Science & Technology Facilities Council

Technical Report

Adaptive cubic overestimation methods for unconstrained optimization

C Cartis N I M Gould Ph L Toint

October 1, 2007

RAL-TR-2007-007

Library and Information Services
SFTC Rutherford Appleton Laboratory
Harwell Science and Innovation Campus
Didcot
OX11 0QX
UK
Tel: +44 (0)1235 445384
Fax: +44(0)1235 446403
Email: library@rl.ac.uk

ISSN 1358-6254

RAL-TR-2007-007

Adaptive cubic overestimation methods for unconstrained optimization

Coralia Cartis[1,2,3], Nicholas I. M. Gould[2,3,4] and Philippe L. Toint[5]

ABSTRACT

An Adaptive Cubic Overestimation (ACO) algorithm for unconstrained optimization, generalizing a method due to Nesterov & Polyak (*Math. Programming* **108**, 2006, pp 177-205), is proposed. At each iteration of Nesterov & Polyak's approach, the global minimizer of a local cubic overestimator of the objective function is determined, and this ensures a significant improvement in the objective so long as the Hessian of the objective is Lipschitz continuous and its Lipschitz constant is available. The twin requirements of global model optimality and the availability of Lipschitz constants somewhat limit the applicability of such an approach, particularly for large-scale problems. However the promised powerful worst-case theoretical guarantees prompt us to investigate variants in which estimates of the required Lipschitz constant are refined and in which computationally-viable approximations to the global model-minimizer are sought. We show that the excellent global and local convergence properties and worst-case iteration complexity bounds obtained by Nesterov & Polyak are retained, and sometimes extended to a wider class of problems, by our ACO approach. Numerical experiments with small-scale test problems from the CUTEr set show superior performance of the ACO algorithm when compared to a trust-region implementation.

---

[1] School of Mathematics, University of Edinburgh,
The King's Buildings, Edinburgh, EH9 3JZ, Scotland, EU.
Email: coralia.cartis@ed.ac.uk .

[2] Computational Science and Engineering Department, Rutherford Appleton Laboratory,
Chilton, Oxfordshire, OX11 0QX, England, EU.
Email: n.i.m.gould@rl.ac.uk .
Current reports available from "http://www.numerical.rl.ac.uk/reports/reports.shtml".

[3] This work was supported by the EPSRC grant GR/S42170.

[4] Oxford University Computing Laboratory, Numerical Analysis Group, Wolfson Building,
Parks Road, Oxford, OX1 3QD, England, EU.
Email: nick.gould@comlab.ox.ac.uk .
Current reports available from "http://web.comlab.ox.ac.uk/oucl/publications/natr/index.html".

[5] Department of Mathematics, Facultés Universitaires ND de la Paix,
61, rue de Bruxelles, B-5000 Namur, Belgium, EU.
Email : philippe.toint@fundp.ac.be .
Current reports available from "http://www.fundp.ac.be/∼phtoint/pht/publications.html".

Computational Science and Engineering Department
Atlas Centre
Rutherford Appleton Laboratory
Oxfordshire OX11 0QX

October 1, 2007

1 Introduction

Trust-region [3] and line-search [7] methods are two commonly-used convergence schemes for unconstrained optimization. Although they are often used to globalise Newton-like iterations, it is not known whether their overall complexity when applied to non-convex problems is better than that achieved by the steepest descent method. Recently, Nesterov and Polyak [20] proposed a cubic regularisation scheme for Newton's method with a provably better global-complexity bound; superior bounds are known in the (star) convex and other special cases, while subsequently Nesterov [19] has proposed more sophisticated methods which further improve such bounds in the convex case. However, although Nesterov and Polyak's method is certainly implementable, the method requires at each iteration the global minimizer of a specially-structured cubic model as well as (implicit or explicit) knowledge of a global second-order Lipschitz constant. Whilst, remarkably, such cubic models may be minimized efficiently, it is nevertheless of interest to ask if similar convergence and complexity results may be established for more general methods, particularly those geared towards large-scale problems, and in the absence of Lipschitz bounds. Equally, it is important to discover whether the promise of such methods is borne out in practice. It is these issues which we shall address here.

The new method for unconstrained minimization introduced in [20] computes in each iteration, the global minimizer of a local cubic overestimator of the objective function, which gives a guaranteed improvement provided the Hessian of the objective is Lipschitz continuous. Specifically, suppose that we wish to find a local minimizer of $f : \mathbb{R}^n \to \mathbb{R}$, and that x_k is our current best estimate. Furthermore, suppose that the Hessian $\nabla_{xx} f(x)$ is globally Lipschitz continuous on $\mathbb{R}^n$ with ℓ_2-norm Lipschitz constant L. Then

$$\begin{aligned} f(x_k + s) = &\quad f(x_k) + s^T g(x_k) + \tfrac{1}{2} s^T H(x_k) s + \textstyle\int_0^1 (1-\tau) s^T [H(x_k + \tau s) - H(x_k)] s d\tau \\ \leq &\quad f(x_k) + s^T g(x_k) + \tfrac{1}{2} s^T H(x_k) s + \tfrac{1}{6} L \|s\|_2^3 \stackrel{\text{def}}{=} m_k^{NP}(s), \quad \text{for all } s \in \mathbb{R}^n, \end{aligned} \tag{1.1}$$

where we have defined $g(x) \stackrel{\text{def}}{=} \nabla_x f(x)$ and $H(x) \stackrel{\text{def}}{=} \nabla_{xx} f(x)$. Thus, so long as

$$m_k^{NP}(s_k) < m_k^{NP}(0) = f(x_k), \tag{1.2}$$

$x_{k+1} = x_k + s_k$ improves $f(x)$.

The bound (1.1) is well known, see for example [3, Thm.3.1.5]. However, the use of the model $m_k^{NP}(s)$ *for computing a step* was first considered by Griewank [16] (in an unpublished technical report) as a means for constructing affine-invariant variants of Newton's method which are globally convergent to second-order critical points. In his report, Griewank also investigates the properties of the local minimizers of $m_k^{NP}(s)$. More recently and independently, Weiser, Deuflhard and Erdmann [22] pursue similar ideas with the same motivation, elaborating on those presented in [8]. Neither of these papers provides a characterisation of the global minimizer of $m_k^{NP}(s)$ nor consider complexity issues. By contrast, Nesterov and Polyak provide a computable characterisation of the global minimizer of this model, and use this as the step s_k, thus ensuring that (1.2) is satisfied and that good complexity results may be derived. All of these contributions assume global Lipschitz continuity of the Hessian.

In view of our aims of generalizing Nesterov and Polyak's [20] scheme for practical purposes while preserving its excellent convergence and complexity properties, we consider modifying it in three important ways. Firstly, we relax the need to compute a global minimizer over $\mathbb{R}^n$. Secondly, we do not insist that $H(x)$ be Lipschitz continuous in general, and therefore introduce a dynamic positive parameter σ_k instead of the scaled Lipschitz constant[1] $\frac{1}{2}L$ in (1.1). Lastly, we allow for a symmetric approximation B_k to the local Hessian $H(x_k)$ in the cubic model on each iteration; this may be highly useful in practice. Thus, instead of (1.1), it is the more general model

$$m_k(s) \stackrel{\text{def}}{=} f(x_k) + s^T g_k + \tfrac{1}{2} s^T B_k s + \tfrac{1}{3} \sigma_k \|s\|^3, \tag{1.3}$$

[1] The factor $\frac{1}{2}$ is for later convenience.

that we employ as an approximation to f in each iteration of our Adaptive Cubic Overestimation (ACO) algorithm (see page 4). Here, and for the remainder of the paper, for brevity we write $g_k = g(x_k)$ and $\|\cdot\| = \|\cdot\|_2$.

The rules for updating the parameter σ_k in the course of the ACO algorithm are justified by analogy to trust-region methods. In such a framework, σ_k might be regarded as the reciprocal of the trust-region radius (see our comments following the proof of Theorem 3.1 and the updating rules for the trust-region radius in [3]). Thus σ_k is increased if insufficient decrease is obtained in some measure of relative objective change, but decreased or unchanged otherwise.

Since finding a global minimizer of the model $m_k(s)$ may not be essential in practice, and as doing so might be prohibitively expensive from a computational point of view, we relax this requirement by letting s_k be an approximation to such a minimizer. Initially, we only require that s_k ensures that the decrease in the model is at least as good as that provided by a suitable Cauchy point. In particular, a milder condition than the inequality in (1.1) is required for the computed step s_k to be accepted. Provided the Hessian of the objective function and the approximation B_k are bounded above on the convex hull of the iterates and for all k, respectively, we show in §2.2 that the ACO algorithm is globally convergent to first-order critical points. Furthermore, in §6.1, under the same assumptions, we obtain a worst-case complexity bound on the total number of iterations the ACO algorithm takes to drive the norm of the gradient of f below ϵ. This bound is of order ϵ^{-2}, the same as for the steepest descent method [18, p.29], which is to be expected since the Cauchy-point condition requires no more than a move in the negative gradient direction.

To improve on the performance and properties of the ACO algorithm, we further require that the step s_k globally minimizes the model (1.3) in a larger subspace. Suitable candidates include the Krylov subspaces generated by a Lanczos process or, in the limit, the whole of $\mathbb{R}^n$—recall that the Lanczos process is particularly appropriate for large-scale problems (see §7.2 and §8). Additional termination rules are specified for the inner iterations, which guarantee that the steps s_k are not too short (see Lemmas 4.7, 4.9 and 6.4). Any of these rules makes the ACO algorithm converge asymptotically at least Q-superlinearly (see Corollary 4.8 and the first remark following its proof), under appropriate assumptions but without assuming local or global Lipschitz continuity of the Hessian (Theorem 4.3). We also show that the well-known Dennis-Moré condition [6] on the Hessian approximation B_k is sufficient, and certain quasi-Newton formulae, such as BFGS, are thus appropriate. In the same context, we also show that the parameter σ_k stays bounded above and all steps s_k are eventually accepted (see Theorem 4.3). Under an asymptotic local Lipschitz assumption on $H(x)$, and slightly stronger agreement between B_k and $H(x_k)$ along s_k, Q-quadratic convergence of the iterates is shown when a specific termination criteria is employed (Corollary 4.10). We remark however that, in our numerical experiments, this rule is not the most efficient (see §8). Requiring asymptotic agreement between B_k and $H(x_k)$ (see (4.17)), without requiring Lispchitz continuity of the Hessian, we show, in a similar fashion to the analogous trust-region results, that the sequence of iterates $\{x_k\}$ is attracted to one of its limit points which is a local minimizer (Theorem 4.5). Without requiring local convexity of the objective as in the latter result, but assuming global Lipschitz continuity of the objective Hessian, we prove that any limit point of the sequence of iterates is weak second-order critical in the sense that the Hessian restricted to the subspaces of minimization is positive semidefinite in the limit (Theorem 5.4).

The steepest-descent-like complexity bounds obtained when the Cauchy condition holds, can be improved when s_k is the global minimizer of the model (1.3) in a subspace containing the gradient g_k and an appropriate termination criterion is employed. In particular, assuming $H(x)$ to be globally Lipschitz continuous, and the approximation B_k to satisfy $\|(H(x_k) - B_k)s_k\| = O(\|s_k\|^2)$, we show that the ACO algorithm has an overall worst-case iteration count of order $\epsilon^{-3/2}$ for generating $\|g(x_k)\| \leq \epsilon$ (see Corollary 6.5), and of order ϵ^{-3} for achieving approximate nonnegative curvature in a subspace containing s_k (see Corollary 6.6 and the remarks following its proof). These bounds match those proved by Nesterov and Polyak [20, §3] for their Algorithm (3.3). However, our framework, at least for the first-order results, is more general, as we allow more freedom in the choice of s_k and of B_k.

Despite the good convergence and complexity properties of the ACO algorithm, its practical efficiency

ultimately relies on the ability to exactly or approximately minimize the cubic model m_k. Though m_k is non-convex, Theorem 3.1—first proved by different means in [20]—gives a powerful characterization of its global solutions over $\mathbb{R}^n$ that can be exploited computationally as we show in §7.1. Our investigations suggest that the model can be globally minimized surprisingly efficiently, provided the factorization of the matrix B_k is (inexpensively) available. Since the latter may not be the case in large-scale optimization, we also address computing cheaper and approximate minimizers of m_k, namely, global minimizers of m_k over certain subspaces, that do not involve explicit factorizations of B_k, only matrix-vector products (see §7.2). Our approach involves using the Lanczos process to build up an orthogonal basis for the Krylov subspace formed by successively applying B_k to $g(x_k)$, and each direction s_k is the global minimizer of the model over the current Krylov subspace. It is easily shown that this technique of approximately minimizing the cubic model when employed with either of our termination criterias, is fully covered by our theoretical results. Furthermore, numerical experience with a Matlab implementation of this approach in the ACO algorithm shows this code to perform consistently better than a trust-region implementation when tested on all the small unconstrained problems problems from the CUTEr test set; see §8 and Figure 8.1 for details.

The outline of the paper is as follows. Section 2.1 introduces the ACO algorithm, while §2.2 shows it to be globally convergent to first-order critical points. Section 3.1 gives a new proof to a known characterization of the global minimizer of the cubic model over $\mathbb{R}^n$, while §3.2 defines some more general properties that are satisfied by global minimizers of m_k over subspaces of $\mathbb{R}^n$. Then §3.3 prescribes some suitable termination criterias for the inner iterations employed to minimize the cubic model approximately. Using the results in §3, we show asymptotic convergence properties of the ACO algorithm in the presence of local convexity in §4.1, while in §4.2, we prove that then, the ACO algorithm converges at least Q-superlinearly. Without assuming local convexity, §5 addresses conditions for the global convergence of the iterates to (weak) second-order critical limit points. Section 6 is devoted to a worst-case complexity analysis of the ACO algorithm, with §6.1 addressing the case when we only require the step s_k satisfies the Cauchy-point condition, and §6.2 giving improved complexity bounds when s_k minimizes the cubic model in a subspace. Section 7 addresses ways of globally minimizing the cubic model both to high accuracy (§7.1) as well as approximately using Lanczos (§7.2). We detail our numerical experiments in §8 and in Appendix A, and draw final conclusions in §9.

2 Cubic overestimation for unconstrained minimization

2.1 The method

Throughout, we assume that

AF.1 $$f \in C^2(\mathbb{R}^n). \tag{2.1}$$

The iterative method we shall consider for minimizing $f(x)$ is the Adaptive Cubic Overestimation (ACO) algorithm summarized below.

Given an estimate x_k of a critical point of f, a step s_k is computed as an approximate (global) minimizer of the model $m_k(s)$ in (1.3). The step is only required to satisfy condition (2.2), and as such may be easily determined. The step s_k is accepted and the new iterate x_{k+1} set to $x_k + s_k$ whenever (a reasonable fraction of) the predicted model decrease $f(x_k) - m_k(s_k)$ is realized by the actual decrease in the objective, $f(x_k) - f(x_k + s_k)$. This is measured by computing the ratio ρ_k in (2.4) and requiring ρ_k to be greater than a prescribed positive constant η_1 (for example, $\eta_1 = 0.1$)—we shall shortly see (Lemma 2.1) that ρ_k is well-defined whenever $g_k \neq 0$. Since the current weight σ_k has resulted in a successful step, there is no pressing reason to increase it, and indeed there may be benefits in decreasing it if good agreement between model and function are observed. By contrast, if ρ_k is smaller than η_1, we judge that the improvement in objective is insufficient—indeed there is no improvement if $\rho_k \leq 0$. If this happens, the step will be rejected and x_{k+1} left as x_k. Under these circumstances, the only recourse

Algorithm 2.1: Adaptive Cubic Overestimation (ACO).

Given x_0, $\gamma_2 \geq \gamma_1 > 1$, $1 > \eta_2 \geq \eta_1 > 0$, and $\sigma_0 > 0$, for $k = 0, 1, \ldots$ until convergence,

1. Compute a step s_k for which
$$m_k(s_k) \leq m_k(s_k^{\mathrm{C}}), \tag{2.2}$$
where the Cauchy point
$$s_k^{\mathrm{C}} = -\alpha_k^{\mathrm{C}} g_k \ \text{ and } \ \alpha_k^{\mathrm{C}} = \ \arg\min_{\alpha \in \mathbb{R}_+} \ m_k(-\alpha g_k) \tag{2.3}$$
2. Compute $f(x_k + s_k)$ and
$$\rho_k = \frac{f(x_k) - f(x_k + s_k)}{f(x_k) - m_k(s_k)}. \tag{2.4}$$
3. Set
$$x_{k+1} = \begin{cases} x_k + s_k & \text{if } \ \rho_k \geq \eta_1 \\ x_k & \text{otherwise} \end{cases}$$
4. Set
$$\sigma_{k+1} \in \begin{cases} (0, \sigma_k] & \text{if } \ \rho_k > \eta_2 & \text{[very successful iteration]} \\ [\sigma_k, \gamma_1 \sigma_k] & \text{if } \ \eta_1 \leq \rho_k \leq \eta_2 & \text{[successful iteration]} \\ [\gamma_1 \sigma_k, \gamma_2 \sigma_k] & \text{otherwise.} & \text{[unsuccessful iteration]} \end{cases} \tag{2.5}$$

available is to increase the weight σ_k prior to the next iteration with the implicit intention of reducing the size of the step.

We note that, for Lipschitz-continuous Hessians, Griewank [16], Weiser, Deuflhard and Erdmann [22] and Nesterov and Polyak's [20] all propose techniques for estimating the global Lipschitz constant L in (1.1). This is not our objective in the update (2.5) since our only concern is local overestimation.

The connection between the construction of the ACO algorithm and of trust-region methods is superficially evident in the choice of measure ρ_k and the criteria for step acceptance. At a deeper level, the parameter σ_k might be viewed as the reciprocal of the trust-region radius (see the remarks following the proof of Theorem 3.1). Thus the ways of updating σ_k in each iteration mimick those of changing the trust-region radius. Note that, as in the case of trust-region methods, finding the Cauchy point is computationally inexpensive as it is a one-dimensional minimization of a (two-piece) cubic polynomial; this involves finding roots of a quadratic polynomial and requires one Hessian-vector and three vector products.

We remark that, due to the equivalence of norms on $\mathbb{R}^n$, the ℓ_2-norm in the model $m_k(s)$ can be replaced by a more general, norm on $\mathbb{R}^n$ of the form $\|x\| \stackrel{\text{def}}{=} \sqrt{x^\top M x}$, $x \in \mathbb{R}^n$, where M is a given symmetric positive definite matrix. We may even allow for M to depend on k as long as it is uniformly positive definite and bounded as k increases, which may be relevant to preconditioning. It is easy to show that the convergence properties of the ACO algorithm established in what follows remain true in such a more general setting, although some of the constants involved change accordingly. The use of different norms may be viewed as an attempt to achieve affine invariance, an idea pursued by Griewank [16] and Weiser, Deuflhard and Erdmann [22]. Note also that regularisation terms of the form $\|s\|^\alpha$, for some $\alpha > 2$, may be employed in $m_k(s)$ instead of the cubic term and this may prove advantageous in certain circumstances (see our comments just before §6.2.1). Griewank [16] has considered just such extensions to cope with the possibility of Hölder rather than Lipschitz continuous Hessians.

Our aim now is to investigate the global convergence properties of the ACO algorithm.

2.2 Global convergence to first-order critical points

Throughout, we denote the index set of all successful iterations of the ACO algorithm by

$$\mathcal{S} \stackrel{\text{def}}{=} \{k \geq 0 : \ k \text{ successful or very successful in the sense of (2.5)}\}. \tag{2.6}$$

We first obtain a guaranteed lower bound on the decrease in f predicted from the cubic model. This also shows that the analogue of (1.2) for m_k holds, provided $g_k \neq 0$.

Lemma 2.1. Suppose that AF.1 holds and that the step s_k satisfies (2.2). Then for $k \geq 0$, we have

$$f(x_k) - m_k(s_k) \geq f(x_k) - m_k(s_k^{\text{C}}) \geq \frac{\|g_k\|^2}{6\sqrt{2}\max\left(1+\|B_k\|, 2\sqrt{\sigma_k\|g_k\|}\right)} = \frac{\|g_k\|}{6\sqrt{2}}\min\left(\frac{\|g_k\|}{1+\|B_k\|}, \frac{1}{2}\sqrt{\frac{\|g_k\|}{\sigma_k}}\right). \tag{2.7}$$

Proof. Due to (2.2) and since the equality in (2.7) is straightforward, it remains to show the second inequality in (2.7). For any $\alpha \geq 0$, using the Cauchy-Schwarz inequality, we have

$$\begin{array}{rcl} m_k(s_k^{\text{C}}) - f(x_k) & \leq & m_k(-\alpha g_k) - f(x_k) \\ & = & -\alpha\|g_k\|^2 + \frac{1}{2}\alpha^2 g_k^T B_k g_k + \frac{1}{3}\alpha^3\sigma_k\|g_k\|^3 \\ & \leq & \alpha\|g_k\|^2\left\{-1 + \frac{1}{2}\alpha\|B_k\| + \frac{1}{3}\alpha^2\sigma_k\|g_k\|\right\}. \end{array} \tag{2.8}$$

Now $m(s_k^{\text{C}}) \leq f(x_k)$ provided $-1 + \frac{1}{2}\alpha\|B_k\| + \frac{1}{3}\alpha^2\sigma_k\|g_k\| \leq 0$ and $\alpha \geq 0$, the latter two inequalities being equivalent to

$$\alpha \in [0, \overline{\alpha}_k], \ \text{ where } \overline{\alpha}_k \stackrel{\text{def}}{=} \frac{3}{2\sigma_k\|g_k\|}\left[-\frac{1}{2}\|B_k\| + \sqrt{\frac{1}{4}\|B_k\|^2 + \frac{4}{3}\sigma_k\|g_k\|}\right].$$

Furthermore, we can express $\overline{\alpha}_k$ as

$$\overline{\alpha}_k = 2\left[\frac{1}{2}\|B_k\| + \sqrt{\frac{1}{4}\|B_k\|^2 + \frac{4}{3}\sigma_k\|g_k\|}\right]^{-1}.$$

Letting

$$\theta_k \stackrel{\text{def}}{=} \left[\sqrt{2}\max\left(1+\|B_k\|, 2\sqrt{\sigma_k\|g_k\|}\right)\right]^{-1}, \tag{2.9}$$

and employing the inequalities

$$\begin{array}{rl} \sqrt{\frac{1}{4}\|B_k\|^2 + \frac{4}{3}\sigma_k\|g_k\|} & \leq \frac{1}{2}\|B_k\| + \frac{2}{\sqrt{3}}\sqrt{\sigma_k\|g_k\|} \leq 2\max\left(\frac{1}{2}\|B_k\|, \frac{2}{\sqrt{3}}\sqrt{\sigma_k\|g_k\|}\right) \\ & \leq \sqrt{2}\max\left(1+\|B_k\|, 2\sqrt{\sigma_k\|g_k\|}\right), \end{array}$$

and

$$\frac{1}{2}\|B_k\| \leq \max\left(1+\|B_k\|, 2\sqrt{\sigma_k\|g_k\|}\right),$$

it follows that $0 < \theta_k \leq \overline{\alpha}_k$. Thus substituting the value of θ_k in the last inequality in (2.8), we obtain

$$m_k(s_k^{\text{C}}) - f(x_k) \leq \frac{\|g_k\|^2}{\sqrt{2}\max\left(1+\|B_k\|, 2\sqrt{\sigma_k\|g_k\|}\right)}\left\{-1 + \frac{1}{2}\theta_k\|B_k\| + \frac{1}{3}\theta_k^2\sigma_k\|g_k\|\right\} \leq 0. \tag{2.10}$$

It now follows from the definition (2.9) of θ_k that $\theta_k\|B_k\| \leq 1$ and $\theta_k^2\sigma_k\|g_k\| \leq 1$, so that the expression in the curly brackets in (2.10) is bounded above by $(-1/6)$. This and (2.10) imply the second inequality in (2.7). □

In the convergence theory of this section, the quantity $\sqrt{\|g_k\|/\sigma_k}$ plays a role similar to that of the trust-region radius in trust-region methods (compare (2.7) above with the bound (6.3.4) in [3]).

