A Stochastic Objective-Function-Free Adaptive Regularization Method with Optimal Complexity

S. Gratton[*], S. Jerad[†] and Ph. L. Toint[‡]

9 VII 2024

Abstract

A fully stochastic second-order adaptive-regularization method for unconstrained nonconvex optimization is presented which never computes the objective-function value, but yet achieves the optimal $\mathcal{O}(\epsilon^{-3/2})$ complexity bound for finding first-order critical points. The method is noise-tolerant and the inexactness conditions required for convergence depend on the history of past steps. Applications to cases where derivative evaluation is inexact and to minimization of finite sums by sampling are discussed. Numerical experiments on large binary classification problems illustrate the potential of the new method.

Keywords: stochastic optimization, adaptive regularization methods, evaluation complexity, Objective-Function-Free-Optimization (OFFO), nonconvex optimization.

1 Introduction

Adaptive gradient methods such as Adam [28], Adagrad [15], or AMSGrad [18] have become the workhorse of large-scale stochastic optimization, especially when training artificial neural networks. Given their remarkable empirical results, various complexity analyses have been developed in the stochastic regime [1, 46, 7, 3] or the deterministic one [6, 30]. A notable feature of these methods is that they do not compute the function value or an approximation thereof, making them part of OFFO (Objective Free Function Optimization) methods. Examples of such methods are adaptive gradient methods, which use only the current and past gradients, and are ubiquitous in machine learning due to their performance on large-dimensional problems. However, they still theoretically suffer from the worst-case evaluation number of first-order techniques at $\mathcal{O}\left(\epsilon^{-2}\right)$ for finding an ϵ-approximate first order solution [6, 30].

Alternatively, stochastic variants of line search [33, 19, 45], trust region [40, 9], or adaptive regularization methods [44, 38] need (sometimes tight) bounds on the accuracy of the function value proxy to achieve convergence both in theory and in practice. They typically rely on

[*]Université de Toulouse, INP, IRIT, Toulouse, France. Email: serge.gratton@enseeiht.fr. 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"

[†]Université de Toulouse, INP, IRIT, Toulouse, France. Email: sadok.jerad@toulouse-inp.fr

[‡]NAXYS, University of Namur, Namur, Belgium. Email: philippe.toint@unamur.be. Partly supported by ANITI.

second-order information or higher, and achieve a much more favorable $\mathcal{O}\left(\epsilon^{-3/2}\right)$ worst-case evaluation complexity.

To achieve faster OFFO algorithms, the use of high-order derivatives can be considered. It is well-established that second-order methods offer stronger theoretical guarantees than first-order methods, either by demonstrating superior complexity for specific variants or by being less sensitive to the problem's conditioning in practice. This approach has been successfully applied in proposing deterministic OFFO variants of adaptive regularization [36] and trust region methods [47]. These methods, while using significantly less information, still achieve the complexity rate of $\mathcal{O}(\epsilon^{-3/2})$ akin to their standard counterparts [27]. Because they are not affected by errors in the function value, they avoid the need for tight bounds on its accuracy, making the algorithms more robust to noise. This robustness has been confirmed in the numerical experiments proposed in [36]. However, to the best of our knowledge, no theoretical framework exists for high-order OFFO in the stochastic setting.

We propose a theoretical framework for stochastic OFFO adaptive regularization methods as introduced in [36]. Specifically, we present an expected error on the approximative tensors up to the pth order, with conditions dependent on the length of the previous m steps. This approach allows for greater tolerance compared to work that controls error with only the current or previous step [24, 41]. By combining these tensor conditions with classical probability and numerical analysis tools, we can extend the results from the deterministic case in [36]. Since our conditions depend only on past steps, the implementation remains straightforward and can be adapted to machine learning in terms of sampling sizes. The relaxed error bounds yield promising results when applying the second-order algorithm. Our method remains stochastic throughout, unlike other works where a deterministic behavior is adopted at the end [24, 29, 50].

The paper is organized as follows: After restating the algorithm of [36] and situating our condition on the probabilistic derivatives within the literature in Section 2, we develop the complexity rate analysis in Section 3. In Section 4, we outline some potential applications of our algorithm. Initial numerical findings of our algorithm for specific machine learning (ML) problems are presented in Section 5. Finally, conclusions and perspectives are drawn in Section 6.

2 A Stochastic OFFO adaptive regularization algorithm

2.1 Problem Formulation

We consider the problem of finding approximate minimizers of the unconstrained nonconvex optimization problem

$$\min_{x\in\mathbb{R}^n} f(x), \tag{2.1}$$

where f is a sufficiently smooth function from $\mathbb{R}^n$ into $\mathbb{R}$. As motivated in the introduction, our aim is to design an algorithm in which the objective function value is never computed and inexact derivatives may be used. Our approach is based on regularization methods. In such methods, a model of the objective function is built by "regularizing" a truncated inexact Taylor expansion of degree p.

We now detail the assumptions on the problems that we need to establish our results.
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 exists a non-negative constant L_p such that

$$\|\nabla_x^p f(x) - \nabla_x^p f(y)\| \leq L_p \|x-y\| \;\text{ for all } x, y \in \mathbb{R}^n, \qquad \text{with } L_p \geq 3, \tag{2.2}$$

where $\|.\|$ denotes the Euclidean norm for vectors in $\mathbb{R}^n$ and $\|.\|$ the associated subordinate norm for pth order tensors. In the rest of the paper, all probabilistic approximations of exact quantities will be denoted by an overline.
AS.4 If $p > 1$, there exists a constant $\kappa_{\rm high} \geq 0$ such that

$$\min_{\|d\| \leq 1} \overline{\nabla_x^i f}(x)[d]^i \geq -\kappa_{\rm high} \text{ for all } x \in \mathbb{R}^n \text{ et } i \in \{2, \ldots, p\}, \tag{2.3}$$

where $\overline{\nabla_x^i f}(x)$ is the ith approximate stochastic 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 $p = 1$).

The previous assumptions AS.1–AS.3 are standard when studying the complexity of deterministic pth order methods. Note that assumption AS.4 is weaker than imposing uniform boundedness on the sampled derivatives and is standard in the study of Objective Free Function algorithms [36]. We will return to the probabilistic bounds that must be satisfied by the tensor derivatives error after stating the algorithm.

2.2 The OFFO algorithm with stochastic derivatives

Adaptive regularization methods are iterative schemes which compute a step from an iterate x_k to the next by approximately minimizing a pth degree regularized model $m_k(s)$ of $f(x_k+s)$ of the form

$$T_{f,p}(x_k, s) + \frac{\sigma_k}{(p+1)!}\|s\|^{p+1},$$

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

$$T_{f,p}(x,s) \stackrel{\rm def}{=} f(x) + \sum_{i=1}^{p} \frac{1}{i!} \nabla_x^i f(x)[d]^i. \tag{2.4}$$

In particular AS.3 implies [8, Corollary A.8.4] that

$$\|\nabla_x^1 f(x+s) - \nabla_x^1 T_{f,p}(x,s)\| \leq \frac{L}{p!}\|s\|^p. \tag{2.5}$$

In the case where approximative derivatives are used, one then uses an approximate pth order Taylor model

$$\overline{T_{f,p}}(x,s) \stackrel{\rm def}{=} f(x) + \sum_{i=1}^{p} \frac{1}{i!} \overline{\nabla_x^i f}(x)[d]^i, \tag{2.6}$$

and the model m_k is then,

$$m_k(s) \stackrel{\rm def}{=} \overline{T_{f,p}}(x_k, s) + \frac{\sigma_k}{(p+1)!}\|s\|^{p+1}. \tag{2.7}$$

In (2.7), the approximate pth order Taylor series is "regularized" by adding the term $\frac{\sigma_k}{(p+1)!}\|s\|^{p+1}$, where σ_k is known as the "regularization parameter". This term guarantees that $m_k(s)$ is bounded below and thus makes the procedure of finding a step s_k by (approximately) minimizing $m_k(s)$ well-defined. Our proposed algorithm follows the outline line of existing ARp regularization methods [26, 20, 27] and the recent work of [36] on an optimal pth order objective free function method.

We stress that unlike inexact adaptive second-order methods analyzed in [5, 24, 44, 43], we don't evaluate the true function value nor a proxy. In what follows, all random quantities are denoted by capital letters, while the use of small letters is reserved for their realization.

Algorithm 2.1: Stochastic OFFO adaptive regularization of degree p (StOFFARp)

Step 0: Initialization: An initial point $x_0 \in \mathbb{R}^n$, a regularization parameter $\sigma_0 > 0$ and a requested final gradient accuracy $\epsilon_1 \in (0, 1]$ are given, as well as the parameter $\theta_1 > 1$. Set $k = 0$.

Step 1: Compute current derivatives Evaluate $\overline{g_k} \stackrel{\text{def}}{=} \overline{\nabla_x^1 f}(x_k)$ and $\{\overline{\nabla_x^i f}(x_k)\}_{i=2}^p$.

Step 2: Step calculation: Compute a step s_k which sufficiently reduces the model m_k defined in (2.7) in the sense that

$$m_k(s_k) - m_k(0) \le 0 \tag{2.8}$$

and

$$\|\overline{\nabla_s^1 T_{f,p}}(x_k, s_k)\| \le \theta_1 \frac{\sigma_k}{p!} \|s_k\|^p. \tag{2.9}$$

Step 3: Updates. Set

$$x_{k+1} = x_k + s_k \tag{2.10}$$

and

$$\sigma_{k+1} = \sigma_k + \sigma_k \|s_k\|^{p+1}. \tag{2.11}$$

Increment k by one and go to Step 1.

We now define the probabilistic notation which will be used throughout the paper. We emphasize that the approximate derivatives (as evaluated in Step 1) are noisy random evaluations of the exact quantities. The StOFFARp algorithm therefore generates a stochastic process

$$\{X_k, \overline{\nabla_x^i f}(X_k), \Sigma_k, S_k\}$$

on some probability space $(\Omega, \mathcal{F}, \mathbb{P})$. The associated expectation operator will be denoted by $\mathbb{E}[.]$ and $\mathbb{E}_k[.]$ will stand for the conditional expectation knowing $\left\{\overline{\nabla_x^i f}(X_j) | j \in \{0, \dots, k-1\}\right\}$. Note that $\Sigma_0 = \sigma_0$ is deterministic and we allow the initialization x_0 to be a random variable. We also denote by $G_k \stackrel{\text{def}}{=} \nabla_x^1 f(X_k)$ and $\overline{G}_k \stackrel{\text{def}}{=} \overline{\nabla_x^1 f}(X_k)$.

One could have chosen σ_k to be of the form

$$\sigma_k \in [\vartheta \nu_k, \max(\nu_k, \eta_k)] \tag{2.12}$$

where η_k is bounded non-negative sequence, ν_k is given by (2.11) and ϑ a hyperparameter in $(0, 1]$. This may be useful when devising a numerical implementation of the algorithm to closer adapt to loacl variations of the local Lipschitz constant, as done in [36, Section 5]. We preferred to keep (2.11) for the sake of simplicity in our subsequent analysis.
Compared to the vanilla adaptive regularization methods where the inequality in (2.8) must be strict (see for example [20]), we only require a simple decrease, since zero derivatives can occur in the stochastic case.

The test (2.9) follows from [32] and extends the more usual condition where the step s_k is chosen to ensure that

$$\|\nabla_s^1 m_k(s_k)\| \le \theta_1 \|s_k\|^p.$$

It is indeed easy to verify that (2.9) holds at a local minimizer of m_k with $\theta_1 \ge 1$ (see [32] for details).

We propose the following conditions on the expectation of the errors on the derivatives tensors.
AS.5 There exists $\kappa_D \ge 0$ such that at each iteration $k \ge 0$, we have that

$$\mathbb{E}_k \left[\|\nabla_x^i f(X_k) - \overline{\nabla_x^i f}(X_k)\|^{\frac{p+1}{p+1-i}} \right] \le \kappa_D \xi_k, \quad \text{for all} \quad i \in \{1, \dots, p\}, \tag{2.13}$$

with $\xi_k = \sum_{i=1}^m \|S_{k-i}\|^{p+1}$ with the conventions that

$$\|S_{-1}\| = \cdots \|S_{-m}\| \stackrel{\rm def}{=} 1 \quad \text{and} \quad \sigma_j = \frac{\sigma_0}{2^{-j}}, \quad j \in \{-m, \dots, -1\}, \tag{2.14}$$

so that (2.11) is valid even when $k \in \{-m, \dots, -1\}$.

We now discuss our proposed tensor conditions and compare them with previously used requirements on stochastic derivatives, first focussing on the case $m = 1$ (we discuss the usage of (2.13) later). Our subsequent discussion is divided into two parts: the first considers different values of p, and the scond second deals with the practical case where $p = 2$.

First and foremost, note that the condition (2.13) can be related to the following requirements on inexact tensors

$$\|\nabla_x^i f(X_k) - \overline{\nabla_x^i f}(X_k)\| \le \kappa_D \|S_k\|^{p-i+1}, \quad \text{for all} \quad i \in \{1, \dots, p\}, \tag{2.15}$$

proposed both in [41] and [27, Chapter 13]. However, one of the pitfalls of (2.15) is its implicit nature in that S_k is not available when $\overline{\nabla_x^i f}(X_k)$ is evaluated. Our condition (2.13) only uses information available from past iterations. To the best of the authors' knowledge, other stochastic adaptive regularization methods, such as that proposed by [38], additionally require more accurate function value approximations. Specifically, the condition on the approximate function value $\overline{f}(x_k)$ in [38] is that

$$|\overline{f}(x_k) - f(x_k)| \le \eta \left(-\sum_{i=1}^p \frac{1}{i!} \overline{\nabla_x^i f}(x)[s_k]^i \right), \tag{2.16}$$

where η is an algorithmic dependent constant and the term in parenthesis can be shown to be of order $\|s_k\|^{p+1}$, which is more restrictive than (2.15). Moreover, the implicit bound

(2.16) must hold for all iterations, making it somewhat impractical for stochastic problems in machine learning where subsampling is used. Note that the probabilistic assumptions required for the approximate derivatives in [38] do not treat each tensor derivative separately but are slightly more general but also more abstract as they consider their combined effect in the Taylor's expansion. For further details on stochastic adaptive high-order methods, we refer the reader to [38] and the references therein.

