Tuning-Free Stochastic Optimization

Ahmed Khaled [1] **Chi Jin** [1]

Abstract

Large-scale machine learning problems make the cost of hyperparameter tuning ever more prohibitive. This creates a need for algorithms that can tune themselves on-the-fly. We formalize the notion of *"tuning-free"* algorithms that can match the performance of optimally-tuned optimization algorithms up to polylogarithmic factors given only loose hints on the relevant problem parameters. We consider in particular algorithms that can match optimally-tuned Stochastic Gradient Descent (SGD). When the domain of optimization is bounded, we show tuning-free matching of SGD is possible and achieved by several existing algorithms. We prove that for the task of minimizing a convex and smooth or Lipschitz function over an unbounded domain, tuning-free optimization is impossible. We discuss conditions under which tuning-free optimization is possible even over unbounded domains. In particular, we show that the recently proposed DoG and DoWG algorithms are tuning-free when the noise distribution is sufficiently well-behaved. For the task of finding a stationary point of a smooth and potentially non-convex function, we give a variant of SGD that matches the best-known high-probability convergence rate for tuned SGD at only an additional polylogarithmic cost. However, we also give an impossibility result that shows no algorithm can hope to match the optimal expected convergence rate for tuned SGD with high probability.

1. Introduction

The hyperparameters we supply to an optimization algorithm can have a significant effect on the runtime of the algorithm and the quality of the final model (Yang et al., 2021; Sivaprasad et al., 2020). Yet hyperparameter tuning is costly, and for large models might prove intractable (Black et al., 2022). As a result, researchers often resort to using a well-known optimizer like Adam (Kingma & Ba, 2015) or AdamW (Loshchilov & Hutter, 2019) with widely used or default hyperparameters. For example, GPT-3 (Brown et al., 2020), BLOOM (Workshop et al., 2022), LLaMA (Touvron et al., 2023a), and LLaMA2 (Touvron et al., 2023b) all use either Adam or AdamW with identical momentum parameters and similar training recipes.

This situation presents an immense opportunity for algorithms that can tune hyperparameters on-the-fly. Yet such algorithms and their limits are still poorly understood in the setting of stochastic optimization. Let us make our setting more specific. We consider the minimization problem

$$\min_{x \in \mathcal{X}} f(x), \tag{OPT}$$

where $f : \mathcal{X} \to \mathbb{R}$ is differentiable and lower bounded by f_*. We assume that we have access to (stochastic) gradients $g(x)$ that satisfy certain regularity conditions that we shall make precise later.

Our main objects of study are *tuning-free* algorithms. To make this notion more precise, let $\mathcal{A}$ be an optimization algorithm that takes in n problem parameters $a = (a_1, a_2, \ldots, a_n)$ and after T (stochastic) gradient accesses returns a point $\overline{x}$ such that with high probability

$$f(\overline{x}) - f_* \leq \text{Error}_{\mathcal{A}}(f, a, T). \tag{1}$$

The function $\text{Error}_{\mathcal{A}}$ characterizes how well the algorithm $\mathcal{A}$ minimizes the function f in T steps given the supplied parameters. Let $a^* = a^*(f, T)$ denote the set of parameters that minimizes the right hand side of equation (1) for a specific function f and number of steps T. In order for an algorithm to find $a^*(f, T)$, it must start *somewhere*. We assume that we can easily find lower and upper bounds on the optimal parameters: two sets $\underline{a}$ and $\overline{a}$ such that for $i = 1, 2, \ldots, n$ we have

$$\underline{a}_i \leq a_i^* \leq \overline{a}_i.$$

Such *hints* on problem parameters can often be easily estimated in practice, and are a much easier ask than the optimal parameters. To be a *tuning-free* version of $\mathcal{A}$, an algorithm $\mathcal{B}$ has to approximately match the performance of $\mathcal{A}$ with optimally tuned parameters given those hints, a definition we make rigorous next.

[1]Electrical and Computer Engineering Department, Princeton University, Princeton, NJ, USA. Correspondence to: Ahmed Khaled <ahmed.khaled@princeton.edu>.

Proceedings of the 41^{st} International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

Definition 1.1. ***(Tuning-free algorithms).*** *We call an algorithm $\mathcal{B}$ a* tuning-free *version of $\mathcal{A}$ if given hints $\underline{a}, \overline{a}$ on the optimal parameters a^* for a function f it achieves the same error as $\mathcal{A}$ with the optimal parameters up to only polylogarithmic degradation that depends on the hints and the number of (stochastic) gradient accesses T. That is, if $\mathcal{A}$ achieves error $f(\overline{x}) - f_* \leq \mathrm{Error}_{\mathcal{A}}(f, a^*(f,T), T)$, then $\mathcal{B}$ achieves the guarantee:*

$$f(\overline{x}) - f_* \leq \iota \cdot \mathrm{Error}_{\mathcal{A}}(f, a^*, T), \qquad (\mathcal{B}\text{-error})$$

where $\iota = \mathrm{poly}\log\left(\frac{\overline{a}_1}{\underline{a}_1}, \ldots, \frac{\overline{a}_n}{\underline{a}_n}, T\right)$ is a polylogarithmic function of the hints.

Clearly, asking a tuning-free algorithm $\mathcal{B}$ to achieve exactly the same error as $\mathcal{A}$ is too much: we ought to pay some price for not knowing a^* upfront. On the other hand, if we allow polynomial dependencies on the hints, then our hints have to be very precise to avoid large errors. This beats the point of being tuning-free in the first place.

The algorithm $\mathcal{A}$ that we are primarily concerned with is Stochastic Gradient Descent (SGD). SGD and its variants dominate in practice, owing to their scalability and low memory requirements (Bottou et al., 2018). We consider three classes of functions: (a) L-smooth and convex functions, (b) G-Lipschitz and convex functions, and (c) L-smooth and potentially nonconvex functions. We ask for tuning-free algorithms for each of these function classes. We give precise definitions of these classes and our oracle model later in Section 1.1.

In the setting of deterministic optimization, we have a very good understanding of tuning-free optimization: there are many methods that, given only hints on the problem parameters required by Gradient Descent (GD), achieve the same rate as GD up to only polylogarithmic degradation. We review this case briefly in Section 4. Despite immense algorithmic developments in stochastic optimization (Duchi et al., 2010; Levy, 2017; Levy et al., 2018; Li & Orabona, 2019; Kavis et al., 2019; Carmon & Hinder, 2022; Ivgi et al., 2023; Cutkosky et al., 2023) and the related setting of online learning (Orabona & Pál, 2016; Cutkosky & Boahen, 2016; Cutkosky, 2019; Mhammedi et al., 2019; Mhammedi & Koolen, 2020; Orabona & Cutkosky, 2020) we are not aware of *any* algorithms that fit our definition as tuning-free counterparts of SGD for *any* of the function classes we consider. The main question of our work is thus:

> Can we find tuning-free counterparts for SGD in the setting of stochastic optimization and the classes of functions we consider (convex and smooth functions, convex and Lipschitz functions, and nonconvex and smooth functions)?

Our contributions. We answer the above question in the negative for the first two function classes and make some progress towards answering it for the third. In particular, our main contributions are:

- For **convex optimization**: if the domain of optimization is bounded, we highlight results from the literature showing tuning-free optimization matching SGD is possible. If the domain of optimization $\mathcal{X}$ is unbounded, we give an **impossibility result** that shows no algorithm can be a tuning-free counterpart of SGD for smooth and convex functions (Theorem 2), as well as for Lipschitz and convex functions (Theorem 3). Additionally if the stochastic gradient noise has a certain large *signal-to-noise* ratio (defined in Section 4.2), then tuning-free optimization is possible even when the domain of optimization $\mathcal{X}$ is unbounded and can be achieved by the recently-proposed DoG (Ivgi et al., 2023) and DoWG (Khaled et al., 2023) algorithms for both smooth and/or Lipschitz functions (Theorem 4).
- For **nonconvex optimization**: We consider two different notions of tuning-free optimization that correspond to the best-known convergence error bounds for SGD in expectation (Ghadimi & Lan, 2013) and with high probability (Liu et al., 2023). We show tuning-free optimization is **impossible** in the former setting (Theorem 5). On the other hand, for the latter, slightly weaker notion, we give a **positive result** and give a tuning-free variant of SGD (Theorem 6).

1.1. Preliminaries

In this section we review some preliminary notions and definitions that we shall make use of throughout the paper. We say that a function $f : \mathcal{X} \to \mathbb{R}$ is convex if for any $x, y \in \mathcal{X}$ we have

$$f(tx + (1-t)y) \leq tf(x) + (1-t)f(y) \text{ for all } t \in [0,1].$$

We call a function G-Lipschitz if $|f(x) - f(y)| \leq G\|x - y\|$ for all $x, y \in \mathcal{X}$. All norms considered in this paper are Euclidean. We let $\log_+ x \stackrel{\text{def}}{=} 1 + \log x$. A differentiable function f is L-smooth if for any $x, y \in \mathcal{X}$ we have $\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$.

Oracle model. All algorithms we consider shall access gradients through one of the two oracles defined below.

Definition 1.2. *We say that $\mathcal{O}(f)$ is a **deterministic first-order oracle** for the function f if given a point $x \in \mathcal{X}$ the oracle returns the pair $\{f(x), \nabla f(x)\}$.*

If we only allow the algorithm access to stochastic gradients, then we call this a stochastic oracle. Our main lower bounds are developed under the following noise model.

Assumption 1.1. *The stochastic gradient estimates are bounded almost surely. That is, there exists some $R \geq 0$*

such that for all $x \in \mathcal{X}$

$$\|\hat{g}(x) - \nabla f(x)\| \leq R.$$

The stochastic oracle model we consider allows for access to both function values and stochastic gradients satisfying Assumption 1.1.

Definition 1.3. *We say that* $\mathcal{O}(f, R_f)$ *is a* ***stochastic first-order oracle*** *for the function* f ***with bound*** σ_f *if, given a point* $x \in \mathcal{X}$*, it returns a pair of random variables* $\left[\hat{f}(x), \hat{g}(x)\right]$ *such that (a) the estimates are unbiased* $\mathbb{E}[\hat{f}(x)] = f(x)$, $\mathbf{E}[\hat{g}(x)] = \nabla f(x)$*, and (b) the stochastic gradients satisfy Assumption 1.1 with* $R = \sigma_f$.

The above oracle restricts the noise to be bounded almost surely. We shall develop our lower bounds under that oracle. However, for some of the upper bounds we develop, we shall relax the requirement on the noise from boundedness to being sub-gaussian (see Section 4.2), and we shall make this clear then.

2. Related Work

This section reviews existing approaches in the literature aimed at reducing or eliminating hyperparameter tuning.

Parameter-free optimization. An algorithm $\mathcal{A}$ is called *parameter-free* if it achieves the convergence rate $\tilde{\mathcal{O}}\left(\frac{G\|x_0 - x_*\|}{\sqrt{T}}\right)$ given T stochastic gradient accesses for any convex function f with stochastic subgradients bounded in norm by G, with possible knowledge of G (Orabona, 2023, Remark 1). There exists a vast literature on such methods, particularly in the setting of online learning, see e.g. (Orabona & Cutkosky, 2020). Parameter-free optimization differs from tuning-free optimization in two ways: (a) the $\tilde{\mathcal{O}}(\cdot)$ can suppress higher-order terms that are not permitted according to the tuning-free definition, and (b) gives the algorithm possible knowledge of a parameter like G whereas tuning-free algorithms can only get to see upper and lower bounds on G. Nevertheless, many parameter-free methods do not need any knowledge of G (Cutkosky, 2019; Mhammedi et al., 2019; Mhammedi & Koolen, 2020). However, Cutkosky & Boahen (2017b; 2016) give lower bounds showing that any online learning algorithm insisting on a linear dependence on $\|x_0 - x_*\|$ (as in optimally tuned SGD) must suffer from potentially exponential regret. If we do not insist on a linear dependence on $\|x_0 - x_*\|$, then the best achievable convergence bound scales $\|x_0 - x_*\|^3$, and this is tight (Mhammedi & Koolen, 2020). None of the aforementioned lower bounds apply to the setting of stochastic optimization, since in general online learning assumes an adversarial oracle, which is stronger than a stochastic oracle.

Tuning-free algorithms in the deterministic setting. Gradient descent augmented with line search (Nesterov, 2014; Beck, 2017) is tuning-free for smooth convex and nonconvex optimization. Bisection search (Carmon & Hinder, 2022) is tuning-free for both convex and smooth as well as convex and Lipschitz optimization, as is a restarted version of gradient descent with Polyak stepsizes (Hazan & Kakade, 2019). In the smooth setting, the adaptive descent method of (Malitsky & Mishchenko, 2020) is also tuning-free. There are also accelerated methods (Lan et al., 2023), methods for the Lipschitz setting (Defazio & Mishchenko, 2023), methods based on online learning (Orabona, 2023), and others. Renegar & Grimmer (2021) show that for strongly convex optimization, a simple restarting scheme suffices to obtain tuning-free algorithms in the deterministic setting.

Algorithms for the stochastic setting. Observe that because online learning is a more general setting than the stochastic one, we can apply algorithms from online convex optimization here, like e.g. (Mhammedi & Koolen, 2020) coupled with an appropriate online-to-batch conversion (Hazan, 2022). In more recent work (Carmon & Hinder, 2022; Ivgi et al., 2023), we see algorithmic developments specific to the stochastic setting. We discuss the convergence rates these algorithms achieve in more detail in Section 4.1. There are algorithms based on line search in the stochastic setting, but proving their convergence requires either extra assumptions like interpolation (Vaswani et al., 2019), or using large minibatch sizes (Paquette & Scheinberg, 2020). This drawback of stochastic line search is unavoidable, as (Vaswani et al., 2021, Theorem 4) shows applying stochastic line search on a quadratic objective results in non-convergence.

Other hyperparameter tuning approaches. In practice, hyperparameters are often found by grid search, random search, or methods based on Bayesian optimization (Bischl et al., 2023); None of these approaches come with efficient theoretical guarantees. Another approach is "meta-optimization" where we have a sequence of optimization problems and seek to minimize the cumulative error over this sequence. Often, another optimization algorithm is then used to select the learning rates, e.g. hypergradient descent (Baydin et al., 2018). Meta-optimization approaches are quite difficult to establish theoretical guarantees for, and only recently have some theoretical results been shown (Chen & Hazan, 2023). Our setting in this paper is different, since rather than seek to minimize regret over a sequence of optimization problems, we have a single function and an oracle that gives us (stochastic) gradient estimates for this function.

Concurrent work. In concurrent work, Carmon & Hinder (2024) and Attia & Koren (2024) also study lower bounds for first-order stochastic optimization. In both papers, like

in our work, the algorithm is provided with a certain range that the problem parameters fall in (what we term as *hints*) and must make use of only that to minimize the function with stochastic gradient evaluations. Carmon & Hinder (2024) study what is the minimum possible multiplicative factor slowdown any algorithm must suffer compared to optimally-tuned baselines when provided access only to hints, which they term the *price of adaptivity*. They provide lower bounds for stochastic convex optimization for Lipschitz functions in expectation and with high probability, and also consider the case where some of the problem parameters have no uncertainty (e.g. when we know the Lipschitz constant but not the initial distance to the optimum). Our lower bound in this setting (Theorem 3) rules out any polylogarithmic price of adaptivity as impossible. Additionally, we also give lower bounds for nonconvex and smooth convex optimization (Theorems 2 and 5).

Attia & Koren (2024) study stochastic optimization in a similar setting to ours, and give a new upper bound for restarted non-convex SGD that achieves a similar convergence guarantee to Theorem 6. We give our upper bound under a slightly more general noise distribution (that the noise has subgaussian norm) at the cost of a polylogarithmic dependence on the problem dimension. We also additionally give a lower bound that rules out the stronger in-expectation convergence guarantee for nonconvex SGD. In the convex setting, Attia & Koren (2024) give lower bounds for smooth and nonsmooth stochastic optimization that show a polynomial dependence on the hints is the best we can hope to achieve, and give a matching upper bound based on restarted SGD with AdaGrad-like stepsizes. In contrast, for our upper bounds in this case we study more specifically which noise distributions are amenable to optimization and prove results for the DoG and DoWG algorithms (with no restarting procedures). Additionally, we also investigate whether tuning-free optimization is possible under a bounded domain and provide guarantees for DoG/DoWG there (Theorem 1).

3. Tuning-Free Optimization Under a Bounded Domain

We begin our investigation by studying the **bounded setting**, where we make the following assumption on the minimization problem (OPT):

Assumption 3.1. *The optimization domain $\mathcal{X}$ is bounded. There exists some constant $D > 0$ such that $\|x - y\| \leq D$ for all $x, y \in \mathcal{X}$.*

We seek a tuning-free version of SGD. Recall that SGD achieves with probability at least $1 - \delta$ the following convergence guarantee (Jain et al., 2019; Liu et al., 2023)

$$f(x_{\text{out}}) - f_* \leq \upsilon \cdot \begin{cases} \frac{LD^2}{T} + \frac{\sigma D}{\sqrt{T}} & \text{if } f \text{ is } L\text{-smooth}, \\ \frac{\sqrt{G^2+\sigma^2}D}{\sqrt{T}} & \text{if } f \text{ is } G\text{-Lipschitz}, \end{cases} \tag{2}$$

where $\nu = \operatorname{poly}\log\frac{1}{\delta}$ and σ is an upper bound on the stochastic gradient noise (per Assumption 1.1). To achieve the convergence guarantee given by equation (2), we need to know the parameters D, σ, and L in the smooth case or G in the nonsmooth case. Per Definition 1.1, a tuning-free version of SGD will thus be given the hints $D \in [\underline{D}, \overline{D}]$, $\sigma \in [\underline{\sigma}, \overline{\sigma}]$, and either $L \in [\underline{L}, \overline{L}]$ in the smooth setting or $G \in [\underline{G}, \overline{G}]$ in the nonsmooth setting. Given those hints, we then ask the algorithm to achieve the same rate as equation (2) up to a multiplicative polylogarithmic function of the hints.

It turns out that tuning-free optimization under a bounded domain is solvable in many ways. Many methods from the online learning literature, e.g. (Cutkosky & Boahen, 2017a; Mhammedi et al., 2019; Cutkosky, 2019) can solve this problem when combined with standard online-to-batch conversion bounds. We give the details for this construction for one such algorithm in the next proposition:

Proposition 1. *Coin betting through Online Newton Steps with Hints (Cutkosky, 2019, Algorithm 1) is tuning-free in the bounded setting.*

The proof of this result is provided in the appendix, and essentially just combines (Cutkosky, 2019, Theorem 2) with online-to-batch conversion.

In this paper, we shall focus particularly on methods that fit the stochastic gradient descent paradigm, i.e. that use updates of the form $x_{k+1} = x_k - \eta_k g_k$, where g_k is the stochastic gradient at step k. Two methods that fit this paradigm are DoG (Ivgi et al., 2023) and DoWG (Khaled et al., 2023). DoG uses stepsizes of the form

$$\eta_t = \frac{\overline{r}_t}{\sqrt{u_t}}, \quad \overline{r}_t = \max_{k \leq t}(\|x_k - x_0\|, r_\epsilon), \quad u_t = \sum_{k=0}^{t} \|g_k\|^2, \tag{3}$$

where r_ϵ is a parameter that we will always set to $\underline{D}$. Similarly, DoWG uses stepsizes of the form

$$\eta_t = \frac{\overline{r}_t^2}{\sqrt{v_t}}, \quad \overline{r}_t = \max_{k \leq t}(\|x_k - x_0\|, r_\epsilon), \quad v_t = \sum_{k=0}^{t} \overline{r}_k^2 \|g_k\|^2. \tag{4}$$

The next theorem shows that in the bounded setting, both DoG and DoWG are tuning-free.

Theorem 1. *DoG and DoWG are tuning-free in the bounded setting. That is, there exists some* $\iota = \operatorname{poly}\log(\frac{\overline{D}}{\underline{D}}, \frac{\overline{\sigma}}{\underline{\sigma}}, T, \delta^{-1})$ *such that*

$$f(x_{\text{out}}) - f_* \leq \iota \cdot \begin{cases} \frac{LD^2}{T} + \frac{\sigma D}{\sqrt{T}} & \text{if } f \text{ is } L\text{-smooth}, \\ \frac{\sqrt{G^2+\sigma^2}D}{\sqrt{T}} & \text{if } f \text{ is } G\text{-Lipschitz}. \end{cases}$$

This rate is achieved simultaneously for both classes of functions without prior knowledge of whether f *is smooth or Lipschitz (and thus no usage of the hints* $\underline{L}, \overline{L}, \underline{G}, \overline{G}$*).*

This theorem essentially comes for free by modifying the results in (Ivgi et al., 2023; Khaled et al., 2023), and while the proof modifications are quite lengthy we claim no significant novelty here. We note further that unlike (Cutkosky, 2019, Algorithm 1), both DoG and DoWG are *single-loop* algorithms– they do not restart the optimization process or throw away progress. This is a valuable property and one of the reasons we focus on these algorithms in the paper. Moreover, DoG and DoWG are *universal*. An algorithm is universal if it achieves the same rate as SGD for Lipschitz functions and also takes advantage of smoothness when it exists (Levy, 2017), without any prior knowledge of whether f is smooth. DoG and DoWG enjoy this property in the bounded domain setting.

