An optimally fast objective-function-free minimization algorithm using random subspaces

Stefania Bellavia[*], Serge Gratton[†], Benedetta Morini[*], Philippe L. Toint[‡]

24 X 2023

Abstract

An algorithm for unconstrained non-convex optimization is described, which does not evaluate the objective function and in which minimization is carried out, at each iteration, within a randomly selected subspace. It is shown that this random approximation technique does not affect the method's convergence nor its evaluation complexity for the search of an ϵ-approximate first-order critical point, which is $\mathcal{O}(\epsilon^{-(p+1)/p})$, where p is the order of derivatives used. A variant of the algorithm using approximate Hessian matrices is also analyzed and shown to require at most $\mathcal{O}(\epsilon^{-2})$ evaluations. Preliminary numerical tests show that the random-subspace technique can significantly improve performance on some problems, albeit, unsurprisingly, not for all.

Keywords: nonlinear optimization, stochastic adaptive regularization methods, sketching, evaluation complexity, objective-function-free optimization (OFFO).

1 Introduction

Recent years have seen the emergence of random concepts in iterative algorithms for nonconvex optimization (see [12] and reference therein and [1, 2, 3, 5, 26]). In particular, several authors [14, 16, 27, 29] have suggested algorithms where the search for a better iterate is carried out in random subspaces of the space of variables, instead of, as is more traditional and often more costly, in the complete space. In these proposals, the Johnson-Lindenstrauss embedding Lemma (see [13] for a simple exposition) is used to ensure that the relevant information can be very efficiently found in the selected subspace with high probability, and this leads to an elegant analysis yielding optimal complexity bounds for "random-subspace" variants of the standard trust-region and adaptive regularization methods for unconstrained minimization. In parallel with this interesting development, alternative non-standard optimization methods have also been introduced, notably motivated by applications in neural network training for deep learning, where the objective function of the problem is never computed (these algorithms use derivatives' values only). This new class of "objective-function-free optimization" (OFFO) methods includes such popular first-order algorithms as ADAM or ADAGRAD, and has been investigated for instance in [15, 24, 28, 19, 32].

It is the purpose of the present paper to discuss an algorithm which combines these two ideas for the first time while maintaining the desirable properties of both. More specifically, we describe an OFFO adaptive regularization method using first- or higher-order models defined in random

[1]Dipartimento di Ingegneria Industriale, Università degli Studi di Firenze, Italy. Member of the INdAM Research Group GNCS. Email: stefania.bellavia@unifi.it, benedetta.morini@unifi.it. Work partially supported by MUR (Ministero dell'Università e della Ricerca) through Progetto PRIN 2022 "Numerical Optimization with Adaptive Accuracy and Applications to Machine Learning" and by GNCS through Progetti di Ricerca GNCS.

[2]Université de Toulouse, INP, IRIT, Toulouse, France. Work partially supported by 3IA Artificial and Natural Intelligence Toulouse Institute (ANITI), French "Investing for the Future - PIA3" program under the Grant agreement ANR-19-PI3A-0004. Email: serge.gratton@enseeiht.fr.

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

subspaces, and show that this algorithm still enjoys the optimal global rate of convergence known for comparable adaptive regularization methods. Independently of the practical interest for such a method, which we argue below has to evaluated on a problem-by-problem basis, our analysis is a new step in the "information thinning" question, which is to isolate what information is necessary for a minimization method to achieve optimal complexity. Indeed, while [20] proves that function values are unnecessary, the present paper further shows that this is also the case for "full space" information[1] under suitable probabilistic assumptions.

Our approach has a further advantage compared to existing proposals, like the random-subspace trust-region and random-subspace regularization methods of [27] and [6]. Because no evaluation of the objective function is involved, the algorithm generates a much simpler random process (there is now only one random event per iteration), in turn considerably simplifying the proofs as the number of iteration types whose number must be estimated (in [27, Chapter 4]) is now reduced to only two. While our theory covers the case where derivatives of higher order than one are estimated, our practical focus will be on its first-order variant, mostly, as we discuss in Section 4, because it is applicable to a much larger class of problems.

The paper is organized as follows. The new algorithm is proposed in Section 2, while its evaluation complexity is analyzed under general embedding conditions in Section 3. A brief discussion of a possible way to select the random subspaces are presented in Section 4. The numerical behaviour of the first-order variant is illustrated in Section 5. Some conclusions are finally presented in Section 6. A discussion of a variant using quadratically regularized inexact quadratic models is proposed and analyzed in appendix.

2 An OFFO adaptive regularization algorithm using random subspaces

The problem of interest in what follows is the standard nonconvex unconstrained minimization of a (sufficiently) smooth objective function, that is

$$\min_{x\in\mathbb{R}^n} f(x), \tag{1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$. As indicated in the introduction, our aim is to design an adaptive regularization algorithm in which the objective function value is never computed, and in which the step is computed by approximately minimizing a suitable model of the objective function in a random subspace. To ensure that this approach is sensible, we make the following assumptions.

AS.1 f is p times continuously differentiable in $\mathbb{R}^n$.
AS.2 There exists a constant $f_{\rm low}$ such that $f(x) \geq f_{\rm low}$ for all $x \in \mathbb{R}^n$.
AS.3 The pth derivative of f is globally Lipschitz continuous, that is, there exist a non-negative constant L_p such that

$$\|\nabla_x^p f(x) - \nabla_x^p f(y)\| \leq L_p\|x - y\| \ \text{ for all } x, y \in \mathbb{R}^n,$$

where $\|.\|$ denotes the Euclidean norm for vectors in $\mathbb{R}^n$ and the corresponding subordinate norm for tensors.
AS.4 The gradient of f is bounded, that is there exists a constant $\kappa_{\rm g} \geq 0$ such that, for all $x \in \mathbb{R}^n$,

$$\|\nabla_x^1 f(x)\| \leq \kappa_{\rm g}.$$

AS.5 If $p > 1$, there exists a constant $\kappa_{\rm high} \geq 0$ such that

$$\min_{\|d\|\leq 1} \nabla_x^i f(x)[d]^i \geq -\kappa_{\rm high} \ \text{ for all } \ x \in \mathbb{R}^n \ \text{ and } \ i \in \{2, \ldots, p\},$$

[1] One might argue that it has long been known that information along the directions given by the gradient and the step suffices, but this requires the step to be known and thus amounts to an *a posteriori* observation instead of an *a priori* algorithmically exploitable strategy.

where $\nabla_x^i f(x)$ is the ith derivative tensor of f computed at x, and where $T[d]^i$ denotes the i-dimensional tensor T applied on i copies of the vector d. (For notational convenience, we set $\kappa_{\rm high} = 0$ if $p = 1$.)

We refer the reader to [10, Appendix 6] for details on derivative tensors. Observe that AS.5 is irrelevant in the case where $p = 1$, which is of special interest here. Should one be interested in higher-order methods, AS.5 is weaker than assuming uniform boundedness of the derivative tensors of degree two and above (there is no upper bound on the value of $\nabla_x^i f(x)[d]^i$), or, equivalently, Lipschitz continuity of derivatives of degree one to $p-1$.

2.1 The SKOFFARp algorithm

As suggested above, adaptive regularization methods are iterative schemes which compute a step from an iterate x_k to the next by approximately minimizing a p-th degree regularized model $m_k(s)$ of $f(x_k + s)$ of the form

$$m_k(s) \stackrel{\rm def}{=} T_{f,p}(x_k, s) + \frac{\sigma_k}{(p+1)!}\|s\|^{p+1}, \tag{2}$$

where $T_{f,p}(x, s)$ is the pth order Taylor expansion of functional f at x truncated at order p, that is,

$$T_{f,p}(x, s) \stackrel{\rm def}{=} f(x) + \sum_{i=1}^{p} \frac{1}{i!}\nabla_x^i f(x)[s]^i. \tag{3}$$

To obtain the model (5), the p-th order Taylor series (6) is "regularized" by adding the term $\frac{\sigma_k}{(p+1)!}\|s\|^{p+1}$ (where σ_k is the iteration-dependent regularization parameter), thereby ensuring that $m_k(s)$ is bounded below and that a step s_k (approximately) minimizing this model is well-defined.

Following [27], we propose to compute a random subspace step at iteration k as follows. Given an iteration-independent distribution $\mathcal{S}$ of $\ell \times n$ random matrices (with $\ell < n$), let S_k be drawn from this distribution and consider minimizing the sketched regularized model

$$\widehat{m}_k(\widehat{s}) \stackrel{\rm def}{=} \widehat{T}_{f,p}(x_k, \widehat{s}) + \frac{\sigma_k}{(p+1)!}\|S_k^T\widehat{s}\|^{p+1}, \tag{4}$$

as a function of $\widehat{s} \in \mathbb{R}^\ell$, where the sketched Taylor model $\widehat{T}_{f,p}(x, \widehat{s})$ is given by

$$\widehat{T}_{f,p}(x, \widehat{s}) \stackrel{\rm def}{=} f(x) + \sum_{i=1}^{p} \frac{1}{i!}\nabla_x^i f(x)[S_k^T\widehat{s}]^i.$$

The full dimensional step is then defined by $s = S_k^T\widehat{s}$. We note that $\widehat{T}_{f,p}(x, \widehat{s}) = T_{f,p}(x, s)$ and

$$\widehat{m}_k(\widehat{s}) = m_k(s). \tag{5}$$

A few comments on this algorithm are necessary.

1. It is crucial to observe that, while the definition of the model in (7) involves the function value $f(x_k)$ (in $\widehat{T}_{f,p}(x_k, \widehat{s})$), this function value is never needed in the algorithm (it cancels out in (13)) and therefore must not be evaluated. The algorithm thus belong to the OFFO class. Of course, the minimization of the model may require the evaluation of the sketched derivatives $\{\nabla_{\widehat{s}}^j f(x_k)[S_k\cdot]^j\}_{j=1}^p$, at least along some directions[2]. This makes the use of derivatives of degree higher than two potentially useable in practice, especially if the objective function is partially separable [23, 11].

[2] In the course of a Krylov subproblem solver for $p = 2$, say.

Algorithm 2.1: Sketching OFFO adaptive regularization of degree p (SKOFFAR$_p$)

Step 0: Initialization: An initial point $x_0 \in \mathbb{R}^n$, a regularization parameter $\nu_0 > 0$ and a requested final gradient accuracy $\epsilon \in (0, 1]$ are given, as well as the parameters

$$\theta > 1,\ \mu_{-1} \geq 0 \quad \text{and} \quad 0 < \vartheta < 1.$$

Set $k = 0$.