Next we obtain a bound on the step that will be employed in the proof of Lemma 2.3.

Lemma 2.2. Suppose that AF.1 holds and that the step s_k satisfies (2.2). Then

$$\|s_k\| \leq \frac{3}{\sigma_k} \max(\|B_k\|, \sqrt{\sigma_k \|g_k\|}), \quad k \geq 0. \tag{2.11}$$

Proof. Consider

$$\begin{aligned} m_k(s) - f(x_k) &= s^T g_k + \tfrac{1}{2} s^T B_k s + \tfrac{1}{3}\sigma_k \|s\|^3 \\ &\geq -\|s\|\|g_k\| - \tfrac{1}{2}\|s\|^2 \|B_k\| + \tfrac{1}{3}\sigma_k \|s\|^3 \\ &= (\tfrac{1}{9}\sigma_k \|s\|^3 - \|s\|\|g_k\|) + (\tfrac{2}{9}\sigma_k \|s\|^3 - \tfrac{1}{2}\|s\|^2 \|B_k\|). \end{aligned}$$

But then $\frac{1}{9}\sigma_k\|s\|^3 - \|s\|\|g_k\| > 0$ if $\|s\| > 3\sqrt{\|g_k\|/\sigma_k}$, while $\frac{2}{9}\sigma_k\|s\|^3 - \frac{1}{2}\|s\|^2\|B_k\| > 0$ if $\|s\| > \frac{9}{4}\|B_k\|/\sigma_k$. Hence $m_k(s) > f(x_k)$ whenever

$$\|s\| > \frac{3}{\sigma_k} \max(\|B_k\|, \sqrt{\sigma_k \|g_k\|}).$$

But $m_k(s_k) \leq f(x_k)$ due to (2.7), and thus (2.11) holds. □

For the proof of the next lemma, and some others to follow, we need to show that, under certain conditions, a step k is very successful in the sense of (2.5). Provided $f(x_k) > m_k(s_k)$, and recalling (2.4), we have

$$\rho_k > \eta_2 \quad \Longleftrightarrow \quad r_k \stackrel{\text{def}}{=} f(x_k + s_k) - f(x_k) - \eta_2[m_k(s_k) - f(x_k)] < 0. \tag{2.12}$$

Whenever $f(x_k) > m_k(s_k)$, we can express r_k as

$$r_k = f(x_k + s_k) - m_k(s_k) + (1 - \eta_2)\left[m_k(s_k) - f(x_k)\right], \quad k \geq 0. \tag{2.13}$$

We also need to estimate the difference between the function and the model at $x_k + s_k$. A Taylor expansion of $f(x_k + s_k)$ and its agreement with the model to first-order gives

$$f(x_k + s_k) - m_k(s_k) = \frac{1}{2} s_k^\top [H(\xi_k) - B_k] s_k - \frac{\sigma_k}{3}\|s_k\|^3, \quad k \geq 0, \tag{2.14}$$

for some ξ_k on the line segment $(x_k, x_k + s_k)$.

The following assumptions will occur frequently in our results. For the function f, we assume

AF.2 $$\|H(x)\| \leq \kappa_{\text{H}}, \text{ for all } x \in X, \text{ and some } \kappa_{\text{H}} \geq 1, \tag{2.15}$$

where X is an open convex set containing all the iterates generated. For the model m_k, suppose

AM.1 $$\|B_k\| \leq \kappa_{\text{B}}, \text{ for all } k \geq 0, \text{ and some } \kappa_{\text{B}} \geq 0. \tag{2.16}$$

We are now ready to give our next result, which claims that it is always possible to make progress from a nonoptimal point ($g_k \neq 0$).

Lemma 2.3. Let AF.1–AF.2 and AM.1 hold. Also, assume that $g_k \neq 0$ and that

$$\sqrt{\sigma_k \|g_k\|} > \frac{54\sqrt{2}}{1 - \eta_2}(\kappa_{\text{H}} + \kappa_{\text{B}}) \stackrel{\text{def}}{=} \kappa_{\text{HB}}. \tag{2.17}$$

Then iteration k is very successful and

$$\sigma_{k+1} \leq \sigma_k. \tag{2.18}$$

Proof. Since $f(x_k) > m_k(s_k)$ due to $g_k \neq 0$ and (2.7), (2.12) holds. We are going to derive an upper bound on the expression (2.13) of r_k, which will be negative provided (2.17) holds.

From (2.14), we have

$$f(x_k + s_k) - m_k(s_k) \leq \tfrac{1}{2}(\kappa_{\text{H}} + \kappa_{\text{B}})\|s_k\|^2, \tag{2.19}$$

where we also employed AF.2, AM.1 and $\sigma_k \geq 0$. Now, (2.17), $\eta_2 \in (0,1)$ and $\kappa_{\text{H}} \geq 0$ imply $\sqrt{\sigma_k\|g_k\|} \geq \kappa_{\text{B}} \geq \|B_k\|$, and so the bound (2.11) becomes

$$\|s_k\| \leq 3\sqrt{\frac{\|g_k\|}{\sigma_k}}. \tag{2.20}$$

Substituting (2.20) into (2.19), we obtain

$$f(x_k + s_k) - m_k(s_k) \leq \frac{9}{2}(\kappa_{\text{H}} + \kappa_{\text{B}})\frac{\|g_k\|}{\sigma_k}. \tag{2.21}$$

Let us now evaluate the second difference in the expression (2.13) of r_k. It follows from (2.17), $\eta_2 \in (0,1)$ and $\kappa_{\text{H}} \geq 1$ that

$$2\sqrt{\sigma_k\|g_k\|} \geq 1 + \kappa_{\text{B}} \geq 1 + \|B_k\|,$$

and thus the bound (2.7) becomes

$$m_k(s_k) - f(x_k) \leq -\frac{1}{12\sqrt{2}} \cdot \frac{\|g_k\|^{3/2}}{\sqrt{\sigma_k}}. \tag{2.22}$$

Now, (2.21) and (2.22) provide an upper bound for r_k,

$$r_k \leq \frac{\|g_k\|}{\sigma_k}\left[\frac{9}{2}(\kappa_{\text{H}} + \kappa_{\text{B}}) - \frac{1-\eta_2}{12\sqrt{2}}\sqrt{\sigma_k\|g_k\|}\right], \tag{2.23}$$

which together with (2.17), implies $r_k < 0$. □

The next lemma indicates that the parameter σ_k will not blow up at nonoptimal points.

Lemma 2.4. Let AF.1–AF.2 and AM.1 hold. Also, assume that there exists a constant $\epsilon > 0$ such that $\|g_k\| \geq \epsilon$ for all k. Then

$$\sigma_k \leq \max\left(\sigma_0, \frac{\gamma_2}{\epsilon}\kappa_{\text{HB}}^2\right) \stackrel{\text{def}}{=} L_\epsilon, \quad \text{for all } k, \tag{2.24}$$

where κ_{HB} is defined in (2.17).

Proof. For any $k \geq 0$, we have the implication

$$\sigma_k > \frac{\kappa_{\text{HB}}^2}{\epsilon} \quad \Longrightarrow \quad \sigma_{k+1} \leq \sigma_k, \tag{2.25}$$

due to $\|g_k\| \geq \epsilon$, (2.17) and Lemma 2.3. Thus, when $\sigma_0 \leq \gamma_2\kappa_{\text{HB}}^2/\epsilon$, (2.25) implies $\sigma_k \leq \gamma_2\kappa_{\text{HB}}^2/\epsilon$, $k \geq 0$, where the factor γ_2 is introduced for the case when σ_k is less than $\kappa_{\text{HB}}^2/\epsilon$ and the iteration k is not very successful. Letting $k = 0$ in (2.25) gives (2.24) when $\sigma_0 \geq \gamma_2\kappa_{\text{HB}}^2/\epsilon$, since $\gamma_2 > 1$. □

Next, we show that provided there are only finitely many successful iterations, all subsequent iterates to the last of these are first-order critical points.

Lemma 2.5. Let AF.1–AF.2 and AM.1 hold. Suppose furthermore that there are only finitely many successful iterations. Then $x_k = x_*$ for all sufficiently large k and $g(x_*) = 0$.

Proof. After the last successful iterate is computed, indexed by say k_0, the construction of the algorithm implies that $x_{k_0+1} = x_{k_0+i} \stackrel{\text{def}}{=} x_*$, for all $i \geq 1$. Since all iterations $k \geq k_0 + 1$ are unsuccessful, σ_k increases by at least a fraction γ_1 so that $\sigma_k \to \infty$ as $k \to \infty$. If $\|g_{k_0+1}\| > 0$, then $\|g_k\| = \|g_{k_0+1}\| > 0$, for all $k \geq k_0 + 1$, and Lemma 2.4 implies that σ_k is bounded above, $k \geq k_0 + 1$, and we have reached a contradiction. □

We are now ready to prove the first convergence result for the ACO algorithm In particular, we show that provided f is bounded from below, either we are in the above case and $g_k = 0$ for some finite k, or there is a subsequence of (g_k) converging to zero.

Theorem 2.6. Suppose that AF.1–AF.2 and AM.1 hold. Then either

$$g_l = 0 \ \text{ for some } l \geq 0 \tag{2.26}$$

or

$$\lim_{k\to\infty} f(x_k) = -\infty \tag{2.27}$$

or

$$\liminf_{k\to\infty} \|g_k\| = 0. \tag{2.28}$$

Proof. Lemma 2.5 shows that the result is true when there are only finitely many successful iterations. Let us now assume infinitely many successful iterations occur, and recall the notation (2.6).

We also assume that

$$\|g_k\| \geq \epsilon, \ \text{ for some } \epsilon > 0 \text{ and for all } k \geq 0. \tag{2.29}$$

Let $k \in \mathcal{S}$. Then the construction of the ACO algorithm, Lemma 2.1 and AM.1 imply

$$f(x_k) - f(x_{k+1}) \geq \eta_1[f(x_k) - m_k(s_k)] \geq \frac{\eta_1}{6\sqrt{2}} \cdot \min\left(\frac{\|g_k\|^{3/2}}{2\sigma_k^{1/2}}, \frac{\|g_k\|^2}{1+\kappa_{\text{B}}}\right). \tag{2.30}$$

Substituting (2.24) and (2.29) in (2.30), we obtain

$$f(x_k) - f(x_{k+1}) \geq \frac{\eta_1}{6\sqrt{2}} \cdot \frac{\epsilon^2}{\max(2\sqrt{\epsilon L_\epsilon}, 1+\kappa_{\text{B}})} := \delta_\epsilon, \tag{2.31}$$

where L_ϵ is defined in (2.24). Summing up over all iterates from 0 to k, we deduce

$$f(x_0) - f(x_{k+1}) = \sum_{j=0, j\in\mathcal{S}}^{k} [f(x_j) - f(x_{j+1})] \geq i_k \delta_\epsilon, \tag{2.32}$$

where i_k denotes the number of successful iterations up to iteration k. Since $\mathcal{S}$ is not finite, $i_k \to \infty$ as $k \to \infty$. Relation (2.32) now implies that $\{f(x_k)\}$ is unbounded below. Conversely, if $\{f(x_k)\}$ is bounded below, then our assumption (2.29) does not hold and so $\{\|g_k\|\}$ has a subsequence converging to zero. □

Furthermore, as we show next, the whole sequence of gradients g_k converges to zero provided f is bounded from below and g_k is not zero after finitely many iterations.

Corollary 2.7. Let AF.1–AF.2 and AM.1 hold. Then either

$$g_l = 0 \text{ for some } l \geq 0 \tag{2.33}$$

or

$$\lim_{k\to\infty} f(x_k) = -\infty \tag{2.34}$$

or

$$\lim_{k\to\infty} \|g_k\| = 0. \tag{2.35}$$

Proof. Following on from the previous theorem, let us now assume that (2.33) and (2.34) do not hold. We will show that (2.35) is achieved. Let us assume that $\{f(x_k)\}$ is bounded below and that there is a subsequence of successful iterates, indexed by $\{t_i\} \subseteq \mathcal{S}$ such that

$$\|g_{t_i}\| \geq 2\epsilon, \tag{2.36}$$

for some $\epsilon > 0$ and for all i. We remark that only successful iterates need to be considered since the gradient remains constant on all the other iterates due to the construction of the algorithm, and we know that there are infinitely many successful iterations since we assumed (2.33) does not hold. The latter also implies that for each t_i, there is a first successful iteration $l_i > t_i$ such that $\|g_{l_i}\| < \epsilon$. Thus $\{l_i\} \subseteq \mathcal{S}$ and

$$\|g_k\| \geq \epsilon, \text{ for } t_i \leq k < l_i, \quad \text{and} \quad \|g_{l_i}\| < \epsilon. \tag{2.37}$$

Let

$$\mathcal{K} \stackrel{\text{def}}{=} \{k \in \mathcal{S} : t_i \leq k < l_i\}, \tag{2.38}$$

where the subsequences $\{t_i\}$ and $\{l_i\}$ were defined above. Since $\mathcal{K} \subseteq \mathcal{S}$, the construction of the ACO algorithm, AM.1 and Lemma 2.1 provide that for each $k \in \mathcal{K}$,

$$f(x_k) - f(x_{k+1}) \geq \eta_1[f(x_k) - m_k(s_k)] \geq \frac{\eta_1}{6\sqrt{2}}\|g_k\| \cdot \min\left(\frac{1}{2}\sqrt{\frac{\|g_k\|}{\sigma_k}}, \frac{\|g_k\|}{1+\kappa_{\text{B}}}\right), \tag{2.39}$$

which further becomes, by employing (2.37),

$$f(x_k) - f(x_{k+1}) \geq \frac{\eta_1\epsilon}{6\sqrt{2}} \cdot \min\left(\frac{1}{2}\sqrt{\frac{\|g_k\|}{\sigma_k}}, \frac{\epsilon}{1+\kappa_{\text{B}}}\right), \quad k \geq \mathcal{K}. \tag{2.40}$$

Since $\{f(x_k)\}$ is monotonically decreasing and bounded from below, it is convergent, and (2.40) implies

$$\frac{\sigma_k}{\|g_k\|} \to \infty, \quad k \in \mathcal{K}, \quad k \to \infty, \tag{2.41}$$

and furthermore, due to (2.37),

$$\sigma_k \to \infty, \quad k \in \mathcal{K}, \quad k \to \infty. \tag{2.42}$$

It follows from (2.40) and (2.41) that

$$\sqrt{\frac{\sigma_k}{\|g_k\|}} \geq \frac{1+\kappa_{\text{B}}}{2\epsilon}, \text{ for all } k \in \mathcal{K} \text{ sufficiently large}, \tag{2.43}$$

and thus, again from (2.40),

$$\sqrt{\frac{\|g_k\|}{\sigma_k}} \leq \frac{12\sqrt{2}}{\eta_1 \epsilon}[f(x_k) - f(x_{k+1})], \quad \text{for all } k \in \mathcal{K} \text{ sufficiently large.} \tag{2.44}$$

We have

$$\|x_{l_i} - x_{t_i}\| \leq \sum_{k=t_i, k\in\mathcal{K}}^{l_i-1} \|x_k - x_{k+1}\| = \sum_{k=t_i, k\in\mathcal{K}}^{l_i-1} \|s_k\|, \quad \text{for each } l_i \text{ and } t_i. \tag{2.45}$$

Recall now the upper bound (2.11) on $\|s_k\|$, $k \geq 0$, in Lemma 2.2. It follows from (2.37) and (2.42) that

$$\sqrt{\sigma_k \|g_k\|} \geq \kappa_{\mathrm{B}}, \quad \text{for all } k \in \mathcal{K} \text{ sufficiently large,} \tag{2.46}$$

and thus (2.11) becomes

$$\|s_k\| \leq 3\sqrt{\frac{\|g_k\|}{\sigma_k}}, \quad \text{for all } k \in \mathcal{K} \text{ sufficiently large.} \tag{2.47}$$

Now, (2.44) and (2.45) provide

$$\|x_{l_i} - x_{t_i}\| \leq 3 \sum_{k=t_i, k\in\mathcal{K}}^{l_i-1} \sqrt{\frac{\|g_k\|}{\sigma_k}} \leq \frac{36\sqrt{2}}{\eta_1 \epsilon}[f(x_{t_i}) - f(x_{l_i})], \tag{2.48}$$

for all t_i and l_i sufficiently large. Since $\{f(x_j)\}$ is convergent, $\{f(x_{t_i}) - f(x_{l_i})\}$ converges to zero as $i \to \infty$. Therefore, $\|x_{l_i} - x_{t_i}\|$ converges to zero as $i \to \infty$, and by continuity, $\|g_{l_i} - g_{t_i}\|$ tends to zero. We have reached a contradiction, since (2.36) and (2.37) imply $\|g_{l_i} - g_{t_i}\| \geq \|g_{t_i}\| - \|g_{l_i}\| \geq \epsilon$. □

From now on, we assume throughout that

$$g_k \neq 0, \quad \text{for all } k \geq 0; \tag{2.49}$$

we will discuss separately the case when $g_l = 0$ for some l (see our remarks at the end of §3.2, §5 and §6.2.2). It follows from (2.7) and (2.49) that

$$f(x_k) > m_k(s_k), \quad k \geq 0. \tag{2.50}$$

A comparison of the above results to those in §6.4 of [3] outlines the similarities of the two approaches, as well as the differences. Compare for example, Lemma 2.4, Theorem 2.6 and Corollary 2.7 to Theorems 6.4.3, 6.4.5 and 6.4.6 in [3], respectively.

3 On approximate minimizers of the model

3.1 Optimality conditions for the minimizer of m_k over $\mathbb{R}^n$

In this first subsection, we give a different proof to a fundamental result concerning necessary and sufficient optimality conditions for the *global* minimizer of the cubic model, first showed by Nesterov and Polyak [20, §5.1]. Our approach is closer in spirit to trust-region techniques, thus offering new insight into this surprising result, as well as a proper fit in the context of our paper.

We may express the derivatives of the cubic model $m_k(s)$ in (1.3) as

$$\nabla_s m_k(s) = g_k + B_k s + \lambda s \ \text{ and } \ \nabla_{ss} m_k(s) = B_k + \lambda I + \lambda \left(\frac{s}{\|s\|}\right)\left(\frac{s}{\|s\|}\right)^T, \tag{3.1}$$

where $\lambda = \sigma_k \|s\|$ and I is the n by n identity matrix.

We have the following global optimality result.

Theorem 3.1. Any s_k^* is a global minimizer of $m_k(s)$ over $\mathbb{R}^n$ if and only if it satisfies the system of equations

$$(B_k + \lambda_k^* I)s_k^* = -g_k, \tag{3.2}$$

where $\lambda_k^* = \sigma_k \|s_k^*\|$ and $B_k + \lambda_k^* I$ is positive semidefinite. If $B_k + \lambda_k^* I$ is positive definite, s_k^* is unique.

Proof. In this proof, we drop the iteration subscript k for simplicity. Firstly, let s^* be a global minimizer of $m(s)$ over $\mathbb{R}^n$. It follows from (3.1) and the first- and second-order necessary optimality conditions at s^* that

$$g + (B + \lambda^* I)s^* = 0,$$

and hence that (3.2) holds, and that

$$w^T \left(B + \lambda^* I + \lambda^* \left(\frac{s}{\|s\|} \right) \left(\frac{s}{\|s\|} \right)^T \right) w \geq 0 \tag{3.3}$$

for all vectors w.

If $s^* = 0$, (3.3) is equivalent to $\lambda^* = 0$ and B being positive semi-definite, which immediately gives the required result. Thus we need only consider $s^* \neq 0$.

There are two cases to consider. Firstly, suppose that $w^T s^* = 0$. In this case, it immediately follows from (3.3) that

$$w^T (B + \lambda^* I) w \geq 0 \ \text{ for all } \ w \ \text{ for which } \ w^T s^* = 0. \tag{3.4}$$

It thus remains to consider vectors w for which $w^T s^* \neq 0$. Since w and s_* are not orthogonal, the line $s^* + \alpha w$ intersects the ball of radius $\|s^*\|$ at two points, s^* and $u^* \neq s^*$, say, and thus

$$\|u^*\| = \|s^*\|. \tag{3.5}$$

We let $w^* = u^* - s^*$, and note that w^* is parallel to w.

Since s^* is a global minimizer, we immediately have that

$$\begin{array}{rl} 0 \leq & m(u^*) - m(s^*) \\ = & g^T(u^* - s^*) + \frac{1}{2}(u^*)^T B u^* - \frac{1}{2}(s^*)^T B s^* + \frac{\sigma}{3}(\|u^*\|^3 - \|s^*\|^3) \\ = & g^T(u^* - s^*) + \frac{1}{2}(u^*)^T B u^* - \frac{1}{2}(s^*)^T B s^*, \end{array} \tag{3.6}$$

where the last equality follows from (3.5). But (3.2) gives that

$$g^T(u^* - s^*) = (s^* - u^*)^T B s^* + \lambda^* (s^* - u^*)^T s^*. \tag{3.7}$$

In addition, (3.5) shows that

$$(s^* - u^*)^T s^* = \tfrac{1}{2}(s^*)^T s^* + \tfrac{1}{2}(u^*)^T u^* - (u^*)^T s^* = \tfrac{1}{2}(w^*)^T w^*. \tag{3.8}$$

Thus combining (3.6)–(3.7), we find that

$$\begin{array}{rl} 0 \leq & \frac{1}{2}\lambda^*(w^*)^T w^* + \frac{1}{2}(u^*)^T B u^* - \frac{1}{2}(s^*)^T B s^* + (s^*)^T B s^* - (u^*)^T B s^* \\ = & \frac{1}{2}(w^*)^T (B + \lambda^* I) w^* \end{array} \tag{3.9}$$

from which we deduce that

$$w^T (B + \lambda^* I) w \geq 0 \ \text{ for all } \ w \ \text{ for which } \ w^T s^* \neq 0. \tag{3.10}$$

Hence (3.4) and (3.10) together show that $B + \lambda^* I$ is positive semidefinite. The uniqueness of s^* when $B + \lambda^* I$ is positive definite follows immediately from (3.2). For the sufficiency implication, note that any s^* that satisfies (3.2) is a local minimizer of $m(s)$ due to (3.1). To show it is a global minimizer, assume the contrary, and so there exists a $u^* \in \mathbb{R}^n$ such that $m(u^*) < m(s_k^*)$ with $\|u^*\| \geq \|s^*\|$. A contradiction with the strict inequality above can now be derived from the first equality in (3.6), $\|u^*\| \geq \|s^*\|$, (3.8), (3.9) and (3.10). □

Note how similar this result and its proof are to those for the trust-region subproblem (see [3, Theorem 7.2.1]), for which we aim to minimize $g_k^T s + \frac{1}{2} s^T B_k s$ within an ℓ_2-norm trust region $\|s\| \leq \Delta_k$ for some "radius" $\Delta_k > 0$. Often, the global solution s_k^* of this subproblem satisfies $\|s_k^*\| = \Delta_k$. Then, recalling that s_k^* would also satisfy (3.2), we have from Theorem 3.1 that $\sigma_k = \lambda_k^*/\Delta_k$. Hence one might interpret the parameter σ_k in the ACO algorithm as inversely proportional to the trust-region radius.