4. Tuning-free Optimization Under an Unbounded Domain

We now continue our investigation to the general, unbounded setting where $\mathcal{X} = \mathbb{R}^d$. Now, the diameter D in Assumption 3.1 is infinite. The convergence of SGD is then characterized by the initial distance to the optimum $D_* = \|x_0 - x_*\|$ (Liu et al., 2023). We can show that SGD with optimally-tuned stepsizes achieves with probability at least $1 - \delta$ the convergence rates

$$f(x_{\text{out}}) - f_* \leq \upsilon \cdot \begin{cases} \frac{LD_*^2}{T} + \frac{\sigma D_*}{\sqrt{T}} & \text{if } f \text{ is } L\text{-smooth}, \\ \frac{\sqrt{G^2+\sigma^2}D_*}{\sqrt{T}} & \text{if } f \text{ is } G\text{-Lipschitz}, \end{cases} \tag{5}$$

where $\upsilon = \operatorname{poly}\log\frac{1}{\delta}$, σ is the maximum stochastic gradient noise norm, and $D_* = \|x_0 - x_*\|$ is the initial distance from the minimizer. An algorithm is a tuning-free version of SGD in the unbounded setting if it can match the best SGD rates given by Equation (5) up to polylogarithmic factors given access to the hints $\underline{D}, \overline{D}, \underline{\sigma}, \overline{\sigma}$, and $\underline{G}, \overline{G}$ or $\underline{L}, \overline{L}$. This is a tall order: unlike in the bounded setting, a tuning-free algorithm now has to compete with SGD's convergence on *any* possible initialization.

Deterministic setting. When there is no stochastic gradient noise, i.e. $\sigma = 0$ and the algorithm accesses gradients according to the deterministic first-order oracle (Definition 1.2), Tuning-free versions of gradient descent exist. For example, the Adaptive Polyak algorithm (Hazan & Kakade, 2019), a restarted version of gradient descent with the Polyak stepsizes (Polyak, 1987) is tuning-free:

Proposition 2 (Hazan & Kakade (2019))**.** *The Adaptive Polyak algorithm from (Hazan & Kakade, 2019) is tuning-free in the deterministic setting.*

This is far from the only solution, and we mention a few others next. Parameter-free methods augmented with normalization are also tuning-free and universal, e.g. plugging in $d_0 = \underline{D}$ in (Orabona, 2023) gives tuning-free algorithms matching SGD. The bisection algorithm from (Carmon & Hinder, 2022) is also tuning-free, as is the simple doubling trick. Finally, T-DoG and T-DoWG, variants of DoG and DoWG which use polylogarithmically smaller stepsizes than DoG and DoWG, are also tuning-free, as the following direct corollary of (Ivgi et al., 2023; Khaled et al., 2023) shows.

Proposition 3. *T-DoG and T-DoWG are tuning-free in the deterministic setting.*

T-DoG and T-DoWG use the same stepsize structure as DoG and DoWG (given in equations (3) and (4)), but divide these stepsizes by running logarithmic factors as follows

$$\text{T-DoG:} \quad \eta_t = \frac{\overline{r}_t}{\sqrt{u_t}\log_+ \frac{u_t}{u_0}},$$

$$\text{T-DoWG:} \quad \eta_t = \frac{\overline{r}_t^2}{\sqrt{v_t}\log_+ \frac{v_t}{v_0}}.$$

Both methods achieve the same convergence guarantee as in equation (5) up to polylogarithmic factors in the hints.

4.1. Impossibility Results in the Stochastic Setting

The positive results in the deterministic setting give us some hope to obtain a tuning-free algorithm. Unfortunately, the stochastic setting turns out to be a tougher nut to crack. Our first major result, given below, slashes any hope of finding a tuning-free algorithm for smooth and stochastic convex optimization.

Theorem 2. *For any polylogarithmic function* $\iota : \mathbb{R}^4 \to \mathbb{R}$ *and any algorithm* $\mathcal{A}$*, there exists a time horizon* T*, an* L*-smooth and convex function* f*, and a stochastic oracle* $\mathcal{O}(f, \sigma_f)$*, and valid hints* $\underline{L}, \overline{L}, \underline{D}, \overline{D}, \underline{\sigma}, \overline{\sigma}$ *such that the algorithm* $\mathcal{A}$ *initialized at some* x_0 *returns with some constant probability a point* x_{out} *satisfying*

$$\begin{aligned} \text{Error}_{\mathcal{A}} &= f(x_{\text{out}}) - f_* \\ &> \iota\left(\frac{\overline{L}}{\underline{L}}, \frac{\overline{D}}{\underline{D}}, \frac{\overline{\sigma}}{\underline{\sigma}}, T\right) \cdot \left[\frac{LD_*^2}{T} + \frac{\sigma_f D_*}{\sqrt{T}}\right], \end{aligned}$$

where $D_* = \|x_0 - x_*\|$ *is the initial distance to the optimum and* σ_f *is the maximum norm of the stochastic gradient noise.*

Proof idea. This lower bound is achieved by 1-dimensional functions. In particular, we construct two one-dimensional quadratic functions f and h with associated oracles $\mathcal{O}(f, \sigma_f)$ and $\mathcal{O}(h, \sigma_h)$, and we supply the algorithm with hints that are valid for both functions and oracles. We show that with some constant probability, the algorithm observes the same stochastic gradients from both $\mathcal{O}(f, \sigma_f)$ and $\mathcal{O}(h, \sigma_h)$ for the entire run. Since the algorithm cannot tell apart either oracle, it must guarantee that equation (5) holds with high probability for both f and h if it is to be tuning-free. Now, if we choose f and h further apart, ensuring that their respective oracles return the same gradients with some constant probability becomes harder. On the other hand, if we choose f and h too close, the algorithm can conceivably guarantee that equation (5) holds (up to the same polylogarithmic factor of the hints) for both of them. By carefully choosing f and h to balance out this tradeoff, we show that no algorithm can be tuning-free in the unbounded and stochastic setting. The full proof is provided in Section 8.3 in the appendix.

Comparison with prior lower bounds. The above theorem shows a fundamental separation between the deterministic and stochastic settings when not given knowledge of the problem parameters. The classical lower bounds for deterministic and stochastic optimization algorithms (Nesterov, 2018; Woodworth & Srebro, 2016; Carmon et al., 2019) rely on a chain construction that is agnostic to whether the optimization algorithm has access to problem parameters. On the other hand, lower bounds from the online learning literature show that tuning-free optimization matching SGD is impossible when the oracle can be adversarial (and not stochastic), see e.g. (Cutkosky & Boahen, 2017b; 2016). However, adversarial oracles are much stronger than stochastic oracles, as they can change the function being optimized in response to the algorithm's choices. Our lower bound is closest in spirit to the lower bounds from the stochastic multi-armed bandits literature that also rely on confusing the algorithm with two close problems (Mannor & Tsitsiklis, 2004).

Our next result shows that tuning-free optimization is also impossible in the nonsmooth case.

Theorem 3. *For any polylogarithmic function $\iota : \mathbb{R}^4 \to \mathbb{R}$ and any algorithm $\mathcal{A}$, there exists a time horizon T, valid hints $\underline{L}, \overline{L}, \underline{D}, \overline{D}, \underline{\sigma}, \overline{\sigma}$, an G-Lipschitz and convex function f and an oracle $\mathcal{O}(f, \sigma_f)$ such that the algorithm $\mathcal{A}$ returns with some constant probability a point x_{out} satisfying*

$$\begin{aligned}\text{Error}_{\mathcal{A}} &= f(x_{\text{out}}) - f_* \\ &> \iota\left(\frac{\overline{G}}{\underline{G}}, \frac{\overline{D}}{\underline{D}}, \frac{\overline{\sigma}}{\underline{\sigma}}, T\right) \cdot \left[\frac{\sqrt{G^2 + \sigma^2} D_*}{\sqrt{T}}\right].\end{aligned}$$

The proof technique used for this result relies on a similar construction as Theorem 2 but uses the absolute loss instead of quadratics.

Existing algorithms and upper bounds in the stochastic setting. Carmon & Hinder (2022) give a restarted variant of SGD with bisection search. If f is G-Lipschitz and all the stochastic gradients are also bounded by G, their method uses the hint $\overline{G} \geq G$ and achieves the following convergence rate with probability at least $1 - \delta$

$$f(\hat{x}) - f_* \leq c\iota\left(\eta_\epsilon\left(G + \frac{\iota\overline{G}}{T}\right) + \frac{D_* G}{\sqrt{T}} + \frac{D_*\overline{G}}{T}\right), \quad (6)$$

where c is an absolute constant, ι a poly-logarithmic and double-logarithmic factor, and η_ϵ an input parameter. If we set $\eta_\epsilon = \underline{D}/T \leq \frac{D_*}{T}$, then the guarantee of this method becomes

$$f(\hat{x}) - f_* \leq c\iota\left(\frac{D_* G}{\sqrt{T}} + \frac{D_*\overline{G}}{T}\right). \quad (7)$$

Unfortunately, this dependence does not meet our bar as the polynomial dependence on $\overline{G}$ is in a higher-order term. A similar result is achieved by DoG (Ivgi et al., 2023). A different sort of guarantee is achieved by Mhammedi & Koolen (2020), who give a method with regret

$$\frac{1}{T}\sum_{t=0}^{T-1}\langle g_t, x_t - x_*\rangle \leq c\iota\left(\frac{GD_*}{\sqrt{T}} + \frac{GD_*^3}{T}\right), \quad (8)$$

for some absolute constant c and polylogarithmic factor ι. This result is in the adversarial setting of online learning, and clearly does not meet the bar for tuning-free optimization matching SGD due to the cubic term D_*^3.

4.2. Guarantees Under Benign Noise

In the last subsection, we saw that tuning-free optimization is in general impossible. However, it is clear that *sometimes* it is possible to get within the performance of tuned SGD with self-tuning methods (Ivgi et al., 2023; Defazio & Mishchenko, 2023). However, the oracles used in Theorems 2 and 3 provide stochastic gradients $g(x)$ such that the noise is almost surely bounded (i.e. satisfies Assumption 1.1):

$$\|g(x) - \nabla f(x)\| \leq R.$$

So boundedness is clearly not enough to enable tuning-free optimization. However, we know from prior results (e.g. (Carmon & Hinder, 2022)) that if we can reliably estimate the upper bound R on the noise, we can adapt to unknown distance to the optimum D_* or the smoothness constant L. The main issue that the oracles in the lower bound of Theorem 2 make it impossible to do that: while the noise $n(x) = g(x) - \nabla f(x)$ is bounded almost surely by R, the

algorithm only gets to observe the same noise $n(x)$ for the entire optimization run. This foils any attempt at estimating R from the observed trajectory.

A note on notation in this section and the next. In the past section we used σ to denote a uniform upper bound on the gradient noise, while in this section and the next we use σ to denote the *variance* of the stochastic gradient noise $n(x)$ rather than a uniform upper bound on it. Instead, we use R to denote the uniform upper bound on the noise.

We will see that for some notion of *benign noise*, tuning-free optimization matching SGD is possible. We will develop our results under a more general assumption on the distribution of the stochastic gradient noise $g(x) - \nabla f(x)$

Assumption 4.1. *(Noise with Sub-Gaussian norm). For all $x \in \mathbb{R}^d$, the noise vector $n(x) = g(x) - \nabla f(x)$ satisfies*

- *$n(x)$ is unbiased: $\mathbb{E}[g(x)] = \nabla f(x)$.*
- *$n(x)$ has sub-gaussian norm with modulus R:*

$$\text{Prob}\left(\|n(x)\| \geq t\right) \leq 2\exp\left(\frac{-t^2}{2R^2}\right).$$

- *$n(x)$ has bounded variance: $\mathbb{E}\left[\|n(x)\|^2\right] = \sigma^2 < +\infty$.*

This assumption is very general, it subsumes bounded noise (where $R = \sigma_f$) and sub-gaussian noise. The next definition gives a notion of *signal-to-noise* that turns out to be key in characterizing benign noise distributions.

Definition 4.1. *Suppose the stochastic gradient noise satisfies Assumption 4.1. We define the **signal-to-noise** ratio associated with the noise as*

$$K_{\text{snr}} = \frac{\sigma}{R} \leq 1.$$

To better understand the meaning of K_{snr}, we consider the following example. Let Y be a random vector with mean $\mathbb{E}[Y] = \mu$ and variance $\mathbb{E}\left[\|Y-\mu\|^2\right] = \sigma^2$. Suppose further that the errors $\|Y - \mu\|$ are bounded almost surely by some R. Then $Y - \mu$ satisfies the assumptions in Assumption 4.1. Let $Y_1, \ldots, Y_b$ be independent copies of Y. Through standard concentration results (see Lemma 8) we can show that with high probability and for large enough b

$$\hat{\sigma} \overset{\text{def}}{=} \frac{1}{b}\sum_{i=1}^{b}\|Y-\mu\|^2 \approxeq \sigma^2.$$

Now observe that if the ratio K_{snr} is small, then we cannot use the sample variance $\hat{\sigma}$ as an estimator for the almost-sure bound R. But if the ratio K_{snr} is closer to 1, then we have $\sigma^2 \approxeq R^2$ and we can use $\hat{\sigma}$ as an estimator for R. This fixes the problem we highlighted earlier: now we are able to get an accurate estimate of R from the observed stochastic gradients. The next proposition gives examples of noise distributions where K_{snr} is close to 1.

Proposition 4. *Suppose that the noise vectors $g(x) - \nabla f(x)$ follow one of the following two distributions:*

- *A Gaussian distribution with mean 0 and covariance $\frac{\sigma^2}{d} \cdot I_{d\times d}$, with $\sigma > 0$.*
- *A Bernoulli distribution, where $[g(x) - \nabla f(x)] = \pm\sigma\phi(x)$ with equal probability for some ϕ such that $\|\phi(x)\|_2 = 1$ almost surely.*

Then $K_{\text{snr}} = \mathcal{O}(1)$.

We now give an algorithm whose convergence rate characterized by the signal-to-noise ratio K_{snr}. We combine a variane estimation procedure with the T-DoG/T-DoWG algorithms in Algorithms 2 and 3. The next theorem gives the convergence of this algorithm. This theorem is generic, and does not guarantee any tuning-free matching of SGD, but can lead to tuning-free matching of SGD if the signal to noise ratio is high enough.

Theorem 4. *Suppose we are given access to stochastic gradient estimates $g(x)$ such that the noise vectors $[g(x) - \nabla f(x)] \in \mathbb{R}^d$ satisfy Assumption 4.1 with modulus R and signal-to-noise ratio K_{snr}. If we run T-DoG or T-DoWG with variance estimation (Algorithms 2 and 3) with a minibatch size $b \geq 2$ large enough to satisfy*

$$c\cdot\left[\sqrt{\frac{\log\frac{2bT}{\delta}}{b}} + \frac{\log\frac{2(b\vee d)T}{\delta}}{b}\right] \leq K_{\text{snr}}^2 - \theta,$$

where c is some absolute constant and $\theta \in [0, K_{\text{snr}}]$ is some known constant. Then Algorithms 2 and 3 with either option returns a point x_{out} such that with probability at least $1-\delta$,

- *If f is L-smooth:*

$$f(x_{\text{out}}) - f_* \leq c\iota\left(\frac{LD_*^2 b}{T_{\text{total}}} + \frac{\theta^{-1}RD_*\sqrt{b}}{\sqrt{T_{\text{total}}}}\right),$$

  where $D_* = \|x_0 - x_*\|$, c is an absolute constant, ι is a polylogarithmic factor of the hints, and T_{total} denotes the total number of stochastic gradient accesses.

- *If f is G-Lipschitz:*

$$f(x_{\text{out}}) - f_* \leq c\iota\frac{\sqrt{G^2 + \theta^{-2}R^2}D_*\sqrt{b}}{\sqrt{T_{\text{total}}}}.$$

Note on dimension dependence in Theorem 4. We note that the logarithmic dimension dependence on the dimension d can be removed if, rather than assuming the norm of the noise is subgaussian, we assumed it was bounded.

On the surface, it looks like Theorem 4 simply trades off knowledge of the absolute bound on the noise R with knowledge of some constant θ that lies in the interval $[0, K_{\text{snr}}]$. In order to see how Theorem 4 can be useful, consider the special cases given in Proposition 4. For these noise distributions, we see that choosing a minibatch size $b \approx \mathcal{O}(\log \frac{2dT}{\delta} + 1)$ suffices to ensure Algorithms 2 and 3 converges with the simple choice $\theta = \frac{1}{2}$. Even though we had no apriori knowledge of the variance σ^2 and did not assume the noise distribution was stationary, we could still optimize the function. In general, the minibatch size $b \approx \mathcal{O}(\log \frac{2dT}{\delta} + 1)$ suffices as long as K_{snr} is bounded away from zero by some constant. The final cost of running the algorithm is $T_{\text{total}} = b \cdot T = \iota T$, where ι is some polylogarithmic factor. Therefore, we only pay a logarithmic price for not knowing the distribution. Of course, if K_{snr} is small enough, there can be no optimization– there is not enough signal to do any estimation of the sub-gaussian modulus R. The distribution used in Theorem 2 does force $K_{\text{snr}} \leq \frac{1}{T}$.

5. Nonconvex Tuning-Free Optimization

In this section, we consider the case where the optimization problem (OPT) is possibly nonconvex. Throughout the section, we assume that f is L-smooth and lower bounded by some $f_* \in \mathbb{R}$. In this setting, SGD with a tuned stepsize achieves the following rate in expectation

$$\frac{1}{T}\sum_{t=0}^{T-1} \|\nabla f(x_t)\|^2 \leq c\left[\sqrt{\frac{L(f(x_0)-f_*)\sigma^2}{T}} + \frac{L(f(x_0)-f_*)}{T}\right], \tag{9}$$

for some absolute constant $c > 0$. This rate is known to be tight for convergence in expectation (Arjevani et al., 2019). However, it is not known if it is tight for returning a *high probability* guarantee. The best-known high-probability convergence rate for SGD is given by (Liu et al., 2023, Theorem 4.1) and guarantees with probability at least $1-\delta$ that

$$\frac{1}{T}\sum_{t=0}^{T-1} \|\nabla f(x_t)\|^2 \leq 5\sqrt{\frac{L(f(x_0)-f_*)R^2}{T}} + \frac{2(f(x_0)-f_*)L}{T} + \frac{12R^2\log\frac{1}{\delta}}{T}. \tag{10}$$

We now consider tuning-free algorithms that can match the performance of SGD characterized by either equation (9) or equation (1). Per Definition 1.1, an algorithm $\mathcal{B}$ is given (1) an initialization x_0, (2) a budget of T_{total} stochastic gradient accesses, and (3) hints $\underline{L}, \overline{L}, \underline{R}, \overline{R}, \underline{\Delta}, \overline{\Delta}$ on the problem parameters such that (a) if L is the smoothness constant of f then $L \in [\underline{L}, \overline{L}]$, (b) $R \in [\underline{R}, \overline{R}]$, and (c) $\Delta \stackrel{\text{def}}{=} f(x_0) - f_* \in [\underline{\Delta}, \overline{\Delta}]$. We call $\mathcal{B}$ **strongly** tuning-free if it matches the performance of SGD characterized by equation (9) up to polylogarithmic factors. Alternatively, if it instead matches the weaker guarantee given by equation (10) then we call it **weakly** tuning-free.

Algorithm 1 Restarted SGD

Require: Initialization y_0, probability δ, hints $\underline{R}, \overline{R}, \underline{\Delta}, \overline{\Delta}, \underline{L}, \overline{L}$, total budget T_{total}

Set $\eta_\epsilon = \frac{1}{\overline{L}}$ and

$$N = 1 + \left\lceil \log\left(\frac{\min(\overline{L}, \sqrt{\frac{5T\overline{R}^2}{2\underline{\Delta}}})}{\max(\underline{L}, \sqrt{\frac{5T\underline{R}^2}{\overline{\Delta}}})}\right)\right\rceil \tag{11}$$

if $T_{\text{total}} < N$ **then**
  Return y_0.
end if
Set the per-epoch iteration budget as $T = \lceil T_{\text{total}}/N \rceil$.
for $n = 1$ to N **do**
  $\eta = \eta_\epsilon 2^n$
  Run SGD for T iterations with stepsize η starting from y_0 to get outputs $x_1^n, \ldots, x_T^n$.
  Set $y_n, \hat{g}_n = \text{FindLeader}(S, \delta, T)$ (see Algorithm 4).
end for
Return $\overline{y} = \arg\min_{n\in[N]} \|\hat{g}_n\|$.

Our first result in this setting shows that we cannot hope to achieve the rate given by equation (9) in high probability, even given access to hints on all the problem parameters.

Theorem 5. *For any polylogarithmic function $\iota : \mathbb{R}^4 \to \mathbb{R}$ and any algorithm $\mathcal{A}$, there exists a time horizon T, valid hints $\underline{L}, \overline{L}, \underline{\Delta}, \overline{\Delta}, \underline{\sigma}, \overline{\sigma}$, an L-smooth and lower-bounded function f and an oracle $\mathcal{O}(f, \sigma_f)$ such that the algorithm $\mathcal{A}$ returns with some constant probability a point x_{out} satisfying*

$$\text{Error}_{\mathcal{A}} = \|\nabla f(x_{\text{out}})\|^2 > \iota\left(\frac{\overline{L}}{\underline{L}}, \frac{\overline{\Delta}}{\underline{\Delta}}, \frac{\overline{\sigma}}{\underline{\sigma}}, T\right) \cdot \left[\sqrt{\frac{L\Delta\sigma^2}{T}} + \frac{L\Delta}{T}\right],$$

where $\Delta = f(x_0) - f_*$

Surprisingly, our next theorem shows that the rate given by equation (10) *is* achievable up to polylogarithmic factors given only access to hints. To achieve this, we use a restarted variant of SGD (Algorithm 6) combined with a "Leader

Finding" procedure that selects a well-performing iterate by subsampling.

