Math. Program., Ser. A
DOI 10.1007/s10107-012-0617-9

FULL LENGTH PAPER

On the complexity of finding first-order critical points in constrained nonlinear optimization

Coralia Cartis · Nicholas I. M. Gould · Philippe L. Toint

Received: 14 April 2011 / Accepted: 20 November 2012

Abstract The complexity of finding ϵ-approximate first-order critical points for the general smooth constrained optimization problem is shown to be no worse that $O(\epsilon^{-2})$ in terms of function and constraints evaluations. This result is obtained by analyzing the worst-case behaviour of a first-order short-step homotopy algorithm consisting of a feasibility phase followed by an optimization phase, and requires minimal assumptions on the objective function. Since a bound of the same order is known to be valid for the unconstrained case, this leads to the conclusion that the presence of possibly nonlinear/nonconvex inequality/equality constraints is irrelevant for this bound to apply.

Keywords Evaluation complexity · Worst-case analysis · Constrained nonlinear optimization

Mathematics Subject Classification 68Q25 · 90C60 · 90C30

C. Cartis
School of Mathematics, University of Edinburgh,
The King's Buildings, Edinburgh EH9 3JZ, Scotland, UK
e-mail: coralia.cartis@ed.ac.uk

N. I. M. Gould
Computational Science and Engineering Department,
Rutherford Appleton Laboratory, Oxfordshire, Chilton OX11 0QX, UK
e-mail: nick.gould@sftc.ac.uk

Ph. L. Toint (✉)
Namur Center for Complex Systems (naXys), Department of Mathematics,
FUNDP-University of Namur, 61, rue de Bruxelles, 5000 Namur, Belgium
e-mail: philippe.toint@fundp.ac.be

Published online: 08 December 2012

1 Introduction

Evaluation complexity analysis for nonconvex smooth optimization problems has recently been a very active area of research and has covered both standard methods for the unconstrained case, such as steepest-descent (see [2,11,14]), trust-region methods (see [10]), Newton's algorithm (see [2]) or finite-difference and derivative-free approaches (see [8,15]), along with newer techniques involving regularization (see [4,12]). The issue considered in this paper is to bound the number of objective function and constraints evaluations that are necessary to find an approximate first-order critical (i.e., KKT) point for the problem

$$\operatorname*{minimize}_{x\in\mathbb{R}^n} \; f(x) \text{ such that } c_E(x) = 0 \text{ and } c_I(x) \geq 0, \tag{1.1}$$

where f, c_E and c_I are continuously differentiable possibly nonconvex functions from (possibly a subdomain of) $\mathbb{R}^n$ to $\mathbb{R}$, $\mathbb{R}^m$ and $\mathbb{R}^p$, with Lipschitz continuous gradient and Jacobians, respectively.

In the unconstrained case, namely,

$$\operatorname*{minimize}_{x\in\mathbb{R}^n} \; f(x),$$

an approximate critical point for this problem is defined as a point x such that

$$\|g(x)\| \leq \epsilon, \tag{1.2}$$

where $\epsilon \in (0, 1)$ is a user-specified accuracy, $\|\cdot\|$ is the Euclidean norm and $g(x) \overset{\text{def}}{=} \nabla_x f(x)$. For steepest-descent methods with exact or inexact linesearches and for trust-region algorithms with linear models, it has been shown that the maximum number of objective function (and gradient) evaluations is bounded above by

$$\left\lceil \frac{\kappa}{\epsilon^2} \right\rceil \tag{1.3}$$

for some constant $\kappa > 0$ independent of ϵ but dependent on the Lipschitz constant of the gradient and other problem and algorithm parameters [10,11]. Moreover, Cartis et al. [2] proved that this order in ϵ is sharp for steepest-descent methods with inexact linesearches. A first extension of this kind of result to constrained problems was provided by Cartis et al. [9], where it is shown that (1.3) also holds for a first-order projection-based method for the more general problem

$$\operatorname*{minimize}_{x\in\mathbb{R}^n} \; f(x) \text{ such that } x \in \mathcal{C}, \tag{1.4}$$

where $\mathcal{C}$ is a convex set and where (1.2) is suitably adapted to define an ϵ-approximate first-order critical point for the constrained problem (1.4). More recently, Cartis et al. [5] considered a first-order exact penalty function algorithm for solving the general nonlinearly constrained nonconvex optimization problem (1.1). They proved that the complexity of finding an ϵ-approximate first-order critical (KKT) point for (1.1) is given by an appropriate variant of (1.3) when the penalty parameters are uniformly bounded above independently of ϵ, and is bounded above by $O(\epsilon^{-5})$ otherwise.[1] Though it is reasonable to expect the penalty parameters to be finite due to the exactness of the penalty function, this is not always the case and even when true, it is not known a priori. Thus only the worse bound $O(\epsilon^{-5})$ can be guaranteed to apply. In addition, the derivation of these bounds unfortunately requires the undesirable assumption that the objective function f is bounded below on the whole of $\mathbb{R}^n$, which is well-known to fail even for nonconvex quadratic programming problems [13, p. 500].

