Strong Evaluation Complexity of An Inexact Trust-Region Algorithm for Arbitrary-Order Unconstrained Nonconvex Optimization

C. Cartis,[*] N. I. M. Gould[†] and Ph. L. Toint[‡]

2 XI 2020

Abstract

A trust-region algorithm using inexact function and derivatives values is introduced for solving unconstrained smooth optimization problems. This algorithm uses high-order Taylor models and allows the search of strong approximate minimizers of arbitrary order. The evaluation complexity of finding a q-th approximate minimizer using this algorithm is then shown, under standard conditions, to be $\mathcal{O}\big(\min_{j \in \{1,\ldots,q\}} \epsilon_j^{-(q+1)} \big)$ where the ϵ_j are the order-dependent requested accuracy thresholds. Remarkably, this order is identical to that of classical trust-region methods using exact information.

Context: The material of this report is part of a forthcoming book of the authors on the evaluation complexity of optimization methods for nonconvex problems.

1 Inexact Algorithms Using Dynamic Accuracy

Most of the literature on optimization assumes that evaluations of the objective function, as well as evaluations of its derivatives of relevant order(s), can be carried out exactly. Unfortunately, this assumption is not always fulfilled in practice and there are many applications where either the objective-function values or those of its derivatives (or both) are only known approximately. This can happen in several contexts. The first is when the values in questions are computed by some kind of experimental process whose accuracy can possibly be tuned (with the understanding that more accurate values maybe be, sometimes substantially, more expensive in terms of computational effort). A second related case is when objective-function or derivatives values result from some (hopefully convergent) iteration: obtaining more accuracy is also possible by letting the iteration converge further, but again at the price of possibly significant additional computing. A third context, quite popular nowadays in the framework of machine learning, is when the values of the objective functions and/or its derivatives are obtained by sampling (say among the terms of a sum involving a very large number of them). Again, using a larger sample size results in probabilistically better accuracy, but at a cost.

---

[*]Mathematical Institute, Oxford University, Oxford OX2 6GG, England. Email: coralia.cartis@maths.ox.ac.uk

[†]Computational Mathematics Group, STFC-Rutherford Appleton Laboratory, Chilton OX11 0QX, England. Email: nick.gould@stfc.ac.uk . The work of this author was supported by EPSRC grant EP/M025179/1

[‡]Namur Center for Complex Systems (naXys), University of Namur, 61, rue de Bruxelles, B-5000 Namur, Belgium. Email: philippe.toint@unamur.be

Extending ideas proposed in [1][(1)], this report discusses a trust-region algorithm which can handle such contexts, under what we call the "dynamic accuracy" requirement: we assume that *the required values* (objective-function or derivatives) *can always be computed with an accuracy which is specified, before the calculation, by the algorithm itself.* It is also understood in what follows that *the algorithm should require high accuracy only if necessary*, while *guaranteeing final results to full accuracy.* In this situation, it is hoped that many function's or derivative's evaluations can be carried out with a fairly loose accuracy (we will refer to these as "inexact values"), thereby resulting in a significantly cheaper optimization process.

2 Taylor decrements and enforcing accuracy

We consider the problem of minimizing a smooth, potentially nonconvex, function f from $\mathbb{R}^n$ into $\mathbb{R}$ without constraints on the variables. This problem has generated a literature too abundant to be reviewed here, but it is probably fair to say that trust-region methods feature among the most successful algorithms for its solution, showing excellent practical performance and solid theoretical background (see [8] for an in-depth discussion). These methods are based on using n Taylor-series models, which clearly depend on values and derivatives of the objective function at a sequence of points (iterates), but in the scenario we are about to consider, we do not assume that we can calculate them exactly. That is, rather than having true problem function and derivatives values, $f(x)$ and $\nabla_x^j f(x)$ for $j \in \{1, \ldots, q\}$ at x, we are provided with approximations $\overline{f}(x)$ and $\overline{\nabla_x^j f}(x)$—here and hereafter, we denote inexact quantities and approximations with an overbar.

Consequently, while high-degree exact approaches (see [4, 3, 7, 2] for instance) deal with a p-th degree Taylor-series approximation

$$T_{f,p}(x,s) = f(x) + \sum_{i=1}^{p} \frac{1}{i!} \nabla_x^i f(x)[s]^i \equiv T_{f,p}(x,0) + \sum_{i=1}^{p} \frac{1}{i!} [\nabla_v^i T_{f,p}(x,v)]_{v=0}[s]^i$$

of f for perturbations s around x, in our new framework, we have to be content with an inexact equivalent

$$\overline{T}_{f,p}(x,s) = \overline{f}(x) + \sum_{i=1}^{p} \frac{1}{i!} \overline{\nabla_x^i f}(x)[s]^i \equiv \overline{T}_{f,p}(x,0) + \sum_{i=1}^{p} \frac{1}{i!} [\nabla_v^i \overline{T}_{f,p}(x,v)]_{v=0}[s]^i.$$

It is therefore pertinent to investigate the effect of inexact derivatives on such approximations. As we shall see, for our purposes it will be important to achieve sufficient *relative* accuracy on the value of the Taylor model. More specifically, we will be concerned with the *Taylor decrement* defined, at x and for a step s, by

$$\begin{aligned}
\Delta T_{f,p}(x,s) &\stackrel{\text{def}}{=} T_{f,p}(x,0) - T_{f,p}(x,s) \\
&= -\sum_{i=1}^{p} \frac{1}{i!} [\nabla_v^i T_{f,p}(x,v)]_{v=0}[s]^i \\
&\equiv -\sum_{i=1}^{p} \frac{1}{i!} \nabla_x^i f(x)[s]^i.
\end{aligned}$$

[(1)]For regularization methods.

While our traditional algorithms depend on this quantity, it is of course of the question to use them in the present context, as we only have approximate values. But an obvious alternative is to consider instead the inexact Taylor decrement

$$\overline{\Delta T}_{f,j}(x,s) \overset{\text{def}}{=} \overline{T}_{f,j}(x,0) - \overline{T}_{f,j}(x,s) = -\sum_{i=1}^{j} \frac{1}{i!} [\nabla_v^i \overline{T}_{f,j}(x,v)]_{v=0} [s]^i. \tag{2.1}$$

We shall suppose in what follows that a relative accuracy parameter $\omega \in (0,1)$ is given, and we will then require that

$$|\overline{\Delta T}_{f,p}(x,s) - \Delta T_{f,p}(x,s)| \leq \omega \overline{\Delta T}_{f,p}(x,s) \tag{2.2}$$

whenever $\overline{\Delta T}_{f,p}(x,s) > 0$. It is not obvious at this point how to enforce this relative error bound, and we now discuss how this can be achieved.

But Taylor models also occur in termination rule for high-order approximate minimizers. In particular, it has been argued in [7] that "strong" approximate q-th order minimizers satisfy the necessary optimality condition

$$\phi_{f,j}^{\delta_j}(x_\epsilon) \leq \epsilon_j \frac{\delta_j^j}{j!} \quad \text{for all} \quad j \in \{1, \ldots, q\}. \tag{2.3}$$

for some $\delta_j \in (0,1)$, where

$$\phi_{f,j}^{\delta}(x) \overset{\text{def}}{=} f(x) - \min_{\|d\| \leq \delta} T_{f,j}(x,d), \tag{2.4}$$

which is the *largest decrease of the j-th order Taylor-series model $T_{f,j}(x,s)$ achievable by a point at distance at most δ from x*. Note that $\phi_{f,j}^{\delta}(x)$ is a continuous function of x and δ for $f \in C^j$ [13, Th. 7]. It is also important to observe that $\phi_{f,j}^{\delta}(x)$ is *independent of the value of $f(x_k)$*, because the zero-th degree terms cancel in (2.4). In what follows, we will mostly consider $\delta \leq 1$, but this is not necessary. Thus $\phi_{f,j}^{\delta_j}(x)$ is itself based on a Taylor-series model and thus is of importance since we plan to use (2.3) as a termination rule for our proposed algorithm. This reinforces the need to understand how to enforce the accuracy which is necessary for the algorithm to finally produce an exact approximate minimizer.

2.1 Enforcing the relative error on Taylor decrements

For clarity, we shall temporarily neglect the iteration index k. While there may be circumstances in which (2.2) can be enforced directly, we consider here that the only control the user has on the accuracy of $\overline{\Delta T}_{f,j}(x,s)$ is by imposing bounds on the *absolute* errors of the derivative tensors $\{\nabla_x^i f(x)\}_{i=1}^{j}$. In other words, we seek to ensure (2.2) by selecting absolute accuracies $\{\zeta_i\}_{i=1}^{j}$ such that the desired accuracy requirement follows whenever

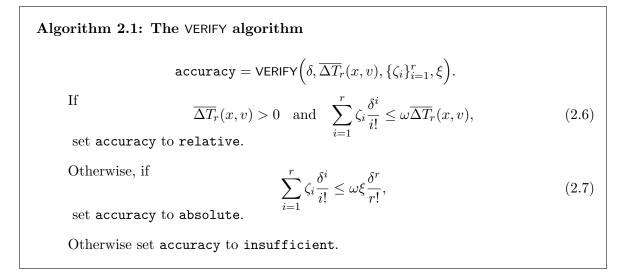$$\|\overline{\nabla_x^i f}(x) - \nabla_x^i f(x)\| \leq \zeta_i \ \text{for} \ i \in \{1, \ldots, j\}, \tag{2.5}$$

where $\|\cdot\|$ denotes the Euclidean norm for vectors and the induced operator norm for matrices and tensors. As one may anticipate by examining (2.2), a suitable relative accuracy requirement can be achieved so long as $\overline{\Delta T}_{f,j}(x,s)$ remains safely away from zero. However, if exact

computations are to be avoided, we may have to accept a simpler absolute accuracy guarantee when $\overline{\Delta T}_{f,j}(x,s)$ is small, but one that still guarantees our final optimality conditions.

Of course, not all derivatives need to be inexact in our framework. If derivatives of order $i \in \mathcal{E} \subseteq \{1, \ldots, q\}$ are exact, then the left-hand side of (2.5) vanishes for $i \in \mathcal{E}$ and the choice $\zeta_i = 0$ for $i \in \mathcal{E}$ is perfectly adequate. However, we avoid carrying this distinction in the arguments that follow for the sake of notational simplicity.

We now start by describing a crucial tool that we use to achieve (2.2), the VERIFY algorithm, stated as Algorithm 2.1 below. We use this to assess the relative model-accuracy whenever needed in the algorithms we describe later in this section.