Theorem 6. *(Convergence of Restarted SGD) Let f be an L-smooth function lower bounded by f_* and suppose the stochastic gradient noise vectors satisfy Assumption 4.1. Suppose that we are given the following hints on the problem parameters: (a) $L \in [\underline{L}, \overline{L}]$, (b) $R_f \in [\underline{R}, \overline{R}]$, and (c) $\Delta_f \overset{def}{=} f(x_0) - f_* \in [\underline{\Delta}, \overline{\Delta}]$. Then there exists some absolute constant c such that the output of Algorithm 1 satisfies after $T_{\text{total}} \cdot \log_+ \frac{1}{\delta}$ stochastic gradient evaluations*

$$\|\nabla f(\overline{y})\|^2 \leq c \cdot \frac{R^2 \log \frac{2d \max(\log \frac{1}{\delta}, N)}{\delta}}{T_{\text{total}}} + c \cdot N \cdot \left[\sqrt{\frac{L(f(y_0) - f_*)R^2}{T_{\text{total}}}} + \frac{(f(y_0) - f_*)L}{T_{\text{total}}} \right],$$

where c is an absolute constant, N is a polylogarithmic function of the hints defined in equation (11), and d is the problem dimensionality.

Discussion of Theorem 6. This theorem shows that in the nonconvex setting, we pay only an additional polylogarithmic factor to achieve the same high-probability rate as when we know all parameters. We emphasize that we do not know if the rate given by equation (10) is tight, but it is the best in the literature. Finally, the logarithmic dimension dependence on the dimension d can be removed if, rather than assuming the norm of the noise is subgaussian, we assumed that it was bounded almost surely.

Proof Idea. The proof is an application of the so-called "doubling trick" with a careful comparison procedure. If we start with a small enough stepsize, we only need to double a logarithmic number of times until we find a stepsize η' such that $\frac{\eta_*}{2} \leq \eta' \leq \eta_*$, where η_* is the optimal stepsize for SGD on this problem. We therefore run SGD for N epochs with a carefully chosen N, each time doubling the stepsize. At the end of every SGD run, we run the FindLeader procedure (Algorithm 4) to get with high probability a point y_n such that

$$\|\nabla f(y_n)\|^2 \leq \frac{1}{T} \sum_{t=0}^{T-1} \|\nabla f(x_t^n)\|^2,$$

where $x_1^n, \ldots, x_T^n$ are the SGD iterates from the n-th epoch. Finally, we know that at least one of these N points $y_1, \ldots, y_N$ has small gradient norm, so we return the point with the minimal estimated gradient norm and bound the estimation error as a function of T. The total number of gradient accesses performed is at most $N(T + MT) \approx T_{\text{total}} \cdot \log_+ \frac{1}{\delta}$. Therefore, both the restarting and comparison procedures add at most a logarithmic number of gradient accesses.

Related work. Many papers give high probability bounds for SGD or AdaGrad and their variants in the nonconvex setting (Ghadimi & Lan, 2013; Madden et al., 2020; Lei & Tang, 2021; Li & Orabona, 2019; 2020; Faw et al., 2022; Kavis et al., 2022), but to the best of our knowledge none give a tuning-free algorithm matching SGD per Definition 1.1. The FindLeader procedure is essentially extracted from (Madden et al., 2020, Theorem 13), and is similar to the post-processing step in (Ghadimi & Lan, 2013).

Comparison with the convex setting. The rate achieved by Theorem 6 stands in contrast to the best-known rates in the convex setting, where we suffer from a polynomial dependence on the hints, as in equation (7). One potential reason for this divergence is the difficulty of telling apart good and bad points. In the convex setting, we ask for a point $\overline{y}$ with a small function value $f(\overline{y})$. And while the oracle gives us access to stochastic realizations of $f(\overline{y})$, the error in those realization is *not* controlled. Instead, to compare between two points y_1 and y_2 we have to rely on stochastic gradient information to approximate $f(y_1) - f(y_2)$, and this seems to be too difficult without apriori control on the distance between y_1 and y_2. On the other hand, in the nonconvex setting, such comparison is feasible through sampling methods like e.g. Algorithm 4.

6. Conclusion and Open Problems

We have reached the end of our investigation. To summarize: we defined tuning-free algorithms and studied several settings where tuning-free optimization was possible, and several where we proved impossibility results. Yet, many open questions remain. For example, tuning-free optimization might be possible in the finite-sum setting where we can periodically evaluate the function value exactly. The upper bounds we develop in both the convex and nonconvex settings require quite stringent assumptions on the noise (such as boundedness or sub-gaussian norm), and it is not known if they can be relaxed to expected smoothness (Gower et al., 2019; Khaled & Richtárik, 2020) or some variant of it. In the nonconvex case we only consider smooth objectives whereas in deep learning the objectives are usually highly nonsmooth, and exploring this area may yield more practically useful insights. Finally, we did not study tuning-free algorithms for strongly convex objectives. We leave these questions to future work.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

Acknowledgements

We thank Aaron Defazio, Yair Carmon, and Oliver Hinder for discussions during the preparation of this work. This work is partially supported by National Science Foundation Grant NSF-OAC-2411299.

References

Arjevani, Y., Carmon, Y., Duchi, J. C., Foster, D. J., Srebro, N., and Woodworth, B. Lower bounds for non-convex stochastic optimization. *arXiv preprint arXiv:1912.02365*, abs/1912.02365, 2019.

Attia, A. and Koren, T. SGD with AdaGrad stepsizes: Full adaptivity with high probability to unknown parameters, unbounded gradients and affine variance. *arXiv preprint arXiv:2302.08783*, abs/2302.08783, 2023.

Attia, A. and Koren, T. How free is parameter-free stochastic optimization? *arXiv preprint arXiv:2402.03126*, abs/2402.03126, 2024.

Baydin, A. G., Cornish, R., Martínez-Rubio, D., Schmidt, M., and Wood, F. Online learning rate adaptation with hypergradient descent. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

Beck, A. *First-order methods in optimization*. MOS-SIAM series on optimization. Society for Industrial and Applied Mathematics ; Mathematical Optimization Society, 2017. ISBN 9781611974997.

Bischl, B., Binder, M., Lang, M., Pielok, T., Richter, J., Coors, S., Thomas, J., Ullmann, T., Becker, M., Boulesteix, A., Deng, D., and Lindauer, M. Hyperparameter optimization: Foundations, algorithms, best practices, and open challenges. *WIREs Data Mining and Knowledge Discovery*, 13(2), January 2023. ISSN 1942-4795. doi: 10.1002/widm.1484.

Black, S., Biderman, S., Hallahan, E., Anthony, Q., Gao, L., Golding, L., He, H., Leahy, C., McDonell, K., Phang, J., Pieler, M., Prashanth, U. S., Purohit, S., Reynolds, L., Tow, J., Wang, B., and Weinbach, S. GPT-NeoX-20B: an open-source autoregressive language model. *arXiv preprint arXiv:2204.06745*, abs/2204.06745, 2022.

Bottou, L., Curtis, F. E., and Nocedal, J. Optimization methods for large-scale machine learning. *SIAM Review*, 60(2):223–311, 2018. doi: 10.1137/16M1080173.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. Language models are few-shot learners. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

Carmon, Y. and Hinder, O. Making SGD parameter-free. In Loh, P. and Raginsky, M. (eds.), *Conference on Learning Theory, 2-5 July 2022, London, UK*, volume 178 of *Proceedings of Machine Learning Research*, pp. 2360–2389. PMLR, 2022.

Carmon, Y. and Hinder, O. The price of adaptivity in stochastic convex optimization. *arXiv preprint arXiv:2402.10898*, abs/2402.10898, 2024.

Carmon, Y., Duchi, J. C., Hinder, O., and Sidford, A. Lower bounds for finding stationary points II: first-order methods. *Mathematical Programming*, 185(1–2): 315–355, September 2019. ISSN 1436-4646. doi: 10.1007/s10107-019-01431-x.

Chen, X. and Hazan, E. A nonstochastic control approach to optimization. *arXiv preprint arXiv:2301.07902*, abs/2301.07902, 2023.

Cutkosky, A. Artificial constraints and hints for unbounded online learning. In Beygelzimer, A. and Hsu, D. (eds.), *Proceedings of the Thirty-Second Conference on Learning Theory*, volume 99 of *Proceedings of Machine Learning Research*, pp. 874–894. PMLR, 25–28 Jun 2019.

Cutkosky, A. and Boahen, K. A. Online convex optimization with unconstrained domains and losses. In Lee, D. D., Sugiyama, M., von Luxburg, U., Guyon, I., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pp. 748–756, 2016.

Cutkosky, A. and Boahen, K. A. Stochastic and adversarial online learning without hyperparameters. In Guyon, I., von Luxburg, U., Bengio, S., Wallach, H. M., Fergus, R., Vishwanathan, S. V. N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 5059–5067, 2017a.

Cutkosky, A. and Boahen, K. A. Online learning without prior information. In Kale, S. and Shamir, O. (eds.), *Proceedings of the 30th Conference on Learning Theory, COLT 2017, Amsterdam, The Netherlands, 7-10 July*

2017, volume 65 of *Proceedings of Machine Learning Research*, pp. 643–677. PMLR, 2017b.

Cutkosky, A., Defazio, A., and Mehta, H. Mechanic: a learning rate tuner. *arXiv preprint arXiv:2306.00144*, abs/2306.00144, 2023.

Defazio, A. and Mishchenko, K. Learning-Rate-Free learning by D-adaptation. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org, 2023.

Duchi, J. C., Hazan, E., and Singer, Y. Adaptive subgradient methods for online learning and stochastic optimization. In Kalai, A. T. and Mohri, M. (eds.), *COLT 2010 - The 23rd Conference on Learning Theory, Haifa, Israel, June 27-29, 2010*, pp. 257–269. Omnipress, 2010.

Faw, M., Tziotis, I., Caramanis, C., Mokhtari, A., Shakkottai, S., and Ward, R. The power of adaptivity in SGD: self-tuning step sizes with unbounded gradients and affine variance. In Loh, P. and Raginsky, M. (eds.), *Conference on Learning Theory, 2-5 July 2022, London, UK*, volume 178 of *Proceedings of Machine Learning Research*, pp. 313–355. PMLR, 2022.

Ghadimi, S. and Lan, G. Stochastic first- and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4):2341–2368, 2013. doi: 10.1137/120880811.

Gower, R. M., Loizou, N., Qian, X., Sailanbayev, A., Shulgin, E., and Richtárik, P. SGD: General analysis and improved rates. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 5200–5209. PMLR, 2019.

Hazan, E. *Introduction to Online Convex Optimization*. Book draft, MIT University Press, 2022.

Hazan, E. and Kakade, S. Revisiting the Polyak step size. *arXiv preprint arXiv:1905.00313*, abs/1905.00313, 2019.

Ivgi, M., Hinder, O., and Carmon, Y. DoG is SGD's best friend: a parameter-free dynamic step size schedule. *arXiv preprint arXiv:2302.12022*, abs/2302.12022, 2023.

Jain, P., Nagaraj, D., and Netrapalli, P. Making the last iterate of SGD information theoretically optimal. In Beygelzimer, A. and Hsu, D. (eds.), *Conference on Learning Theory, COLT 2019, 25-28 June 2019, Phoenix, AZ, USA*, volume 99 of *Proceedings of Machine Learning Research*, pp. 1752–1755. PMLR, 2019.

Jin, C., Netrapalli, P., Ge, R., Kakade, S. M., and Jordan, M. I. A short note on concentration inequalities for random vectors with subgaussian norm. *arXiv preprint arXiv:1902.03736*, abs/1902.03736, 2019.

Kavis, A., Levy, K. Y., Bach, F. R., and Cevher, V. Unixgrad: A universal, adaptive algorithm with optimal guarantees for constrained optimization. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E. B., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 6257–6266, 2019.

Kavis, A., Levy, K. Y., and Cevher, V. High probability bounds for a class of nonconvex algorithms with adagrad stepsize. In *ICLR*. OpenReview.net, 2022.

Khaled, A. and Richtárik, P. Better theory for SGD in the nonconvex world. *arXiv preprint arXiv:2002.03329*, abs/2002.03329, 2020.

Khaled, A., Mishchenko, K., and Jin, C. DoWG unleashed: An efficient universal parameter-free gradient descent method. In Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S. (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 6748–6769. Curran Associates, Inc., 2023.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In Bengio, Y. and LeCun, Y. (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

Lan, G., Ouyang, Y., and Zhang, Z. Optimal and parameter-free gradient minimization methods for smooth optimization. *arXiv preprint arXiv:2310.12139*, abs/2310.12139, 2023.

Lei, Y. and Tang, K. Learning rates for stochastic gradient descent with nonconvex objectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(12): 4505–4511, December 2021. ISSN 1939-3539. doi: 10.1109/tpami.2021.3068154.

Levy, K. Y. Online to offline conversions, universality and adaptive minibatch sizes. In Guyon, I., von Luxburg, U., Bengio, S., Wallach, H. M., Fergus, R., Vishwanathan, S. V. N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 1613–1622, 2017.

Levy, K. Y., Yurtsever, A., and Cevher, V. Online adaptive methods, universality and acceleration. In Bengio,

S., Wallach, H. M., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pp. 6501–6510, 2018.

Li, X. and Orabona, F. On the convergence of stochastic gradient descent with adaptive stepsizes. In Chaudhuri, K. and Sugiyama, M. (eds.), *The 22nd International Conference on Artificial Intelligence and Statistics, AISTATS 2019, 16-18 April 2019, Naha, Okinawa, Japan*, volume 89 of *Proceedings of Machine Learning Research*, pp. 983–992. PMLR, 2019.

Li, X. and Orabona, F. A high probability analysis of adaptive SGD with momentum. *arXiv preprint arXiv:2007.14294*, abs/2007.14294, 2020.

Liu, Z., Nguyen, T. D., Nguyen, T. H., Ene, A., and Nguyen, H. L. High probability convergence of stochastic gradient methods. *arXiv preprint arXiv:2302.14843*, abs/2302.14843, 2023.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

Madden, L., Dall'Anese, E., and Becker, S. High-probability convergence bounds for non-convex stochastic gradient descent. *arXiv preprint arXiv:2006.05610*, abs/2006.05610, 2020.

Malitsky, Y. and Mishchenko, K. Adaptive gradient descent without descent. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 6702–6712. PMLR, 2020.

Mannor, S. and Tsitsiklis, J. N. The sample complexity of exploration in the multi-armed bandit problem. *J. Mach. Learn. Res.*, 5:623–648, 2004. ISSN 1532-4435.

Mhammedi, Z. and Koolen, W. M. Lipschitz and comparator-norm adaptivity in online learning. In Abernethy, J. D. and Agarwal, S. (eds.), *Conference on Learning Theory, COLT 2020, 9-12 July 2020, Virtual Event [Graz, Austria]*, volume 125 of *Proceedings of Machine Learning Research*, pp. 2858–2887. PMLR, 2020.

Mhammedi, Z., Koolen, W. M., and van Erven, T. Lipschitz adaptivity with multiple learning rates in online learning. In Beygelzimer, A. and Hsu, D. (eds.), *Conference on Learning Theory, COLT 2019, 25-28 June 2019, Phoenix, AZ, USA*, volume 99 of *Proceedings of Machine Learning Research*, pp. 2490–2511. PMLR, 2019.

Nesterov, Y. Universal gradient methods for convex optimization problems. *Mathematical Programming*, 152 (1-2):381–404, 2014. doi: 10.1007/s10107-014-0790-0.

Nesterov, Y. *Lectures on Convex Optimization*. Springer Publishing Company, Incorporated, 2nd edition, 2018. ISBN 3319915770.

Orabona, F. Normalized gradients for all. *arXiv preprint arXiv:2308.05621*, abs/2308.05621, 2023.

Orabona, F. and Cutkosky, A. ICML 2020 tutorial on parameter-free online optimization. *ICML Tutorials*, 2020.

Orabona, F. and Pál, D. Coin betting and parameter-free online learning. In Lee, D. D., Sugiyama, M., von Luxburg, U., Guyon, I., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pp. 577–585, 2016.

Paquette, C. and Scheinberg, K. A stochastic line search method with expected complexity analysis. *SIAM Journal on Optimization*, 30(1):349–376, 2020. doi: 10.1137/18M1216250.

Polyak, B. T. *Introduction to Optimization*. Translations Series in Mathematics and Engineering. Optimization Software Inc., New York, 1987.

Renegar, J. and Grimmer, B. A simple nearly optimal restart scheme for speeding up first-order methods. *Foundations of Computational Mathematics*, 22(1):211–256, 2021. doi: 10.1007/s10208-021-09502-2.

Sivaprasad, P. T., Mai, F., Vogels, T., Jaggi, M., and Fleuret, F. Optimizer benchmarking needs to account for hyperparameter tuning. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9036–9045. PMLR, 2020.

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., Rodriguez, A., Joulin, A., Grave, E., and Lample, G. LLaMA: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, abs/2302.13971, 2023a.

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., Bikel, D., Blecher, L., Ferrer, C. C., Chen, M., Cucurull, G., Esiobu, D., Fernandes, J., Fu, J., Fu, W., Fuller, B., Gao, C., Goswami, V., Goyal, N., Hartshorn, A., Hosseini, S., Hou, R., Inan, H., Kardas, M., Kerkez, V., Khabsa, M., Kloumann, I., Korenev, A., Koura, P. S.,

Lachaux, M.-A., Lavril, T., Lee, J., Liskovich, D., Lu, Y., Mao, Y., Martinet, X., Mihaylov, T., Mishra, P., Molybog, I., Nie, Y., Poulton, A., Reizenstein, J., Rungta, R., Saladi, K., Schelten, A., Silva, R., Smith, E. M., Subramanian, R., Tan, X. E., Tang, B., Taylor, R., Williams, A., Kuan, J. X., Xu, P., Yan, Z., Zarov, I., Zhang, Y., Fan, A., Kambadur, M., Narang, S., Rodriguez, A., Stojnic, R., Edunov, S., and Scialom, T. LLaMA 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023b.

Vaswani, S., Mishkin, A., Laradji, I., Schmidt, M., Gidel, G., and Lacoste-Julien, S. Painless stochastic gradient: Interpolation, line-search, and convergence rates. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Vaswani, S., Dubois-Taine, B., and Babanezhad, R. Towards noise-adaptive, problem-adaptive stochastic gradient descent. *arXiv preprint arXiv:2110.11442*, abs/2110.11442, 2021.

Vershynin, R. High-dimensional probability: An introduction with applications in data science. 2018. doi: 10.1017/9781108231596.

Woodworth, B. E. and Srebro, N. Tight complexity bounds for optimizing composite objectives. In Lee, D. D., Sugiyama, M., von Luxburg, U., Guyon, I., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pp. 3639–3647, 2016.

