $(\mathcal{R}_{11} \cup \mathcal{R}_{12} = \mathcal{R}_1) \cap \{x \in \mathbb{R}^n : f(x) - f_{\inf} \geq \epsilon_f\}$ has been proved in (3.13)–(3.14). □

Since our strategy for partitioning the search space is based on the reductions achieved in a regularized pth-order derivative model, one might ask whether an algorithm based on minimizing a regularized pth-order Taylor series model is guaranteed to possess the best complexity guarantees in our characterization approach. The answer is *no*. Consider, for example, an algorithm that at each iterate x_k computes the steps $s_{w_1}(x_k)$ and $s_{w_2}(x_k)$ and chooses which to take based on which is larger, $\Delta m_1(x_k)$ or $\Delta m_2(x_k)$; see [10]. The analysis in the previous subsections reveals that the number of iterations for such a method over $(\mathcal{R}_1 \cup \mathcal{R}_2) \cap \{x \in \mathbb{R}^n : f(x) - f_{\text{ref}} > \epsilon_f\}$ is $\mathcal{O}(\log(\Delta f_{\text{ref}}/\epsilon_f))$, which is better than for both the regularized gradient and Newton methods considered in this section. In particular, it is better than the worst-case behavior of (RG) since it achieves the same worst-case behavior of (RN) over $\mathcal{R}_2$, and it is better than the worst-case behavior of (RN) since one does not need to include the $\mathcal{O}(l_2^2 \Delta f_{\text{inf}}/r_1^3)$ term from the "Case 2" phase that (RN) has over $\mathcal{R}_{11}$.

4 Analyses of Second-Order Trust Region Methods

Let us now turn to analyzing the worst-case behavior of a few variants of second-order trust region methods. We consider multiple variants, rather than only one, to illustrate the different properties one might find using our characterization strategy. In contrast to our analyses for (RG) and (RN) in the previous subsections, it is not our intention here to provide the details of a contemporary-type analysis for each of these methods in terms of determining upper bounds on the numbers of iterations to achieve approximate first- or second-order stationarity. Rather, our intention here is to demonstrate the application of our new characterization strategy to these methods.

In each of the methods in this subsection, we suppose that, for all $k \in \mathbb{N}$, a trial step $s_k \in \mathbb{R}^n$ is computed as a global minimizer of

$$\min_{s \in \mathbb{R}^n} \; t_2(x_k, s) \quad \text{s.t.} \quad \|s\|_2 \leq \delta_k \quad \text{for some} \quad \delta_k \in \mathbb{R}_{>0}. \tag{4.1}$$

The variants that we consider derive from the manner in which $\{\delta_k\}$ is chosen.

4.1 Gradient-Dependent Radii

Let us first consider a second-order trust region method that operates in the following manner.

- For $k = 0$, one selects $\nu_0 \in [\underline{\nu}, \overline{\nu}] \subset \mathbb{R}_{>0}$.
- For all $k \in \mathbb{N}$, one sets $\delta_k \leftarrow \nu_k \|\nabla f(x_k)\|_2$ and computes s_k by solving (4.1).
- For all $k \in \mathbb{N}$ and some $(\eta, \beta) \in (0,1) \times (0,1)$, if

$$\rho_k := \frac{f(x_k) - f(x_k + s_k)}{t_2(x_k, 0) - t_2(x_k, s_k)} \geq \eta,$$

  then one sets $x_{k+1} \leftarrow x_k + s_k$ and $\nu_{k+1} \in [\underline{\nu}, \overline{\nu}]$ (i.e., the step is "successful"); otherwise, one sets $x_{k+1} \leftarrow x_k$ and $\nu_{k+1} \leftarrow \beta \nu_k$ (i.e., the step is "rejected").

This is a standard type of trust region method with an iterate and trust region radius update that depends on an actual-to-predicted reduction in the objective from the previous step, except that at each new distinct iterate the trust region radius is initialized to a fraction of the norm of the gradient at the iterate. Such an approach has been consider previously in the literature; e.g., see [15, 30].

We may analyze this approach as in [22], where we note that under Assumption 2.1 with $\bar{p} \geq 1$, it follows (see [23, Lemma 1.2.2]) that there exists $H_{max} \in \mathbb{R}_{>0}$ such that $\|\nabla^2 f(x_k)\|_2 \leq H_{max}$ for all $k \in \mathbb{N}$. First, it follows by standard trust region theory, in particular related to Cauchy decrease (see [9, Thm. 6.3.1]), that

$$\begin{aligned} t_2(x_k, 0) - t_2(x_k, s_k) &\geq \frac{1}{2} \min\left\{ \frac{\|\nabla f(x_k)\|_2}{1 + \|\nabla^2 f(x_k)\|_2}, \delta_k \right\} \|\nabla f(x_k)\|_2 \\ &\geq \frac{1}{2} \min\left\{ \frac{1}{1 + H_{max}}, \nu_k \right\} \|\nabla f(x_k)\|_2^2. \end{aligned} \tag{4.2}$$

Second, using [9, Thm. 6.4.1], the discrepancy between the objective at $x_k + s_k$ and the second-order Taylor approximation value at s_k satisfies

$$|f(x_k + s_k) - t_2(x_k, s_k)| \leq H_{max} \nu_k^2 \|\nabla f(x_k)\|_2^2. \tag{4.3}$$