To put our exposition in a general context, we suppose that we have a Taylor series $T_r(x,v)$ of a given function about x in the direction v, along with an approximation $\overline{T}_r(x,v)$, both of degree r, as well as the decrement $\overline{\Delta T}_r(x,v)$. We suppose that a bound $\delta \geq \|v\|$ is given, and that *required* relative and absolute accuracies ω and $\xi > 0$ are on hand. Moreover, we assume that the *current* upper bounds $\{\zeta_j\}_{j=1}^r$ on absolute accuracies of the derivatives of $\overline{T}_r(x,v)$ with respect to v at $v = 0$ are provided. Because it will always be the case when we need it, we will assume for simplicity that $\overline{\Delta T}_r(x,v) \geq 0$. Moreover, the relative accuracy constant $\omega \in (0,1)$ will fixed throughout the forthcoming algorithms, and we assume that it is given when needed in VERIFY.

Algorithm 2.1: The VERIFY algorithm

$$\texttt{accuracy} = \textsf{VERIFY}\Big(\delta, \overline{\Delta T}_r(x,v), \{\zeta_i\}_{i=1}^r, \xi\Big).$$

If

$$\overline{\Delta T}_r(x,v) > 0 \quad \text{and} \quad \sum_{i=1}^r \zeta_i \frac{\delta^i}{i!} \leq \omega \overline{\Delta T}_r(x,v), \tag{2.6}$$

set `accuracy` to `relative`.

Otherwise, if

$$\sum_{i=1}^r \zeta_i \frac{\delta^i}{i!} \leq \omega \xi \frac{\delta^r}{r!}, \tag{2.7}$$

set `accuracy` to `absolute`.

Otherwise set `accuracy` to `insufficient`.

It will be convenient to say informally that `accuracy` is `sufficient`, if it is either `absolute` or `relative`.

We may formalise the accuracy guarantees that result from applying the VERIFY algorithm as follows.

Lemma 2.1 Let $\omega \in (0, 1]$ and δ, ξ and $\{\zeta_i\}_{i=1}^r > 0$. Suppose that $\overline{\Delta T}_r(x, v) \geq 0$, that

$$\texttt{accuracy} = \textsf{VERIFY}\Big(\delta, \overline{\Delta T}_r(x, v), \{\zeta_i\}_{i=1}^r, \xi\Big).$$

and that

$$\left\| \left[\nabla_v^i \overline{T}_r(x, v) \right]_{v=0} - \left[\nabla_v^i T_r(x, v) \right]_{v=0} \right\| \leq \zeta_i \text{ for } i \in \{1, \ldots, r\}. \tag{2.8}$$

Then
(i) `accuracy` is `sufficient` whenever

$$\sum_{i=1}^r \zeta_i \frac{\delta^i}{i!} \leq \omega \xi \frac{\delta^r}{r!}, \tag{2.9}$$

(ii) if `accuracy` is `absolute`,

$$\max\left[\overline{\Delta T}_r(x, v), \left| \overline{\Delta T}_r(x, w) - \Delta T_r(x, w) \right| \right] \leq \xi \frac{\delta^r}{r!} \tag{2.10}$$

for all w with $\|w\| \leq \delta$.
(iii) if `accuracy` is `relative`, $\overline{\Delta T}_r(x, v) > 0$ and

$$\left| \overline{\Delta T}_r(x, w) - \Delta T_r(x, w) \right| \leq \omega \overline{\Delta T}_r(x, v), \text{ for all } w \text{ with } \|w\| \leq \delta, \tag{2.11}$$

Proof. We first prove proposition (i), and assume that (2.9) holds, which clearly ensures that (2.7) is satisfied. Thus either (2.6) or (2.7) must hold and termination occurs, proving the first proposition.

It follows by definition of the Taylor series, the triangle inequality and (2.8) that

$$\begin{aligned}
\left| \overline{\Delta T}_r(x, w) - \Delta T_r(x, w) \right| &= \left| \sum_{i=1}^r \frac{(\nabla_w^i \overline{T}_r(x, w) - \nabla_w^i T_r(x, w))[w]^i}{i!} \right| \\
&\leq \sum_{i=1}^r \frac{\|\nabla_w^i \overline{T}_r(x, w) - \nabla_w^i T_r(x, w)\| \|w\|^i}{i!} \\
&\leq \sum_{i=1}^r \zeta_i \frac{\|w\|^i}{i!}.
\end{aligned} \tag{2.12}$$

Consider now the possible `sufficient` termination cases for the algorithm and suppose first that termination occurs with `accuracy` as `absolute`. Then, using (2.12), (2.7) and $\omega < 1$, we have that, for any w with $\|w\| \leq \delta$,

$$\left| \overline{\Delta T}_r(x, w) - \Delta T_r(x, w) \right| \leq \sum_{i=1}^r \zeta_i \frac{\delta^i}{i!} \leq \omega \xi \frac{\delta^r}{r!} \leq \xi \frac{\delta^r}{r!}. \tag{2.13}$$

If $\overline{\Delta T}_r(x, v) = 0$, we may combine this with (2.13) to derive (2.10). By contrast, if $\overline{\Delta T}_r(x, v) > 0$, then since (2.6) failed but (2.7) holds,

$$\omega \overline{\Delta T}_r(x, w) < \sum_{i=1}^r \zeta_i \frac{\delta^i}{i!} \leq \omega \xi \frac{\delta^r}{r!}.$$

Combining this inequality with (2.13) yields (2.10). Suppose now that `accuracy` is `relative`. Then (2.6) holds, and combining it with (2.12) gives that

$$\left| \overline{\Delta T}_r(x, w) - \Delta T_r(x, w) \right| \leq \sum_{i=1}^{r} \zeta_i \frac{\delta^i}{i!} \leq \omega \overline{\Delta T}_r(x, v_\omega),$$

for any w with $\|w\| \leq \delta$, which is (2.11). □

Clearly, the outcome corresponding to our initial aim to obtain a relative error at most ω corresponds to the case where `accuracy` is `relative`. As we will shortly discover, the two other cases are also needed.

2.2 Computing the approximate optimality measures

Our next concern is how one might compute an optimality measure, given an inexactly computed $\overline{\Delta T}_{f,p}(x, s)$. Using the crucial measure of optimality

$$\phi_{f,j}^\delta(x) \stackrel{\text{def}}{=} \max_{\|d\| \leq \delta} \Delta T_{f,j}(x, d), \tag{2.14}$$

is out of the question, but an obvious alternative is to consider instead the *inexact measure*

$$\overline{\phi}_{f,j}^\delta(x) \stackrel{\text{def}}{=} \max_{\|d\| \leq \delta} \overline{\Delta T}_{f,j}(x, d) \tag{2.15}$$

that depends on an equivalent sufficiently accurate inexact Taylor decrement. We immediately observe that $\overline{\phi}_{f,j}^\delta(x)$ is independent of the value of $\overline{f}(x_k)$. Natural questions are then how well a particular $\overline{\phi}_{f,j}^\delta(x)$ approximates $\phi_{f,j}^\delta(x)$ and, if there is reasonable agreement, what is a sensible alternative to the stopping rule (2.3)?

We answer both questions in Algorithm 2.2 below, which shows one way to compute $\overline{\phi}_{f,j}^\delta(x)$. For analysis purposes, this algorithm involves a counter i_ζ of the number of times accuracy on the derivatives has been improved.

Observe that known values of derivatives for $i < j$ may be reused in Step 1.1 if required.

We now establish that Algorithm 2.2 produces values of the required optimality measures that are adequate in the sense that either an approximate minimizer is detected or a suitable approximation of the exact optimality measure is obtained.

Lemma 2.2 If Algorithm 2.2 terminates within Step 1.3 when `accuracy`$_j$ is `absolute`, then

$$\phi_{f,j}^{\delta_k}(x_k) \leq \epsilon_j \frac{\delta_k^j}{j!}. \tag{2.18}$$

Otherwise, if it terminates with `accuracy`$_j$ as `relative`, then

$$(1 - \omega)\overline{\phi}_{f,j}^{\delta_k}(x_k) \leq \phi_{f,j}^{\delta_k}(x_k) \leq (1 + \omega)\overline{\phi}_{f,j}^{\delta_k}(x_k). \tag{2.19}$$

Moreover, termination with one of these two outcomes must occur if

$$\max_{i \in \{1, \dots, j\}} \zeta_{i, i_\zeta} \leq \frac{\omega}{4} \epsilon_j \frac{\delta_k^{j-1}}{j!} \tag{2.20}$$

Algorithm 2.2: Computing $\overline{\phi}_{f,j}^{\delta}(x)$

The iterate x_k, the index $j \in \{1, \ldots, q\}$ and the radius $\delta_k \in (0,1]$ are given, as well as the constant $\gamma_\zeta \in (0,1)$. The counter i_ζ, the relative accuracy $\omega \in (0,1]$ and the absolute accuracies bounds $\{\zeta_{i,i_\zeta}\}_{i=1}^{q}$ are also given.

Step 1.1: If they are not yet available, compute $\{\overline{\nabla_x^i f}(x_k)\}_{i=1}^{j}$ satisfying

$$\|\overline{\nabla_x^i f}(x_k) - \nabla_x^i f(x_k)\| \le \zeta_{i,i_\zeta} \quad \text{for} \ \ i \in \{1, \ldots, j\}.$$

Step 1.2: Find

$$d_{k,j} = \arg \max_{\|d\| \le \delta_k} \overline{\Delta T}_{f,j}(x_k, d)$$

and the corresponding Taylor decrement $\overline{\Delta T}_{f,j}(x_k, d_{k,j})$. Compute

$$\texttt{accuracy}_j = \textsf{VERIFY}\Big(\delta_k, \overline{\Delta T}_{f,j}(x_k, d_{k,j}), \{\zeta_{i,i_\zeta}\}_{i=1}^{j}, \tfrac{1}{2}\epsilon_j\Big). \qquad (2.16)$$

Step 1.3: If $\texttt{accuracy}_j$ is $\texttt{sufficient}$, return

$$\overline{\phi}_{f,j}^{\delta_k}(x_k) = \overline{\Delta T}_{f,j}(x_k, d_{k,j}).$$

Step 1.4: Otherwise (i.e. if $\texttt{accuracy}_j$ is $\texttt{insufficient}$), set

$$\zeta_{i,i_\zeta+1} = \gamma_\zeta \zeta_{i,i_\zeta} \quad \text{for} \ \ i \in \{1, \ldots, j\}, \qquad (2.17)$$

increment i_ζ by one and return to Step 1.1.