Step 1: Step calculation: If $k = 0$, set $\sigma_0 = \nu_0$. Otherwise, select

$$\sigma_k \in \Big[\vartheta\nu_k, \max[\nu_k, \mu_k]\Big], \tag{6}$$

where

$$\mu_k = \max\left[\mu_{k-1}, \frac{\|S_{k-1}g_k\| - \|\nabla^1_{\widehat{s}}\widehat{T}_{f,p}(x_{k-1}, \widehat{s}_{k-1})\|}{\kappa_{S,k-1}.\|s_{k-1}\|^p}\right], \tag{7}$$

with some $\kappa_{S,k-1}$ such that $\|S_{k-1}\| \leq \kappa_{S,k-1}$. Draw a random matrix $S_k \in \mathbb{R}^{\ell\times n}$ from $\mathcal{S}$ and compute a step $s_k = S_k^T\widehat{s}_k$ such that $\widehat{s}_k$ sufficiently reduces the model $\widehat{m}_k$ defined in (7) in the sense that

$$\widehat{m}_k(\widehat{s}_k) - \widehat{m}_k(0) < 0 \tag{8}$$

and

$$\|\nabla^1_{\widehat{s}}\widehat{T}_{f,p}(x_k, \widehat{s}_k)\| \leq \theta\frac{\sigma_k}{p!}\|S_k^T\widehat{s}_k\|^{p-1}\|S_kS_k^T\widehat{s}_k\|. \tag{9}$$

Step 2: Updates. Set

$$x_{k+1} = x_k + s_k$$

and

$$\nu_{k+1} = \nu_k + \nu_k\|s_k\|^{p+1}. \tag{10}$$

Increment k by one and go to Step 1.

2. Since
$$\nabla^1_{\widehat{s}}\|S_k^T\widehat{s}\|^{p+1} = (p+1)\|S_k^T\widehat{s}\|^{p-1}\, S_kS_k^T\widehat{s},$$
one verifies that conditions (13) and (14) do hold at an exact minimizer of $\widehat{m}_k$ (the latter with $\theta = 1$). A step satisfying these conditions is therefore guaranteed to exist. Note that (14) is a condition on the norm of the gradient of the Taylor series for f, at variance with [10, 27] where the condition is on the gradient of the regularized model (5).

3. The value of μ_k in the definition (11) of σ_k is chosen to help the regularization parameter σ_k to grow fast enough, given the knowledge at iteration k. We will show in Lemma 3.5 that μ_k is bounded above by $\max[\mu_{-1}, L_p]$ irrespective of the choice of $\kappa_{S,k-1}$. As a consequence, the specific values of $\kappa_{S,k-1}$ play no role in our complexity analysis, albeit they obviously affect the practical performance of the method.

The SKOFFARp algorithm can be seen as a stochastic process. Randomness occurs because the selection of S_k is (possibly) random and, as the algorithm proceeds, from the random realizations[3] of the iterates x_k and the steps s_k. The objective of our forthcoming complexity analysis for this algorithm is to derive a probabilistic bound on the process hitting time
$$N_1(\epsilon) \stackrel{\rm def}{=} \min\{k \in \mathbb{N} \mid \|g_k\| \leq \epsilon\}, \tag{11}$$
where we denote $g_k \stackrel{\rm def}{=} \nabla^1_x f(x_k)$ for all k. $N_1(\epsilon)$ is the number of iterations that a particular realization of the algorithm requires to obtain an ϵ-approximate first-order critical point.

3 Evaluation complexity for the SKOFFARp algorithm

Before discussing our analysis of evaluation complexity, we first restate some classical lemmas for ARp algorithms, starting with Lipschitz error bounds.

Lemma 3.1 Suppose that AS.1 and AS.3 hold. Then
$$f(x_{k+1}) - \widehat{T}_{f,p}(x_k, \widehat{s}_k) = f(x_{k+1}) - T_{f,p}(x_k, s_k) \leq \frac{L_p}{(p+1)!}\|s_k\|^{p+1}, \tag{12}$$
and
$$\|g_{k+1} - \nabla^1_s T_{f,p}(x_k, s_k)\| \leq \frac{L_p}{p!}\|s_k\|^p. \tag{13}$$

Proof. This is a standard result (see [9, Lemma 2.1] for instance). □

We next state a simple lower bound on the Taylor series' decrease.

Lemma 3.2
$$\Delta T_{f,p}(x_k, s_k) \stackrel{\rm def}{=} T_{f,p}(x_k, 0) - T_{f,p}(x_k, s_k) > \frac{\sigma_k}{(p+1)!}\|s_k\|^{p+1}. \tag{14}$$

Proof. The bound directly results from $\widehat{m}_k(\widehat{s}_k) = m_k(s_k)$, (13) and (5). □

[3] Formally, the iterates and steps are random variables on some implicitly defined probability space, and x_k and s_k are their realizations.

This and AS.2 allow us to establish a lower bound on the decrease in the objective function (although it is never computed).

Lemma 3.3 Suppose that AS.1 and AS.3 hold and that $\sigma_k \geq 2L_p$. Then

$$f(x_k) - f(x_{k+1}) > \frac{\sigma_k}{2(p+1)!}\|s_k\|^{p+1}. \tag{15}$$

Proof. From (18) and (20), we obtain that

$$f(x_k) - f(x_{k+1}) > \frac{\sigma_k - L_p}{(p+1)!}\|s_k\|^{p+1}$$

and (21) immediately follows from our assumption on σ_k. □

We now recall an upper bound on $\|s_k\|$ generalizing those proposed in [7, 22] to the case where p is arbitrary.

Lemma 3.4 Suppose that AS.1 and AS.5 hold. At each iteration k, we have that

$$\|s_k\| \leq 2\eta + 2\left(\frac{(p+1)!\|g_k\|}{\sigma_k}\right)^{\frac{1}{p}}, \tag{16}$$

where

$$\eta = \sum_{i=2}^{p}\left[\frac{\kappa_{\rm high}(p+1)!}{i!\,\vartheta\nu_0}\right]^{\frac{1}{p-i+1}}. \tag{17}$$

Proof. See [20, Lemma 3.6]. Note that this result does not involve S_k as it is valid for any step which reduces m_k and, using (9) and (13), $m_k(s_k) = \widehat{m}_k(\widehat{s}_k) < \widehat{m}_k(0) = m_k(0)$. □

Our next step is to show that μ_k is bounded.

Lemma 3.5 For all $k \geq 0$,

$$\mu_k \leq \max[\mu_{-1}, L_p]. \tag{18}$$

Proof. We have that $\nabla^1_{\widehat{s}}\widehat{T}_{f,p}(x_{k-1}, \widehat{s}_{k-1}) = \nabla^1_{\widehat{s}}T_{f,p}(x_{k-1}, S^T_{k-1}s_{k-1}) = S_{k-1}\nabla^1_s T_{f,p}(x_{k-1}, s_{k-1})$, so that, using the triangular inequality, (19) and (14),

$$\begin{aligned}\|S_{k-1}g_k\| &\leq \|S_{k-1}(g_k - \nabla^1_x T_{f,p}(x_{k-1}, s_{k-1}))\| + \|S_{k-1}\nabla^1_x T_{f,p}(x_{k-1}, s_{k-1})\| \\ &\leq \|S_{k-1}\|L_p\|s_{k-1}\|^p + \|\nabla^1_{\widehat{s}}\widehat{T}_{f,p}(x_{k-1}, \widehat{s}_{k-1})\|, \\ &\leq \|S_{k-1}\|L_p\|s_{k-1}\|^p\end{aligned}$$

and thus

$$L_p \geq \frac{\|S_{k-1}g_k\| - \|\nabla^1_{\widehat{s}}\widehat{T}_{f,p}(x_{k-1}, \widehat{s}_{k-1})\|}{\|S_{k-1}\|\|s_{k-1}\|^p}. \tag{19}$$

The inequality (24) then follows from (12) and $\|S_{k-1}\| \leq \kappa_{S,k-1}$. □

The proof of this lemma shows that a tighter lower bound on L_p (see (25)) is also available at the often significant cost of evaluating $\|S_{k-1}\|$, thus motivating the introduction of the (hopefully) cheaper $\kappa_{S,k-1}$.

Since our objective is to minimize f, obtaining a decrease as stated by Lemma 3.3 is important. The condition $\sigma_k \geq 2L_p$ in this lemma and (11) together suggest that the condition

$$\nu_k \geq \frac{2L_p}{\vartheta} \tag{20}$$

is important for our subsequent analysis. Remembering that ν_k is increasing with k, we therefore define

$$k_1 \stackrel{\rm def}{=} \min\left\{k \geq 1 \mid \nu_k \geq \frac{2L_p}{\vartheta}\right\} \tag{21}$$

the index of the first iterate (in a given realization) such that significant objective function decrease is guaranteed by Lemma 3.3. Note that k_1 may fail to exist, which is why we define the random event

$$\mathcal{K}_1 \stackrel{\rm def}{=} \{k_1 \text{ as defined by (27) exists and is finite}\}. \tag{22}$$

We now pursue our analysis under the condition that $\mathcal{K}_1$ occurs. The next series of Lemmas provides bounds[4] on $f(x_{k_1})$ and ν_{k_1}, which in turn allows establishing an upper bound on the regularization parameter, only depending on the problem and the fixed algorithmic parameters.

Lemma 3.6 Suppose that AS.1, AS.3, AS.4 and AS.5 hold and consider a realization of the SKOFFARp algorithm where $\mathcal{K}_1$ occurs. Then

$$\nu_{k_1} \leq \nu_{\max} \stackrel{\rm def}{=} \frac{2L_p}{\vartheta}\left[1 + \left(2\eta + 2\left(\frac{(p+1)!\kappa_{\rm g}}{\vartheta\nu_0}\right)^{\frac{1}{p}}\right)^{p+1}\right], \tag{23}$$

where η is defined in (23) and $\kappa_{\rm g}$ in AS.4.