In §7.1, we discuss ways of computing the global minimizer s_k^*.

3.2 Minimizing the cubic model in a subspace

The only requirement on the step s_k computed by the ACO algorithm has been that it satisfies the Cauchy condition (2.2). As we showed in §2.2, this is enough for the algorithm to converge to first-order critical points. To be able to guarantee stronger convergence properties for the ACO algorithm, further requirements need to be placed on s_k. The strongest such conditions are, of course, the first and second order (necessary) optimality conditions that s_k satisfies provided it is the (exact) global minimizer of $m_k(s)$ over $\mathbb{R}^n$ (see Theorem 3.1). This choice of s_k, however, may be in general prohibitively expensive from a computational point of view, and thus, for most (large-scale) practical purposes, (highly) inefficient (see §7.1). As in the case of trust-region methods, a much more useful approach in practice is to compute an approximate global minimizer of $m_k(s)$ by (globally) minimizing the model over a sequence of (nested) subspaces, in which each such subproblem is computationally quite inexpensive (see §7.2). Thus the conditions we require on s_k in what follows, are some derivations of first- and second-order optimality when s_k is the global minimizer of m_k over a subspace (see (3.11), (3.12) and Lemma 3.2). Then, provided each subspace includes g_k, not only do the previous results still hold, but we can prove further convergence properties (see §4.1) and deduce good complexity bounds (see §6.2) for the ACO algorithm. Furthermore, our approach and results widen the scope of the convergence and complexity analysis in [20] which addresses solely the case of the exact global minimizer of m_k over $\mathbb{R}^n$.

In what follows, we require that s_k satisfies

$$g_k^\top s_k + s_k^\top B_k s_k + \sigma_k \|s_k\|^3 = 0, \quad k \geq 0, \tag{3.11}$$

and

$$s_k^\top B_k s_k + \sigma_k \|s_k\|^3 \geq 0, \quad k \geq 0. \tag{3.12}$$

Note that (3.11) is equivalent to $\nabla_s m_k(s_k)^\top s_k = 0$, due to (3.1).

The next lemma presents some suitable choices for s_k that achieve (3.11) and (3.12).

Lemma 3.2. Suppose that s_k is the global minimizer of $m_k(s)$, for $s \in \mathcal{L}_k$, where $\mathcal{L}_k$ is a subspace of $\mathbb{R}^n$. Then s_k satisfies (3.11) and (3.12). Furthermore, letting Q_k denote any orthogonal matrix whose columns form a basis of $\mathcal{L}_k$, we have that

$$Q_k^\top B_k Q_k + \sigma_k \|s_k\| I \text{ is positive semidefinite.} \tag{3.13}$$

In particular, if s_k^* is the global minimizer of $m_k(s)$, $s \in \mathbb{R}^n$, then s_k^* achieves (3.11) and (3.12).

Proof. Let s_k be the global minimizer of m_k over some $\mathcal{L}_k$, i. e., s_k solves

$$\underset{s \in \mathcal{L}_k}{\text{minimize}} \quad m_k(s). \tag{3.14}$$

Let l denote the dimension of the subspace $\mathcal{L}_k$. Let Q_k be an orthogonal $n \times l$ matrix whose columns form a basis of $\mathcal{L}_k$. Thus $Q_k^\top Q_k = I$ and for all $s \in \mathcal{L}_k$, we have $s = Q_k u$, for some $u \in \mathbb{R}^l$. Recalling that s_k solves (3.14), and letting

$$s_k = Q_k u_k, \tag{3.15}$$

we have that u_k is the global minimizer of

$$\underset{u \in \mathbb{R}^l}{\text{minimize}} \quad m_{k,r}(u) \stackrel{\text{def}}{=} f(x_k) + (Q_k^\top g_k)^\top u + \frac{1}{2} u^\top Q_k^\top B_k Q_k u + \frac{1}{3}\sigma_k \|u\|^3, \tag{3.16}$$

where we have used the following property of the Euclidean norm when applied to orthogonal matrices,

$$\|Q_k u\| = \|u\|, \quad \text{for all } u. \tag{3.17}$$

Applying Theorem 3.1 to the reduced model $m_{k,r}$ and u_k, it follows that

$$Q_k^\top B_k Q_k u_k + \sigma_k \|u_k\| u_k = -Q_k^\top g_k,$$

and multiplying by u_k, we have

$$u_k^\top Q_k^\top B_k Q_k u_k + \sigma_k \|u_k\|^3 = -g_k^\top Q_k u_k,$$

which is the same as (3.11), due to (3.15) and (3.17). Moreover, Theorem 3.1 implies that $Q_k^\top B_k Q_k + \sigma_k \|u_k\| I$ is positive semidefinite. Due to (3.15) and (3.17), this is (3.13), and also implies

$$u_k^\top Q_k^\top B_k Q_k u_k + \sigma_k \|u_k\|^3 \geq 0,$$

which is (3.12). □

Note that the Cauchy point (2.3) satisfies (3.11) and (3.12) since it globally minimizes m_k over the subspace generated by $-g_k$. To improve the properties and performance of ACO, however, it may be necessary to minimize m_k over (increasingly) larger subspaces.

The next lemma gives a lower bound on the model decrease when (3.11) and (3.12) are satisfied.

Lemma 3.3. Suppose that s_k satisfies (3.11). Then

$$f(x_k) - m_k(s_k) = \frac{1}{2} s_k^\top B_k s_k + \frac{2}{3}\sigma_k \|s_k\|^3. \tag{3.18}$$

Additionally, if s_k also satisfies (3.12), then

$$f(x_k) - m_k(s_k) \geq \frac{1}{6}\sigma_k \|s_k\|^3. \tag{3.19}$$

Proof. Relation (3.18) can be obtained by eliminating the term $s_k^\top g_k$ from (1.3) and (3.11). It follows from (3.12) that $s_k^\top B_k s_k \geq -\sigma_k \|s_k\|^3$, which we then substitute into (3.18) and obtain (3.19). □

Requiring that s_k satisfies (3.11) may not necessarily imply (2.2), unless $s_k = -g_k$. Nevertheless, when minimizing m_k globally over successive subspaces, condition (2.2) can be easily ensured by including g_k in each of the subspaces. This is the approach we take in our implementation of the ACO algorithm,

where the subspaces generated by Lanczos method naturally include the gradient (see §7 and §8). Thus, throughout, we assume the Cauchy condition (2.2) still holds.

The assumption (2.49) provides the implication

$$s_k \text{ satisfies (3.11)} \quad \Longrightarrow \quad s_k \neq 0. \tag{3.20}$$

To see this, assume $s_k = 0$. Then (3.18) gives $f(x_k) = m_k(s_k)$. This, however, contradicts (2.50).

In the case when $g(x_k) = 0$ for some $k \geq 0$ and thus assumption (2.49) is not satisfied, we need to be more careful. If s_k minimizes m_k over a subspace $\mathcal{L}_k$ generated by the columns of some orthogonal matrix Q_k, we have

$$\text{(3.13) holds and } \lambda_{\min}(Q_k^T B_k Q_k) < 0 \quad \Longrightarrow \quad s_k \neq 0, \tag{3.21}$$

since Lemma 3.2 holds even when $g_k = 0$. But if $\lambda_{\min}(Q_k^T B_k Q_k) \geq 0$ and $g(x_k) = 0$, then $s_k = 0$ and the ACO algorithm will terminate. Hence, if our intention is to identify whether B_k is indefinite, it will be necessary to build Q_k so that $Q_k^T B_k Q_k$ predicts negative eigenvalues of B_k. This will ultimately be the case with probability one if Q_k is built as the Lanczos basis of the Krylov space $\{B_k^l v\}_{l \geq 0}$ for some random initial vector $v \neq 0$. Note that we have the implication

$$\text{(3.19), (3.21) and } \sigma_k > 0 \quad \Longrightarrow \quad \text{(2.50)}, \tag{3.22}$$

and thus the step will reduce the model.

3.3 Termination criteria for the approximate minimization of m_k

In the previous section, the bound (3.19) on the model decrease was deduced. However, for this to be useful for investigating rates of convergence and complexity bounds for the ACO algorithm, we must ensure that s_k does not become too small compared to the size of the gradient. To deduce a lower bound on $\|s_k\|$, we need to be more specific about the ACO algorithm. In particular, suitable termination criteria for the method used to minimize $m_k(s)$ need to be made precise.

Let us assume that some iterative solver is used on each (major) iteration k to approximately minimize $m_k(s)$. Let us set the termination criteria for its inner iterations i to be

$$\|\nabla_s m_k(s_{i,k})\| \leq \theta_{i,k} \|g_k\|, \tag{3.23}$$

where

$$\theta_{i,k} \stackrel{\text{def}}{=} \kappa_\theta \min(1, h_{i,k}), \tag{3.24}$$

where $s_{i,k}$ are the inner iterates generated by the solver, κ_θ is any constant in $(0, 1)$, and

$$h_{i,k} \stackrel{\text{def}}{=} h_{i,k}(\|s_{i,k}\|, \|g_k\|)$$

are positive parameters. In particular, we are interested in two choices for $h_{i,k}$, namely,

$$h_{i,k} = \|s_{i,k}\|, \quad i \geq 0, \quad k \geq 0, \tag{3.25}$$

and

$$h_{i,k} = \|g_k\|^{1/2}, \quad i \geq 0, \quad k \geq 0. \tag{3.26}$$

The first choice gives improved complexity for the ACO algorithm (see §6.2), while the second yields the best numerical performance of the algorithm in our experiments (see §8). Note that $g_k = \nabla_s m_k(0)$.

The condition (3.23) is always satisfied by any minimizer $s_{i,k}$ of m_k, since then $\nabla_s m_k(s_{i,k}) = 0$. Thus condition (3.23) can always be achieved by an iterative solver, the worst that could happen is to iterate until an exact minimizer of m_k is found. We hope in practice to terminate well before this inevitable outcome.

It follows from (3.23) and (3.24) that

TC.h $$\|\nabla_s m_k(s_k)\| \leq \theta_k \|g_k\|, \text{ where } \theta_k = \kappa_\theta \min(1, h_k), \quad k \geq 0, \tag{3.27}$$

where $h_k \stackrel{\text{def}}{=} h_{i,k} > 0$ with i being the last inner iteration. In particular, for the choice (3.25), we have

TC.s $$\|\nabla_s m_k(s_k)\| \leq \theta_k \|g_k\|, \quad \text{where} \quad \theta_k = \kappa_\theta \min(1, \|s_k\|), \quad k \geq 0, \tag{3.28}$$

while for the choice (3.26), we obtain

TC.g $$\|\nabla_s m_k(s_k)\| \leq \theta_k \|g_k\|, \quad \text{where} \quad \theta_k = \kappa_\theta \min(1, \|g_k\|^{1/2}), \quad k \geq 0. \tag{3.29}$$

The lower bounds on s_k that the criteria TC.h, TC.s and TC.g provide are given in Lemmas 4.7, 4.9 and 6.4.

4 Local convergence properties

4.1 Locally convex models

In this section, we investigate the convergence properties of the ACO algorithm in the case when the approximate Hessians B_k become positive definite asymptotically, at least along the direction s_k. Some results in this section follow closely those of §6.5 in [3].

Our main assumption in this section is that s_k satisfies (3.11). We remark that condition (3.12) is automatically achieved when B_k is positive semidefinite. Thus at present, we do not assume explicitly that s_k satisfies (3.12). Furthermore, no requirement of a termination criteria for the inner iterations is made (thus none of the definitions in §3.3 are employed in this section). Significantly, none of the results in this section requires the Hessian of the objective to be globally or locally Lipschitz continuous.

Let

$$R_k(s_k) \stackrel{\text{def}}{=} \frac{s_k^T B_k s_k}{\|s_k\|^2}, \quad k \geq 0, \tag{4.1}$$

denote the Rayleigh quotient of s_k with respect to B_k, representing the curvature of the quadratic part of the model m_k along the step. We show that if (3.11) holds, we can guarantee stronger lower bounds on the model decrease than (3.19).

Lemma 4.1. Let AF.1 hold and s_k satisfy (3.11). Then

$$f(x_k) - m_k(s_k) \geq \frac{1}{2} R_k(s_k) \|s_k\|^2, \tag{4.2}$$

where $R_k(s_k)$ is the Rayleigh quotient (4.1). In particular,

$$f(x_k) - m_k(s_k) \geq \frac{1}{2} \lambda_{\min}(B_k) \|s_k\|^2, \tag{4.3}$$

where $\lambda_{\min}(B_k)$ denotes the leftmost eigenvalue of B_k.

Proof. The bound (4.2) follows straightforwardly from (3.18) and (4.1), while for (4.3), we also employed the Rayleigh quotient inequality ([3, p.19]). □

When the Rayleigh quotient (4.1) is uniformly positive, the size of s_k is of order $\|g_k\|$, as we show next.

Lemma 4.2. Suppose that AF.1 holds and that s_k satisfies (3.11). If the Rayleigh quotient (4.1) is positive, then

$$\|s_k\| \leq \frac{1}{R_k(s_k)}\|g_k\|. \tag{4.4}$$

Furthermore, if B_k is positive definite, then

$$\|s_k\| \leq \frac{1}{\lambda_{\min}(B_k)}\|g_k\|. \tag{4.5}$$

Proof. The following relations are derived from (3.11) and the Cauchy-Schwarz inequality

$$R_k(s_k)\|s_k\|^2 \leq s_k^\top B_k s_k + \sigma_k\|s_k\|^3 = -g_k^\top s_k \leq \|g_k\| \cdot \|s_k\|.$$

The first and the last terms above give (4.5) since $s_k \neq 0$ because of (3.20), and $R_k(s_k) > 0$. The bound (4.5) follows from (4.4) and the Rayleigh quotient inequality. □

The next theorem shows that all iterations are ultimately very successful provided some further assumption on the level of resemblance between the approximate Hessians B_k and the true Hessians $H(x_k)$ holds as the iterates converge to a local minimizer. In particular, we require

AM.2 $$\frac{\|(B_k - H(x_k))s_k\|}{\|s_k\|} \to 0, \quad \text{whenever} \quad \|g_k\| \to 0. \tag{4.6}$$

The first limit in (4.6) is known as the Dennis–Moré condition [6]. It is achieved if certain Quasi-Newton techniques, such as those using the BFGS or symmetric rank one updates, are used to compute B_k [21, Chapter 8].

Theorem 4.3. Let AF.1–AF.2 and AM.1–AM.2 hold, and also let s_k satisfy (3.11), and

$$x_k \to x_*, \quad \text{as} \quad k \to \infty, \tag{4.7}$$

where $H(x_*)$ is positive definite. Then there exists $R_{\min} > 0$ such that

$$R_k(s_k) \geq R_{\min}, \quad \text{for all } k \text{ sufficiently large.} \tag{4.8}$$

Also, we have

$$\|s_k\| \leq \frac{1}{R_{\min}}\|g_k\|, \quad \text{for all } k \text{ sufficiently large.} \tag{4.9}$$

Furthermore, all iterations are eventually very successful, and σ_k is bounded from above.

Proof. Since f is continuous, the limit (4.7) implies $(f(x_k))$ is bounded below. Thus Corollary 2.7 provides that x_* is a first-order critical point and $\|g_k\| \to 0$. The latter limit and AM.2 imply

$$\frac{\|(H(x_k) - B_k)s_k\|}{\|s_k\|} \to 0, \quad k \to \infty, \tag{4.10}$$

i. e., the Dennis–Moré condition holds. Since $H(x_*)$ is positive definite, so is $H(x_k)$ for all k sufficiently large. In particular, there exists a constant $R_{\min}$ such that

$$\frac{s_k^\top H(x_k)s_k}{\|s_k\|^2} > 2R_{\min} > 0, \quad \text{for all } k \text{ sufficiently large.} \tag{4.11}$$

From (4.1), (4.10) and (4.11), we obtain that for all sufficiently large k,

$$2R_{\min}\|s_k\|^2 \leq s_k^T H(x_k)s_k = s_k^T[H(x_k) - B_k]s_k + s_k^T B_k s_k \leq [R_{\min} + R(s_k)]\,\|s_k\|^2,$$

which gives (4.8). The bound (4.9) now follows from (4.4) and (4.8).
It follows from (2.50) that the equivalence (2.12) holds. We are going to derive an upper bound on the expression (2.13) of r_k and show that it is negative for all k sufficiently large. From (2.14), we have, also since $\sigma_k \geq 0$,

$$f(x_k + s_k) - m_k(s_k) \leq \frac{1}{2}\|(H(\xi_k) - B_k)s_k\| \cdot \|s_k\|, \tag{4.12}$$

where ξ_k belongs to the line segment $(x_k, x_k + s_k)$. Relation (4.2) in Lemma 4.1, and (4.8), imply

$$f(x_k) - m_k(s_k) \geq \frac{1}{2}R_{\min}\|s_k\|^2, \text{ for all } k \text{ sufficiently large.} \tag{4.13}$$

It follows from (2.13), (4.12) and (4.13) that

$$r_k \leq \frac{1}{2}\|s_k\|^2 \left\{ \frac{\|(H(\xi_k) - B_k)s_k\|}{\|s_k\|} - (1 - \eta_2)R_{\min} \right\}, \text{ for all } k \text{ sufficiently large.} \tag{4.14}$$

We have

$$\frac{\|(H(\xi_k) - B_k)s_k\|}{\|s_k\|} \leq \|H(x_k) - H(\xi_k)\| + \frac{\|(H(x_k) - B_k)s_k\|}{\|s_k\|}, \quad k \geq 0. \tag{4.15}$$

Since $\xi_k \in (x_k, x_k + s_k)$, we have $\|\xi_k - x_k\| \leq \|s_k\|$, which together with (4.9) and $\|g_k\| \to 0$, gives $\|\xi_k - x_k\| \to 0$. This, (4.7) and $H(x)$ continuous, give $\|H(x_k) - H(\xi_k)\| \to 0$, as $k \to \infty$. It now follows from (4.10) and (4.15) that

$$\frac{\|(H(\xi_k) - B_k)s_k\|}{\|s_k\|} \to 0, \; k \to \infty.$$

We deduce from (4.9) that $\|(H(\xi_k) - B_k)s_k\| < (1 - \eta_2)R_{\min}$, for all k sufficiently large. This, together with (3.20) and (4.14), imply $r_k < 0$, for all k sufficiently large.
Since σ_k is not allowed to increase on the very successful steps of the ACO algorithm, and every k sufficiently large is very successful, σ_k is bounded from above. □

The next two theorems address conditions under which the assumption (4.7) holds.

Theorem 4.4. Suppose that AF.1–AF.2, AM.1 and (3.11) hold, and that $\{f(x_k)\}$ is bounded below. Also, assume that (x_{k_i}) is a subsequence of iterates converging to some x_* and that there exists $\underline{\lambda} > 0$ such that

$$\lambda_{\min}(B_k) \geq \underline{\lambda}, \tag{4.16}$$

whenever x_k is sufficiently close to x_*. Let $H(x_*)$ be nonsingular. Then $x_k \to x_*$, as $k \to \infty$.

Proof. The conditions of Corollary 2.7 are satisfied. Thus, since f is bounded below and its gradient is continuous, we must have $\|g_k\| \to 0$, $k \to \infty$. We deduce that $g(x_*) = 0$ and x_* is a first-order critical point. By employing (4.5) in Lemma 4.2, the proof now follows similarly to that of [3, Theorem 6.5.2]. □

We remark that the sequence of iterates (x_k) has a converging subsequence provided, for example, the level set of $f(x_0)$ is bounded.
The above theorem does not prevent the situation when the iterates converge to a critical point that is not a local minimizer. In the next theorem, besides assuming that x_* is a strict local minimizer, we require the

approximate Hessians B_k to resemble the true Hessians $H(x_k)$ whenever the iterates approach a first-order critical point, namely,

AM.3 $$\|H(x_k) - B_k\| \to 0,\ k \to \infty, \qquad \text{whenever} \quad \|g_k\| \to 0,\ k \to \infty. \tag{4.17}$$

This condition is ensured, at least from a theoretical point of view, when B_k is set to the approximation of $H(x_k)$ computed by finite differences [7, 21]. It is also satisfied when using the symmetric rank one approximation to update B_k and the steps are linearly independent [1, 2].

Theorem 4.5. Let AF.1–AF.2, AM.1, AM.3 and (3.11) hold. Let also $\{f(x_k)\}$ be bounded below. Furthermore, suppose that (x_{k_i}) is a subsequence of iterates converging to some x_* with $H(x_*)$ positive definite. Then the whole sequence of iterates (x_k) converges to x_*, all iterations are eventually very successful, and σ_k stays bounded above.

Proof. Corollary 2.7 and f bounded below provide that x_* is a first-order critical point and $\|g_k\| \to 0$. The latter limit and AM.3 imply

$$\|H(x_k) - B_k\| \to 0, \quad k \to \infty. \tag{4.18}$$

Let (k_i) index all the successful iterates x_{k_i} that converge to x_* (recall that the iterates remain constant on unsuccessful iterations). Since $H(x_*)$ is positive definite and $x_{k_i} \to x_*$, it follows from (4.18) that B_{k_i} is positive definite for all sufficiently large i, and thus there exists $\underline{\lambda} > 0$ such that (4.16) holds. Theorem 4.4 now provides that the whole sequence (x_k) converges to x_*.
The conditions of Theorem 4.3 now hold since AM.3 implies AM.2. Thus the latter part of Theorem 4.5 follows from Theorem 4.3. □

We remark that in the conditions of Theorem 4.5, B_k is positive definite asymptotically.

4.2 Asymptotic rate of convergence

In this section, the termination criteria in §3.3 are employed to show that the steps s_k do not become too small compared to the size of g_k (Lemmas 4.7 and 4.9), which then implies, in the context of Theorems 4.3 and 4.5, that the ACO algorithm is at least Q-superlinearly convergent (Corollaries 4.8 and 4.10).

Firstly, a technical result is deduced from the termination criterion TC.h.

Lemma 4.6. Let AF.1–AF.2 and TC.h hold. Then, for each $k \in \mathcal{S}$, with $\mathcal{S}$ defined in (2.6), we have

$$(1 - \kappa_\theta)\|g_{k+1}\| \leq \left\| \int_0^1 H(x_k + \tau s_k) d\tau - H(x_k) \right\| \cdot \|s_k\| + \|(H(x_k) - B_k)s_k\| + \kappa_\theta \kappa_{\text{H}} h_k \|s_k\| + \sigma_k \|s_k\|^2, \tag{4.19}$$

where $\kappa_\theta \in (0, 1)$ occurs in TC.h.

