

Subgradient Methods for Nonsmooth Convex Functions with Adversarial Errors

Martijn Gösgens ^{*1} and Bart P.G. Van Parys ^{†1}

¹CWI Amsterdam

October 6, 2025

Abstract

We consider minimizing nonsmooth convex functions with bounded subgradients. However, instead of directly observing a subgradient at every step $k \in [0, \dots, N-1]$, we assume that the optimizer receives an *adversarially corrupted* subgradient. The adversary’s power is limited to a finite corruption budget, but allows the adversary to strategically time its perturbations. We show that the classical averaged subgradient descent method, which is optimal in the noiseless case, has worst-case performance that deteriorates *quadratically* with the corruption budget. Using performance optimization programming, (i) we construct and analyze the performance of three novel subgradient descent methods, and (ii) propose a novel lower bound on the worst-case suboptimality gap of any first-order method satisfying a mild cone condition proposed by Fatkhullin et al. (2025). The worst-case performance of each of our methods degrades only *linearly* with the corruption budget. Furthermore, we show that the relative difference between their worst-case suboptimality gap and our lower bound decays as $O(\log(N)/N)$, so that all three proposed subgradient descent methods are *near-optimal*. Our methods achieve such near-optimal performance without a need for momentum or averaging. This suggests that these techniques are not necessary in this context, which is in line with recent results by Zamani and Glineur (2025).

1 Introduction

We consider the classical problem of minimizing a nonsmooth convex objective function $f : X \rightarrow \mathbb{R}$ with subgradients bounded in norm by L , i.e., $g \in \partial f(x) \implies \|g\|^2 \leq L^2$ for all x in X , a linear vector space, and g in its dual X^* . We assume that the problem is well-posed, i.e., $\min_x f(x) = f(x_*) = f_* > -\infty$ and $\|x_* - x_0\|^2 \leq R^2$ for $x_0 \in X$. We will denote the problem class collecting all such functions as $\mathcal{F}$.

Subgradient methods are particularly simple iterative algorithms which have been studied following the pioneering work of Shor (1962) with desirable properties for solving this class of optimization problems. Starting from the initial iterate x_0 , subgradient methods construct the sequence

$$\begin{aligned} x_{k+1} &= x_k - h_k \tilde{g}_k & \forall k \in [0, \dots, N-1], \\ x_{N+1} &= \frac{\sum_{k=0}^N h_k x_k}{\sum_{k=0}^N h_k} \end{aligned} \tag{1}$$

^{*}martijn.gosgens@cwi.nl

[†]bart.van.parys@cwi.nl

with subgradients $\tilde{g}_k \in \partial f(x_k)$ for a fixed step size schedule $h = (h_k)_{k \in [0, \dots, N]}$. Under the aforementioned assumptions, the performance of a generic fixed-step subgradient method satisfies for any $f \in \mathcal{F}$ the classical guarantee

$$f_{N+1} - f_\star \leq E(h) := \frac{R^2 + L^2 \sum_{k=0}^N h_k^2}{2 \sum_{k=0}^N h_k}, \quad (2)$$

where we write $f_k := f(x_k)$ for all $k \in [0, \dots, N+1]$; see for instance Boyd et al. (2003), Lan (2020).

Remarkably, the performance estimate E from (2) is a convex function of the step size schedule. This simple observation enables designing an optimized subgradient method by considering the performance optimization problem $h^\star \in \arg \min_{h \geq 0} E(h)$. This performance optimization problem admits the analytical solution

$$E(h^\star) = \frac{RL}{\sqrt{N+1}} \quad \text{with} \quad h_k^\star = \frac{R}{L\sqrt{N+1}} \quad (3)$$

for all $k \in [0, \dots, N]$, i.e., fixed-step subgradient descent with subsequent iterate averaging.

We remark, however, that from the previous it does not immediately follow that this subgradient method is optimal since the classical performance estimate (2) is not particularly sharp. Indeed, consider a vanishing step size schedule where $h_k = 0$ for all $k \in [0, \dots, N-1]$. Then trivially we have $f_{N+1} - f_\star = f_0 - f_\star \leq RL$, whereas the performance estimate is degenerate for $R > 0$. More surprisingly, Zamani and Glineur (2025) have recently shown that a non-constant step size schedule

$$h_k = \frac{R(N-k)}{L(N+1)^{3/2}} \quad (4)$$

for all $k \in [0, N-1]$ and $h_N = \infty$ (so that $x_{N+1} = x_N$) in fact enjoys the same performance guarantee $f_{N+1} - f_\star \leq E(h^\star) = RL/\sqrt{N+1}$ even though the classical performance estimate for this step size schedule is also degenerate. Alternatively, performance estimation programming initiated by Drori and Teboulle (2014) allows to exactly characterize the worst-case performance of a generic subgradient as a tractable semidefinite optimization problem. Although performance estimation programming has witnessed a surge of recent interest (Taylor et al. 2017, Das Gupta et al. 2024), finding a subgradient method with best worst-case performance results in a nonconvex performance optimization problem. In fact, verifying that a given subgradient method with step size schedule $h^\star$ enjoys the best worst-case performance is algorithmically hard. Instead, subgradient methods with equal step sizes are shown to be worst-case optimal indirectly, by showing that no black-box optimization method can guarantee better performance (Drori and Teboulle 2016).

1.1 Contributions

In this paper, we generalize these results to a setting with adversarially corrupted subgradients. That is, the optimizer does not directly observe subgradients $g_k \in \partial f(x_k)$, but instead receives corrupted subgradients

$$\tilde{g}_k = g_k + e_k.$$

We consider a bounded *corruption budget* $\sum_{k=0}^{N-1} \|e_k\|^2 \leq \gamma^2$. This means that the adversary must time its perturbations strategically. Such adversaries are of fundamental interest and have received a surge of recent attention in the optimization (Chang et al. 2022), bandit learning (Lykouris et al. 2018), and adversarial neural networks (Wang et al. 2021) communities.

Clearly, if $\gamma = 0$, then the problem studied here reduces to classical nonsmooth optimization admitting algorithms which reduce the suboptimality gap at rate $\mathcal{O}(N^{-1/2})$. On the other hand, if $\gamma \geq L\sqrt{N}$, then the adversary can fully corrupt the subgradients by the choice $e_k = -g_k$, so that $x_{N+1} = x_0$ and no progress can be made. Because of this, we study the interesting intermediate regime $\gamma \in (0, L\sqrt{N})$.

The following result illustrates that the classical subgradient descent method may suffer arbitrarily poor performance in the presence of adversarial noise:

Lemma 1. *Let $\gamma \in (0, L\sqrt{N})$. For the classical subgradient method with step sizes given in Equation (3), there exists a problem instance where*

$$f_{N+1} - f_\star \geq \frac{\gamma^2 R}{8L\sqrt{N+1}}$$

which exceeds the trivial bound RL for $\gamma > 2\sqrt{2}(N+1)^{1/4}L$ and grows unbounded for $\gamma \gg N^{1/4}$.

Lemma 1 shows that any performance guarantee for the classical subgradient descent grows at least *quadratically* with γ . However, we will show in this work that it is possible to construct a step size schedule with a performance guarantee that only grows *linearly* with γ , for $\gamma \in (0, L\sqrt{N})$.

We derive a performance estimate which yields a convex performance optimization program resulting in a (nearly) optimal subgradient method. Our performance guarantee (which we describe in Corollary 1) is of the form

$$f_{N+1} - f_\star \leq \frac{RL}{\sqrt{N+1}} u_N^{\mathbb{S}}(\gamma/L), \quad (5)$$

where $u_N^{\mathbb{S}}(\sigma) \geq 1$ can be computed by solving a convex semidefinite optimization problem. Comparing (5) to the classical bound (3), we see that the perturbations affect the suboptimality gap by a factor that only depends on N and $\sigma := \gamma/L$. To prove that (5) is close to optimal, we construct an auxiliary dual performance optimization problem with an almost matching performance lower bound. The performance lower bound holds for a class of optimization methods which satisfy the cone condition $x_0 - x_k \in \text{cone}(\tilde{g}_0, \dots, \tilde{g}_{k-1})$. This class contains any subgradient method with non-negative step sizes, the Nesterov accelerated gradient descent method and indeed most practically relevant variable step size algorithms; see also Fatkhullin et al. (2025). For any such optimization method, we prove in Corollary 2 that there exists a problem instance where

$$f_{N+1} - f_\star \geq \frac{RL}{\sqrt{N+1}} \ell_N(\sigma).$$

For $\gamma = 0$ (i.e., uncorrupted subgradients), our performance bound coincides with the known universal lower bound for (uncorrupted) nonsmooth optimization (Drori and Teboulle 2016).

To obtain analytic performance guarantees, we further bound our convex performance optimization program. In Lemma 2, this leads to explicit formulas for a step size schedule with performance guarantee

$$f_{N+1} - f_\star \leq \frac{RL}{\sqrt{N+1}} u(\sigma), \quad (6)$$

where the function $u(\sigma) \geq 1$ is defined implicitly as the solution in

$$\sigma^2 = u^2 - 1 - 2 \log u.$$

We thus see that this factor only depends on σ . The fact that the performance guarantee (6) grows linearly with σ (and hence also γ) follows from the bound $u(\sigma) \leq 1 + \sigma$. The bound (6) implies that we need $N = \mathcal{O}\left(\left((\gamma + L)\frac{R}{\varepsilon}\right)^2\right)$ iterations to achieve $f_{N+1} - f_\star \leq \varepsilon$. We further prove in Theorem 2 that for all $N \geq 1$ and $\sigma \in [0, \sqrt{N}]$, it holds that

$$\left(1 - \frac{5 \log(N+1)}{2N}\right) u(\sigma) \leq \ell_N(\sigma) \leq u_N^{\mathbb{S}}(\sigma) \leq u(\sigma).$$

This means that the performance guarantees of both the subgradient method associated with the convex semidefinite performance optimization problem and the explicit subgradient method attain a worst-case suboptimality gap which

is asymptotically equivalent to the universal lower bound.

The explicit step sizes that achieve the performance guarantee (6) are of the form

$$h_k = \frac{R(N-k)}{L(N+1)^{3/2}} \cdot \frac{u(\sigma)}{u(\sigma)^2 - (u(\sigma)^2 - 1)^{\frac{k}{N+1}}} \cdot \xi_N(\sigma), \quad (7)$$

for $k \in [0, \dots, N-1]$ and $h_N = \infty$. Here, $\xi_N(\sigma)$ is a small correction factor that we describe in Section 3. This correction factor satisfies the bounds $1 \leq \xi_N(\sigma) \leq 1 + \frac{2}{N}$ for all $N \geq 1$ and $\sigma \in [0, \sqrt{N}]$. Moreover, the method given in (7) does not use averaging or momentum, which suggests that these techniques are not necessary in this setting.

1.2 Related Work

The problem of convex optimization with *exact* subgradients has been studied extensively. In *smooth* optimization, the function f is assumed to have Lipschitz continuous gradients. Nesterov (1983) proposed *Fast Gradient Descent* (FGM), which outperforms the classical gradient descent by an order of magnitude via a *momentum* technique. This momentum technique has been further refined by *Optimized Gradient Descent* (OGM, Kim and Fessler (2016)), which improves the performance guarantee by a constant factor. We refer to Nesterov (2018) for a complete overview of convex optimization with exact (sub)gradients.

In many practical problems, it is infeasible or even impossible to obtain exact (sub)gradients. Liu and Tajbakhsh (2024) give several examples of applications where exact gradients are unavailable. For example, when gradients need to be approximated by finite difference formulas or when the evaluating f involves solving another optimization problem (Ghadimi and Wang 2018). Perhaps the most common application where computing exact gradients is infeasible comes from training machine learning models on large data sets: computing an exact gradient of the loss function requires an iteration over the entire training set, which can be prohibitively expensive. To overcome this, one can randomly sample from the training set to obtain an unbiased estimate of the gradient (Bottou 2010). This results in *Stochastic Gradient Descent* (SGD), where the gradient perturbations are modeled by random variables (Robbins and Monro 1951, Kiefer and Wolfowitz 1952). In the SGD literature, these perturbations are typically assumed to be unbiased and independent.

In other optimization problems, however, it may not be realistic to assume that the perturbations are unbiased and independent. In those settings, it makes sense to pose deterministic constraints on the perturbations and consider worst-case performance. Optimization with inexact gradients has mainly been studied in the context of smooth optimization:

Devolder et al. (2014) consider smooth optimization in a setting where both the gradient and the function value are inexact. They assume that the observed function value and gradient satisfy stage-wise constraints. It is observed that momentum methods are more vulnerable to error accumulation than standard gradient descent methods. Liu and Tajbakhsh (2024) use PEP to derive performance bounds of OGM and FGM for smooth optimization and stage-wise bounded errors, i.e., $\|e_k\| \leq \varepsilon$ for every k . Their results confirm that these momentum methods are sensitive to accumulation of errors. The considered stage-wise corruption constraints mean that the adversary does not have to time their corruptions strategically, in contrast to the total corruption budget constraint that we consider in this work.

Instead of these stage-wise corruption budgets, Chang et al. (2022) limit the adversarial power by constraining the cumulative corruption $\sum_{i=0}^k \|e_i\|$ for every $k \in [0, \dots, N-1]$. They assume the objective function f satisfies the Polyak-Łojasiewicz smoothness condition and derive performance guarantees for gradient descent methods with variable step sizes. In contrast to their step-wise cumulative corruption budgets, we consider a single total corruption budget, which allows the adversary to corrupt the first subgradients more heavily.

Schmidt et al. (2011) study *proximal gradient descent* methods where both the gradient and the proximal

operator are inexact. For constant step sizes, they prove a performance guarantee that increases quadratically with the total error $\sum_{k=0}^{N-1} \|e_k\|$. Atchadé et al. (2017, Theorem 2) extends these results to non-constant step sizes and provides sufficient conditions on the step size sequence and the perturbation sequence $\|e_k\|$ to guarantee convergence of the optimization method.

Alistarh et al. (2018) combine SGD with adversarial corruptions in a distributed setting where a fraction of the ‘workers’ provide adversarial gradient information. They show how SGD can be adapted to be robust to these *Byzantine* failures. Similarly, Wang et al. (2021) study SGD for training neural networks in a setting where a fraction of the training data has been adversarially corrupted. In the context of stochastic bandits, there have been similar efforts to robustify algorithms to adversarially corrupted output (Lykouris et al. 2018).

Notation

For sequences a_N, b_N , we write $a_N \ll b_N$ or $a_N = o(b_N)$ if $\lim_{N \rightarrow \infty} \frac{a_N}{b_N} = 0$. We write $a_N \sim b_N$ if $\lim_{N \rightarrow \infty} \frac{a_N}{b_N} = 1$, and we write $a_N = \mathcal{O}(b_N)$ if there exists a $c > 0$ such that $|a_N| \leq c|b_N|$ for all N .