Proof. Since $\mathcal{K}_1$ is assumed to occur, k_1 is well-defined and finite. Successively using Lemma 3.4 and the update rule for ν_k (16), we derive that

$$\nu_{k_1} \stackrel{(16)}{=} \nu_{k_1-1} + \nu_{k_1-1}\|s_{k_1-1}\|^{p+1} \stackrel{(22)}{\leq} \nu_{k_1-1} + \nu_{k_1-1}\left(2\left((p+1)!\frac{\|g_{k_1-1}\|}{\sigma_{k_1-1}}\right)^{\frac{1}{p}} + 2\eta\right)^{p+1}$$

and the desired result follows by using AS.4, the definition of k_1 in (27) and the inequalities $\sigma_{k_1-1} \geq \vartheta\nu_{k_1-1} \geq \vartheta\nu_0$. □

Lemma 3.6 allows us to establish an upper bound on $f(x_{k_1})$ as a function of $\nu_{\max}$.

Lemma 3.7 Suppose that AS.1, AS.3, AS.4 and AS.5 hold and consider a realization of the SKOFFARp algorithm where $\mathcal{K}_1$ occurs. Then

$$f(x_{k_1}) \leq f_{\max} \stackrel{\rm def}{=} f(x_0) + \frac{1}{(p+1)!}\left(\frac{L_p}{\sigma_0}\nu_{\max} + \vartheta\sigma_0\right). \tag{24}$$

[4]Conditional on $\mathcal{K}_1$.

Proof. Lemma 3.8 in [20] shows that

$$f(x_{k_1}) \leq f(x_0) + \frac{1}{(p+1)!}\left(\frac{L_p \nu_{k_1}}{\sigma_0} + \vartheta\sigma_0)\right).$$

The desired results then follows from Lemma 3.6. □

The two bounds stated in Lemmas 3.7 and 3.6 are useful in that they now imply an upper bound on the regularization parameter, an important step in standard theory for regularization methods.

Lemma 3.8 Suppose that AS.1, AS.3, AS.4 and AS.5 hold and consider a realization of the SKOFFARp algorithm. Then

$$\begin{array}{rcl} \sigma_k & \leq & \sigma_{\max} \\ & \stackrel{\rm def}{=} & \max\left[\frac{2(p+1)!}{\vartheta}\left[f(x_0) - f_{\rm low} + \frac{1}{(p+1)!}\left(\frac{L_p}{\sigma_0}\nu_{\max} + \vartheta\sigma_0\right)\right] + \nu_{\max}, \mu_{-1}, \frac{2L_p}{\vartheta}, L_p, \nu_0\right]. \end{array} \tag{25}$$

Proof. First consider the case where $\mathcal{K}_1$ occurs. From, Lemma [20, Lemma 3.9], whose proof remains valid here given the upper bound on μ_k stated in Lemma 3.5, we deduce that

$$\sigma_k \leq \sigma_{\max} \stackrel{\rm def}{=} \max\left[\frac{2(p+1)!}{\vartheta}\left[f(x_0) - f_{\rm low} + \frac{1}{(p+1)!}\left(\frac{L_p}{\sigma_0}\nu_{\max} + \vartheta\sigma_0\right)\right] + \nu_{\max}, \mu_{-1}, L_p, \nu_0\right].$$

If $\mathcal{K}_1$ does not occur, $\nu_k \leq \frac{2L_p}{\vartheta}$ for all k. Thus we obtain, using (11) and (24), that $\sigma_k \leq \max[\frac{2L_p}{\vartheta}, \mu_{-1}]$ for all k, and (31) also holds. □

The theory of adaptive regularization method crucially depends on the relation between the steplength $\|s_k\|$ and the norm of the gradient at the next iteration $\|g_{k+1}\|$ (see Lemmas 3.3.3 and 4.1.3 in [10], for instance), which is itself bounded below by ϵ before convergence. Here we choose to consider this dependence as a random event, depending on the choice of S_k. This is formalized in the following definition.

Definition 3.9 *Given some $\omega \in (0,1)$ independent of k, iteration $k \in \{0,\ldots,N_1(\epsilon)-2\}$ is said to be ω-true for some $\omega \in (0,1)$ independent of k whenever*

$$\|s_k\|^p \geq \omega\epsilon. \tag{26}$$

We discuss in Section 4 conditions which may enforce this property, but immediately note that it automatically holds if S_k is of rank n [20, Lemma 3.4]. We also define

$$\mathcal{T}_k^{(\omega)} \stackrel{\rm def}{=} \{j \in \{0,\ldots,k-1\} \mid \text{iteration } j \text{ is } \omega\text{-true}\}, \tag{27}$$

the index set $\mathcal{T}_k^{(\omega)}$ of all ω-true iterations in the first k.

Given these definitions, we now need to establish under which condition the event $\mathcal{K}_1$ occurs with high probability. Such a condition is obtained in two stages, the first is inspired by [19, Lemma 7] and [20, Lemma 3.5] and investigates, in our probabilistic setting, the effect of accumulating ω-true iterations.

Lemma 3.10 Suppose that AS.1 and AS.3 hold and consider a particular realization of the SKOFFARp algorithm. Let $k_0 < N_1(\epsilon)$ be an iteration index (in this realization) such that k_* ω-true iterations have been performed among those of index 0 to $k_0 - 1$, where

$$k_* \stackrel{\rm def}{=} \left\lceil \frac{2L_p \epsilon^{-\frac{p+1}{p}}}{\vartheta \nu_0 \, \omega^{\frac{p+1}{p}}} \right\rceil. \tag{28}$$

Then k_1 exists, $k_1 \leq k_0$ and, for all $k \geq k_1$,

$$\sigma_k \geq 2L_p. \tag{29}$$

Proof. First observe that (35) is a direct consequence of (11) if $\nu_k \geq 2L_p/\vartheta$. Suppose now that, for some $k \in \{k_0, \ldots, N_1(\epsilon) - 1\}$, $\nu_k < 2L_p/\vartheta$. Since $\{\nu_k\}$ is a non-decreasing sequence, we deduce that this inequality holds for $j \in \{0, \ldots, k\}$. Successively using the form of the ν_k update rule (16), (32), (11) and the fact that $k < N_1(\epsilon)$, we obtain that

$$\begin{aligned}\nu_k &\stackrel{(16)}{>} \sum_{j=0}^{k-1} \nu_j \|s_j\|^{p+1} \stackrel{(33)}{>} \sum_{j \in \mathcal{T}_k^{(\omega)}} \nu_j \|s_j\|^{p+1} \stackrel{(32)}{\geq} \sum_{j \in \mathcal{T}_k^{(\omega)}} \nu_j (\omega\epsilon)^{\frac{p+1}{p}} \\ &\stackrel{(11)}{\geq} \sum_{j \in \mathcal{T}_k^{(\omega)}} \nu_0 \, (\omega\epsilon)^{\frac{p+1}{p}} \stackrel{(34)}{>} k_* \, \nu_0 (\omega\epsilon)^{\frac{p+1}{p}}.\end{aligned}$$

Substituting the definition of k_* in the last inequality, we obtain that

$$\frac{2L_p}{\vartheta} < \nu_k < \frac{2L_p}{\vartheta},$$

which is impossible. Hence no index $k \in \{k_0, \ldots, N_1(\epsilon) - 1\}$ exists such that $\nu_k < 2L_p/\vartheta$. Thus, $k_1 \leq k_0$ exists by definition of k_1 in (27). By the same definition, we finally deduce that $\nu_k \geq 2L_p/\vartheta$ for all $k \geq k_1$, in turn implying (35) because of (11). □

Observe that (34) depends on the ratio L_p/ν_0 which is the fraction by which ν_0 underestimates the Lipschitz constant. This lemma thus implies that the probability of $\mathcal{K}_1$ is at least the probability that k_* ω-true iterations are performed, which we now investigate under the following assumption.

AS.6 There exists an $\omega \in (0, 1)$ and a $\pi_S^{(1)} > 0$ such that for S_k drawn randomly,

$$\mathbb{P}\Big[\text{iteration } k \text{ is } \omega\text{-true} \mid x_k = \bar{x}_k, \, \sigma_k = \bar{\sigma}_k\Big] \geq \pi_S^{(1)},$$

for any $\bar{x}_k \in \mathbb{R}^n$, any $\bar{\sigma}_k \in [\vartheta\nu_0, \sigma_{\max}]$ and any $k \in \{0, \ldots, N_1(\epsilon) - 2\}$, where $\mathbb{P}[X]$ denotes the probability of the event X. Moreover, the occurrence of k-th iteration being ω-true is conditionally independent of the occurrence of iterations $0, \ldots, k-1$ being ω-true given $x_k = \bar{x}_k$ and $\sigma_k = \bar{\sigma}_k$.

This assumption differs from Assumption 1 in [27, page 71] in that it now it makes the probability of an ω-true iteration conditional not only on x_k but also on σ_k, which we feel is reasonable given the isotropic nature of the regularization term in (5). Note that a suitable value for ω may depend on the bounds on σ_k (as we will see below in Lemmas 4.1, 4.2 and A.4). Before using AS.6 and $\pi_S^{(1)}$ directly, we first recall a known probabilistic result.

Lemma 3.11 For all nonnegative i, let $\mathcal{A}_i$ be an event which can be true or false and is conditionally independent of $\mathcal{A}_0, \mathcal{A}_1, \ldots \mathcal{A}_{i-1}$. For any x_i $\mathbb{R}^n$, $\sigma_i \in [\vartheta\nu_0, \sigma_{\max}]$, suppose that the probability of $\mathcal{A}_i$ being true is at least $\pi \in (0,1)$ for all i. For $k \geq 0$, let $\mathcal{W}_k = \{i \in \{0, \ldots, k-1\} \mid \mathcal{A}_i \text{ is true}\}$. Then, for any given $\delta_1 \in (0,1)$,

$$\mathbb{P}\Big[|\mathcal{W}_k| > (1-\delta_1)\pi k\Big] \geq 1 - e^{-\frac{\delta_1^2}{2}\pi k}. \tag{30}$$

Proof. See [27, Lemma 4.3.1] where, as mentioned above, we now consider the "state" of the algorithm at iteration i to comprise both x_i and σ_i. □

We are now in position to use this result to obtain a lower bound on the probability that k_* ω-true iterations are performed, and that k_1 is well-defined.

Lemma 3.12 Suppose that AS.1, AS.3 and AS.6 hold and let $\delta_1 \in (0,1)$ be given. Let

$$k_\diamond \stackrel{\text{def}}{=} \left\lceil \frac{k_*}{(1-\delta_1)\pi_S^{(1)}} \right\rceil, \tag{31}$$

where k_* is given by (34). Then

$$\mathbb{P}\Big[\mathcal{K}_1 \mid N_1(\epsilon) > k_\diamond\Big] \geq 1 - e^{-\frac{\delta_1^2}{2}\pi_S^{(1)} k_\diamond} \stackrel{\text{def}}{=} \pi_1^{(1)}. \tag{32}$$

Proof. Identifying $A_i = \{\text{iteration } i \text{ is } \omega\text{-true}\}$, Lemma 3.11 with $\pi = \pi_S^{(1)}$ and $k_0 = k_\diamond$ gives that the probability that at least k_* ω-true iterations have been performed during iterations 0 to $k_\diamond - 1$ is at least $\pi_1^{(1)}$. The desired conclusion then follows from Lemma 3.10. □

We finally propose a variant of the well-known "telescoping sum" argument adapted to our probabilistic setting to derive the desired evaluation complexity bound.