Proof. Consider $j \in \{1, \ldots, q\}$. We first notice that Step 1.1 of Algorithm 2.2 yields (2.8) with $T_r = T_r^f$ and $r = j$, so that the assumptions of Lemma 2.1 are satisfied. Note first that, because Step 1.2 finds the global maximum of $\overline{\Delta T}_{f,j}(x_k, d)$, we have that $\overline{\Delta T}_{f,j}(x_k, d_{k,j}) \geq 0$. Suppose now that, in Step 1.2, the VERIFY algorithm returns accuracy = absolute and that $\overline{\Delta T}_{f,j}(x_k, d_k) = 0$. This means that x_k is a global minimizer of $\overline{T}_{f,j}(x_k, d)$ in the ball of radius $\delta_{k,j}$ and $\overline{\Delta T}_{f,j}(x_k, d) \leq 0$ for any d in this ball. Thus, for any such d, we obtain from (2.10) with $\xi = \frac{1}{2}\epsilon_j$ that

$$\Delta T_{f,j}(x_k, d) \leq \overline{\Delta T}_{f,j}(x_k, d) + \left| \overline{\Delta T}_{f,j}(x_k, d) - \Delta T_{f,j}(x_k, d) \right| \leq \tfrac{1}{2}\epsilon_j \frac{\delta_{k,j}^j}{j!},$$

which implies (2.18). Suppose next that the VERIFY algorithm returns accuracy = absolute but now $\overline{\Delta T}_{f,j}(x_k, d_{k,j}) > 0$ and thus $d_{k,j} \neq 0$. Using the fact that the nature of Step 1.2 ensures that $\overline{\Delta T}_{f,j}(x_k, d) \leq \overline{\Delta T}_{f,j}(x_k, d_{k,j})$ for d with $\|d\| \leq \delta_{k,j}$ we have, using (2.10) with $\xi = \frac{1}{2}\epsilon_j$, that, for all such d,

$$\begin{aligned}
\Delta T_{f,j}(x_k, d) &\leq \overline{\Delta T}_{f,j}(x_k, d) + \left| \overline{\Delta T}_{f,j}(x_k, d) - \Delta T_{f,j}(x_k, d) \right| \\
&\leq \overline{\Delta T}_{f,j}(x_k, d_{k,j}) + \left| \overline{\Delta T}_{f,j}(x_k, d) - \Delta T_{f,j}(x_k, d) \right| \\
&\leq \epsilon_j \frac{\delta_{k,j}^j}{j!}
\end{aligned}$$

yielding (2.18). If the VERIFY algorithm returns accuracy = relative, then, for any d with $\|d\| \leq \delta_{k,j}$,

$$\begin{aligned}
\Delta T_{f,j}(x_k, d) &\leq \overline{\Delta T}_{f,j}(x_k, d) + \left| \overline{\Delta T}_{f,j}(x_k, d) - \Delta T_{f,j}(x_k, d) \right| \\
&\leq (1 + \omega)\overline{\Delta T}_{f,j}(x_k, d_{k,j}).
\end{aligned}$$

Thus, for all d with $\|d\| \leq \delta_{k,j}$,

$$\begin{aligned}
\max \left[0, \Delta T_{f,j}(x_k, d) \right] &\leq (1 + \omega) \max \left[0, \overline{\Delta T}_{f,j}(x_k, d_{k,j}) \right] \\
&= (1 + \omega)\overline{\phi}_{f,j}^{\delta_{k,j}}(x_k)
\end{aligned}$$

and the rightmost part of (2.19) follows. Similarly, for any d with $\|d\| \leq \delta_{k,j}$,

$$\begin{aligned}
\Delta T_{f,j}(x_k, d) &\geq \overline{\Delta T}_{f,j}(x_k, d) - \left| \overline{\Delta T}_{f,j}(x_k, d) - \Delta T_{f,j}(x_k, d) \right| \\
&\geq \overline{\Delta T}_{f,j}(x_k, d) - \omega \overline{\Delta T}_{f,j}(x_k, d_{k,j}).
\end{aligned}$$

Hence

$$\begin{aligned}
\max_{\|d\| \leq \delta_{k,j}} \Delta T_{f,j}(x_k, d) &\geq \max_{\|d\| \leq \delta_{k,j}} \left[\overline{\Delta T}_{f,j}(x_k, d) - \omega \overline{\Delta T}_{f,j}(x_k, d_{k,j}) \right] \\
&= (1 - \omega)\overline{\Delta T}_{f,j}(x_k, d_{k,j}).
\end{aligned}$$

Since $\overline{\Delta T}_{f,j}(x_k, d_{k,j}) > 0$ when the VERIFY algorithm returns accuracy = relative, we then obtain that, for all $\|d\| \leq \delta_{k,j}$,

$$\begin{aligned}
\max \left[0, \max_{\|d\| \leq \delta_{k,j}} \Delta T_{f,j}(x_k, d) \right] &\geq (1 - \omega)\overline{\Delta T}_{f,j}(x_k, d_{k,j}) \\
&= (1 - \omega)\overline{\phi}_{f,j}^{\delta_{k,j}}(x_k),
\end{aligned}$$

which is the leftmost part of (2.19). In order to prove the last statement of the lemma, suppose that (2.20) holds. Then

$$\sum_{i=1}^{j} \zeta_{i,i_\zeta} \frac{\delta_k^i}{i!} \leq \max_{i \in \{1,\dots,j\}} \zeta_{i,i_\zeta} \sum_{i=1}^{j} \frac{\delta_k^i}{i!} \leq (\exp(1)-1)\delta_k \max_{i \in \{1,\dots,j\}} \zeta_{i,i_\zeta} \leq 2\delta_k \max_{i \in \{1,\dots,j\}} \zeta_{i,i_\zeta} \leq \tfrac{1}{2}\omega\epsilon_j \frac{\delta_k^j}{j!}$$

and Lemma 2.1 (i) then ensures that the call to VERIFY in Step 1.2 returns $\texttt{accuracy}_j$ as $\texttt{sufficient}$, causing Algorithm 2.2 to terminate in Step 1.3. $\square$

Notice that if we apply Algorithm 2.2 for all $j \in \{1,\dots,q\}$ and each returned $\texttt{accuracy}_j$ is $\texttt{absolute}$, the bound (2.18) then ensures that x_k is an (ϵ, δ_k)-approximate q-th-order minimizer. Moreover, (2.10) then guarantees that

$$\overline{\phi}_{f,j}^{\delta_j}(x) \leq \left(\frac{\epsilon_j}{1+\omega}\right) \frac{\delta_j^j}{j!} \quad \text{for} \quad j \in \{1,\dots,q\}. \tag{2.21}$$

If $\texttt{accuracy}_j$ is $\texttt{relative}$ and (2.21) holds for $x = x_k$ and $\delta = \delta_k$ the same is true because of (2.19). Thus checking (2.21) is an adequate verification of the j-th order optimality condition. Moreover, the call the VERIFY in Step 1.1 must return $\texttt{relative}$ if (2.21) fails. Importantly, these conclusions do not require that the $\{\overline{\phi}_{f,j}^{\delta_j}(x)\}_{j=1}^{q}$ use the same set of approximate derivatives for all $j \in \{1,\dots,q\}$, but merely that their accuracy is deemed $\texttt{sufficient}$ by the VERIFY algorithm.

3 The TR*q*DA algorithm and its complexity

In what follows, we shall first consider an trust-region optimization algorithm, named TR*q*DA (the DA suffix refers to the Dynamic Accuracy framework) whose purpose is to find a vector $x = x_\epsilon$ for which (2.21) holds for some vector of optimality radii $\delta \in (0,1]^q$. This is important as we have just shown (in Lemma 2.2) that any x_k investigated by Algorithm 2.2 for all $j \in \{1,\dots,q\}$ is either directly an (ϵ, δ)-approximate q-th-order minimizer of $f(x)$ because of (2.18) or will be if (2.21) holds at $x = x_k$ because of (2.19).

An initial outline of the TR*q*DA algorithm is presented on the next page.

This algorithm does not specify how to find the step required in Step 2. This vital ingredient will be the subject of what will follow. In addition, we stress that although (3.3) and (3.4) might suggest that we need to know the true f, this is not the case, rather we simply need some mechanism to ensure that x_k and x_k+s_k satisfy the required bounds. These bounds are needed to guarantee convergence. Notice that the value of $\overline{\phi}_{f,j}^{\delta_k}(x_k)$ and $\overline{\Delta T}_{f,j}(x_k, s_k)$ do not depend on the value of $\overline{f}(x_k)$, and so Step 1 and 2 are also independent of this value. In particular, this allows to postpone the choice of $\overline{f}(x_0)$ to Step 3. At iteration k, a new value of $\overline{f}(x_k)$ has to be computed to ensure (3.4) in Step 3 only when $\overline{\Delta T}_{f,j}(x_{k-1}, s_{k-1}) > \overline{\Delta T}_{f,j}(x_k, s_k)$. If this is the case, the (inexact) function value is computed twice rather than once in that iteration. Finally note that the choice $\vartheta = 1$ is acceptable since we have assumed that $\epsilon_j \leq 1$ for all $j \in \{1,\dots,q\}$.

As usual for trust-region algorithms, iteration k is said to be successful when $\rho_k \geq \eta_1$ and $x_{k+1} = x_k + s_k$, and we define $\mathcal{S}$, $\mathcal{S}_k$ and $\mathcal{U}_k$ as

$$\mathcal{S} \overset{\text{def}}{=} \{k \in \mathbf{N} \mid \rho_k \geq \eta_1\} \quad \text{and} \quad \mathcal{U} \overset{\text{def}}{=} \mathbf{N} \setminus \mathcal{S}, \tag{3.7}$$

**Algorithm 3.1: Trust Region with Dynamic Accuracy
(TRqDA, basic version)**

Step 0: Initialisation. A criticality order q, a starting point x_0 and an initial trust-region radius Δ_0 are given, as well as accuracy levels $\epsilon \in (0,1)^q$ and an initial set of bounds on absolute derivative accuracies $\{\zeta_{j,0}\}_{j=1}^q$. The constants ω, ϑ, κ_ζ, η_1, η_2, γ_1, γ_2, γ_3 and $\Delta_{\max}$ are also given and satisfy

$$\vartheta \in [\min_{j \in \{1,\dots,q\}} \epsilon_j, 1], \quad \Delta_0 \le \Delta_{\max}, \quad 0 < \eta_1 \le \eta_2 < 1, \quad 0 < \gamma_1 < 1 < \gamma_2 < \gamma_3,$$

$$\omega \in \left(0, \min\left[\tfrac{1}{2}\eta_1, \tfrac{1}{4}(1 - \eta_2)\right]\right) \quad \text{and} \quad \zeta_{j,0} \le \kappa_\zeta \quad \text{for} \quad j \in \{1,\dots,p\}.$$

Set $k = 0$ and $i_\zeta = 0$.