2 Performance Optimization Problems

In this section, we provide optimization programs that yield upper and lower bounds on the worst-case suboptimality gap.

2.1 Admissible Subgradient Methods

Given that in the noiseless case, a simple subgradient algorithm with fixed step size schedule is optimal, it is natural to also consider these methods in the context of adversarial gradient noise. In fact, in what follows we will restrict attention to *admissible* subgradient methods ($h \in \mathcal{H}$) for which $\sum_{j=k+1}^N h_j / (N - k)$ is nondecreasing in $k \in [0, \dots, N - 1]$. The main contribution of this section is to construct a desirable performance estimate for such admissible subgradient methods. In Section 2.2 we will quantify the extent to which this restriction causes a loss of optimality.

Algorithm 1: Admissible subgradient method with step sizes h_k .

Input: Function $f : \mathbb{R}^d \rightarrow \mathbb{R}$, number of iterations N , step size schedule $h \in \mathcal{H}$ and initial iterate $x_0 \in \mathbb{R}^d$.

for $k = 0, \dots, N - 1$ **do**

Retrieve a noisy subgradient $\tilde{g}_k \in \partial f(x_k) + e_k$.

$x_{k+1} = x_k - h_k \tilde{g}_k$

Output: $x_{N+1} = \sum_{k=0}^N h_k x_k / \sum_{k=0}^N h_k$.

A characteristic property of any fixed step size subgradient method is that we may write

$$x_{N+1} = x_0 - \sum_{k=0}^{N-1} \alpha_k \tilde{g}_k. \quad (8)$$

In other words, the final iterate x_{N+1} is equal to the initial iterate and a conic combination of the noisy subgradients observed along the way. Straightforward calculation indicates that the relevant conic combination can be deduced from the step size schedule as $\alpha_k = h_k \cdot \sum_{i=k+1}^N h_i / \sum_{i=0}^N h_i \geq 0$ for all $k \in [0, \dots, N - 1]$. It is noteworthy to point out that two distinct subgradient methods for different step size schedules can be associated with the same conic combination α . We call two subgradient methods equivalent if they share the same conic combination α . We denote by

$$\mathcal{H}(\alpha) = \left\{ h \in \mathcal{H} : h_k \cdot \sum_{i=k+1}^N h_i / \sum_{i=0}^N h_i = \alpha_k \quad \forall k \in [0, \dots, N - 1] \right\}$$

the equivalence class of admissible subgradient methods associated with a particular conic combination α . A quick calculation reveals that both the classical subgradient method with averaging (see Equation (3)) as well as the subgradient method of Zamani and Glineur (2025) (see Equation (4)) satisfy Equation (8) for

$$\alpha_k^* = \frac{R(N-k)}{L\sqrt{N+1}^3}, \quad (9)$$

and are therefore equivalent. Moreover, for any $h_N \in [R/(L\sqrt{N+1}), \infty]$, we can find $h_0, \dots, h_{N-1}$ so that $h \in \mathcal{H}(\alpha^*)$. We reveal in this section that this implies that there is a manifold of optimal subgradient methods which interpolates between the classical subgradient method with averaging ($h_N = R/(L\sqrt{N+1})$) and the subgradient method from Zamani and Glineur (2025) (corresponding to $h_N = \infty$).

In the following theorem we will advance a performance estimate which depends on the step size schedule only through its associated conic combination α . That is, two equivalent subgradient methods will enjoy precisely the same performance guarantee. In particular, this suggests to design a subgradient method by optimizing over the conic combination α rather than the step size schedule directly.

Proposition 1. *Consider Algorithm 1 with step size schedule $h \in \mathcal{H}(\alpha)$. For any noise level $\gamma \geq 0$, Algorithm 1 satisfies*

$$f_{N+1} - f_\star \leq E(\alpha) :=$$

$$RL \cdot \min \left(\tau \sigma^2 + \nu_\star + \sum_{k=0}^N \nu_k \right)$$

$$\text{s.t. } \tau \in \mathbb{R}_+, \nu_k \geq 0 \quad \forall k \in [\star, 0, \dots, N],$$

$$0 \preceq \Lambda(\nu, \tau, \alpha L/R) =$$

$$\begin{pmatrix} \nu_\star & \frac{-1}{2(N+1)} & \frac{-1}{2(N+1)} & \frac{-1}{2(N+1)} & \cdots & \frac{-1}{2(N+1)} & \frac{-1}{2(N+1)} & 0 & 0 & 0 & \cdots & 0 \\ \frac{-1}{2(N+1)} & \nu_0 & \frac{\alpha_0 L}{2NR} & \frac{\alpha_0 L}{2NR} & \cdots & \frac{\alpha_0 L}{2NR} & \frac{\alpha_0 L}{2NR} & 0 & 0 & 0 & \cdots & 0 \\ \frac{-1}{2(N+1)} & \frac{\alpha_0 L}{2NR} & \nu_1 & \frac{\alpha_1 L}{2(N-1)R} & \cdots & \frac{\alpha_1 L}{2(N-1)R} & \frac{\alpha_1 L}{2(N-1)R} & \frac{\alpha_0 L}{2NR} & 0 & 0 & \cdots & 0 \\ \frac{-1}{2(N+1)} & \frac{\alpha_0 L}{2NR} & \frac{\alpha_1 L}{2(N-1)R} & \nu_2 & \cdots & \frac{\alpha_2 L}{2(N-2)R} & \frac{\alpha_2 L}{2(N-2)R} & \frac{\alpha_0 L}{2NR} & \frac{\alpha_1 L}{2(N-1)R} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{-1}{2(N+1)} & \frac{\alpha_0 L}{2NR} & \frac{\alpha_1 L}{2(N-1)R} & \frac{\alpha_2 L}{2(N-2)R} & \cdots & \nu_{N-1} & \frac{\alpha_{N-1} L}{2R} & \frac{\alpha_0 L}{2NR} & \frac{\alpha_1 L}{2(N-1)R} & \frac{\alpha_2 L}{2(N-2)R} & \cdots & 0 \\ \frac{-1}{2(N+1)} & \frac{\alpha_0 L}{2NR} & \frac{\alpha_1 L}{2(N-1)R} & \frac{\alpha_2 L}{2(N-2)R} & \cdots & \frac{\alpha_{N-1} L}{2R} & \nu_N & \frac{\alpha_0 L}{2NR} & \frac{\alpha_1 L}{2(N-1)R} & \frac{\alpha_2 L}{2(N-2)R} & \cdots & \frac{\alpha_{N-1} L}{2R} \\ \hline 0 & 0 & \frac{\alpha_0 L}{2NR} & \frac{\alpha_0 L}{2NR} & \cdots & \frac{\alpha_0 L}{2NR} & \frac{\alpha_0 L}{2NR} & \tau & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & \frac{\alpha_1 L}{2(N-1)R} & \cdots & \frac{\alpha_1 L}{2(N-1)R} & \frac{\alpha_1 L}{2(N-1)R} & 0 & \tau & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & \cdots & \frac{\alpha_2 L}{2(N-2)R} & \frac{\alpha_2 L}{2(N-2)R} & 0 & 0 & \tau & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 0 & \frac{\alpha_{N-1} L}{2R} & 0 & 0 & 0 & \cdots & \tau \end{pmatrix}.$$

Since $h = (\alpha_0, \dots, \alpha_{N-1}, \infty) \in \mathcal{H}(\alpha)$, we have that $\mathcal{H}(\alpha) \neq \emptyset \iff \alpha \geq 0$. Hence, the performance optimization problem of finding the subgradient method with best performance estimate reduces to the following convex semidefinite optimization problem:

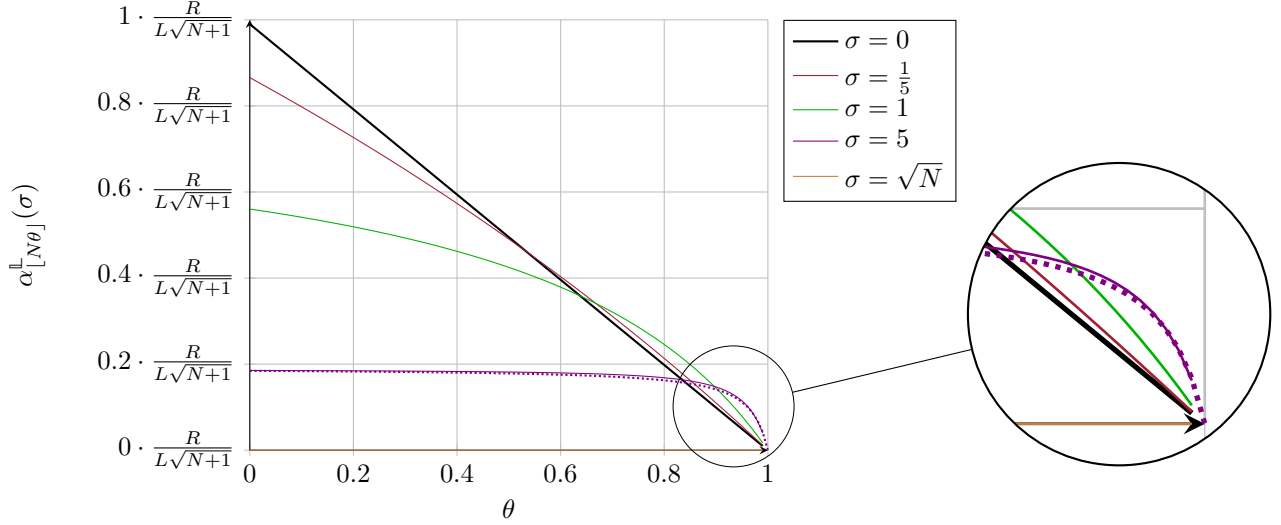


Figure 1: The conic combination $\alpha^{\mathbb{L}}$ (reindexed in $\theta \in [0, 1)$) proposed in Proposition 2 for $N = 100$ and various noise levels σ . The dashed line corresponds to the step sizes α'_k from Lemma 2 for $\sigma = 5$.

Corollary 1. Let $u_N^{\mathbb{S}}(\sigma)$ be given by

$$u_N^{\mathbb{S}}(\sigma) := \begin{cases} \sqrt{N+1} \cdot \min \tau \sigma^2 + \nu_{\star} + \sum_{k=0}^N \nu_k \\ \text{s.t. } \alpha \in \mathbb{R}_+^N, \tau \in \mathbb{R}_+, \nu_k \geq 0 \quad \forall k \in [\star, 0, \dots, N], \\ \Lambda(\nu, \tau, \alpha L/R) \succeq 0, \end{cases} \quad (10)$$

and let α^* denote the corresponding solution. Any $h \in \mathcal{H}(\alpha^*)$ enjoys the performance guarantee

$$f_{N+1} - f_{\star} \leq \frac{RL}{\sqrt{N+1}} u_N^{\mathbb{S}}(\sigma).$$

Although the performance optimization problem (10) reduces to a semidefinite optimization problem, it is difficult to analyze analytically. In the following result, we introduce a more manageable second-order cone performance optimization problem which will allow us to construct near optimal analytic step size schedules in Section 3.

Proposition 2. Consider the convex optimization problem

$$\begin{aligned} (u_N^{\mathbb{L}}(\sigma))^2 &:= \min \frac{\sigma^2 + \sum_{k=0}^N y_k}{(N+1)y_0} \\ \text{s.t. } &y_0 \geq 0, \\ &y_{k+1} \geq y_k + y_k^2 \quad \forall k \in [0, \dots, N-1]. \end{aligned} \quad (11)$$

Let

$$\alpha_k^{\mathbb{L}} = \frac{R}{L} \cdot \frac{N-k}{(N+1)^{3/2}} \cdot \frac{y_k^{\star}}{y_0^{\star} u_N^{\mathbb{L}}(\sigma)} \quad \forall k \in [0, \dots, N-1]$$

with $y^{\star}$ an optimal solution in (11). Any subgradient method with step size schedule $h \in \mathcal{H}(\alpha^{\mathbb{L}})$ enjoys the performance guarantee

$$f_{N+1} - f_{\star} \leq \frac{RL}{\sqrt{N+1}} \cdot u_N^{\mathbb{L}}(\sigma).$$

In Theorem 3 in the appendix, we further characterize the optimal $y_k^{\star}$ from Proposition 2, as well as the resulting performance guarantee. In Section 3, we provide a simpler admissible step size schedule that is asymptotically

equivalent to $\alpha_k^{\frac{1}{k}}$. Figure 1 shows the step size schedule from Proposition 2 for various σ . We see that in the noiseless case, the step sizes coincide with the known optimal step sizes from Zamani and Glineur (2025). Unsurprisingly, increased gradient corruption begets a less aggressive overall step size schedule. However, the proposed subgradient method is more cautious in earlier iterations than in later ones, where it is, in fact, more aggressive than in the absence of corruption.

2.2 Performance lower bounds

Consider any algorithm which generates iterates that satisfy the following cone condition

$$\begin{aligned} x_k &= x_0 - \text{cone}(g_0 + e_0, \dots, g_{k-1} + e_{k-1}) \quad \forall k \in [1, \dots, N], \\ x_{N+1} &= x_0 - \text{cone}(g_0 + e_0, \dots, g_{N-1} + e_{N-1}). \end{aligned} \quad (12)$$

This class of algorithms includes any subgradient method with non-negative step sizes, the Nesterov accelerated gradient descent method and most practically relevant variable step size algorithms. Intuitively, it captures any algorithm which moves into the negative of the (noisy) subgradients observed up to that point.

We now propose a lower bound on the performance of any method which satisfies Equation (12) by choosing the initial condition $x_0 - x_*$, subgradients $g_0, \dots, g_{N+1}$ and noise vectors $e_0, \dots, e_{N-1}$ adversarially. As is standard in performance estimation optimization (Drori and Teboulle 2016, Taylor et al. 2017), we will do so implicitly by considering its Grammian matrix. As the name suggests, this Grammian encodes all inner products between the variables of interest as entries in a symmetric positive semidefinite matrix G . For notational convenience, we will write $G(x_0 - x_*, g_i) := \langle x_0 - x_*, g_i \rangle$ to denote the entry related to the inner product between the initial condition $x_0 - x_*$ and the gradient g_i .