We now turn to the case $p = 2$ and compare our framework with previous stochastic cubic methods. The use of the past step to control the error on the inexact gradient and the Hessian was first proposed for numerical experiments in [24] with good empirical success, although the theory requires the use of the current step as in (2.15). This approach was later investigated theoretically in [11] in an inexact cubic regularization algorithm, where (2.15) is assumed to hold with $\|S_{k-1}\|$ instead of $\|S_k\|$. However, the authors unrealistically assume knowledge of the Lipschitz constant. More recently, this idea has been combined with variance reduction in [17] to devise efficient cubic regularization algorithms, but knowledge of the problem's geometry is still required.

One drawback of using only the last step size to control the error is that it may make the method exact after a few iterations, as illustrated in [24]. In contrast, we hope that using the last m steps may provide better control. Intuitively, we are able to use the last m steps to control the errors as our σ_k update rule (2.11) accumulates the past steps size lengths.

Other notable stochastic adaptive cubic regularization methods have been developed in the literature; see for example [49, 13, 37, 12, 21, 48, 17] to name a few. Let us now briefly review these references and highlight the novelty of our approach. First, note that [49, 13, 12, 21, 17] do not provide an adaptation mechanism for the regularization parameter and typically assume knowledge of the Lipschitz Hessian constant. We should also mention that the conditions proposed in [21] are very similar to ours, as they also propose bounds on

$$\mathbb{E}_k\left[\|\nabla_x^1 f(X_k) - \overline{\nabla_x^1 f}(X_k)\|^{\frac{3}{2}}\right] \quad \text{and} \quad \mathbb{E}_k\left[\|\nabla_x^2 f(X_k) - \overline{\nabla_x^2 f}(X_k)\|^3\right].$$

However, it should be noted that the analysis is limited to the second-order case and again assumes the knowledge of the Hessian Lipschitz constant.

Another line of work [37, 39], although adaptive, still requires an accurate approximation of the function value to successfully adjust the regularization parameter σ_k. Note that [37] also proposes inexact conditions that are dynamic (as is the case here in (2.13)) and controlled by the inexact gradient norm. Note also that this latter work imposes exact evaluation of the objective-function value and is restricted to the second-order case.

Finally, a "fully" stochastic cubic method has been proposed in [48], where the gradient and the Hessian satisfy a condition related to (2.15) with some probability. However, they impose additional conditions on the stochastic oracle of the function value. To our knowledge, no practical case for machine learning has been proposed in [48]. In contrast, our paper later proposes practical variants of (2.15) for machine learning problems.

3 Evaluation complexity for the inexact StOFFARp algorithm

We start our analysis of evaluation complexity by defining the following constants for notational convenience:

$$\chi_p^1 \stackrel{\text{def}}{=} \sum_{i=1}^{p} \frac{i}{i!(p+1)}, \quad \chi_p^2 \stackrel{\text{def}}{=} \sum_{i=1}^{p} \frac{p+i-1}{i!(p+1)}, \quad \kappa_p = \frac{(2p)^{\frac{p+1}{p}}}{(2p)} = (2p)^{\frac{1}{p}}. \tag{3.1}$$

$$\chi_p^3 = \sum_{i=2}^{p} \left(\frac{(i-1)}{(i-1)!p} \right)^{\frac{p+1}{p}}, \quad \chi_p^4 = 1 + \sum_{i=2}^{p} \left(\frac{(p-i+1)}{(i-1)!p} \right)^{\frac{p+1}{p}}. \tag{3.2}$$

We may now state local decrease bounds resulting from classical Taylor inequalities and the step computation mechanism. They combine the standard bounds coming from AS.3 while also taking into account the assumption on inexact derivatives AS.5 that holds in expectation.

Lemma 3.1 Suppose that AS.1, AS.3 and AS.5 hold and let $\alpha > 0$. Then

$$\mathbb{E}_k \left[\frac{\Sigma_k}{(p+1)!} \|S_k\|^{p+1} \right] \leq \mathbb{E}_k \left[f(X_k) - f(X_{k+1}) \right] + \kappa_{\rm a} \mathbb{E}_k \left[\|S_k\|^{p+1} \right] + \kappa_D \chi_p^2 \xi_k \tag{3.3}$$

and

$$\mathbb{E}_k \left[\frac{\|G_{k+1} - \overline{\nabla_s^1 T_{f,p}}(X_k, S_k)\|^{\frac{p+1}{p}}}{\Sigma_{k+1}^{\alpha}} \right] \leq \kappa_{\rm b} \mathbb{E}_k \left[\frac{\|S_k\|^{p+1}}{\Sigma_{k+1}^{\alpha}} \right] + \kappa_p \kappa_D \chi_p^4 \frac{\xi_k}{\Sigma_k^{\alpha}}, \tag{3.4}$$

where

$$\kappa_{\rm a} \stackrel{\rm def}{=} \frac{L_p}{(p+1)!} + \chi_p^1 \quad \text{and} \quad \kappa_{\rm b} \stackrel{\rm def}{=} \kappa_p \left(\frac{L_p}{p!} \right)^{\frac{p+1}{p}} + \chi_p^3. \tag{3.5}$$

Proof. By combining (2.4), (2.6), (2.5) and basic tensor inequalities, we obtain that

$$\begin{aligned} f(X_{k+1}) - \overline{T_{f,p}}(X_k, S_k) &= f(X_{k+1}) - T_{f,p}(X_k, S_k) + T_{f,p}(X_k, S_k) - \overline{T_{f,p}}(X_k, S_k) \\ &\leq \frac{L_p}{(p+1)!} \|S_k\|^{p+1} + \sum_{i=1}^{p} \frac{1}{i!} \left| \left(\nabla_x^i f(S_k) - \overline{\nabla_x^i f}(X_k) \right) [S_k]^i \right| \\ &\leq \frac{L_p}{(p+1)!} \|S_k\|^{p+1} + \sum_{i=1}^{p} \frac{1}{i!} \|\nabla_x^i f(X_k) - \overline{\nabla_x^i f}(X_k)\| \|S_k\|^i. \end{aligned}$$

Using now Young's inequality with $p_i = \frac{p+1}{p+1-i}$ and $q_i = \frac{p+1}{i}$, we derive that

$$\begin{aligned} f(X_{k+1}) - \overline{T_{f,p}}(X_k, S_k) \leq{}& \frac{L_p}{(p+1)!} \|S_k\|^{p+1} \\ &+ \sum_{i=1}^{p} \frac{1}{i!} \left(\frac{(p+1-i)\|\nabla_x^i f(X_k) - \overline{\nabla_x^i f}(X_k)\|^{\frac{p+1}{p+1-i}}}{p+1} + \frac{i\|S_k\|^{p+1}}{p+1} \right). \end{aligned}$$

Using the definition of χ_p^1 in (3.1), that of $\kappa_{\rm a}$ in (3.5) and the fact that (2.8) gives that $\overline{T_{f,p}}(X_k, S_k) \leq f(X_k) - \frac{\Sigma_k}{(p+1)!} \|S_k\|^{p+1}$, we obtain that

$$\begin{aligned} &\left(f(X_{k+1}) - f(X_k) + \frac{\Sigma_k}{(p+1)!} \|S_k\|^{p+1} \right) \leq f(X_k) - \overline{T_{f,p}}(X_k, S_k) \\ &\leq \kappa_{\rm a} \|S_k\|^{p+1} + \sum_{i=1}^{p} \frac{1}{i!} \left(\frac{(p+1-i)\|\nabla_x^i f(X_k) - \overline{\nabla_x^i f}(X_k)\|^{\frac{p+1}{p+1-i}}}{p+1} \right). \end{aligned}$$

Taking now $\mathbb{E}_k[.]$, using (2.13) and rearranging

$$\mathbb{E}_k\left[\frac{\Sigma_k}{(p+1)!}\|S_k\|^{p+1}\right] \le \mathbb{E}_k\left[f(X_{k+1}) - f(X_k)\right] + \kappa_{\rm a}\mathbb{E}_k\left[\|S_k\|^{p+1}\right] + \kappa_D \sum_{i=1}^{p} \frac{p+1-i}{(p+1)i!}\,\xi_k.$$

Using now χ_p^2 definition in (3.1), we obtain (3.3).

We turn now to the proof of (3.4). Using the triangle inequality, (2.5), (2.4) and (2.6) yields that

$$\begin{aligned}
\|G_{k+1} - \overline{\nabla_s^1 T_{f,p}}(X_k,S_k)\| &\le \|G_{k+1} - \nabla_s^1 T_{f,p}(X_k,S_k)\| + \|\nabla_s^1 T_{f,p}(X_k,S_k) - \overline{\nabla_s^1 T_{f,p}}(X_k,S_k)\| \\
&\le \frac{L_p}{p!}\|S_k\|^p + \sum_{i=1}^{p}\frac{1}{(i-1)!}\left|\left(\nabla_x^i f(X_k) - \overline{\nabla_x^i f}(X_k)\right)[S_k]^{i-1}\right| \\
&\le \frac{L_p}{p!}\|S_k\|^p + \sum_{i=1}^{p}\frac{1}{(i-1)!}\|\nabla_x^i f(X_k) - \overline{\nabla_x^i f}(X_k)\|\|S_k\|^{i-1} \\
&\le \frac{L_p}{p!}\|S_k\|^p + \|\nabla_x^1 f(X_k) - \overline{\nabla_x^1 f}(X_k)\| \\
&\quad + \sum_{i=2}^{p}\frac{1}{(i-1)!}\|\nabla_x^i f(X_k) - \overline{\nabla_x^i f}(X_k)\|\|S_k\|^{i-1}.
\end{aligned}$$

Again using Young's inequality with $p_i = \frac{p}{p+1-i}$ and $q_i = \frac{p}{i-1}$ for $i \in \{2,\ldots,p\}$, we derive that

$$\begin{aligned}
\|G_{k+1} - \overline{\nabla_s^1 T_{f,p}}(X_k,S_k)\| &\le \frac{L_p}{p!}\|S_k\|^p + \|\nabla_x^1 f(X_k) - \overline{\nabla_x^1 f}(X_k)\| \\
&\quad + \sum_{i=2}^{p}\frac{1}{(i-1)!}\left(\frac{p+1-i}{p}\|\nabla_x^i f(X_k) - \overline{\nabla_x^i f}(X_k)\|^{\frac{p}{p+1-i}} + \frac{i-1}{p}\|S_k\|^p\right).
\end{aligned}$$

Taking the last inequality to the $\frac{p+1}{p}$ power, using the fact that $x^{\frac{p+1}{p}}$ is a convex function, the definition of κ_p in (3.1), the fact that the left-hand side has $2p$ terms and dividing both sides of the inequality by Σ_{k+1}^{α} gives that

$$\begin{aligned}
\frac{\|G_{k+1} - \overline{\nabla_s^1 T_{f,p}}(X_k,S_k)\|^{\frac{p+1}{p}}}{\Sigma_{k+1}^{\alpha}} &\le \kappa_p\left(\frac{L_p}{p!}\right)^{\frac{p+1}{p}}\frac{\|S_k\|^{p+1}}{\Sigma_{k+1}^{\alpha}} + \kappa_p\frac{\|\nabla_x^1 f(X_k) - \overline{\nabla_x^1 f}(X_k)\|^{\frac{p+1}{p}}}{\Sigma_{k+1}^{\alpha}} \\
&\quad + \kappa_p\sum_{i=2}^{p}\frac{1}{(i-1)!^{\frac{p+1}{p}}}\left(\left[\frac{p+1-i}{p}\right]^{\frac{p+1}{p}}\frac{\|\nabla_x^i f(X_k) - \overline{\nabla_x^i f}(X_k)\|^{\frac{p+1}{p+1-i}}}{\Sigma_{k+1}^{\alpha}}\right. \\
&\quad \left. + \left[\frac{i-1}{p}\right]^{\frac{p+1}{p}}\frac{\|S_k\|^{p+1}}{\Sigma_{k+1}^{\alpha}}\right).
\end{aligned}$$

Taking the conditional expectation over the past iterations, using (2.13) and the fact that

$\frac{1}{\Sigma_{k+1}^\alpha} \leq \frac{1}{\Sigma_k^\alpha}$ for the terms $\|\nabla_x^i f(X_k) - \overline{\nabla_x^i f}(X_k)\|$, we derive that

$$\begin{aligned}\mathbb{E}_k\left[\frac{\|G_{k+1} - \overline{\nabla_s^1 T_{f,p}}(X_k,S_k)\|^{\frac{p+1}{p}}}{\Sigma_{k+1}^\alpha}\right] &\leq \kappa_p\left(\frac{L_p}{p!}\right)^{\frac{p+1}{p}}\mathbb{E}_k\left[\frac{\|S_k\|^{p+1}}{\Sigma_{k+1}^\alpha}\right] + \kappa_p\kappa_D\frac{\xi_k}{\Sigma_k^\alpha}\\ &\quad + \kappa_p\sum_{i=2}^p \frac{1}{(i-1)!^{\frac{p+1}{p}}}\left(\left[\frac{p+1-i}{p}\right]^{\frac{p+1}{p}}\kappa_D\frac{\xi_k}{\Sigma_k^\alpha}\right.\\ &\quad \left. + \left[\frac{i-1}{p}\right]^{\frac{p+1}{p}}\mathbb{E}_k\left[\frac{\|S_k\|^{p+1}}{\Sigma_{k+1}^\alpha}\right]\right).\end{aligned}$$

Using now the definitions of χ_p^3 and χ_p^4 in (3.2) and that of $\kappa_{\rm b}$ in (3.5), we obtain (3.4).
□

The next lemma provides two useful upper bounds on the gradient norm at iteration $k+1$ divided by the regularization parameter. This also clarifies why Lemma 3.1 was stated with a generic α parameter: we will need this result for two different values of α in the following proof, which is inspired by [20, Lemma 2.3], but it also takes into account the derivative tensor errors that hold in expectation (AS.5) and the update rule of the regularization parameter σ_k in (2.11).