We may now prove that there is a uniform lower bound on the sequence $\{\nu_k\}$. Indeed, from the inequalities in (4.2) and (4.3), one finds

$$|\rho_k - 1| = \frac{|f(x_k + s_k) - t_2(x_k, s_k)|}{t_2(x_k, 0) - t_2(x_k, s_k)} \leq \frac{2 H_{max} \nu_k^2}{\min\{(1 + H_{max})^{-1}, \nu_k\}},$$

which, if $\nu_k \leq \nu_{dec} := \min\{(1+H_{max})^{-1}, (1-\eta)(2H_{max})^{-1}\}$, means that

$$|\rho_k - 1| \leq 2H_{max}\nu_k \leq 1-\eta \implies \rho_k \geq \eta.$$

Hence, by the manner in which ν_k is set for all $k \in \mathbb{N}$, this shows that

$$\nu_k \geq \nu_{min} := \min\{\underline{\nu}, \beta\nu_{dec}\} \quad \text{for all} \quad k \in \mathbb{N},$$

from which it follows that the maximum number of rejected steps between a successful step (or the initial point) and the next successful step is bounded by

$$\chi := 1 + \left\lceil \frac{\log(\nu_{min}) - \log(\overline{\nu})}{\log(\beta)} \right\rceil. \tag{4.4}$$

This analysis allows us to prove the worst-case behavior of the algorithm over $\mathcal{R}_1$. With Theorem 2.1 and (4.2), for any $k \in \mathbb{N}$ such that the step is successful, one has

$$\begin{aligned} f(x_k) - f(x_{k+1}) &\geq \eta(t_2(x_k, 0) - t_2(x_k, s_k)) \\ &\geq \frac{\eta}{2} \min\left\{\frac{1}{1+H_{max}}, \nu_{min}\right\} \|\nabla f(x_k)\|_2^2 \\ &= r_1 \eta \min\left\{\frac{1}{1+H_{max}}, \nu_{min}\right\} \Delta m_1(x_k) \\ &\geq r_1 \eta \kappa \min\left\{\frac{1}{1+H_{max}}, \nu_{min}\right\} (f(x_k) - f_{\text{ref}}). \end{aligned}$$

Since one is interested only when $f(x_k) - f_{\text{ref}} \geq \epsilon_f$ and $f(x_{k+1}) \geq f_{\text{ref}}$ (since otherwise the objective value tolerance is already satisfied or will be satisfied within one iteration), which in particular implies that the reduction in f in the bound above is at least a fraction of $f(x_k) - f_{\text{ref}}$, we may assume without loss of generality that $r_1\eta\kappa \min\left\{(1+H_{max})^{-1}, \nu_{min}\right\} \in (0,1)$. Following from here the argument in the proof of Theorem 3.1, but now accounting for the potential rejected steps between accepted steps (or from the start of the algorithm), we have proved the following.

Theorem 4.1. *Suppose that Assumption 2.1 holds with $\bar{p} \geq 1$, which implies the existence of a scalar $H_{max} \in \mathbb{R}_{>0}$ such that $\|\nabla^2 f(x_k)\|_2 \leq H_{max}$ for all $k \in \mathbb{N}$. If the iterate sequence $\{x_k\}$ is generated by the second-order trust region method with gradient-dependent radii described in this subsection, then, with $\phi := r_1\eta \min\{(1+H_{max})^{-1}, \nu_{min}\}$ and for any $\epsilon_f \in (0, \Delta f_{ref})$, the number of iterations in the set $\mathcal{R}_1 \cap \{x \in \mathbb{R}^n : f(x) - f_{ref} \geq \epsilon_f\}$ is at most*

$$\chi \left\lceil \log\left(\frac{\Delta f_{ref}}{\epsilon_f}\right) \left(\log\left(\frac{1}{1-\phi\kappa}\right)\right)^{-1} \right\rceil = \mathcal{O}\left(\log\left(\frac{\Delta f_{ref}}{\epsilon_f}\right)\right),$$

where χ defined in (4.4) bounds, independently from ϵ_f, the number of consecutive iterations that can involve the step being rejected.

In short, we have attained Theorem 4.1 for the second-order trust region described in this section since we have been able to provide a uniform bound on the number of consecutive rejected steps and, for a accepted step from x_k, we have shown that the reduction in the objective is at least proportional to $\|\nabla f(x_k)\|_2^2$, which in turn is proportional to $\Delta m_1(x_k)$. On the other hand, it is easily seen that one cannot bound the number of iterations that the algorithm might spend in $\mathcal{R}_2$, since, e.g., at any first-order stationary point (with $\|\nabla f(x_k)\|_2 = 0$) that is not second-order stationary (i.e., where $\lambda(\nabla^2 f(x_k)) < 0$), the trust region radius would be $\delta_k \leftarrow \nu_k \|\nabla f(x_k)\|_2 = 0$, causing $x_{k+1} \leftarrow x_k$ for all subsequent $k \in \mathbb{N}$.

4.2 Gradient- and Hessian-Dependent Radii

With hopes to overcome the behavior with respect to $\mathcal{R}_2$ of the method in the previous subsection, let us now consider a second-order trust region method that operates in the following manner.