Theorem 1. *For any optimization algorithm which satisfies (12) there is a function $f \in \mathcal{F}$ so that*

$$f_{N+1} - f_* \geq \begin{cases} \max_{G \succeq 0, \Delta \geq 0} \Delta \\ \text{s.t.} \quad G(g_j, g_i) + G(e_j, g_i) = 0 \quad \forall (i, j) \in [0, \dots, N+1] \times [0, \dots, N-1] : j < i \\ G(g_j, g_i) + G(e_j, g_i) \geq 0 \quad (i, j) \in [0, \dots, N+1] \times [0, \dots, N-1] : j \geq i \\ G(x_0 - x_*, g_i) = \Delta \quad \forall i \in [0, \dots, N+1] \\ G(x_0 - x_*, x_0 - x_*) \leq R^2 \\ G(g_i, g_i) \leq L^2 \quad \forall i \in [0, \dots, N+1] \\ \sum_{i=0}^{N-1} G(e_i, e_i) \leq \gamma^2. \end{cases} \quad (13)$$

The previous result gives a lower bound on the performance of any algorithm satisfying Equation (12) in the form of a tractable convex semidefinite optimization problem. The following results will make the discussed lower bound much more explicit. Let $\nu \in [0, 1]$ be the unique solution of

$$\sum_{k=0}^{N-1} \frac{(N-k)\nu^2}{1 + (N-(k+1))\nu} = \sigma^2, \quad (14)$$

and introduce the increasing sequence

$$\gamma_k^2 := L^2 \frac{(N-k)\nu^2}{1 + (N-(k+1))\nu} \in [0, L^2)$$

for all $k \in [0, \dots, N-1]$. The following result gives a lower bound on the performance by considering only an adversaries which corrupts the subgradients by allocating their budget as $\|e_k\|^2 = \gamma_k^2$ for all $k \in [0, \dots, N-1]$.

Corollary 2. *Any optimization algorithm which satisfies (12) has the performance lower bound*

$$f_{N+1} - f_\star \geq \frac{RL}{\sqrt{N+1}} \cdot \ell_N(\sigma),$$

where $\ell_N(\sigma) = \sqrt{1 + N\nu}$ with $\nu \in [0, 1]$ the unique solution of Equation (14).

When $\sigma = 0$, then the solution of (14) is $\nu = 0$, which leads to the known bounds (3). This lower bound also matches the upper bound in Proposition 2, with $y_k = 0$ for all k , and we obtain $\alpha_k^{\mathbb{L}} = \alpha_k^*$ from (4). Proposition 2 implies in fact that the entire manifold of stepsizes $h \in \mathcal{H}(\alpha^*)$ given in (4) enjoys the same optimal worst-case suboptimality gap.

When $\sigma = \sqrt{N}$, then $\nu = 1$, which corresponds to the trivial upper bound

$$f_{N+1} - f_\star = f_0 - f_\star \leq RL,$$

This agrees with the observation that no progress is possible since the adversary can maximally corrupt the subgradient $\tilde{g}_k = g_k + e_k = 0$ for $k \in [0, \dots, N-1]$. In this regime, an optimal solution of the performance optimization problem (10) is given by $\alpha = 0$, $\nu_\star = \frac{1}{2}$, $\nu_k = \frac{1}{2}(N+1)^{-1}$ for all $k \in [0, \dots, N]$ and $\tau = 0$, resulting in a matching upper bound. This suboptimality upper bound is attained with equality for the function $f(x) = L|x|$ with $x_0 = R$.

In Section 3, we study the asymptotics of $\ell_N(\sigma)$ for $0 < \sigma < \sqrt{N}$ and compare it to the asymptotics of $u_N^{\mathbb{S}}(\sigma)$ and $u_N^{\mathbb{L}}(\sigma)$.

3 Analysis

In this section, we analyze the upper and lower bounds on the suboptimality gap presented in Proposition 2 and Corollary 2 for $\sigma \in (0, L\sqrt{N})$. Figure 2 shows the relative worst-case suboptimality performance gap

$$\max_{\sigma \in [0, \sqrt{N}]} \frac{u_N^{\{\mathbb{S}, \mathbb{L}\}}(\sigma) - \ell_N(\sigma)}{\ell_N(\sigma)} \quad (15)$$

between our upper and lower bounds on the worst-case suboptimality gap. We see that this relative difference never exceeds 1%. That is, the best subgradient method found by either Proposition 1 and Proposition 2 can not be (significantly) improved by any algorithm satisfying the cone condition in Equation (12). In the remainder of this section, we analyze this difference also via analytical techniques. In the following lemma, we present an explicit step size schedule that is admissible w.r.t. Proposition 2 and provides an upper bound for $u_N^{\{\mathbb{S}, \mathbb{L}\}}$.

We denote the *generalized harmonic numbers* by

$$H_m(a) = \sum_{k=0}^{m-1} \frac{1}{a+k},$$

so that $H_m(1)$ corresponds to the m -th harmonic number.

Lemma 2. *Consider the step sizes*

$$\alpha'_k = \frac{R}{L} \frac{N-k}{(N+1)^{3/2}} \frac{u(\sigma)}{u(\sigma)^2 - (u(\sigma)^2 - 1)^{\frac{k}{N+1}}} \sqrt{\frac{\sigma^2 + 2 \log u(\sigma)}{\sigma^2 + H_{N+1} \left(1 + \frac{N+1}{u(\sigma)^2 - 1}\right)}}$$

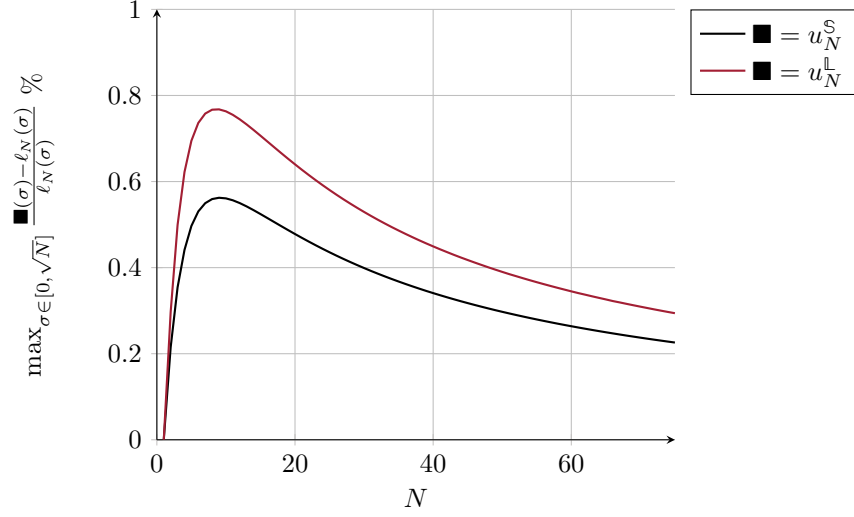


Figure 2: Relative difference between upper and lower worst-case performance bounds as a function of the number of steps N . For every N , the shown value is the maximum of the relative difference over $\sigma \in [0, \sqrt{N}]$. Theorem 2 implies that this difference decays asymptotically to zero at rate at least $\mathcal{O}(\log(N)/N)$.

where the function $u(\sigma) \geq 1$ is defined implicitly as the solution in

$$\sigma^2 = u^2 - 1 - 2 \log u. \quad (16)$$

Any subgradient method $h \in \mathcal{H}(\alpha')$ enjoys the worst-case suboptimality gap

$$f_{N+1} - f_\star \leq \frac{RL}{\sqrt{N+1}} \cdot u(\sigma).$$

Furthermore,

$$1 \leq \xi_N(\sigma) = \sqrt{\frac{\sigma^2 + 2 \log u(\sigma)}{\sigma^2 + H_{N+1} \left(1 + \frac{N+1}{u(\sigma)^2 - 1}\right)}} \leq 1 + \frac{2}{N}, \quad (17)$$

for any $\sigma \in [0, \sqrt{N}]$.

The bounds in Equation (17) show that the square root factor in the expression of α'_k is nearly negligible. Comparing the performance guarantee from Lemma 2 to that of Proposition 2, we see that the N -dependent quantity $u_N^L(\sigma)$ is replaced by $u(\sigma)$, which does not depend on N . Finally, we remark that the function u can be represented with the help of the *Lambert W-function* (Corless et al. 1996) as

$$u = \sqrt{-W_{-1}(-e^{-1-\sigma^2})},$$

where $W_{-1}(z)$ is the negative real solution in $we^w = z \in [-e^{-1}, 0)$. However, in what follows we find it more convenient to derive properties directly from its implicit representation in Equation (16).

Lemma 3. *The function $u(\sigma)$ defined by Equation (16) is convex, satisfies the bounds*

$$\max \left\{ \sqrt{2}\sigma, \sigma^2 + \log(1 + \sigma^2) \right\} \leq u(\sigma)^2 - 1 \leq \sqrt{2}\sigma + \sigma^2,$$

and has asymptotics

$$u(\sigma) = \begin{cases} \sigma + \mathcal{O}(\sigma^{-1} \log \sigma) & \text{as } \sigma \rightarrow \infty, \\ 1 + \sqrt{\frac{1}{2}}\sigma + \mathcal{O}(\sigma^2), & \text{as } \sigma \rightarrow 0. \end{cases}$$

The following theorem shows that the essential behaviour of the bounds introduced so far, i.e., u_N^S , u_N^L and ℓ_N , are all captured by the function u defined in Equation (16).

Theorem 2. *For any N and $\sigma = \gamma/L \in [0, \sqrt{N}]$, the following inequalities hold*

$$\left(1 - \frac{5 \log(N+1)}{2N}\right) u(\sigma) \leq \ell_N(\sigma) \leq u_N^S(\sigma) \leq u_N^L(\sigma) \leq u(\sigma).$$

Theorem 2 tells us that the performance of a subgradient method with associated analytic conic combination α' is asymptotically equivalent to a subgradient method with the associated conic combination α^L proposed in Proposition 2. It furthermore analytically shows that the relative worst-case suboptimality gap depicted in Figure 2 is small as indeed $(15) \leq \max_{\sigma \in [0, \sqrt{N}]} (u(\sigma) - \ell_N(\sigma)) / \ell_N(\sigma) \leq 5/2 \log(N+1)/N / (1 - 5/2 \log(N+1)/N) = \mathcal{O}(\log(N)/N)$. The next result shows that the step sizes themselves are also asymptotically equivalent for moderately small σ .

Lemma 4. *For $\sigma \ll N^{1/4}$, the optimal conic combination α^L in Proposition 2 are asymptotically equivalent to the analytic conic combination α' from Lemma 2. That is,*

$$\alpha_k^L \sim \frac{R(N-k)}{L(N+1)^{3/2}} \cdot \frac{u(\sigma)}{u(\sigma)^2 - (u(\sigma)^2 - 1)^{\frac{k}{N+1}}}, \quad (18)$$

where $u(\sigma) \geq 1$ is defined implicitly in Equation (16).

The expression (18) helps explain the shape of the step size sequences depicted in Figure 1. Let us reindex the iterations using $\theta \in [0, 1)$ via $k = \lfloor N\theta \rfloor$ and rescale it appropriately $\alpha^L(\theta) = \alpha_{\lfloor N\theta \rfloor}^L$ for $\theta \in [0, 1)$. Recall that in the special case $\sigma = 0$, we have that $\alpha_k^L(\sigma)$ coincides with the conic combination in Equation (9) and hence $\alpha^L(\theta) \sim R/L\sqrt{N+1}(1-\theta)$ corresponding to the linear line in Figure 1. More generally, from Lemma 4 it follows that

$$\alpha^L(\theta) \sim \frac{R}{L\sqrt{N+1}} \cdot \frac{1-\theta}{u(\sigma)(1 - (1 - u(\sigma)^{-2}) \cdot \theta)}$$

which is depicted as the dotted line in Figure 1 for $\sigma = 5$. From the previous we also deduce that the (near) optimal subgradient methods for $\sigma > 0$ become more aggressive than the noiseless subgradient method associated with the conic combination in Equation (9) in the regime $\theta > u(\sigma)/(1 + u(\sigma))$.

The bound given in Proposition 1 improves the trivial bound RL whenever $u_N^L(\sigma) < \sqrt{N+1}$. Using the upper bound $u_N^L(\sigma) \leq u(\sigma)$, we can guarantee that for

$$\sigma^2 < N - \log(N+1),$$

it holds that $u_N^L(\sigma) \leq u(\sigma) < \sqrt{N+1}$.

4 Discussion

In this paper we advance three subgradient methods (see Corollary 1, Proposition 2, and Lemma 2, respectively) each of which, as implied by Theorem 2, enjoy near-optimal performance in terms of their relative worst-case suboptimality gap when minimizing nonsmooth convex functions f faced with adversarial subgradient corruption. Each of these subgradient methods can hence be regarded as an inexact generalization of the classical subgradient method (3). In this section, we discuss several possible extensions of this work.

Projected Subgradient Methods. In the presence of a convex constraint $x \in C$, projected subgradient methods

$$x_{k+1} = P_C(x_k - h_k \tilde{g}_k) \quad (19)$$

are considered instead where $P_C(y) = \arg \min_{x \in C} \|y - x\|^2$ denotes the projection operator. A well known property of the projection operator guarantees that iteration (19) can be represented equivalently as

$$\|x_{k+1} - y\|^2 \leq \|x_k - h_k \tilde{g}_k - y\|^2 \quad \forall y \in C. \quad (20)$$

In the classical proof (Boyd et al. 2003, Lan 2020) of Equation (2), it is established that the claimed performance guarantee holds for any iteration scheme which satisfies merely

$$\|x_{k+1} - x^*\|^2 \leq \|x_k - h_k \tilde{g}_k - x^*\|^2 \quad \forall k \in [0, \dots, N-1]. \quad (21)$$

As performance lower bound are not affected by auxiliary constraints (the restriction C may indeed be chosen as X), it follows that through projection the subgradient methods remains worst-case optimal even when facing convex restrictions on which projection is simple. Extending the results in this paper to work with convex restrictions may at first glance seem daunting as our upper performance bound result in Lemma 6 uses in Equation (27) the affine relation $x_{k+1} = x_0 + \sum_{j=0}^k h_j \tilde{g}_j$ between iterates and subgradients which clearly fails to hold if $C \neq X$. However, inspired by Equation (20) and akin to (21), it is rather straightforward (though very tedious) to show that the result in Proposition 1 remains valid for any subgradient iterations which satisfies

$$\|x_{k+1} - y\|^2 \leq \|x_k - h_k \tilde{g}_k - y\|^2 \quad \forall y \in \{x^*, x_0, \dots, x_N\}, \quad \forall k \in [0, \dots, N-1]. \quad (22)$$

Hence, although we chose to omit the details of this generalization as not to negatively affect the exposition of this paper, all results in this paper still hold when f is restricted to C and where the projection iteration suggested in Equation (19) is used instead.