Theorem 3.13 Suppose that AS.1, AS.2, AS.3, AS.4, AS.5 and AS.6 hold, that $\delta_1 \in (0,1)$ is given and that the SKOFFARp algorithm is applied to problem (1). Define

$$\kappa_{\text{SKOFFARp}} \stackrel{\text{def}}{=} \frac{4\,[L_p + (p+1)!(f_{\max} - f_{\text{low}})]}{\vartheta\nu_0\omega^{\frac{p+1}{p}}(1-\delta_1)\pi_S^{(1)}} \tag{33}$$

where $f_{\max}$ is defined in (30). Then

$$\mathbb{P}\Big[N_1(\epsilon) \leq \kappa_{\text{SKOFFARp}}\,\epsilon^{-\frac{p+1}{p}} + 4\Big] \geq \left(1 - e^{-\frac{\delta_1^2}{2}\pi_S^{(1)} k_\diamond}\right)^2 \tag{34}$$

where $k_\diamond$ is defined by (37).

Proof. First note that (34) and (37) imply that

$$k_\diamond \leq \frac{1}{(1-\delta_1)\pi_S^{(1)}} \left(\frac{2L_p}{\vartheta\nu_0\omega^{\frac{p+1}{p}}}\right) \epsilon^{-\frac{p+1}{p}} + 1. \tag{35}$$

Thus, given (39),

$$\mathbb{P}\Big[N_1(\epsilon) \leq \kappa_{\text{SKOFFARp}}\,\epsilon^{-\frac{p+1}{p}} + 4 \;\mid\; N_1(\epsilon) \leq 2k_\diamond + 2\Big] = 1. \tag{36}$$

Suppose now that $N_1(\epsilon) > k_\diamond + 2 > k_\diamond$ and that $\mathcal{K}_1$ occurs. Consider an iteration $j > k_\diamond \geq k_1$ (note that k_1 is well defined) such that $j+1 < N_1(\epsilon)$ and suppose furthermore that iteration j is ω-true, a situation which occurs with probability at least $\pi_S^{(1)}$ because of AS.6. From the fact that $\mathcal{K}_1$ occurs, $N_1(\epsilon) > k_\diamond$ and the definition of k_1 in (27), we have that $\sigma_j \geq 2L_p$ and we may apply Lemma 3.3, yielding (21) for iteration j. Since this iteration is also ω-true, (21) and inequality (32) also hold for iteration j. Moreover, the fact $\mathcal{K}_1$ occurs ensures (because of Lemma 3.8, (11), the non-decreasing nature of ν_k and the identity $\sigma_0 = \nu_0$) that $\sigma_j \in [\vartheta\sigma_0, \sigma_{\max}]$. Finally, $\|g_{j+1}\| \geq \epsilon$ because $j+1 < N_1(\epsilon)$. Combining these observations, we obtain that

$$f(x_j) - f(x_{j+1}) \geq \frac{\sigma_j\|s_j\|^{p+1}}{2(p+1)!} \geq \frac{\sigma_j\omega^{\frac{p+1}{p}}\|g_{j+1}\|^{\frac{p+1}{p}}}{2(p+1)!} \geq \frac{\vartheta\nu_0\omega^{\frac{p+1}{p}}\epsilon^{\frac{p+1}{p}}}{2(p+1)!} \tag{37}$$

with probability (conditional to $\mathcal{K}_1$ and $N_1(\epsilon) > k_\diamond+2$) at least $\pi_S^{(1)}$. Applying now Lemma 3.11 to iterations of index $k_\diamond + 1$ to j with

$$\mathcal{A}_{i-k_\diamond} = \{\ (43) \text{ holds at iteration } i-k_\diamond\ \},\quad \pi = \pi_S^{(1)} \ \text{ and } \ k = j - k_\diamond,$$

we deduce that, for all $j \in \{k_\diamond + 1, \ldots, N_1(\epsilon) - 2\}$,

$$\mathbb{P}\left[|\mathcal{V}_j| \geq (j-k_\diamond)(1-\delta_1)\pi_S^{(1)} \,|\, \mathcal{K}_1 \ \text{ and } \ N_1(\epsilon) > k_\diamond\right] \geq 1 - e^{-\frac{\delta_1^2}{2}\pi_S^{(1)}(j-k_\diamond)}$$

where $\mathcal{V}_j \stackrel{\text{def}}{=} \{i \in \{k_\diamond+1, \ldots, j\} \mid (43) \text{ holds at iteration } i\}$. In particular, we have that

$$\mathbb{P}\left[|\mathcal{V}_j| \geq (j-k_\diamond)(1-\delta_1)\pi_S^{(1)} \,|\, \mathcal{K}_1 \ \text{ and } \ N_1(\epsilon) > 2k_\diamond + 2\right] \geq \pi_1^{(1)}, \tag{38}$$

with $\pi_1^{(1)}$ defined in (38), for all $j \in \{2k_\diamond + 1, \ldots, N_1(\epsilon) - 2\}$. We also know from Lemma 3.3 and the definition of k_1 in (27) that the sequence $\{f(x_j)\}$ is non-increasing for $j \geq k_1$, and thus that

$$f(x_{k_1}) - f(x_{j+1}) = \sum_{i=k_1}^{j}[f(x_i) - f(x_{i+1})] \geq \sum_{i=k_\diamond+1}^{j}[f(x_i) - f(x_{i+1})] \geq |\mathcal{V}_j| \min_{i\in\mathcal{V}_j}[f(x_i) - f(x_{i+1})].$$

Combining this inequality with (43) and (44) then yields that

$$\mathbb{P}\left[f(x_{k_1}) - f(x_{j+1}) \geq (j-k_\diamond)(1-\delta_1)\pi_S^{(1)}\,\kappa_2^{-1}\epsilon^{\frac{p+1}{p}} \;\mid\; \mathcal{K}_1 \ \text{ and } \ N_1(\epsilon) > 2k_\diamond + 2\right] \geq \pi_1^{(1)}$$

where

$$\kappa_2 = \frac{2(p+1)!}{\vartheta\nu_0\omega^{\frac{p+1}{p}}}, \tag{39}$$

and thus, because of AS.3, that

$$\mathbb{P}\left[f(x_{k_1}) - f_{\text{low}} \geq \kappa_2^{-1}(1-\delta_1)\pi_S^{(1)}(j-k_\diamond)\,\epsilon^{\frac{p+1}{p}} \;\mid\; \mathcal{K}_1 \ \text{ and } \ N_1(\epsilon) > 2k_\diamond + 2\right] \geq \pi_1^{(1)}.$$

Furthermore, (30) in Lemma 3.7 then implies that

$$\mathbb{P}\left[j - k_\diamond \leq \frac{\kappa_2}{(1-\delta_1)\pi_S^{(1)}}(f_{\max} - f_{\text{low}})\,\epsilon^{-\frac{p+1}{p}} \;\mid\; \mathcal{K}_1 \ \text{ and } \ N_1(\epsilon) > 2k_\diamond + 2\right] \geq \pi_1^{(1)},$$

Since j is arbitrary between $2k_\diamond + 1$ and $N_1(\epsilon) - 2$, we obtain that

$$\mathbb{P}\left[N_1(\epsilon) \leq \frac{\kappa_2}{(1-\delta_1)\pi_S^{(1)}}(f_{\max} - f_{\rm low})\,\epsilon^{-\frac{p+1}{p}} + k_\diamond + 2 \;\mid\; \mathcal{K}_1 \;\text{ and }\; N_1(\epsilon) > 2k_\diamond + 2\right] \geq \pi_1^{(1)},$$

which, given the definitions of κ_2 in (45), of $\kappa_{\rm SKOFFARp}$ in (39) and inequality (41), yields that

$$\mathbb{P}\left[N_1(\epsilon) \leq \kappa_{\rm SKOFFARp}\,\epsilon^{-\frac{p+1}{p}} + 4 \;\mid\; \mathcal{K}_1 \;\text{ and }\; N_1(\epsilon) > 2k_\diamond + 2\right] \geq \pi_1^{(1)}.$$

Therefore, from (42), the fact that

$$\mathbb{P}\Big[\mathcal{K}_1 \mid N_1(\epsilon) > 2k_\diamond + 2\Big] \geq \mathbb{P}\Big[\mathcal{K}_1 \mid N_1(\epsilon) > k_\diamond\Big]$$

and Lemma 3.12, we finally obtain that

$$\begin{aligned}
&\mathbb{P}\left[N_1(\epsilon) \leq \kappa_{\rm SKOFFARp}\,\epsilon^{-\frac{p+1}{p}} + 4\right] \\
&\quad= \mathbb{P}\left[N_1(\epsilon) \leq \kappa_{\rm SKOFFARp}\,\epsilon^{-\frac{p+1}{p}} + 4 \;\mid\; \mathcal{K}_1 \;\text{ and }\; N_1(\epsilon) > 2k_\diamond + 2\right] \\
&\qquad\quad \times \mathbb{P}\Big[\mathcal{K}_1 \mid N_1(\epsilon) > 2k_\diamond + 2\Big] \times \mathbb{P}\Big[N_1(\epsilon) > 2k_\diamond + 2\Big] + 1 \times \mathbb{P}\Big[N_1(\epsilon) \leq 2k_\diamond + 2\Big] \\
&\quad\geq \mathbb{P}\left[N_1(\epsilon) \leq \kappa_{\rm SKOFFARp}\,\epsilon^{-\frac{p+1}{p}} + 4 \;\mid\; \mathcal{K}_1 \;\text{ and }\; N_1(\epsilon) > 2k_\diamond + 2\right] \\
&\qquad\quad \times \mathbb{P}\Big[\mathcal{K}_1 \mid N_1(\epsilon) > k_\diamond\Big] \\
&\quad\geq (\pi_1^{(1)})^2.
\end{aligned}$$

Substituting the values of $\pi_1^{(1)}$ given by (38) in this inequality then yields (40). □

We now comment on this result.