- For $k = 0$, one selects $\nu_0 \in [\underline{\nu}, \overline{\nu}] \subset \mathbb{R}_{>0}$.
- For all $k \in \mathbb{N}$, one sets

$$\delta_k \leftarrow \nu_k \begin{cases} \|\nabla f(x_k)\|_2 & \text{if } \|\nabla f(x_k)\|_2^2 \geq \max\{0, -\lambda(\nabla^2 f(x_k))\}^3 \\ |\lambda(\nabla^2 f(x_k))| & \text{otherwise.} \end{cases}$$

- For all $k \in \mathbb{N}$ and some $(\eta, \beta) \in (0,1) \times (0,1)$, if

$$\rho_k := \frac{f(x_k) - f(x_k + s_k)}{t_2(x_k, 0) - t_2(x_k, s_k)} \geq \eta,$$

  then one sets $x_{k+1} \leftarrow x_k + s_k$ and $\nu_{k+1} \in [\underline{\nu}, \overline{\nu}]$ (i.e., the step is "successful"); otherwise, one sets $x_{k+1} \leftarrow x_k$ and $\nu_{k+1} \leftarrow \beta \nu_k$ (i.e., the step is "rejected").

From the analysis in §4.1, it follows that if $\|\nabla f(x_k)\|_2^2 \geq \max\{0, -\lambda(\nabla^2 f(x_k))\}^3$ for some $k \in \mathbb{N}$, then the number of iterations until the next accepted step is bounded above by χ in (4.4), and the next accepted step, occuring at $\overline{k} \geq k$, yields

$$f(x_{\overline{k}}) - f(x_{\overline{k}+1}) \geq \frac{\eta}{2} \min\left\{ \frac{1}{1 + H_{max}}, \nu_{min} \right\} \|\nabla f(x_{\overline{k}})\|_2^2.$$

On the other hand, suppose that $\|\nabla f(x_k)\|_2^2 < \max\{0, -\lambda(\nabla^2 f(x_k))\}^3$. Letting u_k represent a unit eigenvector corresponding to the leftmost eigenvalue of the Hessian of f at x_k scaled by ± 1 so that $\nabla f(x_k)^T u_k \leq 0$, it follows that

$$\begin{aligned} & \min_{\tau \geq 0} \; t_2(x_k, \tau u_k) \;\text{ s.t. }\; \|\tau u_k\|_2 \leq \delta_k = \nu_k |\lambda(\nabla^2 f(x_k))| \\ = & \min_{\tau \geq 0} \; f(x_k) + \nabla f(x_k)^T (\tau u_k) + \frac{1}{2} (\tau u_k)^T \nabla^2 f(x_k)(\tau u_k) \;\text{ s.t. }\; \tau \leq -\nu_k \lambda(\nabla^2 f(x_k)) \\ = & \min_{\tau \geq 0} \; f(x_k) + \tau \nabla f(x_k)^T u_k + \frac{1}{2} \tau^2 \lambda(\nabla^2 f(x_k)) \;\text{ s.t. }\; \tau \leq -\nu_k \lambda(\nabla^2 f(x_k)), \end{aligned}$$

which, since $\nabla f(x_k)^T u_k \leq 0$ and $\lambda(\nabla^2 f(x_k)) < 0$, shows that the minimum occurs at $\tau_k = -\nu_k \lambda(\nabla^2 f(x_k))$, yielding

$$\begin{aligned} t_2(x_k, \tau_k u_k) &= f(x_k) - \nu_k \lambda(\nabla^2 f(x_k)) \nabla f(x_k)^T u_k - \frac{1}{2} \nu_k^2 |\lambda(\nabla^2 f(x_k))|^3 \\ &\leq f(x_k) - \frac{1}{2} \nu_k^2 |\lambda(\nabla^2 f(x_k))|^3. \end{aligned}$$

Hence, the reduction in the second-order Taylor series approximation satisfies

$$t_2(x_k, 0) - t_2(x_k, s_k) \geq t_2(x_k, 0) - t_k(x_k, \tau_k u_k) \geq \frac{1}{2} \nu_k^2 |\lambda(\nabla^2 f(x_k))|^3. \tag{4.5}$$

We may now prove that there is a uniform lower bound on the sequence $\{\nu_k\}$ when $\|\nabla f(x_k)\|_2^2 < \max\{0, -\lambda(\nabla^2 f(x_k))\}^3$. Indeed, by Assumption 2.1 with $\bar{p} \geq 2$ and the inequality in (4.5), one finds that since $\|s_k\|_2 \leq \delta_k = \nu_k |\lambda(\nabla^2 f(x_k))|$ and with $\nu_k \leq \nu_{dec,2} := 3(1-\eta)/(2L_2)$ one has

$$1 - \rho_k = \frac{f(x_k + s_k) - t_2(x_k, s_k)}{t_2(x_k, 0) - t_2(x_k, s_k)} \leq \frac{2L_2 \|s_k\|_2^3}{3\nu_k^2 |\lambda(\nabla^2 f(x_k))|^3} \leq \frac{2L_2 \nu_k}{3} \leq 1 - \eta,$$

from which it follows that $\rho_k \geq \eta$. This means that $\nu_k \geq \nu_{min,2} := \min\{\underline{\nu}, \beta\nu_{dec,2}\}$ for all $k \in \mathbb{N}$ when $\|\nabla f(x_k)\|_2^2 < \max\{0, -\lambda(\nabla^2 f(x_k))\}^3$, from which it follows that, with χ from (4.4), the maximum number of rejected steps between a successful step and the next (or from the initial iteration) is bounded above by

$$\chi_2 := \max\left\{\chi, 1 + \left\lceil \frac{\log(\nu_{min,2}) - \log(\overline{\nu})}{\log(\beta)} \right\rceil\right\}. \tag{4.6}$$

