Complexity Guarantees for Polyak Steps with Momentum

Mathieu BARRÉ MATHIEU.BARRE@INRIA.FR
INRIA, Département d'informatique de l'ENS, Ecole normale supérieure, CNRS, PSL Research University, Paris, France

Adrien TAYLOR ADRIEN.TAYLOR@INRIA.FR
INRIA, Département d'informatique de l'ENS, Ecole normale supérieure, CNRS, PSL Research University, Paris, France

Alexandre d'ASPREMONT ASPREMON@ENS.FR
Département d'informatique de l'ENS, Ecole normale supérieure, CNRS, PSL Research University, Paris, France.

Editors: Jacob Abernethy and Shivani Agarwal

Abstract

In smooth strongly convex optimization, knowledge of the strong convexity parameter is critical for obtaining simple methods with accelerated rates. In this work, we study a class of methods, based on Polyak steps, where this knowledge is substituted by that of the optimal value, f_*. We first show slightly improved convergence bounds than previously known for the classical case of simple gradient descent with Polyak steps, we then derive an accelerated gradient method with Polyak steps and momentum, along with convergence guarantees.

1. Introduction

We focus on unconstrained optimization problems of the form

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

where f is strongly convex and has a Lipschitz continuous gradient with respect to the Euclidean norm. Very broadly speaking, the current numerical toolbox to solve these convex minimization problems contains two types of methods. On one hand, simple numerical schemes with explicit albeit conservative theoretical guarantees. These include gradient methods and their accelerated variants, and require knowing problem *parameters*, such as strong convexity parameters, or Hölderian error bounds (Bolte et al., 2007). On the other hand, *adaptive methods*, such as conjugate gradients or quasi-Newton, adapting much better to the objective function by estimating some of its regularity properties. For these methods, we typically have no theoretical justification for their improved performances or no computational complexity bounds at all.

Empirically, adaptive methods often perform significantly better than their parametric counterparts, and, by nature, require much less tuning. For example, roughly estimating regularity constants on-the-fly and plugging these estimates in parametric algorithms often produces fast algorithms with no theoretical guarantees. This phenomenon is illustrated in Figure 1 on logistic regression.

Although many advances have been made in designing optimization schemes adaptive to some types of parameters (e.g., Lipschitz constants, see discussions below), these results still leave a huge

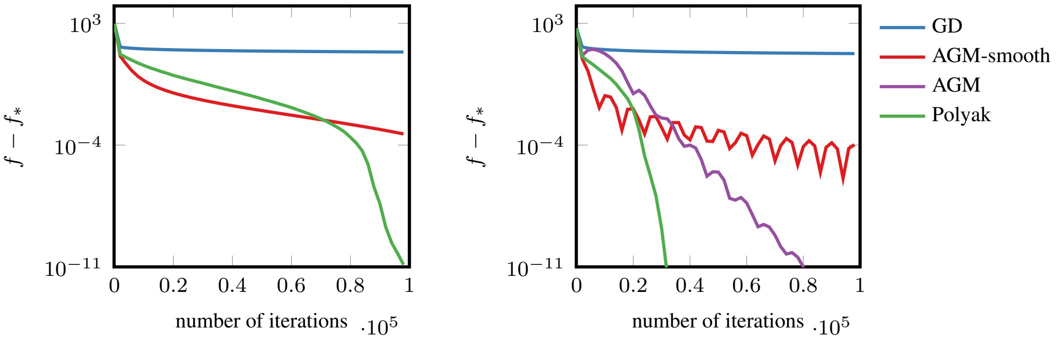

Figure 1: Convergence of gradient descent (GD), accelerated gradient method for smooth optimization (AGM-smooth) (Nesterov, 1983), accelerated gradient method with constant momentum (AGM)—described below as Algorithm 2 with (Const-mom)—where the momentum is set using the value of the regularization parameter and gradient method with Polyak steps (Polyak). Experiments on regularized logistic regression for the Sonar dataset without any tuning of the methods. Left: regularization parameter 10^{-7}. Right: regularization parameter 10^{-4}. For Polyak steps, the best iterate is displayed. Observe that Polyak method is a (non-accelerated) adaptive method, which performs comparatively well against accelerated schemes.