Universal Subgradient Methods. The near-optimal subgradient methods identified here depend on the problem parameters L and $\sigma = \gamma/L$. In practice, finding good values for these parameters may prove challenging. In the noiseless case, this can be addressed by *normalized subgradient descent*, where $\|g_k\|$ is substituted for L in the step sizes (3) or (4), which enjoys the same optimal (Boyd et al. 2003, Zamani and Glineur 2025) performance guarantee (2) while being adaptive to L . Generalizing this observation to inexact subgradients presents a promising direction of research but has to face the problem that the normalization $\|g_k\|$ is not observed directly, but the corrupted version $\|\tilde{g}_k\|$ will have to be used instead. We note, however, that the lower bound Corollary 2 does hold for normalized subgradient descent.

Finally, the near-optimal subgradient methods depend on the power of the adversary as characterized by σ . Unlike the Lipschitz constant L , there does not appear to be a candidate estimator for this parameter. This is a common challenge that is faced in adversarial environments. Typically, the value of σ necessarily reflects a certain amount of domain expertise which is to be taken at face value.

Smooth Convex Optimization. In this work, we focused on *nonsmooth* optimization. However, the PEP approach that is at the core of Lemma 6 and Theorem 1 has been extended to smooth (strongly) convex functions by Taylor et al. (2017), De Klerk et al. (2017). Gradient methods with optimal worst-case suboptimality $\mathcal{O}(N^{-2})$ make use of *momentum* (Kim and Fessler 2016, Nesterov 1983) whereas a simple gradient method with constant step size suffer a worst-case suboptimality of $\mathcal{O}(N^{-1})$. By allowing nonconstant step sizes, as we do here, recent concurrent work by Grimmer (2024) and Altschuler and Parrilo (2025) prove that worst-case suboptimality can be

improved to $\mathcal{O}(N^{-1-\delta})$ with $\delta \approx 0.27$ which is conjectured to be unimprovable without momentum. We expect this richer landscape of smooth convex optimization to translate to a significantly more challenging analysis when studying the impact of adversarial noise. It has been observed that momentum methods are more sensitive to error accumulation (Devolder et al. 2014, Liu and Tajbakhsh 2024) than simple gradient methods. This suggests that optimal methods for nonsmooth optimization with corrupted gradients may rely less on momentum as the level of noise increases.

References

- Dan Alistarh, Zeyuan Allen-Zhu, and Jerry Li. Byzantine stochastic gradient descent. In *Advances in Neural Information Processing Systems*, volume 31, 2018.
- Jason M. Altschuler and Pablo A. Parrilo. Acceleration by stepsize hedging: Multi-step descent and the silver stepsize schedule. *Journal of the ACM*, 72(2):1–38, 2025.
- Yves F. Atchadé, Gersende Fort, and Eric Moulines. On perturbed proximal gradient algorithms. *Journal of Machine Learning Research*, 18(10):1–33, 2017.
- Léon Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT’2010: 19th International Conference on Computational Statistics*, pages 177–186. Springer, 2010.
- Stephen Boyd, Lin Xiao, and Almir Mutapcic. Subgradient methods. Lecture notes of EE392o, Stanford University, 2003.
- Stephen P Boyd and Lieven Vandenbergh. *Convex optimization*. Cambridge university press, 2004.
- Fu-Chieh Chang, Farhang Nabiei, Pei-Yuan Wu, Alexandru Cioba, Sattar Vakili, and Alberto Bernacchia. Gradient descent: Robustness to adversarial corruption. In *Optimization for Machine Learning (NeurIPS 2022 Workshop)*, 2022.
- R. M. Corless, G. H. Gonnet, D. E. G. Hare, D. J. Jeffrey, and D. E. Knuth. On the LambertW function. *Advances in Computational Mathematics*, 5(1):329–359, 1996.
- Shuvomoy Das Gupta, Bart P. G. Van Parys, and Ernest K. Ryu. Branch-and-bound performance estimation programming: a unified methodology for constructing optimal optimization methods. *Mathematical Programming*, 204:567–639, 2024.
- Etienne De Klerk, François Glineur, and Adrien B Taylor. On the worst-case complexity of the gradient method with exact line search for smooth strongly convex functions. *Optimization Letters*, 11(7):1185–1199, 2017.
- Olivier Devolder, François Glineur, and Yurii Nesterov. First-order methods of smooth convex optimization with inexact oracle. *Mathematical Programming*, 146:37–75, 2014.
- Yoel Drori and Marc Teboulle. Performance of first-order methods for smooth convex minimization: a novel approach. *Mathematical Programming*, 145:451–482, 2014.
- Yoel Drori and Marc Teboulle. An optimal variant of Kelley’s cutting-plane method. *Mathematical Programming*, 160:321–351, 2016.
- Ilyas Fatkhullin, Florian Hübler, and Guanghui Lan. Can SGD handle heavy-tailed noise? *arXiv preprint arXiv:2508.04860*, 2025.
- Saeed Ghadimi and Mengdi Wang. Approximation methods for bilevel programming. *arXiv preprint arXiv:1802.02246*, 2018.
- Benjamin Grimmer. Provably faster gradient descent via long steps. *SIAM Journal on Optimization*, 34(3):2588–2608, 2024.
- Jack Kiefer and Jacob Wolfowitz. Stochastic estimation of the maximum of a regression function. *The Annals of Mathematical Statistics*, 23(3):462–466, 1952.
- Donghwan Kim and Jeffrey A. Fessler. Optimized first-order methods for smooth convex minimization. *Mathematical Programming*, 159:81–107, 2016.
- Guanghui Lan. *First-Order and Stochastic Optimization Methods for Machine Learning*. Springer Series in the Data Sciences. Springer, 2020.
- Yin Liu and Sam Davanloo Tajbakhsh. Nonasymptotic analysis of accelerated methods with inexact oracle under absolute error bound. *arXiv preprint arXiv:2408.00720*, 2024.

- Thodoris Lykouris, Vahab Mirrokni, and Renato Paes Leme. Stochastic bandits robust to adversarial corruptions. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 114–122, 2018.
- Yurii Nesterov. A method for solving the convex programming problem with convergence rate $O(1/k^2)$. *Dokl. Akad. Nauk SSSR*, 269:543–547, 1983.
- Yurii Nesterov. *Lectures on Convex Optimization*, volume 137 of *Springer Optimization and Its Applications*. Springer International Publishing, 2018.
- Herbert Robbins and Sutton Monro. A stochastic approximation method. *The Annals of Mathematical Statistics*, 22(3): 400–407, 1951.
- Mark Schmidt, Nicolas Roux, and Francis Bach. Convergence rates of inexact proximal-gradient methods for convex optimization. In *Advances in Neural Information Processing Systems*, volume 24, 2011.
- Naum Z. Shor. An application of the method of gradient descent to the solution of the network transportation problem. *Materialy Nauchnoy Seminara po Teoret i Priklad. Voprosam Kibernet. i Issted. Operacii, Nucnyi Sov. po Kibernet, Akad. Nauk Ukrain. SSSR, vyp*, 1:9–17, 1962.
- Adrien B. Taylor, Julien M. Hendrickx, and François Glineur. Smooth strongly convex interpolation and exact worst-case performance of first-order methods. *Mathematical Programming*, 161:307–345, 2017.
- Yunjuan Wang, Poorya Mianjy, and Raman Arora. Robust learning for data poisoning attacks. In *International Conference on Machine Learning*, pages 10859–10869. PMLR, 2021.
- Moslem Zamani and François Glineur. Exact convergence rate of the last iterate in subgradient methods. *SIAM Journal on Optimization*, 35(3):2182–2201, 2025.

A Proofs for Section 1 (Introduction)

Proof of Lemma 1. Consider the resisting function

$$f(x) = \frac{\gamma}{2\sqrt{N}}|x|.$$

Note that this function is convex, has minimizer $x_\star = 0$ and is L -Lipschitz, since

$$\|g_k\| \leq \frac{\gamma}{2\sqrt{N}} \leq \frac{L}{2}.$$

We let the adversary corrupt the subgradient with noise $e_k = -g_k - \frac{\gamma}{2\sqrt{N}}$, which is within the budget, since

$$\sum_{k=0}^{N-1} \|e_k\|^2 \leq \sum_{k=0}^{N-1} (\gamma/2\sqrt{N} + \gamma/2\sqrt{N})^2 = \gamma^2.$$

We pick $x_0 = 0$, so that the iterates become

$$x_k = x_{k-1} - h_{k-1}\tilde{g}_k = x_{k-1} + h \frac{\gamma}{2\sqrt{N}} = \frac{\gamma}{2\sqrt{N}}kh.$$

The averaging step results in

$$x_{N+1} = \frac{\gamma h}{2(N+1)\sqrt{N}} \sum_{k=0}^N k = \frac{\gamma h \sqrt{N}}{4}.$$

So that the suboptimality gap is

$$f_{N+1} - f_\star = \frac{\gamma}{2\sqrt{N}} \cdot \frac{\gamma h \sqrt{N}}{4} = \frac{\gamma^2 h}{8},$$

which exceeds the trivial bound RL for $\gamma^2 \geq 8RL/h$. The result follows after substituting $h = h^*$ from (3). $\square$

B Proofs for Section 2 (Performance Optimization Problems)

Lemma 5 (Admissible Subgradient Methods). *We have that $h \in \mathcal{H}$ if and only if there exists $\lambda'_0 \in \mathbb{R}_+, \dots, \lambda'_{N+1} \in \mathbb{R}_+$ satisfying*

$$(N-k)\lambda'_k - \sum_{i=0}^{k-1} \lambda'_i = \frac{1}{(N+1)} - \frac{h_k}{\sum_{i=0}^N h_i} \quad \forall k \in [0, \dots, N-1]. \quad (23)$$

Proof. Fix step sizes $h_0, \dots, h_N \geq 0$ and pick $\lambda'_0, \dots, \lambda'_{N+1}$ to satisfy (23) (possibly with $\lambda_k < 0$ for some values). We will prove that $\lambda_k \geq 0$ for all k is equivalent to $\sum_{j=k+1}^N h_j / (N-k)$ being nondecreasing. Summing (23) for $k = 0, \dots, j$ yields

$$(N-j) \sum_{i=0}^j \lambda'_i = \frac{j+1}{N+1} - \frac{\sum_{k=0}^j h_k}{\sum_{i=0}^N h_i} = \frac{\sum_{k=j+1}^N h_k}{\sum_{i=0}^N h_i} - \frac{N-j}{N+1},$$

which can be rewritten to

$$\left(\sum_{i=0}^N h_i \right) \sum_{i=0}^j \lambda'_i = \frac{\sum_{k=j+1}^N h_k}{N-j} - \frac{\sum_{i=0}^N h_i}{N+1}.$$

From this equation, we can see that $\sum_{i=0}^j \lambda'_i$ is nondecreasing if and only if $\frac{1}{N-j} \sum_{k=j+1}^N h_k$ is nondecreasing. Finally, the λ'_k are all nonnegative if and only if the sum $\sum_{i=0}^j \lambda'_i$ is nondecreasing and $\lambda'_0 \geq 0$. Thus, we still need

to verify $\lambda'_0 \geq 0$. For $k = 0$, Equation (23) reads

$$N\lambda'_0 = \frac{1}{N+1} - \frac{h_0}{\sum_{i=0}^N h_i},$$

which is nonnegative if and only if

$$h_0 \leq \frac{1}{N} \sum_{i=1}^N h_i.$$

This is equivalent to

$$\frac{1}{N+1} \sum_{i=0}^N h_i = \frac{1}{N+1} h_0 + \frac{N}{N+1} \left(\frac{1}{N} \sum_{i=1}^N h_i \right) \leq \left(\frac{1}{N+1} + \frac{N}{N+1} \right) \cdot \frac{1}{N} \sum_{i=1}^N h_i,$$

which completes the proof. $\square$

Lemma 6. Consider a step size schedule $h \in \mathcal{H}(\alpha)$ satisfying the condition give in Equation (23). Then, the suboptimality gap of the averaged iterate x_{N+1} satisfies the bound

$$f_{N+1} - f_* \leq \sum_{k=0}^N \frac{1}{N+1} \langle g_k, x_0 - x_* \rangle - \sum_{k=0}^N \sum_{j=0}^{k-1} \frac{\alpha_j}{N-j} \langle g_k, g_j + e_j \rangle. \quad (24)$$

Proof of Lemma 6. By convexity, the subgradients satisfy the following inequalities

$$\begin{aligned} f_* &\geq f_k + \langle g_k, x_* - x_k \rangle \quad \forall k \in [0, \dots, N] \\ f_k &\geq f_{N+1} + \langle g_{N+1}, x_k - x_{N+1} \rangle \quad \forall k \in [0, \dots, N] \\ f_i &\geq f_k + \langle g_k, x_i - x_k \rangle \quad \forall k \in [0, \dots, N], \forall i \in [0, N] : k \geq i+1. \end{aligned}$$

Summing the constraints after multiplying by $1/(N+1)$, $h_k/\sum_{i=0}^N h_i$ and $\lambda_i \geq 0$, respectively, gives

$$\begin{aligned} f_* + \frac{1}{\sum_{k=0}^N h_k} \sum_{k=0}^N h_k f_k + \sum_{i=0}^N \sum_{k=i+1}^N \lambda_i f_i &\geq \\ \frac{1}{N+1} \sum_{k=0}^N (f_k + \langle g_k, x_* - x_k \rangle) + f_{N+1} + \left\langle g_{N+1}, \frac{1}{\sum_{k=0}^N h_k} \sum_{k=0}^N h_k x_k - x_{N+1} \right\rangle &+ \sum_{i=0}^N \sum_{k=i+1}^N \lambda_i (f_k + \langle g_k, x_i - x_k \rangle). \end{aligned}$$

Our choice of x_{N+1} yields

$$\frac{1}{\sum_{k=0}^N h_k} \sum_{k=0}^N h_k x_k - x_{N+1} = 0,$$

so that one of the inner products vanishes. Interchanging the double sum and re-ordering terms yields

$$\begin{aligned} &\sum_{k=0}^N \frac{1}{N+1} \langle g_k, x_k - x_* \rangle + \sum_{k=0}^N \sum_{i=0}^{k-1} \lambda_i \langle g_k, x_k - x_i \rangle \\ &\geq f_{N+1} - f_* + \sum_{k=0}^N f_k \left(\frac{1}{N+1} - \left(\frac{h_k}{\sum_{i=0}^N h_i} \right) - (N-k)\lambda_k + \sum_{i=0}^{k-1} \lambda_i \right). \end{aligned}$$