Lemma 3.2 Suppose that AS.1, AS.3 and AS.5 hold and let $k \geq 0$. Then,

$$\mathbb{E}_k\left[\frac{\|G_{k+1}\|^{\frac{p+1}{p}}}{\Sigma_{k+1}^{\frac{p+1}{p}}}\right] \leq \kappa_{\rm c}\mathbb{E}_k\left[\frac{\Sigma_{k+1}-\Sigma_k}{\Sigma_{k+1}}\right] + \kappa_{\rm d}\sum_{j=k-m}^{k-1}\frac{\Sigma_{j+1}-\Sigma_j}{\Sigma_{j+1}} \tag{3.6}$$

and

$$\mathbb{E}_k\left[\frac{\|G_{k+1}\|^{\frac{p+1}{p}}}{\Sigma_{k+1}^{\frac{1}{p}}}\right] \leq \kappa_{\rm c}\mathbb{E}_k\left[\Sigma_{k+1}-\Sigma_k\right] + \kappa_{\rm d}\sum_{j=k-m}^{k-1}[\Sigma_{j+1}-\Sigma_j], \tag{3.7}$$

where

$$\kappa_{\rm c} \stackrel{\rm def}{=} \frac{2^{\frac{1}{p}}}{\sigma_0^{\frac{p+1}{p}}}\left(\kappa_{\rm b} + \frac{\theta_1^{\frac{p+1}{p}}}{p!^{\frac{p+1}{p}}}\sigma_0^{\frac{p+1}{p}}\right) \quad \text{and} \quad \kappa_{\rm d} \stackrel{\rm def}{=} \frac{2^{\frac{mp+1}{p}}\kappa_p\kappa_D\chi_p^4}{\sigma_0^{\frac{p+1}{p}}}, \tag{3.8}$$

with κ_p, χ_p^3 and χ_p^4 defined in (3.1) and (3.2) and $\kappa_{\rm b}$ given by (3.5).

Proof. First consider $\alpha \in \{\frac{1}{p}, \frac{p+1}{p}\}$. From the triangular inequality and the fact that

$(x+y)^{\frac{p+1}{p}} \le 2^{\frac{1}{p}}\left(x^{\frac{p+1}{p}}+y^{\frac{p+1}{p}}\right)$ for $x,y\ge 0$ and (2.9), we obtain

$$
\begin{aligned}
\frac{\|G_{k+1}\|^{\frac{p+1}{p}}}{\Sigma_{k+1}^{\alpha}} &\le \frac{\left(\|G_{k+1}-\overline{\nabla_s^1 T_{f,p}}(X_k,S_k)\|+\|\overline{\nabla_s^1 T_{f,p}}(X_k,S_k)\|\right)^{\frac{p+1}{p}}}{\Sigma_{k+1}^{\alpha}}\\
&\le 2^{\frac{1}{p}}\frac{\|G_{k+1}-\overline{\nabla_s^1 T_{f,p}}(X_k,S_k)\|^{\frac{p+1}{p}}}{\Sigma_{k+1}^{\alpha}} + 2^{\frac{1}{p}}\frac{\|\overline{\nabla_s^1 T_{f,p}}(X_k,S_k)\|^{\frac{p+1}{p}}}{\Sigma_{k+1}^{\alpha}}\\
&\le 2^{\frac{1}{p}}\left(\frac{\|G_{k+1}-\overline{\nabla_s^1 T_{f,p}}(X_k,S_k)\|^{\frac{p+1}{p}}}{\Sigma_{k+1}^{\alpha}} + \frac{(\theta_1\Sigma_k)^{\frac{p+1}{p}}}{p!^{\frac{p+1}{p}}\Sigma_{k+1}^{\alpha}}\|S_k\|^{p+1}\right).
\end{aligned}
$$

Taking $\mathbb{E}_k[.]$ and using (3.4), we derive that

$$
\mathbb{E}_k\left[\frac{\|G_{k+1}\|^{\frac{p+1}{p}}}{\Sigma_{k+1}^{\alpha}}\right] \le 2^{\frac{1}{p}}\kappa_{\rm b}\mathbb{E}_k\left[\frac{\|S_k\|^{p+1}}{\Sigma_{k+1}^{\alpha}}\right] + 2^{\frac{1}{p}}\kappa_p\kappa_D\chi_p^4\frac{\xi_k}{\Sigma_k^{\alpha}} + 2^{\frac{1}{p}}\frac{\theta_1^{\frac{p+1}{p}}}{p!^{\frac{p+1}{p}}}\mathbb{E}_k\left[\Sigma_k^{\frac{p+1}{p}}\frac{\|S_k\|^{p+1}}{\Sigma_{k+1}^{\alpha}}\right]. \tag{3.9}
$$

We first prove (3.6) and start with $\alpha=\frac{p+1}{p}$. Using that $\xi_k=\sum_{j=1}^m\|S_{k-j}\|^{p+1}$, that $\|S_j\|^{p+1}=\frac{\Sigma_{j+1}-\Sigma_j}{\Sigma_j}$ for $j\in\{k-m,\dots,k\}$, (3.9), and also that Σ_k is non-decreasing, also $\Sigma_k\ge\sigma_0$, $\Sigma_{k-m}\ge\frac{\sigma_0}{2^m}$ for $k\ge 0$, both facts resulting from (2.11) and (2.14), we derive that

$$
\begin{aligned}
\mathbb{E}_k\left[\frac{\|G_{k+1}\|^{\frac{p+1}{p}}}{\Sigma_{k+1}^{\frac{p+1}{p}}}\right] &\le 2^{\frac{1}{p}}\kappa_{\rm b}\mathbb{E}_k\left[\frac{\Sigma_{k+1}-\Sigma_k}{\Sigma_{k+1}}\frac{1}{\Sigma_k\Sigma_{k+1}^{\frac{1}{p}}}\right] + 2^{\frac{1}{p}}\kappa_p\kappa_D\chi_p^4\sum_{j=k-m}^{k-1}\frac{\Sigma_{j+1}-\Sigma_j}{\Sigma_j}\frac{1}{\Sigma_{j+1}\Sigma_k^{\frac{1}{p}}}\\
&\quad + 2^{\frac{1}{p}}\frac{\theta_1^{\frac{p+1}{p}}}{p!^{\frac{p+1}{p}}}\mathbb{E}_k\left[\frac{\Sigma_{k+1}-\Sigma_k}{\Sigma_{k+1}}\frac{\Sigma_k^{\frac{1}{p}}}{\Sigma_{k+1}^{\frac{1}{p}}}\right]\\
&\le \frac{2^{\frac{1}{p}}}{\sigma_0^{\frac{p+1}{p}}}\kappa_{\rm b}\mathbb{E}_k\left[\frac{\Sigma_{k+1}-\Sigma_k}{\Sigma_{k+1}}\right] + \kappa_{\rm d}\sum_{j=k-m}^{k-1}\frac{\Sigma_{j+1}-\Sigma_j}{\Sigma_{j+1}} + 2^{\frac{1}{p}}\frac{\theta_1^{\frac{p+1}{p}}}{p!^{\frac{p+1}{p}}}\mathbb{E}_k\left[\frac{\Sigma_{k+1}-\Sigma_k}{\Sigma_{k+1}}\right],
\end{aligned}
$$

where $\kappa_{\rm d}$ is defined in (3.8). Rearranging the last inequality and using the definition of $\kappa_{\rm c}$ in (3.8) yields inequality (3.6).

Consider now the case where $\alpha=\frac{1}{p}$. Again, using the same arguments used to prove (3.6), we deduce that

$$
\begin{aligned}
\mathbb{E}_k\left[\frac{\|G_{k+1}\|^{\frac{p+1}{p}}}{\Sigma_{k+1}^{\frac{1}{p}}}\right] &\le 2^{\frac{1}{p}}\kappa_{\rm b}\mathbb{E}_k\left[\frac{\Sigma_{k+1}-\Sigma_k}{\Sigma_k}\frac{1}{\Sigma_{k+1}^{\frac{1}{p}}}\right] + 2^{\frac{1}{p}}\kappa_p\kappa_D\chi_p^4\sum_{j=k-m}^{k-1}\frac{\Sigma_{j+1}-\Sigma_j}{\Sigma_j}\frac{1}{\Sigma_k^{\frac{1}{p}}}\\
&\quad + 2^{\frac{1}{p}}\frac{\theta_1^{\frac{p+1}{p}}}{p!^{\frac{p+1}{p}}}\mathbb{E}_k\left[\frac{\Sigma_k^{\frac{1}{p}}\left(\Sigma_{k+1}-\Sigma_k\right)}{\Sigma_{k+1}^{\frac{1}{p}}}\right]\\
&\le \frac{2^{\frac{1}{p}}}{\sigma_0^{\frac{p+1}{p}}}\kappa_{\rm b}\mathbb{E}_k\left[\Sigma_{k+1}-\Sigma_k\right] + \kappa_{\rm d}\sum_{j=k-m}^{k-1}\left[\Sigma_{j+1}-\Sigma_j\right] + 2^{\frac{1}{p}}\frac{\theta_1^{\frac{p+1}{p}}}{p!^{\frac{p+1}{p}}}\mathbb{E}_k\left[\Sigma_{k+1}-\Sigma_k\right].
\end{aligned}
$$

Rearranging the last inequality gives the second result of the lemma. □

The following lemma restates a result similar to that developed when analyzing the exact version of Algorithm 2.1 in [36], but we extend it by providing a bound on $\|S_k\|^{p+1}$, under the assumption that $\|S_k\|^{p+1}$ is bounded by a constant depending on AS.4.

Lemma 3.3 Suppose that AS.1 and AS.4 hold. At each iteration k, we have that

$$\|S_k\| \leq 2\max\left(\eta, \left(\frac{(p+1)!\|\overline{G_k}\|}{\Sigma_k}\right)^{\frac{1}{p}}\right), \tag{3.10}$$

where

$$\eta = \max_{i\in\{2,\ldots,p\}} \left[\frac{\kappa_{\text{high}}(p+1)!}{i!\,\sigma_0}\right]^{\frac{1}{p-i+1}}. \tag{3.11}$$

Moreover,

$$\|S_k\|^{p+1}\mathbb{1}_{\|S_k\|\leq 2\eta} \leq (1+2^{p+1}\eta^{p+1})\frac{\Sigma_{k+1}-\Sigma_k}{\Sigma_{k+1}}\mathbb{1}_{\|S_k\|\leq 2\eta}. \tag{3.12}$$

Proof. See Appendix A for the proof of inequality (3.10). We now turn to establishing (3.12). From (2.11), and the fact that $\|S_k\|^{p+1}\mathbb{1}_{\|S_k\|\leq 2\eta} \leq (2\eta)^{p+1}$, we hav that

$$\Sigma_{k+1}\mathbb{1}_{\|S_k\|\leq 2\eta} = \Sigma_k\mathbb{1}_{\|S_k\|\leq 2\eta} + \|S_k\|^{p+1}\Sigma_k\mathbb{1}_{\|S_k\|\leq 2\eta} \leq \Sigma_k\mathbb{1}_{\|S_k\|\leq 2\eta}\left(1+(2\eta)^{p+1}\right),$$

which yields that

$$\frac{\Sigma_{k+1}\mathbb{1}_{\|S_k\|\leq 2\eta}}{1+(2\eta)^{p+1}} \leq \Sigma_k\mathbb{1}_{\|S_k\|\leq 2\eta}.$$

Multiplying both sides of the previous inequality by $\|S_k\|^{p+1}$, adding $\Sigma_k\mathbb{1}_{\|S_k\|\leq 2\eta}$, and using identity (2.11), we derive that

$$\Sigma_k\mathbb{1}_{\|S_k\|\leq 2\eta} + \frac{\Sigma_{k+1}}{1+(2\eta)^{p+1}}\|S_k\|^{p+1}\mathbb{1}_{\|S_k\|\leq 2\eta} \leq \mathbb{1}_{\|S_k\|\leq 2\eta}\left(\Sigma_k + \Sigma_k\|S_k\|^{p+1}\right) = \Sigma_{k+1}\mathbb{1}_{\|S_k\|\leq 2\eta}.$$

Now rearranging the last inequality yields (3.12). □

Why we have showcased the term $\frac{\Sigma_{k+1}-\Sigma_k}{\Sigma_{k+1}}$ in the upper bound of (3.12), it will become clear later in the paper. We now turn to proving a bound on $\mathbb{E}\left[\left(\frac{\|\overline{G_k}\|}{\Sigma_k}\right)^{\frac{p+1}{p}}\right]$, as it will allow us to derive a bound on $\mathbb{E}\left[\|S_k\|^{p+1}\right]$.

Lemma 3.4 Suppose that AS.1, AS.3 and AS.5 hold and consider an iteration $k \geq 1$. Then, we have that

$$\mathbb{E}\left[\left(\frac{\|\overline{G_k}\|}{\Sigma_k}\right)^{\frac{p+1}{p}}\right] \leq \frac{\kappa_D 2^{\frac{mp+2}{p}}}{\sigma_0^{\frac{p+1}{p}}} \sum_{j=k-m}^{k-1} \mathbb{E}\left[\frac{\Sigma_{j+1}-\Sigma_j}{\Sigma_{j+1}}\right] + 2^{\frac{1}{p}}\kappa_{\rm d} \sum_{j=k-m-1}^{k-2} \mathbb{E}\left[\frac{\Sigma_{j+1}-\Sigma_j}{\Sigma_{j+1}}\right]$$
$$+ 2^{\frac{1}{p}}\kappa_{\rm c}\mathbb{E}\left[\frac{\Sigma_k-\Sigma_{k-1}}{\Sigma_k}\right], \tag{3.13}$$

where $\kappa_{\rm c}$ is given by (3.8). We also have that, for $k=0$,

$$\mathbb{E}\left[\left(\frac{\|\overline{G_0}\|}{\sigma_0}\right)^{\frac{p+1}{p}}\right] \leq 2^{\frac{1}{p}}\left(\frac{\kappa_D}{\sigma_0^{\frac{p+1}{p}}} + \frac{\mathbb{E}\left[\|G_0\|^{\frac{p+1}{p}}\right]}{\sigma_0^{\frac{p+1}{p}}}\right). \tag{3.14}$$

Proof. Consider an arbitrary positive k. From the triangle inequality and the fact that $(x+y)^{\frac{p+1}{p}} \leq 2^{\frac{1}{p}}\left(x^{\frac{p+1}{p}} + y^{\frac{p+1}{p}}\right)$ for $x, y \geq 0$, we derive that

$$\left(\frac{\|\overline{G_k}\|}{\Sigma_k}\right)^{\frac{p+1}{p}} \leq \left(\frac{\|\overline{G_k}-G_k\| + \|G_k\|}{\Sigma_k}\right)^{\frac{p+1}{p}} \leq 2^{\frac{1}{p}}\left(\frac{\|\overline{G_k}-G_k\|^{\frac{p+1}{p}}}{\Sigma_k^{\frac{p+1}{p}}} + \frac{\|G_k\|^{\frac{p+1}{p}}}{\Sigma_k^{\frac{p+1}{p}}}\right). \tag{3.15}$$

Taking $k=0$, using the fact that $\Sigma_k = \sigma_0$ and (2.13) with $i=1$ yields (3.14).

Consider now $k \geq 1$. From the inequality (3.15), it is sufficient to provide a bound on the two terms of the left-hand side in order to establish the lemma's result.