The analysis above allows us to analyze the worst-case behavior of the algorithm over $\mathcal{R}_1$ and $\mathcal{R}_2$. Consider any $\epsilon_f \in (0, \Delta f_{\text{ref}})$. First, over $\mathcal{R}_1$, for any $k \in \mathbb{N}$ such that the step is successful, one finds with Theorem 2.1 that

$$\begin{aligned}
&f(x_k) - f(x_{k+1}) \\
&\geq \eta(t_2(x_k, 0) - t_2(x_k, s_k)) \\
&\geq \frac{\eta}{2}\begin{cases} \min\left\{\dfrac{1}{1+H_{max}}, \nu_{min}\right\}\|\nabla f(x_k)\|_2^2 & \text{if } \|\nabla f(x_k)\|_2^2 \geq \max\{0, -\lambda(\nabla^2 f(x_k))\}^3 \\ \nu_{min,2}|\lambda(\nabla^2 f(x_k))|^3 & \text{otherwise} \end{cases} \\
&\geq \frac{\eta}{2}\min\left\{\frac{1}{1+H_{max}}, \nu_{min}, \nu_{min,2}\right\}\|\nabla f(x_k)\|_2^2 \\
&\geq \eta r_1 \min\left\{\frac{1}{1+H_{max}}, \nu_{min}, \nu_{min,2}\right\}\Delta m_1(x_k) \\
&\geq \eta r_1 \kappa \min\left\{\frac{1}{1+H_{max}}, \nu_{min}, \nu_{min,2}\right\}(f(x_k) - f_{\text{ref}}).
\end{aligned}$$

As we have argued previously, since one is interested only when $f(x_k) - f_{\text{ref}} \geq \epsilon_f$ and $f(x_{k+1}) \geq f_{\text{ref}}$ (since otherwise the objective value tolerance is already satisfied or will be satisfied within one iteration), which in particular implies that the reduction in f in the bound above is at least a fraction of $f(x_k) - f_{\text{ref}}$, it follows with

$$\mu_1 := \eta r_1 \kappa \min\left\{\frac{1}{1+H_{max}}, \nu_{min}, \nu_{min,2}\right\} \in (0, 1)$$

that one finds

$$\begin{aligned}
&(f(x_k) - f_{\text{ref}}) - (f(x_{k+1}) - f_{\text{ref}}) \geq \mu_1(f(x_k) - f_{\text{ref}}) \\
\implies\ &(1 - \mu_1)(f(x_k) - f_{\text{ref}}) \geq (f(x_{k+1}) - f_{\text{ref}}).
\end{aligned}$$

Applying this fact repeatedly, one finds that, with χ_2 defined in (4.6), the number of iterations in the set $\mathcal{R}_1 \cap \{x \in \mathbb{R}^n : f(x) - f_{\text{ref}} \geq \epsilon_f\}$ is bounded above by

$$\chi_2 \left\lceil \frac{\log(\epsilon_f) - \log(\Delta f_{\text{ref}})}{\log(1 - \mu_1)} \right\rceil = \mathcal{O}\left(\chi_2 \log\left(\frac{\Delta f_{\text{ref}}}{\epsilon_f}\right)\right).$$

Now consider $\mathcal{R}_2$. For any $k \in \mathbb{N}$ such that the step is successful, one finds with Theorem 2.1 that the reduction in the objective satisfies

$$\begin{aligned}
&f(x_k) - f(x_{k+1}) \\
&\geq \eta(t_2(x_k, 0) - t_2(x_k, s_k)) \\
&\geq \frac{\eta}{2}\begin{cases} \min\left\{\dfrac{1}{1+H_{max}}, \nu_{min}\right\}\|\nabla f(x_k)\|_2^2 & \text{if } \|\nabla f(x_k)\|_2^2 \geq \max\{0, -\lambda(\nabla^2 f(x_k))\}^3 \\ \nu_{min,2}|\lambda(\nabla^2 f(x_k))|^3 & \text{otherwise} \end{cases} \\
&\geq \frac{\eta}{2}\min\left\{\frac{1}{1+H_{max}}, \nu_{min}, \nu_{min,2}\right\}|\lambda(\nabla^2 f(x_k))|^3 \\
&\geq 3\eta r_2^2 \min\left\{\frac{1}{1+H_{max}}, \nu_{min}, \nu_{min,2}\right\}\Delta m_2(x_k) \\
&\geq 3\eta r_2^2 \kappa \min\left\{\frac{1}{1+H_{max}}, \nu_{min}, \nu_{min,2}\right\}(f(x_k) - f_{\text{ref}}).
\end{aligned}$$

By the same reasoning as for $\mathcal{R}_1$, it follows with

$$\mu_2 := 3\eta r_2^2 \kappa \min\left\{\frac{1}{1+H_{max}}, \nu_{min}, \nu_{min,2}\right\}$$

that the number of iterations in $\mathcal{R}_2 \cap \{x \in \mathbb{R}^n : f(x) - f_{\text{ref}} \geq \epsilon_f\}$ is bounded by

$$\chi_2 \left\lceil \frac{\log(\epsilon_f) - \log(\Delta f_{\text{ref}})}{\log(1-\mu_2)} \right\rceil = \mathcal{O}\left(\log\left(\frac{\Delta f_{\text{ref}}}{\epsilon_f}\right)\right),$$

where χ_2 is a constant independent of ϵ_f. We have proved the following.