Take now $\lambda_i \geq 0$ satisfying Equation (23) to get a valid bound. Summing Equation (23) over $k = 0, \dots, j$ gives the

equivalent set of conditions

$$(N-j) \sum_{i=0}^j \lambda_i = (j+1)/(N+1) - \sum_{k=0}^j h_k / \sum_{i=0}^N h_i \quad \forall j \in [0, \dots, N-1].$$

Or equivalently,

$$\sum_{i=0}^j \lambda_i = \frac{1}{N-j} \left(1 - \frac{N-j}{(N+1)} - \sum_{i=0}^j h_i / \sum_{i=0}^N h_i \right) = \frac{\sum_{i=j+1}^N h_i}{(N-j) \sum_{i=0}^N h_i} - \frac{1}{N+1}. \quad (25)$$

With this choice of λ_i our bound becomes

$$f_{N+1} - f_* \leq \sum_{k=0}^N \frac{1}{N+1} \langle g_k, x_k - x_* \rangle + \sum_{k=0}^N \sum_{i=0}^{k-1} \lambda_i \langle g_k, x_k - x_i \rangle. \quad (26)$$

We use

$$x_k - x_i = - \sum_{j=i}^{k-1} h_j (g_j + e_j) \quad (27)$$

to rewrite

$$\sum_{i=0}^{k-1} \lambda_i \langle g_k, x_k - x_i \rangle = - \sum_{i=0}^{k-1} \sum_{j=i}^{k-1} \lambda_i h_j \langle g_k, g_j + e_j \rangle = - \sum_{j=0}^{k-1} h_j \langle g_k, g_j + e_j \rangle \sum_{i=0}^j \lambda_i \quad (28)$$

Multiplying Equation (25) with h_j yields

$$h_j \sum_{i=0}^j \lambda_i = \frac{\alpha_j}{N-j} - \frac{h_j}{N+1}.$$

Substituting this into (28) and (26) improves our bound to

$$f_{N+1} - f_* \leq \sum_{k=0}^N \frac{1}{N+1} \langle g_k, x_k - x_* \rangle - \sum_{k=0}^N \sum_{j=0}^{k-1} \left(\frac{\alpha_j}{N-j} - \frac{h_j}{N+1} \right) \langle g_k, g_j + e_j \rangle.$$

Similarly, substituting

$$x_k = x_0 - \sum_{j=0}^{k-1} h_j (g_j + e_j)$$

yields

$$f_{N+1} - f_* \leq \sum_{k=0}^N \frac{1}{N+1} \langle g_k, x_0 - x_* \rangle - \sum_{k=0}^N \sum_{j=0}^{k-1} \frac{\alpha_j}{N-j} \langle g_k, g_j + e_j \rangle,$$

which completes the proof. $\square$

Proof of Theorem 1. The right-hand side in the bound from Lemma 6 can be rewritten to

$$\sum_{k=0}^N \frac{1}{N+1} \langle g_k, x_0 - x_* \rangle - \sum_{k=0}^N \sum_{j=0}^{k-1} \frac{\alpha_j}{N-j} \langle g_k, g_j + e_j \rangle = \langle -\Lambda(0, 0, \alpha), G \rangle$$

where the Grammian $G = [x_0 - x_* | g_0, \dots, g_N | e_0, \dots, e_{N-1}]^\top [x_0 - x_* | g_0, \dots, g_N | e_0, \dots, e_{N-1}] \succeq 0$ collects all

relevant inner products. Hence,

$$f_{N+1} - f_{\star} \leq \begin{cases} \max & \langle -\Lambda(0, 0, \alpha), G \rangle \\ \text{s.t.} & G(x_0 - x_{\star}, x_0 - x_{\star}) = \|x_0 - x_{\star}\|^2 \leq R^2 & [\text{Dual Variable : } \nu_{\star} \geq 0] \\ & G(g_k, g_k) = \|g_k\|^2 \leq L^2 & [\text{Dual Variable : } \nu_k \geq 0] \quad \forall k \in [0, \dots, N] \\ & \sum_{k=0}^{N-1} G(e_k, e_k) = \sum_{k=0}^{N-1} \|e_k\|^2 \leq \gamma^2 & [\text{Dual Variable : } \tau \geq 0]. \end{cases}$$

The claimed result now follows from a standard application Lagrangian duality. $\square$

Proof of Proposition 2. Following the partition of Λ into submatrices from Theorem 1 we write

$$\Lambda(\nu, \tau, \alpha) = \begin{pmatrix} A & B \\ B^{\top} & \tau \cdot I \end{pmatrix}.$$

By Schur's complement, $A - \frac{1}{\tau}BB^{\top} \succeq 0$ and $\tau \geq 0$ are sufficient to guarantee $\Lambda(\nu, \tau, \alpha) \succeq 0$. We now calculate Schur's complement. For $i \in [0, \dots, N]$ and $j \in [0, \dots, N-1]$, we write

$$B_{ij} = \begin{cases} 0 & \text{if } i \leq j+1, \\ \frac{\alpha_j}{2(N-j)} & \text{else.} \end{cases}$$

For $i, j \in [0, \dots, N]$, we calculate

$$(BB^{\top})_{ij} = \sum_{k=0}^{N-1} B_{ik}B_{jk} = \sum_{k=0}^{\min\{i,j\}-1} B_{ik}B_{jk} = \frac{1}{4} \sum_{k=0}^{\min\{i,j\}-1} \frac{\alpha_k^2}{(N-k)^2}.$$

Let us define

$$\mu_{\ell} = \frac{1}{4\tau} \sum_{k=0}^{\ell-1} \frac{\alpha_k^2}{(N-k)^2},$$

so that $\frac{1}{\tau}(BB^{\top})_{ij} = \mu_{\min\{i,j\}-1}$. Then Schur's complement is given by

$$\left(A - \frac{1}{\tau}BB^{\top}\right)_{ij} = \begin{cases} \nu_{\star} & \text{if } i = j = 0, \\ \frac{-1}{2(N+1)} & \text{if } i = 0 \vee j = 0, \\ \nu_{i-1} - \mu_{i-1} & \text{if } i = j > 0, \\ \frac{\alpha_{\min\{i,j\}-1}}{2(N+1-\min\{i,j\})} - \mu_{\min\{i,j\}-1} & \text{else.} \end{cases}$$

We set the dual variable ν to

$$\begin{aligned} \nu_{\star} &= \frac{N}{2(N+1)^2\tilde{\alpha}_0} \\ \nu_0 &= \frac{\alpha_0}{2N} \\ \nu_{k+1} &= \nu_k + \frac{\alpha_k^2}{4(N-k)^2\tau} \quad \forall k \in [0, \dots, N-1]. \end{aligned}$$

With these variables, the diagonal entries are

$$\left(A - \frac{1}{\tau}BB^\top\right)_{ii} = \begin{cases} \frac{N}{2(N+1)^2\alpha_0} & \text{if } i = 0, \\ \frac{\alpha_0}{2N} & \text{else.} \end{cases}$$

Again applying Schur, we can write

$$A - \frac{1}{\tau}BB^\top = \begin{pmatrix} \nu_* & \frac{1}{2(N+1)} \cdot \mathbf{1}^\top \\ \frac{1}{2(N+1)} \cdot \mathbf{1} & C \end{pmatrix},$$

where

$$C_{ij} = \begin{cases} \frac{\alpha_0}{2N} & \text{if } i = j, \\ \frac{\alpha_{\min\{i,j\}}}{2(N - \min\{i,j\})} - \mu_{\min\{i,j\}} & \text{else.} \end{cases}$$

Then $A - \frac{1}{\tau}BB^\top$ is positive semidefinite iff

$$C - \frac{\alpha_0}{2N} \mathbf{1} \cdot \mathbf{1}^\top$$

is positive semidefinite. This matrix has a zero diagonal. The off-diagonal elements are given by

$$\frac{\alpha_{\min\{i,j\}}}{2(N - \min\{i,j\})} - \mu_{\min\{i,j\}} - \frac{\alpha_0}{2N}.$$

Suppose that

$$\frac{\alpha_{k+1}}{N - k - 1} = \frac{\alpha_k}{N - k} + \frac{\alpha_k^2}{2(N - k)^2\tau} \quad \forall k \in [0, \dots, N - 2]$$

then the off-diagonal elements are zero. Indeed, for $\min\{i, j\} = 0$ we have trivially,

$$\frac{\alpha_0}{2N} - \mu_0 - \frac{\alpha_0}{2N} = 0.$$

Observe that for any k we have

$$\begin{aligned} & \frac{\alpha_{k+1}}{2(N - k - 1)} - \mu_{k+1} - \frac{\alpha_0}{2N} \\ &= \frac{\alpha_{k+1}}{2(N - k - 1)} - \mu_k - \frac{\alpha_k^2}{4\tau(N - k)^2} - \frac{\alpha_0}{2N} \\ &= \frac{\alpha_k}{2(N - k)} + \frac{\alpha_k^2}{4(N - k)^2\tau} - \mu_k - \frac{\alpha_k^2}{4\tau(N - k)^2} - \frac{\alpha_0}{2N} \\ &= \frac{\alpha_k}{2(N - k)} - \mu_k - \frac{\alpha_0}{2N} \end{aligned}$$

and hence by induction on $\min\{i, j\} = 0$ all off-diagonal elements are zero and hence the matrix is positive semidefinite. Hence, an upper bound on the performance of the best subgradient method is given as

$$\begin{aligned} f_{N+1} - f^* &\leq \min \quad \tau\gamma^2 + \frac{NR^2}{2(N+1)^2\alpha_0} + \sum_{k=0}^N \nu_k L^2 \\ \text{s.t.} \quad &\tau \geq 0, \nu \geq 0, \alpha \geq 0, \\ &\nu_0 = \frac{\alpha_0}{2N}, \\ &\nu_{k+1} \geq \nu_k + \frac{\alpha_k^2}{4(N-k)^2\tau} \quad \forall k \in [0, \dots, N-1] \\ &\frac{\alpha_{k+1}}{N-k-1} \geq \frac{\alpha_k}{N-k} + \frac{\alpha_k^2}{2(N-k)^2\tau} \quad \forall k \in [0, \dots, N-1]. \end{aligned}$$

After the change of variables $y_k = \alpha_k/2(N-k)\tau$ and letting $\nu_k = y_k/2$ for all $k \in [0, \dots, N-1]$ we get

$$\begin{aligned} f_{N+1} - f^* &\leq \begin{cases} \min & \tau\gamma^2 + \frac{R^2}{4(N+1)^2 y_0 \tau} + \tau \sum_{k=0}^N y_k L^2 \\ \text{s.t.} & \tau \geq 0, y_0 \geq 0, \\ & y_{k+1} \geq y_k + y_k^2 \quad \forall k \in [0, \dots, N-1] \end{cases} \\ = \frac{RL}{\sqrt{N+1}} \cdot u_N^{\mathbb{L}}(\sigma) &= \begin{cases} \min & \sqrt{R^2(\gamma^2 + \sum_{k=0}^N y_k L^2)/((N+1)^2 y_0)} \\ \text{s.t.} & y_0 \geq 0, \\ & y_{k+1} \geq y_k + y_k^2 \quad \forall k \in [0, \dots, N-1] \end{cases} \end{aligned}$$

establishing the claim. $\square$

The objective function in the performance optimization problem (11) is nondecreasing in $y_1, \dots, y_{N-1}$. Therefore, an optimal y can also be found which satisfy the recursion

$$y_{k+1} = y_k + y_k^2. \quad (29)$$

for all $k \in [0, \dots, N-1]$. The sequence y_k grows monotonously and once it exceeds 1, its growth becomes doubly-exponential. This can be seen from $y_{k+1} > y_k^2$ which implies $y_k > y_0^{2^k}$, resulting in doubly-exponential growth for $y_0 > 1$. For $0 < y_0 < 1$, we have $y_k \geq y_0 + ky_0^2$, so that y_k will eventually exceed 1 and start its doubly-exponential growth. Finally, we denote with $S_N(y_0) = \sum_{k=0}^N y_k$ the associated partial sum.

We introduce first the following lemma to study the asymptotics of the recursion in Equation (29).

Lemma 7. *For a starting point $y_0 \geq 0$, we define a sequence $(y_k)_{k \in \mathbb{N}}$ satisfying recursion (29). We have*

$$y_0 \cdot \sum_{r=0}^{2^N} (k)_r y_0^r \leq y_k \leq y_0 \cdot \sum_{r=0}^{2^N} k^r y_0^r,$$

where $(k)_r = k \cdot (k-1) \cdots (k-r+1)$ is the falling factorial. Moreover,

$$\sum_{r=1}^{\infty} \frac{1}{r} (N+1)_r y_0^r \leq S_N(y_0) \leq \sum_{r=1}^{\infty} \frac{1}{r} (N+1)^r y_0^r.$$

Proof. We rewrite the recursion to

$$\frac{y_{k+1}}{y_k} = 1 + y_k,$$

from which we obtain the form

$$y_{k+1} = y_0 \prod_{i=0}^k (1 + y_i).$$

The first two values are given by

$$\begin{aligned} y_1 &= y_0 + y_0^2, \\ y_2 &= (y_0 + y_0^2) \cdot (1 + y_0 + y_0^2) = y_0 + 2y_0^2 + 2y_0^3 + y_0^4. \end{aligned}$$

In general, we see that

$$y_k = \sum_{i=1}^{2^k} a_{k,i} y_0^i,$$

for positive integer coefficients $a_{k,i}$, with $a_{k,1} = a_{k,2^k} = 1$. From $y_{k+1} = y_k(1 + y_k)$, we see that the coefficients

satisfy the recursion

$$a_{k+1,i} = a_{k,i} + \sum_{j=1}^{i-1} a_{k,j} a_{k,i-j}.$$

For $i = 2$, we see that

$$a_{k+1,2} = a_{k,2} + a_{k,1}^2 = a_{k,2} + 1,$$

which results in $a_{k,2} = k$. Similarly,

$$a_{k+1,3} = a_{k,3} + 2a_{k,1}a_{k,2} = a_{k,3} + 2k,$$

which results in $a_{k,3} = k(k-1)$. We will prove by induction that $a_{k,i} < k^{i-1}$ for $i \geq 3$. This holds for $i = 3$ and $k \geq 2$ since we have $a_{k,3} = k(k-1) < k^2$. Note that this trivially holds for $i > 2^k$, since $a_{k,i} = 0 < k^{i-1}$. We show that if the induction hypothesis holds for $i \geq 3$, it will also hold for $i+1$:

$$a_{k+1,i+1} = a_{k,i+1} + \sum_{j=1}^i a_{k,j} a_{k,i+1-j} \leq a_{k,i+1} + \sum_{j=1}^i k^{j-1} k^{i-j} = a_{k,i+1} + i \cdot k^{i-1}.$$