1. As in the methods of [27] and [6], it is not necessary to evaluate the full-space derivatives $\{\nabla_x^j f(x_k)\}_{j=1}^p$ because only their sketched versions $\{\nabla_x^j f(x_k)[S_k\cdot]^j\}_{j=1}^p$ are used. As a consequence, the cost of evaluating the derivatives (not to mention that of computing the step) is potentially reduced by a typically significant factor ℓ/n. We discuss below whether this advantage may be offset by the choice of ω in AS.6.

2. Because it is proved in [20, Theorem 3.12] that the $\mathcal{O}(\epsilon^{-(p+1)/p})$ order bound for finding ϵ-approximate critical points is sharp for the OFFARp algorithm, the same is also true for Theorem 3.13 above, because SKOFFARp subsumes[5] OFFARp if $S_k = I$ for all k.

3. Considered as a worst-case evaluation complexity bound for $p = 2$, the order bound $\mathcal{O}(\epsilon^{-3/2})$ is known to be optimal for a large class of methods using first- and second-derivatives [8], justifying the title of this paper.

4. Note that (34) and (37) not only imply (41), but also that $k_\diamond$ is at least a (significant) fraction of $\epsilon^{-(p+1)/p}$, which, for meaningul values of ϵ, is a reasonably large number. Moreover, $(k_\epsilon - k_\diamond)$ is expected to be at least of the same order. Thus the factor

$$\left(1 - e^{-\frac{\delta_1^2}{2}\pi_S^{(1)}k_\diamond}\right)$$

in the right-hand side of (40) is expected to be very close to 1.

[5]The different conditions on the regularization parameter σ_k only result in differences in the constants.

5. The parameter δ_1, which we are still free to choose in (0,1) occurs in (39) and in the exponentials of (40). A quick calculation indicates that choosing δ_1 close to 1 can improves the bound on the right-hand side of (40) (although marginally because of our previous comment) while its possibly detrimental effect on (39) occurs because of the factor $1/(1-\delta_1)$ which must be kept bounded. Given the magnitude of the other factors in these formulae, values such as $\delta_1 = \frac{1}{2}$ or $\delta_1 = \frac{1}{10}$ could be considered acceptable.

6. As can be expected, the conditions for a random embedding given by (48) and AS.6 have a significant impact on the result, which significantly degrades if ω and/or $\pi_S^{(1)}$ tends to zero.

7. The facts that the objective function is not evaluated by the SKOFFARp algorithm and that the trial point $x_k + s_k$ is always accepted as the next iterate have for consequence that no distinction is necessary in the stochastic analysis between "successful" iterations (where the step is accepted because the objective function has decreased enough) and "unsuccessful" ones. This distinction had however to be taken into account in the analysis of [27] for more standard trust-region and adaptive regularisation methods using functions values, leading to several different types of iterations whose numbers have to be be bounded.

4 Selecting random subspaces

We now turn to ways in which ω-true iterations can be shown to happen with suitable probability $\pi_S^{(1)}$, thereby satisfying AS.6. A natural approach is to rely on Johnson-Lindenstrauss embeddings. Restricting ourselves to the case where $p \in 1, 2$ and inspired by [27, Definition 5.3.1] (see also [31], for instance), we say that, for some given "preservation parameter" $\alpha_S \in (0, 1)$, iteration k is α_S-true whenever

$$\|S_k\| \leq S_{\max}, \tag{40}$$

for some positive scalar $S_{\max}$ independent of k, and for

$$M_k \stackrel{\rm def}{=} [g_k, H_k] \in \mathbb{R}^{n\times n+1}, \tag{41}$$

we have that

$$\|S_k M_k z\| \geq \alpha_S \|M_k z\| \quad \text{for all} \quad z \in \mathbb{R}^{n+1}, \tag{42}$$

where $H_k = \nabla_s^2 f(x_k)$ if $p = 2$ and $H_k = 0_{n\times n}$ if $p = 1$. This condition is said to define a *one-sided random embedding* of the second-order Taylor's series.

Given such a one-sided random embedding, we now adapt an argument of [22] and verify that (32) holds at α_S-true iterations.

Lemma 4.1 Suppose that $p \in \{1, 2\}$, that AS.1, AS.3, AS.4 and AS.5 hold and that iteration $k \geq 0$ of the SKOFFARp algorithm is α_S-true (in the sense of (48)). Then

$$\|s_k\|^p \geq \frac{p!\,\alpha_S}{\alpha_S L_p + \theta S_{\max}\sigma_{\max}} \|g_{k+1}\|. \tag{43}$$

Thus iteration $k \in \{0, \ldots, N_1(\epsilon)-2\}$ is ω-true (in the sense of (32)) with $\omega = \frac{p!\,\alpha_S}{\alpha_S L_p + \theta S_{\max}} \sigma_{\max}$.

Proof. First note that applying the chain rule gives that

$$\nabla_{\widehat{s}}^1 \widehat{T}_{f,p}(x_k, \widehat{s}_k) = S_k \nabla_s^1 T_{f,p}(x_k, s_k) = S_k(g_k + H_k s_k) = S_k M_k (1, s_k^T)^T$$

and, since the iteration k is α_S-true, (48) gives that

$$\|\nabla_{\widehat{s}}^1 \widehat{T}_{f,p}(x_k, \widehat{s}_k)\| \geq \alpha_S \|M_k (1, s_k^T)^T\| = \alpha_S \|\nabla_s^1 T_{f,p}(x_k, s_k)\|.$$

Condition (14), the definition $s_k = S_k^T \widehat{s}_k$ and (46) then yield that

$$\|\nabla_s^1 T_{f,p}(x_k, s_k)\| \leq \frac{\|\nabla_{\widehat{s}}^1 \widehat{T}_{f,p}(x_k, \widehat{s}_k)\|}{\alpha_S} \leq \frac{\theta \frac{\sigma_k}{p!} S_{\max} \|S_k^T \widehat{s}_k\|^p}{\alpha_S} \leq \frac{\theta S_{\max} \sigma_k}{p!\, \alpha_S} \|s_k\|^p. \tag{44}$$

Successively using the triangle inequality, condition (50) and (19) (for $p \in \{1, 2\}$), we deduce that

$$\|g_{k+1}\| \leq \|g_{k+1} - \nabla_s^1 T_{f,p}(x_k, s_k)\| + \|\nabla_s^1 T_{f,p}(x_k, s_k)\| \leq \frac{1}{p!} L_p \|s_k\|^p + \frac{\theta S_{\max} \sigma_k}{p!\, \alpha_S} \|s_k\|^p.$$

The inequality (32) follows by rearranging the terms and using the bound (31) in Lemma 3.8. That iteration k is ω-true for $k \in \{0, \ldots, N_1(\epsilon) - 2\}$ follows from the fact that, by definition, $\|g_{k+1}\| \geq \epsilon$ for these values of k. □

While this lemma essentially recovers the result of [27, Lemma 5.3.2], its proof is considerably simpler. Note that (49) is significantly stronger than (32), suggesting that (48) might itself be stronger than necessary. Also observe that we could replace condition (14) by the more permissive

$$\|\nabla_{\widehat{s}}^1 \widehat{T}_{f,p}(x_k, \widehat{s}_k)\| \leq \theta \frac{\sigma_k}{p!} \|S_k\| \, \|S_k^T \widehat{s}_k\|^p$$

or

$$\|\nabla_{\widehat{s}}^1 \widehat{T}_{f,p}(x_k, \widehat{s}_k)\| \leq \theta \frac{\sigma_k}{p!} \kappa_{S,k} \, \|S_k^T \widehat{s}_k\|^p$$

without altering the above theory, but at the price of computing $\|S_k\|$ or estimating a uniform bound on $\kappa_{S,k}$ (such as $S_{\max}$).

As it turns out, it is also possible to generalize Shao's approach to "sparse Hessians" (for $p = 2$) as follows. For some constant (α_S, γ_S) such that $\alpha_S \in (0, 1)$ and $\gamma_S \in [0, 2\alpha_S)$, we now (re)define iteration k to be (α_S, γ_S)-true whenever

$$\|S_k\| \leq S_{\max}, \quad \|S_k g_k\| \geq \alpha_S \|g_k\| \quad \text{and} \quad \|S_k H_k\| \leq \sqrt{\gamma_S \|g_{k+1}\|} \tag{45}$$

We then obtain the following result, inspired by [27, Lemma 5.4.1].

Lemma 4.2 Suppose that AS.1 and AS.3 hold and that, for a particular realization, iteration $k \geq 0$ of the SKOFFAR2 algorithm is (α_S, γ_S)-true (in the sense of (53)). Then (32) holds and iteration k is ω-true.

Proof. Let $a = \|S_k H_k\|$. Then (14) gives that

$$\alpha_S \|g_k\| \leq \|S_k g_k\| \leq \|S_k(g_k + H_k s_k)\| + \|S_k H_k s_k\| \leq \tfrac{1}{2} \theta S_{\max} \sigma_k \|s_k\|^2 + a\|s_k\|,$$

and therefore, using the triangle inequality, (19) and the fact that iteration k is (α_S, γ_S)-true,

$$\alpha_S \|g_{k+1}\| \leq \alpha_S \|g_{k+1} - g_k\| + \alpha_S \|g_k\| \leq \tfrac{1}{2} \alpha_S L_2 \|s_k\|^2 + \tfrac{1}{2} \theta S_{\max} \sigma_k \|s_k\|^2 + a\|s_k\|.$$

Defining $b = \alpha_S L_2 + \theta S_{\max} \sigma_k$, we obtain that

$$\|s_k\|^2 + \left(\frac{2a}{b}\right) \|s_k\| - \frac{2\alpha_S \|g_{k+1}\|}{b} \geq 0,$$

yielding that

$$\left(\|s_k\| + \frac{a}{b}\right)^2 \geq \frac{2\alpha_S \|g_{k+1}\|}{b} + \left(\frac{a}{b}\right)^2$$

and thus that

$$\|s\| \geq \sqrt{\frac{2\alpha_S\|g_{k+1}\|}{b} + \left(\frac{a}{b}\right)^2} - \frac{a}{b}.$$