Theorem 4.2. *Suppose that Assumption 2.1 holds with $\bar{p} \geq 2$. If the iterate sequence $\{x_k\}$ is generated by the second-order trust region method with gradient- and Hessian-dependent radii, then, for any $\epsilon_f \in (0, \Delta f_{ref})$, the number of iterations in $(\mathcal{R}_1 \cup \mathcal{R}_2) \cap \{x \in \mathbb{R}^n : f(x) - f_{ref} \geq \epsilon_f\}$ is at most*

$$\mathcal{O}\left(\log\left(\frac{\Delta f_{ref}}{\epsilon_f}\right)\right).$$

4.3 Summary of Second-Order Trust Region Methods

In the previous subsections, we have presented worst-cases analyses for second-order trust region methods, the first employing a gradient-dependent strategy for setting the trust region radii and the second employing a gradient- and Hessian-dependent strategy. In particular, we have shown that the former method spends at most $\mathcal{O}(\log(\Delta f_{\text{ref}})/\epsilon_f)$ iterations in $\mathcal{R}_1$, with no guarantees with respect to $\mathcal{R}_2$, while the latter method—with extra computational effort, such as needing to compute the left-most eigenvalue of the Hessian $\nabla^2 f(x_k)$ for all $k \in \mathbb{N}$—spends at most $\mathcal{O}(\log(\Delta f_{\text{ref}})/\epsilon_f)$ iterations in $\mathcal{R}_1 \cup \mathcal{R}_2$. For both of these algorithms, our analyses relied exclusively on Cauchy decrease properties of the steps, meaning that these results can also hold for variants in which the steps are computed as inexact solutions of the trust region subproblems.

With these results in hand, one might ask: Would any reasonable second-order trust region method possess similar worst-case guarantees with a similarly straightforward analysis? The answer is no.

For example, consider a second-order trust region method in which the trust region radius is updated in the following traditional manner: for all $k \in \mathbb{N}$, one sets $\delta_{k+1} \leftarrow \beta_1 \delta_k$ for some $\beta_1 \geq 1$ if the step is accepted or sets $\delta_{k+1} \leftarrow \beta_2 \delta_k$ for some $\beta_2 \in (0,1)$ if the step is rejected. A worst-case analysis of this approach is complicated for multiple reasons. First, it may happen that δ_k is significantly less than $\nu\|\nabla f(x_k)\|_2$ for $\nu \in \mathbb{R}_{>0}$. In this case, a step might be accepted such that the reduction $f(x_k) - f(x_{k+1})$ is not sufficiently large with respect to $\Delta m_1(x_k)$, causing an insufficient reduction with respect to the reference objective error Δf_{ref}. Second, it may happen that δ_k is significantly greater than $\nu\|\nabla f(x_k)\|_2$ for some $k \in \mathbb{N}$, in which case it might not be possible to establish a uniform upper bound on the number of rejected steps between one accepted step (or the start of the run) and the next.

For another example, related to the approach proposed in §4.2, consider a second-order trust region method in which the trust region is set as

$$\delta_k \leftarrow \nu_k \max\{\|\nabla f(x_k)\|_2, |\lambda(\nabla^2 f(x_k))|\}$$

if $k = 0$ or $k \geq 1$ and $x_k \neq x_{k-1}$, or set as $\delta_k \leftarrow \beta \delta_{k-1}$ otherwise. Following the analysis as in §4.2, the number of rejected steps between one successful step (or the start of the algorithm) and the next successful step would be bounded above by χ_2 in (4.6). However, the reduction in f attained at each step would not be guaranteed to be proportional to Δm_1 over $\mathcal{R}_1$ and guaranteed to be proportional to Δm_2 over $\mathcal{R}_2$ as it is for the method in §4.2. For instance, over $\mathcal{R}_1$ and with Theorem 2.1,

$$\begin{aligned} & f(x_k) - f(x_{k+1}) \\ \geq\ & \eta(t_2(x_k, 0) - t_2(x_k, s_k)) \end{aligned}$$

$$\geq \frac{\eta}{2} \begin{cases} \min\left\{ \dfrac{1}{1+H_{max}}, \nu_{min} \right\} \|\nabla f(x_k)\|_2^2 & \text{if } \|\nabla f(x_k)\|_2 \geq \max\{0, -\lambda(\nabla^2 f(x_k))\} \\ \nu_{min,2} |\lambda(\nabla^2 f(x_k))|^3 & \text{otherwise} \end{cases}$$

with which one is only able to ensure the existence of $\xi \in (0,1)$ such that

$$f(x_k) - f(x_{k+1}) \geq \xi \min\{f(x_k) - f_{\text{ref}}, (f(x_k) - f_{\text{ref}})^{3/2}\}.$$

This shows that the algorithm possesses different behavior when $(f(x_k) - f_{\text{ref}})^{1/2} \geq 1$ than it does when the reference objective error is small, and in particular the algorithm does not attain reductions indicative of a linear rate of convergence over $\mathcal{R}_1$ when $f(x_k) - f_{\text{ref}}$ is small and the trust region radius is set as a multiple of $|\lambda(\nabla^2 f(x_k))|$. (One can see that the algorithm in §4.2 avoids these issues by employing a related, but critically different strategy for setting the trust region radii. This strategy in the algorithm in §4.2 might be extendable to higher order methods in order to achieve strong worst-case performance over $\mathcal{R}_p$ for $p \geq 3$ as well. See also the related idea in Theorem 5.1 in the next section.)