In this paper, we present a novel first-order target-following algorithm for (1.1) that can unprecedentedly be shown to satisfy a bound of order (1.3) for achieving ϵ-approximate first-order criticality for (1.1). Despite being mainly a theoretical approach, this stronger result only requires f to be bounded in a small, 1-neighbourhood of the feasible set, which is considerably weaker than assuming this property on the whole space. In particular, it is satisfied for the quadratic programming case if the feasible set is bounded.

The paper is organized as follows. The Short-Step Steepest-Descent algorithm for approximately solving the equality constrained problem is introduced in Sect. 2, and its complexity is shown in Sect. 3 to be bounded above by (1.3). Section 4 briefly covers the simple extension of this result to the general problem (1.1). Some conclusions and perspectives are given in Sect. 5.

2 The Short-Step Steepest-Descent algorithm for the equality constrained problem

For the sake of simplicity, we start by considering the equality-constrained problem

$$\underset{x\in\mathbb{R}^n}{\text{minimize}} \;\; f(x) \text{ such that } c(x) = 0, \tag{2.1}$$

where $f : \mathbb{R}^n \longrightarrow \mathbb{R}$ and $c : \mathbb{R}^n \longrightarrow \mathbb{R}^m$. We define a slightly larger set than the set of approximately feasible points, namely,

$$\mathcal{C}_1 \overset{\text{def}}{=} \{x \in \mathbb{R}^n : \; \|c(x)\| < \kappa_{c_1}\}, \tag{2.2}$$

where $\kappa_{c_1} > \epsilon$ is a small constant independent of ϵ and where $\epsilon \in (0, 1]$ is the accuracy tolerance to which we aim to solve (2.1). We assume that

A.1: The function c is continuously differentiable on $\mathbb{R}^n$ and f is continuously differentiable in the set

[1] Note that the reason for the worsening of the bound is not the rate at which the penalty parameter is increased, but the value it must reach to ensure approximate first-order criticality with respect to the feasibility measure. The latter depends on ϵ and comes into the complexity bound through the Lipschitz constant of the (subgradient of the) merit function. In fact, it is our experience that this is a disadvantage of all commonly-used parametrized methods for constrained problems when estimating their worst-case evaluation complexity, which we have only been able to overcome through the two-phase target-following approach proposed here.

$$\mathcal{C}_2 \overset{\text{def}}{=} \mathcal{C}_1 + \mathcal{B}(0, \delta\Delta_1), \tag{2.3}$$

where $\delta > 1$ is a constant slightly larger than 1, Δ_1 is the initial choice of trust-region radius in our algorithm and $\mathcal{B}(0, \delta\Delta_1)$ is the open Euclidean ball centred at the origin and of radius $\delta\Delta_1$.

The algorithm we now describe consists of two phases. In the first, a first-order algorithm is applied to minimize $\|c(x)\|$ (independently of the objective function f), resulting in a point which is either (approximately) feasible, or is an approximate infeasible stationary point of $\|c(x)\|$. This last outcome is not desirable if one wishes to solve (2.1), but cannot be avoided by any algorithm not relying on global minimization. If an (approximate) feasible point has been found, Phase 2 of the algorithm then performs short steps along generalized steepest-descent directions so long as first-order criticality is not satisfied. These steps are computed by attempting to preserve feasibility of the iterates while producing values of the objective function that are close to a sequence of decreasing 'targets'.

Both phases rely on the first-order trust-region algorithm[2] proposed in [5], which can be used to solve the problem

$$\underset{x\in\mathbb{R}^n}{\text{minimize}} \;\; \theta\big(u(x)\big), \tag{2.4}$$

where θ is a (potentially nonsmooth) convex function from $\mathbb{R}^p$ into $\mathbb{R}$ and $u(x)$ is a (potentially nonconvex) continuously differentiable function from $\mathbb{R}^n$ into $\mathbb{R}^p$ with Jacobian $A(x)$. In this algorithm, a 'Cauchy step' s_k is obtained from the iterate x_k by solving the linearized model problem

$$\underset{s\in\mathbb{R}^n}{\text{minimize}} \;\; \theta\big(u(x_k) + A(x_k)s\big) \text{ such that } \|s\| \leq \Delta_k, \tag{2.5}$$

where Δ_k is a trust-region radius. Because θ is convex and its argument in (2.5) linear, this problem is computationally tractable. The rest of the algorithm specification follows standard trust-region technology.

We now return to the solution of problem (2.1) proper, and define the merit function[3]

$$\phi(x, t) \overset{\text{def}}{=} \|c(x)\| + |f(x) - t|, \tag{2.6}$$

[2] We make this choice for simplicity of exposition, but other methods can be considered with similar results. In particular, the quadratic regularization technique of Cartis et al. [5] or the trust-region technique proposed by Byrd et al. [1] are also adequate.