Repeating this inequality $k-1$ times and applying an integral bound

$$\begin{aligned} a_{k+1,i+1} &\leq a_{1,i+1} + i \cdot \sum_{j=1}^k j^{i-1} \leq 0 + \int_1^{k+1} i z^{i-1} dz \\ &= [z^i]_1^{k+1} = (k+1)^i - 1. \end{aligned}$$

Hence, for $ky_0 < 1$,

$$y_k \leq y_0 \cdot \sum_{j=0}^{\infty} (ky_0)^j = y_0 \frac{1}{1 - ky_0}.$$

For $(N+1)y_0 < 1$, this leads to the upper bound

$$\begin{aligned} S_N(y_0) &\leq \sum_{k=0}^N y_0 \frac{1}{1 - ky_0} < \int_0^{(N+1)} \frac{y_0 dz}{1 - zy_0} \\ &= -[\log(1 - zy_0)]_0^{(N+1)y_0} = -\log(1 - (N+1)y_0) \\ &= \sum_{r=1}^{\infty} \frac{1}{r} (N+1)^r y_0^r. \end{aligned}$$

We will similarly prove by induction that $a_{k,i} \geq (k)_{i-1}$. It holds (with equality) up to $i = 3$. Substituting this into the recursion, we obtain

$$a_{k'+1,i+1} - a_{k',i+1} = \sum_{j=1}^i a_{k',j} a_{k',i+1-j} \geq \sum_{j=1}^i (k')_{j-1} (k')_{i-j} \geq i \cdot (k')_{i-1}.$$

Summing this inequality from $k' = i-1$ to $k' = k-1$, we obtain

$$\begin{aligned} a_{k,i+1} &= a_{k,i+1} - a_{1,i+1} \geq i \sum_{k'=1}^{k-1} (k')_{i-1} \\ &= i \sum_{k'=i-1}^{k-1} (k')_{i-1} = i! \sum_{k'=i-1}^{k-1} \binom{k'}{i-1} \\ &= i! \binom{k}{i} = (k)_i. \end{aligned}$$

where the last step follows from the hockey-stick identity of binomial coefficients. Hence,

$$S_N(y) = \sum_{i=1}^{2^N} \sum_{k=0}^N a_{k,i} y^i \geq \sum_{i=1}^{2^N} y^i \sum_{k=0}^N \binom{k}{i-1} = \sum_{i=1}^{2^N} \frac{(N+1)_i}{i} y^i = \sum_{i=1}^{\infty} \frac{(N+1)_i}{i} y^i.$$

□

The following theorem presents the optimal step sizes w.r.t. the bound proven in Proposition 2:

Theorem 3. *Let y_0 be the solution of*

$$\sigma^2 = y_0 S'_N(y_0) - S_N(y_0), \quad (30)$$

we have $u_N^{\mathbb{L}}(\sigma) = \sqrt{S'_N(y_0)}$. Let y_k follow the recursion in Equation (29) initialized at y_0 . The step sizes

$$\alpha_k^{\mathbb{L}} = \frac{R(N-k)}{L(N+1)^{3/2}} \frac{y_k}{u_N^{\mathbb{L}}(\sigma)y_0}, \quad (31)$$

enjoy the performance

$$f_{N+1} - f_{\star} \leq \frac{RL}{\sqrt{N+1}} u_N^{\mathbb{L}}(\sigma). \quad (32)$$

Proof. We want to minimize (11). Since y_k for $k > 0$ is fully determined by y_0 , we simply need to find the y_0 that minimizes

$$\frac{\sigma^2 + \sum_{k=0}^N y_k}{(N+1)y_0} = \frac{\sigma^2 + S_N(y_0)}{(N+1)y_0}.$$

We take the derivative w.r.t. y_0 and obtain

$$\frac{S'_N(y_0)}{(N+1)y_0} - \frac{\sigma^2 + S_N(y_0)}{(N+1)y_0^2} = 0,$$

which can be rewritten to (30). The corresponding performance bound becomes

$$\frac{RL}{\sqrt{N+1}} \sqrt{\frac{\sigma^2 + S_N(y_0)}{(N+1)y_0}}.$$

Substituting $S'_N(y_0) = \frac{1}{y_0}(\sigma^2 + S_N(y_0))$, the above can be rewritten to (32). Similarly substituting $u_N^{\mathbb{L}}(\sigma)$ into the step sizes from Proposition 2 yields (31), which completes the proof. □

Proof of Proposition 1. We remark that the Grammian must satisfy $\sum_{i=0}^{N-1} G(e_i, e_i) := \sum_{i=0}^{N-1} \langle e_i, e_i \rangle \leq \gamma^2$ as the power of the adversary is bounded.

Suppose now that we find a Grammian that additionally satisfies the conditions

$$G(g_j, g_i) + G(e_j, g_i) = \langle g_j + e_j, g_i \rangle = 0 \quad (i, j) \in [0, \dots, N+1] \times [0, \dots, N-1] : j < i \quad (33)$$

$$G(g_j, g_i) + G(e_j, g_i) = \langle g_j + e_j, g_i \rangle \geq 0 \quad (i, j) \in [0, \dots, N+1] \times [0, \dots, N-1] : j \geq i \quad (34)$$

and

$$G(x_0 - x_{\star}, g_i) = \langle x_0 - x_{\star}, g_i \rangle = \Delta \quad \forall i \in [0, \dots, N+1] \quad (35)$$

for some constant $\Delta \geq 0$. We claim that this implies that the suboptimality gap of any method satisfying Equation (12) is at least Δ . Additionally, we set $f^{\star} = 0$ and $f_k = f(x_k) = \Delta$ for all $k \in [0, \dots, N+1]$. We need to verify the fact that the adversarially chosen initial condition $x_0 - x_{\star}$, subgradients $g_0, \dots, g_{N+1}$ errors $e_0, \dots, e_{N-1}$

and associated function values $f_0, \dots, f_{N+1}$ are indeed compatible with the considered function class $\mathcal{F}$, i.e., the condition

$$f \in \mathcal{F}, \quad f_k = f(x_k) \quad \text{and} \quad g_k \in \partial f(x_k) \quad \forall k \in [\star, 0, \dots, N+1] \quad (36)$$

holds. A well known result (Boyd and Vandenberghe 2004, Drori and Teboulle 2016) is that this infinite dimensional interpolation condition can be reduced to a finite system of subgradient inequalities

$$(36) \iff \begin{cases} g_\star = 0, \quad \langle x_0 - x_\star, x_0 - x_\star \rangle \leq R^2, \\ \langle g_k, g_k \rangle \leq L^2 & \forall k \in [0, \dots, N+1], \\ f_j \geq f_i + \langle g_i, x_j - x_i \rangle & \forall i, j \in [\star, 0, \dots, N+1]. \end{cases}$$

It remains now to verify these conditions as the claim follows immediately from optimizing over all G satisfying the stated conditions.

First, we find that for all $i \in [0, \dots, N+1]$ we have

$$\begin{aligned} f_i - f_\star &\geq \langle x_i - x_\star, g_i \rangle \\ \iff \Delta &\geq \langle x_0 - x_\star, g_i \rangle - \langle \text{cone}(g_0 + e_0, \dots, g_{i-1 \wedge N-1} + e_{i-1 \wedge N-1}), g_i \rangle = \Delta. \end{aligned}$$

Here the first equivalence follows from condition (12) and $f_i = \Delta$ for all i and $f^\star = 0$. The last equality follows (33) and (35). Second, we verify convexity by

$$\begin{aligned} f_j &\geq f_i + \langle g_i, x_j - x_i \rangle \\ \iff 0 &\geq \langle g_i, x_j - x_i \rangle \\ \iff 0 &\geq -\langle g_i, \text{cone}(g_0 + e_0, \dots, g_{j-1 \wedge N-1} + e_{j-1 \wedge N-1}) \rangle + \langle g_i, \text{cone}(g_0 + e_0, \dots, g_{i-1 \wedge N-1} + e_{i-1 \wedge N-1}) \rangle \\ \iff 0 &\geq -\langle g_i, \text{cone}(\{g_k + e_k : \forall k \in [i, \dots, j-1 \wedge N-1]\}) \rangle \geq 0 \end{aligned}$$

for all $i \in [0, \dots, N+1]$ and j in $[0, \dots, N+1]$. The first equivalence follows from our choice $f_k = \Delta$ for all $k \in [0, \dots, N+1]$. The second equivalence follows from the condition (12). The third equivalence is a result of conditions (33) and (34). $\square$

Proof of Corollary 2. We consider candidate Grammian matrices $G = (A, B^\top; B, C)$ of the form

$$\left(\begin{array}{cccccc|cccccc} R^2 & \star & \star & \star & \dots & \star & \star & \star & \star & \star & \dots & \star \\ F\nu L^2 & L^2 & \star & \star & \dots & \star & \star & \star & \star & \star & \dots & \star \\ F\nu L^2 & \nu L^2 & L^2 & \star & \dots & \star & \star & \star & \star & \star & \dots & \star \\ F\nu L^2 & \nu L^2 & \nu L^2 & L^2 & \dots & \star & \star & \star & \star & \star & \dots & \star \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ F\nu L^2 & \nu L^2 & \nu L^2 & \nu L^2 & \dots & L^2 & L^2 & \star & \star & \star & \dots & \star \\ F\nu L^2 & \nu L^2 & \nu L^2 & \nu L^2 & \dots & L^2 & L^2 & \star & \star & \star & \dots & \star \\ \hline -F\gamma_0^2 & -\gamma_0^2 & -\nu L^2 & -\nu L^2 & \dots & -\nu L^2 & -\nu L^2 & \gamma_0^2 & \star & \star & \dots & \star \\ -F\gamma_1^2 & -\gamma_1^2 & -\gamma_1^2 & -\nu L^2 & \dots & -\nu L^2 & -\nu L^2 & \gamma_1^2 & \gamma_1^2 & \star & \dots & \star \\ -F\gamma_2^2 & -\gamma_2^2 & -\gamma_2^2 & -\gamma_2^2 & \dots & -\nu L^2 & -\nu L^2 & \gamma_2^2 & \gamma_2^2 & \gamma_2^2 & \dots & \star \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ -F\gamma_{N-1}^2 L^2 & -\gamma_{N-1}^2 & -\gamma_{N-1}^2 & -\gamma_{N-1}^2 & \dots & -\nu L^2 & -\nu L^2 & \gamma_{N-1}^2 & \gamma_{N-1}^2 & \gamma_{N-1}^2 & \dots & \gamma_{N-1}^2 \end{array} \right)$$

for $F \geq 0$ where here $[\star]$ indicate symmetric entries which are omitted for the sake of brevity.

We now check whether the matrix G is indeed feasible in Equation (13) for some $\Delta \geq 0$. Observe first that

$$G(g_j, g_i) + G(e_j, g_i) = \nu L^2 - \nu L^2 = 0 \quad \forall (i, j) \in [0, \dots, N+1] \times [0, \dots, N-1] : j < i$$

and

$$G(g_j, g_i) + G(e_j, g_i) = L^2 - \gamma_j^2 \geq 0 \quad \forall (i, j) \in [0, \dots, N+1] \times [0, \dots, N-1] : j \geq i.$$

Second, we clearly have

$$G(x_0 - x_\star, g_i) = F\nu L^2 =: \Delta \geq 0 \quad \forall i \in [0, \dots, N+1]$$

and $G(x_0 - x_\star, x_0 - x_\star) = R^2$, $G(g_i, g_i) = L^2$ for all $i \in [0, \dots, N]$ and $\sum_{i=0}^{N-1} G(e_i, e_i) = \sum_{i=0}^N \gamma_i^2 = \gamma^2$ from Equation (14).

It finally remains to verify that the candidate Grammian G is indeed positive semidefinite. From Schur's complement it suffices to verify that $C \succ 0$ and $A - B^\top C^{-1} B \succeq 0$. We establish that C is positive definite in Lemma 8. From Lemma 9 it follows immediately that $S = A - B^\top C^{-1} B$ is identically zero outside a 2×2 block in the top left corner where it takes on the values

$$\begin{pmatrix} S_{11} & \star \\ S_{2,1} & S_{2,2} \end{pmatrix} = \begin{pmatrix} R^2 - F^2 \gamma_0^2 & \star \\ F\nu L^2 - F\gamma_0 & L^2 - \gamma_0^2 \end{pmatrix}.$$

Hence, as we have here that $\gamma_0^2 < L^2$ it follows that for $S \succeq 0$ it suffices to have $S_{11} = S_{2,1}^2/S_{2,2}$. This leads to

$$\begin{aligned} &\iff \left(R^2 - \frac{F^2 N \nu^2 L^2}{1 + (N-1)\nu} \right) \left(L^2 - \frac{N \nu^2 L^2}{1 + (N-1)\nu} \right) = F^2 \left(\nu L^2 - \frac{N \nu^2 L^2}{1 + (N-1)\nu} \right)^2 \\ &\iff \left(R^2 - \frac{F^2 N \nu^2 L^2}{1 + (N-1)\nu} \right) \left(1 - \frac{N \nu^2}{1 + (N-1)\nu} \right) = F^2 L^2 \left(\nu - \frac{N \nu^2}{1 + (N-1)\nu} \right)^2 \\ &\iff R^2 \left(1 - \frac{N \nu^2}{1 + (N-1)\nu} \right) = F^2 L^2 \left(\left(\nu - \frac{N \nu^2}{1 + (N-1)\nu} \right)^2 + \frac{N \nu^2}{1 + (N-1)\nu} \left(1 - \frac{N \nu^2}{1 + (N-1)\nu} \right) \right) \\ &\iff R^2 L^2 \left(1 - \frac{N \nu^2}{1 + (N-1)\nu} \right) = F^2 L^4 \nu^2 \left(1 - \frac{2\nu N}{1 + (N-1)\nu} + \frac{N}{1 + (N-1)\nu} \right) \\ &\iff R^2 L^2 (1 + (N-1)\nu - N \nu^2) = F^2 L^4 \nu^2 (1 + (N-1)\nu - 2\nu N + N) \\ &\iff F^2 L^4 \nu^2 = \frac{R^2 L^2 (1 + (N-1)\nu - N \nu^2)}{N + 1 - \nu(N+1)} \\ &\iff \Delta = FL^2 \nu = RL \frac{\sqrt{1 + (N-1)\nu - N \nu^2}}{\sqrt{N+1 - \nu(N+1)}} = RL \frac{\sqrt{1 + N \nu}}{\sqrt{N+1}} \end{aligned}$$

from which the claim follows. □

Lemma 8. *Let $c_0 > c_1 > \dots > c_{N-1} > 0$ for any $N \geq 1$. Then*

$$\begin{pmatrix} c_0^2 & \star & \star & \dots & \star \\ c_1^2 & c_1^2 & \star & \dots & \star \\ c_2^2 & c_2^2 & c_2^2 & \dots & \star \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_{N-1}^2 & c_{N-1}^2 & c_{N-1}^2 & \dots & c_{N-1}^2 \end{pmatrix} \succ 0.$$