5 Function Classes of Interest

The purpose of this section is to investigate certain function classes of interest and their properties vis-à-vis the definitions of the regions introduced in §2. In fact, the definitions of these regions suggests a certain classification for functions in the context of optimization methods, namely, those for which $\mathcal{R}_1 \cup \cdots \cup \mathcal{R}_p$ for some $p \in \mathbb{N}$ characterizes the entire search space. We start this section by providing some definitions and observations along these lines. We then discuss how certain other functions of broad interest—both convex and nonconvex—relate to these function classes and to the analyses provided in §3.

5.1 "pth-order Reduction" Functions

We introduce the following.

Definition 5.1. *A function $f : \mathbb{R}^n \to \mathbb{R}$ is a* pth-order reduction function *if, for any initial point $x_0 \in \mathbb{R}^n$, Assumption 2.1 holds for $\bar{p} \geq p$ and there exists a pair $(\kappa, f_{ref}) \in (0,1) \times [f_{inf}, f(x_0)]$ such that*

$$\bigcup_{j=1}^{p} \mathcal{R}_j = \mathcal{L}.$$

Combining this definition and Theorem 3.1, we are led to the following.

Theorem 5.1. *If $f : \mathbb{R}^n \to \mathbb{R}$ is a pth-order reduction function, then for an iterate sequence $\{x_k\}$ generated by an algorithm defined by the iteration*

$$x_{k+1} \leftarrow x_k + s_i(x_k) \quad \textit{where} \quad i \leftarrow \arg \max_{j \in \{1,\ldots,p\}} \Delta m_j(x_k),$$

the number of iterations in the set $\{x \in \mathbb{R}^n : f(x) - f_{ref} \geq \epsilon_f\}$ is at most (3.2) for any given scalar $\epsilon_f \in (0, \Delta f_{ref})$.

Proof. The result follows, using similar reasoning as in the proof of Theorem 3.1, from the fact that, at any $x_k \in \mathcal{L}$ such that $f(x_k) - f_{\text{ref}} \geq \epsilon_f$, the reduction achieved in the objective is at least proportional to $f(x_k) - f_{\text{ref}}$. □

5.2 Functions Satisfying the Polyak-Łojasiewicz Condition

Let us now turn to a well-known class of functions, namely, those satisfying what is referred to as the Polyak-Łojasiewicz (PL) condition [19]. The PL condition is said to hold for a continuously differentiable function f if it has a nonempty set of minimizers with optimal value $f_{\inf} \in \mathbb{R}$ and there exists $c \in (0, \infty)$ such that

$$\frac{1}{2}\|\nabla f(x)\|_2^2 \geq c(f(x) - f_{\inf}) \quad \text{for all} \quad x \in \mathbb{R}^n. \tag{5.1}$$

It is worthwhile to mention that the PL condition does not imply that f has a unique global minimizer. However, it does ensure that any stationary point is a global minimizer. The PL condition holds when f is *c-strongly convex*, but it is also satisfied for other functions that are not convex. The interested reader should refer to [19] for a discussion on the relationship between the PL and other types of conditions that have been employed in the context of analyzing optimization methods, such as the *error bounds*, *essential strong convexity*, *weak strong convexity*, *restricted secant inequality*, and *quadratic growth* conditions. When f has a Lipschitz continuous gradient, the PL condition is the weakest of these conditions except for the quadratic growth condition. That said, PL and quadratic growth are equivalent when f is convex.

Functions with a Lipschitz continuous gradient satisfying the PL condition correspond to first-order reduction functions, as we show in the following result.

Theorem 5.2. *A continuously differentiable function f with a Lipschitz continuous gradient satisfies the PL condition if and only if it is a first-order reduction function (i.e., $\mathcal{R}_1 = \mathcal{L}$ for any $x_0 \in \mathbb{R}^n$).*

Proof. Suppose f is continuously differentiable with a Lipschitz continuous gradient and satisfies the PL condition. It follows that Assumption 2.1 holds with $\bar{p} \geq 1$, meaning that for any $x \in \mathcal{L}$ one finds with Theorem 2.1 that

$$\Delta m_1(x) = \frac{1}{2r_1}\|\nabla f(x)\|_2^2 \geq \frac{c}{r_1}(f(x) - f_{\inf}).$$

This means that, with $(\kappa, f_{\text{ref}}) = (r_1^{-1}c, f_{\inf})$, one has $\mathcal{R}_1 = \mathcal{L}$, so f is a first-order reduction function. Conversely, if f is a first-order reduction function, then it is continuously differentiable and, for any $x \in \mathbb{R}^n$, one has

$$\frac{1}{2r_1}\|\nabla f(x)\|_2^2 = \Delta m_1(x) \geq \kappa(f(x) - f_{\text{ref}}),$$

so (5.1) holds with $(c, f_{\inf}) = (\kappa r_1, f_{\text{ref}})$. □

5.3 Functions with Strict Saddle Points

Consider now the following theorem and the subsequent discussion. In particular, in this subsection, we show a relationship between second-order reduction functions and those with *strict saddle points*, which have been a popular topic in the literature in recent years.

Theorem 5.3. *Suppose f is twice-continuously differentiable with Lipschitz continuous gradient and Hessian functions over the sublevel set $\mathcal{L}$ such that, at all $x \in \mathcal{L}$ and for some $\zeta \in (0, \infty)$, it follows that*

$$\max\{\|\nabla f(x)\|_2^2, (-\lambda(\nabla^2 f(x)))^3\} \geq \zeta(f(x) - f_{inf}). \tag{5.2}$$

Then, f is a second-order reduction function.