gap between theory and practice (as in Figure 1). In particular, estimating strong convexity coefficients while preserving convergence guarantees remains a challenging issue. Restart schemes are probably the most effective option among existing approaches for adapting to this type of parameters and do provide improved complexity estimates without any knowledge of strong convexity parameters, at the expense of a log scale grid search. However, while on paper the complexity of these schemes is nearly optimal, the presence of an outer loop clearly limits their practical effectiveness and their capacity to adapt to the function's local regularity, which leaves a lot of margin for improvement, on the numerical front. Producing single loop algorithms adapting to local strong convexity (or Hölderian error bounds) and have nearly optimal complexity bounds is an important open problem which is the main focus of this work.

Here, we study the complexity of adaptive methods using Polyak steps, estimating the momentum term using information on the optimum objective value f_* instead of the strong convexity constant. In some scenarios, such as "interpolation" in machine learning problems, the value of f_* is known a priori (usually zero), and estimating it is much easier than estimating strong convexity, see e.g., (Asi and Duchi, 2019) for a recent discussion on these model assumptions.

The obvious next research question in this direction is to substitute knowledge on f_* by weaker bounds. A first step in this direction is for example (Hazan and Kakade, 2019) which uses successive refinements of a lower bound on f_*. As it is, the proof in (Hazan and Kakade, 2019) contains several errors, but can be fixed. We hope, and believe, that such a mechanism could be used together with our momentum version of the Polyak steps.

1.1. Related works

Gradient and accelerated gradient methods. For smooth optimization problems, simple line search strategies provide accelerated algorithms that adapt to the local gradient Lipschitz constant (Nesterov, 2013) and explicit adaptive complexity bounds can be derived for certain variants using the mean root Lipschitz constant (Scheinberg et al., 2014).

Restarts. For smooth and strongly convex optimization problems (or more generally problems satisfying Hölderian error bounds), accelerated methods with optimal complexity bounds require knowledge of the strong convexity constant to compute iterates (Nesterov, 2013, 2018). In particular, Arjevani and Shamir (2016) show that this information is necessary when using oblivious steps. This quantity can be hard to estimate and a lot of effort has been put in the development of adaptive optimization methods preserving fast convergence rates (Lin and Xiao, 2014; Fercoq and Qu, 2016; Roulet and d'Aspremont, 2017). All these works are based on restart strategies (O'Donoghue and Candes, 2015; Nesterov, 2013) and although they exhibit fast theoretical convergence rates, they often contain parameters that have to be tuned in order to get good practical results, or require additional information on the function itself (e.g., its minimum f^*). Once again, while on paper the complexity of restart schemes is nearly optimal, the presence of an outer loop generally limits their capacity to adapt to the function's local regularity and significantly affects empirical performance.

Quasi-Newton methods. An important family of adaptive algorithms is composed with quasi-Newton methods. As the name suggests, these methods try to mimic the behavior of Newton schemes, by constructing an estimate of the hessian at the current point, using previous gradients. The most notable quasi-Newton method is certainly L-BFGS (Liu and Nocedal, 1989). These commonly used algorithms exhibit very fast empirical converge rates but only classical convergence rates comparable to that of gradient descent have been proven at this point (Byrd et al., 1987).

Conjugate gradient methods. Conjugate gradient methods are probably among the most famous examples of adaptive algorithm. Firstly introduced for quadratic minimization (Hestenes and Stiefel, 1952), and motivated by nice theoretical guarantees (such as finite-time convergence), many variants have been introduced for going beyond quadratics (Fletcher and Reeves, 1964; Polyak, 1969; Fletcher, 1987)—see, for example, the nice survey (Hager and Zhang, 2006). Roughly speaking, at each iteration, the method constructs an update direction based on the gradient at the current iterate, and on the knowledge of the previous search directions. The next iterate is obtained by line-search in the update direction. Whereas conjugate gradient methods are widely used in practice (e.g Rodi and Mackie (2001); Volkwein (2004); Zhao et al. (2015)), and perform very well when they applies, there are barely any non-asymptotic convergence guarantees for those methods beyond unconstrained quadratic minimization.