Workshop, B., Scao, T. L., Fan, A., Akiki, C., Pavlick, E., Ilić, S., Hesslow, D., Castagné, R., Luccioni, A. S., Yvon, F., Gallé, M., Tow, J., Rush, A. M., Biderman, S., Webson, A., Ammanamanchi, P. S., Wang, T., Sagot, B., Muennighoff, N., Moral, A. V. d., Ruwase, O., Bawden, R., Bekman, S., McMillan-Major, A., Beltagy, I., Nguyen, H., Saulnier, L., Tan, S., Suarez, P. O., Sanh, V., Laurençon, H., Jernite, Y., Launay, J., Mitchell, M., Raffel, C., Gokaslan, A., Simhi, A., Soroa, A., Aji, A. F., Alfassy, A., Rogers, A., Nitzav, A. K., Xu, C., Mou, C., Emezue, C., Klamm, C., Leong, C., Strien, D. v., Adelani, D. I., Radev, D., Ponferrada, E. G., Levkovizh, E., Kim, E., Natan, E. B., Toni, F. D., Dupont, G., Kruszewski, G., Pistilli, G., Elsahar, H., Benyamina, H., Tran, H., Yu, I., Abdulmumin, I., Johnson, I., Gonzalez-Dios, I., Rosa, J. d. l., Chim, J., Dodge, J., Zhu, J., Chang, J., Frohberg, J., Tobing, J., Bhattacharjee, J., Almubarak, K., Chen, K., Lo, K., Werra, L. V., Weber, L., Phan, L., allal, L. B., Tanguy, L., Dey, M., Muñoz, M. R., Masoud, M., Grandury, M., Saško, M., Huang, M., Coavoux, M., Singh, M., Jiang, M. T.-J., Vu, M. C., Jauhar, M. A., Ghaleb, M., Subramani, N., Kassner, N., Khamis, N., Nguyen, O., Espejel, O., Gibert, O. d., Villegas, P., Henderson, P., Colombo, P., Amuok, P., Lhoest, Q., Harliman, R., Bommasani, R., López, R. L., Ribeiro, R., Osei, S., Pyysalo, S., Nagel, S., Bose, S., Muhammad, S. H., Sharma, S., Longpre, S., Nikpoor, S., Silberberg, S., Pai, S., Zink, S., Torrent, T. T., Schick, T., Thrush, T., Danchev, V., Nikoulina, V., Laippala, V., Lepercq, V., Prabhu, V., Alyafeai, Z., Talat, Z., Raja, A., Heinzerling, B., Si, C., Taşar, D. E., Salesky, E., Mielke, S. J., Lee, W. Y., Sharma, A., Santilli, A., Chaffin, A., Stiegler, A., Datta, D., Szczechla, E., Chhablani, G., Wang, H., Pandey, H., Strobelt, H., Fries, J. A., Rozen, J., Gao, L., Sutawika, L., Bari, M. S., Al-shaibani, M. S., Manica, M., Nayak, N., Teehan, R., Albanie, S., Shen, S., Ben-David, S., Bach, S. H., Kim, T., Bers, T., Fevry, T., Neeraj, T., Thakker, U., Raunak, V., Tang, X., Yong, Z.-X., Sun, Z., Brody, S., Uri, Y., Tojarieh, H., Roberts, A., Chung, H. W., Tae, J., Phang, J., Press, O., Li, C., Narayanan, D., Bourfoune, H., Casper, J., Rasley, J., Ryabinin, M., Mishra, M., Zhang, M., Shoeybi, M., Peyrounette, M., Patry, N., Tazi, N., Sanseviero, O., Platen, P. v., Cornette, P., Lavallée, P. F., Lacroix, R., Rajbhandari, S., Gandhi, S., Smith, S., Requena, S., Patil, S., Dettmers, T., Baruwa, A., Singh, A., Cheveleva, A., Ligozat, A.-L., Subramonian, A., Névéol, A., Lovering, C., Garrette, D., Tunuguntla, D., Reiter, E., Taktasheva, E., Voloshina, E., Bogdanov, E., Winata, G. I., Schoelkopf, H., Kalo, J.-C., Novikova, J., Forde, J. Z., Clive, J., Kasai, J., Kawamura, K., Hazan, L., Carpuat, M., Clinciu, M., Kim, N., Cheng, N., Serikov, O., Antverg, O., Wal, O. v. d., Zhang, R., Zhang, R., Gehrmann, S., Mirkin, S., Pais, S., Shavrina, T., Scialom, T., Yun, T., Limisiewicz, T., Rieser, V., Protasov, V., Mikhailov, V., Pruksachatkun, Y., Belinkov, Y., Bamberger, Z., Kasner, Z., Rueda, A., Pestana, A., Feizpour, A., Khan, A., Faranak, A., Santos, A., Hevia, A., Unldreaj, A., Aghagol, A., Abdollahi, A., Tammour, A., HajiHosseini, A., Behroozi, B., Ajibade, B., Saxena, B., Ferrandis, C. M., McDuff, D., Contractor, D., Lansky, D., David, D., Kiela, D., Nguyen, D. A., Tan, E., Baylor, E., Ozoani, E., Mirza, F., Ononiwu, F., Rezanejad, H., Jones, H., Bhattacharya, I., Solaiman, I., Sedenko, I., Nejadgholi, I., Passmore, J., Seltzer, J., Sanz, J. B., Dutra, L., Samagaio, M., Elbadri, M., Mieskes, M., Gerchick, M., Akinlolu, M., McKenna, M., Qiu, M., Ghauri, M., Burynok, M., Abrar, N., Rajani, N., Elkott, N., Fahmy, N., Samuel, O., An, R., Kromann, R., Hao, R., Alizadeh, S., Shubber, S., Wang, S., Roy, S., Viguier, S., Le, T., Oyebade, T., Le, T., Yang, Y., Nguyen, Z., Kashyap, A. R., Palasciano, A., Callahan, A., Shukla, A., Miranda-Escalada, A., Singh, A., Beilharz, B., Wang, B., Brito, C., Zhou, C., Jain, C., Xu, C., Fourrier, C., Periñán, D. L., Molano, D., Yu, D., Manjavacas, E., Barth, F., Fuhrimann,

F., Altay, G., Bayrak, G., Burns, G., Vrabec, H. U., Bello, I., Dash, I., Kang, J., Giorgi, J., Golde, J., Posada, J. D., Sivaraman, K. R., Bulchandani, L., Liu, L., Shinzato, L., Bykhovetz, M. H. d., Takeuchi, M., Pàmies, M., Castillo, M. A., Nezhurina, M., Sänger, M., Samwald, M., Cullan, M., Weinberg, M., Wolf, M. D., Mihaljcic, M., Liu, M., Freidank, M., Kang, M., Seelam, N., Dahlberg, N., Broad, N. M., Muellner, N., Fung, P., Haller, P., Chandrasekhar, R., Eisenberg, R., Martin, R., Canalli, R., Su, R., Su, R., Cahyawijaya, S., Garda, S., Deshmukh, S. S., Mishra, S., Kiblawi, S., Ott, S., Sang-aroonsiri, S., Kumar, S., Schweter, S., Bharati, S., Laud, T., Gigant, T., Kainuma, T., Kusa, W., Labrak, Y., Bajaj, Y. S., Venkatraman, Y., Xu, Y., Xu, Y., Xu, Y., Tan, Z., Xie, Z., Ye, Z., Bras, M., Belkada, Y., and Wolf, T. BLOOM: a 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*, abs/2211.05100, 2022.

Yang, G., Hu, E. J., Babuschkin, I., Sidor, S., Liu, X., Farhi, D., Ryder, N., Pachocki, J., Chen, W., and Gao, J. Tuning large neural networks via zero-shot hyperparameter transfer. In Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, 2021.

Appendix

7. Proofs for Section 7

Proposition 1. *Coin betting through Online Newton Steps with Hints (Cutkosky, 2019, Algorithm 1) is tuning-free in the bounded setting.*

Proof. In the bounded setting, Cutkosky (2019) give an algorithm that takes as parameters ϵ, α and achieves the following regret

$$\sum_{t=0}^{T-1} \langle g_t, x_t - x_* \rangle \leq \epsilon + GD + \|x_* - x_0\| G \log \left[\frac{\|x_* - x_0\| G \exp(\alpha/4G^2)}{\epsilon} \left(1 + \frac{\sum_{t=0}^{T-1} \|g_t\|^2}{\alpha} \right)^{4.5} \right] + \|x_* - x_0\| \sqrt{\sum_{t=0}^{T-1} \|g_t\|^2 \log \left[\frac{\left(\sum_{t=0}^{T-1} \|g_t\|^2 \right)^{10} \exp(\alpha/2G^2) \|x_*\|^2}{\epsilon^2} + 1 \right]}.$$

If we set $\epsilon = \underline{D} \cdot \underline{G}$, $\alpha = \underline{G}^2$, and use the upper bound $\|x_0 - x_*\| \leq D$ and simplify we get the regret

$$\sum_{t=0}^{T-1} \langle g_t, x_t - x_* \rangle \leq \underline{GD} + GD + DG \log \left[\frac{DG}{\underline{DG}} \left(1 + \frac{G^2 T}{\underline{G}^2} \right)^{4.5} \right] + D \sqrt{\sum_{t=0}^{T-1} \|g_t\|^2} \sqrt{\log \frac{T^{10} G^{20} D^2}{\underline{D}^2 \underline{G}^2}}$$

Observe that because $\underline{G} \leq G$ and $\underline{D} \leq D$ the above can be simplified to

$$\sum_{t=0}^{T-1} \langle g_t, x_t - x_* \rangle \leq GD \log_+ \left[\frac{DG}{\underline{DG}} \left(1 + \frac{G^2 T}{\underline{G}^2} \right)^{4.5} \right] + D \sqrt{\sum_{t=0}^{T-1} \|g_t\|^2} \sqrt{\log \frac{T^{10} G^{20} D^2}{\underline{D}^2 \underline{G}^2}}$$

Call the maximum of the two log terms ι, then the above rate is

$$\sum_{t=0}^{T-1} \langle g_t, x_t - x_* \rangle \leq GD\iota + D \sqrt{\sum_{t=0}^{T-1} \|g_t\|^2} \sqrt{\iota}. \tag{12}$$

Applying online-to-batch conversion starting from equation (12) proves the algorithm is tuning-free. For the smooth setting, it suffices to observe that under a bounded domain we have for any t

$$\begin{aligned} \|g_t\| &\leq \|g_t - \nabla f(x_t)\| + \|\nabla f(x_t)\| \\ &= \|g_t - \nabla f(x_t)\| + \|\nabla f(x_t) - \nabla f(x_*)\| \\ &\leq \sigma + L \|x_t - x_*\| \\ &\leq \sigma + LD. \end{aligned}$$

Combining this and following online-to-batch conversion as in (Levy, 2017) shows the algorithm considered is tuning-free in the smooth setting as well.

□

We will make use of the following two lemmas throughout the upper bound proofs for DoG and DoWG.

Lemma 1. *(Ivgi et al., 2023, Lemma 7). Let S be the set of nonnegative and nondecreasing sequences. Let $C_t \in \mathcal{F}_{t-1}$ and let X_t be a martingale difference sequence adapted to $\mathcal{F}_t$ such that $|X_t| \leq C_t$ with probability 1 for all t. Then for all $\delta \in (0,1)$ and $\hat{X}_t \in \mathcal{F}_{t-1}$ such that $\left|\hat{X}_t\right| \leq C_t$ with probability 1, we have that with probability $1 - \delta$ that for all $c > 0$*

$$\left| \sum_{i=1}^{t} y_i X_i \right| \leq 8 y_t \sqrt{\theta_{t,\delta} \sum_{i=1}^{t} (X_i - \hat{X}_i)^2 + [c]\, \theta_{t,\delta}^2} + \text{Prob}\left(\exists t \leq T \mid C_t > c \right)$$

Lemma 2. *(Ivgi et al., 2023, Lemma 3). Let $s_0, s_1, \dots, s_T$ be a positive increasing sequence. Then,*

$$\max_{t\le T}\sum_{i=0}^{t-1}\frac{s_i}{s_t}\ge\frac{1}{e}\left(\frac{T}{\log_+\frac{s_T}{s_0}}-1\right).$$

Lemma 3. *(Ivgi et al., 2023, Lemma 1). Suppose that f is convex and has a minimizer x_*. Then the iterates generated by DoG satisfy for each t:*

$$\sum_{k=a}^{b-1}\overline{r}_k\left\langle g_k,x_k-x_*\right\rangle\le\overline{r}_b(2\overline{d}_b+\overline{r}_b)\sqrt{u_{b-1}}.$$

Lemma 4. *Suppose that f is convex and has a minimizer x_*. Then iterates generated by DoWG satisfy for every t:*

$$\sum_{k=0}^{t-1}\overline{r}_k^2\left\langle\nabla f(x_k),x_k-x_*\right\rangle\le2\overline{r}_t\left[\overline{d}_t+\overline{r}_t\right]\sqrt{v_{t-1}}+\sum_{k=0}^{t-1}\overline{r}_k^2\left\langle\nabla f(x_k)-g_k,x_k-x_*\right\rangle$$

Proof. This is a modification of (Khaled et al., 2023, Lemma 3) to account for the case where $g_k\neq\nabla f(x_k)$ (i.e. when the gradients used are not deterministic), following (Ivgi et al., 2023, Lemma 1). We start

$$\begin{aligned}d_{k+1}^2&\le\left\|x_k-\eta_kg_k-x_*\right\|^2\\&=\left\|x_k-x_*\right\|^2+\eta_k^2\|g_k\|^2-2\eta_k\left\langle g_k,x_k-x_*\right\rangle.\end{aligned}$$

Rearranging we get

$$2\eta_k\left\langle g_k,x_k-x_*\right\rangle\le d_k^2-d_{k+1}^2+\eta_k^2\|g_k\|^2$$

Multiplying both sides by $\frac{\overline{r}_k^2}{2\eta_k}$ we get

$$\overline{r}_k^2\left\langle g_k,x_k-x_*\right\rangle\le\frac{1}{2}\frac{\overline{r}_k^2}{\eta_k}\left(d_k^2-d_{k+1}^2\right)+\frac{\overline{r}_k^2\eta_k}{2}\|g_k\|^2.$$

We then follow the same proof as in (Khaled et al., 2023, Lemma 3) to get

$$\sum_{k=0}^{t-1}\overline{r}_k^2\left\langle g_k,x_k-x_*\right\rangle\le2\overline{r}_t\left[\overline{d}_t+\overline{r}_t\right]\sqrt{v_{t-1}}.\tag{13}$$

We then decompose

$$\sum_{k=0}^{t-1}\overline{r}_k^2\left\langle g_k,x_k-x_*\right\rangle=\sum_{k=0}^{t-1}\overline{r}_k^2\left\langle g_k-\nabla f(x_k),x_k-x_*\right\rangle+\sum_{k=0}^{t-1}\overline{r}_k^2\left\langle\nabla f(x_k),x_k-x_*\right\rangle.$$

Plugging back into equation (13) we get

$$\sum_{k=0}^{t-1}\overline{r}_k^2\left\langle\nabla f(x_k),x_k-x_*\right\rangle\le2\overline{r}_t\left[\overline{d}_t+\overline{r}_t\right]\sqrt{v_{t-1}}+\sum_{k=0}^{t-1}\overline{r}_k^2\left\langle\nabla f(x_k)-g_k,x_k-x_*\right\rangle\tag{14}$$

□

7.1. Proof of Theorem 1

Theorem 1. *DoG and DoWG are tuning-free in the bounded setting. That is, there exists some $\iota=\operatorname{poly}\log(\frac{\overline{D}}{\underline{D}},\frac{\overline{\sigma}}{\underline{\sigma}},T,\delta^{-1})$ such that*

$$f(x_{\text{out}})-f_*\le\iota\cdot\begin{cases}\frac{LD^2}{T}+\frac{\sigma D}{\sqrt{T}} & \text{if } f \text{ is } L\text{-smooth},\\ \frac{\sqrt{G^2+\sigma^2}D}{\sqrt{T}} & \text{if } f \text{ is } G\text{-Lipschitz}.\end{cases}$$

This rate is achieved simultaneously for both classes of functions without prior knowledge of whether f is smooth or Lipschitz (and thus no usage of the hints $\underline{L},\overline{L},\underline{G},\overline{G}$).

Proof of Theorem 1. We first handle the case that $T < 4\log_+ \frac{\overline{D}}{\underline{D}}$. In this case we just return x_0. If f is G-Lipschitz, then by convexity we have

$$f(x_0) - f_* \leq \langle \nabla f(x_0), x_0 - x_* \rangle \leq \|\nabla f(x_0)\| \|x_0 - x_*\| \leq GD \leq \frac{2GD}{\sqrt{T}} \sqrt{\log_+ \frac{\overline{D}}{\underline{D}}}.$$

If f is L-smooth, then by smoothness we have

$$f(x_0) - f_* \leq \frac{L}{2}\|x_0 - x_*\|^2 \leq \frac{LD^2}{2} \leq \frac{2LD^2}{T} \log_+ \frac{\overline{D}}{\underline{D}}.$$

Therefore in both cases the point we return achieves a small enough loss almost surely. Throughout the rest of the proof, we shall assume that $T \geq 4\log_+ \frac{\overline{D}}{\underline{D}}$.

Part 1: DoG. In the nonsmooth setting, this is a straightforward consequence of (Ivgi et al., 2023, Proposition 3). In particular, when using DoG with $r_\epsilon = \underline{D}$, then Corollary 1 in their work gives that with probability $1-\delta$ there exists some $\tau \in [T]$ and some absolute constant $c > 0$ such that

$$f(\overline{x}_\tau) - f_* \leq c \cdot \frac{DG}{\sqrt{T}} \log \frac{60 \log 6t}{\delta} \log \frac{2D}{\underline{D}},$$

where $\hat{x}_t \stackrel{\text{def}}{=} \frac{1}{\sum_{i=0}^{t-1} \overline{r}_i} \sum_{i=0}^{t-1} \overline{r}_i x_i$.

For the smooth setting, we start with Lemma 3 to get

$$\sum_{k=0}^{t-1} \overline{r}_k \langle \nabla f(x_k), x_k - x_* \rangle \leq \overline{r}_t \left(2\overline{d}_t + \overline{r}_t\right) \sqrt{u_{t-1}} + \sum_{k=0}^{t-1} \overline{r}_k \langle \nabla f(x_k) - g_k, x_k - x_* \rangle. \tag{15}$$

We follow (Ivgi et al., 2023, Proposition 3) and modify the proof in a straightforward manner to accommodate the assumption of bounded noise (rather than bounded gradients). Define

$$\tau_k = \min\left\{\min\left\{i \mid \overline{r}_i \geq 2\overline{r}_{\tau_i - 1}\right\}, T\right\}, \qquad \tau_0 \stackrel{\text{def}}{=} 0.$$

We denote by K the first index such that $\tau_K = T$. Define

$$X_k = \left\langle g_k - \nabla f(x_k), \frac{x_k - x_*}{\overline{d}_k} \right\rangle, \qquad \hat{X}_k = 0, \qquad y_k = \overline{r}_k \overline{d}_k.$$

Observe that x_k is determined by $\mathcal{F}_{k-1}$, and since $\overline{r}_k = \max_{t \leq k}\left(\|x_k - x_0\|, r_\epsilon\right)$, it is also determined by $\mathcal{F}_{k-1}$. Therefore

$$\mathbb{E}\left[X_k \mid \mathcal{F}_{k-1}\right] = \overline{r}_k^2 \left\langle \mathbb{E}\left[g_k - \nabla f(x_k)\right], \frac{x_k - x_*}{\overline{d}_k} \right\rangle = 0.$$

Moreover, observe that

$$|X_k| \leq \|g_k - \nabla f(x_k)\| \frac{\|x_k - x_*\|}{\overline{d}_k} \leq \sigma.$$

Therefore the X_k form a martingale. Then we can apply Lemma 1 to get that with probability $1-\delta$ that for every $t \in [K]$

$$\begin{aligned}
\left|\sum_{k=0}^{t-1} \overline{r}_k \langle g_k - \nabla f(x_k), x_k - x_* \rangle\right| &\leq 8\overline{d}_t \overline{r}_t \theta_{t,\delta} \sqrt{\sum_{k=0}^{t-1} (X_k)^2 + \sigma^2} \\
&\leq 8\overline{d}_t \overline{r}_t \theta_{t,\delta} \sqrt{\sum_{k=0}^{t-1} \|g_k - \nabla f(x_k)\|^2 + \sigma^2} \\
&\leq 8\overline{d}_t \overline{r}_t \theta_{t,\delta} \sqrt{\sigma^2 t + \sigma^2} \\
&\leq 16\overline{d}_t \overline{r}_t \theta_{t,\delta} \sigma \sqrt{T}.
\end{aligned} \tag{16}$$

Now observe that we can use equation (16) to get

$$\begin{aligned}\left|\sum_{k=\tau_{i-1}}^{\tau_i-1}\overline{r}_k\left\langle g_k-\nabla f(x_k),x_k-x_*\right\rangle\right| &\le \left|\sum_{k=0}^{\tau_i-1}\overline{r}_k\left\langle g_k-\nabla f(x_k),x_k-x_*\right\rangle\right|+\left|\sum_{k=0}^{\tau_{i-1}-1}\overline{r}_k\left\langle g_k-\nabla f(x_k),x_k-x_*\right\rangle\right| \\ &\le 16\overline{d}_{\tau_i}\overline{r}_{\tau_i}\theta_{t,\delta}\sigma\sqrt{T}+16\overline{d}_{\tau_{i-1}}\overline{r}_{\tau_{i-1}}\theta_{t,\delta}\sigma\sqrt{T} \\ &\le 32\overline{d}_{\tau_i}\overline{r}_{\tau_i}\theta_{t,\delta}\sigma\sqrt{T}. \end{aligned} \tag{17}$$

Now observe that by convexity we have for $k\in\{\tau_{i-1},\tau_{i-1}+1,\dots,\tau_i-1\}$

$$0\le f(x_k)-f_*\le\left\langle\nabla f(x_k),x_k-x_*\right\rangle\le\frac{\overline{r}_k}{\overline{r}_{\tau_{i-1}}}\left\langle\nabla f(x_k),x_k-x_*\right\rangle.$$

Summing up from $k=\tau_{i-1}$ to $k=\tau_i-1$ we get

$$\begin{aligned}\sum_{k=\tau_{i-1}}^{\tau_i-1}\left\langle\nabla f(x_k),x_k-x_*\right\rangle &\le \frac{1}{\overline{r}_{\tau_{i-1}}}\sum_{k=\tau_{i-1}}^{\tau_i-1}\overline{r}_k\left\langle\nabla f(x_k),x_k-x_*\right\rangle \\ &= \frac{1}{\overline{r}_{\tau_{i-1}}}\sum_{k=\tau_{i-1}}^{\tau_i-1}\overline{r}_k\left\langle\nabla f(x_k),x_k-x_*\right\rangle \\ &= \frac{1}{\overline{r}_{\tau_{i-1}}}\sum_{k=\tau_{i-1}}^{\tau_i-1}\overline{r}_k\left\langle\nabla f(x_k)-g_k,x_k-x_*\right\rangle+\frac{1}{\overline{r}_{\tau_{i-1}}}\sum_{k=\tau_{i-1}}^{\tau_i-1}\overline{r}_k\left\langle g_k,x_k-x_*\right\rangle. \end{aligned} \tag{18}$$

We now use Lemma 3 to get that

$$\sum_{k=\tau_{i-1}}^{\tau_i-1}\overline{r}_k\left\langle g_k,x_k-x_*\right\rangle\le 2\overline{r}_{\tau_i}\left(\overline{d}_{\tau_i}+\overline{r}_{\tau_i}\right)\sqrt{u_{\tau_i-1}}. \tag{19}$$

Plugging in the upper bounds from equations (17) and (19) into equation (18) we get

$$\sum_{k=\tau_{i-1}}^{\tau_i-1}\left\langle\nabla f(x_k),x_k-x_*\right\rangle\le\frac{\overline{r}_{\tau_i}}{\overline{r}_{\tau_{i-1}}}\left[2\left(\overline{d}_{\tau_i}+\overline{r}_{\tau_i}\right)\sqrt{u_{\tau_i-1}}+32\overline{d}_{\tau_i}\theta_{t,\delta}\sigma\sqrt{T}\right]. \tag{20}$$