Step 1: Termination test. Set $\delta_k = \min[\Delta_k, \vartheta]$. For $j = 1,\dots,q$,

1. Evaluate $\overline{\nabla_x^j f}(x_k)$ and compute $\overline{\phi}_{f,j}^{\delta_k}(x_k)$ using Algorithm 2.2.

2. If
$$\overline{\phi}_{f,j}^{\delta_k}(x_k) > \left(\frac{\epsilon_j}{1+\omega}\right)\frac{\delta_k^j}{j!},\tag{3.1}$$

go to Step 2 with d_k, the optimality displacement associated with $\overline{\phi}_{f,j}^{\delta_k}(x_k)$.

If the loop on j finishes, terminate with $x_\epsilon = x_k$ and $\delta_\epsilon = \delta_k$.

Step 2: Step computation. If $\Delta_k \le \vartheta$, set $s_k = d_k$ and $\overline{\Delta T}_{f,j}(x_k, s_k) = \overline{\Delta T}_{f,j}(x_k, d_k)$. Otherwise, compute a step s_k such that $\|s_k\| \le \Delta_k$,

$$\overline{\Delta T}_{f,j}(x_k, s_k) \ge \overline{\Delta T}_{f,j}(x_k, d_k)\tag{3.2}$$

and (2.2) holds—see Algorithm 3.2 below for details.

Step 3: Accept the new iterate. Compute $\overline{f}(x_k + s_k)$ ensuring that
$$|\overline{f}(x_k + s_k) - f(x_k + s_k)| \le \omega \overline{\Delta T}_{f,j}(x_k, s_k).\tag{3.3}$$

Also ensure (by setting $\overline{f}(x_k) = \overline{f}(x_{k-1} + s_{k-1})$ or by recomputing $\overline{f}(x_k)$) that
$$|\overline{f}(x_k) - f(x_k)| \le \omega \overline{\Delta T}_{f,j}(x_k, s_k).\tag{3.4}$$

Then compute
$$\rho_k = \frac{\overline{f}(x_k) - \overline{f}(x_k + s_k)}{\overline{\Delta T}_{f,j}(x_k, s_k)}.\tag{3.5}$$

If $\rho_k \ge \eta_1$, then set $x_{k+1} = x_k + s_k$; otherwise set $x_{k+1} = x_k$.

Step 4: Update the trust-region radius. Set

$$\Delta_{k+1} \in \begin{cases} [\gamma_1 \Delta_k, \gamma_2 \Delta_k] & \text{if } \rho_k < \eta_1, \\ [\gamma_2 \Delta_k, \Delta_k] & \text{if } \rho_k \in [\eta_1, \eta_2), \\ [\Delta_k, \min(\Delta_{\max}, \gamma_3 \Delta_k)] & \text{if } \rho_k \ge \eta_2, \end{cases}\tag{3.6}$$

Increment k by one and go to Step 2 with $d_{k+1} = d_k$ if $x_{k+1} = x_k$ and $\Delta_{k+1} \ge \theta$, or to Step 1 otherwise.

the sets of *successful* and *unsuccessful* iterations, respectively, and

$$\mathcal{S}_k \stackrel{\text{def}}{=} \{j \in \{0,\dots,k\} \mid \rho_j \geq \eta_1\} \quad \text{and} \quad \mathcal{U}_k \stackrel{\text{def}}{=} \{0,\dots,k\} \setminus \mathcal{S}_k, \tag{3.8}$$

the corresponding sets up to iteration k. Notice that $x_{k+1} = x_k + s_k$ for $k \in \mathcal{S}$, while $x_{k+1} = x_k$ for $k \in \mathcal{U}$.

For future reference, we now state a property of the TRqDA algorithm that solely depends on the mechanism (3.6) to update the trust-region radius.

Lemma 3.1 Suppose that the TR1 algorithm is used and that $\Delta_k \geq \Delta_{\min}$ for some $\Delta_{\min} \in (0, \Delta_0]$. Then

$$k \leq |\mathcal{S}_k| \left(1 + \frac{\log \gamma_3}{|\log \gamma_2|}\right) + \frac{1}{|\log \gamma_2|} \left|\log\left(\frac{\Delta_{\min}}{\Delta_0}\right)\right|, \tag{3.9}$$

Proof. Observe that (3.6) and our assumption imply that

$$\Delta_{i+1} \leq \gamma_3 \Delta_i, \quad i \in \mathcal{S}_k \quad \text{and} \quad \Delta_{i+1} \leq \gamma_2 \Delta_i, \quad i \in \mathcal{U}_k.$$

Using our assumption, we thus deduce inductively that

$$\Delta_{\min} \leq \Delta_k \leq \Delta_0 \gamma_3^{|\mathcal{S}_k|} \gamma_2^{|\mathcal{U}_k|}.$$

which gives that

$$\gamma_3^{|\mathcal{S}_k|} \gamma_2^{|\mathcal{U}_k|} \geq \frac{\Delta_{\min}}{\Delta_0}$$

and we obtain inequality (3.9) by taking logarithms on both sides and recalling that $\gamma_2 \in (0,1)$ and that $k = |\mathcal{S}_k| + |\mathcal{U}_k|$. $\square$

In words, so long as the trust-region radius is bounded from below, the total number of iterations performed thus far is bounded in terms of the number of successful ones. Note that this lemma is independent of the specific choice of s_k.

3.1 Computing the step s_k

We now have to specify how to compute the step s_k required by Step 2 whenever $\Delta_k > \vartheta$, in which case $\delta_k = \vartheta$. While any step satisfying both $\|s_k\| \leq \Delta_k$ and (3.2) is acceptable, we still have to provide a mechanism that ensures (2.2). This is the aim of Algorithm 3.2.

The next lemma reassuringly shows that Algorithm 3.2 must terminate, and provides useful details of the outcome.

Algorithm 3.2: Detailed Step 2 of the TRqDA algorithm when $\Delta_k > \vartheta$

The iterate x_k, the relative accuracy ω, the requested accuracy $\epsilon_j \in (0,1]^q$, the constants $\gamma_\zeta \in (0,1)$, the counter i_ζ and the absolute accuracies $\{\zeta_{j,i_\zeta}\}_{j=1}^q$ are given. The index $j \in \{1,\ldots,q\}$, the optimality displacement d_k and the constant $\vartheta \in (0,1]$ are also given such that, by (3.1),

$$\overline{\Delta T}_{f,j}(x_k, d_k) > \frac{\epsilon_j}{1+\omega} \frac{\vartheta^j}{j!}. \tag{3.10}$$

Step 2.1: If they are not yet available, compute $\{\overline{\nabla_x^i f}(x_k)\}_{i=1}^j$ satisfying

$$\|\overline{\nabla_x^i f}(x_k) - \nabla_x^i f(x_k)\| \leq \zeta_{i,i_\zeta} \text{ for } i \in \{1,\ldots,j\}.$$

Step 2.2: Step computation. Compute a step s_k such that $\|s_k\| \leq \Delta_k$ and yielding a decrease $\overline{\Delta T}_{f,j}(x_k, s_k)$ satisfying (3.2). Compute

$$\texttt{accuracy}_s =$$
$$\mathsf{VERIFY}\Big(\|s_k\|, \overline{\Delta T}_{f,j}(x_k, s_k), \{\zeta_{i,i_\zeta}\}_{i=1}^j, \frac{\epsilon_j}{4(1+\omega)} \Big(\frac{\vartheta}{\max\big[\vartheta, \|s_k\|\big]}\Big)^j\Big). \tag{3.11}$$

Step 2.3: If $\texttt{accuracy}_s$ is $\texttt{relative}$, go to Step 3 of Algorithm 3.1 with the step s_k and the associated $\overline{\Delta T}_{f,j}(x_k, s_k)$.

Step 2.4: Otherwise, set

$$\zeta_{i,i_\epsilon+1} = \gamma_\zeta \zeta_{i,i_\epsilon} \text{ for } i \in \{1,\ldots,j\}, \tag{3.12}$$

increment i_ζ by one and go to Step 2.1.

Lemma 3.2 Suppose that the detailed Step 2 given by Algorithm 3.2 is used in the TRqDA algorithm whenever $\Delta_k > \vartheta$. If this condition holds, the outcome of the call to VERIFY in Step 2.2 is `relative` and termination must occur with this outcome if

$$\max_{i \in \{1,\ldots,j\}} \zeta_{i,i_\zeta} \leq \frac{\omega \vartheta^{j-1}}{8j!(1+\omega)} \epsilon_j. \tag{3.13}$$

. In all cases, we have that $\overline{\Delta T}_{f,j}(x_k, s_k) > 0$ and

$$\left| \overline{\Delta T}_{f,j}(x_k, s_k) - \Delta T_{f,j}(x_k, s_k) \right| \leq \omega \overline{\Delta T}_{f,j}(x_k, s_k). \tag{3.14}$$

Proof. Suppose first that $\Delta_k \leq \vartheta$. Then $s_k = d_k$ and (3.1) gives $\overline{\Delta T}_{f,j}(x_k, s_k) > 0$. Moreover, our comment at the end of Section 2.2 shows that the outcome of the VERIFY algorithm called in Step 1.1 for order j must be `relative`. Lemma 2.1(iii) then ensures that (3.14) holds.