Proof. Let $k \in \mathcal{S}$, and so $g_{k+1} = g(x_k + s_k)$. Then

$$\|g_{k+1}\| \leq \|g(x_k + s_k) - \nabla_s m_k(s_k)\| + \|\nabla_s m_k(s_k)\| \leq \|g(x_k + s_k) - \nabla_s m_k(s_k)\| + \theta_k \|g_k\|, \tag{4.20}$$

where we used TC.h to derive the last inequality. We also have from Taylor's theorem and (3.1)

$$\|g(x_k + s_k) - \nabla_s m_k(s_k)\| \leq \left\| \int_0^1 [H(x_k + \tau s_k) - B_k] s_k d\tau \right\| + \sigma_k \|s_k\|^2. \tag{4.21}$$

Again from Taylor's theorem, and from AF.2, we obtain

$$\|g_k\| = \left\| g_{k+1} - \int_0^1 H(x_k + \tau s_k) s_k d\tau \right\| \leq \|g_{k+1}\| + \kappa_{\rm H} \|s_k\|. \tag{4.22}$$

Substituting (4.22) and (4.21) into (4.20), we deduce

$$(1 - \theta_k)\|g_{k+1}\| \leq \left\| \int_0^1 [H(x_k + \tau s_k) - B_k] s_k d\tau \right\| + \theta_k \kappa_{\rm H} \|s_k\| + \sigma_k \|s_k\|^2. \tag{4.23}$$

It follows from the definition of θ_k in (3.27) that $\theta_k \leq \kappa_\theta h_k$ and $\theta_k \leq \kappa_\theta$, and (4.23) becomes

$$(1 - \kappa_\theta)\|g_{k+1}\| \leq \left\| \int_0^1 [H(x_k + \tau s_k) - B_k] s_k d\tau \right\| + \kappa_\theta \kappa_{\rm H} h_k \|s_k\| + \sigma_k \|s_k\|^2. \tag{4.24}$$

The triangle inequality provides

$$\left\| \int_0^1 [H(x_k + \tau s_k) - B_k] s_k d\tau \right\| \leq \left\| \int_0^1 H(x_k + \tau s_k) d\tau - H(x_k) \right\| \cdot \|s_k\| + \|(H(x_k) - B_k) s_k\|, \tag{4.25}$$

and so (4.19) follows from (4.24). □

The next lemma establishes conditions under which the TC.h criterion provides a lower bound on s_k.

Lemma 4.7. Let AF.1–AF.2, AM.2 and the limit $x_k \to x_*$, $k \to \infty$, hold. Let TC.h be achieved with

$$h_k \to 0, \ \text{ as } k \to \infty, \ k \in \mathcal{S}. \tag{4.26}$$

Then s_k satisfies

$$\|s_k\|(d_k + \sigma_k \|s_k\|) \geq (1 - \kappa_\theta)\|g_{k+1}\| \quad \text{for all } k \in \mathcal{S}, \tag{4.27}$$

where $d_k > 0$ for all $k \geq 0$, and

$$d_k \to 0, \ \text{ as } k \to \infty, \ k \in \mathcal{S}. \tag{4.28}$$

Proof. The inequality (4.19) can be expressed as

$$(1 - \kappa_\theta)\|g_{k+1}\| \leq \left\{ \left\| \int_0^1 [H(x_k + \tau s_k) - H(x_k)] d\tau \right\| + \frac{\|(H(x_k) - B_k) s_k\|}{\|s_k\|} + \kappa_\theta \kappa_{\rm H} h_k \right\} \cdot \|s_k\| + \sigma_k \|s_k\|^2,$$

where $k \in \mathcal{S}$. Let d_k denote the term in the curly brackets multiplying $\|s_k\|$. Then $d_k > 0$ since $h_k > 0$. Furthermore, $x_k + \tau s_k \in (x_k, x_{k+1})$, for all $\tau \in (0, 1)$, and $x_k \to x_*$, imply

$$\left\| \int_0^1 [H(x_k + \tau s_k) - H(x_k)] d\tau \right\| \to 0, \ \text{ as } k \to \infty, \tag{4.29}$$

since the Hessian of f is continuous. Recalling that $\|g_k\| \to 0$ due to Corollary 2.7, it now follows from AM.2, (4.26) and (4.29) that $d_k \to 0$, as the index k of successful iterations increases. □

By employing Lemma 4.7 in the context of Theorem 4.3, we show that the ACO algorithm is asymptotically Q-superlinearly convergent.

Corollary 4.8. In addition to the conditions of Theorem 4.3, assume that TC.h holds with $h_k \to 0$, $k \to \infty$, $k \in \mathcal{S}$. Then

$$\frac{\|g_{k+1}\|}{\|g_k\|} \to 0, \quad \text{as } k \to \infty, \tag{4.30}$$

and

$$\frac{\|x_{k+1} - x_*\|}{\|x_k - x_*\|} \to 0, \quad \text{as } k \to \infty. \tag{4.31}$$

In particular, the limits (4.30) and (4.31) hold when $h_k = \|s_k\|$, $k \geq 0$, or $h_k = \|g_k\|^{1/2}$, $k \geq 0$, i. e., in the case of the termination criterias TC.s and TC.g, respectively.

Proof. Since the conditions of Lemma 4.7 hold, so does the bound (4.27). Moreover, as Theorem 4.3 gives that all iterates are eventually very successful and σ_k is bounded above, say by some σ_{sup}, (4.27) holds for all k sufficiently large and thus

$$\|s_k\|(d_k + \sigma_{sup}\|s_k\|) \geq (1 - \kappa_\theta)\|g_{k+1}\| \quad \text{for all } k \text{ sufficiently large}, \tag{4.32}$$

where $d_k > 0$ and $\kappa_\theta \in (0, 1)$. Employing the upper bound (4.9) on s_k, (4.32) becomes

$$\frac{1}{R_{\min}}\left(d_k + \frac{\sigma_{sup}}{R_{\min}}\|g_k\|\right)\|g_k\| \geq \|s_k\|(d_k + \sigma_{sup}\|s_k\|) \geq (1 - \kappa_\theta)\|g_{k+1}\|, \quad \text{for all } k \text{ sufficiently large}, \tag{4.33}$$

and further, because of (2.49),

$$\frac{\|g_{k+1}\|}{\|g_k\|} \leq \frac{R_{\min} d_k + \sigma_{sup}\|g_k\|}{R_{\min}^2(1 - \kappa_\theta)}, \quad \text{for all } k \text{ sufficiently large}. \tag{4.34}$$

The right-hand side of (4.34) tends to zero as $k \to \infty$, since

$$d_k \to 0 \ \text{ and } \ \|g_k\| \to 0, \quad \text{as } k \to \infty; \tag{4.35}$$

the first limit above comes from (4.28) and all k sufficiently large being successful, while the second limit follows from Corollary 2.7. Thus (4.30) holds. The limit (4.31) is obtained from standard Taylor expansions of g_k and g_{k+1} around x_*, and from $g(x_*) = 0$ with positive definite $H(x_*)$.
The bound (4.9) and the second limit in (4.35) imply that the choices of h_k in TC.s and TC.g converge to zero, and thus the limits (4.30) and (4.31) hold for these choices of h_k. □

Note that the limits (4.30) and (4.31) also hold if we let $h_k = \|s_k\|/\sigma_k$, $k \geq 0$, in TC.h, provided the conditions of Theorem 4.3 hold (since then, σ_k is bounded above asymptotically). See (8.3) in §8.

Note also that no assumption on the Hessian of f being globally or locally Lipschitz continuous has been imposed in Lemma 4.7 or in Corollary 4.8. Our next results, however, make a *local* Lipschitz continuity assumption on the Hessian of f in a neighbourhood of a given point x_*, i. e.,

AF.3 $\|H(x) - H(y)\| \leq L_*\|x - y\|$, for all x, y sufficiently close to x_*, and some $L_* > 0$,

and show a tighter bound on s_k than (4.27) (see Lemma 4.9), and further, Q-quadratic asymptotic convergence of the iterates (Corollary 4.10). In this context, we also slightly strengthen the condition AM.2, by requiring that B_k satisfies

AM.4 $$\|(H(x_k) - B_k)s_k\| \leq C\|s_k\|^2, \quad \text{for all } k \geq 0, \text{ and some constant } C > 0. \tag{4.36}$$

We remark that if the inequality in AM.4 holds for sufficiently large k, it also holds for all $k \geq 0$. The condition AM.4 is trivially satisfied with $C = 0$ when we set $B_k = H(x_k)$ for all $k \geq 0$. Quasi-Newton methods may still satisfy AM.4 in practice, though theoretically, only condition AM.2 can be ensured.

Lemma 4.9. Let AF.1–AF.3, AM.4 and TC.s hold. Suppose also that $x_k \to x_*$, as $k \to \infty$. If

$$\sigma_k \leq \sigma_{\max}, \quad \text{for all } k \geq 0, \tag{4.37}$$

for some $\sigma_{\max} > 0$, then s_k satisfies

$$\|s_k\| \geq \kappa_g^* \sqrt{\|g_{k+1}\|} \quad \text{for all sufficiently large } k \in \mathcal{S}, \tag{4.38}$$

where κ_g^* is the positive constant

$$\kappa_g^* \stackrel{\text{def}}{=} \sqrt{\frac{1-\kappa_\theta}{\frac{1}{2}L_* + C + \sigma_{\max} + \kappa_\theta \kappa_{\text{H}}}}. \tag{4.39}$$

Proof. The conditions of Lemma 4.6 are satisfied with $h_k = \|s_k\|$. Thus, for any $k \in \mathcal{S}$ sufficiently large, (4.19) becomes, due also to AM.4 and (4.37),

$$(1-\kappa_\theta)\|g_{k+1}\| \leq \left\| \int_0^1 [H(x_k + \tau s_k) - H(x_k)] d\tau \right\| \cdot \|s_k\| + C\|s_k\|^2 + (\sigma_{\max} + \kappa_\theta \kappa_{\text{H}})\|s_k\|^2. \tag{4.40}$$

Since $x_k \to x_*$, AF.3 and $x_k + \tau s_k$ being on the line segment (x_k, x_{k+1}) for any $\tau \in (0,1)$, imply

$$\left\| \int_0^1 [H(x_k + \tau s_k) - H(x_k)] d\tau \right\| \leq \int_0^1 \|H(x_k + \tau s_k) - H(x_k)\| d\tau \leq \tfrac{1}{2} L_* \|s_k\|, \tag{4.41}$$

for all sufficiently large $k \in \mathcal{S}$. Thus (4.40) becomes

$$(1-\kappa_\theta)\|g_{k+1}\| \leq (\tfrac{1}{2}L_* + C + \sigma_{\max} + \kappa_\theta \kappa_{\text{H}}) \|s_k\|^2, \tag{4.42}$$

which together with (4.39) provides (4.38). □

Our next result employs Lemma 4.9 to show Q-quadratic asymptotic convergence of the ACO algorithm.

Corollary 4.10. In addition to the conditions of Theorem 4.3, assume that AF.3, AM.4 and TC.s hold. Then g_k converges to zero, and x_k, to x_*, Q-quadratically, as $k \to \infty$.

Proof. Note that AM.4 implies AM.2, since (4.8) and $\|g_k\| \to 0$ give $\|s_k\| \to 0$, as $k \to \infty$. Theorem 4.3 implies that σ_k is bounded above and thus (4.37) holds. Recalling that all the iterates are eventually very successful, Lemma 4.9 now implies that

$$\|s_k\| \geq \kappa_g^* \sqrt{\|g_{k+1}\|}, \quad \text{for all } k \text{ sufficiently large}, \tag{4.43}$$

where $\kappa_g^* > 0$. It follows from (4.8) and (4.43) that

$$\frac{1}{R_{\min}} \|g_k\| \geq \|s_k\| \geq \kappa_g^* \sqrt{\|g_{k+1}\|}, \quad \text{for all } k \text{ sufficiently large}, \tag{4.44}$$

and thus

$$\frac{\|g_{k+1}\|}{\|g_k\|^2} \leq \frac{1}{R_{\min}^2 (\kappa_g^*)^2}, \quad \text{for all } k \text{ sufficiently large}, \tag{4.45}$$

and g_k converges Q-quadratically. The Q-quadratic rate of convergence of the iterates follows in a standard way, using Taylor's theorem. □

Analogues of Corollaries 4.8 and 4.10 hold in the case when the stronger conditions of Theorem 4.5 are satisfied. In particular, we require the stronger condition AM.3, instead of AM.2 or AM.4, to be achieved by B_k; then, the limit $x_k \to x_*$ is guaranteed to hold. The weaker assumption AM.2, however, makes Corollary 4.8 applicable to Quasi-Newton methods (see our remarks following (4.6)).

Note that no positive lower bound on σ_k was required for the convergence results in §2.2, §4.1 and §4.2 to hold. In particular, asymptotically, it may be desirable in implementations to let σ_k to go to zero, possibly at the same rate as $\|g_k\|$. This feature is included in our implementation of the ACO algorithm (see §8).

5 Global convergence to second-order critical points

This section addresses the convergence of the sequence of iterates to second-order critical points in a framework that does not require global or local convexity of the model or the function f at the iterates or their limit points. Then, however, we shall see that other conditions such as $H(x)$ being globally Lipschitz continuous, need to be imposed. A common assumption in this section and in §6.2 is that

$$\sigma_k \geq \sigma_{\min}, \quad k \geq 0, \tag{5.1}$$

for some $\sigma_{\min} > 0$. The first lemma gives a useful property of the steps s_k, derived from (3.19).

Lemma 5.1. Let AF.1 hold, and $\{f(x_k)\}$ be bounded below by f_{low}. Also, assume that s_k satisfies (3.11) and (3.12), and σ_k, the bound (5.1). Then, recalling (2.6), we have

$$\|s_k\| \to 0, \ \text{ as } k \to \infty, \ k \in \mathcal{S}. \tag{5.2}$$

Proof. The construction of the ACO algorithm, (2.50), the model decrease (3.19) and (5.1) give

$$f(x_k) - f(x_{k+1}) \geq \eta_1[f(x_k) - m_k(s_k)] \geq \tfrac{1}{6}\eta_1\sigma_{\min}\|s_k\|^3, \quad k \in \mathcal{S}. \tag{5.3}$$

Summing up over all iterates from 0 to k, we obtain from (5.3)

$$f(x_0) - f(x_{k+1}) \geq \frac{\eta_1}{6}\sigma_{\min} \sum_{j=0,\, j\in\mathcal{S}}^{k} \|s_j\|^3, \quad k \geq 0,$$

which further gives, together with $\{f(x_k)\}$ being bounded below,

$$\frac{6}{\eta_1\sigma_{\min}}[f(x_0) - f_{\text{low}}] \geq \sum_{j=0,\, j\in\mathcal{S}}^{k} \|s_j\|^3, \quad k \geq 0. \tag{5.4}$$

Thus the series $\sum_{j=0,\, j\in\mathcal{S}}^{\infty} \|s_j\|^3$ is convergent, and (5.2) holds. □

The next lemma shows that σ_k cannot blow up provided the objective f has a globally Lipschitz continuous Hessian, namely,

AF.4 $\|H(x) - H(y)\| \leq L\|x - y\|, \ \text{ for all } x, y \in \mathbb{R}^n, \text{ where } L > 0,$ (5.5)

and B_k and $H(x_k)$ agree along s_k in the sense of AM.3.

Lemma 5.2. Let AF.1, AF.4 and AM.4 hold. Then

$$\sigma_k \leq \max\left(\sigma_0, \tfrac{3}{2}\gamma_2(C + L)\right) \stackrel{\text{def}}{=} L_0, \ \text{ for all } k \geq 0. \tag{5.6}$$

Proof. Let $L_1 \stackrel{\rm def}{=} 3(C+L)/2$. To prove (5.6), it is enough to show the implication

$$\sigma_k \geq L_1 \quad \Longrightarrow \quad k \text{ very successful}, \tag{5.7}$$

which further gives $\sigma_{k+1} \leq \sigma_k$. We allow the factor γ_2 in L_0 for the case when σ_k is only slightly less than L_1 and k is not very successful, while the term σ_0 in (5.6) accounts for choices at start-up.
To show (5.7), we deduce from (2.14) that for each $k \geq 0$,

$$\begin{array}{ll} f(x_k+s_k) - m_k(s_k) & \leq \frac{1}{2}\|H(\xi_k) - H(x_k)\| \cdot \|s_k\|^2 + \frac{1}{2}\|(H(x_k) - B_k)s_k\| \cdot \|s_k\| - \frac{\sigma_k}{3}\|s_k\|^3, \\ & \leq \left(\frac{C+L}{2} - \frac{\sigma_k}{3}\right)\|s_k\|^3, \end{array} \tag{5.8}$$

where to obtain the second inequality, we employed AF.4, $\|\xi_k - x_k\| \leq \|s_k\|$ and AM.4. It follows from (5.8) that

$$\sigma_k \geq L_1 \quad \Longrightarrow \quad f(x_k+s_k) \leq m_k(s_k). \tag{5.9}$$

The second inequality in (5.9) and (2.50) imply that the ratio (2.4) satisfies $\rho_k \geq 1$ and so $\rho_k > \eta_2$, for any $\eta_2 \in (0,1)$. Thus the step k is very successful. □

Next, we show that all the limit points of the sequence of Rayleigh quotients of B_k and of $H(x_k)$ at successful steps s_k are nonnegative.

Theorem 5.3. Let AF.1, AF.4 and AM.4 hold, and $\{f(x_k)\}$ be bounded below by $f_{\rm low}$. Also, assume that s_k satisfies (3.11) and (3.12), and σ_k, (5.1). Then, recalling (4.1), we have

$$\liminf_{k \in \mathcal{S},\, k \to \infty} R_k(s_k) \geq 0 \qquad \text{and} \qquad \liminf_{k \in \mathcal{S},\, k \to \infty} \frac{s_k^T H(x_k) s_k}{\|s_k\|^2} \geq 0. \tag{5.10}$$

Proof. For all $k \geq 0$ such that $R_k(s_k) < 0$, (3.12), (3.20) and (5.6) imply

$$L_0\|s_k\| \geq \sigma_k\|s_k\| \geq -R_k(s_k) = |R_k(s_k)|.$$

Now (5.2) implies the limit $R_k(s_k) \to 0$, $k \in \mathcal{S}$ and $k \to \infty$, for all $R_k(s_k) < 0$. This proves the first limit in (5.10). The second inequality in (5.10) now follows from the first, (5.2) and the inequalities

$$R_k(s_k) \leq \frac{\|[H(x_k) - B_k]s_k\|}{\|s_k\|} + \frac{s_k^T H(x_k) s_k}{\|s_k\|^2} \leq C\|s_k\| + \frac{s_k^T H(x_k) s_k}{\|s_k\|^2},\ k \geq 0,$$

where we employed AM.4 to obtain the last inequality. □

The next theorem gives conditions which ensure that the limit points of the sequence of iterates are second order critical points. Beforehand, we remark a useful property concerning the Hessian $H(x_k)$ and its approximation B_k. Given any matrix Q_k with orthogonal columns, [10, Corollary 8.1.6] provides the first inequality below

$$|\lambda_{\min}(Q_k^T H(x_k) Q_k) - \lambda_{\min}(Q_k^T B_k Q_k)| \leq \|Q_k^T[H(x_k) - B_k]Q_k\| \leq \sqrt{n}\|H(x_k) - B_k\|,\ k \geq 0, \tag{5.11}$$

while the second inequality above employs $\|Q_k^T\| \leq \sqrt{n}$ and $\|Q_k\| = 1$.

Theorem 5.4. Let AF.1, AF.4 and AM.4 hold. Assume that $\{f(x_k)\}$ is bounded below by $f_{\low}$, and that σ_k satisfies (5.1). Also, let s_k be a global minimizer of m_k over a subspace $\mathcal{L}_k$, and let Q_k be any orthogonal matrix whose columns form a basis of $\mathcal{L}_k$. Then any subsequence of negative leftmost eigenvalues $\{\lambda_{\min}(Q_k^T B_k Q_k)\}$ converges to zero as $k \to \infty$, $k \in \mathcal{S}$, and thus

$$\liminf_{k \in \mathcal{S},\, k \to \infty} \lambda_{\min}(Q_k^T B_k Q_k) \geq 0. \tag{5.12}$$

Additionally, assume that AF.2, AM.1 and AM.3 hold. Then any subsequence of negative leftmost eigenvalues $\{\lambda_{\min}(Q_k^T H(x_k) Q_k)\}$ converges to zero as $k \to \infty$, $k \in \mathcal{S}$, and thus

$$\liminf_{k \in \mathcal{S},\, k \to \infty} \lambda_{\min}(Q_k^T H(x_k) Q_k) \geq 0. \tag{5.13}$$

Furthermore, if Q_k becomes a full orthogonal basis of $\mathbb{R}^n$ as $k \to \infty$, $k \in \mathcal{S}$, then, provided it exists, any limit point of the sequence of iterates $\{x_k\}$ is second-order critical.

Proof. For all $k \geq 0$ such that $\lambda_{\min}(Q_k^T B_k Q_k) < 0$, we employ Lemma 3.2, in particular, (3.13), and also (5.6), to obtain

$$L_0 \|s_k\| \geq \sigma_k \|s_k\| \geq -\lambda_{\min}(Q_k^T B_k Q_k) = |\lambda_{\min}(Q_k^T B_k Q_k)|.$$

Now (5.2) implies the limit

$$\lambda_{\min}(Q_k^T B_k Q_k) \to 0, \quad k \in \mathcal{S} \text{ and } k \to \infty, \text{ for all } \lambda_{\min}(Q_k^T B_k Q_k) < 0, \tag{5.14}$$

which gives (5.12).
Assume now that AF.2, AM.1 and AM.3 holds. Then AF.2, AM.1, $\{f(x_k)\}$ bounded below, (2.49) and Corollary 2.7 give that

$$\|g_k\| \to 0, \quad k \to \infty, \tag{5.15}$$

so that the first limit in AM.3 holds, i. e.,

$$\|H(x_k) - B_k\| \to 0, \quad k \to \infty. \tag{5.16}$$

We deduce from (5.11) that for all $k \geq 0$ with $\lambda_{\min}(Q_k^T H(x_k) Q_k) < 0$, we have

$$\begin{aligned} 0 \leq -\lambda_{\min}(Q_k^T H(x_k) Q_k) &\leq \sqrt{n}\|H(x_k) - B_k\| - \lambda_{\min}(Q_k^T B_k Q_k) \\ &\leq \sqrt{n}\|H(x_k) - B_k\| + \max(0, -\lambda_{\min}(Q_k^T B_k Q_k)). \end{aligned} \tag{5.17}$$

It follows from (5.14) and (5.16) that the right-hand side of (5.17) converges to zero as $k \in \mathcal{S} \to \infty$, and so $\lambda_{\min}(Q_k^T H(x_k) Q_k) \to 0$, $k \in \mathcal{S} \to \infty$ with $\lambda_{\min}(Q_k^T H(x_k) Q_k) < 0$. This gives (5.13).
Assume now that there exists x_* such that $x_k \to x_*$, $k \in \mathcal{K}$ and $k \to \infty$. Then (5.15) and the arguments that give (5.15) imply $\|g(x_*)\| = 0$ and $\mathcal{K} \subseteq \mathcal{S}$, where the set inclusion also uses the fact that the iterates remain constant on unsuccessful iterations. Employing AF.1, we have $H(x_k) \to H(x_*)$, $k \in \mathcal{K}$, $k \to \infty$. Since the set of orthogonal matrices is compact and Q_k becomes full-dimensional as $k \to \infty$, $k \in \mathcal{K}$, any limit point, say Q, of $\{Q_k\}_{k \in \mathcal{K}}$ is a full orthogonal basis of $\mathbb{R}^n$. Due to similarity and from (5.13), we now have $\lambda_{\min}(H(x_*)) = \lambda_{\min}(Q^T H(x_*) Q) \geq 0$, and so x_* is second-order critical. □

In our implementation of the ACO algorithm, though we minimize m_k only in certain subspaces, our particular approach (see §7.2) implies that ever more accurate Ritz approximations to the extreme eigenvalues of B_k are computed provided the gradient is not orthogonal to any eigenvector of B_k [11]. In other words, except for the latter case, we expect that the orthogonal bases of the generated subspaces become full-dimensional asymptotically, and so Theorem 5.4 implies that the solutions we obtain will be second-order critical in the limit.