[3] Note that the merit function (2.6) can be viewed as an exact penalty function with penalty parameter equal to 1, that penalizes in equal measure both the distance from feasibility and from some target value for the objective. This allows us to keep proximity simultaneously to the constraints and the set target values for the objective. As it is only the target values t that we can update freely and monitor to give sufficient decrease, we must ensure that the objective function values stay close to the targets in order to guarantee termination and good complexity of the algorithm [see (2.17) and (3.12)]. Furthermore, approximate critical points x of $\phi(x, t)$ correspond to approximate KKT points of (2.1); see Lemma 3.5.

where t is meant as a "target" for $f(x)$.[4] We also define the local linearizations of $\|c(x)\|$ and $\phi(x,t)$ given by

$$\ell_c(x,s) \overset{\text{def}}{=} \|c(x)+J(x)s\| \;\text{ and }\; \ell_\phi(x,t,s) \overset{\text{def}}{=} \ell_c(x,s) + |f(x)+\langle g(x),s\rangle - t|,$$

(where $\langle\cdot,\cdot\rangle$ is the Euclidean inner product). The value of the decrease of the linearized model in a ball of unit radius may then be considered as a first-order criticality measure for the problems of minimizing $\|c(x)\|$ and $\phi(x,t)$, yielding the measures

$$\psi(x) \overset{\text{def}}{=} \ell_c(x,0) - \min_{\|s\|\le 1} \ell_c(x,s) \;\text{ and }\; \chi(x,t) \overset{\text{def}}{=} \ell_\phi(x,t,0) - \min_{\|s\|\le 1} \ell_\phi(x,t,s).$$

Note that $\psi(x)$ is zero if and only if x is first-order critical for the problem of minimizing $\|c(x)\|$, while $\chi(x,t)$ is zero if and only if (x,t) is a first-order critical point for the problem

$$\underset{x\in\mathbb{R}^n}{\text{minimize}} \;\; \phi(x,t), \tag{2.7}$$

where t is fixed; see equations (2.2)–(2.3) in [5].

In Phase 1 of our algorithm (aiming for feasiblility), we apply the first-order trust-region algorithm of [5] by identifying

$$p = m, \quad u(x) = c(x), \quad \text{and} \;\; \theta(\cdot) = \|\cdot\|, \tag{2.8}$$

in (2.4), yieding $\theta\big(u(x)\big) = \|c(x)\|$. For Phase 2 (the optimality phase), we choose in (2.4), for t fixed,

$$p = m+1, \quad u(x) = (c(x), f(x)-t) \;\text{ and }\; \theta(\cdot) = \|\cdot\| + |\cdot|, \tag{2.9}$$

which gives $\theta\big(u(x)\big) = \phi(x,t)$. Note that $\theta(\cdot)$ is clearly convex with global Lipschitz constant equal to one in both cases.

[4] Alternatively, we could use the merit function $\|c(x)\| + \max(f(x)-t, 0)$ instead of $\phi(x,t)$, which has the advantage that it allows $f(x)$ to decrease (possibly by large amounts) below the target t, while keeping the feasibility term in check. A complexity bound of order (1.3) for a Short-Step Steepest-Descent variant using this merit function can be shown similarly to the results that follow.

We are now ready to formalize our Short-Step Steepest-Descent algorithm.

Algorithm 2.1: The Short-Step Steepest-Descent algorithm.

Let $\kappa_f \in (0,1)$, $\epsilon \in (0,1]$ and $\Delta_1 > 0$ be given, together with a starting point x_0.

Phase 1:

Starting from x_0, minimize $\|c(x)\|$ (using (2.4) and (2.8) and the trust-region method of [5]) until a point x_1 is found such that

$$\psi(x_1) \leq \epsilon.$$

If $\|c(x_1)\| > \kappa_f \epsilon$, terminate [locally infeasible].

Phase 2:

1. Set $t_1 = \|c(x_1)\| + f(x_1) - \epsilon$ and $k = 1$.
2. While $\chi(x_k, t_k) \geq \epsilon$,
   - (a) Compute a first-order step s_k by solving

     $$\underset{s \in \mathbb{R}^n}{\text{minimize}} \;\; \ell_\phi(x_k, t_k, s) \;\; \text{such that} \;\; \|s\| \leq \Delta_k. \tag{2.10}$$

   - (b) Compute $\phi(x_k + s_k, t_k)$ and define

     $$\rho_k = \frac{\phi(x_k, t_k) - \phi(x_k + s_k, t_k)}{\ell_\phi(x_k, t_k, 0) - \ell_\phi(x_k, t_k, s_k)}. \tag{2.11}$$

     If $\rho_k \geq \eta$, then $x_{k+1} = x_k + s_k$; else $x_{k+1} = x_k$.
   - (c) Set

     $$\Delta_{k+1} = \begin{cases} \Delta_k & \text{if } \rho_k \geq \eta \quad [k \text{ successful}] \\ \gamma\Delta_k & \text{if } \rho_k < \eta \quad [k \text{ unsuccessful}] \end{cases} \tag{2.12}$$