Suppose now that $\Delta_k > \vartheta$ and thus $\delta_k = \vartheta$. We therefore have that Algorithm 3.2 was used to compute s_k. Because derivatives may be re-evaluated within the course of this algorithm, we need to identify the particular inexact Taylor series we are considering: we will therefore distinguish $\overline{T}^0_{f,j}(x_k, d_k)$, $\overline{\Delta T}^0_{f,j}(x_k, d_k)$ and the corresponding accuracy bounds $\{\zeta_i^0\}_{i=1}^j$ using the derivatives $\{\overline{\nabla_x^i f}(x_k)\}_{i=1}^j$ available on entry of the algorithm , from $\overline{T}^+_{f,j}(x_k, d_k)$, $\overline{\Delta T}^+_{f,j}(x_k, d_k)$ and $\{\zeta_i^+\}_{i=1}^j$ using derivatives after one or more executions of Step 2.4. By construction, we have that

$$\zeta_i^+ < \zeta_i^0 \quad \text{for} \quad i \in \{1, \ldots, j\}. \tag{3.15}$$

We also note that, by (3.1) and (3.2),

$$\overline{\Delta T}^0_{f,j}(x_k, s_k^0) \geq \overline{\Delta T}^0_{f,j}(x_k, d_k) > \frac{\epsilon_j}{1+\omega} \frac{\vartheta^j}{j!} > 0, \tag{3.16}$$

where s_k^0 is computed using $\overline{T}^0_{f,j}$.

Observe now that the TRqDA has not terminated at Step 1 and thus that (3.1) holds. This in turn implies that

$$2\overline{\Delta T}^0_{f,j}(x_k, d_k) > (1+\omega_k)\overline{\Delta T}^0_{f,j}(x_k, d_k) > \epsilon_j \frac{\vartheta^j}{j!}$$

since $\omega < 1$, and hence the call the VERIFY in Step 1.2 of Algorithm 2.2 has returned `accuracy`$_j$ as `relative`. Therefore (2.6) must hold with $\zeta_i = \zeta_i^0$, $\delta = \vartheta$ and $\xi = \frac{1}{2}\epsilon_j$, yielding that

$$\sum_{i=1}^j \zeta_j^0 \frac{\vartheta^i}{i!} \leq \omega \overline{\Delta T}^0_{f,j}(x_k, d_k). \tag{3.17}$$

As a consequence, we find that

$$
\begin{aligned}
\Delta T_{f,j}(x_k, d_k) &\geq \overline{\Delta T^0_{f,j}}(x_k, d_k) - |\overline{\Delta T^0_{f,j}}(x_k, d_k) - \Delta T_{f,j}(x_k, d_k)| \\
&\geq \overline{\Delta T^0_{f,j}}(x_k, d_k) - \sum_{i=1}^{j} \zeta_j^0 \frac{\vartheta^i}{i!} \\
&\geq (1-\omega)\overline{\Delta T^0_{f,j}}(x_k, d_k) \\
&> \frac{1-\omega}{1+\omega} \epsilon_j \frac{\vartheta^j}{j!}
\end{aligned} \tag{3.18}
$$

from the triangle inequality, (2.12), the definition of $\{\zeta_i^0\}_{i=1}^j$, the fact that $\|d_k\| \leq \delta_k = \vartheta$ and (3.16). Using similar reasoning, but now with (3.15), we also deduce that

$$
\begin{aligned}
\overline{\Delta T^+_{f,j}}(x_k, d_k) &\geq \Delta T_{f,j}(x_k, d_k) - |\overline{\Delta T^+_{f,j}}(x_k, d_k) - \Delta T_{f,j}(x_k, d_k)| \\
&\geq \Delta T_{f,j}(x_k, d_k) - \sum_{i=1}^{j} \zeta_j^+ \frac{\vartheta^i}{i!} \\
&> \Delta T_{f,j}(x_k, d_k) - \sum_{i=1}^{j} \zeta_j^0 \frac{\vartheta^i}{i!}.
\end{aligned} \tag{3.19}
$$

Combining this with (3.17) and (3.18)

$$
\begin{aligned}
\overline{\Delta T^+_{f,j}}(x_k, d_k) &\geq \Delta T_{f,j}(x_k, d_k) - \omega\overline{\Delta T^0_{f,j}}(x_k, d_k) \\
&\geq \Delta T_{f,j}(x_k, d_k) - \left(\frac{\omega}{1-\omega}\right) \Delta T_{f,j}(x_k, d_k) \\
&\geq \frac{1-\omega}{1+\omega}\left(1 - \frac{\omega}{1-\omega}\right) \epsilon_j \frac{\vartheta^j}{j!} \\
&> \frac{\epsilon_j}{4(1+\omega)} \frac{\vartheta^j}{j!},
\end{aligned}
$$

where we have used the fact that $\omega < \frac{1}{4}(1-\eta_2) < \frac{1}{4}$ to deduce the last inequality. Hence, because of (3.2),

$$
\overline{\Delta T^+_{f,j}}(x_k, s_k^+) \geq \overline{\Delta T^+_{f,j}}(x_k, d_k) > \frac{\epsilon_j}{4(1+\omega)} \frac{\vartheta^j}{j!} > 0. \tag{3.20}
$$

Suppose now that $\overline{\Delta T_{f,j}}(x_k, s_k)$ is any of $\overline{\Delta T^0_{f,j}}(x_k, s_k)$ or $\overline{\Delta T^+_{f,j}}(x_k, s_k)$, and that the call to VERIFY in (3.11) returns `absolute`. Applying Lemma 2.1 (ii), we deduce that

$$
\overline{\Delta T_{f,j}}(x_k, s_k) \leq \frac{\epsilon_j}{4(1+\omega)} \frac{\vartheta^j}{\max\left[\vartheta, \|s_k\|\right]^j} \frac{\|s_k\|^j}{j!} \leq \frac{\epsilon_j}{4(1+\omega)} \frac{\vartheta^j}{j!},
$$

which contradicts both (3.16) and (3.20). This is thus impossible and the call to VERIFY in (3.11) must also return either `relative` or `insufficient`. It also follows from (3.16) and (3.20) that

$$
\omega\overline{\Delta T_{f,j}}(x_k, s_k) > \frac{\omega\epsilon_j}{4(1+\omega)} \frac{\vartheta^j}{j!} > 0, \tag{3.21}
$$

and thus if Step 2.4 continues to be called, ultimately (3.12) will ensure that

$$\frac{\omega\epsilon_j}{4(1+\omega)}\frac{\vartheta^j}{j!} \geq \sum_{i=1}^{j} \zeta_{i,i_\zeta}\frac{\vartheta^i}{i!}. \tag{3.22}$$

This and (3.21) then imply that eventually (2.6) in the call to VERIFY in (3.11) will hold, and hence `accuracy_s` is `relative`. Thus the exit test in Step 2.3 will ultimately be satisfied, and Algorithm 3.2 will terminate in a finite number of iterations with $\overline{\Delta T}_{f,j}(x_k, s_k) > 0$, because of (3.16) and (3.20), and `accuracy_s` as `relative`. We may then apply Lemma 2.1 (iii) to obtain (3.14). Finally observe that, since $\vartheta \leq 1$, we have that

$$\sum_{i=1}^{j} \zeta_{i,i_\zeta}\frac{\vartheta^i}{i!} \leq \max_{i\in\{1,\dots,j\}} \zeta_{i,i_\zeta}\sum_{i=1}^{j}\frac{\vartheta^i}{i!} \leq (\exp(1)-1)\vartheta \max_{i\in\{1,\dots,j\}} \zeta_{i,i_\zeta} \leq 2\vartheta \max_{i\in\{1,\dots,j\}} \zeta_{i,i_\zeta}.$$

Combining this with (3.13) and using (3.21), we deduce that (2.6) in the call to VERIFY in (3.11) will hold, `accuracy_s` is `relative`, and termination of Algorithm 3.2 in Step 2.3 will occur. □

The aim of the mechanism of the second item of Step 2.2 should now be clear: the choice of the last argument in the call to VERIFY in (3.11) is designed to ensure that the outcome `absolute` cannot happen. This is achieved by ensuring progressively shorter steps are taken unless a large inexact decrement is obtained. Observe that the choice $s_k = d_k$ is always possible and guarantees that inordinate accuracy is never needed.

Observe that the mechanism of Algorithm 3.2 allows loose accuracy if the inexact decrease $\overline{\Delta T}_{f,j}(x_k, s_k)$ is large—the test (2.6) will be satisfied in the call to VERIFY in Step 2, and thus VERIFY ignores its last, absolute accuracy argument (ξ) in this case—even if the trust-region radius is small, while it demands higher absolute accuracy if a large step results in a small decrease.

For future reference, we note that the last argument in the call to VERIFY in (3.11) satisfies

$$\frac{\epsilon_j}{4(1+\omega)}\frac{\vartheta^j}{\max\left[\vartheta, \|s_k\|\right]^j} \geq \frac{\epsilon_j}{4(1+\omega)}\frac{\vartheta^j}{\max\left[1, \Delta_{\max}\right]^j} \tag{3.23}$$

3.2   Evaluation complexity for the TR*q*DA algorithm

We are now ready to analyse the complexity of the TR*q*DA algorithm of on page 10, where Step 2 is implemented as in Algorithm 3.2. We first state our assumptions.

AS.1 The function f from $\mathbb{R}^n$ to $\mathbb{R}$ is p times continuously differentiable and each of its derivatives $\nabla_x^\ell f(x)$ of order $\ell \in \{1,\dots,p\}$ is Lipschitz continuous, that is, for every $j \in \{1,\dots,p\}$ there exists a constant $L_{f,j} \geq 1$ such that, for all $x, y \in \mathbb{R}^n$,

$$\|\nabla_x^j f(x) - \nabla_x^j f(y)\| \leq L_{f,j}\|x - y\|, \tag{3.24}$$

AS.2 There is a constant f_{low} such that $f(x) \geq f_{\text{low}}$ for all $x \in \mathbb{R}^n$.

For simplicity of notation, define

$$L_f \stackrel{\text{def}}{=} \max[1, \max_{j\in\{1,\dots,q\}} L_{f,j}]. \tag{3.25}$$

The Lipschitz continuity of the derivatives of f has a crucial consequence.

Lemma 3.3 Suppose that AS.1 holds. Then for all $x, s \in \mathbb{R}^n$,

$$|f(x + s) - T_{f,j}(x, s)| \leq \frac{L_{f,j}}{(j + 1)!} \|s\|^{j+1}. \tag{3.26}$$

Proof. See [6, Lemma 2.1] with $\beta = 1$. □

We start our analysis with a simple observation.