When $Q_k = I$ and $B_k = H(x_k)$ for all $k \geq 0$, the last part of Theorem 5.4 is essentially [20, Theorem 2].

Encountering zero gradient values. Recall the discussion in the last paragraph of §3.2, where we assume that there exists $k \geq 0$ such that $g_k = 0$ and thus (2.49) does not hold. Then (3.21) provides $s_k \neq 0$ and (2.50) holds. These two relations imply that Lemmas 5.1 and 5.2 continue to hold even when some of the iterates have zero gradients (and the ACO algorithm continues to iterate to try to attain second-order criticality in the subspaces). Employing these Lemmas and the conditions of Corollary 5.4, the limit (5.12) can be shown as before since the value of the gradient was irrelevant in its derivation. To ensure (5.13), we further assume, in addition to the requirements of Corollary 5.4, that

$$B_k = H_k \text{ for all } k \text{ for which } g_k = 0. \tag{5.18}$$

The proof of (5.13) follows as before, except that if there are infinitely many k_l such that

$$g_{k_l} = 0 \text{ and } \lambda_{\min}(Q_{k_l}^\top H(x_{k_l}) Q_{k_l}) < 0, \tag{5.19}$$

then (5.13) and (5.18) give $\liminf_{k_l \to \infty, k_l \in \mathcal{S}} \lambda_{\min}(Q_{k_l}^\top H(x_{k_l}) Q_{k_l}) \geq 0$. Note that the ACO algorithm ultimately moves away from iterates satisfying (5.19): since σ_k is bounded above as in (5.6), the ACO algorithm cannot take an infinite number of unsuccessful steps at x_{k_l} (when σ_k is increased by a fixed fraction).
The last statement of Corollary 5.4 also holds in this case provided Q_k is full-dimensional also when $g_k = 0$.

6 Worst-case iteration complexity bounds

In this section, we present worst-case complexity bounds on the number of successful and unsuccessful iterations required by the ACO algorithm to generate an iterate within ϵ of first- and second-order optimality. In the first subsection, the complexity bounds are obtained under the general and weak assumptions of the convergence analysis in §2.2. In particular, the overall bound we obtain for the ACO algorithm to reach approximate first-order optimality is $\mathcal{O}(\epsilon^{-2})$ (see Corollary 6.3), the same as for the steepest descent method [18, p.29]. This is to be expected because the only requirement on the direction s_k in this case is the Cauchy condition (2.2), which implies no more than a move along the steepest descent direction. Strengthening the conditions on s_k to be (3.11), (3.12) and a global variant of (4.38), we deduce in §6.2, that the ACO algorithm has a worst-case iteration complexity of order $\epsilon^{-3/2}$ for generating $\|g_k\| \leq \epsilon$ (see Corollary 6.5), and of order ϵ^{-3} for approximately driving the negative curvature in B_k along s_k, to zero (Corollary 6.6; see also the remarks following its proof). We remark that the latter bounds require also that f has Lipschitz continuous Hessian and that the approximate Hessians B_k satisfy AM.4.

Firstly, let us present a generic worst-case result regarding the number of unsuccessful iterations that occur up to any given iteration. Given any $j \geq 0$, denote the iteration index sets

$$\mathcal{S}_j \stackrel{\text{def}}{=} \{k \leq j : k \text{ successful or very successful}\} \quad \text{and} \quad \mathcal{U}_j \stackrel{\text{def}}{=} \{i \leq j : i \text{ unsuccessful}\}, \tag{6.1}$$

which form a partition of $\{0, \ldots, j\}$. Let $|\mathcal{S}_j|$ and $|\mathcal{U}_j|$ denote their respective cardinalities. In this context, we require that on each very successful iteration $k \in \mathcal{S}_j$, σ_{k+1} is chosen such that

$$\sigma_{k+1} \geq \gamma_3 \sigma_k, \text{ for some } \gamma_3 \in (0, 1]. \tag{6.2}$$

The condition (6.2) is a weaker requirement than (5.1) since it allows $\{\sigma_k\}$ to converge to zero on very successful iterations (but no faster than $\{\gamma_3^k\}$).

Theorem 6.1. For any fixed $j \geq 0$, let $\mathcal{S}_j$ and $\mathcal{U}_j$ be defined in (6.1). Assume that (6.2) holds and let $\overline{\sigma} > 0$ be such that

$$\sigma_k \leq \overline{\sigma}, \quad \text{for all } k \leq j. \tag{6.3}$$

Then

$$|\mathcal{U}_j| \leq \left\lceil -\frac{\log \gamma_3}{\log \gamma_1} |\mathcal{S}_j| + \frac{1}{\log \gamma_1} \log \left(\frac{\overline{\sigma}}{\sigma_0} \right) \right\rceil. \tag{6.4}$$

In particular, if σ_k satisfies (5.1), then it also achieves (6.2) with $\gamma_3 = \sigma_{\min}/\overline{\sigma}$, and we have that

$$|\mathcal{U}_j| \leq \left\lceil (|\mathcal{S}_j| + 1) \frac{1}{\log \gamma_1} \log \left(\frac{\overline{\sigma}}{\sigma_{\min}} \right) \right\rceil. \tag{6.5}$$

Proof. It follows from the construction of the ACO algorithm and from (6.2) that

$$\gamma_3 \sigma_k \leq \sigma_{k+1}, \quad \text{for all } k \in \mathcal{S}_j,$$

and

$$\gamma_1 \sigma_i \leq \sigma_{i+1}, \quad \text{for all } i \in \mathcal{U}_j.$$

Thus we deduce inductively

$$\sigma_0 \gamma_3^{|\mathcal{S}_j|} \gamma_1^{|\mathcal{U}_j|} \leq \sigma_j. \tag{6.6}$$

We further obtain from (6.3) and (6.6) that $|\mathcal{S}_j| \log \gamma_3 + |\mathcal{U}_j| \log \gamma_1 \leq \log (\overline{\sigma}/\sigma_0)$, which gives (6.4), recalling that $\gamma_1 > 1$ and that $|\mathcal{U}_j|$ is an integer. If (5.1) holds, then it implies, together with (6.3), that (6.2) is satisfied with $\gamma_3 = \sigma_{\min}/\overline{\sigma} \in (0, 1]$. The bound (6.5) now follows from (6.4) and $\sigma_0 \geq \sigma_{\min}$. □

Let $F_k \stackrel{\text{def}}{=} F(x_k, g_k, B_k, H_k) \geq 0$, $k \geq 0$, be some measure of optimality related to our problem of minimizing f. For example, for first-order optimality, we may let $F_k = \|g_k\|$, $k \geq 0$. Given any $\epsilon > 0$, and recalling (2.6), let

$$\mathcal{S}_{\mathrm{F}}^{\epsilon} \stackrel{\text{def}}{=} \{k \in \mathcal{S} : F_k > \epsilon\}, \tag{6.7}$$

and let $|\mathcal{S}_{\mathrm{F}}^{\epsilon}|$ denote its cardinality. To allow also for the case when an upper bound on the entire $|\mathcal{S}_{\mathrm{F}}^{\epsilon}|$ cannot be provided (see Corollary 6.3), we introduce a generic index set $\mathcal{S}_o$ such that

$$\mathcal{S}_o \subseteq \mathcal{S}_{\mathrm{F}}^{\epsilon}, \tag{6.8}$$

and denote its cardinality by $|\mathcal{S}_o|$. The next theorem gives an upper bound on $|\mathcal{S}_o|$.

Theorem 6.2. Let $\{f(x_k)\}$ be bounded below by f_{low}. Given any $\epsilon > 0$, let $\mathcal{S}_{\mathrm{F}}^{\epsilon}$ and $\mathcal{S}_o$ be defined in (6.7) and (6.8), respectively. Suppose that the successful iterates x_k generated by the ACO algorithm have the property that

$$f(x_k) - m_k(s_k) \geq \alpha \epsilon^p, \quad \text{for all } k \in \mathcal{S}_o, \tag{6.9}$$

where α is a positive constant independent of k and ϵ, and $p > 0$. Then

$$|\mathcal{S}_o| \leq \lceil \kappa_p \epsilon^{-p} \rceil, \tag{6.10}$$

where $\kappa_p \stackrel{\text{def}}{=} (f(x_0) - f_{\text{low}})/(\eta_1 \alpha)$.

Proof. It follows from (2.4) and (6.9) that

$$f(x_k) - f(x_{k+1}) \geq \eta_1 \alpha \epsilon^p, \text{ for all } k \in \mathcal{S}_o. \tag{6.11}$$

The construction of the ACO algorithm implies that the iterates remain unchanged over unsuccessful iterations. Furthermore, from (2.7), we have $f(x_k) \geq f(x_{k+1})$, for all $k \geq 0$. Thus summing up (6.11) over all iterates $k \in \mathcal{S}_o$, with say $j_m \leq \infty$ as the largest index, we deduce

$$f(x_0) - f(x_{j_m}) = \sum_{k=0, k\in\mathcal{S}}^{j_m-1} [f(x_k) - f(x_{k+1})] \geq \sum_{k=0, k\in\mathcal{S}_o}^{j_m-1} [f(x_k) - f(x_{k+1})] \geq |\mathcal{S}_o| \eta_1 \alpha \epsilon^p. \tag{6.12}$$

Recalling that $\{f(x_k)\}$ is bounded below, we further obtain from (6.12) that $j_m < \infty$ and that

$$|\mathcal{S}_o| \leq \frac{1}{\eta_1 \alpha \epsilon^p} (f(x_0) - f_{\text{low}}),$$

which immediately gives (6.10) since $|\mathcal{S}_o|$ must be an integer. □

If (6.9) holds with $\mathcal{S}_o = \mathcal{S}_{\text{F}}^{\epsilon}$, then (6.10) gives an upper bound on the total number of successful iterations with $F_k > \epsilon$ that occur. In particular, it implies that the ACO algorithm takes at most $\lceil \kappa_p \epsilon^{-p} \rceil$ successful iterations to generate an iterate k such that $F_{k+1} \leq \epsilon$.

In the next two subsections, we give conditions under which (6.9) holds with $F_k = \|g_k\|$ for $p = 2$ and $p = 3/2$. The conditions for the former value of p are more general, while the complexity for the latter p is better.

6.1 An iteration complexity bound based on the Cauchy condition

The complexity analysis in this section is built upon the global convergence results in §2.2. In particular, the conditions of the results are such that Corollary 2.7 applies, and so $\|g_k\| \to 0$ as $k \to \infty$. Estimating the number of iterations performed in order to drive the gradient norm below a desired accuracy is the motivation for revisiting the latter result.

Next we show that the conditions of Theorem 6.2 are satisfied with $F_k = \|g_k\|$, which provides an upper bound on the number of successful iterations. To bound the number of unsuccessful iterations, we then employ Theorem 6.1. Finally, we combine the two bounds to deduce one on the total number of iterations.

Corollary 6.3. Let AF.1–AF.2 and AM.1 hold, and $\{f(x_k)\}$ be bounded below by f_{low}. Given any $\epsilon \in (0, 1]$, assume that $\|g_0\| > \epsilon$ and let $j_1 < \infty$ be the first iteration such that $\|g_{j_1+1}\| \leq \epsilon$. Then the ACO algorithm takes at most

$$\text{L}_1^{\text{s}} \stackrel{\text{def}}{=} \lceil \kappa_{\text{C}}^{\text{s}} \epsilon^{-2} \rceil \tag{6.13}$$

successful iterations to generate $\|g_{j_1+1}\| \leq \epsilon$, where

$$\kappa_{\text{C}}^{\text{s}} \stackrel{\text{def}}{=} (f(x_0) - f_{\text{low}})/(\eta_1 \alpha_{\text{C}}), \quad \alpha_{\text{C}} \stackrel{\text{def}}{=} [6\sqrt{2} \max(1 + \kappa_{\text{B}}, 2 \max(\sqrt{\sigma_0}, \kappa_{\text{HB}} \sqrt{\gamma_2}))]^{-1} \tag{6.14}$$

and κ_{HB} is defined in (2.17) Additionally, assume that on each very successful iteration k, σ_{k+1} is chosen such that (6.2) is satisfied. Then

$$j_1 \leq \lceil \kappa_{\text{C}} \epsilon^{-2} \rceil \stackrel{\text{def}}{=} \text{L}_1, \tag{6.15}$$

and so the ACO algorithm takes at most L_1 (successful and unsuccessful) iterations to generate $\|g_{j_1+1}\| \leq \epsilon$,

where

$$\kappa_{\rm C} \stackrel{\rm def}{=} \left(1 - \frac{\log\gamma_3}{\log\gamma_1}\right)\kappa_{\rm C}^s + \kappa_{\rm C}^u, \quad \kappa_{\rm C}^u \stackrel{\rm def}{=} \frac{1}{\log\gamma_1}\max\left(1, \frac{\gamma_2\kappa_{\rm HB}^2}{\sigma_0}\right) \tag{6.16}$$

and $\kappa_{\rm C}^s$ is defined in (6.14).

Proof. The definition of j_1 in the statement of the Corollary is equivalent to

$$\|g_k\| > \epsilon, \ \text{ for all } k = 0, \ldots, j_1, \quad \text{and} \qquad \|g_{j_1+1}\| \leq \epsilon. \tag{6.17}$$

The first ingredient we need is a version of Lemma 2.4 that does not assume that $\|g_k\| > \epsilon$ on all iterations $k \geq 0$. In particular, assuming (6.17), we argue similarly to the proof of Lemma 2.4 that the implication (2.25) holds for any $k \in \{0, \ldots, j_1\}$. The remaining argument in the proof of Lemma 2.4 now provides the bound

$$\sigma_k \leq \max\left(\sigma_0, \frac{\gamma_2}{\epsilon}\kappa_{\rm HB}^2\right), \ \text{ for all } k = 0, \ldots, j_1. \tag{6.18}$$

The first and last terms in (2.7), together with AM.1 and (6.18), give

$$f(x_k) - m_k(s_k) \geq \alpha_{\rm C}\epsilon^2, \ \text{ for all } k = 0, \ldots, j_1, \tag{6.19}$$

where $\alpha_{\rm C}$ is defined in (6.14). Letting $j = j_1$ in (6.1), Theorem 6.2 with $F_k = \|g_k\|$, $\mathcal{S}_{\rm F}^\epsilon = \{k \in \mathcal{S} : \|g_k\| > \epsilon\}$, $\mathcal{S}_o = \mathcal{S}_{j_1}$ and $p = 2$ yields the complexity bound

$$|\mathcal{S}_{j_1}| \leq {\rm L}_1^{\rm s}, \tag{6.20}$$

with ${\rm L}_1^{\rm s}$ defined in (6.13), which proves the first part of the Corollary.
Let us now give an upper bound on the number of unsuccessful iterations that occur up to j_1. It follows from (6.18) and $\epsilon \leq 1$ that we may let $\overline{\sigma} \stackrel{\rm def}{=} \max\left(\sigma_0, \gamma_2\kappa_{\rm HB}^2\right)/\epsilon$ and $j = j_1$ in Theorem 6.1. Then (6.4), the inequality $\log(\overline{\sigma}/\sigma_0) \leq \overline{\sigma}/\sigma_0$ and the bound (6.20) imply that

$$|\mathcal{U}_{j_1}| \leq \left\lceil -\frac{\log\gamma_3}{\log\gamma_1}{\rm L}_1^{\rm s} + \frac{\kappa_{\rm C}^u}{\epsilon}\right\rceil, \tag{6.21}$$

where $\mathcal{U}_{j_1}$ is (6.1) with $j = j_1$ and $\kappa_{\rm C}^u$ is defined in (6.16).
Since $j_1 = |\mathcal{S}_{j_1}| + |\mathcal{U}_{j_1}|$, the bound (6.15) is the sum of the upper bounds (6.13) and (6.21) on the number of consecutive successful and unsuccessful iterations k with $\|g_k\| > \epsilon$ that occur. □

We remark (again) that the complexity bound (6.15) is of the same order as that for the steepest descent method [18, p.29]. This is to be expected because of the (only) requirement (2.2) that we imposed on the step, which implies no more than a move along the steepest descent direction.

Similar complexity results for trust-region methods are given in [14, 15].

6.2 An iteration complexity bound when minimizing the model in a subspace

When s_k satisfies (3.11), (3.12) and the condition (6.22) below, a better complexity bound can be proved for the ACO algorithm. In particular, the overall iteration complexity bound is $\mathcal{O}(\epsilon^{-3/2})$ for first-order optimality within ϵ, and $\mathcal{O}(\epsilon^{-3})$, for approximate second-order conditions in a subspace containing s_k. These bounds match those proved by Nesterov and Polyak for their Algorithm (3.3) [20, p.185]. As in [20], we also require f to have a globally Lipschitz continuous Hessian. We allow more freedom in the cubic model, however, since B_k does not have to be the exact Hessian, as long as it satisfies AM.4.

In view of the global complexity analysis to follow, we would like to obtain a tighter bound on the model decrease than in (2.7). For that, we use the bound (3.19) and a lower bound on s_k to be deduced

in the next lemma. There, the upper bound (5.6) on σ_k is employed to show that (4.38) holds globally, for all successful iterations.

Lemma 6.4. Let AF.1–AF.2, AF.4, AM.4 and TC.s hold. Then s_k satisfies

$$\|s_k\| \geq \kappa_g \sqrt{\|g_{k+1}\|} \text{ for all successful iterations } k, \tag{6.22}$$

where κ_g is the positive constant

$$\kappa_g \stackrel{\text{def}}{=} \sqrt{\frac{1-\kappa_\theta}{\frac{1}{2}L + C + L_0 + \kappa_\theta \kappa_{\text{H}}}} \tag{6.23}$$

and κ_θ is defined in (3.28) and L_0, in (5.6).

Proof. The conditions of Lemma 5.2 are satisfied, and so the bound (5.6) on σ_k holds. The proof of (6.22) follows similarly to that of Lemma 4.9, by letting $\sigma_{\max} = L_0$ and $L_* = L$, and recalling that we are now in a non-asymptotic regime. □

We are now ready to give improved complexity bounds for the ACO algorithm.

Corollary 6.5. Let AF.1–AF.2, AF.4, AM.1, AM.4 and TC.s hold, and $\{f(x_k)\}$ be bounded below by f_{low}. Let s_k satisfy (3.11) and (3.12), and σ_k be bounded below as in (5.1). Let $\epsilon > 0$. Then the total number of successful iterations with

$$\min\left(\|g_k\|, \|g_{k+1}\|\right) > \epsilon \tag{6.24}$$

that occur is at most

$$\tilde{\text{L}}_1^s \stackrel{\text{def}}{=} \left\lceil \kappa_{\text{S}}^s \epsilon^{-3/2} \right\rceil, \tag{6.25}$$

where

$$\kappa_{\text{S}}^s \stackrel{\text{def}}{=} (f(x_0) - f_{\text{low}})/(\eta_1 \alpha_{\text{S}}), \quad \alpha_{\text{S}} \stackrel{\text{def}}{=} (\sigma_{\min} \kappa_g^3)/6 \tag{6.26}$$

and κ_g is defined in (6.23). Assuming that (6.24) holds at $k = 0$, the ACO algorithm takes at most $\tilde{\text{L}}_1^{\text{s}} + 1$ successful iterations to generate a (first) iterate, say l_1, with $\|g_{l_1+1}\| \leq \epsilon$.
Furthermore, when $\epsilon \leq 1$, we have

$$l_1 \leq \left\lceil \kappa_{\text{S}} \epsilon^{-3/2} \right\rceil \stackrel{\text{def}}{=} \tilde{\text{L}}_1, \tag{6.27}$$

and so the ACO algorithm takes at most $\tilde{\text{L}}_1$ (successful and unsuccessful) iterations to generate $\|g_{l_1+1}\| \leq \epsilon$, where

$$\kappa_{\text{S}} \stackrel{\text{def}}{=} (1 + \kappa_{\text{S}}^u)(2 + \kappa_{\text{S}}^s) \quad \text{and} \quad \kappa_{\text{S}}^u \stackrel{\text{def}}{=} \log(L_0/\sigma_{\min})/\log \gamma_1, \tag{6.28}$$

with L_0 defined in (5.6) and κ_{S}^s, in (6.26).

Proof. Let

$$\mathcal{S}_g^\epsilon \stackrel{\text{def}}{=} \{k \in \mathcal{S} : \min\left(\|g_k\|, \|g_{k+1}\|\right) > \epsilon\}, \tag{6.29}$$

and let $|\mathcal{S}_g^\epsilon|$ denote its cardinality. It follows from (3.19), (5.1), (6.22) and (6.29) that

$$f(x_k) - m_k(s_k) \geq \alpha_{\text{S}} \epsilon^{3/2}, \text{ for all } k \in \mathcal{S}_g^\epsilon, \tag{6.30}$$

where α_{S} is defined in (6.26). Letting $F_k = \min(\|g_k\|, \|g_{k+1}\|)$, $\mathcal{S}_{\mathrm{F}}^{\epsilon} = \mathcal{S}_o = \mathcal{S}_g^{\epsilon}$ and $p = 3/2$ in Theorem 6.2, we deduce that $|\mathcal{S}_g^{\epsilon}| \leq \tilde{\mathrm{L}}_1^{\mathrm{s}}$, with $\tilde{\mathrm{L}}_1^{\mathrm{s}}$ defined in (6.25). This proves the first part of the Corollary and, assuming that (6.24) holds with $k = 0$, it also implies the bound

$$|\mathcal{S}_{l_+}| \leq \tilde{\mathrm{L}}_1^{\mathrm{s}}, \tag{6.31}$$

where $\mathcal{S}_{l_+}$ is (6.1) with $j = l_+$ and l_+ is the first iterate such that (6.24) does not hold at $l_+ + 1$. Thus $\|g_k\| > \epsilon$, for all $k = 0, \ldots, (l_+ + 1)$ and $\|g_{l_+ + 2}\| \leq \epsilon$. Recalling the definition of l_1 in the statement of the Corollary, it follows that $\mathcal{S}_{l_1} \setminus \{l_1\} = \mathcal{S}_{l_+}$, where $\mathcal{S}_{l_1}$ is (6.1) with $j = l_1$. From (6.31), we now have

$$|\mathcal{S}_{l_1}| \leq \tilde{\mathrm{L}}_1^{\mathrm{s}} + 1. \tag{6.32}$$

A bound on the number of unsuccessful iterations up to l_1 follows from (6.32) and from (6.5) in Theorem 6.1 with $j = l_1$ and $\overline{\sigma} = L_0$, where L_0 is provided by (5.6) in Lemma 5.2. Thus we have

$$|\mathcal{U}_{l_1}| \leq \left\lceil (2 + \tilde{\mathrm{L}}_1^{\mathrm{s}}) \kappa_{\mathrm{S}}^{u} \right\rceil, \tag{6.33}$$

where $\mathcal{U}_{l_1}$ is (6.1) with $j = l_1$ and κ_{S}^{u} is defined in (6.28). Since $l_1 = |\mathcal{S}_{l_1}| + |\mathcal{U}_{l_1}|$, the upper bound (6.27) is the sum of (6.32) and (6.33), where we also employ the expression (6.25) of $\tilde{\mathrm{L}}_1^{s}$. □

Note that we may replace the cubic term $\sigma_k \|s\|^3/3$ in $m_k(s)$ by $\sigma_k \|s\|^{\alpha}/\alpha$, for some $\alpha > 2$. Let us further assume that then, we also replace AM.4 by the condition $\|(H(x_k) - B_k)s_k\| \leq C\|s_k\|^{\alpha-1}$, and AF.4 by $(\alpha - 2)-$Hölder continuity of $H(x)$, i. e., there exists $C_{\mathrm{H}} > 0$ such that

$$\|H(x) - H(y)\| \leq C_{\mathrm{H}} \|x - y\|^{\alpha-2}, \text{ for all } x, y \in \mathbb{R}^n.$$

In these conditions and using similar arguments as for $\alpha = 3$, one can show that

$$l_\alpha \leq \lceil \kappa_\alpha \epsilon^{-\alpha/(\alpha-1)} \rceil,$$

where l_α is a (first) iteration such that $\|g_{l_\alpha+1}\| \leq \epsilon$, $\epsilon \in (0, 1)$ and $\kappa_\alpha > 0$ is a constant independent of ϵ. Thus, when $\alpha \in (2, 3)$, the resulting variants of the ACO algorithm have better worst-case iteration complexity than the steepest descent method under weaker assumptions on $H(x)$ and B_k than Lipchitz continuity and AM.4, respectively. When $\alpha > 3$, the complexity of the ACO α-variants is better than the $\mathcal{O}(\epsilon^{-3/2})$ of the ACO algorithm, but the result applies to a smaller class of functions than those with Lipschitz continuous Hessians.