Polyak step-sizes. When the optimal value of the objective function value is known, a well-known adaptive strategy consists in using the so-called "Polyak step-sizes"—see e.g., (Polyak, 1987, Section 5.3.2) or (Nedic and Bertsekas, 2001; Boyd et al., 2003). The method consists in iterating gradient steps with step-sizes proportional to the primal gap at the current iterate. As opposed to most adaptive gradient methods mentioned above, this method comes with explicit theoretical properties, even beyond the quadratic optimization case.

Barzilai-Borwein step-sizes. The Barzilai-Borwein (Barzilai and Borwein, 1988; Fletcher, 2005) method consists in gradient steps with adaptive step-sizes. It is another case with complete theory for quadratic optimization, but barely any performance guarantees in non-quadratic cases (it is even known to diverge on some problem instances).

Adaptive gradient steps In (Malitsky and Mishchenko, 2019) the authors developed a step-size policy that adapts to the local geometry, together with nice theoretical guarantees.

1.2. Contributions

We develop and analyze an accelerated variant of the gradient method with Polyak steps that includes a momentum term and has better dependence on the condition number. We believe the Performance Estimation Program (PEP) technique used for obtaining the worst-case convergence guarantees is also of independent interest. As a byproduct, we also slightly improve convergence bounds for variants of the classical gradient method with Polyak steps (i.e. without momentum).

2. Classical Polyak Steps and Variants

We denote f_* the minimum of f. Let $0 \leq \mu < L$, the class of L-smooth and μ-strongly convex functions is denoted $\mathcal{F}_{\mu,L}$. Functions in this class satisfy (see e.g., (Nesterov, 2018)) $\forall x, y \in \mathbb{R}^n$:

$$\begin{aligned} f(y) &\leq f(x) + \langle \nabla f(x), y - x \rangle + \tfrac{L}{2}\|y - x\|^2 \quad \text{(smoothness)}, \\ f(y) &\geq f(x) + \langle \nabla f(x), y - x \rangle + \tfrac{\mu}{2}\|y - x\|^2 \quad \text{(strong convexity)}. \end{aligned}$$

Let us start with complexity bounds for gradient methods with Polyak steps for smooth and strongly convex optimization problems. Note that Polyak step sizes are usually discussed in the nondifferentiable setting—see (Polyak, 1987, Section 5.3.2) or (Nedic and Bertsekas, 2001; Boyd et al., 2003). We first recall the complexity of the gradient method with Polyak steps in the smooth strongly convex case, then derive similar bounds for two variants. For the first variant, we scale the steps by a factor two compared to standard Polyak steps, yielding a simple convergence proof with slightly improved theoretical guarantees. The second variant is a descent method, where the complexity bound is written in terms of the primal gap. We delay a full discussion of the proof mechanisms to Section 4, and the proofs themselves to the appendix.

Algorithm 1 Adaptive gradient method

```
Input: $x_0 \in \mathbb{R}^n$, $f_* \in \mathbb{R}$
for $k \geq 0$ do
    compute $\gamma_k$
    $x_{k+1} = x_k - \gamma_k \nabla f(x_k)$
end for
Output: $x_{k+1}$
```

Regular Polyak steps: $\gamma_k = \frac{f(x_k)-f_*}{\|\nabla f(x_k)\|^2}$ (Polyak)

Polyak steps, variant I: $\gamma_k = 2\frac{f(x_k)-f_*}{\|\nabla f(x_k)\|^2}$ (Variant I)

Polyak steps, variant II: $\gamma_k = \left(2 - \frac{\|\nabla f(x_k)\|^2}{2L(f(x_k)-f_*)}\right)/L$ (Variant II)