Lemma 3.4 At iteration k before termination of the TRqDA algorithm, define

$$\widehat{\phi}_{f,k} \overset{\text{def}}{=} \frac{j! \, \overline{\phi}_{f,j}^{\delta_k}(x_k)}{\delta_k^j}, \tag{3.27}$$

where j is the index for which $\overline{\phi}_{f,j}^{\delta_k}(x_k) > \epsilon_j/(1+\omega)\delta_k^j/j!$ in Step 1 of the iteration. Then

$$\min_{i \in \{0,\ldots,k\}} \widehat{\phi}_{f,i} \geq \frac{\epsilon_{\min}}{1 + \omega} \tag{3.28}$$

where $\epsilon_{\min} = \min_{j \in \{1,\ldots,q\}} \epsilon_j$. Moreover,

$$T_{f,j}(x_k, 0) - T_{f,j}(x_k, s_k) \geq \widehat{\phi}_{f,k} \frac{\delta_k^j}{j!} \tag{3.29}$$

Proof. Let k be the index of an iteration before termination. Then the mechanism of Step 1 ensures the existence of j such that

$$\overline{\phi}_{f,k}^{\delta_k}(x_k) > \frac{\epsilon_j}{1 + \omega} \frac{\delta_k^j}{j!} \geq \frac{\epsilon_{\min}}{1 + \omega} \left(\frac{\delta_k^j}{j!} \right). \tag{3.30}$$

The definiton of $\widehat{\phi}_{f,k}$ then directly implies that

$$\widehat{\phi}_{f,k} \geq \frac{\epsilon_{\min}}{1 + \omega}. \tag{3.31}$$

Since termination has not yet occurred at iteration k, the same inequality must hold for all iterations $i \in \{0, \ldots, k\}$, yielding (3.28). The bound (3.29) directly results from

$$T_{f,j}(x_k, 0) - T_{f,j}(x_k, s_k) \geq T_{f,j}(x_k, 0) - T_{f,j}(x_k, d_k) = \overline{\phi}_{f,j}^{\delta_k}(x_k) = \widehat{\phi}_{f,k} \frac{\delta_k^j}{j!},$$

where we have used (3.2) to derive the first inequality and the definitions of $\overline{\phi}_{f,j}^{\delta_k}(x_k)$ and $\widehat{\phi}_{f,k}$ to obtain the equalities. □

We now derive an "inexact" variant of the condition that ensures that an iteration is very successful.

Lemma 3.5 Suppose that AS.1 holds, and that $\widehat{\phi}_{f,k}$ is defined by (3.27). Suppose also that

$$\Delta_k \leq \min\left\{\vartheta, \frac{1-\eta_2}{4\max[1, L_f]}\widehat{\phi}_{f,k}\right\} \tag{3.32}$$

at iteration k of Algorithm 3.1. Then $\rho_k \geq \eta_2$, iteration k is very successful and $\Delta_{k+1} \geq \Delta_k$.

Proof. We first note that (3.32) implies that $\delta_k = \min[\vartheta, \Delta_k] = \Delta_k$. Then we may use (3.5), the triangle inequality, (3.3) and (3.4) and (3.14) (see Lemma 3.2) successively to deduce that

$$
\begin{aligned}
|\rho_k - 1| &\leq \frac{|\overline{f}(x_k+s_k) - \overline{T}_{f,j}(x_k,s_k)|}{\overline{\Delta T}_{f,j}(x_k,s_k)} \\
&\leq \frac{1}{\overline{\Delta T}_{f,j}(x_k,s_k)}\Big[|\overline{f}(x_k+s_k) - f(x_k+s_k)| \\
&\quad + |f(x_k+s_k) - T_{f,j}(x_k,s_k)| + |\overline{T}_{f,j}(x_k,s_k) - T_{f,j}(x_k,s_k)|\Big] \\
&\leq \frac{1}{\overline{\Delta T}_{f,j}(x_k,s_k)}\Big[|f(x_k+s_k) - T_{f,j}(x_k,s_k)| + 3\omega\overline{\Delta T}_{f,j}(x_k,s_k)\Big].
\end{aligned}
$$

Invoking (3.26) in Lemma 3.3, the bound $\|s_k\| \leq \Delta_k = \delta_k$, (3.25), (3.29), the fact that $\omega \leq \frac{1}{4}(1-\eta_2)$, and (3.32), we deduce that

$$|\rho_k - 1| \leq \frac{L_{f,j}\,\delta_k^{j+1}}{(j+1)\,\delta_k^j\widehat{\phi}_{f,k}} + 3\omega \leq \frac{L_f\Delta_k}{\widehat{\phi}_{f,k}} + \tfrac{3}{4}(1-\eta_2) \leq 1 - \eta_2$$

and thus that $\rho_k \geq \eta_2$. Then iteration k is very successful and (3.6) then yields that $\Delta_{k+1} \geq \Delta_k$. □

This allows us to derive lower bounds on the trust-region radius and the model decrease.

Lemma 3.6 Suppose that AS.1 holds. Then, for all $k \geq 0$,

$$\Delta_k \geq \min\left\{\vartheta, \kappa_\Delta \min_{i\in\{0,\dots,k\}}\widehat{\phi}_{f,i}\right\} \tag{3.33}$$

where $\widehat{\phi}_{f,i}$ is defined in (3.27) and

$$\kappa_\Delta \stackrel{\text{def}}{=} \frac{\gamma_1(1-\eta_2)}{\max[1, L_f]}\min\left[\vartheta, \frac{\Delta_0 \min_{j\in\{1,\dots,q\}}\delta_{0,j}^j}{2q(\max_{i\in\{1,\dots,q\}}\|\nabla_x^i f(x_0)\| + \kappa_\zeta)}\right] \tag{3.34}$$

Proof. Note that, using (3.27), (2.4) and the bounds $\|\overline{\nabla_x^i f}(x_0)\| \le \|\nabla_x^i f(x_0)\| + \kappa_\zeta$ and $\delta_{0,j} \le 1$, we have that

$$
\begin{aligned}
\widehat{\phi}_{f,0} &\le \max_{j \in \{1,\dots,q\}} \frac{j! \, \overline{\phi}_{f,j}^{\delta_{0,j}}(x_0)}{\delta_{0,j}^j} \\[2ex]
&\le q \max_{j \in \{1,\dots,q\}} \frac{\max_{i \in \{1,\dots,j\}} \|\overline{\nabla_x^i f}(x_0)\|}{\delta_{0,j}^j} \sum_{i=1}^{j} \frac{\delta_{0,i}^i}{i!} \\[2ex]
&\le q \frac{\max_{i \in \{1,\dots,q\}} \|\overline{\nabla_x^i f}(x_0)\|}{\min_{j \in \{1,\dots,q\}} \delta_{0,j}^j} \left(\exp(\delta_{0,j}) - 1 \right) \\[2ex]
&\le 2q \frac{\max_{i \in \{1,\dots,q\}} \|\nabla_x^i f(x_0)\| + \kappa_\zeta}{\min_{j \in \{1,\dots,q\}} \delta_{0,j}^j}
\end{aligned}
$$

and thus, since $\gamma_1(1 - \eta_2) < 1$,

$$
\kappa_\Delta \le \frac{\gamma_1(1 - \eta_2)}{\max[1, L_f]} \min \left[\vartheta, \frac{\Delta_0}{\widehat{\phi}_{f,0}} \right] \le \frac{\Delta_0}{\widehat{\phi}_{f,0}}.
$$

As a consequence, (3.33) holds for $k = 0$. Suppose now that $k \ge 1$ is the first iteration such that (3.33) is violated. The updating rule (3.6) then ensures that

$$
\Delta_{k-1} < \frac{1 - \eta_2}{L_f} \min_{i \in \{0,\dots,k\}} \widehat{\phi}_{f,i} \ \text{ and } \ \Delta_{k-1} \le \vartheta. \tag{3.35}
$$

Moreover, since

$$
\widehat{\phi}_{f,k-1} \ge \min_{i \in \{0,\dots,k-1\}} \widehat{\phi}_{f,i} \ge \min_{i \in \{0,\dots,k\}} \widehat{\phi}_{f,i}
$$

we deduce that

$$
\frac{\Delta_{k-1}}{\widehat{\phi}_{f,k-1}} \le \frac{\Delta_{k-1}}{\min_{i \in \{0,\dots,k-1\}} \widehat{\phi}_{f,i}} \le \frac{\Delta_{k-1}}{\min_{i \in \{0,\dots,k\}} \widehat{\phi}_{f,i}} < \frac{1 - \eta_2}{L_f}. \tag{3.36}
$$

Lemma 3.5 and the second part of (3.35) then ensure that $\Delta_{k-1} \le \Delta_k$. Using this bound, the second inequality of (3.36) and the fact that (3.33) is violated at iteration k, we obtain that

$$
\frac{\Delta_{k-1}}{\min_{i \in \{0,\dots,k-1\}} \widehat{\phi}_{f,i}} \le \frac{\Delta_k}{\min_{i \in \{0,\dots,k\}} \widehat{\phi}_{f,i}} < \gamma_1 \frac{1 - \eta_2}{L_f}
$$

As a consequence (3.33) is also violated at iteration $k - 1$. But this contradicts the assumption that iteration k is the first such that this inequality fails. This latter assumption is thus impossible, and no such iteration can exist. □

Lemma 3.7 Suppose that AS.1 holds. Then, for all $k \geq 0$ before termination,

$$T_{f,j}(x_k, 0) - T_{f,j}(x_k, s_k) \geq \frac{\kappa_\delta^{q+1}}{q!} \epsilon_{\min}^{q+1}, \tag{3.37}$$

where

$$\kappa_\delta \stackrel{\text{def}}{=} \frac{\kappa_\Delta}{1+\omega}. \tag{3.38}$$

Proof. Suppose first that $\Delta_k > \vartheta$ and therefore $\delta_k = \vartheta$. Then (3.29) and the bound $\vartheta \geq \epsilon_{\min}$ give that

$$T_{f,j}(x_k, 0) - T_{f,j}(x_k, s_k) \geq \frac{\vartheta^j}{j!} \widehat{\phi}_{f,k} \geq \frac{\vartheta^q}{q!} \widehat{\phi}_{f,k} \geq \frac{\epsilon_{\min}^q}{q!} \frac{\epsilon_{\min}}{1+\omega},$$

which yields (3.37) since $\kappa_\Delta < 1$. If $\Delta_k \leq \vartheta$, then $\delta_k = \Delta_k$, and (3.33) implies that

$$\delta_k \geq \min\left\{ \vartheta, \kappa_\Delta \min_{i \in \{0,\dots,k\}} \widehat{\phi}_{f,i} \right\}.$$

Therefore, using (3.29) again,

$$
\begin{aligned}
T_{f,j}(x_k, 0) - T_{f,j}(x_k, s_k) &\geq \frac{1}{j!} \widehat{\phi}_{f,k} \min\left\{ \vartheta, \kappa_\Delta \min_{i \in \{0,\dots,k\}} \widehat{\phi}_{f,k} \right\}^j \\
&\geq \frac{1}{q!} \min\left\{ \vartheta, \kappa_\Delta \min_{i \in \{0,\dots,k\}} \widehat{\phi}_{f,k} \right\}^{q+1}.
\end{aligned} \tag{3.39}
$$

Moreover, if termination hasn't occurred at iteration k, we have that (3.28) holds, and, because $\kappa_\delta \leq 1$ and $\vartheta \geq \epsilon_{\min}$, (3.39) in turn implies (3.37). $\qquad\square$

We may now state the complexity bound for the TRqDA algorithm.