Let us first provide a bound on $\frac{\|\overline{G_k}-G_k\|^{\frac{p+1}{p}}}{\Sigma_k^{\frac{p+1}{p}}}$ in expectation. Using that Σ_k is measurable with respect to the past, (2.13) for $i=1$, (2.11) and that $\Sigma_j \geq \frac{\sigma_0}{2^m}$ for $j \geq -m$ from (2.14) and $\Sigma_k \geq \sigma_0$ for $k \geq 0$ and that $\xi_k = \sum_{j=1}^m \|S_{k-j}\|^{p+1}$, we derive

$$\begin{aligned}
\mathbb{E}_k\left[\frac{\|\overline{G_k}-G_k\|^{\frac{p+1}{p}}}{\Sigma_k^{\frac{p+1}{p}}}\right] &= \frac{1}{\Sigma_k^{\frac{p+1}{p}}}\mathbb{E}_k\left[\|\overline{G_k}-G_k\|^{\frac{p+1}{p}}\right] \\
&\leq \frac{\kappa_D}{\Sigma_k^{\frac{p+1}{p}}}\xi_k = \frac{\kappa_D}{\Sigma_k^{\frac{p+1}{p}}} \sum_{j=k-m}^{k-1} \frac{\Sigma_{j+1}-\Sigma_j}{\Sigma_j} \leq \frac{\kappa_D 2^{\frac{mp+1}{p}}}{\sigma_0^{\frac{p+1}{p}}} \sum_{j=k-m}^{k-1} \frac{\Sigma_{j+1}-\Sigma_j}{\Sigma_{j+1}}.
\end{aligned}$$

Taking the full expectation in the last inequality yields

$$\mathbb{E}\left[\frac{\|\overline{G_k}-G_k\|^{\frac{p+1}{p}}}{\Sigma_k^{\frac{p+1}{p}}}\right] \leq \frac{\kappa_D 2^{\frac{mp+1}{p}}}{\sigma_0^{\frac{p+1}{p}}} \sum_{j=k-m}^{k-1} \mathbb{E}\left[\frac{\Sigma_{j+1}-\Sigma_j}{\Sigma_{j+1}}\right]. \tag{3.16}$$

Let us now focus on the second term $\frac{\|G_k\|^{\frac{p+1}{p}}}{\Sigma_k^{\frac{p+1}{p}}}$. Using (3.6), and taking the full expectation yields that

$$\mathbb{E}\left[\frac{\|G_k\|^{\frac{p+1}{p}}}{\Sigma_k^{\frac{p+1}{p}}}\right] \leq \kappa_{\rm c}\mathbb{E}\left[\frac{\Sigma_k - \Sigma_{k-1}}{\Sigma_k}\right] + \kappa_{\rm d}\sum_{j=k-m-1}^{k-2}\mathbb{E}\left[\frac{\Sigma_{j+1} - \Sigma_j}{\Sigma_{j+1}}\right]. \tag{3.17}$$

Note that as $k \geq 1$, the last term in the right-hand side of the previous inequality is always well-defined. Injecting now (3.16) and (3.17) in (3.15) gives the desired result. □

The next lemma explains the specific dependency of the bounds (3.13) and (3.12) with respect to Σ_k.

Lemma 3.5 Let $\{(a_j)_{j\in\{m,\dots,n\}}\}$ be a positive nondecreasing sequence with $m < n$ and $(m, n) \in \mathbb{Z}^2$. Then, we have that

$$\sum_{j=m+1}^{n}\frac{a_j - a_{j-1}}{a_j} \leq \log(a_n) - \log(a_m). \tag{3.18}$$

Proof. Let $j \geq m+1$ and suppose that $a_j > a_{j-1}$. By using the concavity of the logarithm and since the sequence a_i is nonnegative and further rearranging, we derive

$$\frac{a_j - a_{j-1}}{a_j} \leq \log(a_j) - \log(a_{j-1}).$$

Note that the last inequality is still valid even when $a_j = a_{j-1}$. Thus, summing the last inequality for $j \in \{m+1, \dots, n\}$ yields (3.18). □

Combining the results of Lemmas 3.3, 3.4 and 3.5, we are now able to provide a bound on $\sum_{j=0}^{k}\mathbb{E}\left[\|S_j\|^{p+1}\right]$.

Lemma 3.6 Suppose that AS.1, AS.3, AS.4 and AS.5 hold. Then

$$\sum_{j=0}^{k} \mathbb{E}\left[\|S_j\|^{p+1}\right] \leq \kappa_{\rm e} + \kappa_{\rm f} \log(\mathbb{E}\left[\Sigma_{k+1}\right]), \tag{3.19}$$

where $\kappa_{\rm e}$ and $\kappa_{\rm f}$ are defined by

$$\begin{aligned}\kappa_{\rm e} &\stackrel{\rm def}{=} 2^{p+1}(p+1)!^{\frac{p+1}{p}}\left(2^{\frac{1}{p}}\left(\frac{\kappa_D}{\sigma_0^{\frac{p+1}{p}}} + \frac{\mathbb{E}\left[\|G_0\|^{\frac{p+1}{p}}\right]}{\sigma_0^{\frac{p+1}{p}}}\right) + 2^{\frac{1}{p}}\kappa_{\rm d}\left(\log(2)\frac{m+1}{2} - \log(\sigma_0)\right)\right.\\ &\left.\quad - 2^{\frac{1}{p}}\kappa_{\rm c}\log(\sigma_0) + \frac{\kappa_D 2^{\frac{mp+2}{p}} m}{\sigma_0^{\frac{p+1}{p}}}\left(\log(2)\frac{m-1}{2} - \log(\sigma_0)\right)\right) - (1+2^{p+1}\eta^{p+1})\log(\sigma_0),\end{aligned} \tag{3.20}$$

where $\kappa_{\rm c}$ is given by (3.8) and

$$\kappa_{\rm f} \stackrel{\rm def}{=} 1 + 2^{p+1}\eta^{p+1} + \frac{2^{p+1}(p+1)!^{\frac{p+1}{p}}}{\sigma_0^{\frac{p+1}{p}}}\left(\kappa_D 2^{\frac{mp+2}{p}} m + 2^{\frac{mp+2}{p}}\chi_p^4 \kappa_D \kappa_p m + 2^{\frac{2}{p}}\kappa_{\rm c}\right). \tag{3.21}$$

Proof. From inequalities (3.10) and (3.12), we have that

$$\begin{aligned}\|S_j\|^{p+1} &\leq \|S_j\|^{p+1}\mathbb{1}_{\|S_j\|\leq 2\eta} + \|S_j\|^{p+1}\mathbb{1}_{\|S_j\|\leq 2\left(\frac{(p+1)!\|\overline{G}_j\|}{\Sigma_j}\right)^{\frac{1}{p}}}\\ &\leq (1+2^{p+1}\eta^{p+1})\frac{\Sigma_{j+1}-\Sigma_j}{\Sigma_{j+1}} + 2^{p+1}\left(\frac{(p+1)!\|\overline{G}_j\|}{\Sigma_j}\right)^{\frac{p+1}{p}}.\end{aligned}$$

Taking the full expectation in the last inequality, summing for $j \in \{0,\ldots,k\}$, using (3.13) for $j \geq 1$ and (3.14) when $j = 0$, we derive that

$$\begin{aligned}\sum_{j=0}^{k}\mathbb{E}\left[\|S_j\|^{p+1}\right] &\leq (1+2^{p+1}\eta^{p+1})\sum_{j=0}^{k}\mathbb{E}\left[\frac{\Sigma_{j+1}-\Sigma_j}{\Sigma_{j+1}}\right] + 2^{p+1}(p+1)!^{\frac{p+1}{p}}2^{\frac{1}{p}}\left(\frac{\kappa_D}{\sigma_0^{\frac{p+1}{p}}} + \frac{\mathbb{E}\left[\|G_0\|^{\frac{p+1}{p}}\right]}{\sigma_0^{\frac{p+1}{p}}}\right)\\ &\quad + 2^{p+1}(p+1)!^{\frac{p+1}{p}}\sum_{j=1}^{k}\left(\frac{\kappa_D 2^{\frac{mp+2}{p}}}{\sigma_0^{\frac{p+1}{p}}}\left[\sum_{\ell=j-m}^{j-1}\mathbb{E}\left[\frac{\Sigma_{\ell+1}-\Sigma_\ell}{\Sigma_{\ell+1}}\right] + \chi_p^4\kappa_p\sum_{\ell=j-m-1}^{j-2}\mathbb{E}\left[\frac{\Sigma_{\ell+1}-\Sigma_\ell}{\Sigma_{\ell+1}}\right]\right]\right.\\ &\quad\left. + \frac{2^{\frac{2}{p}}}{\sigma_0^{\frac{p+1}{p}}}\kappa_{\rm c}\mathbb{E}\left[\frac{\Sigma_j-\Sigma_{j-1}}{\Sigma_j}\right]\right).\end{aligned}$$

We now provide a bound on the two sums involving $\mathbb{E}\left[\frac{\Sigma_{\ell+1}-\Sigma_\ell}{\Sigma_{\ell+1}}\right]$. By inverting the two sums, the linearity of the expectation, Lemma 3.5 and the fact that Σ_k is non-decreasing, we derive after some simplification, that

$$\begin{aligned}\sum_{j=1}^{k}\sum_{\ell=j-m}^{j-1}\mathbb{E}\left[\frac{\Sigma_{\ell+1}-\Sigma_\ell}{\Sigma_{\ell+1}}\right] &= \sum_{\ell=1}^{m}\sum_{j=1}^{k}\mathbb{E}\left[\frac{\Sigma_{j-\ell+1}-\Sigma_{j-\ell}}{\Sigma_{j-\ell+1}}\right]\\ &\leq \sum_{\ell=1}^{m}\mathbb{E}\left[\log(\Sigma_{k-\ell+1})-\log(\Sigma_{1-\ell})\right] \leq m\mathbb{E}\left[\log(\Sigma_k)\right]-\sum_{\ell=1}^{m}\log\left(\frac{\sigma_0}{2^{1-l}}\right)\\ &\leq m(\mathbb{E}\left[\log(\Sigma_k)\right]-\log(\sigma_0))+\log(2)\frac{m^2-m}{2}. \end{aligned}\tag{3.22}$$

Similarly for $\sum_{j=1}^{k}\sum_{\ell=j-m-1}^{j-2}\mathbb{E}\left[\frac{\Sigma_{\ell+1}-\Sigma_\ell}{\Sigma_{\ell+1}}\right]$ and using the same arguments as above yields that

$$\sum_{j=1}^{k}\sum_{\ell=j-m-1}^{j-2}\mathbb{E}\left[\frac{\Sigma_{\ell+1}-\Sigma_\ell}{\Sigma_{\ell+1}}\right] = \sum_{\ell=1}^{m}\sum_{j=1}^{k}\mathbb{E}\left[\frac{\Sigma_{j-\ell}-\Sigma_{j-\ell-1}}{\Sigma_{j-\ell}}\right] \leq m(\mathbb{E}\left[\log(\Sigma_{k-1})\right]-\log(\sigma_0))+\log(2)\frac{m^2+m}{2}.\tag{3.23}$$

Now using (3.22), (3.23) and the linearity of the expectation, we obtain that

$$\begin{aligned}\sum_{j=0}^{k}\mathbb{E}\left[\|S_j\|^{p+1}\right] \leq{}& (1+2^{p+1}\eta^{p+1})\mathbb{E}\left[\log(\Sigma_{k+1})-\log(\sigma_0)\right]+2^{p+1}(p+1)!^{\frac{p+1}{p}}2^{\frac{1}{p}}\left(\frac{\kappa_D}{\sigma_0^{\frac{p+1}{p}}}+\frac{\mathbb{E}\left[\|G_0\|^{\frac{p+1}{p}}\right]}{\sigma_0^{\frac{p+1}{p}}}\right)\\ &+2^{p+1}(p+1)!^{\frac{p+1}{p}}\left(\frac{\kappa_D 2^{\frac{mp+2}{p}}}{\sigma_0^{\frac{p+1}{p}}}m\left(\mathbb{E}\left[\log(\Sigma_k)-\log(\sigma_0)\right]+\frac{m-1}{2}\log(2)\right)\right.\\ &+\frac{2^{\frac{mp+2}{p}}\chi_p^4\kappa_D\kappa_p}{\sigma_0^{\frac{p+1}{p}}}m\left(\mathbb{E}\left[\log(\Sigma_{k-1})-\log(\sigma_0)\right]+\log(2)\frac{m+1}{2}\right)\\ &\left.+\frac{2^{\frac{2}{p}}}{\sigma_0^{\frac{p+1}{p}}}\kappa_{\rm c}\mathbb{E}\left[\log(\Sigma_k)-\log(\sigma_0)\right]\right).\end{aligned}$$

Using Jensen inequality, the fact that Σ_k is a non-decreasing sequence, and the definition of $\kappa_{\rm e}$ and $\kappa_{\rm f}$ in (3.20) and (3.21) yields the desired result. □

We are now ready to give an upper bound on $\mathbb{E}\left[\Sigma_k\right]$, a crucial step in the theory of adaptive regularization methods (see [20] or [36] for instance). We will also need a result on the solutions of a nonlinear equation that combines logarithmic, linear, and constant terms. The latter is given in Appendix B.

Lemma 3.7 Suppose that AS.1–AS.5 hold. Then for all $k \geq 0$, we have that

$$\mathbb{E}\left[\Sigma_k\right] \leq \sigma_{\max} \stackrel{\text{def}}{=} -(p+1)!(\kappa_{\text{a}}\kappa_{\text{f}} + \kappa_D \chi_p^2 \kappa_{\text{f}})\, W_{-1}\left(\frac{-e^{-\kappa_{\text{g}}}}{(p+1)!(\kappa_{\text{a}}\kappa_{\text{f}} + \kappa_D\chi_p^2\kappa_{\text{f}})}\right), \quad (3.24)$$

where κ_{a} is given by (3.5), κ_{f} by (3.21) and

$$\kappa_{\text{g}} = \frac{\Gamma_0 + \frac{\Sigma_0}{(p+1)!} + \kappa_{\text{a}}\kappa_{\text{e}} + \kappa_D\chi_p^2(1+\kappa_{\text{e}})}{\kappa_{\text{a}}\kappa_{\text{f}} + \kappa_D\chi_p^2\kappa_{\text{f}}} \quad (3.25)$$

with

$$\Gamma_0 \stackrel{\text{def}}{=} \mathbb{E}\left[f(X_0) - f_{\text{low}}\right] \quad (3.26)$$

and κ_{e} given by (3.21).