6.2.1 A complexity bound for achieving approximate second-order optimality in a subspace

The next corollary addresses the complexity of achieving approximate nonnegative curvature in the Hessian approximation B_k along s_k and in a subspace. Note that the approach in §2.2 and in §6.1, when we require at least as much model decrease as given by the Cauchy point, is not expected to provide second-order optimality of the iterates asymptotically as it is, essentially, steepest descent method. The framework in §6.2, however, when we globally minimize the model over subspaces that may even equal $\mathbb{R}^n$ asymptotically, does provide second-order optimality information, at least in these subspaces, as shown in §5. We now analyse the global complexity of reaching within ϵ of the limit (5.12) in Theorem 5.4.

Corollary 6.6. Let AF.1–AF.2, AF.4, AM.1, AM.4 and TC.s hold. Let $\{f(x_k)\}$ be bounded below by f_{low} and σ_k, as in (5.1). Let s_k be the global minimizer of $m_k(s)$ over a subspace $\mathcal{L}_k$ that is generated by the columns of an orthogonal matrix Q_k and let $\lambda_{\min}(Q_k^\top B_k Q_k)$ denote the leftmost eigenvalue of $Q_k^\top B_k Q_k$. Then, given any $\epsilon > 0$, the total number of successful iterations with negative curvature

$$-\lambda_{\min}(Q_k^\top B_k Q_k) > \epsilon \tag{6.34}$$

that occur is at most

$$\mathrm{L}_2^{\mathrm{s}} \stackrel{\text{def}}{=} \lceil \kappa_{\text{curv}} \epsilon^{-3} \rceil, \tag{6.35}$$

where

$$\kappa_{\text{curv}} \stackrel{\text{def}}{=} (f(x_0) - f_{\text{low}})/(\eta_1 \alpha_{\text{curv}}) \quad \text{and} \quad \alpha_{\text{curv}} \stackrel{\text{def}}{=} \sigma_{\min}/(6L_0^3), \tag{6.36}$$

with $\sigma_{\min}$ and L_0 defined in (5.1) and (5.6), respectively. Assuming that (6.34) holds at $k = 0$, the ACO algorithm takes at most $\mathrm{L}_2^{\mathrm{s}}$ successful iterations to generate a (first) iterate, say l_2, with $-\lambda_{\min}(Q_{l_2+1}^\top B_{l_2+1} Q_{l_2+1}) \leq \epsilon$. Furthermore, when $\epsilon \leq 1$, we have

$$l_2 \leq \left\lceil \kappa_{\text{curv}}^t \epsilon^{-3} \right\rceil \stackrel{\text{def}}{=} \mathrm{L}_2, \tag{6.37}$$

and so the ACO algorithm takes at most L_2 (successful and unsuccessful) iterations to generate $-\lambda_{\min}(Q_{l_2+1}^\top B_{l_2+1} Q_{l_2+1}) \leq \epsilon$, where $\kappa_{\text{curv}}^t \stackrel{\text{def}}{=} (1 + \kappa_{\mathrm{S}}^u)\kappa_{\text{curv}} + \kappa_{\mathrm{S}}^u$ and κ_{S}^u is defined in (6.28).

Proof. Lemma 3.2 implies that the matrix $Q_k^\top B_k Q_k + \sigma_k \|s_k\| I$ is positive semidefinite and thus,

$$\lambda_{\min}(Q_k^\top B_k Q_k) + \sigma_k \|s_k\| \geq 0, \quad \text{for } k \geq 0,$$

which further gives

$$\sigma_k \|s_k\| \geq |\lambda_{\min}(Q_k^\top B_k Q_k)|, \quad \text{for any } k \geq 0 \text{ such that } -\lambda_{\min}(Q_k^\top B_k Q_k) > \epsilon, \tag{6.38}$$

since the latter inequality implies $\lambda_{\min}(Q_k^\top B_k Q_k) < 0$. It follows from (3.19), (5.6) and (6.38) that

$$f(x_k) - m_k(s_k) \geq \alpha_{\text{curv}} \epsilon^3, \quad \text{for all } k \geq 0 \text{ with } -\lambda_{\min}(Q_k^\top B_k Q_k) > \epsilon, \tag{6.39}$$

where α_{curv} is defined in (6.36). Define $\mathcal{S}_\lambda^\epsilon \stackrel{\text{def}}{=} \{k \in \mathcal{S} : -\lambda_{\min}(Q_k^\top B_k Q_k) > \epsilon\}$ and $|\mathcal{S}_\lambda^\epsilon|$, its cardinality. Letting $F_k = |\lambda_{\min}(Q_k^\top B_k Q_k)|$, $\mathcal{S}_o = \mathcal{S}_{\mathrm{F}}^\epsilon = \mathcal{S}_\lambda^\epsilon$ and $p = 3$ in Theorem 6.2 provides the bound

$$|\mathcal{S}_\lambda^\epsilon| \leq \mathrm{L}_2^{\mathrm{s}}, \quad \text{where } \mathrm{L}_2^{\mathrm{s}} \text{ is defined in (6.35).} \tag{6.40}$$

Assuming that (6.34) holds at $k = 0$, and recalling that l_2 is the first iteration such that (6.34) does not hold at $l_2 + 1$ and that $\mathcal{S}_{l_2}$ is (6.1) with $j = l_2$, we have $\mathcal{S}_{l_2} \subseteq \mathcal{S}_\lambda^\epsilon$. Thus (6.40) implies

$$|\mathcal{S}_{l_2}| \leq \mathrm{L}_2^{\mathrm{s}}. \tag{6.41}$$

A bound on the number of unsuccessful iterations up to l_2 can be obtained in the same way as in the proof of Corollary 6.5, since Theorem 6.1 does not depend on the choice of optimality measure F_k. Thus we deduce, also from (6.41),

$$|\mathcal{U}_{l_2}| \leq \lceil (1 + |\mathcal{S}_{l_2}|)\kappa_{\mathrm{S}}^u \rceil \leq \lceil (1 + \mathrm{L}_2^{\mathrm{s}})\kappa_{\mathrm{S}}^u \rceil, \tag{6.42}$$

where $\mathcal{U}_{l_2}$ is given in (6.1) with $j = l_2$ and κ_{S}^u, in (6.28). Since $l_2 = |\mathcal{S}_{l_2}| + |\mathcal{U}_{l_2}|$, the bound (6.37) readily follows from $\epsilon \leq 1$, (6.41) and (6.42). □

Note that the complexity bounds in Corollary 6.6 also give a bound on the number of the iterations at which negative curvature occurs along the step s_k by considering $\mathcal{L}_k$ as the subspace generated by the normalized s_k.

Assuming s_k minimizes m_k globally over the subspace generated by the columns of the orthogonal matrix Q_k for $k \geq 0$, let us now briefly remark on the complexity of driving the leftmost negative eigenvalue of $Q_k^T H(x_k) Q_k$ — as opposed to $Q_k^T B_k Q_k$ — below a given tolerance, i. e.,

$$-\lambda_{\min}(Q_k^T H(x_k) Q_k) \leq \epsilon. \tag{6.43}$$

In the conditions of Corollary 6.6, let us further assume that

$$\|B_k - H(x_k)\| \leq \epsilon_2, \quad \text{for all } k \geq k_1 \text{ where } k_1 \text{ is such that } \|g_{k_1}\| \leq \epsilon_1, \tag{6.44}$$

for some positive parameters ϵ_1 and ϵ_2, with $\epsilon_2 \sqrt{n} < \epsilon$. Then Corollary 6.5 gives an upper bound on the (first) iteration k_1 with $\|g_k\| \leq \epsilon_1$, and we are left with having to estimate $k \geq k_1$ until (6.43) is achieved. But (5.11) and (6.44) give

$$|\lambda_{\min}(Q_k^T H_k Q_k) - \lambda_{\min}(Q_k^T B_k Q_k)| \leq \epsilon_2 \sqrt{n}, \ k \geq k_1, \tag{6.45}$$

and thus, (6.43) is satisfied when

$$-\lambda_{\min}(Q_k^T B_k Q_k) \leq \epsilon - \epsilon_2 \sqrt{n} \stackrel{\text{def}}{=} \epsilon_3. \tag{6.46}$$

Now Corollary 6.6 applies and gives us an upper bound on the number of iterations k such that (6.46) is achieved, which is $\mathcal{O}(\epsilon_3^{-3})$.

If we make the choice $B_k = H(x_k)$ and Q_k is full-dimensional for all $k \geq 0$, then the above argument or the second part of Corollary 6.6 imply that (6.43) is achieved for k at most $\mathcal{O}(\epsilon^{-3})$, which recovers the result obtained by Nesterov and Polyak [20, p. 185] for their Algorithm (3.3).

6.2.2 A complexity bound for achieving approximate first- and second-order optimality

Finally, in order to estimate the complexity of generating an iterate that is both approximately first- and second-order critical, let us combine the results in Corollaries 6.5 and 6.6.

Corollary 6.7. Let AF.1–AF.2, AF.4, AM.1, AM.4 and TC.s hold, and $\{f(x_k)\}$ be bounded below by f_{low}. Let σ_k be bounded below as in (5.1), and s_k be the global minimizer of $m_k(s)$ over a subspace $\mathcal{L}_k$ that is generated by the columns of an orthogonal matrix Q_k. Given any $\epsilon \in (0,1)$, the ACO algorithm generates $l_3 \geq 0$ with

$$\max\left(\|g_{l_3+1}\|, -\lambda_{\min}(Q_{l_3+1}^T B_{l_3+1} Q_{l_3+1})\right) \leq \epsilon \tag{6.47}$$

in at most $\lceil \kappa_{\text{fs}}^s \epsilon^{-3} \rceil$ successful iterations, where

$$\kappa_{\text{fs}}^s \stackrel{\text{def}}{=} \kappa_{\text{S}}^s + \kappa_{\text{curv}} + 1, \tag{6.48}$$

and κ_{S}^s and κ_{curv} are defined in (6.26) and (6.36), respectively. Furthermore, $l_3 \leq \lceil \kappa_{\text{fs}} \epsilon^{-3} \rceil$, where $\kappa_{\text{fs}} \stackrel{\text{def}}{=} (1 + \kappa_{\text{S}}^u)\kappa_{\text{fs}}^s + \kappa_{\text{S}}^u$ and κ_{S}^u is defined in (6.28).

Proof. The conditions of Corollaries 6.5 and 6.6 are satisfied. Thus the sum of the bounds (6.25) and (6.40), i. e.,

$$\lceil \kappa_{\text{S}}^s \epsilon^{-3/2} + \kappa_{\text{curv}} \epsilon^{-3} \rceil, \tag{6.49}$$

is an upper bound on all the possible successful iterations that may occur either with $\min(\|g_k\|, \|g_{k+1}\|) > \epsilon$ or with $-\lambda_{\min}(Q_k^T B_k Q_k) > \epsilon$. As the first of these criticality measures involves both iterations k and $k+1$, the latest such a successful iteration is given by (6.48). The bound on l_3 follows from Theorem 6.1, as in the proof of Corollary 6.5. □

The above result shows that the better bound (6.27) for approximate first-order optimality is obliterated by (6.37) for approximate second-order optimality (in the minimization subspaces) when seeking accuracy in both these optimality conditions.

Counting zero gradient values. Recall the discussion in the last paragraph of §3.2 and §5 regarding the case when there exists $k \geq 0$ such that $g_k = 0$ and thus (2.49) does not hold. Then the conditions of Corollary 6.6 are satisfied since $s_k \neq 0$ and (2.50) holds due to (3.21) and (3.22), respectively. Furthermore, (6.39) remains satisfied even when (5.19) holds with $k_l = k$, since our derivation of (6.39) in the proof of Corollary 6.6 does not depend on the value of the gradient. Similarly, Corollary 6.7 also continues to hold in this case.

7 Methods for approximately minimizing the cubic model

While the ACO algorithm provides a powerful general framework for unconstrained minimization, the practicality and efficiency of this algorithm is obviously related to our ability to find a suitable (approximate) minimizer of the cubic model at each iteration. In this section we consider this issue in some detail. The optimality conditions in Theorem 3.1 for the global minimizer of $m_k(s)$ over $\mathbb{R}^n$ are highly suggestive of efficient algorithms in many cases, as we discuss in the first subsection. We then concentrate on one way in which this minimizer may be approximated in practice, while retaining both the convergence and complexity properties of the true model minimizer. Most especially, the method we propose is "matrix-free"—that is, we only requires Hessian-vector products rather than access to the Hessian itself—and thus may be used in principle for large, unstructured problems.

Throughout this section, we drop the (major) iteration subscript k for convenience.

7.1 Computing the global solution

To start with, we suppose that we wish to compute the global model minimizer of $m(s)$ over $\mathbb{R}^n$. Theorem 3.1 shows that we seek a pair (s, λ) for which

$$(B + \lambda I)s = -g \;\text{ and }\; \lambda^2 = \sigma^2 s^T s \tag{7.1}$$

and for which $B + \lambda I$ is positive semidefinite. Just as in the trust-region case [3, §7.3.1], suppose that B has an eigendecomposition

$$B = U^T \Lambda U, \tag{7.2}$$

where Λ is a diagonal matrix of eigenvalues $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$ and U is an orthonormal matrix of associated eigenvectors. Then

$$B(\lambda) = U^T(\Lambda + \lambda I)U.$$

We deduce immediately from Theorem 3.1 that the value of λ we seek must satisfy $\lambda \geq -\lambda_1$ as only then is $B(\lambda)$ positive semidefinite.

Suppose that $\lambda > -\lambda_1$. Then $B + \lambda I$ is positive definite, and thus $m(s)$ has a unique global minimizer

$$s(\lambda) = -(B + \lambda I)^{-1} g = -U^T(\Lambda + \lambda I)^{-1} U g. \tag{7.3}$$

But, of course, the solution we are looking for depends upon the nonlinear inequality $\|s(\lambda)\|_2 = \lambda/\sigma$. To say more, we need to examine $\|s(\lambda)\|_2$ in detail. For convenience, define $\psi(\lambda) \stackrel{\text{def}}{=} \|s(\lambda)\|_2^2$. We have that

$$\psi(\lambda) = \|U^T(\Lambda + \lambda I)^{-1} U g\|_2^2 = \|(\Lambda + \lambda I)^{-1} U g\|_2^2 = \sum_{i=1}^{n} \frac{\gamma_i^2}{(\lambda_i + \lambda)^2}, \tag{7.4}$$

where γ_i is the ith component of Ug.

If B is positive semidefinite, the required solution is given by the single positive root to either of the equivalent univariate nonlinear equations

$$\theta_2(\lambda) \stackrel{\text{def}}{=} \psi(\lambda) - \frac{\lambda^2}{\sigma^2} = 0 \;\text{ or }\; \theta_1(\lambda) \stackrel{\text{def}}{=} \sqrt{\psi(\lambda)} - \frac{\lambda}{\sigma} = 0. \tag{7.5}$$

If B is indefinite and $\gamma_1 \neq 0$, the solution is likewise the root larger than $-\lambda_1$ of the same equations. But if B is indefinite and $\gamma_1 = 0$, we have difficulties analogous to those for the hard case [3, §7.3.1.3] for the trust-region subproblem in which the required solution s_* is made up as a suitable linear combination of u_1 and $\lim_{s\to-\lambda_1} s(\lambda)$. To determine the values of the coefficients of this linear combination, in place of the trust-region constraint, we employ (7.1), and find a value for $\alpha \in \mathbb{R}$ such that

$$-\lambda_1 = \sigma\|s(-\lambda_1) + \alpha u_1\|.$$

See Figure 7.1 for an illustration of the graphs of θ_1 and θ_2.

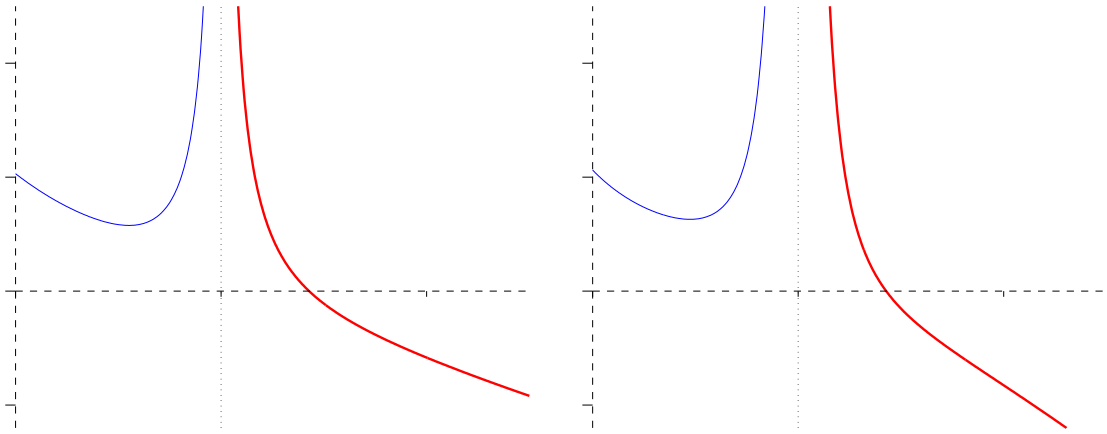

Figure 7.1: Graphs of the functions $\theta_1(\lambda)$ (left) and $\theta_2(\lambda)$ (right) from (7.5) when $g = (0.25 \quad 1)^T$, $H = \text{diag}(-1 \;\; 1)$ and $\sigma = 2$.

In practice, just as in the trust-region case, it may be preferable to solve one of

$$\begin{aligned} \phi_2(\lambda) &\stackrel{\text{def}}{=} \frac{1}{\psi(\lambda)} - \frac{\sigma^2}{\lambda^2} = 0, \qquad \phi_1(\lambda) \stackrel{\text{def}}{=} \frac{1}{\sqrt{\psi(\lambda)}} - \frac{\sigma}{\lambda} = 0, \\ \beta_2(\lambda) &\stackrel{\text{def}}{=} \frac{\lambda^2}{\psi(\lambda)} - \sigma^2 = 0 \;\text{ or } \qquad \beta_1(\lambda) \stackrel{\text{def}}{=} \frac{\lambda}{\sqrt{\psi(\lambda)}} - \sigma = 0 \end{aligned} \tag{7.6}$$

instead of (7.5). Figures 7.2 and 7.3 illustrate these alternatives.

In any event, a safeguarded univariate Newton iteration to find the required root of whichever of the functions (7.5) or (7.6) we prefer is recommended, but in all cases this requires the solution of a sequence of linear equations

$$B(\lambda^{(k)})s^{(k)} \equiv (B + \lambda^{(k)}I)s^{(k)} = -g$$

for selected $\lambda^{(k)} > \max(0, -\lambda_1)$. Clearly to use Newton's method, derivatives of (simple functions of) $\psi(\lambda)$ will be required, but fortunately these are easily obtained once a factorization of $B + \lambda^{(k)}I$ is known. In particular, we have the result below.

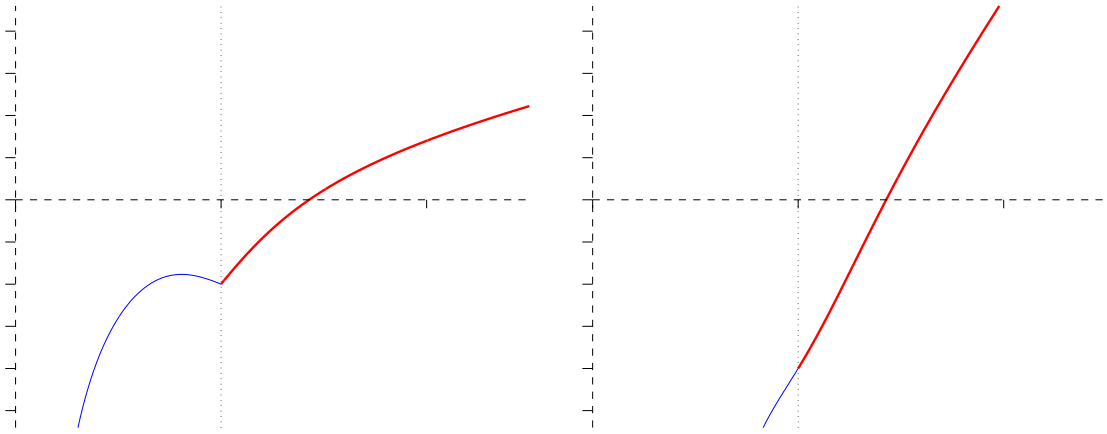

Figure 7.2: Graphs of the functions $\phi_1(\lambda)$ (left) and $\phi_2(\lambda)$ (right) from (7.6) when $g = (0.25 \quad 1)^T$, $H = \mathrm{diag}(-1 \quad 1)$ and $\sigma = 2$.