Assuming, without loss of generality, that $b = \alpha_S L_2 + \theta S_{\max}\sigma_k \geq 1$, we deduce that

$$\|s\| \geq \frac{1}{b}\left[\sqrt{2\alpha_S\|g_{k+1}\| + a^2} - a\right]$$

Since the function $\sqrt{c+t^2} - t$ (for $c > 0$) is decreasing as a function of $t \geq 0$ and since $a = \|S_k H_k\| \leq \sqrt{\gamma_S\|g_{k+1}\|}$ because iteration k is (α_S, γ_S)-true, we deduce that

$$\begin{aligned}\|s\| &\geq \frac{1}{b}\left[\sqrt{2\alpha_S\|g_{k+1}\| + \gamma_S\|g_{k+1}\|} - \sqrt{\gamma_S\|g_{k+1}\|}\right] \\ &\geq \frac{\sqrt{2\alpha_S} - \sqrt{\gamma_S}}{\alpha_S L_2 + \theta S_{\max}\sigma_k}\sqrt{\|g_{k+1}\|} \\ &\geq \frac{\sqrt{2\alpha_S} - \sqrt{\gamma_S}}{\alpha_S L_2 + \theta S_{\max}}\sigma_{\max}\sqrt{\|g_{k+1}\|},\end{aligned}$$

where we again used Lemma 3.8 to obtain the last inequality. □

Thus an (α_S, γ_S)-true iteration (in the sense of (53)) is ω-true (in the sense of (32)) for $\omega = (\sqrt{2\alpha_S} - \sqrt{\gamma_S})/(\alpha_S L_2 + \theta S_{\max}\sigma_{\max})$. Also notice that, should we replace (53) by

$$\|S_k\| \leq S_{\max}, \quad \|S_k g_k\| \geq \alpha_S\|g_k\| \quad \text{and} \quad \|S_k H_k\| \leq \sqrt{\gamma_S\epsilon} \quad \text{for} \quad k < N_1(\epsilon) - 1, \tag{46}$$

then definition of an (α_S, γ_S)-true iteration is closer to that of [27], obviously ensuring (53) with a right-hand side of its third part now independent of S_k.

The reader may now recall that AS.6 states that (32), (48), (53) or (54) (or the first part of (53) or (54)) should hold at iteration k with positive probability $\pi_S^{(1)}$. It is argued in [27, Lemma 5.3.1] or [31, Theorem 2.3] (see also [29, Lemma 3.1]) that choosing $\mathcal{S}$ to be the distribution of $\ell \times n$ scaled Gaussian matrices, (48) holds with probability

$$\pi_S^{(1)} = 1 - e^{-\frac{\ell(1-\alpha_S)}{C_\ell} + \mathrm{rank}(M_k)}, \tag{47}$$

where $C_\ell > \frac{1}{4}$ is an absolute constant.

Unfortunately, the expression (55) requires that

$$\mathrm{rank}(M_k) < \frac{\ell(1-\alpha_S)}{C_\ell},$$

thereby significantly limiting the applicability of the result for $p > 1$. Satisfying the third part of (54) with positive probability is possible when H_k is very sparse, also imposing a significant restriction. Other choices for the distribution exist, such as hashing, scaled hashing, sampling matrices or "fast Lindenstrauss transforms" (see [27, Chapter 2] or [31, page 16]). While possibly more economical in term of algebraic operations, they appear to suffer from the same geometric problem: their number of rows ℓ should be of the order of the Hessian's rank, which is problematic for the general case the the Hessian is full-rank. However, note that $\mathrm{rank}(M_k) = 1$ when $p = 1$, essentially avoiding this problem, making the algorithm applicable to a much larger class of problems (and also motivating our emphasis on the first-order variant of the algorithm).

Should one be ready to trade the optimal complexity for getting rid of the low-rank requirement, an algorithm using quadratically regularized quadratic models with inexact Hessians can also be defined and analyzed (see Appendix). Under suitably modified assumptions, the evaluation complexity of this algorithm can be shown to be of order $\mathcal{O}(\epsilon^{-2})$, matching the theoretical results of [6] for a random subspace version of the adaptive regularization algorithm *using function values*. Unfortunately, our numerical experience also matches the cautious conclusions of this reference, which is why we do not investigate it further.

Finally note that the constant (39) involves $S_{\max}^{\frac{p+1}{p}}$ because of its dependence on $\omega^{\frac{p+1}{p}}$. In the case of scaled Gaussian matrices, we know that

$$S_{\max} \leq \beta \stackrel{\rm def}{=} 1.5 + \sqrt{n/\ell} \tag{48}$$

with high probability for the considered values of δ_1 (see [27, Lemma 4.4.4] for instance), yielding a dependence of the constant (39) in $(n/\ell)^{\frac{p+1}{2p}}$. For $p = 1$, this offsets (at least complexity-wise) the benefit of cheaper evaluations of the gradient by the factor ℓ/n, while the advantage of cheaper derivatives is increasingly maintained when p grows (and the method is applicable).

5 Numerical illustration

Because of of its wider applicability and taking the current widespread interest in first-order methods into account, we now provide some numerical tests which illustrate the behaviour of SKOFFAR1, that is the first-order version of SKOFFARp. As is to be expected with stochastic methods of this type, its performance does vary considerably from problem to problem. In general, one expects random projections to work better for problems with more isotropic geometry. For instance if an objective function's geometry features twisting narrow valleys, finding a good direction of descent may have rather low probability, hampering progress of the minimization. Thus, while algorithms like SKOFFARp can bring significant improvements in terms of the total number of full gradients evaluated in some cases, they can also be very slow (albeit convergent, as the above theory and our computational experience shows), as we illustrate below.

Because of this diversity in behaviour across problems, it is the authors' opinion that aggregate performance measures such as performance profiles or other statistics averaged on problem's sets are less informative than a discussion of specific cases. We therefore report results obtained for a few problems from OPM[6] [21], a Matlab incarnation of a subset of the CUTEst test problems [18] using a Matlab implementation of a modified version of the algorithm where we defined $\mathcal{S}$ to be the distribution of $\ell \times n$ scaled Gaussian matrices. The first change is identical to that described in [20] for the OFFARp algorithm, in that (11) is replaced by

$$\sigma_k = \max[\vartheta\nu_k, \xi_k\mu_k]$$

where $\xi_k \in (0, 1)$ is an adaptive scaling parameter (see [20] for details) and where μ_k is defined by (12) with $\mu_{-1} = \max[\|g_0\|, 10^3]$. The second change avoids the (potentially very) costly computation of $\|S_k\|$ by using $\kappa_{S,k} = \beta$ as given by (56). This change was made after running the more expensive code using $\kappa_{S,k} = \|S_k\|$ as suggested by (25) on a few problems and observing that the results obtained with the theoretically weaker $\kappa_{S,k} = \beta$ did not decrease the code efficiency, if at all. We also chose $\vartheta = 10^{-3}$, minimized the regularized linear model exactly (i.e. $\theta = 1$) and terminated the optimization as soon as $\|g_k\| \leq 10^{-3}$. The maximum number of iterations was set to $10^6/\tau$ where $\tau = \ell/n$ and the time limit to two hours per run. All computations were performed on a Dell Precision laptop computer running Ubuntu.

Table 1 reports the total number of full gradient evaluations to reach convergence for decreasing fractions τ from 1 to 0.05, averaged, for SKOFFAR1 with $\tau < 1$, over 10 independent runs. By convention, the evaluation cost of an iteration of SKOFFAR1 is equivalent to τ evaluations of the full gradient. For comparison, we also report (in column ADAG-N) results obtained using the well-known objective-function-free ADAGRAD-Norm algorithm [15, 30], a standard in deep-learning applications. A > sign indicates that at least five but not all ten runs converged, the average being taken on the convergent ones. The string 'time' indicates that five or more runs needed more than the maximum time and 'maxit' indicates that five or more runs needed more that the maximum number of iterations.

The results of the table show a rather contrasted problem-dependent picture. Let us consider first cases where the use of random subspaces pays off, sometimes very significantly. This happens,

[6]The components of the standard starting point for problem `morebv` were multiplied by 25 in order to avoid termination at x_0.

			SKOFFAR1					
Problem	n	ADAG-N	$\tau = 1.00$	0.75	0.50	0.25	0.10	0.05
`arglina`	200	5805	1070	1036	950	810	652	552
`arwhead`	200	803	238	201	197	307	623	778
`broyden3d`	1000	467	118	97	136	158	186	270
`dixmaana`	510	21409	16488	9839	6278	5260	4417	3524
`engval1`	500	58252	27430	22740	28835	20182	16657	14063
`lminsurf`	100	4989	4382	3287	49867	143873	227488	time
`morebv`	100	778	2744	45801	46685	47086	47332	47402
`msqrtals`	100	28347	59754	201517	202978	203110	203273	time
`nzf1`	130	41876	maxit	31170	>39119	58245	16500	13137
`rosenbr`	100	796779	584531	404990	405572	>648586	time	time
`sensors`	200	979	324	303	266	305	492	424
`tridia`	1000	2979	1007	580	543	437	838	996

Table 1: Using ADAGRAD-Norm and the SKOFFAR1 algorithms: number of equivalent full gradient evaluations for varying ratio $\tau = \ell/n$

for at least some values of τ, for problems `arglina`, `arwhead`, `broyden3d`, `dixmaana`, `engval1`, `rosenbr`, `sensors` and `tridia`. However, the performance as a function of τ is not uniform: while it is monotonically increasing with decreasing values of τ for `arglina`, `dixmaana` and `engval1`, it appears to be best for $\tau = 0.75$ or 0.5 in the other cases. There seem to be no correlation between this phenomenon and the dimension of the problem, but we note that the better monotonic behaviour does occur on the better conditioned problems. A typical example of the non-monotonic behaviour is `rosenbr`, a problem known for its narrow curved valleys, where one indeed expects that finding reasonable descent direction in a relative low dimensional random subspace to be difficult.

SKOFFAR1 does not performs well for problems `lminsurf`, `morebv` and `msqrtals`. The first of these is a discretized minimum surface problem with a significantly positive boundary condition and an identically zero starting point within the domain. At initial iterations, the gradients thus belong to a very low dimensional subspace of $\mathbb{R}^n$, a structure which is lost by random sampling. As a consequence, progress along the boundaries is slow (at least for $\tau < 0.75$) while meaningless random changes are made inside the domain, slowing convergence considerably. What makes the algorithm slow on `msqrtals` is less clear, but we suspect the nonconvex multilinear nature of its objective function and its bad conditioning to be part of the difficulty. Fast convergence is observed in the early iterations for the `morebv` problem, but the algorithm then struggles to achieve the reduction of the gradient from 10^{-2} to 10^{-3}.

Finally observe that SKOFFAR1 outperforms ADAGRAD-Norm in all the considered problems except for `msqrtals` and `morebv`.