The classical step size rule (Polyak) was mostly studied in the nonsmooth convex case (Polyak, 1987). For smooth strongly convex problems, it is known (see e.g., (Hazan and Kakade, 2019)) that

$$f(x_N) - f_* \leq (1 - \tfrac{\mu}{L})^N \frac{L\|x_0 - x_*\|^2}{2}. \tag{1}$$

The two following propositions show that different step sizes policies (namely (Variant I) and (Variant II)) produce slightly improved convergence rates, matching the best known rates for gradient methods with known μ and L. The γ_k are always well defined except when x_k has a zero gradient, in this case we can simply stop the method since we have reached optimality. When it is well defined, $\gamma_k \in [\frac{1}{L}, \frac{1}{\mu}]$ for (Variant I) and $\gamma_k \in [\frac{1}{L}, \frac{2-\mu/L}{L}]$ for (Variant II). First, let us state that if we seek to decrease the distance to the optimal point, (Variant I) provides a rate that matches that of gradient descent with optimal (non-adaptive) step sizes (Nesterov, 2018).

Proposition 1 (Appendix A) *Let $f \in \mathcal{F}_{\mu,L}$ and consider Algorithm 1 with step-sizes* (Variant I)*. Then, for any $x_0 \in \mathbb{R}^n$ and $N \in \mathbb{N}$, such that the sequence $\{\gamma_k\}_k$ is well defined, it holds that*

$$\|x_N - x_*\|^2 \leq \left(\prod_{k=0}^{N-1} \rho(\gamma_k)\right) \|x_0 - x_*\|^2,$$

where $\rho(\gamma) = \frac{(\gamma L-1)(1-\gamma\mu)}{\gamma(L+\mu)-1}$, and $\max_{\gamma\in[\frac{1}{L},\frac{1}{\mu}]} \rho(\gamma) = \frac{(L-\mu)^2}{(L+\mu)^2}$. Otherwise $\nabla f(x_k) = 0$ with $k \in [0, N]$.

If on the other hand we seek to decrease the primal gap, (Variant II) provides a rate that matches that of gradient descent with exact line search (de Klerk et al., 2017), at the expense of knowledge on L.

Proposition 2 (Appendix B) *Let $f \in \mathcal{F}_{\mu,L}$ and consider Algorithm 1 with step-sizes* (Variant II)*. Then, for any $x_0 \in \mathbb{R}^n$ and $N \in \mathbb{N}$, such that the sequence $\{\gamma_k\}_k$ is well defined, it holds that*

$$f(x_N) - f_* \leq \left(\prod_{k=0}^{N-1} \rho(\gamma_k)\right) (f(x_0) - f_*),$$

where $\rho(\gamma) = (L\gamma - 1)(L\gamma(3 - \gamma(L+\mu)) - 1)$, and $\max_{\gamma\in[\frac{1}{L},\frac{2L-\mu}{L^2}]} \rho(\gamma) = \frac{(L-\mu)^2}{(L+\mu)^2}$. Otherwise $\nabla f(x_k) = 0$ with $k \in [0, N]$.

In the following section, we study variants of those methods, where we aim to speed up convergence by incorporating a momentum term. Those methods follow in spirit the line of works on Nesterov's acceleration (Nesterov, 2013), where we supersede knowledge of μ by that of f_*.

3. Acceleration with Polyak momentum

In the following, AGM refers to the Accelerated Gradient Method with momentum introduced by Nesterov (Nesterov, 1983, 2018). We are interested in optimizing a function $f \in \mathcal{F}_{\mu,L}$ without any information on the strong convexity constant μ. However, as in the Polyak gradient method, we rely on the knowledge of f^*. We describe a single loop adaptive accelerated method (i.e. without restarts), with convergence rate of order $1 - (\mu/L)^{3/4}$, compared with $1 - \mu/L$ for gradient descent, and $1 - (\mu/L)^{1/2}$ for its accelerated version with perfect knowledge of μ.