Proof. Consider any $x \in \mathcal{L}$ such that $\|\nabla f(x)\|_2^2 \geq (-\lambda(\nabla^2 f(x)))^3$. With (5.2), at such a point, it follows with Theorem 2.1 that

$$\Delta m_1(x) = \frac{1}{2r_1}\|\nabla f(x)\|_2^2 \geq \frac{\zeta}{2r_1}(f(x) - f_{\inf}),$$

meaning that for $(\kappa, f_{\text{ref}}) = ((2r_1)^{-1}\zeta, f_{\inf})$ one has $x \in \mathcal{R}_1$. On the other hand, at any $x \in \mathcal{L}$ with $\|\nabla f(x)\|_2^2 < (-\lambda(\nabla^2 f(x)))^3$, one has with (5.2) and Theorem 2.1 that

$$\Delta m_2(x) = \frac{1}{6r_2^2}(-\lambda(\nabla^2 f(x)))^3 \geq \frac{\zeta}{6r_2^2}(f(x) - f_{\inf}).$$

Hence, with $(\kappa, f_{\text{ref}}) = ((6r_2^2)^{-1}\zeta, f_{\text{inf}})$, such a point has $x \in \mathcal{R}_2$. We have shown that all points in $\mathcal{L}$ are in $\mathcal{R}_1 \cup \mathcal{R}_2$, so f is a second-order reduction function with $\kappa = \zeta \cdot \min\{(2r_1)^{-1}, (6r_2^2)^{-1}\}$ and $f_{\text{ref}} = f_{\text{inf}}$. □

In what settings are functions satisfying (5.2) of interest? They are closely tied to functions of recent interest in the literature on nonconvex optimization, namely, functions that have strict (or ridable) saddle points. To explain this, let us first provide the following definition.

Definition 5.2. *A stationary point for a continuously differentiable function f, i.e., a point $x \in \mathbb{R}^n$ satisfying $\nabla f(x) = 0$, is a saddle point if it is not a local minimizer or a local maximizer. Such a point $x \in \mathbb{R}^n$ is a* strict saddle point *if f is twice-continuously differentiable about x and $\lambda(\nabla^2 f(x)) < 0$.*

Suppose that, corresponding to a twice-continuously differentiable f and starting point x_0, there are at most a finite number of saddle points in the set $\mathcal{L}$ and all are strict. Such a property holds for certain objective functions of interest, e.g., in problems for complete dictionary learning [28], matrix factorization [18], orthoginal tensor decomposition [16], deep learning with linear activation functions [20], and phase retrieval [29]. If, in addition, the function f has at most a finite number of local maximizers in $\mathcal{L}$, and f is locally strongly concave at any such maximizer, then we claim that $(-\lambda(\nabla^2 f(x)))^3 \geq \zeta(f(x) - f_{\text{inf}})$ for some $\zeta \in (0, \infty)$ at any x sufficiently close to any strict saddle point or local maximizer. If, further, the gradient of f is sufficiently large in norm at any x not sufficiently close to such points, then one might have a function satisfying the conditions of Theorem 5.3.

6 Conclusion

We have proposed a strategy for characterizing the worst-case performance of algorithms for solving nonconvex optimization problems. The strategy is based on characterizing problem-dependent regions of the search space that are defined according to the reductions achieved by minimizing regularized derivative models of the objective function. We have shown how this strategy leads to useful characterizations of a few algorithms, and shown that for certain objective functions this leads to a complete characterization of performance. Our strategy is not limited to first- or second-order methods; it is applicable for higher-order methods as well.

It is certainly the case that for some objective functions—even ones that are *nice* in the sense of being twice-continuously differentiable with Lipschitz continuous first- and second-order derivatives—our characterization strategy leaves gaps that might be significant. However, while our approach does not offer a complete characterization for a method employed to minimize such a function, this does not mean that one cannot adapt it to make some meaningful statements. For example, it may happen that for certain starting points, one can argue that our type of characterization applies with f_{ref} chosen *a posteriori* as the limiting value $f_{\text{ref}} = \lim_{k\to\infty} f(x_k)$. For example, such an argument might be made if the starting point is chosen in the neighborhood of a strong minimizer, as is the case in a local convergence analysis. Similarly, one might consider f_{ref} as some intermediate (not necessarily limiting) objective value in order to characterize the behavior of a method from certain starting points to make statements that, while not necessarily applying to the entire run of the optimization method, at least apply until the objective falls below f_{ref}. Overall, we propose that our characterization strategy offers useful insights into the behavior of an algorithm that are complementary to contemporary types of analyses that are more comprehensive.

We claim that our approach to analyzing complexity in the context of nonconvex optimization can be generalized or adapted to settings when Assumption 2.1 does not hold. For example, while Assumption 2.1 requires the pth-order derivatives of f to be Lipschitz continuous over $\mathcal{L}^+$ for all $p \in \{1, \ldots, \bar{p}\}$, one might consider a more general setting when these derivatives are only Hölder continuous with exponent α not necessarily equal to one. In such a setting, one can apply a generalization of our strategy, say by defining regions $\mathcal{R}_1^\alpha$, $\mathcal{R}_2^\alpha$, etc. that depend on the exponent α.

References

[1] E. G. Birgin, J. L. Gardenghi, J. M. Martínez, S. A. Santos, and Ph. L. Toint. Worst-case evaluation complexity for unconstrained nonlinear optimization using high-order regularized models. *Mathematical Programming*, 163(1):359–368, 2017.

[2] E. G. Birgin and J. M. Martínez. The use of quadratic regularization with a cubic descent condition for unconstrained optimization. *SIAM Journal on Optimization*, 27(2):1049–1074, 2017.