Summarizing, this limited set of experiments shows that general conclusions remain elusive, but that the potential value of using SKOFFAR1 is better considered problem by problem.

6 Conclusions and perspectives

We have introduced an OFFO adaptive regularization algorithm for nonconvex unconstrained optimization that uses random subspaces, and have shown that its evaluation complexity is, in order, identical to that of the "optimal" adaptive full-space regularization methods using function values. The analysis covers finding approximate first-order critical points, but it is possible to extend the algorithm to ensure second-order criticality (along the lines of the MOFFAR algorithm in [20]), albeit at the price of a very strong assumption on the recovery of the Hessian's minimum eigenvalue in random subspaces, a notoriously thorny problem (see [4, Section 4.2.3], for instance). Our analysis also allows the use of models of arbitrary degree, but this generality may be of limited

practical use since using degree higher than one appears to be mostly applicable to problems with low-rank or very sparse Hessians (or higher derivatives).

Our theoretical and numerical results show that the approach is theoretically sound and that it can be (sometimes significantly) advantageous, but also that its practical performance and usefulness should, as can be expected, be appraised on a problem-by-problem basis.

Acknowledgement

Ph. Toint acknowledges the partial support of INDAM (through the GNCS grant CUP E53C22001930001) during his visit to the Department of the Industrial Engineering (DIEF) at the University of Florence in the fall 2023. He is also grateful for the continued and friendly support of DIEF and ANITI (Toulouse).

References

[1] S Bellavia, G Gurioli, B Morini and Ph. L.Toint. *Adaptive regularization for nonconvex optimization using inexact function values and randomly perturbed derivatives.* Journal of Complexity 68, 101591, 2022.

[2] S Bellavia, N Krejić, B Morini and S Rebegoldi. *A stochastic first-order trust-region method with inexact restoration for finite-sum minimization.* Computational Optimization and Applications, 84(1);53–84, 2023.

[3] R. Bollapragada and S. M. Wild. *Adaptive sampling quasi-Newton methods for zeroth-order stochastic optimization.* Mathematical Programming Computation, 15(2):327–364, 2023.

[4] E. Boman. *Infeasibility and Negative-Curvature Directions in Optimization.* Ph.D Thesis, Stanford University, California, USA, 1999.

[5] A. S. Berahas, L. Cao, and K. Scheinberg. *Global convergence rate analysis of a generic line search algorithm with noise.* SIAM Journal on Optimization 31(2):1489–1518, 2021

[6] C. Cartis, J. Fowkes, and Z. Shao. *Randomised subspace methods for non-convex optimization, with applications to nonlinear least-squares*, arXiv:2211.09873, 2022.

[7] C. Cartis, N. I. M. Gould, and Ph. L. Toint. *Adaptive cubic overestimation methods for unconstrained optimization. Part I: motivation, convergence and numerical results.* Mathematical Programming, Series A, 127(2):245–295, 2011.

[8] C. Cartis, N. I. M. Gould, and Ph. L. Toint. *Worst-case evaluation complexity and optimality of second-order methods for nonconvex smooth optimization.* In B. Sirakov, P. de Souza, and M. Viana, editors, Invited Lectures, Proceedings of the 2018 International Conference of Mathematicians (ICM 2018), vol. 4, Rio de Janeiro, pages 3729–3768. World Scientific Publishing Co Pte Ltd, 2018.

[9] 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.

[10] C. Cartis, N. I. M. Gould, and Ph. L. Toint. *Evaluation complexity of algorithms for nonconvex optimization.* Number 30 in MOS-SIAM Series on Optimization. SIAM, Philadelphia, USA, June 2022.

[11] A. R. Conn and N. I. M. Gould and Ph. L. Toint. LANCELOT*: a Fortran package for large-scale nonlinear optimization (Release A).* Springer Series in Computational Mathematics, 17, 1992.

[12] F. E. Curtis and K. Scheinberg. *Adaptive stochastic optimization: A framework for analyzing stochastic optimization algorithms.* IEEE Signal Processing Magazine 37(5):32–42, 2020.

[13] S. Dasgupta and A. Gupta. *An elementary proof of a theorem of Johnson and Lindenstrauss.* Random structures Algorithms, 21(1):60–65, 2002.

[14] K. J. Dzahini and S M. Wild. *Stochastic trust-region algorithm in random subspaces with convergence and expected complexity analyses.* arXiv2207.06452, 2022.

[15] J. Duchi, E. Hazan, and Y. Singer. *Adaptive subgradient methods for online learning and stochastic optimization.* Journal of Machine Learning Research, 12, 2011.

[16] Z. Feng, F. Roosta, D. P. Woodruff. *Non-PSD Matrix Sketching with Applications to Regression and Optimization.* Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence (UAI 2021), PMLR. 161:1841–1851, 2021.

[17] N. I. M. Gould, M. Kočvara, D. Robinson, and Ph. L. Toint. *Iterated multigrid subspace minimization.* Technical Report NAGIR20092, STFC-Rutherford Appleton Laboratory, Oxfordshire, England, 2009.

[18] N. I. M. Gould and D. Orban and Ph. L. Toint. CUTEst*: a Constrained and Unconstrained Testing Environment with safe threads for mathematical optimization*, Computational Optimization and Applications, 60:545–557, 2015.

[19] G. N. Grapiglia and G. F. D. Stella. *An adaptive trust-region method without function evaluation.* Computational Optimization and Applications, 82:31–60, 2022.

[20] S. Gratton, S. Jerad, and Ph. L. Toint. *Convergence properties of an objective-function-free optimization regularization algorithm, including an $\mathcal{O}(\epsilon^{-3/2})$ complexity bound.* SIAM Journal on Optimization, 33(3):1621–1646, 2022.

[21] S. Gratton and Ph. L. Toint. OPM*, a collection of Optimization Problems in Matlab,* arXiv:2112.05636, 2021

[22] S. Gratton and Ph. L. Toint. Adaptive regularization minimization algorithms with non-smooth norms. IMA Journal of Numerical Analysis, (to appear), 2022.

[23] A. Griewank and Ph. L. Toint. *Partitioned variable metric updates for large structured optimization problems*, Numerische Mathematik, 39:119–137, 1982.

[24] D. Kingma and J. Ba. *Adam: A method for stochastic optimization.* Proceedings in the International Conference on Learning Representations (ICLR), 2015.

[25] J. Nocedal and S. J. Wright, *Numerical Optimization.* Springer Verlag, Heidelberg, Series in Operations Research, 1999.

[26] K. Scheinberg and M. Xie. *Stochastic Adaptive Regularization Method with Cubics: A High Probability Complexity Bound.* arXiv: 2308.13161, 2023.

[27] Z. Shao. *On Random Embeddings and Their Application to Optimization.* PhD thesis, University of Oxford, Oxford, UK, 2021.

[28] T. Tieleman and G. Hinton. Lecture 6.5-RMSPROP. COURSERA: Neural Networks for Machine Learning, 2012.

[29] K. Vu, P.-L. Poiron, C. D'Ambrosio, and L. Liberti. *Random projections for quadratic programs over a Euclidean ball.* Integer Programming and Combinatorial Optimization (IPCO), Ann Arbour (USA), 2019. HAL-02869206.

[30] R. Ward, X. Wu, and L. Bottou. *Adagrad stepsizes: sharp convergence over nonconvex landscapes.* Proceedings in the International Conference on Machine Learning (ICML2019), 2019.

[31] D. P. Woodruff. *Sketching as a tool for numerical linear algebra.* Foundations and Trends in Theoretical Computer Science, 10(1-2):1–157, 2014.

[32] X. Wu, R. Ward, and L. Bottou. *WNGRAD: Learn the learning rate in gradient descent.* arXiv:1803.02865, 2018.

A Quadratic regularization for approximate second-order models

We discuss here a context in which the low-rank assumption is unnecessary and, motivated by [6], consider using quadratic regularization in conjunction with approximate quadratic models in which the Hessian $\nabla_x^2 f(x)$ is approximated by a positive-semidefinite symmetric matrix B_k. At x_k, the regularized model $m_{k,B}(s)$ of $f(x_k+s)$ then takes the form

$$m_{k,B}(s) \stackrel{\rm def}{=} T_{k,B}(x_k,s) + \frac{\sigma_k}{2}\|s\|^2, \tag{49}$$

with

$$T_{k,B}(x_k,s) \stackrel{\rm def}{=} f(x_k) + \nabla f(x_k)^T s + \frac{1}{2} s^T B_k s. \tag{50}$$

To make the use of this model well-defined, we complete AS.2, AS.4 (for $p=1$) and AS.6 and make the following assumptions.

AS.7 f is continuously differentiable in $\mathbb{R}^n$.

AS.8 The gradient of f is globally Lipschitz continuous, that is, there exist a non-negative constant L_1 such that

$$\|\nabla_x^1 f(x) - \nabla_x^1 f(y)\| \leq L_1 \|x-y\| \text{ for all } x,y \in \mathbb{R}^n.$$

AS.9 The matrix B_k is symmetric, positive-semidefinite and bounded for all $k \geq 0$, so that there exist a positive scalar κ_B such that

$$\|B_k\| \leq \kappa_B \quad \text{for} \quad k \geq 0. \tag{51}$$

Notice that AS.9[7] prevents the quadratic model (57) to be unbounded below. In particular, the use of the Gauss-Newton Hessian approximation for nonlinear least-squares problem is covered by AS.9, as well as the use of several quasi-Newton updating formulae.

Proceeding as in SKOFFARp, we let S_k be drawn from an iteration-independent distribution $\mathcal{S}$ of $\ell \times n$ random matrices (with $\ell < n$), let $s = S_k^T \widehat{s}$ be the full dimensional step and consider minimizing the sketched regularized model

$$\widehat{m}_{k,B}(\widehat{s}) \stackrel{\rm def}{=} \widehat{T}_{k,B}(x_k, \widehat{s}) + \frac{1}{2}\sigma_k \|S_k^T \widehat{s}\|^2, \tag{52}$$

where

$$\widehat{T}_{k,B}(x_k, \widehat{s}) \stackrel{\rm def}{=} f(x_k) + g_k^T S_k^T \widehat{s} + \frac{1}{2}\widehat{s}^T S_k B_k S_k^T \widehat{s}.$$

We note that, similarly to (9), $\widehat{m}_{k,B}(\widehat{s}) = m_{k,B}(s)$. The resulting SKOFFAR2B algorithm is stated on page ??.