Algorithm 2 Accelerated gradient method (AGM)

```
**Input:** $x_0 \in \mathbb{R}^n$, $f_* \in \mathbb{R}$, $L$ smoothness constant.
$y_0 = x_0$,
**for** $k \geq 0$ **do**
    $y_{k+1} = x_k - \frac{1}{L}\nabla f(x_k)$
    compute $\tilde{\mu}_k$ and $\beta_k = \frac{\sqrt{L}-\sqrt{\tilde{\mu}_k}}{\sqrt{L}+\sqrt{\tilde{\mu}_k}}$
    $x_{k+1} = y_{k+1} + \beta_k(y_{k+1} - y_k)$
**end for**
**Output:** $y_{k+1}$
```

$$\text{Constant momentum:} \quad \tilde{\mu}_k = \mu \tag{Const-mom}$$

$$\text{Polyak Acc., variant I:} \quad \tilde{\mu}_k = \frac{\|\nabla f(y_{k+1})\|^2}{2(f(y_{k+1})-f_*)}, \tag{Acc. Variant I}$$

$$\text{Polyak Acc., variant II:} \quad \tilde{\mu}_k = \begin{cases} +\infty & \text{if } k=-1 \\ \min\left(\tilde{\mu}_{k-1}, \frac{\|\nabla f(y_{k+1})\|^2}{2(f(y_{k+1})-f_*)}\right) & \text{otherwise} \end{cases} \tag{Acc. Variant II}$$

Algorithm 2 is based on the AGM algorithm (Nesterov, 2018), in which the knowledge of μ is essential to set the constant momentum term $\beta_k = \beta_* = (\sqrt{L}-\sqrt{\mu})/(\sqrt{L}+\sqrt{\mu})$. Common convergence guarantees require a lower bound on the strong convexity. As a first step towards producing adaptive versions of AGM, Lemma 3 and Corollary 4 below guarantee that AGM with any momentum factor β_k in $[0,1]$ converges at least as fast as the classical gradient method.

Lemma 3 (Convergence of AGM with bad momentum, Appendix C.1) *Let $f \in \mathcal{F}_{\mu,L}$, some iteration number $k \in \mathbb{N}$, and consider Algorithm 2 with $\beta_k \in [0,1]$. Then, for any $x_k, y_k \in \mathbb{R}^n$, it holds that*

$$V(x_{k+1}, y_{k+1}) \leq \rho V(x_k, y_k) \tag{2}$$

where $V(x,y) = \frac{L-\mu}{2}\|x-y\|^2 + f(y) - f_*$ and $\rho = 1 - \frac{\mu}{L}$.

We then get the following corollary on the primal gap.

Corollary 4 *Let $f \in \mathcal{F}_{\mu,L}$, a number of iterations $N \in \mathbb{N}$, and consider Algorithm 2 with a sequence $\{\beta_k\}_k$ satisfying $\beta_k \in [0,1]$ for all $k \in [1,N]$. Then, for any $x_0 \in \mathbb{R}^n$, it holds that*

$$f(y_N) - f_* \leq \left(1 - \frac{\mu}{L}\right)^N (f(x_0) - f_*).$$

Proof. Direct from Lemma 3 with $x_0 = y_0$. $\blacksquare$

This result shows the robustness of AGM with respect to the momentum parameter. Adaptive strategies, that modify the momentum term in the algorithm automatically, thus at least enjoy the gradient method's convergence rate when β_k is kept within the interval $[0,1]$—this is the case for both (Acc. Variant I) and (Acc. Variant II). To our knowledge, only non-blowup properties (Lin and Xiao, 2014, Lemma 1) were known when overestimating μ.

The momentum term in (Acc. Variant I) was designed using the inverse of Polyak's step as an estimate of the strong convexity parameter. The motivation for this choice of strong convexity estimate is the fact that under some mild assumptions on f (i.e., for quadratic or self-concordant f), the quantity $\frac{\|\nabla f(z_k)\|^2}{2(f(z_k)-f_*)}$ converges to the strong convexity constant at optimum when the z_k are iterates of gradient descent algorithm with step-size 1/L.