Proof. Let $j \in \{0, \ldots, k\}$. Summing (3.3) for all j, taking the full expectation and using the tower property, we derive that

$$\begin{aligned}\mathbb{E}\left[\sum_{j=0}^{k} \frac{\Sigma_j}{(p+1)!}\|S_j\|^{p+1}\right] \leq & \sum_{j=0}^{k} \mathbb{E}\left[f(X_j) - f(X_{j+1})\right] + \sum_{j=0}^{k} \kappa_{\text{a}} \mathbb{E}\left[\|S_j\|^{p+1}\right] \\ & + \sum_{j=0}^{k} \kappa_D \chi_p^2 \mathbb{E}\left[\|S_{j-1}\|^{p+1}\right].\end{aligned}$$

Using (2.11) to simplify the left-hand side, AS.2, the fact that $\|S_{-1}\| = 1$ and using (3.19) to bound both $\sum_{j=0}^{k} \mathbb{E}\left[\|S_j\|^{p+1}\right]$ and $\sum_{j=0}^{k-1} \mathbb{E}\left[\|S_j\|^{p+1}\right]$, we obtain that

$$\begin{aligned}\mathbb{E}\left[\frac{\Sigma_{k+1} - \Sigma_0}{(p+1)!}\right] \leq & \mathbb{E}\left[f(X_0)\right] - f_{\text{low}} + \kappa_{\text{a}}(\kappa_{\text{e}} + \kappa_{\text{f}} \log(\mathbb{E}\left[\Sigma_{k+1}\right]) \\ & + \kappa_D\chi_p^2(1 + \kappa_{\text{e}} + \kappa_{\text{f}}\log(\mathbb{E}\left[\Sigma_k\right])).\end{aligned}$$

Using now the definition of Γ_0 in (3.26), the fact that the Σ_j sequence is increasing, the last inequality gives that

$$\begin{aligned}\mathbb{E}\left[\frac{\Sigma_{k+1}}{(p+1)!}\right] \leq & \Gamma_0 + \frac{\Sigma_0}{(p+1)!} + \kappa_{\text{a}}(\kappa_{\text{e}} + \kappa_{\text{f}}\log(\mathbb{E}\left[\Sigma_{k+1}\right])) \\ & + \kappa_D\chi_p^2(1 + \kappa_{\text{e}} + \kappa_{\text{f}}\log(\mathbb{E}\left[\Sigma_{k+1}\right])).\end{aligned} \quad (3.27)$$

Now define

$$\gamma_1 \stackrel{\text{def}}{=} \kappa_{\text{a}}\kappa_{\text{f}} + \kappa_D\chi_p^2\kappa_{\text{f}}, \quad \gamma_2 \stackrel{\text{def}}{=} -\frac{1}{(p+1)!}, \quad u \stackrel{\text{def}}{=} \mathbb{E}\left[\Sigma_{k+1}\right], \quad (3.28)$$

$$\gamma_3 \stackrel{\text{def}}{=} \Gamma_0 + \frac{\Sigma_0}{(p+1)!} + \kappa_{\text{a}}\kappa_{\text{e}} + \kappa_D\chi_p^2(1+\kappa_{\text{e}}) \quad (3.29)$$

and observe that that $-3\gamma_2 < \gamma_1$ since $(p+1)!\kappa_{\text{a}} \geq L_p \geq 3$ and $\kappa_{\text{f}} \geq 1$. Define the function $\psi(t) \stackrel{\text{def}}{=} \gamma_1 \log(t) + \gamma_2 t + \gamma_3$. The inequality (3.27) can then be rewritten as

$$0 \leq \psi(u). \quad (3.30)$$

The constants γ_1, γ_2 and γ_3 satisfy the requirements of Lemma B.1 and ψ therefore admits two roots $\{u_1, u_2\}$ whose expressions are given in (B.2). Moreover, (3.30) is valid only for $u \in [u_1,\, u_2]$. Therefore, we obtain from (3.28), (3.29) and (B.2) that

$$\mathbb{E}\left[\Sigma_{k+1}\right] \le -(p+1)!\gamma_1 W_{-1}\left(\frac{-1}{(p+1)!\gamma_1} e^{-\frac{\gamma_3}{\gamma_1}}\right).$$

We then derive the desired result because the last inequality holds for all $k \ge 0$ and Σ_k is increasing. $\square$

We now discuss the bound obtained (3.24). First note that it is possible to give a more explicit bound on $\sigma_{\max}$ by finding an upper bound on the value of the involved Lambert function. This can be obtained by using [51, Theorem 1] which states that, for $x > 0$,

$$\left|W_{-1}(-e^{-x-1})\right| \le 1 + \sqrt{2x} + x. \tag{3.31}$$

Remembering that, for γ_1 and γ_2 given by (3.28), $\log\left((p+1)!(\kappa_{\rm a}\kappa_{\rm f} + \kappa_D \chi_p^2 \kappa_{\rm f})\right) \ge \log(3) > 1$ and taking $x = \kappa_{\rm g} - 1 + \log\left((p+1)!(\kappa_{\rm a}\kappa_{\rm f} + \kappa_D\chi_p^2\kappa_{\rm f})\right)$ in (3.31) then gives that

$$\begin{aligned}\left|W_{-1}\left(\frac{-1}{(p+1)!(\kappa_{\rm a}\kappa_{\rm f} + \kappa_D\chi_p^2\kappa_{\rm f})} e^{-\kappa_{\rm g}}\right)\right| &\le \kappa_{\rm g} + \log\left((p+1)!(\kappa_{\rm a}\kappa_{\rm f} + \kappa_D\chi_p^2\kappa_{\rm f})\right) \\ &\quad + \sqrt{2\left(\kappa_{\rm g} + \log\left((p+1)!(\kappa_{\rm a}\kappa_{\rm f} + \kappa_D\chi_p^2\kappa_{\rm f})\right) - 1\right)}.\end{aligned}$$

Complexity results for adaptive regularization methods typically bound the norm of the gradient at a specific iteration (see [20] for instance), but our differs in this respect and instead uses the structure of (2.11), property (3.7), and Lemma 3.7 to produce a bound involving the history of past gradients.

Theorem 3.8 Suppose that AS.1–AS.5 hold. Then

$$\min_{j\in\{0,\ldots,k\}} \mathbb{E}\left[\|G_{j+1}\|\right] \le (\kappa_{\rm c} + \kappa_{\rm d} m)^{\frac{p}{p+1}} \frac{\sigma_{\max}}{(k+1)^{\frac{p}{p+1}}} \tag{3.32}$$

where $\kappa_{\rm c}$ and $\kappa_{\rm d}$ are given by (3.8) and $\sigma_{\max}$ is given by (3.24).

Proof. Let $k \ge 1$ and $j \in \{0, \ldots, k\}$. Taking the full expectation in (3.7) and using the tower property, simplifying the upper-bound, using that $\sum_{j=0}^{k}\sum_{\ell=j-m}^{j-1} \Sigma_{j+1} - \Sigma_j \le m\Sigma_k$ from (2.11), and using Lemma 3.7, we derive that

$$\begin{aligned}\sum_{j=0}^{k}\mathbb{E}\left[\frac{\|G_{j+1}\|^{\frac{p+1}{p}}}{\Sigma_{j+1}^{\frac{1}{p}}}\right] &\le \kappa_{\rm c}\sum_{j=0}^{k}\mathbb{E}\left[\Sigma_{j+1} - \Sigma_j\right] + \kappa_{\rm d}\sum_{j=0}^{k}\sum_{\ell=j-m}^{j-1}\mathbb{E}\left[\Sigma_{\ell+1} - \Sigma_\ell\right] \\ &\le \kappa_{\rm c}\mathbb{E}\left[\Sigma_{k+1}\right] + \kappa_{\rm d} m\mathbb{E}\left[\Sigma_k\right] \\ &\le \left(\kappa_{\rm c} + \kappa_{\rm d} m\right)\sigma_{\max}.\end{aligned} \tag{3.33}$$

We now derive a lower-bound on the left-hand side of the last inequality. From the Hölder inequality with $q = \frac{p+1}{p}$ and $r = p+1$ and the fact that (3.24) holds, we obtain that

$$\mathbb{E}\left[\|G_{j+1}\|\right] = \mathbb{E}\left[\frac{\|G_{j+1}\|}{\Sigma_{j+1}^{\frac{1}{p+1}}}\Sigma_{j+1}^{\frac{1}{p+1}}\right] \leq \left(\mathbb{E}\left[\frac{\|G_{j+1}\|^{\frac{p+1}{p}}}{\Sigma_{j+1}^{\frac{1}{p}}}\right]\right)^{\frac{p}{p+1}} \mathbb{E}\left[\Sigma_{j+1}\right]^{\frac{1}{p+1}}$$
$$\leq \left(\mathbb{E}\left[\frac{\|G_{j+1}\|^{\frac{p+1}{p}}}{\Sigma_{j+1}^{\frac{1}{p}}}\right]\right)^{\frac{p}{p+1}} \sigma_{\max}^{\frac{1}{p+1}}. \tag{3.34}$$

Taking the last inequality to the power $\frac{p+1}{p}$ and using the result to find a lower bound on the left-hand side of (3.33) yields that

$$\frac{\min_{j\in\{0,\ldots,k\}} \mathbb{E}\left[\|G_{j+1}\|\right]^{\frac{p+1}{p}}(k+1)}{\sigma_{\max}^{\frac{1}{p}}} \leq \sum_{j=0}^{k} \frac{\mathbb{E}\left[\|G_{j+1}\|\right]^{\frac{p+1}{p}}}{\sigma_{\max}^{\frac{1}{p}}} \leq (\kappa_{\rm b} + \kappa_{\rm d} m)\,\sigma_{\max}.$$

Rearranging this last inequality and taking the $(\frac{p+1}{p})$-th root finally gives the desired result. □

The order of dependence on ϵ given by Theorem 3.8 is consistent with that presented in [20] for the deterministic adaptive regularization algorithm [20, 8], which has been shown to be optimal for pth order nonconvex optimization [22]. It is also consistent, from this point of view, with that proposed in [36] for the deterministic version of our algorithm. Theorem 3.8 however slightly improve on this latter result in another respect: because the present paper uses different and sharper bounding techniques, the dependence of $\sigma_{\max}$ on L_p in the constants of (3.32) is now $\mathcal{O}(L_p^{(2p+1)/p}\log(L_p))$, while that stated in [36] is $\mathcal{O}(L_p^{(3p+1)/p})$.

While the last theorem covers all model degrees, it is worthwhile to isolate the cases where p is either 1 or 2, detailing some of the constants hidden in (3.32). We start with $p = 1$.

Corollary 3.1 Suppose that AS.1–AS.3 and AS.5 hold and that $p = 1$. Then, the gradients of the iterates generated by Algorithm 2.1 verify

$$\min_{j\in\{0,\ldots,k\}} \mathbb{E}\left[\|G_{j+1}\|\right] \leq \sqrt{\left(4L_1^2 + 2\theta_1^2\sigma_0^2 + 2^{m+2}\kappa_D m\right)}\,\frac{\sigma_{\max}}{\sigma_0\sqrt{(k+1)}}$$

where $\sigma_{\max}$ is defined in (3.24).

Thus obtaining an iterate satisfying $\mathbb{E}\left[\|G_{k+1}\|\right] \leq \epsilon$, requires at most $\mathcal{O}\left(\epsilon^{-2}\right)$ iterations, achieving the complexity rate of linesearch steepest descent [27]. This result is not surprising, since our condition (2.13) for $p = 1$ is very similar to the strong growth condition [35]. Well-tuned stochastic gradient descent under appropriate conditions reaches the complexity rate of deterministic first-order methods under this condition. See [23] for more details on the theory of stochastic gradient descent for nonconvex functions.

For $p = 2$, Theorem 3.8 may be rephrased as follows.

Corollary 3.2 Suppose that AS.1–AS.5 hold and that $p = 2$. Then the gradients of the iterates generated by Algorithm 2.1 verify

$$\min_{j \in \{0,\dots,k\}} \mathbb{E}\left[\|G_{j+1}\|\right] \leq \sqrt[3]{2\left(\frac{L_2^{\frac{3}{2}}}{\sqrt{2}} + \frac{\sqrt{2}}{2} + \frac{\theta_1^{\frac{3}{2}}}{2^{\frac{3}{2}}} + 2^{m-1}(4+\sqrt{2})\kappa_D m\right)^2 \frac{\sigma_{\max}}{\sigma_0 (k+1)^{\frac{2}{3}}}}$$

where $\sigma_{\max}$ is defined in (3.24).

Again, if we are interested in reaching an iterate such that $\mathbb{E}\left[\|G_{k+1}\|\right] \leq \epsilon$, $\mathcal{O}\left(\epsilon^{-3/2}\right)$ iterations are required in the worst case, achieving the same rate as optimal second-order methods (see [27] and the references therein). As a consequence, our algorithm is an optimal adaptive cubic regularization method without function evaluation in a fully stochastic setting.

4 Applications of the StOFFARp algorithm

4.1 Inexact Derivatives

Our theory naturally applies to the case where derivatives are inexact. For the sake of clarity, we drop the uppercase notation and use only lowercase in this subsection. For this particular case, (2.13) holds without expectation for all iterations. Specifically, there exists $\kappa_D > 0$ such that the inaccurate derivatives $\overline{\nabla_x^i f}(x_k)$ used to compute the model (2.7) satisfy,

$$\|\nabla_x^i f(x_k) - \overline{\nabla_x^i f}(x_k)\| \leq \kappa_D \sum_{j=1}^{m} \|s_{k-j}\|^{p+1-i} \quad \text{for all} \quad i \in \{1,\dots,p\}. \tag{4.1}$$

The conditions here are very similar to those proposed in (2.15). Again, one of the advantages of (4.1) is that it considers the previous steps and not the current one, allowing (4.1) to be enforced at the beginning of each iteration. This approach formally covers the use of imprecise derivatives, where the approximation of high-order tensors is performed by using finite differences of low-order derivatives. For more details on these algorithmic variants, we refer the reader to [27, Subsection 13.2].

The inexact version of our algorithm also falls under the Explicit Dynamic Accuracy (EDA) framework [8, Section 13.3], since the conditions can be enforced a priori. The aforementioned settings are a hot topic, and algorithms have recently been proposed [42, 31]. These theoretical advances have arisen to take advantage of developments in large-scale modern computing hardware that allow loose numerical approximations of derivatives when needed. An imprecise version of our StOFFARp can be used in this context and may even offer a simpler alternative compared to current explicit dynamic accuracy adaptive regularization methods (see for example [27, Algorithm 13.3.3]).