Algorithm A.1: OFFO adaptive regularization with approximate second-order models (SKOFFAR2B)

Step 0: Initialization: An initial point $x_0 \in \mathbb{R}^n$, a regularization parameter $\nu_0 > 0$ and a requested final gradient accuracy $\epsilon \in (0, 1]$ are given, as well as the parameters $\theta > 1, \mu_{-1} \geq 0$ and $0 < \vartheta < 1$. Set $k = 0$.

Step 1: Step calculation: If $k = 0$, set $\sigma_0 = \nu_0$. Otherwise, select a matrix B_k satisfying AS.9 and

$$\sigma_k \in \Big[\vartheta\nu_k, \max[\nu_k, \mu_k]\Big],$$

where

$$\mu_k = \max\left[\mu_{k-1}, \frac{\|S_{k-1}g_k\| - \|\nabla_s^1 \widehat{T}_{k,B}(x_k, s_k)\|}{\kappa_{S,k-1}.\|s_{k-1}\|}\right]$$

with some $\kappa_{S,k-1}$ such that $\|S_{k-1}\| \leq \kappa_{S,k-1}$. Draw a random matrix $S_k \in \mathbb{R}^{\ell \times n}$ from $\mathcal{S}$ and compute a step $s_k = S_k^T \widehat{s}_k$ such that $\widehat{s}_k$ sufficiently reduces the random model $\widehat{m}_{k,B}$ defined in (61) in the sense that

$$\widehat{m}_{k,B}(\widehat{s}_k) - \widehat{m}_{k,B}(0) < 0 \tag{53}$$

and

$$\|\nabla_{\widehat{s}}^1 \widehat{T}_{k,B}(x_k, \widehat{s}_k)\| \leq \theta\sigma_k \|S_k S_k^T \widehat{s}_k\|. \tag{54}$$

Step 2: Updates. Set $x_{k+1} = x_k + s_k$ and $\nu_{k+1} = \nu_k + \nu_k\|s_k\|^2$. Increment k by one and go to Step 1.

The evaluation complexity analysis for the SKOFFAR2B algorithm is very closely related to that of SKOFFARp, and we now discuss how the results of Section 3 can be adapted to the new context.

1. Restricting our use of the Lipschitz condition to the gradient ($p = 1$), Lemma 3.1 now states that

$$f(x_{k+1}) - \widehat{T}_{k,B}(x_k, \widehat{s}_k) = f(x_{k+1}) - T_{f,p}(x_k, s_k) \leq \frac{\kappa_{LB}}{2}\|s_k\|^2, \tag{55}$$

[7] Alternatively, we could replace the condition that B_k is positive-semidefinite by the weaker condition that $B_k + \sigma_k I$ is positive-semidefinite.

and

$$\|g_{k+1} - \nabla_s^1 T_{k,B}(x_k, s_k)\| \leq \kappa_{LB}\|s_k\|,$$

where $\kappa_{LB} \stackrel{\rm def}{=} L_1 + \kappa_B$.

2. Using now the decrease (63) of the model with quadratic regularization, the decrease condition of Lemma 3.2 becomes

$$T_{k,B}(x_k, 0) - T_{k,B}(x_k, s_k) > \frac{\sigma_k}{2}\|s_k\|^2. \tag{56}$$

3. As in Lemma 3.3, we now exploit (65) to obtain that, if $\sigma_k \geq 2\kappa_{LB}$, then

$$f(x_k) - f(x_{k+1}) > \frac{\sigma_k}{4}\|s_k\|^2. \tag{57}$$

4. Lemma 3.4 is no longer valid because its assumes that the regularization order is one above that of the highest derivative used, while both these orders are now equal to two. But a simple bound on the steplength can still be derived easily.

Lemma A.1 Suppose that AS.7 and AS.9 hold. At each iteration k, we have that

$$\|s_k\| \leq \frac{2\|g_k\|}{\vartheta\nu_0}. \tag{58}$$

Proof. Using (63) and $\widehat{m}_{k,B}(\widehat{s}_k) = m_{k,B}(s_k)$ it follows that

$$\frac{1}{2}\sigma_k\|s_k\|^2 \leq -g_k^T s_k - \frac{1}{2}s_k^T B_k s_k \leq \|g_k\|\|s_k\|$$

and the thesis follows from the fact that $\sigma_k \geq \vartheta\nu_0$. □

5. The proof of Lemma 3.5 is easily adapted to the case where $p = 1$, yielding that, for all $k \geq 0$,

$$\mu_k \leq \max[\mu_{-1}, \kappa_{LB}].$$

6. The bounds (26) and (27) may now be re-writtten as $\nu_k \geq 2\kappa_{LB}/\vartheta$ and

$$k_1 \stackrel{\rm def}{=} \min\left\{k \geq 1 \mid \nu_k \geq \frac{2\kappa_{LB}}{\vartheta}\right\},$$

respectively.

7. Using (69), Lemma 3.6 then becomes

$$\nu_{k_1} \leq \nu_{\max} \stackrel{\rm def}{=} \frac{2\kappa_{LB}}{\vartheta}\left[1 + \left(\frac{\kappa_{\rm g}}{\vartheta\nu_0}\right)^2\right].$$

8. The revised version of inequality (30) in Lemma 3.7 is now given by

$$f(x_{k_1}) \leq f_{\max} \stackrel{\rm def}{=} f(x_0) + \frac{1}{2}\left(\frac{\kappa_{LB}}{\sigma_0}\nu_{\max} + \vartheta\sigma_0\right), \tag{59}$$

and the bound (31) in Lemma 3.8 is now valid with

$$\sigma_{\max} \stackrel{\rm def}{=} \max\left[\frac{4}{\vartheta}\left[f(x_0) - f_{\rm low} + \frac{1}{2}\left(\frac{\kappa_{LB}}{\sigma_0}\nu_{\max} + \vartheta\sigma_0\right)\right] + \nu_{\max}, \mu_{-1}, L_1 + \kappa_B, \frac{2\kappa_{LB}}{\vartheta}, \nu_0\right]. \tag{60}$$

9. It is of course necessary to revise our definition of a true iteration.

   Definition A.2 *Iteration $k \in \{0, \ldots, N_1(\epsilon) - 1\}$ is ω-true whenever,*

   $$\|s_k\| \geq \omega\epsilon. \tag{61}$$

   We say that, for some given "preservation parameter" $\alpha_S \in (0, 1)$ and a constant $S_{\max} > 0$, iteration k is α_S-true whenever,

   $$\|S_k g_k\| \geq \alpha_S \|g_k\| \quad \text{and} \quad \|S_k\| \leq S_{\max}. \tag{62}$$

10. Lemma 3.10 remains valid with

    $$k_* \stackrel{\rm def}{=} \left\lceil \frac{2\kappa_{LB}\epsilon^{-2}}{\vartheta\nu_0\,\omega^2} \right\rceil \quad \text{and} \quad \sigma_k \geq 2\kappa_{LB}, \quad \text{for all} \quad k \geq k_1 \tag{63}$$

    while Lemmas 3.11 and 3.12 are unchanged.

11. Since, for algorithm SKOFFAR2B, $\|g_k\| > \epsilon$ for all $k \leq N_1(\epsilon) - 1$ (instead of $\|g_{k+1}\| > \epsilon$ for $k \leq N_1(\epsilon) - 2$ for SKOFFARp), we may continue to use the proof of Lemma 3.13 and obtain the following evaluation complexity result for the SKOFFAR2B algorithm.

    Theorem A.3 Suppose that AS.2, AS.4, AS.6, AS.7, AS.8 and AS.9 hold, that $\delta_1 \in (0, 1)$ is given and that the SKOFFAR2B algorithm is applied to problem (1). Define

    $$\kappa_{\rm SKOFFAR2B} \stackrel{\rm def}{=} \frac{4\,[L_1 + \kappa_B + 2(f_{\max} - f_{\rm low})]}{\vartheta\nu_0\omega^2(1-\delta_1)\pi_S^{(1)}} \tag{64}$$

    where $f_{\max}$ is defined in (70). Then

    $$\mathbb{P}\Big[N_1(\epsilon) \leq \kappa_{\rm SKOFFAR2B}\,\epsilon^{-2} + 4\Big] \geq \left(1 - e^{-\frac{\delta_1^2}{2}\pi_S^{(1)}k_\diamond}\right)^2$$

    where $k_\diamond = \left\lceil \frac{k_*}{(1-\delta_1)\pi_S^{(1)}} \right\rceil$ with k_* given by (74).

    Of course, using quite loose Hessian approximations in (57) has the consequence that the complexity order is now $\mathcal{O}(\epsilon^{-2})$, which is identical to that of other methods (such as deterministic and stochastic trust-region or regularization) using the same type of approximations and objective function values.

12. We finally consider how Lemma 4.1 can be adapted for the use of Gaussian scaled matrices within the SKOFFAR2B algorithm.

    Lemma A.4 Suppose that AS.4, AS.7, AS.8 and AS.9 hold and that iteration $k \geq 0$ of the SKOFFAR2B algorithm is α_S-true (in the sense of (73)). Then

    $$\|s_k\| \geq \frac{\alpha_S}{S_{\max}(\kappa_B + \theta\sigma_k)}\|g_k\|.$$

    Thus iteration $k \in \{0, \ldots, N_1(\epsilon)-1\}$ is ω-true (in the sense of (72)) with $\omega = \frac{\alpha_S}{S_{\max}(\kappa_B+\theta\sigma_{\max})}$.

Proof. Since

$$S_k g_k = \nabla_s^1 \widehat{T}_{k,B}(x_k, \widehat{s}_k) - S_k B_k S_k^T \widehat{s}_k,$$

we obtain from (64) and the definition of α_S-true iteration that

$$\alpha_S \|g_k\| \leq \|S_k g_k\| \leq S_{\max}(\kappa_B + \theta\sigma_k)\|s_k\|$$

Using the bound (71) yields the desired result. □

We see that the constant (75) now involves $S_{\max}^2$. In the case of scaled Gaussian matrices, (56) then gives a dependence of the constant (75) in n/ℓ, as is the case for the (non-OFFO) trust-region method of [6].

We conclude this discussion by noting that, should the Gauss-Newton method for nonlinear least-squares be considered, AS.3 (for $p = 1$) and AS.4 can be replaced by assuming the Lipschitz continuity of the problem's Jacobian and the boundedness of the Jacobian and residual (see [25, page 295] for a proof that this is sufficient to ensure Lipschitz continuity and boundedness of the objective function's gradient).