[3] K.-H. Borgwardt. The Average number of pivot steps required by the Simplex-Method is polynomial. *Zeitschrift für Operations Research*, 26(1):157–177, 1982.

[4] C. Cartis, N. I. M. Gould, and Ph. L. Toint. On the complexity of steepest descent, Newton's and regularized Newton's methods for nonconvex unconstrained optimization problems. *SIAM Journal on Optimization*, 20(6):2833–2852, 2010.

[5] C. Cartis, N. I. M. Gould, and Ph. L. Toint. Adaptive cubic regularisation methods for unconstrained optimization. Part I: Motivation, convergence and numerical results. *Mathematical Programming*, 127:245–295, 2011.

[6] C. Cartis, N. I. M. Gould, and Ph. L. Toint. Adaptive cubic regularisation methods for unconstrained optimization. Part II: Worst-case function- and derivative-evaluation complexity. *Mathematical Programming*, 130:295–319, 2011.

[7] C. Cartis, N. I. M. Gould, and Ph. L. Toint. Optimal Newton-type methods for nonconvex smooth optimization problems. Technical Report ERGO Technical Report 11-009, School of Mathematics, University of Edinburgh, 2011.

[8] C. Cartis, N. I. M. Gould, and Ph. L. Toint. Evaluation complexity bounds for smooth constrained nonlinear optimisation using scaled kkt conditions, high-order models and the criticality measure χ. *arXiv preprint arXiv:1705.04895*, 2017.

[9] A. R. Conn, N. I. M. Gould, and Ph. L. Toint. *Trust-Region Methods*. Society for Industrial and Applied Mathematics (SIAM), 2000.

[10] F. E. Curtis and D. P. Robinson. Exploiting negative curvature in deterministic and stochastic optimization. Technical Report 17T-003, COR@L Laboratory, Department of ISE, Lehigh University, 2017.

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

[12] F. E. Curtis, D. P. Robinson, and M. Samadi. An Inexact Regularized Newton Framework with a Worst-Case Iteration Complexity of $\mathcal{O}(\epsilon^{-3/2})$ for Nonconvex Optimization. Technical Report 17T-011, COR@L Laboratory, Department of ISE, Lehigh University, 2017.

[13] J.-P. Dussault. ARCq: a new adaptive regularization by cubics. *Optimization Methods and Software*, doi:10.1080/10556788.2017.1322080, 2017.

[14] J.-P. Dussault and D. Orban. Scalable Adaptive Cubic Regularization Methods. Technical Report G-2015-109, GERAD, 2017.

[15] J. Fan and Y. Yuan. A new trust region algorithm with trust region radius converging to zero. In *Proceedings of the 5th International Conference on Optimization: Techniques and Applications*, pages 786–794. ICOTA 2001, Hong Kong, 2001.

[16] R. Ge, F. Huang, C. Jin, and Y. Yuan. Escaping from saddle points — online stochastic gradient for tensor decomposition. In Peter Grnwald, Elad Hazan, and Satyen Kale, editors, *Proceedings of The 28th Conference on Learning Theory*, volume 40 of *Proceedings of Machine Learning Research*, pages 797–842, Paris, France, 03–06 Jul 2015. PMLR.

[17] N. I. M. Gould, M. Porcelli, and Ph. L. Toint. Updating the regularization parameter in the adaptive cubic regularization algorithm. *Computational Optimization and Applications*, 53(1):1–22, 2012.

[18] C. Jin, R. Ge, P. Netrapalli, S. M. Kakade, and M. I. Jordan. How to escape saddle points efficiently. *CoRR*, abs/1703.00887, 2017.

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

[20] K. Kawaguchi. Deep learning without poor local minima. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 586–594. Curran Associates, Inc., 2016.

[21] J. D. Lee, M. Simchowitz, M. I. Jordan, and B. Recht. Gradient descent only converges to minimizers. In Vitaly Feldman, Alexander Rakhlin, and Ohad Shamir, editors, *29th Annual Conference on Learning Theory*, volume 49 of *Proceedings of Machine Learning Research*, pages 1246–1257, Columbia University, New York, New York, USA, 23–26 Jun 2016. PMLR.

[22] Zachary Lubberts and Daniel P. Robinson. A Simple Convergence and Complexity Analysis for a Trust-Region Algorithm. (Technical Report 01-2018), 2018.

[23] Yu. Nesterov. *Introductory Lectures on Convex Optimization*, volume 87. Springer Science & Business Media, 2004.

[24] Yu. Nesterov and B. T. Polyak. Cubic regularization of Newton's method and its global performance. *Mathematical Programming*, 108(1):117–205, 2006.

[25] C. Royer and S. J. Wright. Complexity analysis of second-order line-search algorithms for smooth nonconvex optimization. *arXiv preprint 1706.03131*, 2017.

[26] S. Smale. On the average number of steps of the simplex method of linear programming. *Mathematical Programming*, 27(3):241–262, 1983.

[27] D. A. Spielman and S.-H. Teng. Smoothed analysis of algorithms: Why the simplex algorithm usually takes polynomial time. *J. ACM*, 51(3):385–463, May 2004.

[28] 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.

[29] Ju Sun, Qing Qu, and John Wright. A geometric analysis of phase retrieval. *Foundations of Computational Mathematics*, Aug 2017.

[30] Ph. L. Toint. Nonlinear stepsize control, trust regions and regularizations for unconstrained optimization. *Optimization Methods and Software*, 28(1):82–95, 2013.