In order for $\tilde{\mu}_k$ to be always defined and within the interval $[\mu, L]$, we assume that iterates never reach exactly optimality. Under this condition we have $\beta_k \in [0, \beta_*]$ and Corollary 4 readily applies to both (Acc. Variant I) or (Acc. Variant II). However, this result can be improved for those particular choices, as described in Lemma 5 and Proposition 6, as the rate can be expressed in terms of the local $\tilde{\mu}_k$ instead of μ.

Lemma 5 (Appendix C.2) *Let $f \in \mathcal{F}_{0,L}$, some iteration number $k \in \mathbb{N}$, and consider Algorithm 2 with either* (Acc. Variant I) *or* (Acc. Variant II). *For any $x_k, y_k \in \mathbb{R}^n$ such that $\tilde{\mu}_k$ well defined, it holds that*

$$V(x_{k+1}, y_{k+1}) \leq \rho(\tilde{\mu}_k) V(x_k, y_k) \tag{3}$$

where $V(x,y) = \frac{L}{2}\|x-y\|^2 + f(y) - f_*$ and $\rho(\tilde{\mu}) = \frac{1}{1+\frac{\tilde{\mu}}{L}}$. Otherwise $\nabla f(y_{k+1}) = 0$.

Proposition 6 *Let $f \in \mathcal{F}_{0,L}$, some number of iterations $N \in \mathbb{N}$, and consider Algorithm 2 with either* (Acc. Variant I) *or* (Acc. Variant II). *Then, for any $x_0 \in \mathbb{R}^n$, such that the sequence $\{\tilde{\mu}_k\}_k$ is well defined , it holds that*

$$f(y_N) - f_* \leq \left(\prod_{k=0}^{N-1} \rho(\tilde{\mu}_k) \right) (f(x_0) - f_*)$$

where $\rho(\tilde{\mu}) = \frac{1}{1+\frac{\tilde{\mu}}{L}}$. Otherwise $\nabla f(y_k) = 0$ with $k \in [0, N]$.

Proof. Use Lemma 5 recursively and notice that $V(x_0, y_0) = f(x_0) - f_*$. ■

In fact, these results on (Acc. Variant I) and (Acc. Variant II) also hold under Hölderian error bounds (Bolte et al., 2007, 2017) (also known as Kurdyka-Łojasewicz, Polyak-Łojasewicz, quadratic growth, etc.) which require the existence of $\mu > 0$ such that for all $x \in \mathbb{R}^n$, $f(x) - f_* \leq \frac{1}{2\mu}\|\nabla f(x)\|^2$. This condition holds in particular for strongly convex function but is much weaker.

Corollary 7 *Under the conditions of Proposition 6, if there exists $\mu > 0$ such that for all $x \in \mathbb{R}^n$, $f(x) - f_* \leq \frac{1}{2\mu}\|\nabla f(x)\|^2$ then after $N \in \mathbb{N}$ iterations*

$$f(y_N) - f_* \leq \left(1 + \frac{\mu}{L}\right)^{-N} (f(x_0) - f_*).$$

Looking at Proposition 6 more closely, we notice that when the estimates $\tilde{\mu}_k$ are larger than $\sqrt{L\mu}$, the adaptive accelerated method exhibits an accelerated linear convergence rate $O(1 - \sqrt{\frac{\mu}{L}})$. It remains to study the convergence of the adaptive method in the regime where $\tilde{\mu}_k$ is small. In this case, we provide another robustness result for the AGM algorithm when the momentum β_k is close enough to its classical value (Const-mom).

Lemma 8 (Appendix C.3) *Let $f \in \mathcal{F}_{\mu,L}$, some iteration number $k \in \mathbb{N}$, and consider Algorithm 2 with*

$$\frac{\sqrt{L}-\sqrt[4]{L\mu}}{\sqrt{L}+\sqrt[4]{L\mu}} \leq \beta_k \leq \beta_* = \frac{\sqrt{L}-\sqrt{\mu}}{\sqrt{L}+\sqrt{\mu}}.$$

Then, for any $x_k, y_k \in \mathbb{R}^n$, it holds that

$$V(x_{k+1}, y_{k+1}) \leq \rho V(x_k, y_k) \tag{4}$$

where $V(x,y) = \frac{L}{2}\|\frac{1}{\sqrt{\rho}}(x-x_*) - \sqrt{\rho}(y-x_*)\|^2 + f(y) - f_*$ and $\rho = \left(1+\left(\frac{\mu}{L}\right)^{\frac{3}{4}}\right)^{-1}$.