Theorem 3.8 Suppose that AS.1 and AS.2 hold. Then there exist positive constants $\kappa^{\rm A}_{\rm TR\mathit{q}DA}$, $\kappa^{\rm B}_{\rm TR\mathit{q}DA}$, $\kappa^{\rm C}_{\rm TR\mathit{q}DA}$, $\kappa^{\rm D}_{\rm TR\mathit{q}DA}$, $\kappa^{\rm E}_{\rm TR\mathit{q}DA}$ and $\kappa^{\rm F}_{\rm TR\mathit{q}DA}$ such that, for any $\epsilon \in (0,1]^q$, the TRqDA algorithm requires at most

$$
\#^{\rm F}_{\rm TR\mathit{q}DA} \overset{\rm def}{=} \kappa^{\rm A}_{\rm TR\mathit{q}DA} \frac{f(x_0) - f_{\rm low}}{\min\limits_{j \in \{1,\dots,q\}} \epsilon_j^{q+1}} + \kappa^{\rm B}_{\rm TR\mathit{q}DA} \left| \log\left(\min_{j \in \{1,\dots,q\}} \epsilon_j \right) \right| + \kappa^{\rm C}_{\rm TR\mathit{q}DA}
$$

$$
= \mathcal{O}\left(\max_{j \in \{1,\dots,q\}} \epsilon_j^{-(q+1)} \right)
$$

(3.40)

(inexact) evaluations of f and at most

$$
\#^{\rm D}_{\rm TR\mathit{q}DA} \overset{\rm def}{=} \kappa^{\rm D}_{\rm TR\mathit{q}DA} \frac{f(x_0) - f_{\rm low}}{\min\limits_{j \in \{1,\dots,q\}} \epsilon_j^{q+1}} + \kappa^{\rm E}_{\rm TR\mathit{q}DA} \left| \log\left(\min_{j \in \{1,\dots,q\}} \epsilon_j \right) \right| + \kappa^{\rm F}_{\rm TR\mathit{q}DA}
$$

$$
= \mathcal{O}\left(\max_{j \in \{1,\dots,q\}} \epsilon_j^{-(q+1)} \right)
$$

(3.41)

(inexact) evaluations of $\{\nabla_x^f\}_{j=1}^q$ to produce an iterate x_ϵ and an optimality radius $\delta_\epsilon \in (0,1]$ such that $\phi_{f,q}^{\delta_\epsilon}(x_\epsilon) \leq \epsilon_j \delta_\epsilon^j / j!$ for all $j \in \{1,\dots,q\}$,

Proof. If i is the index of a successful iteration before termination, we have that

$$
\begin{aligned}
f(x_i) - f(x_{i+1}) &\geq & [\overline{f}(x_i) - \overline{f}(x_{i+1})] - 2\omega \overline{\Delta T}_{f,j}(x_i, s_i) \\
&\geq & \eta_1 \overline{\Delta T}_{f,j}(x_i, s_i) - 2\omega \overline{\Delta T}_{f,j}(x_i, s_i) \\
&\geq & \frac{(\eta_1 - 2\omega)\kappa_\delta^{q+1}}{q!} \epsilon_{\min}^{q+1} > 0
\end{aligned}
$$

(3.42)

using successively (3.3) and (3.4) (3.5), (3.37) and the requirement that $\omega < \frac{1}{2}\eta_1$. Now let k be the index of an arbitrary iteration before termination. Using AS.2, the nature of successful iterations and (3.42), we deduce that

$$
f(x_0) - f_{\rm low} \geq f(x_0) - f(x_{k+1}) = \sum_{i \in \mathcal{S}_k} [f(x_i) - f(x_{i+1})] \geq |\mathcal{S}_k| \, [\kappa^{\rm S}_{\rm TR\mathit{q}DA}]^{-1} \, \epsilon_{\min}^{q+1},
$$

where

$$
\kappa^{\rm S}_{\rm TR\mathit{q}DA} = \frac{q!}{(\eta_1 - 2\omega)\kappa_\delta^{q+1}},
$$

(3.43)

and thus that the total number of *successful iterations* before termination is given by

$$
|\mathcal{S}_k| \leq \kappa^{\rm S}_{\rm TR\mathit{q}DA} \frac{f(x_0) - f_{\rm low}}{\epsilon_{\min}^{q+1}}.
$$

(3.44)

Observe now that combining respectively (3.33), (3.34) and (3.28), we obtain that

$$
\Delta_k \geq \min(\vartheta, \Delta_k) \geq \min\left(\vartheta, \kappa_\Delta \min_{i \in \{0,\dots,k\}} \widehat{\phi}_{f,i} \right) = \kappa_\Delta \min_{i \in \{0,\dots,k\}} \widehat{\phi}_{f,i} \geq \kappa_\delta \epsilon_{\min},
$$

(3.45)

We may then invoke Lemma 3.1 to deduce that the total number of iterations required is bounded by

$$|\mathcal{S}_k| \left(1 + \frac{\log \gamma_3}{|\log \gamma_2|} \right) + \frac{1}{|\log \gamma_2|} \left(|\log(\epsilon_{\min})| + \left| \log \left(\frac{\kappa_\delta}{\Delta_0} \right) \right| \right) + 1.$$

and hence the total number of approximate function evaluations is at most twice this number, which yields (3.40) with the coefficients (3.43),

$$\kappa_{\text{TR}q\text{DA}}^{\text{A}} \overset{\text{def}}{=} 2\kappa_{\text{TR}q\text{DA}}^{\text{S}} \left(1 + \frac{\log \gamma_3}{|\log \gamma_2|} \right) \quad \kappa_{\text{TR}q\text{DA}}^{\text{B}} \overset{\text{def}}{=} \frac{2}{|\log \gamma_2|} \tag{3.46}$$

and

$$\kappa_{\text{TR}q\text{DA}}^{\text{C}} \overset{\text{def}}{=} \frac{2}{|\log \gamma_2|} \left| \log \left(\frac{\kappa_\delta}{\Delta_0} \right) \right| + \frac{2}{|\log(\gamma_\zeta)|} + 2. \tag{3.47}$$

In order to derive an upper bound on the the number of derivatives' evaluations, we now have to count the number of additional derivative evaluations caused by the need to approximate them to the desired accuracy. Observe that repeated evaluations at a given iterate x_k are only needed when the current values of the absolute errors are smaller than used previously at x_k. These absolute errors are, by construction, linearly decreasing with rate γ_ζ, Indeed, they are initialised in Step 0 of the TRqDA algorithm, decreased each time by a factor γ_ζ in (2.17) invoked in Step 1.4 of Algorithm 2.2, down to values $\{\zeta_{j,i_\zeta}\}_{j=1}^q$ which are then passed to the modified Step 2, and possibly decreased there further in (3.12) in Step 2.3 of Algorithm 3.2 again by successive multiplication by γ_ζ. We now use (2.20) in Lemma 2.2 and (3.13) in Lemma 3.2 to deduce that the maximal absolute accuracy, $\max_{i \in \{1,\dots,j\}} \zeta_{i,i_\zeta}$, will not be reduced below the value

$$\min \left[\frac{\omega}{4} \epsilon_j \frac{\delta_k^{j-1}}{j!}, \frac{\omega}{8(1+\omega)} \epsilon_j \frac{\delta_k^{j-1}}{j!} \right] = \frac{\omega}{8(1+\omega)} \epsilon_j \frac{\delta_k^{j-1}}{j!} \tag{3.48}$$

at iteration k. But we may now deduce from (3.45) that

$$\delta_k = \min(\vartheta, \Delta_k) \geq \kappa_\delta \epsilon_{\min}. \tag{3.49}$$

This and (3.48) in turn implies that no further reduction of the $\{\zeta_j\}_{j=1}^q$, and hence no further approximation of $\{\overline{\nabla_x^j f}(x_k)\}_{j=1}^q$, can possibly occur in any iteration once the largest initial absolute error $\max_{j \in \{1,\dots,q\}} \zeta_{j,0}$ has been reduced by successive multiplications by γ_ζ sufficiently to ensure that

$$\gamma_\zeta^{i_\zeta} [\max_{j \in \{1,\dots,q\}} \zeta_{j,0}] \leq \frac{\omega \kappa_\delta^{q-1} \epsilon_{\min}^q}{8(1+\omega)} \leq \frac{\omega}{8(1+\omega)} \epsilon_j \frac{(\kappa_\delta \epsilon_{\min})^{j-1}}{j!} \leq \frac{\omega}{8(1+\omega)} \epsilon_j \frac{\delta_k^{j-1}}{j!}. \tag{3.50}$$

Since the $\zeta_{j,0}$ are initialised in the TRqDA algorithm so that $\max_{j \in \{1,\dots,q\}} \zeta_{j,0} \leq \kappa_\zeta$, the bound (3.50) is achieved once i_ζ, the number of decreases in $\{\zeta_j\}_{j=1}^q$, is large enough to guarantee that

$$\gamma_\zeta^{i_\zeta} \kappa_\zeta \leq \kappa_{\text{acc}} \epsilon_{\min}^q \quad \text{where} \quad \kappa_{\text{acc}} \overset{\text{def}}{=} \frac{\omega \kappa_\delta^{q-1}}{8(1+\omega)},$$

which is equivalent to asking

$$i_\zeta \log(\gamma_\zeta) + \log(\kappa_\zeta) \leq q \log(\epsilon_{\min}) + \log(\kappa_{\text{acc}}). \tag{3.51}$$