4.2 Machine Learning Problems

In this subsection, we focus on the case where $p = 2$ as the results of this section are focused on practical machine learning problems. In the latter case (2.1) becomes

$$\min_{x \in \mathbb{R}^n} \{f(x) = \frac{1}{p} \sum_{i=1}^{p} f_i(x, y_i, a_i)\} \tag{4.2}$$

where both p and n (the space of the optimized variable) may exceed to millions and f_i may be nonconvex. The pairs (a_j, y_j) are independent and identically distributed random variables coming from an a priori unknown distribution $\mathcal{D}$. In this case, it is common to randomly sample batches of indices in the expression of f to approximate its derivatives. The sampled gradient and Hessian are therefore given by

$$\overline{\nabla_x^1 f}(X_k) = \frac{1}{b_{g,k}} \sum_{i \in \mathcal{B}_{g,k}} \nabla_x^1 f_i(X_k, y_i, a_i), \quad \text{and} \quad \overline{\nabla_x^2 f}(X_k) = \frac{1}{b_{H,k}} \sum_{i \in \mathcal{B}_{H,k}} \nabla_x^2 f_i(X_k, y_i, a_i), \tag{4.3}$$

where $\mathcal{B}_{g,k}$ and $\mathcal{B}_{H,k}$ are the batches at iteration k of cardinality $b_{g,k}$ and $b_{H,k}$, respectively. In our case, $\nabla_x^1 f_i(X_k, y_i, a_i)$ are i.i.d.[(1)] vector-valued random variables and $\nabla_x^2 f_i(X_k, y_i, a_i)$ are i.i.d. random self-adjoint matrices with dimension $n \times n$. To obtain lower bounds on batch sizes $b_{g,k}$ and $b_{H,k}$ of the stochastic gradient and Hessians (4.3), conditions in expectation on the noise of the gradient and Hessian of each f_i must be assumed. For clarity, we will drop the $\mathbb{E}_k[.]$ notation and keep only $\mathbb{E}[.]$ since we focus only on a specific iteration k. The goal of the next theorem is to provide requirements on $b_{g,k}$ and $b_{H,k}$ under assumptions that are common in the literature [21, 13] in order to satisfy (2.13).

Theorem 4.1 Let k be an iteration of the StOFFAR2 algorithm and suppose that the objective function has the structure given in (4.2) and that for each $i \in \{1, \ldots, p\}$, there exist non-negative constants σ_g and σ_H such that

$$\mathbb{E}\left[\|\nabla_x^1 f_i(X_k, y_i, a_i) - \nabla_x^1 f(X_k)\|^2\right] \le \sigma_g^2 \quad \text{and} \quad \mathbb{E}\left[\|\nabla_x^2 f_i(X_k, y_i, a_i) - \nabla_x^2 f(X_k)\|^3\right] \le \sigma_H^3. \tag{4.4}$$

Then the estimators introduced in (4.3) for problem (4.2) verify conditions (2.13) if

$$b_{g,k} \ge \frac{\sigma_g^2}{\kappa_D^{\frac{4}{3}} \xi_k^{\frac{4}{3}}}, \tag{4.5}$$

and

$$b_{H,k} \ge \frac{9\sigma_H^2 e \log(n)}{2\kappa_D^{\frac{2}{3}} \xi_k^{\frac{2}{3}}}. \tag{4.6}$$

Proof. As the proof combines elements already developed in [21, 13] but adapted to take into account (2.13), it is differed to Appendix C. □

---

(1) independent identically distributed

Before proceeding, we discuss our proposed sampling conditions and provide a discussion when $m = 1$ and $\xi_k = \|S_{k-1}\|^3$. First, note that we have obtained the same order of dependence on the step size as in the work of [24]. Our framework improves on this reference because we have not imposed Lipschitz continuity on f_i or its derivatives. Moreover, our condition covers the use of the previous step to scale the batch-sizes, whereas the theoretical result developed in [24] uses the current step size. Finally, it should be noted that our framework is more flexible than previous works [24, 37] in that it allows the error to depend on the past m steps, rather than just a specific one.

5 Numerical illustration

In this section, we illustrate the numerical behaviour of our proposed StOFFARp algorithm for $p = 1$ and $p = 2$ for the machine-learning problems discussed in Subsection 4.2. The goal of the following experiments is to demonstrate the advantages of high-order objective-free function algorithms for machine-learning problems. We perform numerical tests on two different formulations of the binary classification problem. Throughout this section, $\{a_i, y_i\}_{i=1}^p$ represents the training data with $a_i \in \mathbb{R}^n$ and $y_i \in \{0, 1\}$ representing the ith feature and the ith target label, respectively. For the binary classification, we propose the following formulation as a minimization task:

$$\min_{x \in \mathbb{R}^n} f(x) = \min_{x \in \mathbb{R}^n} \frac{1}{p} \sum_{i=1}^{p} f_i(x, y_i, a_i) = \min_{x \in \mathbb{R}^n} \frac{1}{p} \sum_{i=1}^{p} \left(y_i - \phi(a_i^\intercal x)\right)^2, \tag{5.1}$$

where

$$\phi(a^\intercal x) = \frac{1}{1 + e^{-a^\intercal x}}. \tag{5.2}$$

This minimization problem has already been considered in [37, 43]. We refer the reader to these references for the expressions of both the gradient and the Hessian. We also consider a second case of nonconvex binary classification studied in [24, 23], where a standard binary logistic regression is regularized with a nonconvex term. The binary classification problem is then formalized as:

$$\min_{x \in \mathbb{R}^n} f(x) = \min_{x \in \mathbb{R}^n} \frac{1}{p} \sum_{i=1}^{p} \left(-y_i \log(\phi(a_i^\intercal x)) - (1 - y_i) \log(1 - \phi(a_i^\intercal x)) + \alpha \sum_{j=1}^{n} \frac{x_j^2}{1 + x_j^2} \right), \tag{5.3}$$

where α is a parameter that regulates the strength of the penalization. The rest of this section is organized as follows. Implementation issues are considered in Subsection 5.1. To satisfy AS.5, we run a variant of our StOFFARp with $p = 2$ and various m values, denoted `OFFAR2-m`, that implements a sampling strategy using the scaling rules given in Theorem 4.1. As a baseline, we use a StOFFARp with $p = 1$ and $m = 1$, denoted `WNGRAD`, since our algorithm retrieves the method proposed in [2]. As in Theorem 4.1, we also derive a condition on the sample size to satisfy AS.5 for this method. We have avoided comparison with other second-order stochastic algorithms, such as those proposed in [43, 37, 24, 12], since they either require access to the exact value of the function to adjust the regularization parameter σ_k, or assume knowledge of the Lipschitz constant. Some illustrations of both methods are provided in Subsection 5.2.

5.1 Implementation Issues

Our implementation relies on the code provided in [24][(2)]. The subsampled cubic regularization subroutine is slightly adapted to allow the use of the update rule given in (2.11), to fulfill the condition given in (2.9) when computing the step, and to subsample in accordance with the conditions of Theorem 4.1. Specifically, at the initial iteration of the `OFFAR2-m` algorithm, the values of $b_{h,0}$ and $b_{g,0}$ are set to $0.05 \cdot p$ and $0.20 \cdot p$ in order to compute the approximate Hessian and gradient, as defined in (4.3). Note that this choice of initial subsampling size is consistent with past subsampled methods developed in the literature [24, 37]. For $k \geq 1$, we using the following subsampling strategy:

$$b_{g,k} = \max\left(\frac{c_g}{\xi_k^{\frac{4}{3}}}, 0.20 \cdot p\right), \quad b_{H,k} = \max\left(\frac{c_H}{\xi_k^{\frac{2}{3}}}, 0.05 \cdot p\right), \tag{5.4}$$

where $c_g = b_{g,0} m^{\frac{4}{3}}$ and $c_H = \frac{b_{h,0} m^{\frac{2}{3}}}{\log(n)}$ and m is defined in (2.14). The choices of the constants c_g and c_H are made to ensure that our first subsampled derivatives verify (4.5) and (4.6) with κ_D chosen as $\frac{\sigma_g^{\frac{3}{2}}}{b_{g,0}^{\frac{3}{4}} m}$ for gradient subsampling and $\frac{(9e)^{\frac{3}{2}} (\log(n)\sigma_H)^{\frac{1}{3}}}{(2b_{H,0})^{\frac{3}{2}} m}$ for the Hessian subsampling. We also chose $\sigma_0 = 0.01$ and $\theta_1 = 2$ and ran four variants with $\mathtt{m} \in \{1, 50, 250, 500\}$.

We also developed our own implementation of the `WNGRAD` algorithm where we use an initial batch size of $b_{g,0} = 0.05 \cdot p$, $m = 1$, and subsample for $k \geq 1$ with

$$b_{g,k} = \max\left(0.05\, p, \frac{0.1}{\|S_{k-1}\|^2}\right). \tag{5.5}$$

We also choose $\sigma_0 = 0.1$ for `WNGRAD`. Both methods start from an initial point $x_0 = (0,\, 0,\, \ldots, 0)$ and α in (5.3) is taken equal to 0.001.

The algorithms are stopped when an iterate x_k satisfying

$$\|\overline{\nabla_x^1 f}(x_k)\| \leq \epsilon \quad \text{with} \quad \epsilon = 0.0005 \tag{5.6}$$

is reached. The maximum number of iterations for both `OFFAR2-m` and `WNGRAD` is set to 1000 and 10000, respectively. The datasets are taken from the LIBSVM library [16] (see Appendix D for more detail).

5.2 Results

To evaluate the performance of our methods that involve stochastic ingredients (resulting from approximation by subsampling), all reported results are averages over 20 independent runs. To provide an appropriate comparison between the tested methods which may employ different batch sizes, we report the performance measure

$$\tau_{algo} = \sum_{i=1}^{k} (b_{g,i} + b_{H,i}) \cdot \mathrm{ege}_i, \tag{5.7}$$

where ege_i, the effective gradient evaluation metric, counts the number of Hessian-vector products used at each iteration to compute the step for the `OFFAR2-m` methods in addition to

(2) Available at https://github.com/dalab/subsampled_cubic_regularization.

the number of gradient evaluations. For the `WNGRAD` algorithm, the value of ege_i is equal to one, while the value of $b_{H,i}$ is equal to zero.

Figure 1 shows the standard performance profile [14] for the five methods with respect to the performance measure (5.7). The figure illustrates that our proposed second-order method identifies an approximate first-order stationary point more rapidly than simple adaptive gradient methods represented here by `WNGRAD`. It is evident that the `OFFAR2-50` method is the most efficient. `OFFAR2-1` is the second most efficient, but it may perform less effectively than the other methods on some problems. To illustrate this point, consider Figure 2, in which $f^\star$ denotes the best (i.e., minimum) value obtained among all the four tested methods, and where the number of samples is reported for each method. The two problems considered here, `SUSY` and `w8A`, illustrate cases where the performance of the methods with longer memory ($\mathtt{m} \in \{250, 500\}$) is superior to that of the methods with shorter memory ($\mathtt{m} \in \{1, 50\}$), both in terms of convergence and the number of samples used by the methods.

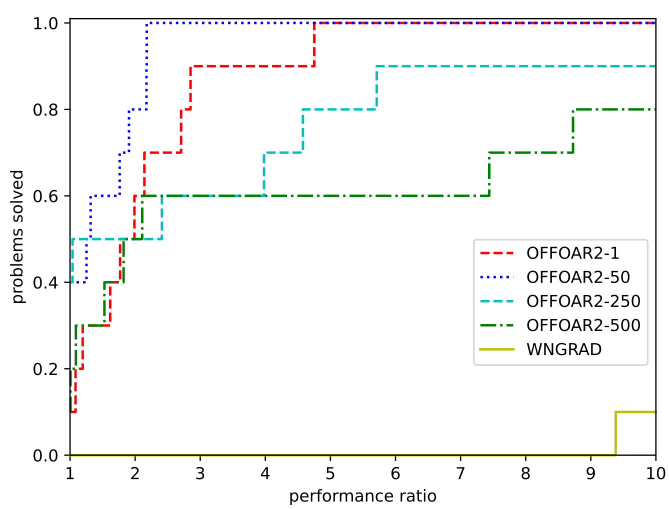

Figure 1: Performance profile of `OFFAR2-m` for `SUSY` and `w8a` for $\mathtt{m} \in \{1, 50, 250, 500\}$ and `WNGRAD`

We remind the reader that one epoch denotes a pass made on the whole data set (samples f_i in (4.2)) when computing the stochastic gradient and the Hessian.

We also use Figure 2 to exemplify a generic problem occuring when using short memory and single-step length control: the resulting method may become exact (and therefore computationally expensive) after only a few iterations, as the required samples involve the entire data set. This (undesirable) behavior is also observed in many methods, including the subsampled cubic regularization method presented in [24]. Other subsampled second-order methods impose a tight probabilistic bound on all iterations (as shown in [29, 5, 10]), which again causes the algorithm to be deterministic for most iterations. The same drawback also appears in some methods that impose the growth of the batch size and the use of the exact Hessian and gradient starting from a specific iteration [50]. In contrast, `OFFAR2-m` methods with long memory allow the error bounds to remain large, resulting in more aggressive sampling until termination. It is worth noting that, empirically, `OFFAR2-m` with large `m` reaches local minima with a lower objective value. Longer memory may however occasionally may

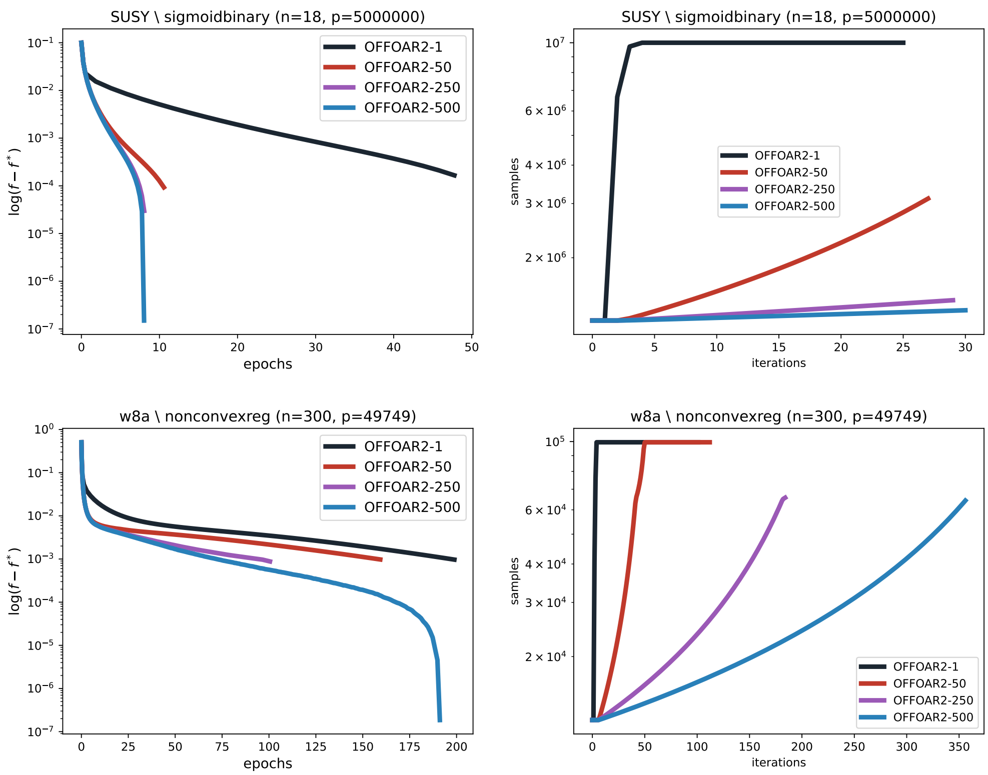

Figure 2: Evolution of the loss function w.r.t the epochs and number of samples along iterations for `SUSY` and `w8a`

result in slower convergence in practice (as illustrated by Figure 1), and satisfying the criteria (5.6) may become costly. The reader is referred to the examples shown in Appendix E to understand some of the problems that arise when using a large `m`.