Now observe that

$$\overline{r}_{k+1}\le\overline{r}_k+\|x_{t+1}-x_t\|=\overline{r}_k\left(1+\frac{\|g_k\|}{\sqrt{u_k}}\right)\le 2\overline{r}_k.$$

It follows that $\frac{\overline{r}_{\tau_i}}{\overline{r}_{\tau_i-1}}\le 2$. Moreover by the definition of the τ_i we have that $\frac{\overline{r}_{\tau_i-1}}{\overline{r}_{\tau_{i-1}}}\le 2$. Therefore

$$\frac{\overline{r}_{\tau_i}}{\overline{r}_{\tau_{i-1}}}=\frac{\overline{r}_{\tau_i}}{\overline{r}_{\tau_i-1}}\frac{\overline{r}_{\tau_i-1}}{\overline{r}_{\tau_{i-1}}}\le 2\cdot 2=4. \tag{21}$$

using equation (21) in equation (20) we get

$$\sum_{k=\tau_{i-1}}^{\tau_i-1}\left\langle\nabla f(x_k),x_k-x_*\right\rangle\le 4\left[2\left(\overline{d}_{\tau_i}+\overline{r}_{\tau_i}\right)\sqrt{u_{\tau_i-1}}+32\overline{d}_{\tau_i}\theta_{t,\delta}\sigma\sqrt{T}\right].$$

Summing up over the i, we get

$$\sum_{t=0}^{T-1}\left\langle\nabla f(x_t),x_t-x_*\right\rangle\le\sum_{i=0}^{K}\sum_{k=\tau_{i-1}}^{\tau_i-1}\left\langle\nabla f(x_k),x_k-x_*\right\rangle\le 4K\left[2\left(\overline{d}_{\tau_i}+\overline{r}_{\tau_i}\right)\sqrt{u_{\tau_i-1}}+32K\overline{d}_{\tau_i}\theta_{t,\delta}\sigma\sqrt{T}\right].$$

Observe that by definition we have

$$K \leq 1 + \log \frac{\overline{r}_T}{r_0} = \log \frac{2\overline{r}_T}{r_0}.$$

Therefore using the last equation and convexity we have

$$\sum_{t=0}^{T-1}(f(x_t) - f_*) \leq 4 \log \frac{2\overline{r}_T}{r_0} \left[2\left(\overline{d}_T + \overline{r}_T\right)\sqrt{u_{T-1}} + 32\overline{d}_T\theta_{T,\delta}\sigma\sqrt{T}\right].$$

Note that because the domain is bounded we have $\max(\overline{r}_T, \overline{d}_T) \leq D$, and we used $r_0 = \underline{D}$, therefore

$$\sum_{t=0}^{T-1}(f(x_t) - f_*) \leq 4 \log \frac{2D}{\underline{D}} \left[4D\sqrt{u_{T-1}} + 32D\theta_{T,\delta}\sigma\sqrt{T}\right]. \tag{22}$$

Observe that by our assumption on the noise and smoothness we have

$$\begin{aligned}
u_{T-1} &= \sum_{k=0}^{T-1} \|g_k\|^2 \\
&\leq 2\sum_{k=0}^{T-1} \|g_k - \nabla f(x_k)\|^2 + 2\sum_{k=0}^{T-1} \|\nabla f(x_k)\|^2 \\
&\leq 2T\sigma^2 + 2\sum_{k=0}^{T-1} \|\nabla f(x_k)\|^2 \\
&\leq 2T\sigma^2 + 4L\sum_{k=0}^{T-1}(f(x_k) - f_*).
\end{aligned}$$

Using this in equation (22) gives

$$\begin{aligned}
\sum_{t=0}^{T-1}(f(x_t) - f_*) &\leq 4 \log \frac{2D}{\underline{D}} \left[8D\sigma\sqrt{T} + 2\sqrt{L}D\sqrt{\sum_{t=0}^{T-1}(f(x_t) - f_*)} + 32D\theta_{T,\delta}\sigma\sqrt{T}\right] \\
&\leq 8 \log \frac{2D}{\underline{D}} D\sqrt{L}\sqrt{\sum_{t=0}^{T-1}(f(x_t) - f_*)} + 160 \log \frac{2D}{\underline{D}}\theta_{T,\delta}\sigma D\sqrt{T}.
\end{aligned} \tag{23}$$

Observe that if $y^2 \leq ay + b$, then by the quadratic equation and the triangle inequality we have

$$y \leq \frac{a + \sqrt{a^2 + 4b}}{2}.$$

Squaring both sides gives

$$y^2 \leq \frac{1}{4}(a + \sqrt{a^2 + 4b})^2 \leq \frac{1}{2}\left(2a^2 + 4b\right) = a^2 + 2b. \tag{24}$$

Applying this to equation (23) with the following choices

$$y = \sqrt{\sum_{t=0}^{T-1} f(x_t) - f_*}, \qquad a = 8 \log \frac{2D}{\underline{D}} D\sqrt{L}, \qquad b = 160 \log \frac{2D}{\underline{D}}\theta_{T,\delta}\sigma D\sqrt{T},$$

then we obtain

$$\sum_{t=0}^{T-1}(f(x_t) - f_*) \leq 64 \log^2 \frac{2D}{\underline{D}} LD^2 + 320 \log^2 \frac{2D}{\underline{D}}\theta_{T,\delta}\sigma D\sqrt{T}.$$

Dividing both sides by T and using Jensen's inequality we finally get

$$\begin{aligned} f(\hat{x}_t) - f_* &\leq \frac{1}{T}\sum_{t=0}^{T-1}(f(x_t) - f_*) \\ &\leq 64\log^2\frac{2D}{\underline{D}}\frac{LD^2}{T} + 320\log^2\frac{2D}{\underline{D}}\theta_{T,\delta}\frac{\sigma D}{\sqrt{T}}. \end{aligned}$$

This shows DoG is tuning-free in this setting.

Plugging back into equation (15) we get with probability $1-\delta$ that

$$\sum_{k=0}^{t-1}\overline{r}_k\left\langle\nabla f(x_k), x_k - x_*\right\rangle \leq \overline{r}_t\left(2\overline{d}_t + \overline{r}_t\right)\sqrt{u_{t-1}} + 16\overline{d}_t\overline{r}_t\theta_{t,\delta}\sigma\sqrt{T}.$$

Now we can divide both sides by $\overline{r}_t$ to get

$$\sum_{k=0}^{t-1}\frac{\overline{r}_k}{\overline{r}_t}\left\langle\nabla f(x_k), x_k - x_*\right\rangle \leq \left(2\overline{d}_t + \overline{r}_t\right)\sqrt{u_{t-1}} + 16\overline{d}_t\theta_{t,\delta}\sigma\sqrt{T}.$$

Part 2: DoWG. By Lemma 4 we have that our iterates satisfy

$$\sum_{k=0}^{t-1}\overline{r}_k^2\left\langle\nabla f(x_k), x_k - x_*\right\rangle \leq 2\overline{r}_t\left[\overline{d}_t + \overline{r}_t\right]\sqrt{v_{t-1}} + \sum_{k=0}^{t-1}\overline{r}_k^2\left\langle\nabla f(x_k) - g_k, x_k - x_*\right\rangle$$

Define

$$X_k = \left\langle g_k - \nabla f(x_k), \frac{x_k - x_*}{\overline{d}_k}\right\rangle, \qquad \hat{X}_k = 0, \qquad y_k = \overline{r}_k^2\overline{d}_k.$$

Observe that x_k is determined by $\mathcal{F}_{k-1}$, and since $\overline{r}_k = \max_{t\leq k}\left(\|x_k - x_0\|, r_\epsilon\right)$, it is also determined by $\mathcal{F}_{k-1}$. Therefore

$$\mathbb{E}\left[X_k \mid \mathcal{F}_{k-1}\right] = \overline{r}_k^2\left\langle\mathbb{E}\left[g_k - \nabla f(x_k)\right], \frac{x_k - x_*}{\overline{d}_k}\right\rangle = 0.$$

Moreover, observe that

$$|X_k| \leq \|g_k - \nabla f(x_k)\|\frac{\|x_k - x_*\|}{\overline{d}_k} \leq \sigma.$$

Therefore the X_k form a martingale. Then we can apply Lemma 1 to get that with probability $1-\delta$ that for every $t\in[T]$

$$\begin{aligned} \left|\sum_{k=0}^{t-1}\overline{r}_k^2\left\langle g_k - \nabla f(x_k), x_k - x_*\right\rangle\right| &\leq 8\overline{d}_t\overline{r}_t^2\theta_{t,\delta}\sqrt{\sum_{k=0}^{t-1}(X_k)^2 + \sigma^2} \\ &\leq 8\overline{d}_t\overline{r}_t^2\theta_{t,\delta}\sqrt{\sum_{k=0}^{t-1}\|g_k - \nabla f(x_k)\|^2 + \sigma^2} \\ &\leq 8\overline{d}_t\overline{r}_t^2\theta_{t,\delta}\sqrt{\sigma^2 t + \sigma^2} \\ &\leq 16\overline{d}_t\overline{r}_t^2\theta_{t,\delta}\sigma\sqrt{T}. \end{aligned}$$

Plugging this back into equation (14) we get

$$\sum_{k=0}^{t-1}\overline{r}_k^2\left\langle\nabla f(x_k), x_k - x_*\right\rangle \leq 2\overline{r}_t\left[\overline{d}_t + \overline{r}_t\right]\sqrt{v_{t-1}} + 16\overline{d}_t\overline{r}_t^2\theta_{t,\delta}\sigma\sqrt{T}. \tag{25}$$

We now divide the proof in two cases:

- If f is G-Lipschitz: then $\sigma = \sup_{x\in\mathbb{R}^d} \|\nabla f(x) - g(x)\| \leq 2G$ and therefore equation (25) reduces to

$$\sum_{k=0}^{t-1} \overline{r}_k^2 \langle \nabla f(x_k), x_k - x_* \rangle \leq 2\overline{r}_t \left[\overline{d}_t + \overline{r}_t\right] \sqrt{v_{t-1}} + 32\overline{d}_t \overline{r}_t^2 \theta_{t,\delta} G\sqrt{T}.$$

And we have

$$v_{t-1} = \sum_{k=0}^{t-1} \overline{r}_k^2 \|g_k\|^2 \leq \overline{r}_t^2 G^2 T.$$

Therefore

$$\begin{aligned}\sum_{k=0}^{t-1} \overline{r}_k^2 \langle \nabla f(x_k), x_k - x_* \rangle &\leq 2\overline{r}_t^2 \left[\overline{d}_t + \overline{r}_t\right] G\sqrt{T} + 32\overline{d}_t \overline{r}_t^2 \theta_{t,\delta} G\sqrt{T} \\ &\leq 34\overline{r}_t^2 \left[\overline{d}_t + \overline{r}_t\right] \theta_{t,\delta} G\sqrt{T} \\ &\leq 68\overline{r}_t^2 DG\sqrt{T}\theta_{t,\delta}.\end{aligned}$$

Using convexity we have

$$\sum_{k=0}^{t-1} \overline{r}_k^2 (f(x_k) - f_*) \leq \sum_{k=0}^{t-1} \overline{r}_k^2 \langle \nabla f(x_k), x_k - x_* \rangle \leq 68\overline{r}_t^2 DG\sqrt{T}\theta_{t,\delta}.$$

Dividing both sides by $\sum_{k=0}^{t-1} \overline{r}_k^2$ and using Jensen's inequality we get

$$\begin{aligned}f(\tilde{x}_t) - f_* &\leq \frac{1}{\sum_{k=0}^{t-1} \overline{r}_k^2} \sum_{k=0}^{t-1} \overline{r}_k^2 (f(x_k) - f_*) \\ &\leq \frac{\overline{r}_t^2}{\sum_{k=0}^{t-1} \overline{r}_k^2} 68DG\sqrt{T}\theta_{t,\delta}. \end{aligned} \tag{26}$$

We now use Lemma 2 to conclude that there exists some $t \leq T$ such that

$$\frac{\overline{r}_t^2}{\sum_{k=0}^{t-1} \overline{r}_k^2} \leq \frac{e}{\left(\frac{T}{2\log_+ \frac{r_k}{r_\epsilon}} - 1\right)} \tag{27}$$

Note that by the fact that $\overline{r}_T \leq \overline{D}$, $r_0 = \underline{D}$, and that we assume $T \geq 4\log_+ \frac{\overline{D}}{\underline{D}}$ (see the beginning of this proof) we have

$$\frac{T}{2\log_+ \frac{\overline{r}_T}{\overline{r}_0}} - 1 \geq \frac{T}{2\log_+ \frac{\overline{D}}{\underline{D}}} - 1 \geq \frac{T}{4\log_+ \frac{\overline{D}}{\underline{D}}}.$$

Plugging this into equation (27) we get

$$\frac{\overline{r}_t^2}{\sum_{k=0}^{t-1} \overline{r}_k^2} \leq \frac{4e}{T} \log_+ \frac{\overline{D}}{\underline{D}} \leq \frac{11}{T} \log_+ \frac{\overline{D}}{\underline{D}}.$$

Using this in conjunction with equation (26) we thus have that for some $t \leq T$

$$f(\tilde{x}_t) - f_* \leq \frac{748DG\theta_{T,\delta}}{\sqrt{T}} \log_+ \frac{\overline{D}}{\underline{D}}.$$

- If f is L-smooth: Observe that by straightforward algebra, our assumption on the noise, and smoothness

$$\begin{aligned} v_{t-1} &= \sum_{k=0}^{t-1} \overline{r}_k^2 \|g_k\|^2 \\ &\leq 2\sum_{k=0}^{t-1} \overline{r}_k^2 \|g_k - \nabla f(x_k)\|^2 + 2\sum_{k=0}^{t-1} \overline{r}_k^2 \|\nabla f(x_k)\|^2 \\ &\leq 2\overline{r}_t^2 \sigma^2 T + 2\sum_{k=0}^{t-1} \overline{r}_k^2 \|\nabla f(x_k)\|^2 \\ &\leq 2\overline{r}_t^2 \sigma^2 T + 4L\sum_{k=0}^{t-1} \overline{r}_k^2 (f(x_k) - f_*). \end{aligned}$$

Using the last line estimate in equation (25) with the triangle inequality we get

$$\sum_{k=0}^{t-1} \overline{r}_k^2 \langle \nabla f(x_k), x_k - x_* \rangle \leq 4\overline{r}_t \left[\overline{d}_t + \overline{r}_t\right] \left[\overline{r}_t \sigma\sqrt{T} + \sqrt{L}\sqrt{\sum_{k=0}^{t-1} \overline{r}_k^2 (f(x_k) - f_*)}\right] + 16\overline{d}_t \overline{r}_t^2 \theta_{t,\delta} \sigma\sqrt{T}.$$

By convexity we have

$$\langle \nabla f(x_k), x_k - x_* \rangle \geq f(x_k) - f_*.$$

Therefore

$$\sum_{k=0}^{t-1} \overline{r}_k^2 (f(x_k) - f_*) \leq 4\overline{r}_t \left[\overline{d}_t + \overline{r}_t\right] \sqrt{L}\sqrt{\sum_{k=0}^{t-1} \overline{r}_k^2 (f(x_k) - f_*)} + 20\overline{r}_t^2 \theta_{t,\delta} \left[\overline{d}_t + \overline{r}_t\right] \sigma\sqrt{T}. \tag{28}$$

Observe that if $y^2 \leq ay + b$, then we have shown in equation (24) that $y^2 \leq a^2 + 2b$. Applying this to equation (28) with $a = 4\overline{r}_t \left[\overline{d}_t + \overline{r}_t\right] \sqrt{L}$ and $b = 20\overline{r}_t^2 \theta_{t,\delta} \left[\overline{d}_t + \overline{r}_t\right] \sigma\sqrt{T}$ gives

$$\begin{aligned} \sum_{k=0}^{t-1} \overline{r}_k^2 (f(x_k) - f_*) &\leq 16\overline{r}_t^2 \left[\overline{d}_t + \overline{r}_t\right]^2 L + 40\overline{r}_t^2 \theta_{t,\delta} \left[\overline{d}_t + \overline{r}_t\right] \sigma\sqrt{T} \\ &= \overline{r}_t^2 \left(16 \left[\overline{d}_t + \overline{r}_t\right]^2 L + 40\theta_{t,\delta} \left[\overline{d}_t + \overline{r}_t\right] \sigma\sqrt{T}\right). \end{aligned}$$

Dividing both sides by $\sum_{k=0}^{t-1} \overline{r}_k^2$ and using Jensen's inequality we get

$$f(\hat{x}_t) - f_* \leq \frac{1}{\sum_{k=0}^{t-1} \overline{r}_k^2} \sum_{k=0}^{t-1} \overline{r}_k^2 (f(x_k) - f_*) \leq \frac{\overline{r}_t^2}{\sum_{k=0}^{t-1} \overline{r}_k^2} \left(16 \left[\overline{d}_t + \overline{r}_t\right]^2 L + 40\theta_{t,\delta} \left[\overline{d}_t + \overline{r}_t\right] \sigma\sqrt{T}\right),$$

where $\hat{x}_t = \frac{1}{\sum_{k=0}^{t-1} \overline{r}_k^2} \sum_{k=0}^{t-1} \overline{r}_k^2 x_k$. We now use Lemma 2 to conclude that there exists some $\tau \leq T$ such that

$$f(\hat{x}_\tau) - f_* \leq \frac{e}{\left(\frac{T}{2\log_+ \frac{r_k}{r_\epsilon}} - 1\right)} \left(16 \left[\overline{d}_t + \overline{r}_t\right]^2 L + 40\theta_{\tau,\delta} \left[\overline{d}_t + \overline{r}_t\right] \sigma\sqrt{T}\right).$$

By assumption on T we have $\frac{T}{2\log_+ \frac{\overline{D}}{\underline{D}}} - 1 \geq \frac{T}{4\log_+ \frac{\overline{D}}{\underline{D}}}$, therefore

$$\begin{aligned} f(\hat{x}_\tau) - f_* &\leq \frac{4e\log_+ \frac{\overline{D}}{\underline{D}}}{T} \left(16 \left[\overline{d}_t + \overline{r}_t\right]^2 L + 40\theta_{\tau,\delta} \left[\overline{d}_t + \overline{r}_t\right] \sigma\sqrt{T}\right) \\ &\leq 700\theta_{T,\delta} \log_+ \frac{\overline{D}}{\underline{D}} \cdot \left(\frac{LD^2}{T} + \frac{\sigma D}{\sqrt{T}}\right), \end{aligned}$$

where in the last line we used that $\max(\overline{d}_t, \overline{r}_t) \leq D$.

□

8. Proofs for Section 4

8.1. Proof of Proposition 2

Proposition 2 (Hazan & Kakade (2019)). *The Adaptive Polyak algorithm from (Hazan & Kakade, 2019) is tuning-free in the deterministic setting.*

Proof. By (Hazan & Kakade, 2019, Theorem 2) we have that the point returned by the algorithm $\overline{x}$ satisfies

$$f(\overline{x}) - f_* \leq \begin{cases} \frac{2GD_*}{\sqrt{T}} \log_+ \frac{f(x_*)-\hat{f}_0}{\frac{GD_*}{\sqrt{T}}} & \text{if } f \text{ is } G\text{-Lipschitz,} \\ \frac{2LD_*^2}{T} \log_+ \frac{f(x_*)-\hat{f}_0}{\frac{LD_*^2}{T}} & \text{if } f \text{ is } L\text{-smooth.} \end{cases}$$

provided that $\hat{f}_0 \leq f_*$, where $\hat{f}_0$ is a parameter supplied to the algorithm. To get a valid lower bound on f_*, observe that by the convexity of f we have

$$f(x_0) - f_* \leq \langle \nabla f(x_0), x_0 - x_* \rangle \leq \|\nabla f(x_0)\| \, \|x_0 - x_*\| \leq \|\nabla f(x_0)\| \, \overline{D}.$$

It follows that

$$f_* \geq f(x_0) - \|\nabla f(x_0)\| \, \overline{D}.$$

And thus we can use $\hat{f}_0 = f(x_0) - \|\nabla f(x_0)\| \, \overline{D}$. □

8.2. Proof of Proposition 3

Proposition 3. *T-DoG and T-DoWG are tuning-free in the deterministic setting.*

Proof. This is shown in (Khaled et al., 2023, Supplementary material section 7) for DoWG. The proof for DoG is similar and we omit it for simplicity. □

8.3. Proof of Theorem 2

Proof. Let $\sigma > 0$. Let $L = \sigma T$. Define the functions

$$\begin{aligned} f_1(x) &\stackrel{\text{def}}{=} \frac{L}{2}x^2 + \sigma x \\ f_2(x) &\stackrel{\text{def}}{=} \frac{L}{2}x^2 - \frac{\sigma}{T-1}x \\ f(x) &\stackrel{\text{def}}{=} \frac{1}{T}f_1(x) + \left(1 - \frac{1}{T}\right) f_2(x) \\ &= \frac{L}{2}x^2. \end{aligned}$$

We shall consider the stochastic oracle $\mathcal{O}(f, \sigma_f)$ that returns function values and gradients as follows:

$$\mathcal{O}(f, \sigma_f)(x) \stackrel{\text{def}}{=} \{f_z(x), \nabla f_z(x)\} = \begin{cases} \{f_1(x), \nabla f_1(x)\} & \text{with probability } \frac{1}{T}, \\ \{f_2(x), \nabla f_2(x)\} & \text{with probability } 1 - \frac{1}{T}. \end{cases}$$

Clearly we have $\mathbb{E}\left[f_z(x)\right] = f(x)$ and $\mathbb{E}\left[\nabla f_z(x)\right] = \nabla f(x)$. Moreover,

$$\|\nabla f_1 - \nabla f(x)\| = \sigma, \qquad \|\nabla f_2(x) - \nabla f(x)\| = \frac{\sigma}{T-1} \leq \sigma.$$

It follows that $\sigma_f \leq \sigma$. Therefore $\mathcal{O}(f, \sigma_f)$ is a valid stochastic first-order oracle. This oracle is similar to the one used by Attia & Koren (2023) in their lower bound on the convergence of AdaGrad-Norm. The minimizer of the function f is clearly $x_*^f = 0$.