This lemma guarantees a linear convergence rate $O\left(1-\left(\frac{\mu}{L}\right)^{3/4}\right)^k$ that is slower than the accelerated rate with full knowledge of μ but faster than the gradient rate. We now combine the convergence results for the two regimes of $\tilde{\mu}_k$, and get a global linear convergence rate for (Acc. Variant II).

Proposition 9 (Appendix C.4) *Let $f \in \mathcal{F}_{\mu,L}$, and $N \in \mathbb{N}$ be a number of iterations. We consider Algorithm 2 with (Acc. Variant II), and let $\{y_k, x_k\}_k$ be the iterates of the method. Then, for any $x_0 \in \mathbb{R}^n$, such that the sequence $\{\tilde{\mu}_k\}_k$ is well defined, we let $m \in N$ be the first integer such that $\frac{\|\nabla f(y_{m+1})\|^2}{2(f(y_{m+1})-f_*)} \leq \sqrt{L\mu}$, (let $m = \infty$ if this never happens during the N iterations),*

$$f(y_N) - f_* \leq \begin{cases} \rho_1^N\left(\frac{L}{2}\left(\frac{1}{\sqrt{\rho_1}}-\sqrt{\rho_1}\right)^2\|x_0-x_*\|^2 + f(x_0) - f_*\right) & \textit{if } m = 0, \\ \rho_2^N(f(x_0)-f_*) & \textit{if } m = \infty, \\ C\rho_1^{N-m}\rho_2^m(f(x_0)-f_*) & \textit{otherwise,} \end{cases}$$

where $C = \left(\left(\frac{1}{\rho_1}-1\right)\left(1+\sqrt{\frac{L}{2\mu}}\right)^2+1\right)$, $\rho_1 = \left(1+\left(\frac{\mu}{L}\right)^{\frac{3}{4}}\right)^{-1}$ and $\rho_2 = \left(1+\sqrt{\frac{\mu}{L}}\right)^{-1}$. Otherwise $\nabla f(y_k) = 0$ with $k \in [0, N]$.

The previous convergence bound is only valid for (Acc. Variant II) mostly for technical reasons. Indeed the min is present in order to have at most one transition between the regime $\tilde{\mu}_k \geq \sqrt{L\mu}$ and $\tilde{\mu}_k \leq \sqrt{L\mu}$. In practice, however, we didn't observe any difference between the behaviours of (Acc. Variant I) and that of (Acc. Variant II).

4. Proof mechanisms

Starting with the work of Drori and Teboulle (2014), computer-aided worst-case analyses of convex optimization methods have provided a generic technique producing convergence rates for many classical first-order algorithms. The results in (Drori and Teboulle, 2014; Taylor et al., 2017) use an interpolation argument to write the problem of finding the worst case behavior of an algorithm, given a convergence criterion, as a tractable semidefinite program—often referred to as a Performance Estimation Program (PEP). We adapted the technique for generating the complexity bounds on gradient methods with Polyak steps.

Our proofs were obtained by searching for Lyapunov (or potential) functions (see e.g. (Bansal and Gupta, 2019) for a recent survey). Due to space constraints, we do not detail how these potentials were obtained here, and refer the reader to the discussions on PEPs in (Taylor and Bach, 2019; Taylor et al., 2018) for more details. A related line of works (equivalent in many situations) is that of integral quadratic constraints (Lessard et al., 2016), which leverage results from control theory to perform worst-case complexity analysis. All these approaches were originally developed for non adaptive methods and in what follows, we show how we used the PEP approach for adaptive algorithms. A similar reasoning would allow adapting IQCs for adaptive methods as well.