These early results suggest that using high-order OFFO algorithms may be beneficial, but the authors are aware that additional numerical experiments are required to better assess their potential. Indeed, refinements on the update rule of the regularization parameter have been proposed in OFFO second-order methods, be it trust-region [47] or adaptive regularization [36], and a thorough analysis of the influence of the values of $\|S_{-1}\|, \ldots, \|S_{-m}\|$ and the length of the memory `m` may be required. We have avoided their discussion here to keep our analysis and numerical experiments concise. Their addition may require a more involved proof and may impose stronger assumptions.

6 Discussion

In this paper, we have developed a fully stochastic theory for an objective-function-free adaptive regularization algorithm described in [36]. Since the algorithm does not use the function value to accept or reject the step, it avoids the need to compute this value with an accuracy higher than that used for the gradient, thereby making it a computationally attractive technique for noisy problems. The new algorithm introduces novel conditions on the probabilistic tensor derivatives, and uses the history of past steps to determine the level of derivatives' accuracy which is acceptable in expectation to ensure convergence. Our analysis shows that its evaluation complexity is optimal in order.

We also discussed two application cases. The first focuses on noisy inexact functions, where inaccuracy arises from lower precision computations or the use of finite differences. The second case is finite-sum minimization (typical of machine learning problems), where we provide sample size conditions to meet the specified requirements under mild assumptions. Applying the algorithm to practical binary classification problems highlighted the advantage of second-order OFFO methods over standard adaptive gradient strategies and also showed that the proposed sampling scheme can remain practical throughout the computation.

Unsurprisingly, an extension of the algorithm to guarantee termination at approximate second-order stationary points is possible, in the vein of what was proposedd in [36, Section 4] for the deterministic case. The analysis would be very similar to that of Section 3, replacing $\|G_k\|$ by the appropriate measure of criticality.

One possible further improvement is to study the OFFO algorithm under the assumption that

$$\mathbb{E}_k\left[\|\nabla_x^i f(X_k) - \overline{\nabla_x^i f}(X_k)\|^{\frac{p+1}{p+1-i}}\right] \leq \kappa_D \|S_{k-1}\|^{p+1} + \kappa_c, \quad \text{for all } i \in \{1,\ldots,p\}.$$

An assumption of this nature has been considered in the analysis of adaptive gradient methods [3], and extending it to higher-order OFFO schemes seems a natural line for future research. A second line may focus on proposing OFFO schemes that incorporate momentum when updating the regularization parameter.

References

[1] Xiaoyu Li and Francesco Orabona. On the convergence of stochastic gradient descent with adaptive stepsizes. In *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, pages 983–992. PMLR, 2019.

[2] Xiaoxia Wu, Rachel Ward, and Léon Bottou. Wngrad: Learn the learning rate in gradient descent. arXiv:1803.02865, 2018.

[3] M. Faw, I. Tziotis, C. Caramanis, A. Mokhtari, S. Shakkottai, and R. Ward. The power of adaptivity in SGD: Self-tuning step sizes with unbounded gradients and affine variance. In *Proceedings of 35th Conference on Learning Theory*, volume 178, pages 313–355, 2022.

[4] Chee Keng Yap. *Fundamental Problems of Algorithmic Algebra*. Oxford University Press, Inc., USA, 1999.

[5] Zhewei Yao, Peng Xu, Fred Roosta, and Michael W. Mahoney. Inexact nonconvex newton-type methods. *INFORMS Journal on Optimization*, 3(2):154–182, January 2021.

[6] R. Ward, X. Wu, and L. Bottou. AdaGrad stepsizes: Sharp convergence over nonconvex landscapes. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pages 6677–6686, 2019.

[7] D. Zhou, J. Chen, Y. Tang, Z. Yang, Y. Cao, and Q. Gu. On the convergence of adaptive gradient methods for nonconvex optimization. arXiv:2080.05671, 2020.

[8] 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.

[9] R. Chen, M. Menickelly, and K. Scheinberg. Stochastic optimization using a trust-region method and random models. *Mathematical Programming*, 169(2):447–487.

[10] Xi Chen, Bo Jiang, Tianyi Lin, and Shuzhong Zhang. Accelerating adaptive cubic regularization of newton's method via random sampling. *Journal of Machine Learning Research*, 23(90):1–38, 2022.

[11] Zhe Wang, Yi Zhou, Yingbin Liang, and Guanghui Lan. A note on inexact gradient and hessian conditions for cubic regularized newton's method. *Operations Research Letters*, 47(2):146–149, 2019.

[12] Zhe Wang, Yi Zhou, Yingbin Liang, and Guanghui Lan. Stochastic variance-reduced cubic regularization for nonconvex optimization. In *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, pages 2731–2740, 2019.

[13] Dongruo Zhou, Pan Xu, and Quanquan Gu. Stochastic variance-reduced cubic regularization methods. *Journal of Machine Learning Research*, 20(134):1–47, 2019.

[14] E. D. Dolan, J. J. Moré, and T. S. Munson. Optimality measures for performance profiles. *SIAM Journal on Optimization*, 16(3):891–909, 2006.

[15] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(61):2121–2159, 2011.

[16] Chih-Chung Chang and Chih-Jen Lin. Libsvm: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3):1–27, 2011.

[17] Junyu Zhang, Lin Xiao, and Shuzhong Zhang. Adaptive stochastic variance reduction for subsampled newton method with cubic regularization. *INFORMS Journal on Optimization*, 4(1):45–64, 2022.

[18] Sashank J. Reddi, Satyen Kale, and Sanjiv Kumar. On the convergence of Adam and beyond. In *6th International Conference on Learning Representations, ICLR 2018*, 2018.

[19] C. Cartis and K. Scheinberg. Global convergence rate analysis of unconstrained optimization methods based on probabilistic models. *Mathematical Programming*, 169(2):337–375, April 2017.

[20] E. G. Birgin, J. L. Gardenghi, J. M. Martínez, S. A. Santos, and Ph. L. Toint. Worst-case evaluation complexity for unconstrained nonlinear optimization using high-order regularized models. *Mathematical Programming*, 163(1-2):359–368, 2016.

[21] El Mahdi Chayti, Nikita Doikov, and Martin Jaggi. Unified convergence theory of stochastic and variance-reduced cubic newton methods. arXiv:2302.11962, 2023.

[22] Yair Carmon, John C. Duchi, Oliver Hinder, and Aaron Sidford. Lower bounds for finding stationary points i. *Mathematical Programming*, 184(1-2):71–120, 2019.

[23] Ahmed Khaled and Peter Richtárik. Better theory for SGD in the nonconvex world. *Transactions on Machine Learning Research*, 2023.

[24] Jonas Moritz Kohler and Aurelien Lucchi. Sub-sampled cubic regularization for non-convex optimization. In *Proceedings of the 34th International Conference on Machine Learning*, pages 1895–1904. PMLR, 2017.

[25] Coralia Cartis, Nicholas I. M. Gould, and Philippe 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.

[26] Coralia Cartis, Nicholas I. M. Gould, and Philippe L. Toint. Adaptive cubic regularisation methods for unconstrained optimization. part II: worst-case function- and derivative-evaluation complexity. *Math. Program.*, 130(2):295–319, 2011.

[27] Coralia Cartis, Nicholas I M Gould, and Philippe L Toint. *Evaluation complexity of algorithms for nonconvex optimization*. Society for Industrial & Applied Mathematics, New York, NY, 2022.

[28] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015.

[29] Farbod Roosta-Khorasani and Michael W. Mahoney. Sub-sampled newton methods. *Mathematical Programming*, 174(1–2):293–326, 2018.

[30] S. Gratton, S. Jerad, and Ph. L. Toint. Complexity of a class of first-order objective-function-free optimization algorithms. *Optimization Methods and Software*, page 1–31, 2024.

[31] S. Gratton and Ph. L. Toint. A note on solving nonlinear optimization problems in variable precision. *Computational Optimization and Applications*, 76(3):917–933, 2020.

[32] S Gratton and Ph L Toint. Adaptive regularization minimization algorithms with nonsmooth norms. *IMA Journal of Numerical Analysis*, 03 2022. drac005.

[33] 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, January 2021.

[34] R. M. Corless, G. H. Gonnet, D. E. G. Hare, D. J. Jeffrey, and D. E. Knuth. On the LambertW function. *Advances in Computational Mathematics*, 5(1):329–359, 1996.

[35] Mark Schmidt and Nicolas Le Roux. Fast convergence of stochastic gradient descent under a strong growth condition, 2013.

[36] Serge Gratton, Sadok Jerad, and Philippe 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, 2023.

[37] S. Bellavia and G. Gurioli. Stochastic analysis of an adaptive cubic regularization method under inexact gradient evaluations and dynamic hessian accuracy. *Optimization*, 71(1):227–261.

[38] 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, 2022.

[39] S. Bellavia, G. Gurioli, B. Morini, and Ph.L. Toint. A stochastic arc method with inexact function and random derivatives evaluations. In *Thirty-seventh International Conference on Machine Learning: ICML2020*, 2020.

[40] Jose Blanchet, Coralia Cartis, Matt Menickelly, and Katya Scheinberg. Convergence rate analysis of a stochastic trust-region method via supermartingales. *INFORMS Journal on Optimization*, 1(2):92–119, 2019.

[41] Artem Agafonov, Dmitry Kamzolov, Pavel Dvurechensky, Alexander Gasnikov, and Martin Takáč. Inexact tensor methods and their application to stochastic convex optimization. *Optimization Methods and Software*, page 1–42, November 2023.

[42] Stefania Bellavia, Serge Gratton, and Elisa Riccietti. A levenberg–marquardt method for large nonlinear least-squares problems with dynamic accuracy in functions and gradients. *Numerische Mathematik*, 140(3):791–825, June 2018.

[43] Stefania Bellavia, Gianmarco Gurioli, and Benedetta Morini. Adaptive cubic regularization methods with dynamic inexact Hessian information and applications to finite-sum minimization. *IMA Journal of Numerical Analysis*, 41(1):764–799, 2020.

[44] Stefania Bellavia, Gianmarco Gurioli, Benedetta Morini, and Philippe L. Toint. Adaptive regularization algorithms with inexact evaluations for nonconvex optimization. *SIAM Journal on Optimization*, 29(4):2881–2915, January 2019.

[45] Courtney Paquette and Katya Scheinberg. A stochastic line search method with expected complexity analysis. *SIAM Journal on Optimization*, 30(1):349–376, 2020.

[46] Alexandre Défossez, Leon Bottou, Francis Bach, and Nicolas Usunier. A simple convergence proof of Adam and Adagrad. *Transactions on Machine Learning Research*, 2022.

[47] Geovani N. Grapiglia and Gabriel F. D. Stella. An adaptive trust-region method without function evaluations. *Computational Optimization and Applications*, 82(1):31–60, 2022.

[48] Katya Scheinberg and Miaolan Xie. Stochastic Adaptive Regularization Method with Cubics: A High Probability Complexity Bound. In *OPT2022: 14th Annual Workshop on Optimization for Machine Learning*, 2022.

[49] Nilesh Tripuraneni, Mitchell Stern, Chi Jin, Jeffrey Regier, and Michael I. Jordan. Stochastic cubic regularization for fast nonconvex optimization. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, page 2904–2913, 2018.

[50] Raghu Bollapragada, Richard H Byrd, and Jorge Nocedal. Exact and inexact subsampled newton methods for optimization. *IMA Journal of Numerical Analysis*, 39(2):545–578, 2018.

[51] I. Chatzigeorgiou. Bounds on the Lambert function and their application to the outage analysis of user cooperation. *IEEE Communications Letters*, 17(8):1505–1508, 2013.

A Proof of (3.10)

Proof. In the following, we use lowercase notation as the Lemma 3.3 is valid for all iterations and all realizations. If $p = 1$, we obtain from (2.8) and the Cauchy-Schwartz inequality that

$$\tfrac{1}{2}\sigma_k\|s_k\|^2 < -\overline{g}_k^\intercal s_k \leq \|\overline{g}_k\|\,\|s_k\|$$

and (3.10) holds with $\eta = 0$. Suppose now that $p > 1$. (2.8) gives that

$$\frac{\sigma_k}{(p+1)!}\|s_k\|^{p+1} \leq -\overline{g}_k^\intercal s_k - \sum_{i=2}^{p} \frac{1}{i!}\overline{\nabla_x^i f}(x_k)[s_k]^i \leq \|\overline{g}_k\|\|s_k\| + \sum_{i=2}^{p} \frac{\kappa_{\rm high}}{i!}\|s_k\|^i,$$

where we applied AS.4 to obtain the last inequality.