Let $u \geq 0$, we shall choose it later. Define

$$h_1(x) \stackrel{\text{def}}{=} \frac{L}{2}(x-u)^2 + (\sigma - (T-1)Lu)x + \frac{(T-1)L}{2}u^2,$$
$$h_2(x) \stackrel{\text{def}}{=} \frac{L}{2}(x-u)^2 + Lux - \frac{\sigma}{T-1}x - \frac{L}{2}u^2,$$
$$h(x) \stackrel{\text{def}}{=} \frac{L}{2}(x-u)^2.$$

with the oracle $\mathcal{O}(h, \sigma_h)$ given by

$$\mathcal{O}(h,\sigma_h)(x) \stackrel{\text{def}}{=} \{h_z(x), \nabla h_z(x)\} = \begin{cases} \{h_1(x), \nabla h_1(x)\} & \text{with probability } \frac{1}{T}, \\ \{h_2(x), \nabla h_2(x)\} & \text{with probability } 1 - \frac{1}{T}. \end{cases}$$

Observe that

$$\begin{aligned}
\mathbb{E}\left[h_z(x)\right] &= \frac{1}{T}\left[\frac{L}{2}(x-u)^2 + \sigma x - (T-1)Lux + \frac{(T-1)L}{2}u^2\right] \\
&\quad + \frac{T-1}{T}\left[\frac{L}{2}(x-u)^2 + Lux - \frac{\sigma x}{T-1} - \frac{L}{2}u^2\right] \\
&= \frac{L}{2}(x-u)^2 + \frac{\sigma x}{T} - \frac{T-1}{T}Lux + \frac{T-1}{T}\frac{L}{2}u^2 + \frac{T-1}{T}Lux - \frac{\sigma x}{T} - \frac{T-1}{T}\frac{L}{2}u^2 \\
&= h(x).
\end{aligned}$$

We can similarly prove that $\mathbb{E}\left[\nabla h_z(x)\right] = h(x)$. Moreover,

$$\|\nabla h_1(x) - \nabla h(x)\| = \|\sigma - (T-1)Lu\| \leq \sigma + (T-1)Lu,$$
$$\|\nabla h_2(x) - \nabla h(x)\| = \left\|\frac{-\sigma}{T-1} + Lu\right\| \leq \frac{\sigma}{T-1} + Lu.$$

It follows that $\sigma_h \leq \sigma + (T-1)Lu$, therefore $\mathcal{O}(h, \sigma_h)$ is a valid stochastic oracle. Finally, observe that the minimizer of h is $x_*^h = u$.

We fix the initialization $x_0 = v > 0$. Then the initial distance from the optimum for both f and h are:

$$D_*(f) = |v - 0| = v, \qquad D_*(h) = |v - u|. \tag{29}$$

And recall that

$$\sigma_f \leq \sigma, \qquad \sigma_h \leq \sigma + (T-1)Lu. \tag{30}$$

Observe that both f and h share the same smoothness constant L. We supply the algorithm with the following estimates:

$$\begin{aligned} \underline{L} &= L, \\ \underline{D} &= \min(v, |u-v|), \\ \underline{\sigma} &= \sigma, \end{aligned} \qquad \begin{aligned} \overline{L} &= L, \\ \overline{D} &= \max(v, |u-v|), \\ \overline{\sigma} &= \sigma + TLu. \end{aligned} \tag{31}$$

We note that in light of equations (29) and (30) and the definitions of f and h, the hints given by equation (31) are valid for both problems. Now observe the following:

$$\begin{aligned}
h_2(x) &= \frac{L}{2}(x-u)^2 + Lux - \frac{\sigma}{T-1}x - \frac{L}{2}u^2 \\
&= \frac{L}{2}(x^2 - 2ux + u^2) + Lux - \frac{\sigma}{T-1}x - \frac{L}{2}u^2 \\
&= \frac{L}{2}x^2 - \frac{\sigma}{T-1}x \\
&= f_2(x).
\end{aligned}$$

And by the linearity of expectation we have that $\nabla h_2(x) = \nabla f_2(x)$. Therefore both oracles $\mathcal{O}(f, \sigma_f)$ and $\mathcal{O}(h, \sigma_h)$ return the same stochastic gradient and stochastic function values with probability $1 - \frac{1}{T}$.

We thus have that over a run of T steps, with probability $(1 - \frac{1}{T})^T \approx e^{-1}$ the algorithm will only get the evaluations $\{h_2(x), \nabla h_2(x)\}$ from either oracle, and will get the same hints defined in equation (31). In this setting, the algorithm cannot distinguish whether it is minimizing h or minimizing f, and therefore must minimize both. This is the main idea behind this proof: we use that the algorithm is tuning-free, which gives us that the output of the algorithm x_{out} satisfies with probability $1 - \delta$

$$h(x_{\text{out}}) - h_* \leq c \cdot \text{poly}\left(\log_+ \frac{\overline{L}}{\underline{L}}, \log_+ \frac{\overline{\sigma}}{\underline{\sigma}}, \log_+ \frac{\overline{D}}{\underline{D}}, \log \frac{1}{\delta}, \log T\right)\left(\frac{LD_*(h)^2}{T} + \frac{\sigma_h D_*(h)}{\sqrt{T}}\right). \tag{32}$$

We shall let $\iota \stackrel{\text{def}}{=} \text{poly}\left(\log_+ \frac{\overline{L}}{\underline{L}}, \log_+ \frac{\overline{\sigma}}{\underline{\sigma}}, \log_+ \frac{\overline{D}}{\underline{D}}, \log \frac{1}{\delta}, \log T\right)$ and note that because all of the relevant parameters (the hints, the horizon T, and the probability δ) supplied to the algorithm are unchanged for h and f, this ι will be the same for h and f. Continuing from equation (32) and substituting the expressions for $D_*(h)$ and σ_h from equations (29) and (30) we get

$$\begin{aligned} h(x_{\text{out}}) - h_* &\leq c\iota\left(\frac{L(u-v)^2}{T} + \frac{(\sigma + (T-1)Lu)\,|u-v|}{\sqrt{T}}\right) \\ &\leq c\iota\left(\frac{L(u-v)^2}{T} + \frac{\sigma\,|u-v|}{\sqrt{T}} + \sqrt{T}Lu\,|u-v|\right). \end{aligned}$$

Using the definition of h and the fact that $h_* = 0$ we have

$$\frac{L}{2}\|x_{\text{out}} - u\|^2 \leq c\iota\left(\frac{L(u-v)^2}{T} + \frac{\sigma\,|u-v|}{\sqrt{T}} + \sqrt{T}Lu\,|u-v|\right).$$

Multiplying both sides by $\frac{2}{L}$ and then using the definition $L = \sigma T$ we get

$$\begin{aligned} \|x_{\text{out}} - u\|^2 &\leq 2c\iota\left(\frac{(u-v)^2}{T} + \frac{\sigma\,|u-v|}{\sqrt{T}L} + \sqrt{T}u\,|u-v|\right) \\ &= 2c\iota\left(\frac{(u-v)^2}{T} + \frac{|u-v|}{T^{\frac{3}{2}}} + \sqrt{T}u\,|u-v|\right) \end{aligned}$$

This gives by taking square roots and using the triangle inequality

$$|x_{\text{out}} - u| \leq \sqrt{2c\iota}\left(|u-v|\,T^{-\frac{1}{2}} + \sqrt{|u-v|}T^{-\frac{3}{4}} + T^{\frac{1}{4}}\sqrt{u\,|u-v|}\right).$$

And finally this implies

$$x_{\text{out}} \geq u - \sqrt{2c\iota}\left(|u-v|\,T^{-\frac{1}{2}} + \sqrt{|u-v|}T^{-\frac{3}{4}} + T^{\frac{1}{4}}\sqrt{u\,|u-v|}\right). \tag{33}$$

Similarly, applying the tuning-free guarantees to f and using that $D_*(f) = v$ we have

$$\frac{L}{2}\|x_{\text{out}}\|^2 = f(x_{\text{out}}) - f_* \leq c\iota\left(\frac{LD_*(f)^2}{T} + \frac{\sigma D_*(f)}{\sqrt{T}}\right) = c\iota\left(\frac{Lv^2}{T} + \frac{\sigma v}{\sqrt{T}}\right)$$

This gives

$$\|x_{\text{out}}\|^2 \leq 2c\iota\left(\frac{v^2}{T} + \frac{\sigma v}{\sqrt{T}L}\right) = 2c\iota\left(\frac{v^2}{T} + \frac{v}{T^{\frac{3}{2}}}\right).$$

Which gives

$$x_{\text{out}} \leq \sqrt{2c\iota}\left(\frac{v}{\sqrt{T}} + \frac{\sqrt{v}}{T^{\frac{3}{4}}}\right) \tag{34}$$

Now let us consider the difference between the lower bound on x_{out} given by equation (33) and the upper bound given by equation (34),

$$u - \sqrt{2c\iota}\left(|u-v|\,T^{-\frac{1}{2}} + \sqrt{|u-v|}T^{-\frac{3}{4}} + T^{\frac{1}{4}}\sqrt{u\,|u-v|}\right) - \sqrt{2c\iota}\left(\frac{v}{\sqrt{T}} + \frac{v}{T^{\frac{3}{4}}}\right) \tag{35}$$

Let us put $u = v + 1$ and $v = T^2$, then equation (35) becomes

$$\left(T^2+1\right) - \sqrt{2c\iota}\left(T^{\frac{-1}{2}} + T^{\frac{-3}{4}} + T^{\frac{1}{4}}\sqrt{T^2+1}\right) - \sqrt{2c\iota}\left(T^{2-\frac{1}{2}} + T^{2-\frac{3}{4}}\right). \tag{36}$$

Now observe that

$$\begin{aligned}
\iota &= \text{poly}\left(\log_+ \frac{\overline{L}}{\underline{L}}, \log_+ \frac{\overline{D}}{\underline{D}}, \log_+ \frac{\sigma + TLu}{\sigma}, \log_+ \frac{1}{\delta}, \log T\right) \\
&= \text{poly}\left(\log_+ 1, \log_+ T^2, \log_+(1+T^2+T^4), \log_+ \frac{1}{\delta}, \log_+ T\right) \\
&= \text{poly}\left(\log_+ T, \log_+ \frac{1}{\delta}\right).
\end{aligned}$$

We set $\delta = \frac{e^{-1}}{4}$, therefore we finally get that $\iota = \text{poly}(\log T)$, plugging back into equation (36) we get that the difference between the lower bound of equation (33) and the upper bound of equation (34) is

$$\left(T^2+1\right) - \sqrt{2c\text{poly}(\log T)}\left(T^{\frac{-1}{2}} + T^{\frac{-3}{4}} + T^{\frac{1}{4}}\sqrt{T^2+1}\right) - \sqrt{2c\text{poly}(\log T)}\left(T^{2-\frac{1}{2}} + T^{2-\frac{3}{4}}\right).$$

It is obvious that for large enough T, this expression is positive. Moreover, this situation happens with a positive probability of at least $\frac{e^{-1}}{2}$ since by the union bound

$$\begin{aligned}
\text{Prob}(\text{Algorithm incorrect for } f, h \cup \text{Oracle doesn't output all } \{h_2, \nabla h_2\}) &\leq 2\delta + \left(1 - (1 - \frac{1}{T})^T\right) \\
&\lessapprox 1 - \frac{e^{-1}}{2}.
\end{aligned}$$

By contradiction, it follows that no algorithm can be tuning-free. □

8.4. Proof of Theorem 3

Proof. We consider the following functions

$$\begin{aligned}
f(x) &= G\,|x|\,, \\
f_1(x) &= G\,|x| + Gx, \\
f_2(x) &= G\,|x| - \frac{G}{T-1}x.
\end{aligned}$$

We consider the stochastic oracle $\mathcal{O}(f, \sigma_f)$ that returns function values and gradients as follows:

$$\mathcal{O}(f,\sigma_f)(x) \overset{\text{def}}{=} \{f_z(x), \nabla f_z(x)\} = \begin{cases} \{f_1(x), \nabla f_1(x)\} & \text{with probability } \frac{1}{T}, \\ \{f_2(x), \nabla f_2(x)\} & \text{with probability } 1 - \frac{1}{T}. \end{cases}$$

Clearly we have $\mathbb{E}\left[f_z(x)\right] = f(x)$ and $\mathbb{E}\left[\nabla f_z(x)\right] = \nabla f(x)$. It is also not difficult to prove that $\|\nabla f(x) - \nabla f_z(x)\| \leq G$. We define a second function

$$\begin{aligned}
h(x) &= G\,|x-u|\,, \\
h_1(x) &= (2-T)G\,|x-u| - (T-1)G\,|x| + Gx, \\
h_2(x) &= G\,|x| - \frac{G}{T-1}x.
\end{aligned} \tag{37}$$

And we shall use the oracle $\mathcal{O}(h, \sigma_h)$ given by

$$\mathcal{O}(h,\sigma_h)(x) \stackrel{\text{def}}{=} \{h_z(x), \nabla h_z(x)\} = \begin{cases} \{h_1(x), \nabla h_1(x)\} & \text{with probability } \frac{1}{T}, \\ \{h_2(x), \nabla h_2(x)\} & \text{with probability } 1 - \frac{1}{T}. \end{cases}$$

By direct computation we have that $\mathbb{E}[h_z(x)] = h(x)$ and $\mathbb{E}[\nabla h_z(x)] = \nabla h(x)$. From the definition of the functions in equation (37) it is immediate that all the gradients and stochastic gradients are bounded by GT. It follows that $\sigma_h \leq GT$. All in all, this shows $\mathcal{O}(h, \sigma_h)$ is a valid stochastic oracle.

We set $x_0 = 1$, observe that, like in Theorem 2, with some small but constant probability both oracles return the same gradients and function values, and therefore the algorithm cannot distinguish between them. It is therefore forced to approximately minimize both, giving us the guarantee:

$$f(x_{\text{out}}) - f_* \leq c \cdot \iota \cdot \frac{G}{\sqrt{T}}$$
$$h(x_{\text{out}}) - h_* \leq c \cdot \iota \cdot \frac{(GT)\,|1-u|}{\sqrt{T}} = c\iota\,|1-u|\,G\sqrt{T}$$

This gives

$$|x_{\text{out}}| \leq \frac{c\iota}{\sqrt{T}} \tag{38}$$
$$|x_{\text{out}} - u| \leq c\iota\,|1-u|\,\sqrt{T}$$

Let us put $u = 1 - \frac{1}{T}$, then

$$\left| x - \left(1 - \frac{1}{T}\right) \right| \leq \frac{c\iota}{\sqrt{T}}$$

This implies

$$x_{\text{out}} \geq 1 - \frac{1}{T} - \frac{c\iota}{\sqrt{T}} \tag{39}$$

And equation (38) implies

$$x_{\text{out}} \leq \frac{c\iota}{\sqrt{T}} \tag{40}$$

Because $\iota = \text{poly}(\log T)$ (by direct computation), we have that the lower bound on x_{out} given by equation (39) exceeds the upper bound on the same iterate given by equation (40) as T becomes large enough, and we get our contradiction. □

9. Proofs for Section 4.2

We have the two following algorithm-independent lemmas:

Lemma 5. *Suppose that Y is a sub-exponential random variable (see Definition 9.1) with mean 0 and sub-exponential modulus R^2, i.e. for all $t > 0$*

$$\text{Prob}\left(|Y| \geq t\right) \leq 2\exp\left(-\frac{t}{R^2}\right).$$

Let $Y_1, \ldots, Y_n$ be i.i.d. copies of Y. Then with probability $1 - \delta$ it holds that

$$\left| \frac{1}{n} \sum_{i=1}^{n} Y_i \right| \leq cR^2 \left[\sqrt{\frac{1}{n} \log \frac{2}{\delta}} + \frac{1}{n} \log \frac{2}{\delta} \right],$$

where $c > 0$ is an absolute constant.

Proof. By Bernstein's inequality (Vershynin, 2018, Corollary 2.8.3) we have

$$\text{Prob}\left(\left|\frac{1}{n}\sum_{i=1}^{n} Y_i\right| \geq t\right) \leq 2\exp\left[-c\min\left(\frac{t^2}{R^4}, \frac{t}{R^2}\right)n\right],$$

for some $c > 0$. Let us set t as follows

$$t = \begin{cases} R^2\sqrt{\frac{1}{cn}\log\frac{2}{\delta}} & \text{if } \frac{1}{cn}\log\frac{2}{\delta} < 1, \\ R^2\left[\frac{1}{cn}\log\frac{2}{\delta}\right] & \text{if } \frac{1}{cn}\log\frac{2}{\delta} \geq 1. \end{cases}$$

Then

$$\frac{t}{R^2} = \begin{cases} \sqrt{\frac{1}{cn}\log\frac{2}{\delta}} & \text{if } \frac{1}{cn}\log\frac{2}{\delta} < 1, \\ \left[\frac{1}{cn}\log\frac{2}{\delta}\right] & \text{if } \frac{1}{cn}\log\frac{2}{\delta} \geq 1. \end{cases}, \qquad \frac{t^2}{R^4} = \begin{cases} \frac{1}{cn}\log\frac{2}{\delta} & \text{if } \frac{1}{cn}\log\frac{2}{\delta} < 1, \\ \left[\frac{1}{cn}\log\frac{2}{\delta}\right]^2 & \text{if } \frac{1}{cn}\log\frac{2}{\delta} \geq 1. \end{cases}$$

By combining the two cases above we get

$$\min\left(\frac{t}{R^2}, \frac{t^2}{R^4}\right) = \frac{1}{cn}\log\frac{2}{\delta}.$$

Therefore

$$2\exp\left[-c\min\left(\frac{t^2}{R^4}, \frac{t}{R^2}\right)n\right] = \delta.$$

It follows that with probability at least $1-\delta$ we have,

$$\begin{aligned}\left|\frac{1}{n}\sum_{i=1}^{n} Y_i\right| &\leq \begin{cases} R^2\sqrt{\frac{1}{cn}\log\frac{2}{\delta}} & \text{if } \frac{1}{cn}\log\frac{2}{\delta} < 1, \\ R^2\left[\frac{1}{cn}\log\frac{2}{\delta}\right] & \text{if } \frac{1}{cn}\log\frac{2}{\delta} \geq 1. \end{cases} \\ &\leq R^2\left[\sqrt{\frac{1}{cn}\log\frac{2}{\delta}} + \frac{1}{cn}\log\frac{2}{\delta}\right].\end{aligned}$$

□

Recall the definition of sub-exponential random variables:

Definition 9.1. *We call a random variable Y R-sub-exponential if*

$$\text{Prob}\left(|Y| \geq t\right) \leq 2\exp\left(\frac{-t}{R}\right)$$

for all $t \geq 0$.

Definition 9.2. *We call a random variable Y R-sub-gaussian if*

$$\text{Prob}\left(|Y| \geq t\right) \leq 2\exp\left(\frac{-t^2}{R^2}\right)$$

for all $t \geq 0$.

Lemma 6. *(Vershynin, 2018, Lemma 2.7.7) A random variable Y is R-sub-gaussian if and only if Y^2 is R^2-sub-exponential.*

Lemma 7. *(Vershynin, 2018, Exercise 2.7.10) If A is E-sub-exponential then $A - \mathbb{E}[A]$ is $c \cdot E$-sub-exponential for some absolute constant c.*

Lemma 8. *Suppose that X is a random variable that satisfies the assumptions in Definition 4.1 and $X_1, \ldots, X_n$ are all i.i.d. copies of X. Then with probability $1-\delta$ we have that*

$$\left|\sum_{i=1}^{n}(\|X_i\|^2 - \sigma^2)\right| \leq c \cdot \sigma^2 \cdot K_{\text{snr}}^{-2}\left[\sqrt{n\log\frac{1}{\delta}} + \log\frac{1}{\delta}\right].$$

Proof. By assumption we have that $\|X_i\|$ is R-sub-gaussian, therefore by Lemma 6 we have that $\|X_i\|^2$ is R^2-sub-exponential. By Lemma 7 we then have that $\|X_i\|^2 - \sigma^2$ is $c_1 \cdot R^2$-sub-exponential for some absolute constant c. By Lemma 5 applied to $Y_i = \|X_i\|^2 - \sigma^2$ we have with probability $1-\delta$ that

$$\left|\frac{1}{n}\sum_{i=1}^{n}(\|X_i\| - \sigma^2)\right| \leq c_2 \cdot (c_1 R^2)\left[\sqrt{\frac{1}{n}\log\frac{2}{\delta}} + \frac{1}{n}\log\frac{2}{\delta}\right],$$

where $c_2 > 0$ is some absolute constant. Using the definition of the signal-to-noise ratio $K_{\text{snr}}^{-1} = \frac{R}{\sigma}$ we get that for some absolute constant c

$$\left|\frac{1}{n}\sum_{i=1}^{n}(\|X_i\| - \sigma^2)\right| \leq c \cdot \sigma^2 \cdot K_{\text{snr}}^{-2}\left[\sqrt{\frac{1}{n}\log\frac{2}{\delta}} + \frac{1}{n}\log\frac{2}{\delta}\right].$$

□

9.1. Proof of Theorem 4

The main idea in the proof is the following lemma, which characterizes the convergence of the sample variance estimator of b i.i.d. random variables by the number of samples b as well as the signal-to-noise ratio K_{snr}^{-1}.