   - (d) If $\rho_k \geq \eta$, set

     $$t_{k+1} = \begin{cases} t_k - \phi(x_k, t_k) + \phi(x_{k+1}, t_k) & \text{if} \;\; f(x_{k+1}) \geq t_k, \\ 2f(x_{k+1}) - t_k - \phi(x_k, t_k) + \phi(x_{k+1}, t_k) & \text{if} \;\; f(x_{k+1}) < t_k. \end{cases} \tag{2.13}$$

     Otherwise, set $t_{k+1} = t_k$.
   - (e) Increment k by one and return to Step 2.
3. Terminate [(approximately) first-order critical]

We next extract from [5] a property which is crucial for proving that Phase 2 of Algorithm 2.1 is well-defined.

Lemma 2.1 *Suppose that A.1 holds. If $x_k \in \mathcal{C}_1$, where $\mathcal{C}_1$ is defined in* (2.2)*, then the model decrease satisfies*

$$\ell_\phi(x_k, t_k, 0) - \ell_\phi(x_k, t_k, s_k) \geq \min(\Delta_k, 1)\, \chi(x_k, t_k). \tag{2.14}$$

Proof (Note that requiring $x_k \in \mathcal{C}_1$ in Phase 2 implies that $x_k \in \mathcal{C}_2$ due to (2.3). Thus A.1 provides that f is differentiable at x_k and so the model $\ell_\phi(x_k, t_k, s)$ can be constructed.) Apply Lemma 2.1 in [5] with $h \overset{\text{def}}{=} \|\cdot\| + |\cdot|$ and $\Phi_h(x) \overset{\text{def}}{=} \phi(x, t_k)$ considered as a function of x only. □

The next lemma proves that not only does x_k belong to $\mathcal{C}_1$ so that Phase 2 is well-defined, but it remains approximately feasible for all Phase 2 iterations, and the objective function values stay close to their targets.

Lemma 2.2 *Suppose that A.1 holds. On each Phase 2 iteration $k \geq 1$ of Algorithm* 2.1*, we have*

$$f(x_k) - t_k > 0, \tag{2.15}$$

$$\phi(x_k, t_k) = \epsilon, \tag{2.16}$$

$$|f(x_k) - t_k| \leq \epsilon, \tag{2.17}$$

$$\|c(x_k)\| \leq \epsilon, \tag{2.18}$$

and so $x_k \in \mathcal{C}_1$.

Proof Firstly, note that (2.6) and (2.16) imply (2.17) and (2.18); the latter implies $x_k \in \mathcal{C}_1$ due to (2.2). Thus it remains to prove (2.15) and (2.16). The proof of these relations is by induction on k. For $k = 1$, recall that we only enter Phase 2 of the algorithm if $\|c(x_1)\| \leq \kappa_f \epsilon < \epsilon$, which gives (2.15) and (2.16) for $k = 1$, due to the particular choice of t_1. [Also, (2.14) holds at $k = 1$ and ρ_1 in (2.11) is well-defined.]

Now let $k > 1$ and assume that (2.15) and (2.16) are satisfied, and so

$$\phi(x_k, t_k) = \epsilon. \tag{2.19}$$

If k is an unsuccessful iteration, $x_{k+1} = x_k$ and $t_{k+1} = t_k$ and so (2.15) and (2.16) continue to hold at x_{k+1}. It remains to consider the case when k is successful. Recall that (2.19) implies $\|c(x_k)\| \leq \epsilon$ and $x_k \in \mathcal{C}_1$ due to (2.2), and so (2.14) holds. Thus, since we have not terminated, (2.11) has a positive denominator, which together with k being successful so that $\rho_k \geq \eta$, implies

$$\phi(x_k, t_k) > \phi(x_{k+1}, t_k).$$

This and (2.13) immediately give that $f(x_{k+1}) - t_{k+1} > 0$ so that (2.15) holds at $k+1$. Using the latter and (2.6), we deduce

$$\phi(x_{k+1}, t_{k+1}) = \|c(x_{k+1})\| + f(x_{k+1}) - t_k + (t_k - t_{k+1}). \tag{2.20}$$

Consider first the case when $f(x_{k+1}) \geq t_k$. Then, using (2.20) and (2.13), we obtain that

$$\phi(x_{k+1}, t_{k+1}) = \phi(x_{k+1}, t_k) + \phi(x_k, t_k) - \phi(x_{k+1}, t_k) = \phi(x_k, t_k).$$

If $f(x_{k+1}) < t_k$, we have that

$$\begin{aligned}\phi(x_{k+1}, t_{k+1}) &= \|c(x_{k+1})\| - f(x_{k+1}) + t_k + \phi(x_k, t_k) - \phi(x_{k+1}, t_k)\\ &= \phi(x_{k+1}, t_k) + \phi(x_k, t_k) - \phi(x_{k+1}, t_k)\\ &= \phi(x_k, t_k),\end{aligned}$$

where we again used (2.20) and (2.13). Combining the two cases and using (2.19), we then deduce that

$$\phi(x_{k+1}, t_{k+1}) = \phi(x_k, t_k) = \epsilon,$$

and thus (2.16) holds at $k+1$. This concludes the induction step and also the proof.