Proof. We will show this result by induction. Clearly the result holds for $N = 1$. Suppose now the results hold for

all $N = k$. Then, consider the $N = k + 1$ and partition the matrix of interest as follows

$$\begin{pmatrix} A & B \\ B^\top & C \end{pmatrix} = \left(\begin{array}{ccccc|c} c_0^2 & \star & \star & \dots & \star & \star \\ c_1^2 & c_1^2 & \star & \dots & \star & \star \\ c_2^2 & c_2^2 & c_2^2 & \dots & \star & \star \\ \vdots & \vdots & \vdots & \ddots & \vdots & \star \\ c_{k-1}^2 & c_{k-1}^2 & c_{k-1}^2 & \dots & c_{k-1}^2 & \star \\ \hline c_k^2 & c_k^2 & c_k^2 & \dots & c_k^2 & c_k^2 \end{array} \right).$$

From the induction hypothesis we know that $A \succ 0$. From Schur's complement it now suffices that

$$S = C - B^\top (A^{-1}B) = c_k^2 - B^\top \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \frac{c_k^2}{c_{k-1}^2} \end{pmatrix} = c_k^2 - c_k^4/c_{k-1}^2 > 0 \iff c_k^2 > c_{k-1}^2$$

to claim that indeed the result also holds for $N = k + 1$. The expression for $A^{-1}B$ above is obtained by solving $Ab = B$. $\square$

Lemma 9. *Consider B, C as given in Corollary 2. We have*

$$BC^{-1}B = \begin{pmatrix} F^2\gamma_0^2 & \star & \star & \star & \dots & \star & \star \\ F\gamma_0^2 & \gamma_0^2 & \star & \star & \dots & \star & \star \\ F\nu L^2 & \nu L^2 & L^2 & \star & \dots & \star & \star \\ F\nu L^2 & \nu L^2 & \nu L^2 & L^2 & \dots & \star & \star \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ F\nu L^2 & \nu L^2 & \nu L^2 & \nu L^2 & \dots & L^2 & L^2 \\ F\nu L^2 & \nu L^2 & \nu L^2 & \nu L^2 & \dots & L^2 & L^2 \end{pmatrix}.$$

Proof. We first show that

$$\begin{aligned} & \begin{pmatrix} \gamma_0^2 & \star & \star & \dots & \star \\ \gamma_1^2 & \gamma_1^2 & \star & \dots & \star \\ \gamma_2^2 & \gamma_2^2 & \gamma_2^2 & \dots & \star \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \gamma_{N-1}^2 & \gamma_{N-1}^2 & \gamma_{N-1}^2 & \dots & \gamma_{N-1}^2 \end{pmatrix} \begin{pmatrix} -F & -1 & -\frac{1}{\nu} - (N-1) & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{\nu} + (N-2) & -\frac{1}{\nu} - (N-2) & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{\nu} + (N-3) & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & -\frac{1}{\nu} & -\frac{1}{\nu} \end{pmatrix} \\ &= \begin{pmatrix} -F\gamma_0^2 & -\gamma_0^2 & -\nu L^2 & -\nu L^2 & \dots & -\nu L^2 & -\nu L^2 \\ -F\gamma_1^2 & -\gamma_1^2 & -\gamma_1^2 & -\nu L^2 & \dots & -\nu L^2 & -\nu L^2 \\ -F\gamma_2^2 & -\gamma_2^2 & -\gamma_2^2 & -\gamma_2^2 & \dots & -\nu L^2 & -\nu L^2 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ -F\gamma_{N-1}^2 & -\gamma_{N-1}^2 & -\gamma_{N-1}^2 & -\gamma_{N-1}^2 & \dots & -\nu L^2 & -\nu L^2 \end{pmatrix} \\ &= B =: [B_k]_{k \in [\star, 0, \dots, N+1]} =: [B_{i,k}]_{i \in [0, \dots, N-1], k \in [\star, 0, \dots, N+1]}. \end{aligned}$$

Clearly, we have that $B_N = B_{N+1} = -C\delta_N/\nu = \gamma_{N-1}^2/\nu 1_N = -\nu L^2 1_N$ where 1_N and δ_N denotes the vector of all ones and the N -th unit vector, respectively, using here that $\gamma_{N-1}^2 = \nu^2 L^2$. Similarly, we also have $B_\star = -CF\delta_1 =$

$-F[\gamma_0^2, \dots, \gamma_{N-1}^2]$ and $B_0 = -C\delta_1 = -[\gamma_0^2, \dots, \gamma_{N-1}^2]$. For any $k \in [1, \dots, N-1]$ and $i \geq k$ we observe that

$$B_{i,k} = \gamma_i^2 \left(-\frac{1}{\nu} - (N - (k-1)) \right) + \gamma_i^2 \left(\frac{1}{\nu} + (N - k) \right) = -\gamma_i^2$$

whereas for any $k \in [1, \dots, N-1]$ and $i < k$ we have from Lemma 10 that

$$B_{i,k} = \gamma_{k-1}^2 \left(-\frac{1}{\nu} - (N - k) \right) + \gamma_k^2 \left(\frac{1}{\nu} + (N - (k+1)) \right) = -\nu L^2.$$

Finally, via straightforward algebraic manipulation we have that

$$\begin{aligned} & B^\top (C^{-1}B) \\ &= \begin{pmatrix} -F\gamma_0^2 & -F\gamma_1^2 & -F\gamma_2^2 & \dots & -F\gamma_{N-1}^2 \\ -\gamma_0^2 & -\gamma_1^2 & -\gamma_2^2 & \dots & -\gamma_{N-1}^2 \\ -\nu L^2 & -\gamma_1^2 & -\gamma_2^2 & \dots & -\gamma_{N-1}^2 \\ -\nu L^2 & -\nu L^2 & -\gamma_2^2 & \dots & -\gamma_{N-1}^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -\nu L^2 & -\nu L^2 & -\nu L^2 & \dots & -\nu L^2 \\ -\nu L^2 & -\nu L^2 & -\nu L^2 & \dots & -\nu L^2 \end{pmatrix} \begin{pmatrix} -F & -1 & -\frac{1}{\nu} - (N-1) & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{\nu} + (N-2) & -\frac{1}{\nu} - (N-2) & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{\nu} + (N-3) & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & -\frac{1}{\nu} & -\frac{1}{\nu} \end{pmatrix} \\ &= \begin{pmatrix} F^2\gamma_0^2 & \star & \star & \star & \dots & \star & \star \\ F\gamma_0^2 & \gamma_0^2 & \star & \star & \dots & \star & \star \\ F\nu L^2 & \nu L^2 & L^2 & \star & \dots & \star & \star \\ F\nu L^2 & \nu L^2 & \nu L^2 & L^2 & \dots & \star & \star \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ F\nu L^2 & \nu L^2 & \nu L^2 & \nu L^2 & \dots & L^2 & L^2 \\ F\nu L^2 & \nu L^2 & \nu L^2 & \nu L^2 & \dots & L^2 & L^2 \end{pmatrix} \end{aligned}$$

with the help of Lemma 10 (to derive the entries resulting in νL^2) as well as Lemma 11 (to derive the entries equal to L^2). $\square$

Lemma 10. *We have*

$$\gamma_{k-1}^2 \left(-\frac{1}{\nu} - (N - k) \right) + \gamma_k^2 \left(\frac{1}{\nu} + (N - (k+1)) \right) = -\nu L^2$$

for all $k \in [1, \dots, N-1]$.

Proof. Observe

$$\begin{aligned} & \gamma_{k-1}^2 \left(-\frac{1}{\nu} - (N - k) \right) + \gamma_k^2 \left(\frac{1}{\nu} + (N - (k+1)) \right) \\ &= L^2 \frac{(N - k + 1)\nu^2}{1 + (N - k)\nu} \left(-\frac{1}{\nu} - (N - k) \right) + L^2 \frac{(N - k)\nu^2}{1 + (N - (k+1))\nu} \left(\frac{1}{\nu} + (N - (k+1)) \right) \\ &= \frac{1}{\nu} L^2 \left(\frac{(N - k + 1)\nu^2}{1 + (N - k)\nu} (-1 - (N - k)\nu) + \frac{(N - k)\nu^2}{1 + (N - (k+1))\nu} (1 + (N - (k+1))\nu) \right) \\ &= \frac{1}{\nu} L^2 \left(-(N - k + 1)\nu^2 + (N - k)\nu^2 \right) = -\nu L^2 \end{aligned}$$

for all $k \in [1, \dots, N-1]$. $\square$

Lemma 11. *We have*

$$-\nu L^2 \left(-\frac{1}{\nu} - (N - k) \right) - \gamma_k^2 \left(\frac{1}{\nu} + (N - (k + 1)) \right) = L^2$$

for all $k \in [0, \dots, N - 1]$.

Proof. Observe

$$\begin{aligned} & -\nu L^2 \left(-\frac{1}{\nu} - (N - k) \right) - \gamma_k^2 \left(\frac{1}{\nu} + (N - (k + 1)) \right) \\ &= L^2 (1 + (N - k)\nu) - L^2 \frac{(N - k)\nu^2}{1 + (N - (k + 1))\nu} \left(\frac{1}{\nu} + (N - (k + 1)) \right) \\ &= L^2 (1 + (N - k)\nu) - \frac{1}{\nu} L^2 \left(\frac{(N - k)\nu^2}{1 + (N - (k + 1))\nu} (1 + (N - (k + 1))\nu) \right) \\ &= L^2 (1 + (N - k)\nu) - \frac{1}{\nu} L^2 (N - k)\nu^2 = L^2 \end{aligned}$$

for all $k \in [0, \dots, N - 1]$. □

C Proofs for Section 3 (Analysis)

Proof of Lemma 2. Note that for any $y'_0 > 0$, the sequence $y'_k = y'_0 / (1 - ky'_0)$ satisfies $y'_{k+1} > y'_k + (y'_k)^2$, so that it satisfies the conditions of Proposition 2. We pick

$$y'_0 = \frac{1 - u(\sigma)^{-2}}{N + 1},$$

so that

$$y'_k = \frac{1 - u(\sigma)^{-2}}{N + 1 - k(1 - u(\sigma)^{-2})} = \frac{1}{\frac{u(\sigma)^2 + N}{u(\sigma)^2 - 1} + N - k}.$$

Hence,

$$\sum_{k=0}^N y'_k = \sum_{k=0}^N \frac{1}{\frac{u(\sigma)^2 + N}{u(\sigma)^2 - 1} + k} = H_{N+1} \left(\frac{u(\sigma)^2 + N}{u(\sigma)^2 - 1} \right) = H_{N+1} \left(1 + \frac{N + 1}{u(\sigma)^2 - 1} \right),$$

for the generalized harmonic number $H_m = \sum_{k=0}^{m-1} (a + k)^{-1}$. This leads to the factor

$$u'_N = \sqrt{\frac{\sigma^2 + H_{N+1}(1 + \frac{N+1}{u(\sigma)^2 - 1})}{(N + 1)y_0}} = \sqrt{\frac{\sigma^2 + H_{N+1}(1 + \frac{N+1}{u(\sigma)^2 - 1})}{1 - u(\sigma)^{-2}}} = u(\sigma) \sqrt{\frac{\sigma^2 + H_{N+1}(1 + \frac{N+1}{u(\sigma)^2 - 1})}{u(\sigma)^2 - 1}}.$$

By an integral bound, the generalized harmonic numbers are bounded by

$$H_m(a) \leq \log \left(1 + \frac{m}{a - 1} \right).$$

This leads to

$$\sqrt{\frac{\sigma^2 + H_{N+1}(1 + \frac{N+1}{u(\sigma)^2 - 1})}{u(\sigma)^2 - 1}} \leq \sqrt{\frac{\sigma^2 + \log(1 + u(\sigma)^2 - 1)}{u(\sigma)^2 - 1}}.$$

By the definition of $u(\sigma)$, we have $u(\sigma)^2 - 1 = \sigma^2 + 2 \log u(\sigma)$, so that the right-hand-side equals 1. This results in $u'_N \leq u(\sigma)$. This holds for the sequence

$$\alpha'_k = \frac{R}{L} \frac{N - k}{(N + 1)^{3/2}} \frac{y'_k}{y'_0 u'_N}. \quad (37)$$

We compute

$$\frac{y'_k}{y'_0} = \frac{1}{1 - ky'_0} = \frac{N+1}{N+1 - k(1 - u(\sigma)^{-2})} = \frac{(N+1)u(\sigma)^2}{(N+1)u(\sigma)^2 - (u(\sigma)^2 - 1)k}.$$

This results in

$$\frac{y'_k}{y'_0 u'_N} = \frac{(N+1)u(\sigma)}{(N+1)u(\sigma)^2 - (u(\sigma)^2 - 1)k} \sqrt{\frac{\sigma^2 + 2 \log u(\sigma)}{\sigma^2 + H_{N+1} \left(1 + \frac{N+1}{u(\sigma)^2 - 1}\right)}}.$$

The expression for the step sizes is obtained by substituting this quantity into (37).