Lemma 9. *Let Y be a random vector in $\mathbb{R}^d$ such that $Z = Y - \mathbb{E}[Y]$ satisfies the assumptions in Definition 4.1. Let $Y_1, Y_2, \ldots, Y_b$ be i.i.d. copies of Y. Define the sample mean and variance as*

$$\hat{Y} = \frac{1}{b}\sum_{i=1}^{b} Y_i, \qquad \hat{\sigma}^2 = \frac{1}{b}\sum_{i=1}^{b}\left\|Y_i - \overline{Y}\right\|^2.$$

Then it holds with probability $1-\delta$ that

$$\left|\frac{\hat{\sigma}^2}{\sigma^2} - 1\right| \leq c \cdot K_{\text{snr}}^{-2} \cdot \left(\sqrt{\frac{\log\frac{2b}{\delta}}{b}} + \frac{\log\frac{2(b\vee d)}{\delta}}{b}\right),$$

where c is an absolute (non-problem-dependent) constant, $b \vee d \overset{def}{=\!=} \max(b,d)$, $\sigma^2 \overset{def}{=\!=} \mathbb{E}\left[\|Y - \mathbb{E}[Y]\|^2\right]$, and K_{snr}^{-2} is the ratio defined in Definition 4.1.

Proof. We shall use the shorthand $\mu = \mathbb{E}[Y]$. We have

$$\begin{aligned}
\hat{\sigma}^2 &= \frac{1}{b}\sum_{i=1}^{b}\left\|Y_i - \hat{Y}\right\|^2 \\
&= \frac{1}{b}\sum_{i=1}^{b}\left\|Y_i - \mu + \mu - \hat{Y}\right\|^2 \\
&= \frac{1}{b}\sum_{i=1}^{b}\left[\|Y_i - \mu\|^2 + \left\|\mu - \hat{Y}\right\|^2 + 2\left\langle Y_i - \mu, \mu - \hat{Y}\right\rangle\right] \\
&= \frac{1}{b}\sum_{i=1}^{b}\|Y_i - \mu\|^2 + \left\|\mu - \hat{Y}\right\|^2 - \frac{2}{b}\sum_{i=1}^{b}\left\langle Y_i - \mu, \hat{Y} - \mu\right\rangle
\end{aligned}$$

We have by the triangle inequality

$$\left|\hat{\sigma}^2 - \sigma^2\right| \leq \left|\frac{1}{b}\sum_{i=1}^{b}\|Y_i - \mu\|^2 - \sigma^2\right| + \left\|\mu - \hat{Y}\right\|^2 + \left|\frac{2}{b}\sum_{i=1}^{b}\left\langle Y_i - \mu, \hat{Y}^i - \mu\right\rangle\right| \tag{41}$$

By Lemma 8, we may bound the first term on the right hand side of equation (41) as

$$\left|\frac{1}{b}\sum_{i=1}^{b}\|Y_i-\mu\|^2-\sigma^2\right|\leq c\cdot\sigma^2\cdot K_{\mathrm{snr}}^{-2}\left[\sqrt{\frac{\log\frac{1}{\delta}}{b}}+\frac{\log\frac{2}{\delta}}{b}\right]. \tag{42}$$

For the second term on the right hand side of equation (41), we apply (Jin et al., 2019, Corollary 7) to $X_i=\mu-Y_i$ and obtain

$$\left\|\sum_{i=1}^{b}[\mu-Y_i]\right\|\leq c\cdot\sqrt{bR^2\log\frac{2d}{\delta}}=cR\sqrt{b\log\frac{2d}{\delta}}.$$

Squaring both sides we get

$$\left\|\sum_{i=1}^{b}[\mu-Y_i]\right\|^2\leq c^2R^2b\log\frac{2d}{\delta}$$

Therefore

$$\left\|\frac{1}{b}\sum_{i=1}^{b}[\mu-Y_i]\right\|^2\leq\frac{c^2R^2\log\frac{2d}{\delta}}{b}. \tag{43}$$

For the third term on the right hand side of equation (41) we have

$$\begin{aligned}\sum_{i=1}^{b}\left\langle Y_i-\mu,\hat{Y}-\mu\right\rangle&=\sum_{i=1}^{b}\left\langle Y_i-\mu,\frac{1}{b}\sum_{j=1}^{b}[Y_i-\mu]\right\rangle\\&=\frac{1}{b}\|Y_i-\mu\|^2+\frac{1}{b}\sum_{j\neq i}\langle Y_i-\mu,Y_j-\mu\rangle.\end{aligned}$$

Taking absolute values of both sides and using the triangle inequality we get

$$\begin{aligned}\left|\frac{1}{b}\sum_{i=1}^{b}\left\langle Y_i-\mu,\hat{Y}-\mu\right\rangle\right|&=\left|\frac{1}{b}\|Y_i-\mu\|^2+\frac{1}{b}\sum_{j\neq i}\langle Y_i-\mu,Y_j-\mu\rangle\right|\\&\leq\frac{1}{b}\|Y_i-\mu\|^2+\left|\frac{1}{b}\sum_{j\neq i}\langle Y_i-\mu,Y_j-\mu\rangle\right|.\end{aligned} \tag{44}$$

By our sub-gaussian assumption on $\|Y-\mu\|$, the first term on the right hand side of equation (44) can be bounded with high probability as

$$\|Y_i-\mu\|\leq c\sqrt{R^2\log\frac{2}{\delta}}=cR\sqrt{\log\frac{2}{\delta}}. \tag{45}$$

Define $Z_{i,j}=\langle Y_i-\mu,Y_j-\mu\rangle$. Observe that for each i, we have that the random vectors $Z_{i,1},\ldots,Z_{i,i-1},Z_{i,i+1},\cdots,Z_{i,n}$ are all independent, and therefore $\mathbb{E}[Y_{i,j}]=0$ for $i\neq j$. Observe that by the Cauchy-Schwartz inequality

$$|Z_{i,j}|=|\langle Y_i-\mu,Y_j-\mu\rangle|\leq\|Y_i-\mu\|\,\|Y_j-\mu\|.$$

Observe that each of $\|Y_i-\mu\|$ and $\|Y_j-\mu\|$ is sub-gaussian with modulus R, therefore by (Vershynin, 2018, Lemma 2.7.7) their product is sub-exponential with modulus R^2. It follows that

$$\mathrm{Prob}\left(|Z_{i,j}|\geq t\right)\leq\mathrm{Prob}\left(\|Y_i-\mu\|\,\|Y_j-\mu\|\geq t\right)\leq 2\exp\left(-\frac{t}{R^2}\right).$$

Therefore $Z_{i,j}$ is also sub-exponential with modulus R^2. By Lemma 5 we then get that for any fixed i, with probability at least $1-\delta$ we have

$$\left|\frac{1}{b-1}\sum_{\substack{j=1,\dots,b\\ j\neq i}} Z_{i,j}\right| \leq c\cdot R^2\left[\sqrt{\frac{1}{b-1}\log\frac{2}{\delta}}+\frac{1}{b-1}\log\frac{2}{\delta}\right], \tag{46}$$

for some absolute constant $c>0$. Multiplying both sides of equation (46) by $\frac{b-1}{b}$ and then using straightforward algebra we get

$$\begin{aligned}\left|\frac{1}{b}\sum_{\substack{j=1,\dots,b\\ j\neq i}} Z_{i,j}\right| &\leq cR^2\left[\sqrt{\frac{1}{b-1}\log\frac{2}{\delta}}+\frac{1}{b-1}\log\frac{2}{\delta}\right]\cdot\frac{b-1}{b}\\ &= cR^2\left[\sqrt{\frac{1}{b}\log\frac{2}{\delta}}\cdot\sqrt{\frac{b}{b-1}}+\frac{1}{b}\log\frac{2}{\delta}\cdot\frac{b}{b-1}\right]\frac{b-1}{b}\\ &= cR^2\left[\sqrt{\frac{1}{b}\log\frac{2}{\delta}}\sqrt{\frac{b-1}{b}}+\frac{1}{b}\log\frac{2}{\delta}\right]\\ &\leq cR^2\left[\sqrt{\frac{1}{b}\log\frac{2}{\delta}}+\frac{1}{b}\log\frac{2}{\delta}\right].\end{aligned} \tag{47}$$

We now use the union bound over all i with the triangle inequality to get

$$\left|\frac{1}{b}\sum_{i=1}^{n}\frac{1}{b}\sum_{\substack{j=1,\dots,b\\ j\neq i}} Z_{i,j}\right| \leq \frac{1}{b}\sum_{i=1}^{n}\left|\frac{1}{b}\sum_{\substack{j=1,\dots,b\\ j\neq i}} Z_{i,j}\right| \leq cR^2\left[\sqrt{\frac{1}{b}\log\frac{2b}{\delta}}+\frac{1}{b}\log\frac{2b}{\delta}\right]. \tag{48}$$

Combining equations (45) and (48) we get that with probability $1-\delta$ there exists some absolute constant $c'>0$

$$\left|\frac{1}{b}\sum_{i=1}^{b}\left\langle Y_i-\mu,\hat{Y}-\mu\right\rangle\right| \leq c'R^2\left[\sqrt{\frac{\log\frac{2b}{\delta}}{b}}+\frac{\log\frac{2b}{\delta}}{b}\right]. \tag{49}$$

Combining equations (42), (43) and (49) into equation (41) we get

$$\begin{aligned}\left|\hat{\sigma}^2-\sigma^2\right| &\leq \left|\frac{1}{b}\sum_{i=1}^{b}\|Y_i-\mu\|^2-\sigma^2\right|+\left\|\mu-\hat{Y}\right\|^2+\left|\frac{2}{b}\sum_{i=1}^{b}\left\langle Y_i-\mu,\hat{Y}^i-\mu\right\rangle\right|\\ &\leq c_1\cdot\sigma^2\cdot K_{\mathrm{snr}}^{-2}\left[\sqrt{\frac{\log\frac{2}{\delta}}{b}}+\frac{\log\frac{2}{\delta}}{b}\right]+c_2\frac{R^2\log\frac{2d}{\delta}}{b}+c_3R^2\left[\sqrt{\frac{\log\frac{2b}{\delta}}{b}}+\frac{\log\frac{2b}{\delta}}{b}\right].\end{aligned}$$

For some absolute constants $c_1,c_2,c_3>0$. Therefore, using the definition $K_{\mathrm{snr}}^{-1}=\frac{R}{\sigma}$ and simplifying in the last equation we finally get that

$$\left|\hat{\sigma}^2-\sigma^2\right| \leq c_4\cdot\sigma^2K_{\mathrm{snr}}^{-2}\left[\sqrt{\frac{\log\frac{2b}{\delta}}{b}}+\frac{\log\frac{2(b\vee d)}{\delta}}{b}\right],$$

for some absolute constant $c_4>0$. Dividing both sides by σ^2 yields the statement of the lemma. □

Algorithm 2 T-DoG + Variance Estimation

Require: initial point $x_0 \in \mathcal{X}$, initial distance estimate $r_\epsilon > 0$, minibatch size b, $\theta > 0$.

Initialize $r_\epsilon = \underline{D}$, $\alpha = 8^4 \cdot \log(60 \log(6T)/\delta) \cdot \theta^{-1}$.
for $t = 0, 1, 2, \ldots, T-1$ **do**
  Update distance estimator: $\overline{r}_t \leftarrow \max\left(\|x_t - x_0\|, \overline{r}_{t-1}\right)$.
  Sample b stochastic gradients $\mu_t^1, \mu_t^2, \ldots, \mu_t^b$ at x_t and compute:

$$\hat{\mu}_t = \frac{1}{b}\sum_{i=1}^{b} \mu_t^i, \qquad \hat{\sigma}_t^2 = \frac{1}{b}\sum_{i=1}^{b} \left\|\mu_t^i - \hat{\mu}_t^i\right\|^2, \qquad \overline{\sigma}_t^2 = \max_{k \le t} \hat{\sigma}_k^2.$$

  Compute a new stochastic gradient g_t evaluated at x_t.
  Update the gradient sum $u_t = u_{t-1} + \|g_t\|^2$.
  Set the stepsize:

$$\eta_t \leftarrow \frac{\overline{r}_t}{\alpha\sqrt{u_t + \beta\overline{\sigma}_t^2}} \frac{1}{\log_+^2\left(1 + \frac{u_t + \overline{\sigma}_t^2}{v_0 + \overline{\sigma}_0^2}\right)}. \tag{50}$$

  Gradient descent step: $x_{t+1} \leftarrow x_t - \eta_t \nabla f(x_t)$.
end for

Algorithm 3 T-DoWG + Variance Estimation

Require: initial point $x_0 \in \mathcal{X}$, initial distance estimate $r_\epsilon > 0$, minibatch size b, $\theta > 0$.

Initialize $r_\epsilon = \underline{D}$, $\alpha = 8^4 \cdot \log(60 \log(6T)/\delta) \cdot \theta^{-1}$.
for $t = 0, 1, 2, \ldots, T-1$ **do**
  Update distance estimator: $\overline{r}_t \leftarrow \max\left(\|x_t - x_0\|, \overline{r}_{t-1}\right)$.
  Sample b stochastic gradients $\mu_t^1, \mu_t^2, \ldots, \mu_t^b$ at x_t and compute:

$$\hat{\mu}_t = \frac{1}{b}\sum_{i=1}^{b} \mu_t^i, \qquad \hat{\sigma}_t^2 = \frac{1}{b}\sum_{i=1}^{b} \left\|\mu_t^i - \hat{\mu}_t^i\right\|^2, \qquad \overline{\sigma}_t^2 = \max_{k \le t} \hat{\sigma}_k^2.$$

  Compute a new stochastic gradient g_t evaluated at x_t.
  Update weighted gradient sum: $v_t \leftarrow v_{t-1} + \overline{r}_t^2 \|g_t\|^2$.
  Set the stepsize:

$$\gamma_t \leftarrow \frac{\overline{r}_t^2}{\alpha\sqrt{v_t + \beta\overline{r}_t^2\overline{\sigma}_t^2}} \frac{1}{\log_+^2\left(1 + \frac{v_t + \overline{r}_t^2\overline{\sigma}_t^2}{v_0 + \overline{r}_0^2\overline{\sigma}_0^2}\right)}. \tag{51}$$

  Gradient descent step: $x_{t+1} \leftarrow x_t - \gamma_t \nabla f(x_t)$.
end for

Proof of Theorem 4. First, observe that at every timestep t, conditioned on $\mathcal{F}_t = \sigma\left(g_{1:t-1}, x_{1:t}\right)$ we have by Lemma 9 that with probability $1 - \frac{\delta}{T}$ that the sample variance $\hat{\sigma}_t^2$ satisfies for some $c > 0$

$$\left|\frac{\hat{\sigma}_t^2}{\sigma^2(x_t)} - 1\right| \leq c \cdot K_{\mathrm{snr}}^{-2} \cdot \left(\sqrt{\frac{\log \frac{2bT}{\delta}}{b}} + \frac{\log \frac{2(b \vee d)T}{\delta}}{b}\right),$$

where c is an absolute constant and $\sigma_t^2 = \sigma^2(x_t)$ denotes the variance of the noise at x_t (we do not assume that the noise distribution is the same for all t). By our assumption on the minibatch size we have that for some $u \in [0, K_{\mathrm{snr}}^2]$

$$c \cdot K_{\mathrm{snr}}^{-2} \cdot \left(\sqrt{\frac{\log \frac{2bT}{\delta}}{b}} + \frac{\log \frac{2(b \vee d)T}{\delta}}{b}\right) \leq 1 - \frac{\theta}{K_{\mathrm{snr}}^2}.$$

And therefore

$$\left|\frac{\hat{\sigma}_t^2}{\sigma^2(x_t)} - 1\right| \leq 1 - \frac{\theta}{K_{\mathrm{snr}}^2}.$$

Which gives

$$\frac{\hat{\sigma}_t^2}{\sigma_t^2} \geq 1 - \left(1 - \frac{\theta}{K_{\mathrm{snr}}^2}\right) = \frac{\theta}{K_{\mathrm{snr}}^2}.$$

Multiplying both sides by σ_t^2 we get

$$\hat{\sigma}_t^2 \geq \sigma_t^2 \frac{\theta}{K_{\mathrm{snr}}^2} \geq R^2 \theta.$$

Therefore $\hat{\sigma}_t^2/\theta$ is, with high probability, an upper bound on any noise norm, and we can use that as normalization in T-DoG/T-DoWG. This is the key idea of the proof, and it's entirely owed to Lemma 9. The rest of the proof follows (Ivgi et al., 2023) with only a few changes to incorporate the variance estimation process.

Following (Ivgi et al., 2023), we define the stopping time

$$\mathcal{T}_{\mathrm{out}} = \min\left\{t \mid \overline{r}_t > 3d_0\right\}.$$

And define the proxy sequences

$$\tilde{\eta}_k = \begin{cases} \eta_k & \text{if } k < \mathcal{T}_{\mathrm{out}}, \\ 0 & \text{otherwise.} \end{cases} \qquad \tilde{\gamma}_k = \begin{cases} \gamma_k & \text{if } k < \mathcal{T}_{\mathrm{out}}, \\ 0 & \text{otherwise.} \end{cases} \tag{52}$$

Lemma 10. *(Modification of (Ivgi et al., 2023, Lemma 8)) Under the conditions of Theorem 4 both the DoG (50) and DoWG (51) updates satisfy for all $t \leq T$*

$$\begin{aligned} \rho_t &\in \sigma(g_0, \mu_0^1, \ldots, \mu_0^b \ldots, g_{t-1}, \mu_0^{t-1}, \ldots, \mu_b^{t-1}), \\ \left|\rho_t \left\langle g_t - \nabla f(x_t), x_t - x_*\right\rangle\right| &\leq \frac{6d_0^2}{8^2\theta_{T,\delta}}, \\ \sum_{k=0}^{t} \rho_k^2 \|g_k\|^2 &\leq \frac{9d_0^2}{8^4\theta_{T,\delta}}, \\ \sum_{k=0}^{t} \left(\rho_k \left\langle g_k, x_k - x_*\right\rangle\right)^2 &\leq \frac{12^2 d_0^4}{8^4\theta_{T,\delta}}, \end{aligned}$$

where ρ_t stands for either the DoG stepsize proxy $\tilde{\eta}_k$ or the DoWG stepsize proxy $\tilde{\gamma}_k$.