We now recall that Step 1 of the TRqDA algorithm is only used (and derivatives evaluated) after successful iterations. As a consequence, we deduce that the number of evaluations of the derivatives of the objective function that occur during the course of the TRpDA algorithm before termination is at most

$$|\mathcal{S}_k| + i_{\zeta,\min}, \tag{3.52}$$

i.e., the number iterations in (3.44) plus

$$
\begin{aligned}
i_{\zeta,\min} \;&\stackrel{\text{def}}{=}\; \left\lfloor \frac{1}{\log(\gamma_\zeta)} \left\{ q|\log(\epsilon_{\min})| + \left|\log\left(\frac{\kappa_{\text{acc}}}{\kappa_\zeta}\right)\right| \right\} \right\rfloor \\
&\leq\; \frac{q}{|\log(\gamma_\zeta)|} |\log(\epsilon_{\min})| + \frac{1}{|\log(\gamma_\zeta)|} \left|\log\left(\frac{\kappa_{\text{acc}}}{\kappa_\zeta}\right)\right| + 1,
\end{aligned}
$$

the smallest value of i_ζ that ensures (3.51). Adding one for the final evaluation at termination, this leads to the desired evaluation bound (3.41) with the coefficients

$$
\kappa_{\text{TRqDA}}^{\text{D}} = \kappa_{\text{TRsDA}}^{\text{S}}, \quad \kappa_{\text{TRqDA}}^{\text{E}} \stackrel{\text{def}}{=} \frac{q}{|\log\gamma_\zeta|} \;\text{ and }\; \kappa_{\text{TRqDA}}^{\text{F}} \stackrel{\text{def}}{=} \frac{1}{|\log(\gamma_\zeta)|} \left|\log\left(\frac{\kappa_{\text{acc}}}{\kappa_\zeta}\right)\right| + 2.
$$

$\square$

4 Discussion of the TRqDA algorithm

In order to further avoid overloading notation and over-complicating arguments, we have made a few simplifying assumptions in the description of the TRqDA algorithm. The first is that, when accuracy is tightened in Steps 1.4 and 2.4, we have stipulated a uniform improvement for all derivatives of orders one to q. A more refined version of the algorithm is obviously possible in which the need to improve accuracy for each derivative is considered separately, and that requires sufficient accuracy on each of the approximate derivatives $\{\overline{\nabla_x^j f}(x_k)\}_{j=1}^q$. Assuming that $\|s_k\| \leq 1$ and remembering that $\|d_k\| \leq \delta_k \leq \vartheta \leq 1$, we might instead consider imposing derivative-specific absolute accuracy requirements

$$\|\overline{\nabla_x^\ell f}(x_k) - \nabla_x^\ell f(x_k)\| \leq \frac{\omega}{3\|s_k\|^\ell} \overline{\Delta T}_{f,p}(x_k, s_k), \quad (\ell \in \{1,\ldots,q\}), \tag{4.1}$$

and

$$\|\overline{\nabla_x^\ell f}(x_k) - \nabla_x^\ell f(x_k)\| \leq \frac{\omega}{3\|d_k\|^\ell} \overline{\Delta T}_{f,j}(x_k, d_k), \quad (j \in \{1,\ldots,q\}, \ell \in \{1,\ldots,j\}), \tag{4.2}$$

rather than (2.2) applied to the directions s_k and d_k (remember that they are the only directions used in the VERIFY tests in Algorithms 2.2 and 3.2.) One can then verify that (2.2) still holds for suitable x and s in the computation of $m_k(s_k)$ and $\overline{\phi}_{f,j}^{\delta_k}(x_k)$. To see this, consider the accuracy of the Taylor series for f evaluated at a general step s, where s is either

s_k or d_k. Using (4.1) or (4.2), we have that, for any $j \in \{1, \ldots, q\}$,

$$
\begin{aligned}
|\overline{\Delta T}_{f,j}(x_k, s) - \Delta T_{f,j}(x_k, s)| &\leq \sum_{\ell=1}^{j} \frac{\|s\|^\ell}{\ell!} \|\overline{\nabla_x^\ell f}(x_k) - \nabla_x^\ell f(x_k)\| \\
&\leq \sum_{\ell=1}^{j} \frac{\omega}{3\ell!} \overline{\Delta T}_{f,j}(x_k, s) \\
&\leq \frac{1}{3} \left(\sum_{\ell=1}^{j} \frac{1}{\ell!} \right) \omega \overline{\Delta T}_{f,j}(x_k, s) \\
&\leq \frac{1}{3} \left(\sum_{i=0}^{j} \frac{1}{i!} \right) \omega \overline{\Delta T}_{f,j}(x_k, s) \\
&< \omega \overline{\Delta T}_{f,j}(x_k, s).
\end{aligned}
$$

Thus (4.1) and (4.2) guarantee that (2.2) holds both for $x = x_k$ and $s = s_k$ when $\|s_k\| < 1$ and for $x = x_k$ and $s = d_k$, as occurring in the computation of $\overline{\phi}_{f,j}^{\delta_k}(x_k)$.

Of course, the detailed "derivative by derivative" conditions (4.1) and (4.2) make no attempt to exploit possible balancing effects between terms of different degrees ℓ in the Taylor-series model and, in that sense, are more restrictive than (2.2). However they illustrate an important point: since the occurrence of small $\|s_k\| < 1$ and $\|d_k\|$ can be expected to happen overwhelmingly often when convergence occurs, the above conditions indicate that the accuracy requirements on derivatives become looser for higher-degree derivatives. This is reminiscent of the situation where a quadratic is minimized using conjugate-gradients with inexact products, a situation for which various authors [16, 17, 11] have shown that the accuracy of the products with the Hessian may be progressively relaxed without affecting convergence.

A second simplifying feature of the TRqDA algorithm relates to the insistence that the absolute accuracies $\{\zeta_{i,i_\zeta}\}_{i=1}^{q}$ are initialised in Step 0 once and for all. As a consequence, the accuracy requirements can only become more severe as the iteration proceeds. This might well be viewed as inefficient because the true need for accurate derivatives depends more on their values at a individual rather than the evolving set of iterates. A version of the algorithm for which the $\{\zeta_{i,i_\zeta}\}_{i=1}^{q}$ are reinitialised at every successful iterate is of course possible, at a moderate increase in the overall complexity bound. Indeed, in such a case, the number of "additional" derivatives evaluations $i_{\zeta,\max}$ (in the proof of Theorem 3.8) would no longer need to cover all iterations, but only what happens at a single iterate. Thus the logarithmic term in $\epsilon_{\min}$ is no longer added to the number of successful iterations, but multiplies it, and the worst-case evaluation complexity for the modified algorithm becomes $\mathcal{O}\left(|\log(\epsilon_{\min})|\epsilon_{\min}^{-(q+1)}\right)$.

5 Conclusions

We have presented an inexact trust-region algorithm using high-order models and capable of finding high-order strong approximate minimizers. We have then shown that it will find such a q-th order approximate minimizer in at most $\mathcal{O}\left(\min_{j \in \{1,\ldots,q\}} \epsilon_j^{-(q+1)}\right)$ inexact evaluations of the objective function and its derivatives. Obviously, the results presented also cover the case when the function and derivatives evaluations are exact (and ω can be set to zero).

References

[1] S. Bellavia, G. Gurioli, B. Morini, and Ph. L. Toint. Adaptive regularization algorithms with inexact evaluations for nonconvex optimization. *SIAM Journal on Optimization*, 29(4):2881–2915, 2019.

[2] S. Bellavia, G. Gurioli, B. Morini, and Ph. L. Toint. High-order evaluation complexity of a stochastic adaptive regularization algorithm for nonconvex optimization using inexact function evaluations and randomly perturbed derivatives. arXiv:2005.04639, 2020.

[3] E. G. Birgin, J.-L. Gardenghi, J. M. Martínez, and S. A. Santos. On the use of third-order models with fourth-order regularization for unconstrained optimization. *Optimization Letters*, 14:815–838, 2020.

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

[5] C. Cartis, N. I. M. Gould, and Ph. L. Toint. On the oracle complexity of first-order and derivative-free algorithms for smooth nonconvex minimization. *SIAM Journal on Optimization*, 22(1):66–86, 2012.

[6] C. Cartis, N. I. M. Gould, and Ph. L. Toint. Sharp worst-case evaluation complexity bounds for arbitrary-order nonconvex optimization with inexpensive constraints. *SIAM Journal on Optimization*, 30(1):513–541, 2020.

[7] C. Cartis, N. I. M. Gould, and Ph. L. Toint. Strong evaluation complexity bounds for arbitrary-order optimization of nonconvex nonsmooth composite functions. arXiv:2001.10802, 2020.

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

[9] A. R. Conn, K. Scheinberg, and Ph. L. Toint. On the convergence of derivative-free methods for unconstrained optimization. In A. Iserles and M. Buhmann, editors, *Approximation Theory and Optimization: Tributes to M. J. D. Powell*, pages 83–108, Cambridge, England, 1997. Cambridge University Press.

[10] A. R. Conn, K. Scheinberg, and L. N. Vicente. *Introduction to Derivative-free Optimization*. MPS-SIAM Series on Optimization. SIAM, Philadelphia, USA, 2009.

[11] S. Gratton, E. Simon, D. Titley-Peloquin, and Ph. L. Toint. Minimizing convex quadratics with variable precision Krylov methods. *Numerical Linear Algebra with Applications*, (to appear), 2020.

[12] S. Gratton, E. Simon, and Ph. L. Toint. An algorithm for the minimization of nonsmooth nonconvex functions using inexact evaluations and its worst-case complexity. *Mathematical Programming, Series A*, (to appear), 2020.

[13] W. Hogan. Point-to-set maps in mathematical programming. *SIAM Review*, 15(3):591–603, 1973.

[14] R. J. LeVecque. *Finite Difference Methods for Ordinary and Partial Differential Equations: Steady-State and Time-Dependent Problems*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2020.

[15] M. J. D. Powell. A direct search optimization method that models the objective by quadratic interpolation. Presentation at the 5th Stockholm Optimization Days, Stockholm, 1994.

[16] V. Simoncini and D. Szyld. Theory of inexact Krylov subspace methods and applications to scientific computing. *SIAM Journal on Scientific Computing*, 25(2):454–477, 2003.

[17] J. van den Eshof and G. L. G. Sleijpen. Inexact Krylov subspace methods for linear systems. *SIAM Journal on Matrix Analysis and Applications*, 26(1):125–153, 2004.

[18] D. Winfield. *Function and functional optimization by interpolation in data tables*. PhD thesis, Harvard University, Cambridge, USA, 1969.

[19] D. Winfield. Function minimization by interpolation in a data table. *Journal of the Institute of Mathematics and its Applications*, 12:339–347, 1973.