Since Algorithm 2.1 makes no pretense of being practical, we have written Steps 2.2.b and 2.2.c by only using the constants

$$0 < \eta < 1, \quad \text{and} \quad 0 < \gamma < 1,$$

instead of the more usual $\eta_1 \le \eta_2$ and $\gamma_1 \le \gamma_2$, a simplified choice which is allowed in the standard trust-region case, including that studied in [5].[5] Note that Algorithm 2.1 requires one evaluation of the objective function and its gradient and one evaluation of the constraint's function and its Jacobian per iteration.

Note also that one could also consider using the ARC algorithm (see [3]) to minimize $\|c(x)\|^2$ to find x_1 such that $\|J(x_1)^T c(x_1)\| \le \epsilon$. We do not consider this (potentially more efficient) possibility here because it would require stronger assumptions on the constraint function c.

3 Complexity of Algorithm 2.1 for the equality constrained problem

Before analyzing the complexity of Algorithm 2.1, we state our assumptions formally (in addition to A.1):

A.2: $J(x)$ is globally Lipschitz continuous in $\mathbb{R}^n$ with Lipschitz constant bounded above by $L_J > 0$, and $g(x)$ is Lipschitz continuous in $\mathcal{C}_2$ with Lipschitz constant bounded above by $L_g \ge 1$, where $\mathcal{C}_2$ is defined in (2.3).

A.3: The objective function is bounded above and below in $\mathcal{C}_1$, where $\mathcal{C}_1$ is defined in (2.2), that is there exist constants f_{low} and $f_{\text{up}} \ge f_{\text{low}} + 1$ such that

$$f_{\text{low}} \le f(x) \le f_{\text{up}} \quad \text{for all } x \in \mathcal{C}_1.$$

We start our analysis by exploiting the results of [5] and bounding the number of Phase 1 iterations.

Lemma 3.1 *Suppose that A.1 and A.2 hold. Then, at most*

$$\left\lceil \|c(x_0)\| \frac{\kappa_1}{\epsilon^2} \right\rceil \tag{3.1}$$

evaluations of $c(x)$ *and its derivatives are needed to complete Phase 1, for some* $\kappa_1 > 0$ *independent of* ϵ *and* x_0.

[5] By selecting $\eta_1 = \eta_2$ and $\gamma_1 = \gamma_2$ in this reference.

Proof Apply Theorem 2.4 in [5] with $h \overset{\text{def}}{=} \|\cdot\|$, $\Phi_h(x) \overset{\text{def}}{=} \|c(x)\|$, $L_h = 1$, $\eta_1 = \eta_2 \overset{\text{def}}{=} \eta$ and $\gamma_1 = \gamma_2 \overset{\text{def}}{=} \gamma$.

We now use Lemma 2.1 to lower bound the trust-region radius.

Lemma 3.2 *Suppose that A.1 and A.2 hold. Then any Phase 2 iteration $k \geq 1$ satisfying $\chi(x_k, t_k) \geq \epsilon$ and*

$$\Delta_k \leq \frac{(1-\eta)\epsilon}{L_g + \frac{1}{2}L_J} \tag{3.2}$$

is successful in the sense of (2.12). *Furthermore, while $\chi(x_k, t_k) \geq \epsilon$, we have*

$$\Delta_k \geq \kappa_\Delta \epsilon, \text{ for all Phase 2 iterations } k \geq 1, \tag{3.3}$$

where

$$\kappa_\Delta \overset{\text{def}}{=} \min\left(\Delta_1, \frac{(1-\eta)\gamma}{L_g + \frac{1}{2}L_J}\right) \tag{3.4}$$

is a constant independent of ϵ.

Proof From (2.11) and (2.6), we have

$$\begin{aligned}
|\rho_k - 1| &= \frac{\left|\phi(x_k + s_k, t_k) - \ell_\phi(x_k, t_k, s_k)\right|}{\ell_\phi(x_k, t_k, 0) - \ell_\phi(x_k, t_k, s_k)} \\
&= \frac{|\|c(x_k + s_k)\| - \|c(x_k) + J(x_k)s_k\| + |f(x_k + s_k) - t_k| - |f(x_k) + \langle g(x_k), s_k\rangle - t_k||}{\ell_\phi(x_k, t_k, 0) - \ell_\phi(x_k, t_k, s_k)} \\
&\leq \frac{|\|c(x_k + s_k)\| - \|c(x_k) + J(x_k)s_k\|| + |f(x_k + s_k) - f(x_k) - \langle g(x_k), s_k\rangle|}{\ell_\phi(x_k, t_k, 0) - \ell_\phi(x_k, t_k, s_k)}.
\end{aligned}$$