Proof. The modification of this lemma to account for bounded noise $g(x_k) - \nabla f(x_k)$ rather than bounded gradients is straightforward, and we omit it for simplicity. □

Lemma 11. *(Modification of (Ivgi et al., 2023, Lemma 9)) Under the conditions of Theorem 4 both the DoG (50) and DoWG (51) updates satisfy for all $t \leq T$ with probability at least $1-\delta$*

$$\sum_{k=0}^{t-1} \tilde{\eta}_k \left\langle g_k - \nabla f(x_k), x_* - x_k \right\rangle \leq d_0^2.$$

Proof. The modification is straightforward and omitted. □

Lemma 12. *(Modification of (Ivgi et al., 2023, Lemma 10)) Under the conditions of Theorem 4, if $\sum_{k=0}^{t-1} \rho_t \left\langle g_k - \nabla f(x_k), x_* - x_k \right\rangle \leq d_0^2$ for all $t \leq T$, then $\mathcal{T}_{\text{out}} > T$.*

Proof. The modification is straightforward and omitted. □

By Lemmas lemmas 11 and 12 we get that $\overline{r}_T \leq 3d_0$ and it follows that $\overline{d}_t = \max_{k \leq t} d_k \leq \max_{k \leq t} r_t + r_0 \leq 4d_0$. Then, a straightforward modification of Theorem 1 to handle the slightly smaller stepsizes used by T-DoG/T-DoWG shows that both methods are tuning-free. The proof is very similar to Theorem 1 and is omitted. □

10. Proofs for Section 5

10.1. Proof of Theorem 5

Proof. We use the exact same construction from Theorem 2 with the following hints:

$$\underline{L} = L, \qquad \overline{L} = L$$
$$\underline{\Delta} = \frac{L}{2}\min(v, |u-v|), \qquad \overline{\Delta} = \frac{L}{2}\max(v, |u-v|).$$
$$\underline{\sigma} = \sigma, \qquad \overline{\sigma} = \sigma + TLu,$$

where $u > 0$ and $v > 0$ are parameters we shall choose later. Suppose that we have that the algorithm's output point x satisfies

$$\|\nabla f(x)\|^2 \leq c\iota \left[\sqrt{\frac{L(f(x_0) - f_*)\sigma^2}{T}} + \frac{L(f(x_0) - f_*)}{T} \right].$$

We now use the fact that $f(x_0) - f_* = \frac{L}{2}(x - x_*)^2$ to get

$$\begin{aligned} L^2 \|x_{\text{out}} - x_*\|^2 &= \|\nabla f(x)\|^2 \\ &\leq c\iota \sqrt{\frac{L^2(x_0 - x_*)^2\sigma_f^2}{T}} + c\iota \frac{L^2(x_0 - x_*)^2}{T} \\ &= c\iota \frac{L\left|x_0 - x_*\right|\sigma_f}{\sqrt{T}} + c\iota \frac{L^2(x_0 - x_*)^2}{T}. \end{aligned}$$

Dividing both sides by L^2 we get

$$\|x_{\text{out}} - x_*\|^2 \leq c\iota \frac{\left|x_0 - x_*\right|\sigma_f}{L\sqrt{T}} + c\iota \frac{\|x_0 - x_*\|^2}{T}.$$

Taking square roots and using the triangle inequality gives

$$|x_{\text{out}} - x_*| \leq \sqrt{c\iota}\sqrt{|x_0 - x_*|}\sqrt{\frac{\sigma_f}{L}}\frac{1}{T^{\frac{1}{4}}} + \sqrt{c\iota}\frac{|x_0 - x_*|}{\sqrt{T}}. \tag{53}$$

Applying equation (53) to the function f with $x_0 = v > 0$, $x_* = 0$, $\sigma_f = \sigma$, and $L = \sigma\sqrt{T}$ we get

$$|x_{\text{out}}| \leq \sqrt{c\iota}\sqrt{\frac{v}{T}} + \sqrt{c\iota}\frac{v}{\sqrt{T}}.$$

Therefore

$$x_{\text{out}} \leq \sqrt{c\iota}\sqrt{\frac{v}{T}} + \sqrt{c\iota}\frac{v}{\sqrt{T}}. \tag{54}$$

On the other hand, applying equation (53) to the function $h = \frac{L}{2}(x-u)^2$ (as in the proof of Theorem 2) we obtain

$$\begin{aligned}
|x_{\text{out}} - u| &\leq \sqrt{c\iota}\sqrt{|u-v|}\sqrt{\frac{(\sigma + LTu)}{L}}\frac{1}{T^{\frac{1}{4}}} + \sqrt{c\iota}\frac{|u-v|}{\sqrt{T}} \\
&= \sqrt{c\iota}\sqrt{|u-v|}\sqrt{\frac{1}{\sqrt{T}} + Tu}\frac{1}{T^{\frac{1}{4}}} + \sqrt{c\iota}\frac{|u-v|}{\sqrt{T}} \\
&\leq \sqrt{c\iota}\sqrt{|u-v|}\left[\frac{1}{T^{\frac{1}{4}}} + \sqrt{Tu}\right]\frac{1}{T^{\frac{1}{4}}} + \sqrt{c\iota}\frac{|u-v|}{\sqrt{T}} \\
&= \sqrt{c\iota}\frac{\sqrt{|u-v|}}{\sqrt{T}} + \sqrt{c\iota}T^{\frac{1}{4}}\sqrt{u\,|u-v|} + \sqrt{c\iota}\frac{|u-v|}{\sqrt{T}}.
\end{aligned}$$

Therefore

$$x_{\text{out}} \geq u - \left[\sqrt{c\iota}\frac{\sqrt{|u-v|}}{\sqrt{T}} + \sqrt{c\iota}T^{\frac{1}{4}}\sqrt{u\,|u-v|} + \sqrt{c\iota}\frac{|u-v|}{\sqrt{T}}\right]. \tag{55}$$

Combining equations (54) and (55) gives

$$u - \left[\sqrt{c\iota}\frac{\sqrt{|u-v|}}{\sqrt{T}} + \sqrt{c\iota}T^{\frac{1}{4}}\sqrt{u\,|u-v|} + \sqrt{c\iota}\frac{|u-v|}{\sqrt{T}}\right] \leq \sqrt{c\iota}\sqrt{\frac{v}{T}} + \sqrt{c\iota}\frac{v}{\sqrt{T}}.$$

Dividing both sides by $\sqrt{c\iota}$,

$$\frac{v+\sqrt{v}}{\sqrt{T}} \geq \frac{u}{\sqrt{c\iota}} - \left[\frac{\sqrt{|u-v|}}{\sqrt{T}} + T^{\frac{1}{4}}\sqrt{u\,|u-v|} + \frac{|u-v|}{\sqrt{T}}\right]$$

Put $v = T^2$ and $u = T^2 + 1$, then we get

$$\sqrt{T} + 1 \geq \frac{T^2+1}{\sqrt{c\iota}} - \left[\frac{1}{\sqrt{T}} + T^{\frac{1}{4}}\sqrt{T^2+1} + \frac{1}{\sqrt{T}}\right].$$

For large enough T, since $\iota = \text{poly}(\log T)$, this inequality does not hold. Therefore we get our contradiction. □

10.2. Proof of Theorem 6

Theorem 7. *((Liu et al., 2023), High-probability convergence of SGD in the nonconvex setting). Let f be L-smooth and possibly nonconvex. Suppose that the stochastic gradient noise is R^2-sub-gaussian. Then for any fixed stepsize η such that $\eta L \leq 1$ we have*

$$\frac{1}{T}\sum_{t=0}^{T-1}\|\nabla f(x_t)\|^2 \leq \frac{2(f(x_0) - f_*)}{\eta T} + 5\eta R^2 + \frac{12R^2\log\frac{1}{\delta}}{T}.$$

Proof. This is a very straightforward generalization of (Liu et al., 2023, Theorem 4.1), and we include it for completeness. By (Liu et al., 2023, Corollary 4.4) we have that if $\eta_t L \leq 1$ and $0 \leq w_t\eta_t^2 L \leq \frac{1}{2R^2}$

$$\sum_{t=1}^{T}\left[w_t\eta_t\left(1 - \frac{\eta_t L}{2}\right) - v_t\right]\|\nabla f(x_t)\|^2 + w_T\Delta_{T+1} \leq w_1\Delta_1 + \left(\sum_{t=2}^{T}(w_t - w_{t-1})\Delta_t + 3R^2\sum_{t=1}^{T}\frac{w_t\eta_t^2 L}{2}\right) + \log\frac{1}{\delta}. \tag{56}$$

Choose $\eta_t = \eta$ and $w_t\eta^2 L = \frac{1}{4R^2}$, $w_t = \frac{1}{6R^2\eta}$.

$$v_t = 3R^2 w_t^2 \eta_t^2 (\eta_t L - 1)^2 = \frac{3R^2\eta^2(\eta L - 1)^2}{36R^4\eta^2} = \frac{(1-\eta L)^2}{12R^2}.$$

Then

$$\begin{aligned} w_t\eta_t\left(1 - \frac{\eta_t L}{2}\right) - v_t &= \frac{1}{6R^2}\left(1 - \frac{\eta L}{2}\right) - \frac{(1-\eta L)^2}{12R^2} \\ &= \frac{1}{6R^2}\left[(1 - \frac{\eta L}{2}) - \frac{(1-\eta L)^2}{2}\right] \\ &= \frac{1}{6R^2}\left[(1 - \frac{\eta L}{2}) - \frac{1 + \eta^2 L^2 - 2\eta L}{2}\right] \\ &= \frac{1}{12R^2}\left[1 + \eta L - \eta^2 L^2\right] \end{aligned}$$

The expression $1 + x - x^2$ is minimized for $x \in [0, 1]$ at $x = 1$ and has value 1. Therefore

$$w_t\eta_t\left(1 - \frac{\eta_t L}{2}\right) - v_t \geq \frac{1}{12R^2}.$$

Plugging into equation (56) we get

$$\sum_{t=1}^{T} \frac{1}{12R^2}\|\nabla f(x_t)\|^2 \leq \frac{\Delta_1}{6R^2\eta} + \left(\frac{3\eta}{8}T\right) + \log\frac{1}{\delta}.$$

Therefore

$$\frac{1}{T}\sum_{t=1}^{T} \|\nabla f(x_t)\|^2 \leq \frac{2\Delta_1}{\eta T} + 5\eta R^2 + \frac{12R^2 \log\frac{1}{\delta}}{T}.$$

□

10.3. Restarting SGD

We will use the following lemma from (Madden et al., 2020):

Lemma 13. *(Madden et al., 2020, Lemma 33) Let $Z = k \in \{1, 2, \ldots, K\}$ with probability p_k and $\sum_{k=1}^{K} p_k = 1$. Let $Z_1, \ldots, Z_m$ be independent copies of Z. Let $Y = (Y_1, \ldots, Y_m)$. Let $X = (X_1, \ldots, X_K)$ be a random vector on the reals independent of Z. Then for any $\gamma > 0$ we have*

$$\text{Prob}\left(\min_{k \in Y} X_k > e\eta\right) \leq \exp(-m) + \text{Prob}\left(\sum_{k=1}^{K} p_t X_k > \gamma\right)$$

Theorem 8. *(Convergence of FindLeader) If we run Algorithm 4 on a set V of P points $v_1, v_2, \ldots, v_P$, with sampling budget M and per-point estimation budget K, then the output of the algorithm satisfies for some absolute constant $c > 0$ and all $\gamma > 0$*

$$\text{Prob}\left(\|\nabla f(s_{\text{lead}})\|^2 > e\gamma + c \cdot \frac{R^2 \log\frac{2dM}{\delta}}{K}\right) \leq \delta + \exp(-M) + \text{Prob}\left(\frac{1}{P}\sum_{p=1}^{P} \|\nabla f(v_p)\|^2 > \gamma\right).$$

And

$$\|g_{m^*} - \nabla f(s_{\text{lead}})\| \leq c \cdot \frac{R^2 \log\frac{2d}{\delta}}{K}. \tag{57}$$

Algorithm 4 FindLeader(S, δ, K)

Require: set of points V, desired accuracy δ, and per-point estimation budget K.
Set $M = \log \frac{1}{\delta}$ and let $P = |V|$.
Construct the set $S = (s_1, \ldots, s_M)$ by sampling M points from $v_1, \ldots, v_P$ with replacement such that

$$\mathrm{Prob}(v_i \in S) \propto \frac{1}{\sqrt{i+1}}, \qquad \sum_{i=1}^{T} \mathrm{Prob}(v_i \in S) = 1.$$

for $m = 1$ to M **do**
    Sample K stochastic gradients $g_1^m, \ldots, g_K^m$ evaluated at s_m and compute their average

$$\hat{g}_m = \frac{1}{K} \sum_{k=1}^{K} g_k.$$

    Compute and store $h_m = \|\hat{g}_m\|$.
end for
Find the point $s_{\mathrm{lead}} \in S$ with the minimal average stochastic gradient norm:

$$m^* = \arg \min_{m \in 1,2,\ldots,M} h_m, \qquad s_{\mathrm{lead}} = S_{m^*}.$$

Return s_{lead} and its estimated gradient norm g_{m^*}.

Proof. The proof of this theorem loosely follows the proofs of (Ghadimi & Lan, 2013, Theorem 2.4) and (Madden et al., 2020, Theorem 13). First, define the following two sets of true gradients for the iterates in V and P respectively:

$$U_V = \{\nabla f(v_1), \nabla f(v_2), \ldots, \nabla f(v_P)\} \qquad U_S = \{\nabla f(s_1), \nabla f(s_2), \ldots, \nabla f(s_M)\}.$$

Lemma 13 gives us

$$\mathrm{Prob}\left(\min_{m \in 1,2,\ldots,M} \|\nabla f(s_m)\|^2 > e\gamma\right) \leq \exp(-M) + \mathrm{Prob}\left(\frac{1}{P}\sum_{p=1}^{P} \|\nabla f(v_p)\|^2 > \gamma\right)$$

We now compute how using the minimum from the stochastic estimates $\hat{g}_m$ affects the error. Fix m. Observe that because the norm of the stochastic gradient noise $\|g(x) - \nabla f(x)\|$ is sub-gaussian with modulus R^2, then using (Jin et al., 2019, Corollary 7) we get with probability at least $1 - \frac{\delta}{M}$ that for some absolute constant c_1

$$\|\hat{g}_m - \nabla f(s_m)\| \leq c_1 \cdot \sqrt{\frac{R^2 \log \frac{2dM}{\delta}}{K}}$$

Squaring both sides gives

$$\|\hat{g}_m - \nabla f(s_m)\|^2 \leq c_1 \cdot \frac{R^2 \log \frac{2dM}{\delta}}{K}.$$

Taking a union bound gives us that for all $m \in [M]$ we have with probability δ that

$$\max_{m \in [M]} \|\hat{g}_m - \nabla f(s_m)\|^2 \leq c_1 \cdot \frac{R^2 \log \frac{2dM}{\delta}}{K}. \tag{58}$$

We have by straightforward algebra

$$\begin{aligned}\min_{m\in S}\|\hat{g}_m\|^2 &\le \min_{m\in[M]}\left[\|\hat{g}_m-\nabla f(s_m)+\nabla f(s_m)\|^2\right]\\ &\le \min_{m\in[M]}\left[2\|\hat{g}_m-\nabla f(s_m)\|^2+2\|\nabla f(s_m)\|^2\right]\\ &\le \min_{m\in[M]}\left[2\max_{\alpha\in[M]}\|\hat{g}_\alpha-\nabla f(s_\alpha)\|^2+2\|\nabla f(s_m)\|^2\right]\\ &= 2\max_{m\in[M]}\|\hat{g}_m-\nabla f(s_m)\|^2+2\min_{m\in[M]}\|\nabla f(s_m)\|^2.\end{aligned}$$

Let m^* be the argmin. Then

$$\begin{aligned}\|\nabla f(s_{m^*})\|^2 &\le 2\|\nabla f(s_{m^*})-\hat{g}_{s_{m^*}}\|^2+2\|\hat{g}_{s_{m^*}}\|^2\\ &\le 2\|\nabla f(s_{m^*})-\hat{g}_{s_{m^*}}\|^2+4\max_{m\in[M]}\|\hat{g}_m-\nabla f(s_m)\|^2+4\min_{m\in[M]}\|\nabla f(s_m)\|^2\\ &\le 6\max_{m\in[M]}\|\hat{g}_m-\nabla f(s_m)\|^2+4\min_{m\in[M]}\|\nabla f(s_m)\|^2\\ &\le 6c_1\frac{R^2\log\frac{2dM}{\delta}}{K}+4\min_{m\in[M]}\|\nabla f(s_m)\|^2.\end{aligned}$$

Therefore there exists some absolute constant c such that

$$\mathrm{Prob}\left(\|\nabla f(s_{m^*})\|^2>e\gamma+c\cdot\frac{R^2\log\frac{2dM}{\delta}}{K}\right)\le\delta+\exp(-M)+\mathrm{Prob}\left(\frac{1}{P}\sum_{p=1}^{P}\|\nabla f(v_p)\|^2>\gamma\right).$$

It remains to put $s_{\mathrm{lead}}=s_{m^*}$. □

Proof of Theorem 6. First, observe that Theorem 7 gives that SGD run for T steps with a fixed stepsize η such that $\eta L\le 1$

$$\frac{1}{T}\sum_{t=0}^{T-1}\|\nabla f(x_t)\|^2\le\frac{2(f(x_0)-f_*)}{\eta T}+5\eta R^2+\frac{12R^2\log\frac{1}{\delta}}{T}. \tag{59}$$

Minimizing the above in η gives

$$\eta_*=\min\left(\frac{1}{L},\sqrt{\frac{2(f(x_0)-f_*)}{5TR^2}}\right).$$

We set

$$\eta_0=\min\left(\frac{1}{\overline{L}},\sqrt{\frac{2\underline{\Delta}}{5T\overline{R}}}\right).$$

Observe that $\eta_0\le\eta_*$. Now let

$$\begin{aligned}N^* &= \lceil\log\frac{\eta_*}{\eta_0}\rceil\\ &= \left\lceil\log\left(\frac{\max(\overline{L},\sqrt{\frac{5T\overline{R}^2}{2\underline{\Delta}}})}{\max(L,\sqrt{\frac{5TR^2}{\Delta}})}\right)\right\rceil.\end{aligned}$$

First, if we exit Algorithm 1 at line 4, i.e. if $T_{\mathrm{total}}<N$, then by the L-smoothness of f we have

$$\begin{aligned}\|\nabla f(y_0)\|^2 &\le 2L(f(y_0)-f_*)\\ &\le N\cdot\frac{2L(f(x_0)-f_*)}{T_{\mathrm{total}}}\\ &\le \log\left(\frac{\max(\overline{L},\sqrt{\frac{5T\overline{R}^2}{2\underline{\Delta}}})}{\max(L,\sqrt{\frac{5TR^2}{\Delta}})}\right)\cdot\frac{L(f(x_0)-f_*)}{T_{\mathrm{total}}}.\end{aligned}$$

This fulfills the theorem's statement. From here on our, we assume that $N \geq T_{\text{total}}$. Observe that our choice of N guarantees that $N \geq N^*$. Let τ be the first n (in the loop on line 2 of Algorithm 1) such that

$$\frac{\eta_*}{2} \leq \eta_\tau \leq \eta_*.$$

Plugging $\eta = \eta_\tau$ into Equation (59) we get with probability at least δ that

$$\begin{aligned}
\frac{1}{T}\sum_{t=0}^{T-1}\|\nabla f(x_t^\tau)\|^2 &\leq \frac{2(f(x_0)-f_*)}{\eta_\tau T} + 5\eta_\tau R^2 + \frac{12R^2\log\frac{1}{\delta}}{T} \\
&\leq \frac{4(f(x_0)-f_*)}{\eta_* T} + 5\eta_* R^2 + \frac{12R^2\log\frac{1}{\delta}}{T} \\
&\leq 2\left[\frac{2(f(x_0)-f_*)}{\eta_* T} + 5\eta_* R^2\right] + \frac{12R^2\log\frac{1}{\delta}}{T} \\
&\leq 13\left[\sqrt{\frac{L(f(x_0)-f_*)R^2}{T}} + \frac{(f(x_0)-f_*)L}{T}\right] + \frac{12R^2\log\frac{1}{\delta}}{T}. \qquad (60)
\end{aligned}$$

We now apply Theorem 8 with the parameters:

$$\begin{aligned}
V &= \left\{x_0^\tau, x_1^\tau, \ldots, x_{T-1}^\tau\right\}, \\
M &= \log\frac{1}{\delta}, \\
K &= T, \\
\gamma &= 13\left[\sqrt{\frac{L(f(x_0)-f_*)R^2}{T}} + \frac{(f(x_0)-f_*)L}{T}\right] + \frac{12R^2\log\frac{1}{\delta}}{T}.
\end{aligned}$$

The theorem combined with equation (60) gives us that with probability at least $1-4\delta$

$$\|\nabla f(y_\tau)\|^2 \leq 13\cdot e\cdot\left[\sqrt{\frac{L(f(x_0)-f_*)R^2}{T}} + \frac{(f(x_0)-f_*)L}{T}\right] + \frac{12R^2\log\frac{1}{\delta}}{T} + c\cdot\frac{R^2\log\frac{2dM}{\delta}}{T}. \qquad (61)$$

By straightforward algebra

$$\begin{aligned}
\|\hat{g}_r\|^2 = \min_{n\in[N]}\|\hat{g}_n\|^2 &\leq \min_{n\in[N]}\left[\|\hat{g}_n - \nabla f(y_n) + \nabla f(y_n)\|^2\right] \\
&\leq \min_{n\in[N]}\left[2\|\hat{g}_n - \nabla f(y_n)\|^2 + 2\|\nabla f(y_n)\|^2\right] \\
&\leq \min_{n\in[N]}\left[2\max_{\alpha\in[N]}\|\hat{g}_\alpha - \nabla f(s_\alpha)\|^2 + 2\|\nabla f(y_n)\|^2\right] \\
&= 2\max_{n\in[N]}\|\hat{g}_n - \nabla f(y_n)\|^2 + 2\min_{n\in[N]}\|\nabla f(y_n)\|^2.
\end{aligned}$$

Recall that we have $r = \arg\min_{n\in[N]}\|\hat{g}_n\|^2$, then as in the proof of Theorem 8 we have

$$\begin{aligned}
\|\nabla f(y_r)\|^2 &\leq 2\|\nabla f(y_r) - \hat{g}_{y_r}\|^2 + 2\|\hat{g}_{y_r}\|^2 \\
&\leq 2\|\nabla f(y_r) - \hat{g}_{y_r}\|^2 + 4\max_{n\in[N]}\|\hat{g}_n - \nabla f(y_n)\|^2 + 4\min_{n\in[N]}\|\nabla f(y_n)\|^2 \\
&\leq 6\max_{n\in[N]}\|\hat{g}_n - \nabla f(y_n)\|^2 + 4\min_{n\in[N]}\|\nabla f(y_n)\|^2. \qquad (62)
\end{aligned}$$

Observe that because that we passed the budget $K = T$ to the FindLeader procedure, we can use Equation (57) and the union bound to that with probability $1-\delta$,

$$\max_{n\in[N]}\|\hat{g}_n - \nabla f(y_n)\|^2 \leq c\cdot\frac{R^2\log\frac{2dN}{\delta}}{T}. \qquad (63)$$

And clearly

$$\min_{n\in[N]} \|\nabla f(y_n)\|^2 \leq \|\nabla f(y_\tau)\|^2. \tag{64}$$

Using the estimates of equations (61), (63) and (64) to upper bound the right hand side of equation (62) gives us that with probability at least $1-5\delta$

$$\|\nabla f(y_r)\|^2 \leq 6c\cdot\frac{R^2\log\frac{2dN}{\delta}}{T}+4\left[13\cdot e\cdot\left[\sqrt{\frac{L(f(x_0)-f_*)R^2}{T}}+\frac{(f(x_0)-f_*)L}{T}\right]+\frac{12R^2\log\frac{1}{\delta}}{T}+c\cdot\frac{R^2\log\frac{2dM}{\delta}}{T}\right].$$

Combining the terms and substituting in the definition of T_{total} gives the theorem's statement. □