Applying now the Lagrange bound for polynomial roots [4, Lecture VI, Lemma 5] with $x = \|s_k\|$, $n = p+1$, $a_0 = 0$, $a_1 = \|\overline{g_k}\|$, $a_i = \kappa_{\rm high}/i!$ $i \in \{2,\ldots,p\}$ and $a_{p+1} = \sigma_k/(p+1)!$, we obtain from (2.8) that the equation $\sum_{i=0}^n a_i x^i = 0$ admits at least one strictly positive root, and we may thus derive that

$$\|s_k\| \leq 2\max\left(\left(\frac{(p+1)!\|\overline{g}_k\|}{\sigma_k}\right)^{\frac{1}{p}}, \left\{\left[\frac{\kappa_{\rm high}(p+1)!}{i!\sigma_k}\right]^{\frac{1}{p-i+1}}\right\}_{i\in\{2,\ldots,p\}}\right).$$

Using now the fact that $\sigma_k \geq \sigma_0$ and the definition of η in (3.11) yields (3.10). □

B Solutions of the equation $\gamma_1 \log(u) + \gamma_2 u + \gamma_3 = 0$

Lemma B.1 Let $(\gamma_1, \gamma_2, \gamma_3) \in \mathbb{R}_+^\star \times \mathbb{R}_-^\star \times \mathbb{R}^+$ and $\frac{\gamma_2}{\gamma_1} \geq -\frac{1}{3}$. Then the equation

$$\gamma_1 \log(u) + \gamma_2 u + \gamma_3 = 0 \tag{B.1}$$

admits two solutions $0 < u_1 < u_2$ given by

$$u_1 = \frac{\gamma_1}{\gamma_2}W_0\left(\frac{\gamma_2}{\gamma_1}e^{\frac{-\gamma_3}{\gamma_1}}\right) \quad\text{and}\quad u_2 = \frac{\gamma_1}{\gamma_2}W_{-1}\left(\frac{\gamma_2}{\gamma_1}e^{\frac{-\gamma_3}{\gamma_1}}\right), \tag{B.2}$$

where W_0 and W_{-1} are the two branches of the Lambert function [34].

Proof. Note that since $e^{\frac{-\gamma_3}{\gamma_1}} \leq 1$ and $-\frac{1}{3} \leq \frac{\gamma_2}{\gamma_1} < 0$, we obtain that

$$-\frac{1}{3} \leq \frac{\gamma_2}{\gamma_1}e^{\frac{-\gamma_3}{\gamma_1}} < 0. \tag{B.3}$$

Let u be a solution of (B.1). Rearranging the equality (B.1) and taking the exponential yields that

$$u = e^{\frac{-\gamma_3}{\gamma_1} - \frac{\gamma_2}{\gamma_1}u}$$

and thus that

$$\frac{\gamma_2}{\gamma_1}ue^{\frac{\gamma_2}{\gamma_1}u} = \frac{\gamma_2}{\gamma_1}e^{\frac{-\gamma_3}{\gamma_1}}.$$

Taking $w \stackrel{\rm def}{=} \frac{\gamma_2}{\gamma_2}u$ and using (B.3), we obtain that the equation

$$we^w = \frac{\gamma_2}{\gamma_1}e^{\frac{-\gamma_3}{\gamma_1}}$$

admits two distinct solutions w_1 and w_2 given by

$$w_1 = W_0\left(\frac{\gamma_2}{\gamma_1}e^{\frac{-\gamma_3}{\gamma_1}}\right), \qquad w_2 = W_{-1}\left(\frac{\gamma_2}{\gamma_1}e^{\frac{-\gamma_3}{\gamma_1}}\right) \quad \text{and} \quad w_2 < w_1 < 0.$$

The desired result then follows from the facts that $u = \frac{\gamma_1}{\gamma_2}w$ and that $\frac{\gamma_1}{\gamma_2} < 0$. $\square$

C Proof of Theorem 4.1

Before proving Theorem 4.1, we need the two following auxiliary lemmas that we state below.

Lemma C.1 Suppose that $z_1, z_2, \dots, z_N$ are i.i.d vector valued random variables with $\mathbb{E}[z_i] = 0$ and $\mathbb{E}\left[\|z_i\|^2\right] < +\infty$. Then

$$\mathbb{E}\left[\left\|\frac{1}{N}\sum_{i=1}^N z_i\right\|^{\frac{3}{2}}\right] \le \frac{1}{N^{\frac{3}{4}}}\left(\mathbb{E}\left[\|z_i\|^2\right]\right)^{\frac{3}{4}}.$$

Proof. See [13, Lemma 31] for the statement of the lemma and Appendix C of this reference for its proof. $\square$

Lemma C.2 Suppose that $q \ge 2$, $n \ge 2$, and fix $r \ge \max(q, 2\log n)$. Consider i.i.d. random self-adjoint matrices $Y_1, \dots, Y_N$ with dimension $n \times n$, $\mathbb{E}[Y_i] = 0$. Then

$$\mathbb{E}\left[\left\|\sum_{i=1}^N Y_i\right\|^q\right]^{\frac{1}{q}} \le 2\sqrt{er}\left\|\left(\sum_{i=1}^N \mathbb{E}\left[Y_i^2\right]\right)^{\frac{1}{2}}\right\| + 4er\left(\mathbb{E}\left[\max_i \|Y_i\|^q\right]\right)^{\frac{1}{q}}.$$

Proof. As for the previous lemma, see [13, Lemma 32] for the statement of the Lemma and its proof. $\square$

We are now in a position to provide the proof of the statement of Theorem 4.1.

Proof. We start by providing a proof on $b_{g,k}$. First, denote $g_{i,k} \stackrel{\text{def}}{=} \nabla_x^1 f_i(X_k, y_i, a_i)$ for $i \in \mathcal{B}_{g,k}$, so that (4.3) and (4.2) give that

$$\mathbb{E}[g_{i,k}] = \nabla_x^1 f(X_k) \quad \text{and} \quad \overline{\nabla_x^1 f}(X_k) = \frac{1}{b_{g,k}}\sum_{i\in\mathcal{B}_{g,k}} g_{i,k}. \tag{C.1}$$

Applying now Lemma C.1 with $z_i = \frac{g_{i,k} - \nabla_x^1 f(X_k)}{b_{g,k}}$ for $i \in \mathcal{B}_{g,k}$ and using the first part of

(4.4), we derive that

$$\mathbb{E}\left[\left\|\frac{1}{b_{g,k}}\sum_{i\in\mathcal{B}_{g,k}} g_{i,k} - \nabla_x^1 f(X_k)\right\|^{\frac{3}{2}}\right] \le \frac{\sigma_g^{\frac{3}{2}}}{b_{g,k}^{\frac{3}{4}}},$$

and so if $b_{g,k}$ is taken as in (4.5), (2.13) holds for $i=1$ and $p=2$.

Again, as for the gradient, we denote $H_{i,k} \stackrel{\rm def}{=} \nabla_x^1 f_i(X_k, y_i, a_i)$., and thus

$$\mathbb{E}\left[H_{i,k}\right] = \nabla_x^2 f(X_k) \quad \text{and} \quad \overline{\nabla_x^2 f}(X_k) = \frac{1}{b_{H,k}}\sum_{i\in\mathcal{B}_{H,k}} H_{i,k}. \tag{C.2}$$

Also note that (4.4) and Jensen's inequality imply that

$$\mathbb{E}\left[\|H_{i,k} - \nabla_x^2 f(X_k)\|^2\right] \le \left(\mathbb{E}\left[\|H_{i,k} - \nabla_x^2 f(X_k)\|^3\right]\right)^{\frac{3}{2}} \le \sigma_H^2. \tag{C.3}$$

Applying now Lemma C.2 with $q=3$, $r=2\log(n)$, $N=b_{H,k}$ and $Y_i = \frac{H_{i,k}-\nabla_x^2 f(X_k)}{b_{H,k}}$, we obtain that

$$\begin{aligned}\mathbb{E}\left[\left\|\frac{1}{b_{H,k}}\sum_{i\in\mathcal{B}_{H,k}} H_{i,k} - \nabla_x^2 f(X_k)\right\|^3\right] \le \Bigg(2\sqrt{2e\log(n)}&\left\|\left(\sum_{i\in\mathcal{B}_{H,k}}\frac{1}{b_{H,k}^2}\mathbb{E}\left[\left(H_{i,k}-\nabla_x^2 f(X_k)\right)^2\right]\right)^{\frac{1}{2}}\right\| \\ &+ \frac{8e\log(n)}{b_{H,k}}\left(\mathbb{E}\left[\max_{i\in\mathcal{B}_{H,k}}\|H_{i,k}-\nabla_x^2 f(X_k)\|^3\right]\right)^{\frac{1}{3}}\Bigg)^3.\end{aligned} \tag{C.4}$$

Let us now establish a bound on $\left\|\left(\sum_{i\in\mathcal{B}_{H,k}}\frac{1}{b_{H,k}^2}\mathbb{E}\left[\left(H_{i,k}-\nabla_x^2 f(X_k)\right)^2\right]\right)^{\frac{1}{2}}\right\|$. Successively using the fact that $\|A^{\frac{1}{2}}\| = \|A\|^{\frac{1}{2}}$ for any positive definite matrix A, the Jensen's inequality, that $\|B^2\| = \|B\|^2$ for any symmetric matrix B, and (C.3), we derive that

$$\begin{aligned}\left\|\left(\sum_{i\in\mathcal{B}_{H,k}}\frac{1}{b_{H,k}^2}\mathbb{E}\left[\left(H_{i,k}-\nabla_x^2 f(X_k)\right)^2\right]\right)^{\frac{1}{2}}\right\| &= \left\|\sum_{i\in\mathcal{B}_{H,k}}\frac{1}{b_{H,k}^2}\mathbb{E}\left[\left(H_{i,k}-\nabla_x^2 f(X_k)\right)^2\right]\right\|^{\frac{1}{2}} \\ &= \left\|\frac{1}{b_{H,k}}\mathbb{E}\left[\left(H_{i,k}-\nabla_x^2 f(X_k)\right)^2\right]\right\|^{\frac{1}{2}} \\ &\le \frac{1}{\sqrt{b_{H,k}}}\mathbb{E}\left[\left\|\left(H_{i,k}-\nabla_x^2 f(X_k)\right)^2\right\|\right]^{\frac{1}{2}} \\ &= \frac{1}{\sqrt{b_{H,k}}}\mathbb{E}\left[\left\|H_{i,k}-\nabla_x^2 f(X_k)\right\|^2\right]^{\frac{1}{2}} \le \frac{\sigma_H}{\sqrt{b_{H,k}}}.\end{aligned}$$

Now injecting the last inequality and (4.4) in (C.4) yields that

$$\mathbb{E}\left[\left\|\frac{1}{b_{H,k}}\sum_{i\in\mathcal{B}_{H,k}} H_{i,k} - \nabla_x^2 f(X_k)\right\|^3\right] \le \left(2\sigma_H\sqrt{\frac{2e\log(n)}{b_{H,k}}} + \frac{8e\log(n)\sigma_H}{b_{H,k}}\right)^3.$$

Imposing the left-hand side of the previous inequality to be less than $\kappa_D \xi_k$ and using the concavity of the square root function then yields that

$$\frac{1}{\sqrt{b_{H,k}}} \le \frac{\sqrt{2e\log(n) + 8e\log(n)\kappa_D^{\frac{1}{3}}\frac{\xi_k^{\frac{1}{3}}}{\sigma_H}} - \sqrt{2e\log(n)}}{8e\log(n)}$$
$$\le \frac{8e\log(n)\kappa_D^{\frac{1}{3}}\xi_k^{\frac{1}{3}}}{12e\sigma_H\log(n)\sqrt{2e\log(n)}} = \frac{2\kappa_D^{\frac{1}{3}}\xi_k^{\frac{1}{3}}}{3\sigma_H\sqrt{2e\log(n)}}.$$

Rearranging the last inequality gives the bound (4.6). □

D Considered Datasets

Dataset	Samples	Features
a9a	32561	123
ijcnn1	49990	22
w8a	49749	300
SUSY	5000000	18
HIGGS	11000000	28

Table 1: Datasets characterization, source: LIBSVM[16]

E Additional Results

As shown in Figure 3, `OFFAR2-m` with large `m` may require a large number of epochs before achieving convergence for some problems. From the sampling plots, we see that a long-memory configuration may become an obstacle when the batch sizes grow too slowly, thereby resulting in a substantial number of iterations (and hence epochs) before achieving convergence.

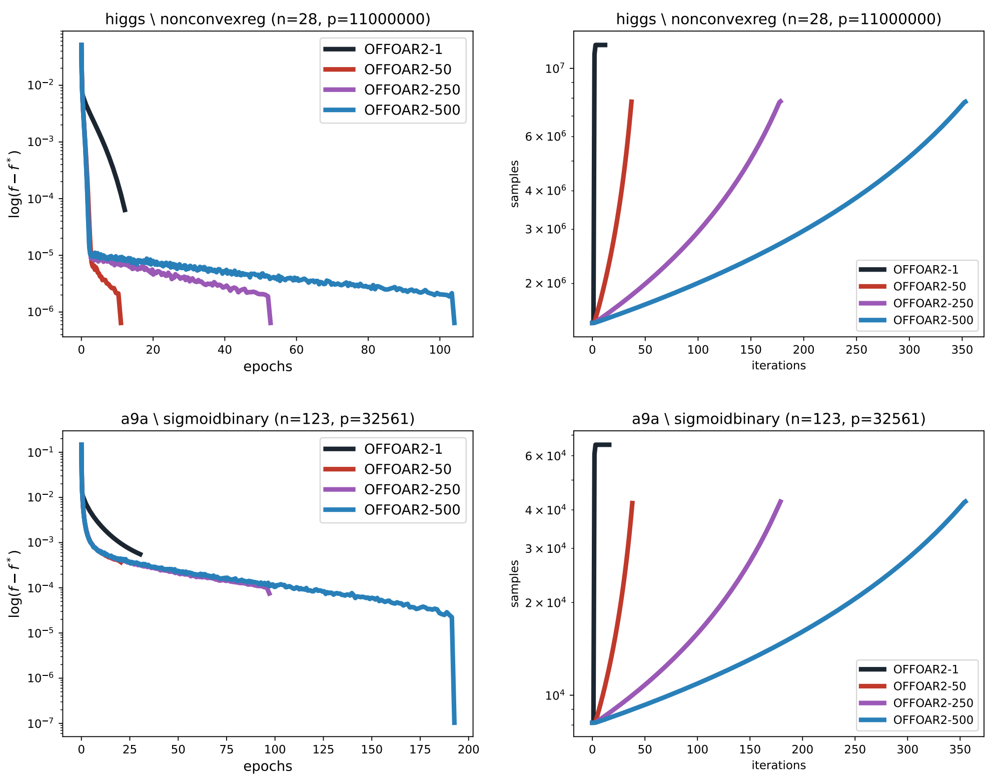

Figure 3: Evolution of the loss function w.r.t the epochs and sampling behavior along iterations for specific problems