We now show that the quantity in the square root converges to 1. Let

$$q = \frac{\sigma^2 + H_{N+1} \left(1 + \frac{N+1}{u(\sigma)^2 - 1}\right)}{\sigma^2 + 2 \log u(\sigma)} = 1 + \frac{H_{N+1} \left(1 + \frac{N+1}{u(\sigma)^2 - 1}\right) - 2 \log u(\sigma)}{\sigma^2 + 2 \log u(\sigma)},$$

so that the quantity in the square root is $q^{-1/2}$. Using the same integral bound as before, it follows that $q \leq 1$. To lower-bound q , we use the other integral bound to obtain

$$\begin{aligned} H_{N+1} \left(\frac{N+u^2}{u^2-1} \right) &\geq \log \left(1 + \frac{N+1}{\frac{N+u^2}{u^2-1}} \right) = \log \left(1 + (u^2-1) \frac{N+1}{N+u^2} \right) \\ &= \log \left(u^2 - \frac{(u^2-1)^2}{N+u^2} \right) = 2 \log u + \log \left(1 - \frac{(u^2-1)^2}{u^2(N+u^2)} \right) \\ &\geq 2 \log u + \log \left(\frac{N+1}{N+u^2} \right) = 2 \log u - \log \left(1 + \frac{u^2-1}{N+1} \right) \\ &\geq 2 \log u - \frac{u^2-1}{N+1}. \end{aligned}$$

This leads to the following bound

$$q - 1 \geq -\frac{\frac{u^2-1}{N+1}}{\sigma^2 + 2 \log u} = -\frac{1}{N+1},$$

where we used $\sigma^2 + 2 \log u = u^2 - 1$ in the last step. We conclude that

$$1 \leq q^{-1/2} \leq \sqrt{\frac{N+1}{N}} \leq 1 + \frac{2}{N}.$$

□

Proof of Lemma 3. To prove convexity, we take the first two derivatives of (16) w.r.t. σ and obtain

$$2\sigma = u' \cdot (2u - 2/u) \Rightarrow u' = \frac{u\sigma}{u^2 - 1},$$

and

$$2 = u'' \cdot (2u - 2/u) + 2(u')^2(1 - u^{-2}) \Rightarrow u'' = \frac{u}{(u^2 - 1)^2} (u^2 - 1 - \sigma^2),$$

which is nonnegative since $u^2 - 1 - \sigma^2 = 2 \log u \geq 0$. The lower bound $\sqrt{2}\sigma$ is derived using

$$u^2 - 1 - \sigma^2 = \log(1 + u^2(\sigma) - 1) \geq u(\sigma)^2 - 1 - \frac{1}{2}(u(\sigma)^2 - 1)^2,$$

which results in $u^2 - 1 \geq \sqrt{2}\sigma^2$. To derive the other lower bound, we write

$$u^2 = 1 + \sigma^2 + \log(u^2). \tag{38}$$

From $u^2 \geq 1$, we deduce

$$u^2 \geq 1 + \sigma^2.$$

Substituting this bound into (38) yields

$$u^2 \geq 1 + \sigma^2 + \log(1 + \sigma^2).$$

For the upper bound, we write

$$\begin{aligned} u(\sigma)^2 - 1 &= \int_0^\sigma 2u(s) \cdot u'(s) ds = \int_0^\sigma 2s \frac{u(s)^2}{u(s)^2 - 1} ds = \int_0^\sigma 2s \left(1 + \frac{1}{u(s)^2 - 1} \right) ds \\ &\leq \int_0^\sigma (2s + \sqrt{2}) ds = \sigma^2 + \sqrt{2}\sigma, \end{aligned}$$

where we used the bound $u(\sigma)^2 - 1 \geq \sqrt{2}\sigma$.

We now derive the asymptotics. Note that $u(\sigma) \rightarrow \infty$ as $\sigma \rightarrow \infty$, so that the right-hand-side of (16) is dominated by u^2 , which leads to the $u(\sigma) \sim \sigma$ asymptotics. For $\sigma \rightarrow 0$, the bounds yield the desired asymptotics, after using

$$u(\sigma) = \sqrt{1 + \sqrt{2}\sigma + \mathcal{O}(\sigma^2)} = 1 + \sqrt{1/2}\sigma + \mathcal{O}(\sigma^2).$$

□

Proof of Theorem 2. The inequalities $\ell_N(\sigma) \leq u_N^S(\sigma) \leq u_N^L(\sigma) \leq u(\sigma)$ follow from the fact that ℓ_N lower bounds the performance of a class of methods that include the minimizer of u_N^S , while u_N^L and u are obtained by adding additional constraints to the minimization problem. It therefore suffices to prove that

$$\frac{\ell_N(\sigma)}{u(\sigma)} \geq \left(1 - \frac{\log N}{N} - \frac{1 + 2\sqrt{N}}{N^2} \right).$$

We can rewrite the ν -constraint to

$$\begin{aligned} \sigma^2 &= \nu \sum_{i=0}^{N-1} \frac{(N-i)\nu}{1-\nu+(N-i)\nu} = N\nu - \nu \sum_{i=0}^{N-1} \frac{1-\nu}{1-\nu+(N-i)\nu} \\ &= N\nu - (1-\nu) \sum_{k=1}^N \frac{1}{\frac{1}{\nu}-1+k} = N\nu - (1-\nu)H_N(1/\nu), \end{aligned} \tag{39}$$

where $H_m(a)$ the generalized harmonic number. We study the asymptotics of $H_m(a)$. Integral bounds result in

$$\log \left(1 + \frac{m}{a} \right) \leq H_m(a) \leq \log \left(1 + \frac{m}{a-1} \right).$$

This leads to

$$(1-\nu) \log(1 + N\nu) \leq N\nu - \sigma^2 \leq (1-\nu) \log \left(1 + \frac{N\nu}{1-\nu} \right),$$

For the right-hand-side, we write $\frac{1}{1-\nu} = 1 + \frac{\nu}{1-\nu}$ and use concavity to bound

$$\log \left(1 + \frac{N\nu}{1-\nu} \right) \leq \log(1 + N\nu) + \frac{\frac{N\nu^2}{1-\nu}}{1 + N\nu},$$

so that

$$N\nu - \log(1 + N\nu) - \sigma^2 \in \left[-\nu \log(1 + N\nu), \frac{N\nu^2}{1 + N\nu} - \nu \log(1 + N\nu) \right].$$

Substituting $\ell_N(\sigma) = \sqrt{1 + N\nu}$, or $\nu = \frac{1}{N}(\ell_N^2 - 1)$ yields

$$\ell_N^2 - 1 - 2 \log \ell_N - \sigma^2 \in \left[-2 \frac{\ell_N^2 - 1}{N} \log \ell_N, \frac{\ell_N^2 - 1}{N} \left(\frac{\ell_N^2 - 1}{\ell_N^2} - 2 \log \ell_N \right) \right].$$

In this, we recognize the definition of $u(\sigma)$ given in (16). This leads to the bounds

$$u \left(\sigma \sqrt{1 - 2 \frac{\ell_N^2 - 1}{N \sigma^2} \log \ell_N} \right) \leq \ell_N \leq u \left(\sigma \sqrt{1 - \frac{\ell_N^2 - 1}{N \sigma^2} \left(2 \log \ell_N - \frac{\ell_N^2 - 1}{\ell_N^2} \right)} \right). \quad (40)$$

Applying $\log(1+x) \leq \frac{x}{x+1}$ to $2 \log \ell_N = \log(1 + \ell_N^2 - 1)$ already tells us that the right-hand-side is at most $u(\sigma)$. Applying $\ell_N(\sigma) \leq u(\sigma)$ to the left-hand-side leads to

$$\ell_N(\sigma) \geq u \left(\sigma \sqrt{1 - 2 \frac{u(\sigma)^2 - 1}{N \sigma^2} \log u(\sigma)} \right).$$

Using the bound $\sqrt{1-x} \geq 1-x$, we obtain

$$\ell_N(\sigma) \geq u \left(\sigma - 2 \frac{u(\sigma)^2 - 1}{N \sigma} \log u(\sigma) \right).$$

Next, we use convexity of $u(\sigma)$ to write

$$u \left(\sigma - 2 \frac{u(\sigma)^2 - 1}{N \sigma} \log u(\sigma) \right) \geq u(\sigma) - 2 \frac{u(\sigma)^2 - 1}{N \sigma} \log u(\sigma) \cdot u'(\sigma).$$

Next, we substitute

$$u'(\sigma) = \frac{u(\sigma) \cdot \sigma}{u(\sigma)^2 - 1}$$

to obtain

$$\ell_N(\sigma) \geq u(\sigma) - \frac{2u(\sigma) \log u(\sigma)}{N} = u(\sigma) \cdot \left(1 - \frac{\log(u(\sigma)^2)}{N} \right).$$

Finally, the bound $u(\sigma)^2 \leq 1 + 2\sigma + \sigma^2$ from Lemma 3 and $\sigma \leq \sqrt{N}$ yield $\log u(\sigma)^2 \leq \log(1 + 2\sqrt{N} + N) \leq \log(N+1) + \frac{2\sqrt{N}}{N+1} \leq (1 + (\log 2)^{-1}) \log(N+1)$ for $N \geq 1$, where the last step follows from the fact that $2\sqrt{N}/((N+1) \log(N+1))$ is decreasing and equal to $(\log 2)^{-1}$ for $N = 1$. The desired bound follows from the fact that $1 + (\log 2)^{-1} < \frac{5}{2}$. $\square$

Proof of Lemma 4. We will use the bounds of Lemma 7 to write

$$\sum_{i=1}^{2^N} (k)_{i-1} y_0^i \leq y_k \leq \sum_{i=1}^{2^N} k^{i-1} y_0^i.$$

For $y_0 < k^{-1}$, we can further bound the right-hand-side to an infinite sum, which results in the known power series

$$y_k \leq \sum_{i=1}^{\infty} k^{i-1} y_0^i = \frac{y_0}{1 - k y_0}.$$

For the lower bound, we will first lower-bound the falling factorial using an integral bound and an expansion of the

logarithm:

$$\begin{aligned}
\log(n)_r &= \sum_{x=n-r+1}^n \log x \geq \int_{n-r}^n \log x dx \\
&= n \log n - (n-r) \log(n-r) - r = r \log n - (n-r) \log \left(1 - \frac{r}{n}\right) - r \\
&\geq r \log n - \frac{r^2}{n},
\end{aligned}$$

so that

$$(n)_r \geq n^r \cdot e^{-r^2/n} \geq n^r \cdot \left(1 - \frac{r^2}{n}\right).$$

Notice that $(k)_{i-1} = 0$ for $i > k+1$. Hence,

$$y_k \geq \sum_{i=1}^{2^N} (k)_{i-1} y_0^i = \sum_{i=1}^{\infty} (k)_{i-1} y_0^i \geq \sum_{i=1}^{\infty} k^{i-1} \cdot \left(1 - \frac{(i-1)^2}{k}\right) y_0^i = \frac{y_0}{1 - ky_0} - \frac{y_0}{k} \sum_{i=0}^{\infty} i^2 y_0^i,$$

for $y_0 < k^{-1}$. Using $\sum_{i=0}^{\infty} i^2 w^i = \frac{w(1+w)}{(1-w)^3}$ for $w = ky_0 < 1$, we obtain the lower bound

$$y_k \geq \frac{y_0}{1 - ky_0} \left(1 - \frac{y_0(1 + ky_0)}{(1 - ky_0)^2}\right),$$

which is asymptotically equivalent to the upper bound whenever

$$\frac{y_0}{(1 - ky_0)^2} \rightarrow 0. \quad (41)$$

To see for what σ this holds, we inspect the asymptotics of (30). We rewrite

$$\sigma^2 = y_0 S'_N(y_0) - S_N(y_0) = \sum_{i=1}^{2^N} (i-1) \cdot s_{N,i} \cdot y_0^i,$$

where $s_{N,i} = \sum_{k=0}^N a_{k,i}$ is the i -th coefficient of $S_N(y_0)$. Lemma 7 proves the bounds

$$\frac{1}{i} (N+1)_i \leq s_{N,i} \leq \frac{1}{i} (N+1)^i.$$

This leads to the bounds

$$\sigma^2 \leq \sum_{i=1}^{\infty} \left(1 - \frac{1}{i}\right) \cdot (N+1)^i \cdot y_0^i = \frac{(N+1)y_0}{1 - (N+1)y_0} + \log(1 - (N+1)y_0),$$

and

$$\begin{aligned}
\sigma^2 &\geq \sum_{i=1}^{\infty} \left(1 - \frac{1}{i}\right) \left(1 - \frac{i^2}{N+1}\right) \cdot (N+1)^i \cdot y_0^i \\
&= \frac{(N+1)y_0}{1 - (N+1)y_0} + \log(1 - (N+1)y_0) - (N+1)y_0^2 \sum_{i=1}^{\infty} i(i-1)(N+1)^{i-2} y_0^{i-2},
\end{aligned}$$

where we were again able to extend the sum to infinity because $(N+1)_i = 0$ for $i > N+1$. Using

$$\sum_{i=2}^{\infty} i(i-1)w^{i-2} = \frac{d^2}{dw^2} \sum_{i=0}^{\infty} w^i = \frac{d^2}{dw^2} \frac{1}{1-w} = \frac{2}{(1-w)^3},$$

we obtain

$$\sigma^2 = \frac{(N+1)y_0}{1-(N+1)y_0} + \log(1-(N+1)y_0) + \mathcal{O}\left(\frac{(N+1)y_0^2}{(1-(N+1)y_0)^3}\right).$$

Introduce $z_0 = \frac{(N+1)y_0}{1-(N+1)y_0}$, then

$$\sigma^2 = z_0 - \log(1+z_0) + \mathcal{O}\left(\frac{z_0^2(1+z_0)}{N+1}\right).$$

We look for regimes where the error term is negligible, so that $z_0 \sim z(\sigma)$, where $z(\sigma)$ is the positive solution in

$$\sigma^2 = z - \log(1+z). \quad (42)$$

Notice that $z_0 - \log(1+z_0) \sim z_0$ for large z_0 and $z_0 - \log(1+z_0) \sim \frac{1}{2}z_0^2$ for small z_0 . Hence, for $\sigma \rightarrow \infty$, we need $z_0 \ll \sqrt{N}$ to get $z_0 \sim \sigma^2$. This occurs whenever $\sigma \ll N^{1/4}$. For $z_0 \rightarrow 0$, we need $\frac{z_0^2(1+z_0)}{N+1} \ll z_0^2$ which always holds since $N \rightarrow \infty$. This tells us that whenever $\sigma \ll N^{1/4}$, we have $z_0 \sim z(\sigma)$, and

$$y_0 = \frac{1}{N+1} \frac{z_0}{z_0+1} \sim \frac{1}{N+1} \frac{z(\sigma)}{z(\sigma)+1}.$$

Returning to the condition (41), we note that

$$\frac{y_0}{(1-ky_0)^2} < \frac{y_0}{(1-(N+1)y_0)^2} = \frac{z_0(1+z_0)}{N+1},$$

which indeed vanishes since $z_n \ll \sqrt{N}$. This tells us that $y_k \sim \frac{y_0}{1-ky_0}$ for $\sigma \ll N^{1/4}$. Finally, from Theorem 2, it follows that $u_N^{\mathbb{L}}(\sigma) \sim u(\sigma)$, so that

$$\alpha_k^{\mathbb{L}} \sim \alpha_k^* \cdot \frac{1}{u(\sigma) \cdot (1-ky_0)},$$

where α_k^* are the optimal step sizes for the noiseless case given in (9). The result follows after substituting

$$y_0 \sim \frac{1}{N+1} \frac{u(\sigma)^2 - 1}{u(\sigma)^2}.$$

□

References for the Appendix

Stephen P Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

Yoel Drori and Marc Teboulle. An optimal variant of Kelley's cutting-plane method. *Mathematical Programming*, 160: 321–351, 2016.