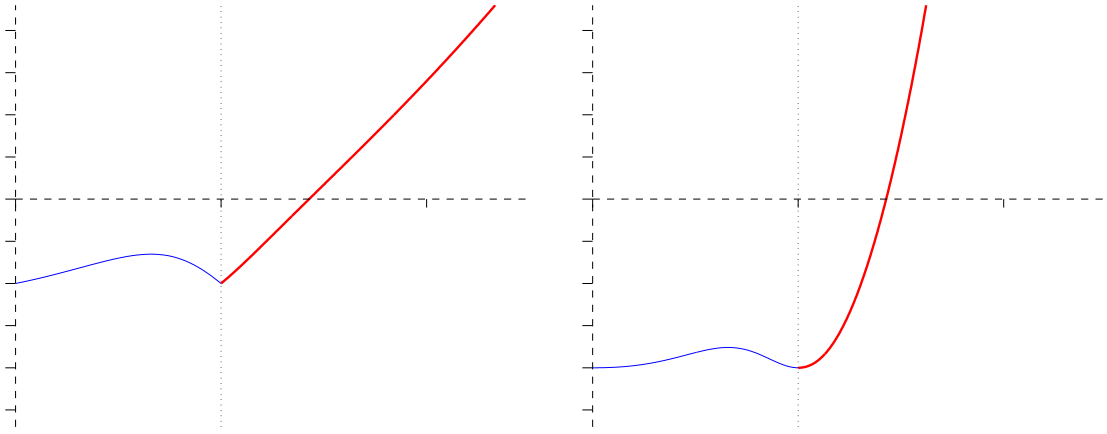

Figure 7.3: Graphs of the functions $\beta_1(\lambda)$ (left) and $\beta_2(\lambda)$ (right) from (7.6) when $g = (0.25 \quad 1)^T$, $H = \mathrm{diag}(-1 \quad 1)$ and $\sigma = 2$.

Lemma 7.1. Suppose that $s(\lambda)$ satisfies (7.3), $\psi(\lambda) = \|s(\lambda)\|_2^2$ and $\lambda > -\lambda_1$. Then

$$\psi'(\lambda) = 2s(\lambda)^T \nabla_\lambda s(\lambda) \;\text{ and }\; \psi''(\lambda) = 6\|\nabla_\lambda s(\lambda)\|_2^2, \tag{7.7}$$

where

$$\nabla_\lambda s(\lambda) = -B(\lambda)^{-1} s(\lambda). \tag{7.8}$$

Moreover, given the Cholesky factorization $B(\lambda) = L(\lambda)L^T(\lambda)$, it follows that

$$s(\lambda)^T \nabla_\lambda s(\lambda) = -\|w(\lambda)\|_2^2,$$

where $L(\lambda)L^T(\lambda)s(\lambda) = -g$ and $L(\lambda)w(\lambda) = s(\lambda)$.

Proof. See the proof of [3, Lem. 7.3.1]. □

Then, for example, for $\phi_1(\lambda)$ from (7.6), we obtain the following expressions.

Corollary 7.2. Suppose $g \neq 0$. Then the function $\phi_1(\lambda)$ is strictly increasing, when $\lambda > \max(0, -\lambda_1)$, and concave. Its first two derivatives are

$$\phi_1'(\lambda) = -\frac{s(\lambda)^T \nabla_\lambda s(\lambda)}{\|s(\lambda)\|_2^3} + \frac{\sigma}{\lambda^2} > 0 \tag{7.9}$$

and

$$\phi_1''(\lambda) = \frac{3\left(s(\lambda)^T \nabla_\lambda s(\lambda)^2 - \|s(\lambda)\|_2^2 \|\nabla_\lambda s(\lambda)\|_2^2\right)}{\|s(\lambda)\|_2^5} - \frac{2\sigma}{\lambda^3} < 0. \tag{7.10}$$

Proof. Again, see the proof of [3, Lem. 7.3.1]. □

In this case, the basic Newton iteration is as follows.

Algorithm 7.1: Newton's method to solve $\phi_1(\lambda) = 0$

Let $\lambda > \max(0, -\lambda_1)$ be given.

Step 1. Factorize $B(\lambda) = LL^T$.

Step 2. Solve $LL^T s = -g$.

Step 3. Solve $Lw = s$.

Step 4. Compute the Newton correction $\Delta\lambda^{\text{N}} = \dfrac{\lambda\left(\|s\|_2 - \dfrac{\lambda}{\sigma}\right)}{\|s\|_2 + \dfrac{\lambda}{\sigma}\left(\dfrac{\lambda\|w\|_2^2}{\|s\|_2^2}\right)}$.

Step 5. Replace λ by $\lambda + \Delta\lambda^{\text{N}}$.

In practice, various safeguards of the kinds described for the trust-region case in [3, §7.3.4–7.3.8] should be added to ensure convergence of Algorithm 7.1 from an arbitrary initial λ. Numerical experience has

indicated that the speed of (global) convergence may be improved by only linearizing the term $\omega(\lambda) \stackrel{\text{def}}{=} 1/\sqrt{\psi(\lambda)}$ of ϕ_1 in (7.6)—and not the σ/λ term as does Newton's method—when computing a correction $\Delta\lambda^{\text{C}}$ to the estimate λ of the required root of ϕ_1. The resulting correction thus satisfies the equation

$$\omega(\lambda) + \omega'(\lambda)\Delta\lambda^{\text{C}} \equiv \frac{1}{\psi^{\frac{1}{2}}(\lambda)} - \tfrac{1}{2}\frac{\psi'(\lambda)}{\psi^{\frac{3}{2}}(\lambda)}\Delta\lambda^{\text{C}} = \frac{\sigma}{\lambda + \Delta\lambda^{\text{C}}}, \tag{7.11}$$

which may be rewritten as a quadratic equation for $\Delta\lambda^{\text{C}}$.

Although Algorithm 7.1 and the variant using (7.11) are not generally globally convergent, there is one very important case in which they will be.

Theorem 7.3. Suppose $\lambda > -\lambda_1$ and $\phi_1(\lambda) < 0$. Then both the Newton iterate $\lambda + \Delta\lambda^{\text{N}}$ and alternative $\lambda + \Delta\lambda^{\text{C}}$ inherit these properties and converge monotonically towards the required root, λ_*. The convergence is globally Q-linear with factor at least

$$\gamma_\lambda \stackrel{\text{def}}{=} 1 - \frac{\phi_1'(\lambda_*)}{\phi_1'(\lambda)} < 1$$

and is ultimately Q-quadratic. Moreover $\lambda + \Delta\lambda^{\text{N}} \leq \lambda + \Delta\lambda^{\text{C}} \leq \lambda_*$.

Proof. The proof in the case of the Newton iterate is essentially identical to that of [3, Lem. 7.3.2]. Since $\omega(\lambda)$ is a concave function of λ, (7.6) and (7.11) give that

$$\phi_1(\lambda + \Delta\lambda^{\text{C}}) = \omega(\lambda + \Delta\lambda^{\text{C}}) - \frac{\sigma}{\lambda + \Delta\lambda^{\text{C}}} \leq \omega(\lambda) + \omega'(\lambda)\Delta\lambda^{\text{C}} - \frac{\sigma}{\lambda + \Delta\lambda^{\text{C}}} = 0.$$

The Newton correction satisfies the linearized equation

$$\omega(\lambda) + \omega'(\lambda)\Delta\lambda^{\text{N}} = \frac{\sigma}{\lambda} - \frac{\sigma}{\lambda^2}\Delta\lambda^{\text{N}}. \tag{7.12}$$

But, as σ/λ is a convex function of λ,

$$\frac{\sigma}{\lambda + \Delta\lambda^{\text{C}}} \geq \frac{\sigma}{\lambda} - \frac{\sigma}{\lambda^2}\Delta\lambda^{\text{C}},$$

and hence

$$\omega(\lambda) + \omega'(\lambda)\Delta\lambda^{\text{C}} \geq \frac{\sigma}{\lambda} - \frac{\sigma}{\lambda^2}\Delta\lambda^{\text{C}},$$

from (7.11). Combining this with (7.12), we obtain

$$\phi_1'(\lambda)(\Delta\lambda^{\text{C}} - \Delta\lambda^{\text{N}}) = (\omega'(\lambda) + \frac{\sigma}{\lambda^2})(\Delta\lambda^{\text{C}} - \Delta\lambda^{\text{N}}) \geq 0$$

and hence $\Delta\lambda^{\text{C}} \geq \Delta\lambda^{\text{N}} > 0$ since Corollary 7.2 gives $\phi_1'(\lambda) > 0$. Thus the alternative iterates improves on the Newton one. □

Similar results are easily derived for the other root functions defined in (7.5) and (7.6).

We conclude this section with an interesting observation concerning the global minimizer $s(\sigma)$ of the cubic model $m(s, \sigma) \equiv m(s)$ in (1.3), where we now make clear the explicit dependence on the parameter σ.

Theorem 7.4. Suppose that $s(\sigma) \neq 0$ is a global minimizer of the model $m(s, \sigma) \equiv m(s)$ in (1.3). Then the length of the minimizer $\nu(\sigma) \stackrel{\text{def}}{=} \|s(\sigma)\|$ is a non-increasing function of σ.

Proof. We have from Theorem 3.1 that

$$(B + \sigma\|s(\sigma)\|I)s(\sigma) = -g \tag{7.13}$$

and that $B + \sigma\|s(\sigma)\|I$ and thus $B + \sigma\|s(\sigma)\|I + \sigma\|s(\sigma)\|^{-1}s(\sigma)s^T(\sigma)$ are positive semidefinite. We consider the derivative $\nu'(\sigma) = \|s(\sigma)\|^{-1}s^T(\sigma)\nabla_\sigma s(\sigma)$. Differentiating (7.13) with respect to σ reveals that

$$(B + \sigma\|s(\sigma)\|I)\nabla_\sigma s(\sigma) + \|s(\sigma)\|s(\sigma) + \sigma\|s(\sigma)\|^{-1}s(\sigma)s^T(\sigma)\nabla_\sigma s(\sigma) = 0$$

and thus that

$$\left(B + \sigma\|s(\sigma)\|I + \sigma\|s(\sigma)\|^{-1}s(\sigma)s^T(\sigma)\right)\nabla_\sigma s(\sigma) = -\|s(\sigma)\|s(\sigma). \tag{7.14}$$

Premultiplying (7.14) by $s^T(\sigma)$ and dividing by $\|s(\sigma)\|$ gives that

$$\nu'(\sigma) = -\frac{\nabla_\sigma s^T(\sigma)(B + \sigma\|s(\sigma)\|I + \sigma\|s(\sigma)\|^{-1}s(\sigma)s^T(\sigma))\nabla_\sigma s(\sigma)}{\|s(\sigma)\|^2} \leq 0$$

since we have seen that $B + \sigma\|s(\sigma)\|I + \sigma\|s(\sigma)\|^{-1}s(\sigma)s^T(\sigma)$ is positive semidefinite. Thus $\nu'(\sigma)$ is a non-increasing function of σ. □

7.2 Computing an approximate solution

Of course, the dominant cost of the methods we have just discussed is that of factorizing $B+\lambda I$ for various λ, and this may be prohibitive for large n; indeed factorization may be impossible. An obvious alternative is to use a Krylov method to approximate the solution. This was first proposed in [11] for trust-region methods.

The Lanczos method may be used to build up an orthogonal basis $\{q_0, \cdots, q_j\}$ for the Krylov space $\{g, Bg, B^2g, \ldots, B^jg\}$, formed by applying successively B to g. Letting $Q_j = (q_0, \cdots q_j)$, the key relationships are

$$Q_j^TQ_j = I, \;\; Q_j^TBQ_j = T_j \;\; \text{and} \;\; Q_j^Tg = \gamma_0 e_1, \tag{7.15}$$

where e_1 is the first unit vector of appropriate length and T_j is a symmetric, tridiagonal matrix.

We shall consider vectors of the form

$$s \in \mathcal{S}_j = \{s \in \mathbb{R}^n \mid s = Q_ju\}$$

and seek

$$s_j = Q_ju_j, \tag{7.16}$$

where s_j solves the problem

$$\underset{s \in \mathcal{S}_j}{\text{minimize}} \;\; m(s). \tag{7.17}$$

It then follows directly from (7.15) that u_j solves the problem

$$\underset{u \in \mathbb{R}^{j+1}}{\text{minimize}} \;\; m_j(u) \stackrel{\text{def}}{=} f + \gamma_0u^Te_1 + \tfrac{1}{2}u^TT_ju + \tfrac{1}{3}\sigma\|u\|_2^3. \tag{7.18}$$

There are a number of crucial observations to be made here. Firstly, as T_j is tridiagonal, it is feasible to use the method broadly described in §7.1 to compute the solution to (7.17) even when n is large. Secondly, having found u_j, the matrix Q_j is needed to recover s_j, and thus the Lanczos vectors q_j will either need to be saved on backing store or regenerated when required. We only need Q_j once we are satisfied that continuing the Lanczos process will give little extra benefit. Thirdly, one would hope that as a sequence of such problems may be solved, and as T_j only changes by the addition of an extra diagonal and superdiagonal entry, solution data from one subproblem may be useful for starting the next. Lastly, this is a clear extension of the GLTR method for the solution of the trust-region problem [11], and many of the implementation issues and details follow directly from there.

Furthermore, employing this approach within the ACO algorithm benefits from the theoretical guarantees of convergence and complexity in §2.2–§6. To see this, let $\mathcal{L}_k = \mathcal{S}_j$ in Lemma 3.2 and note that the current gradient is included in all subspaces $\mathcal{S}_j$.

7.3 Scaled regularization

The preceding development can trivially be generalized if the regularization $\frac{1}{3}\sigma\|s\|_2^3$ is replaced by the scaled variant $\frac{1}{3}\sigma\|s\|_M^3$, where we define $\|s\|_M = s^T M s$ for some symmetric positive definite M. All that changes is that the key second-derivative matrix is $B(\lambda) = B + \lambda M$ in Theorem 3.1 and its successors, and that M-orthogonal vectors are generated using the preconditioned Lanczos method; the regularization in the tridiagonal problem (7.18) is not altered.

7.4 A further possibility

Another possibility is suppose that the optimal $\|s\|_2$ is known to be of size Δ. In this case, the required value of $m(s)$ is $f + s^T g + \frac{1}{2}s^T B s + \frac{1}{3}\sigma\Delta^3$, where s solves the trust-region problem

$$q(\Delta) = \min_{s\in\mathbb{R}^n} \; s^T g + \tfrac{1}{2}s^T B s \;\text{ subject to }\; \|s\|_2 = \Delta.$$

Hence

$$\min_{s\in\mathbb{R}^n} \; m(s) = \min_{\Delta\in\mathbb{R}_+} \; q(\Delta) + \tfrac{1}{3}\sigma\Delta^3$$

and we may use known algorithms for the trust-region problem to accomplish the univariate minimization of $q(\Delta) + \frac{1}{3}\sigma\Delta^3$. We have not considered this possibility further at this stage.

8 Numerical results

We now turn our attention to investigating how the cubic overestimation method performs in practice. We have implemented the ACO algorithm, together with both the exact and inexact subproblem solvers described in §7.1 and §7.2. To be specific, when solving the subproblem, we compute the required root λ of $\phi_1(\lambda)$; we use Algorithm 7.1 to find the required root, replacing the correction $\Delta\lambda$ in Step 4 by the improvement given by (7.11). To simplify matters, we start the root finding from $\max(0, -\lambda_1) + \epsilon$ for some tiny ϵ—this of course entails that we find the eigenvalue λ_1 of B (§7.1 and the built-in MATLAB function `eigs`) or T_j (§7.2 and the specially-designed algorithm given in [11]), which is potentially expensive in the case of B, but far less so for T_j, especially since we have that of T_{j-1} as a starting estimate—in which case Theorem 7.3 will ensure global (and ultimately) rapid convergence.

In view of its suitability for large-scale problems, in the results we shall present, we used the Lanczos-based inexact solver described in §7.2. Using the exact solver gives similar results for the small-scale problems that we tested, since the values of the parameters in the stopping rules we have chosen to use require that we solve to reasonably high relative accuracy in the inexact case (see (8.1)–(8.3)). We considered three stopping rules for the inexact inner iteration, all derived from the TC.h criteria in §3.3. In the first, recalling TC.g, we stop as soon as the approximate solution in Step 2 of the ACO algorithm satisfies

$$\|\nabla m_k(s_k)\| \leq \min(0.0001, \|\nabla m_k(0)\|^{\frac{1}{2}})\|\nabla m_k(0)\|; \tag{8.1}$$

the aim is to try to encourage rapid ultimate convergence [5, 17] without the expense of "over-solving" when far from optimality—we refer to (8.1) as the "g rule" (see Corollary 4.8 where we show the ACO algorithm with such a termination criteria converges Q-superlinearly). The remaining rules are geared more towards ensuring the best overall complexity bound we have obtained (see §6.2). Thus our second "s rule" comes from TC.s, and it is to stop as soon as

$$\|\nabla m_k(s_k)\| \leq \min(0.0001, \|s_k\|)\|\nabla m_k(0)\|. \tag{8.2}$$

However since we were concerned that this might be overly stringent when s_k is small when σ_k is large rather than because $\nabla m_k(0)$ is small, our final "s/σ rule" is to stop as soon as

$$\|\nabla m_k(s_k)\| \leq \min\left(0.0001, \frac{\|s_k\|}{\max(1,\sigma_k)}\right)\|\nabla m_k(0)\|. \tag{8.3}$$

The ACO algorithm converges at least Q-superlinearly also when (8.2) and (8.3) are employed (see Corollaries 4.8 and 4.10, and the remarks inbetween).

The other crucial ingredient is the management of the regularization parameter σ_k in Step 4 of the ACO algorithm. Here, on very successful iterations, we set $\sigma_{k+1} = \max(\min(\sigma_k, \|g_k\|), \epsilon_M)$, where $\epsilon_M \approx 10^{-16}$ is the relative machine precision—the intention is to try to reduce the model rapidly to Newton ($\sigma = 0$) model once convergence sets in, while maintaining some regularization before the asymptotics are entered. For other successful steps we leave σ_k unchanged, while for unsuccessful steps we increase σ_k by 2 (the choice of the "2" factor is for simplicity, and it is likely that better values are possible in a similar vein to [12]). We start with $\sigma_0 = 1$, and use $\eta_1 = 0.1$ and $\eta_2 = 0.9$, similar performance being observed for other choices of initial parameters.

By way of a comparison, we have also implemented a standard trust-region method [3, Alg. 6.1.1.]. Here we have used the GLTR method [11] to find an approximate solution of the trust-region problem, stopping as above as soon as (8.1) is satisfied. The trust-region radius Δ_{k+1} following a very successful iteration is $\min(\max(2\|s_k\|, \Delta_k), 10^{10})$, it is left unchanged if the iteration is merely successful, while an unsuccessful step results in a halving of the radius. The initial radius is always 1.

We give the results obtained by applying both Algorithms to all of the unconstrained problems from the CUTEr collection [13]; for those whose dimensions may be adjusted, we chose small variants simply so as not to overload our (Matlab) computing environment, most particularly the CUTEr interface to Matlab. All of our experiments were performed on a single processor of a 3.06 GHz Dell Precision 650 Workstation. Both our new algorithm, and the trust-region algorithm were implemented as Matlab M-files, and the tests performed using Matlab 7.2.

We give the complete set of results in Appendix A. The algorithm is stopped as soon as the norm of the gradient $\|g(x_k)\|$ is smaller than 10^{-5}. An upper limit of 10000 iterations was set, and any run exceeding this is flagged as a failure. In Figure 8.1, we present the iteration-count performance profile [9] for the methods. We would have liked to have given a similar figure for CPU times, but the Matlab

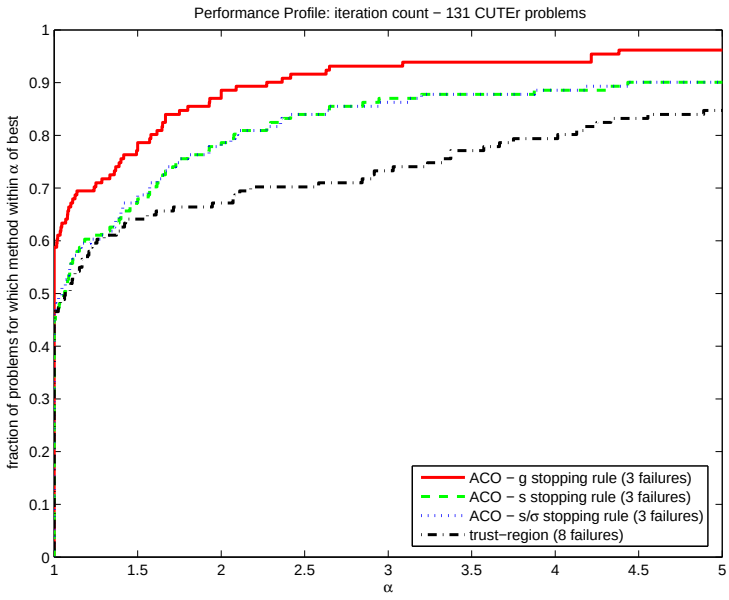

Figure 8.1: Performance profile, $p(\alpha)$: Iteration counts for the 131 problems under consideration.

CPU timer proved too inaccurate for this purpose—we defer such a comparison until we have produced a careful (Fortran) implementation and tested it on larger examples in a controlled computing environment.

While we should regard our results as tentative, since improvements to both algorithms are most likely, we are delighted with the overall performance of our new algorithm. Trust-region methods are generally

considered to be reliable and effective methods for unconstrained minimization, and thus it is gratifying to be able to improve upon them in so many cases. Of course, there are cases where the new algorithm isn't as effective, but in general the algorithm appears able to move to the asymptotic region more rapidly. Whether this is a consequence of a provably good complexity bound or for other reasons of robustness is not clear. For the three cases where the new algorithm fails, slow (but sure) progress is made towards the solution, but these problems are well known to be very hard. Of the three variants of the new method, that which is less concerned with provably superior worst-case complexity, appears to be more promising. This is perhaps not so surprising since the more conservative acceptance rules (8.2) and (8.3) aim for guaranteed rather than speculative reduction. There is very little to choose between the latter pair, indicating that our concern that (8.2) may be over-stringent may have been misplaced.

9 Conclusions

In this paper we have considered the global convergence properties of a new general cubic-overestimation framework for unconstrained optimization which includes the proposal of Nesterov and Polyak [20]. The framework allows for the approximate solution of the key step calculation, and is suitable for large-scale problems. We presented a Lanczos-based approximation method which is covered by our theoretical developments. In practice, the new method is competitive with trust-region methods in tests for small-scale problems.

Norm regularisations of the quadratic model with other powers than cubic are possible, yielding algorithm variants with better first-order worst-case complexity guarantees than the steepest descent and even, than the ACO algorithm for a restricted class of functions, whose practical efficiency remains to be explored.

Our next goal is to implement and test these ideas carefully in the large-scale case. Extensions to these ideas are obvious and far-reaching. In particular, since the use of trust-region models is widespread in optimization, it is worth investigating where cubic models might be employed in their place. Projection methods for bound constrained minimization and penalty/barrier/augmented Lagrangian methods for constrained optimization are obvious candidates. Note that in the case of linear equality constraints, the (semi-)norm will only need to regularise in the null-space of the constraints, and solving the subproblem is likewise easy so long as the Krylov subspace is projected onto the constraint manifold [3]. More generally, difficulties resulting form the incompatibility of the intersection of linearized constraints with trust-region bounds has been a perennial problem in constrained optimization; (cubic) regularisation offers an easy way to avoid this difficulty.

References

[1] R. H. Byrd, H. F. Khalfan and R. B. Schnabel. Analysis of a symmetric rank-one trust region method. *SIAM Journal on Optimization*, 6(4):1025–1039, 1996.