We have the Taylor expansions $f(x_k + s_k) = f(x_k) + g(\xi_k)^T s_k$ for some $\xi_k \in [x_k, x_k + s_k]$, and $c(x_k + s_k) = c(x_k) + \int_0^1 J(x_k + ts_k)s_k dt$. Lemma 2.2 implies $x_k \in \mathcal{C}_1$, and $\|\xi_k - x_k\| \leq \|s_k\| \leq \Delta_k \leq \Delta_1$ (as the radius is never increased in Phase 2) implies that $\xi_k,\ x_k + s_k \in \mathcal{C}_2$. Thus A.2 can be safely applied for these points, which together with the Taylor expansions, gives

$$\begin{aligned}
|f(x_k + s_k) - f(x_k) - \langle g(x_k), s_k\rangle| &\leq L_g \|s_k\|^2 \text{ and} \\
|\|c(x_k + s_k)\| - \|c(x_k) + J(x_k)s_k\|| &\leq \frac{1}{2}L_J \|s_k\|^2.
\end{aligned}$$

Thus, from (2.14) and $\|s_k\| \leq \Delta_k$, we deduce

$$|\rho_k - 1| \leq \frac{\left(L_g + \frac{1}{2}L_J\right)\Delta_k^2}{\min(\Delta_k, 1)\,\chi(x_k, t_k)} \leq \frac{\left(L_g + \frac{1}{2}L_J\right)}{\epsilon}\Delta_k,$$

where to obtain the second inequality, we used $\chi(x_k, t_k) \geq \epsilon$ and $\Delta_k \leq 1$, where the latter follows from (3.2), $L_g \geq 1$ and $\epsilon \in (0, 1]$. Finally, (3.2) implies $|\rho_k - 1| \leq 1 - \eta$, which gives that k is successful due to (2.12).

Now whenever (3.2) holds, (2.12) sets $\Delta_{k+1} = \Delta_k$. This implies that when $\Delta_1 \geq \gamma(1-\eta)\epsilon/(L_g + \frac{1}{2}L_J)$, we have $\Delta_k \geq \gamma(1-\eta)\epsilon/(L_g + \frac{1}{2}L_J)$ for all k, where the factor γ is introduced for the case when Δ_k is greater than $(1-\eta)\epsilon/(L_g + \frac{1}{2}L_J)$ and iteration k is unsuccessful. Applying again the implication resulting from (3.2) and (2.12) for $k = 1$, we deduce (3.3) when $\Delta_1 < \gamma(1-\eta)\epsilon/(L_g + \frac{1}{2}L_J)$ since $\gamma \in (0, 1)$ and $\epsilon \in (0, 1]$. □

We now bound the total number of unsuccessful iterations in the course of Phase 2.

Lemma 3.3 *There are at most $O(|\log \epsilon|)$ unsuccessful iterations in Phase 2 of Algorithm 2.1.*

Proof Note that (2.12) implies that the trust-region radius is never increased, and therefore Lemma 3.2 guarantees that all iterations must be successful once Δ_1 has been reduced (by a factor γ) enough times to ensure (3.2). Hence there are at most

$$\left\lceil \frac{1}{|\log \gamma|} \left| \log \epsilon + \log(1-\eta) - \log \Delta_1 - \log\left(L_g + \frac{1}{2}L_J\right) \right| \right\rceil = O(|\log \epsilon|) \quad (3.5)$$

unsuccessful iterations during the complete execution of the Phase 2. □

The next lemma proves that the targets t_k decrease by a quantity bounded below by a multiple of ϵ^2 at every successful iteration.

Lemma 3.4 *Suppose A.1 and A.2 hold. Then on each successful Phase 2 iteration $k \geq 1$, we have*

$$\phi(x_k + s_k, t_k) \leq \phi(x_k, t_k) - \kappa_C \epsilon^2 \quad (3.6)$$

and

$$t_k - t_{k+1} \geq \kappa_C \epsilon^2 \quad (3.7)$$

where

$$\kappa_C \overset{\text{def}}{=} \eta \kappa_\Delta \quad (3.8)$$

and κ_Δ is defined in (3.4), independently of ϵ.

Proof From (2.11) and k being successful, we deduce

$$\phi(x_k, t_k) - \phi(x_k + s_k, t_k) \geq \eta \left[\ell_\phi(x_k, t_k, 0) - \ell_\phi(x_k, t_k, s_k)\right] \geq \eta \min(\Delta_k, 1)\,\epsilon,$$

where to obtain the second inequality, we used (2.14) and $\chi(x_k, t_k) \geq \epsilon$. Further, we employ the bound (3.3) and obtain

$$\phi(x_k, t_k) - \phi(x_k + s_k, t_k) \geq \eta \min(\kappa_\Delta \epsilon, 1)\,\epsilon = \eta \kappa_\Delta \epsilon^2,$$

where we also used $\epsilon \in (0, 1]$ and $\kappa_\Delta \leq 1$ due to $L_g \geq 1$, η, $\gamma \in (0, 1)$; this gives (3.6). Finally, (3.7) results from (2.13) and (3.6). □

The next lemma connects approximate critical points of the merit functions of Phase 1 and 2 with those of our original problem (2.1).