[2] A. R. Conn, N. I. M. Gould, and Ph. L. Toint. Convergence of quasi-Newton matrices generated by the symmetric rank one update. *Mathematical Programming*, 50(2):177–196, 1991.

[3] A. R. Conn, N. I. M. Gould, and Ph. L. Toint. *Trust-Region Methods*. SIAM, Philadelphia, USA, 2000.

[4] R. S. Dembo, S. C. Eisenstat and T. Steihaug. Inexact-Newton methods. *SIAM Journal on Numerical Analysis*, 19(2):400–408, 1982.

[5] R. S. Dembo and T. Steihaug. Truncated-Newton algorithms for large-scale unconstrained optimization. *Mathematical Programming*, 26(2):190–212, 1983.

[6] J. E. Dennis and J. J. Moré. A characterization of superlinear convergence and its application to quasi-Newton methods. *Mathematics of Computation*, 28(126):549–560, 1974.

[7] J. E. Dennis and R. B. Schnabel. *Numerical methods for unconstrained optimization and nonlinear equations*. SIAM, Philadelphia, USA, 1996.

[8] P. Deuflhard. *Newton Methods for Nonlinear Problems. Affine Invariance and Adaptive Algorithms*. Springer Series in Computational Mathematics, Vol. 35. Springer, Berlin, 2004.

[9] E. D. Dolan and J. J. Moré. Benchmarking optimization software with performance profiles. *Mathematical Programming*, 91(2):201–213, 2002.

[10] G. H. Golub and C. F. Van Loan. *Matrix Computations*. The John Hopkins University Press, Baltimore, USA, 1996.

[11] N. I. M. Gould and S. Lucidi and M. Roma and Ph. L. Toint. Solving the trust-region subproblem using the Lanczos method. *SIAM Journal on Optimization*, 9(2):504–525, 1999.

[12] N. I. M. Gould, D. Orban, A. Sartenaer and Ph. L. Toint. Sensitivity of trust-region algorithms on their parameters. *4OR, Quarterly Journal of the Italian, French and Belgian OR Socities*, 3(3):227–241, 2005.

[13] N. I. M. Gould, D. Orban, and Ph. L. Toint. CUTEr (and SifDec), a Constrained and Unconstrained Testing Environment, revisited. *ACM Transactions on Mathematical Software*, 29(4):373–394, 2003.

[14] S. Gratton, M. Mouffe, Ph. L. Toint and M. Weber-Mendonça. A recursive trust-region method in infinity norm for bound-constrained nonlinear optimization. Technical report no. 07/01, Department of Mathematics, University of Namur, Namur, Belgium, 2007.

[15] S. Gratton, A. Sartenaer and Ph. L. Toint. Recursive trust-region methods for multiscale nonlinear optimization. Technical report no. 04/06, Department of Mathematics, University of Namur, Namur, Belgium, 2004.

[16] A. Griewank. The modification of Newton's method for unconstrained optimization by bounding cubic terms. Technical Report NA/12 (1981), Department of Applied Mathematics and Theoretical Physics, University of Cambridge, United Kingdom, 1981.

[17] A. Griewank and Ph. L. Toint. Numerical experiments with partially separable optimization problems. *Numerical Analysis: Proceedings Dundee 1983*, pages 203–220. Springer, Lecture Notes in Mathematics vol. 1066, 1984.

[18] Yu. Nesterov. *Introductory Lectures on Convex Optimization*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 2004.

[19] Yu. Nesterov. Accelerating the cubic regularization of Newton's method on convex problems. *Mathematical Programming*, 112(1):159–181, 2008.

[20] Yu. Nesterov and B. T. Polyak. Cubic regularization of Newton's method and its global performance. *Mathematical Programming*, 108(1):177–205, 2006.

[21] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer-Verlag, New York, USA, 1999.

[22] M. Weiser, P. Deuflhard and B. Erdmann. Affine conjugate adaptive Newton methods for nonlinear elastomechanics. *Optimization Methods and Software*, 22(3):413–431, 2007.

Appendix A

Here we give the complete set of results from our tests. For each problem, in Table A.1 we report its number of variables (n), along with the number of iterations (= the number of function evaluations) required for convergence (iter), the number of gradient evaluations ($\#g$), and the best objective function value found (f; the subscript gives the base-10 exponent) for the four rival methods. The symbol > indicates that the iteration limit was exceeded.

		Trust-region			ACO with g-rule			ACO with s-rule			ACO with s/σ-rule		
Name	n	iter	$\#g$	f	iter	$\#g$	f	iter	$\#g$	f	iter	$\#g$	f
ALLINITU	4	8	8	5.74	16	9	5.74	16	9	5.74	16	9	5.74
ARGLINA	200	30	30	2.00_{+2}	8	8	2.00_{+2}	8	8	2.00_{+2}	8	8	2.00_{+2}
ARWHEAD	100	6	6	6.59_{-14}	6	6	8.79_{-14}	6	6	8.79_{-14}	6	6	8.79_{-14}
BARD	3	8	8	8.21_{-3}	8	8	8.21_{-3}	8	8	8.21_{-3}	8	8	8.21_{-3}
BDQRTIC	100	14	14	3.79_{+2}	10	10	3.79_{+2}	10	10	3.79_{+2}	10	10	3.79_{+2}
BEALE	2	9	7	7.55_{-14}	10	7	9.89_{-12}	10	7	9.89_{-12}	10	7	9.89_{-12}
BIGGS6	6	200	196	2.43_{-1}	66	47	1.66_{-10}	76	52	1.35_{-11}	76	52	1.35_{-11}
BOX3	3	8	8	2.03_{-11}	9	9	5.84_{-16}	9	9	5.84_{-16}	9	9	5.84_{-16}
BRKMCC	2	3	3	1.69_{-1}	4	4	1.69_{-1}	4	4	1.69_{-1}	4	4	1.69_{-1}
BROWNAL	200	11	11	5.49_{-20}	5	5	8.29_{-17}	3	3	1.47_{-9}	3	3	1.47_{-9}
BROWNBS	2	>	>	9.80_{+11}	28	27	2.17_{-29}	28	27	1.78_{-29}	28	27	1.78_{-29}
BROWNDEN	4	44	44	8.58_{+4}	9	9	8.58_{+4}	9	9	8.58_{+4}	9	9	8.58_{+4}
BROYDN7D	100	22	22	3.24_{+1}	25	17	3.01_{+1}	25	17	3.01_{+1}	25	17	3.01_{+1}
BRYBND	100	18	15	5.54_{-17}	16	10	5.32_{-17}	16	10	5.34_{-17}	16	10	5.32_{-17}
CHAINWOO	100	255	253	3.22_{+1}	60	42	1.00	60	41	1.00	60	42	1.00
CHNROSNB	50	63	61	1.28_{-18}	68	42	1.82_{-16}	68	42	1.80_{-16}	68	42	1.80_{-16}
CLIFF	2	28	28	2.00_{-1}	28	28	2.00_{-1}	28	28	2.00_{-1}	28	28	2.00_{-1}
CRAGGLVY	202	29	29	6.67_{+1}	14	14	6.67_{+1}	14	14	6.67_{+1}	14	14	6.67_{+1}
CUBE	2	32	28	1.00_{-19}	56	27	7.73_{-21}	56	27	7.73_{-21}	56	27	7.73_{-21}
CURLY10	50	15	15	-5.02_{+3}	27	16	-5.02_{+3}	27	16	-5.02_{+3}	27	16	-5.02_{+3}
CURLY20	50	12	12	-5.02_{+3}	29	15	-5.02_{+3}	29	15	-5.02_{+3}	29	15	-5.02_{+3}
CURLY30	50	19	18	-5.02_{+3}	30	15	-5.02_{+3}	30	15	-5.02_{+3}	30	15	-5.02_{+3}
DECONVU	61	46	38	1.40_{-9}	142	56	3.92_{-11}	131	55	1.27_{-10}	144	56	1.41_{-10}
DENSCHNA	2	6	6	2.21_{-12}	6	6	1.83_{-12}	6	6	1.83_{-12}	6	6	1.83_{-12}
DENSCHNB	2	6	6	1.04_{-13}	6	6	2.74_{-15}	6	6	2.74_{-15}	6	6	2.74_{-15}
DENSCHNC	2	11	11	2.18_{-20}	11	11	5.50_{-19}	11	11	5.50_{-19}	11	11	5.50_{-19}
DENSCHND	3	36	36	1.77_{-8}	50	33	8.74_{-9}	50	33	8.74_{-9}	50	33	8.74_{-9}
DENSCHNE	3	16	16	1.10_{-18}	24	14	9.92_{-14}	25	15	1.84_{-11}	25	15	1.84_{-11}
DENSCHNF	2	7	7	6.51_{-22}	7	7	6.93_{-22}	7	7	6.93_{-22}	7	7	6.93_{-22}
DIXMAANA	150	27	27	1.00	8	8	1.00	8	8	1.00	8	8	1.00
DIXMAANB	150	27	27	1.00	8	8	1.00	8	8	1.00	8	8	1.00
DIXMAANC	150	27	27	1.00	22	15	1.00	23	16	1.00	23	16	1.00
DIXMAAND	150	27	27	1.00	26	18	1.00	26	18	1.00	26	18	1.00
DIXMAANE	150	31	31	1.00	12	12	1.00	12	12	1.00	12	12	1.00
DIXMAANF	150	29	29	1.00	24	19	1.00	24	19	1.00	24	19	1.00
DIXMAANG	150	29	29	1.00	29	23	1.00	29	23	1.00	29	23	1.00
DIXMAANH	150	29	29	1.00	31	23	1.00	31	23	1.00	31	23	1.00
DIXMAANI	150	37	37	1.00	17	17	1.00	17	17	1.00	17	17	1.00
DIXMAANJ	150	38	36	1.00	33	28	1.00	33	28	1.00	33	28	1.00
DIXMAANK	150	35	34	1.00	35	28	1.00	35	28	1.00	35	28	1.00
DIXMAANL	150	41	39	1.00	37	29	1.00	37	29	1.00	37	29	1.00
DJTL	2	81	72	-8.95_{+3}	1655	1581	-8.95_{+3}	1655	1581	-8.95_{+3}	1655	1581	-8.95_{+3}
DQRTIC	100	594	594	1.20_{-7}	25	25	2.36_{-8}	25	25	2.36_{-8}	25	25	2.36_{-8}
EDENSCH	100	76	76	6.03_{+2}	12	12	6.03_{+2}	12	12	6.03_{+2}	12	12	6.03_{+2}
EG2	100	4	4	-9.89_{+1}	4	4	-9.89_{+1}	4	4	-9.89_{+1}	4	4	-9.89_{+1}
EIGENALS	110	23	23	2.83_{-15}	25	20	3.38_{-15}	25	20	4.56_{-15}	25	20	6.56_{-16}
EIGENBLS	110	88	83	8.84_{-15}	170	71	7.46_{-11}	202	78	1.40_{-12}	208	81	2.02_{-12}
EIGENCLS	132	46	44	1.02_{-15}	57	35	9.47_{-13}	57	35	1.82_{-11}	56	35	4.34_{-12}
ENGVAL1	100	19	19	1.09_{+2}	9	9	1.09_{+2}	9	9	1.09_{+2}	9	9	1.09_{+2}
ENGVAL2	3	14	14	9.71_{-17}	27	16	3.08_{-20}	27	16	3.08_{-20}	27	16	3.08_{-20}
ERRINROS	50	127	123	4.39_{+1}	66	39	4.39_{+1}	61	41	3.99_{+1}	61	41	3.99_{+1}
EXPFIT	2	10	8	2.41_{-1}	14	7	2.41_{-1}	14	7	2.41_{-1}	14	7	2.41_{-1}
EXTROSNB	100	1619	1591	7.96_{-9}	6826	1197	1.67_{-8}	6984	1219	1.58_{-8}	6768	1187	1.71_{-8}
FLETCBV2	100	3	3	-5.14_{-1}	5	5	-5.14_{-1}	5	5	-5.14_{-1}	5	5	-5.14_{-1}
FLETCBV3	50	>	>	-3.43	>	>	-1.17_{+2}	>	>	-1.17_{+2}	>	>	-1.18_{+2}
FLETCHBV	10	5012	5012	-1.99_{+6}	341	299	-2.17_{+6}	496	419	-2.28_{+6}	465	388	-2.22_{+6}
FLETCHCR	100	154	151	1.00_{-15}	230	154	6.20_{-17}	230	154	6.20_{-17}	230	154	6.20_{-17}
FMINSRF2	121	128	126	1.00	45	38	1.00	45	38	1.00	45	38	1.00
FMINSURF	121	67	66	1.00	43	36	1.00	43	36	1.00	43	36	1.00
FREUROTH	100	55	30	1.20_{+4}	17	12	1.20_{+4}	17	12	1.20_{+4}	17	12	1.20_{+4}
GENHUMPS	10	>	>	1.42_{+3}	7599	3977	1.32_{-10}	7442	3831	3.66_{-17}	7596	3983	1.96_{-10}

Table A.1: Comparison between the trust-region and ACO algorithms

		Trust-region			ACO with g-rule			ACO with s-rule			ACO with s/σ-rule		
Name	n	iter	$\#g$	f	iter	$\#g$	f	iter	$\#g$	f	iter	$\#g$	f
GENROSE	100	79	77	1.00	130	62	1.00	135	63	1.00	133	63	1.00
GENROSEB	500	663	660	1.00	647	290	1.00	652	294	1.00	657	294	1.00
GROWTHLS	3	142	134	1.00	115	69	1.00	113	71	1.00	113	71	1.00
GULF	3	248	246	2.96_{-11}	45	33	2.89_{-11}	45	33	2.90_{-11}	45	33	2.89_{-11}
HAIRY	2	103	100	2.00_{+1}	75	31	2.00_{+1}	81	33	2.00_{+1}	80	33	2.00_{+1}
HATFLDD	3	19	18	6.62_{-8}	26	23	6.62_{-8}	26	23	6.62_{-8}	26	23	6.62_{-8}
HATFLDE	3	20	19	5.12_{-7}	22	17	5.12_{-7}	22	17	5.12_{-7}	22	17	5.12_{-7}
HEART6LS	6	3369	3363	4.09_{-16}	407	310	4.02_{-14}	413	300	2.55_{-16}	426	304	1.32_{-14}
HEART8LS	8	103	99	1.42_{-19}	94	49	1.99_{-17}	98	48	8.26_{-18}	95	50	7.57_{-15}
HELIX	3	13	12	1.16_{-15}	21	12	3.13_{-13}	21	12	3.13_{-13}	21	12	3.13_{-13}
HIMMELBB	2	14	10	2.52_{-21}	21	7	9.71_{-26}	21	7	9.71_{-26}	21	7	9.71_{-26}
HUMPS	2	6015	6007	2.39_{-18}	1552	636	2.02_{-12}	1591	663	1.64_{-10}	1498	659	3.68_{-10}
HYDC20LS	99	>	>	5.37_{-1}	>	>	7.67_{-1}	>	>	7.75_{-1}	>	>	7.68_{-1}
JENSMP	2	10	10	1.24_{+2}	10	10	1.24_{+2}	10	10	1.24_{+2}	10	10	1.24_{+2}
KOWOSB	4	13	10	3.08_{-4}	10	8	3.08_{-4}	10	8	3.08_{-4}	10	8	3.08_{-4}
LIARWHD	100	35	35	3.87_{-14}	12	12	4.44_{-26}	12	12	4.44_{-26}	12	12	4.44_{-26}
LOGHAIRY	2	9349	9346	1.82_{-1}	2737	1208	1.82_{-1}	2696	1198	1.82_{-1}	2617	1145	1.82_{-1}
MANCINO	100	754	754	1.27_{-21}	27	11	1.35_{-21}	27	11	1.35_{-21}	27	11	1.16_{-21}
MEXHAT	2	37	28	-4.00_{-2}	98	27	-4.00_{-2}	98	27	-4.00_{-2}	98	27	-4.00_{-2}
MEYER3	3	>	>	9.02_{+1}	361	234	8.79_{+1}	422	269	8.79_{+1}	420	268	8.79_{+1}
MOREBV	100	95	95	2.87_{-7}	59	59	4.16_{-7}	102	102	2.86_{-7}	102	102	2.86_{-7}
MSQRTALS	100	22	19	1.42_{-14}	18	15	5.84_{-15}	18	15	4.73_{-15}	18	15	5.77_{-15}
MSQRTBLS	100	20	17	3.32_{-12}	18	15	2.07_{-15}	18	15	1.99_{-15}	18	15	2.06_{-15}
NONCVXU2	100	714	714	2.33_{+2}	51	42	2.32_{+2}	49	41	2.32_{+2}	49	41	2.32_{+2}
NONCVXUN	100	689	689	2.35_{+2}	43	33	2.33_{+2}	47	38	2.33_{+2}	47	38	2.33_{+2}
NONDIA	100	7	7	1.50_{-18}	11	8	2.90_{-18}	11	8	2.90_{-18}	11	8	2.90_{-18}
NONDQUAR	100	17	17	2.58_{-15}	16	16	1.42_{-8}	62	38	1.81_{-6}	62	38	1.81_{-6}
NONMSQRT	100	3838	3838	1.81_{+1}	2700	2535	1.81_{+1}	2959	2926	1.81_{+1}	3809	3777	1.81_{+1}
OSBORNEA	5	111	104	5.46_{-5}	468	266	4.69_{-2}	697	465	4.70_{-2}	991	607	4.70_{-2}
OSBORNEB	11	21	20	4.01_{-2}	21	18	4.01_{-2}	21	18	4.01_{-2}	21	18	4.01_{-2}
OSCIPATH	8	3910	3829	5.61_{-8}	8172	2474	8.67_{-8}	8115	2463	8.75_{-8}	8209	2485	8.22_{-8}
PALMER5C	6	64	64	2.13	8	8	2.13	8	8	2.13	8	8	2.13
PALMER6C	8	512	512	1.64_{-2}	124	124	1.64_{-2}	239	239	1.64_{-2}	239	239	1.64_{-2}
PALMER7C	8	1243	1243	6.02_{-1}	55	55	6.02_{-1}	162	162	6.02_{-1}	162	162	6.02_{-1}
PALMER8C	8	590	590	1.60_{-1}	303	303	1.60_{-1}	311	311	1.60_{-1}	311	311	1.60_{-1}
PARKCH	15	1	1	4.73_{-7}	1	1	4.73_{-7}	42	25	1.62_{+3}	56	26	1.62_{+3}
PENALTY1	100	610	602	9.02_{-4}	85	35	9.02_{-4}	85	35	9.02_{-4}	85	35	9.02_{-4}
PENALTY2	200	2	2	4.71_{+13}	4	4	4.71_{+13}	11	11	4.71_{+13}	11	11	4.71_{+13}
PENALTY3	200	1	1	1.16_{+6}	1	1	1.16_{+6}	29	14	1.00_{-3}	24	14	9.97_{-4}
PFIT1LS	3	331	315	1.66_{-12}	870	233	1.07_{-13}	870	233	1.07_{-13}	870	233	1.07_{-13}
PFIT2LS	3	104	92	2.71_{-13}	246	69	1.46_{-16}	247	69	1.16_{-16}	245	69	2.55_{-16}
PFIT3LS	3	131	119	3.44_{-14}	574	168	7.03_{-18}	581	166	1.61_{-13}	581	166	4.21_{-13}
PFIT4LS	3	227	212	4.62_{-16}	1319	417	2.06_{-14}	1319	419	1.01_{-13}	1316	419	9.86_{-14}
POWELLSG	4	5	5	1.93_{-30}	5	5	1.81_{-12}	16	16	4.54_{-9}	16	16	4.54_{-9}
POWER	100	28	28	1.25_{-9}	24	24	1.61_{-9}	24	24	1.61_{-9}	24	24	1.61_{-9}
QUARTC	100	566	566	0.00	12	12	1.42_{-25}	25	25	2.36_{-8}	25	25	2.36_{-8}
ROSENBR	2	4	4	1.71_{-32}	5	5	1.07_{-15}	37	20	1.80_{-12}	37	20	1.80_{-12}
S308	2	1	1	0.00	1	1	0.00	10	10	7.73_{-1}	10	10	7.73_{-1}
SBRYBND	100	>	>	6.76_{+6}	>	>	6.76_{+6}	>	>	6.76_{+6}	>	>	2.44_{+2}
SCHMVETT	100	6	6	-2.94_{+2}	5	5	-2.94_{+2}	5	5	-2.94_{+2}	5	5	-2.94_{+2}
SENSORS	100	25	23	-1.97_{+3}	34	23	-1.94_{+3}	41	25	-1.94_{+3}	34	23	-1.94_{+3}
SINEVAL	2	57	53	1.58_{-25}	94	43	4.77_{-14}	94	43	4.80_{-14}	94	43	4.77_{-14}
SINQUAD	100	16	16	-4.01_{+3}	15	9	-4.01_{+3}	15	9	-4.01_{+3}	15	9	-4.01_{+3}
SISSER	2	13	13	1.07_{-8}	13	13	1.14_{-8}	13	13	1.14_{-8}	13	13	1.14_{-8}
SNAIL	2	12	12	2.54_{-15}	7	7	8.46_{-16}	100	63	2.16_{-17}	100	63	2.16_{-17}
SPARSINE	100	11	11	3.63_{-17}	25	17	1.84_{-17}	25	17	2.17_{-17}	25	17	1.86_{-17}
SPARSQUR	100	17	17	2.32_{-8}	17	17	7.78_{-9}	17	17	7.78_{-9}	17	17	7.78_{-9}
SPMSRTLS	100	12	12	9.34_{-14}	17	16	3.90_{-13}	17	16	4.53_{-13}	17	16	4.53_{-13}
SROSENBR	100	7	7	9.08_{-13}	9	9	6.52_{-16}	9	9	6.52_{-16}	9	9	6.52_{-16}
STREG	4	>	>	1.00_{+20}	104	52	2.95_{-13}	214	147	9.88_{-20}	214	147	9.88_{-20}
TOINTGOR	50	41	41	1.37_{+3}	9	9	1.37_{+3}	9	9	1.37_{+3}	9	9	1.37_{+3}
TOINTGSS	100	32	32	1.01_{+1}	11	10	1.02_{+1}	11	10	1.02_{+1}	11	10	1.02_{+1}
TOINTPSP	50	35	35	2.26_{+2}	30	19	2.26_{+2}	30	19	2.26_{+2}	30	19	2.26_{+2}
TQUARTIC	100	78	75	1.09_{-20}	18	13	1.62_{-23}	18	13	1.62_{-23}	18	13	1.62_{-23}
VARDIM	200	10	10	1.05_{-25}	3	3	7.19_{-26}	30	30	6.79_{-25}	30	30	6.79_{-25}
VAREIGVL	50	20	17	7.98_{-10}	14	14	8.39_{-10}	14	14	7.97_{-10}	14	14	7.99_{-10}
VIBRBEAM	8	>	>	4.54	261	144	1.56_{-1}	304	188	1.56_{-1}	257	140	1.56_{-1}
WATSON	12	11	11	9.13_{-9}	10	10	1.86_{-9}	10	10	1.83_{-9}	10	10	1.83_{-9}
WOODS	4	71	68	2.18_{-21}	69	39	6.99_{-19}	69	39	6.99_{-19}	69	39	6.99_{-19}
YFITU	3	246	245	6.79_{-13}	59	46	6.82_{-13}	59	46	6.82_{-13}	59	46	6.82_{-13}

Table A.1: Comparison between the trust-region and ACO algorithms