Lemma 3.5 *Assume that $\|c(x_k)\| \leq \epsilon$ and $\chi(x_k, t_k) \leq \epsilon$. Then x_k is an approximate critical point in the sense that*

$$\|c(x_k)\| \leq \epsilon \textit{ and } \|J(x_k)^T y - g(x_k)\| \leq \epsilon \tag{3.9}$$

for some vector of multipliers $y \in \mathbb{R}^m$. Similarly, assume that $\psi(x) \leq \epsilon$. Then

$$\|J(x)^T z\| \leq \epsilon \tag{3.10}$$

for some vector of multipliers $z \in \mathbb{R}^m$.

Proof See Theorem 3.1 in [5] and the comments thereafter. □

Finally, we are ready to give the main complexity result of this paper.

Theorem 3.6 *Assume A.1–A.3 hold. Then Algorithm 2.1 generates an ϵ-first-oder critical point for problem (2.1), that is an iterate x_k satisfying either*

$$(3.9) \textit{ or } \left[\ (3.10)\ with\ \|c(x_k)\| > \kappa_f \epsilon\ \right],$$

in at most

$$\left\lceil \left(\|c(x_0)\| + f_{\rm up} - f_{\rm low} \right) \frac{\kappa_2}{\epsilon^2} \right\rceil \tag{3.11}$$

evaluations of c and f (and their derivatives), where $\kappa_2 > 0$ is a constant independent of ϵ and x_0.

Proof We have seen in Lemma 3.1 that the complexity of obtaining x_1 is bounded above by $O(\lceil \|c(x_0)\| \epsilon^{-2} \rceil)$. Thus, as $\psi(x_1) \leq \epsilon$, Lemma 3.5 ensures that (3.10) holds. If the algorithm terminates at this stage, then both (3.10) and $\|c(x_k)\| > \kappa_f \epsilon$ hold, as requested. Assume now that Phase 2 of the algorithm is entered. We then observe that Lemma 3.2 implies that successful iterations must happen as long as $\chi(x_k, t_k) \geq \epsilon$. Moreover, we have that

$$f_{\rm low} \leq f(x_k) \leq t_k + \epsilon \leq t_1 - i_k \kappa_C \epsilon^2 + \epsilon \leq f(x_1) - i_k \kappa_C \epsilon^2 + \epsilon \tag{3.12}$$

where i_k is the number of these successful iterations from iterations 1 to k of Phase 2, and where we have successively used A.3, (2.17) and (3.7). Hence, we obtain from the inequality $f(x_1) \leq f_{\rm up}$ (itself implied by A.3 again) that

$$i_k \leq \left\lceil \frac{f_{\rm up} - f_{\rm low} + \epsilon}{\kappa_C \epsilon^2} \right\rceil. \tag{3.13}$$

The number of Phase 2 iterations satisfying $\chi(x_k, t_k) \geq \epsilon$ is therefore bounded above, and the algorithm must terminate after (3.13) such iterations at most, yielding, because of Lemma 3.5, an ϵ-first-order critical point satisfying (3.9). Remembering that only one evaluation of c and f (and their derivatives, if successful) occurs per iteration, we therefore conclude from (3.13) and Lemma 3.3 that the total number of such evaluations in Phase 2 is bounded above by

$$\left\lceil \frac{f_{\rm up} - f_{\rm low} + \epsilon}{\kappa_C \epsilon^2} \right\rceil + O(|\log \epsilon|)$$

Summing this upper bound with that for the number of iterations in Phase 1 given by Lemma 3.1, and using also that $\epsilon \leq 1 \leq f_{\rm up} - f_{\rm low}$, then yields (3.11). □

4 Including general inequality constraints

If we now return to the solution of problem (1.1), we may consider defining

$$c(x) = \begin{pmatrix} c_E(x) \\ \min[\,0, c_I(x)\,] \end{pmatrix}$$

in the above. The quantity $\|c(x)\|$ can again be considered as the composition of a nonsmooth convex function with the smooth function $(c_E(x)^T, c_I(x)^T)^T$ and the theory developed above applies without modification, except that Lemma 3.5 must be adapted for the presence of inequality constraints. If an inequality constraint is active at an approximate critical point, then its multiplier has to be non-negative because $y \in \partial(\|\min[0, \cdot\,]\|)$ implies that $y \geq 0$. If it is inactive, then it may as well be absent from the problem (and its multiplier must be zero). Hence Lemma 3.5 generalizes to the inequality constraints case (1.1) without difficulty.

5 Conclusions

We have shown that the evaluation complexity to achieve either an ϵ-first-order critical point of the general smooth nonlinear optimization problem (1.1) or an infeasible ϵ-critical point of the infeasibilities of (1.1) is at most $O(\epsilon^{-2})$, where the constant involved is independent of ϵ but depends on algorithm parameters and problem constants—some of the latter may further depend, possibly even exponentially, on the problem dimension [6,7]. This is a marked improvement over the results presented in [5], where the same complexity was achieved only if the penalty parameter of the exact-penalty minimization scheme used there remained bounded, the complexity being $O(\epsilon^{-5})$ otherwise. Moreover, the results obtained in the present paper only assume boundedness of the objective function on a small neighbourhood of the feasible set, rather than on the whole space.

Since Cartis et al. [2] have shown that the O(ϵ^{-2}) bound is essentially sharp, and hence attained, by steepest descent with inexact linesearches in the unconstrained case, and since the method presented here for the constrained case is a steepest-descent-like method, improving this same-order bound in the constrained case seems impossible for methods of the same type.

We fully accept that the Short-Step Steepest-Descent algorithm discussed in Sect. 2 is most likely to be extremely inefficient in practice, because it amounts to following the constraints manifold with very small steps. 'Long steps' variants may be considered in which the setting of the target t_k is more aggressively geared towards minimizing the objective function. Whether such variants can be numerically effective remains to be seen, but their complexity will be difficult to guarantee with the kind of technique used here, as this would rely on global optimization of the constraint violation.

That we expect Algorithm 2.1 to be outperformed in practice is to be welcomed, indicating that the $O(\epsilon^{-2})$ evaluation bound may be as pessimistic for the constrained case as it is for the unconstrained one. But it remains remarkable that this pessimistic bound is unaffected by the presence of possibly nonlinear and nonconvex constraints.

Acknowledgments The work of the first author is funded by EPSRC Grant EP/I028854/1 and of the second author, by EPSRC Grant EP/E053351/1. All three authors are grateful to the Royal Society for its support through the International Joint Project 14265. We also thank two anonymous referees for their useful comments that have greatly improved the paper.

References

1. Byrd, R.H., Gould, N.I.M., Nocedal, J., Waltz, R.A.: On the convergence of successive linear-quadratic programming algorithms. SIAM J. Optim. **16**(2), 471–489 (2005)
2. Cartis, C., Gould, N.I.M., Toint, Ph.L.: On the complexity of steepest descent, Newton's and regularized Newton's methods for nonconvex unconstrained optimization. SIAM J. Optim. **20**, 2833–2852 (2010)
3. Cartis, C., Gould, N.I.M., Toint, Ph.L.: Adaptive cubic regularisation methods for unconstrained optimization. Part I: motivation, convergence and numerical results. Math. Program. Ser. A **127**(2), 245–295 (2011)
4. Cartis, C., Gould, N.I.M., Toint, Ph.L.: Adaptive cubic regularisation methods for unconstrained optimization. Part II: worst-case function- and derivative-evaluation complexity. Math. Program. Ser. A **130**, 295–319 (2011)
5. Cartis, C., Gould, N.I.M., Toint, Ph.L.: On the evaluation complexity of composite function minimization with applications to nonconvex nonlinear programming. SIAM J. Optim. **21**(4), 1721–1739 (2011)
6. Cartis, C., Gould, N.I.M., Toint, Ph.L.: Optimal Newton-type methods for nonconvex smooth optimization problems. ERGO technical report 11-009, School of Mathematics, University of Edinburgh (2011)
7. Cartis, C., Gould, N.I.M., Toint, Ph.L.: A note about the complexity of minimizing Nesterov's smooth Chebyshev–Rosenbrock function. ERGO technical report 11–013, School of Mathematics, University of Edinburgh (2011)
8. Cartis, C., Gould, N.I.M., Toint, Ph.L.: On the oracle complexity of first-order and derivative-free algorithms for smooth nonconvex minimization. SIAM J. Optim. **22**(1), 66–86 (2012)
9. Cartis, C., Gould, N.I.M., Toint, Ph.L.: An adaptive cubic regularization algorithm for nonconvex optimization with convex constraints and its function-evaluation complexity. IMA J. Numer. Anal. **32**(4), 1662–1695 (2012)
10. Gratton, S., Sartenaer, A., Toint, Ph.L.: Recursive trust-region methods for multiscale nonlinear optimization. SIAM J. Optim. **19**(1), 414–444 (2008)
11. Nesterov, Y.: Introductory Lectures on Convex Optimization. Applied Optimization. Kluwer, Dordrecht (2004)
12. Nesterov, Y., Polyak, B.T.: Cubic regularization of Newton method and its global performance. Math. Program. Ser. A **108**(1), 177–205 (2006)
13. Nocedal, J., Wright, S.J.: Numerical Optimization, 2nd edn. Springer, New York (2006)

14. Vavasis, S.A.: Black-box complexity of local minimization. SIAM J. Optim. **3**(1), 60–80 (1993)
15. Vicente, L.N.: Worst case complexity of direct search. Preprint 10-17, Department of Mathematics, University of Coimbra, Coimbra, Portugal, Euro J. Comput. Optim. (2010, to appear)