To fix ideas and illustrate our procedure, we first analyze the worst case complexity of a variant of the classical gradient method with Polyak steps, and show improved convergence bounds compared to classical results (see Hazan and Kakade (2019) for a recent treatment). We consider the gradient method with Polyak steps described in Algorithm 1 with (Variant I) for $f \in \mathcal{F}_{\mu,L}$. Notice that there is a factor two in the step-size that is not present in the original Polyak step. This factor simplifies, and improves, the analysis for the convergence in terms of distance to the optimum.

To prove a linear convergence rate, we can focus on the improvement yielded by a single iteration of the form

$$x_{k+1} := x_k - \gamma_k \nabla f(x_k), \quad \text{where} \quad \gamma_k := 2\frac{f(x_k) - f_*}{\|\nabla f(x_k)\|^2}. \tag{5}$$

We seek to bound the worst case (i.e., smallest) decrease in $\|x_{k+1} - x_*\|^2$ relative to $\|x_k - x_*\|^2$ when x_{k+1} is obtained using the iteration in (5) for any function $f \in \mathcal{F}_{\mu,L}$ and any point x_k. In other words we seek to solve the following optimization problem

$$\begin{array}{ll} \text{maximize} & \dfrac{\|x_{k+1} - x_*\|^2}{\|x_k - x_*\|^2} \\ \text{subject to} & x_{k+1} = x_k - 2\frac{f(x_k) - f_*}{\|\nabla f(x_k)\|^2}\nabla f(x_k), \\ & f \in \mathcal{F}_{\mu,L},\ x_k \in \mathbb{R}^n. \end{array} \tag{6}$$

in the variables $f \in \mathcal{F}_{\mu,L}$ and $x_k, x_{k+1}, x_*, \nabla f(x_k) \in \mathbb{R}^n$, with parameter $f^* \in \mathbb{R}$. The following lemma from (Taylor et al., 2017) shows necessary conditions satisfied by any function $f \in \mathcal{F}_{\mu,L}$.

Lemma 10 *(Taylor et al., 2017, Theorem 4) Given $f \in \mathcal{F}_{\mu,L}$, for any $(x, y) \in \mathbb{R}^n \times \mathbb{R}^n$*

$$\begin{array}{l} f(x) - f(y) + \nabla f(x)^T(y - x) + \frac{1}{2L}\|\nabla f(x) - \nabla f(y)\|^2 \\ + \frac{\mu}{2(1-\frac{\mu}{L})}\|x - y - \frac{1}{L}(\nabla f(x) - \nabla f(y))\|^2 \leq 0 \end{array}$$

The key argument in (Drori and Teboulle, 2014; Taylor et al., 2017) is that the constraint on the regularity of the function f in problem (6) can be replaced by a finite number of inequalities from Lemma 10. We get an upper bound on the optimum of problem (6) by relaxing the constraint $f \in \mathcal{F}_{\mu,L}$, keeping just two inequalities from Lemma 10 relating x_k and x_* to obtain the following relaxed problem

$$\begin{array}{ll} \text{maximize} & \dfrac{\|x_{k+1} - x_*\|^2}{\|x_k - x_*\|^2} \\ \text{subject to} & f_k - f_* + g_k^T(x_* - x_k) + \frac{1}{2L}\|g_k\|^2 + \frac{\mu}{2(1-\frac{\mu}{L})}\|x_k - x_* - \frac{1}{L}g_k\|^2 \leq 0 \\ & f_* - f_k + \frac{1}{2L}\|g_k\|^2 + \frac{\mu}{2(1-\frac{\mu}{L})}\|x_k - x_* - \frac{1}{L}g_k\|^2 \leq 0 \\ & x_{k+1} = x_k - 2\frac{f_k - f_*}{\|g_k\|^2}g_k \end{array} \tag{7}$$

in the variables $x_k, x_*, g_k \in \mathbb{R}^n$ and $f_k, f_* \in \mathbb{R}$. This relaxed problem is finite dimensional, but still depends on the dimension of the ambient space while we are interested in convergence rates independent of the dimension. One of the key insights of the PEP approach is to notice that (7) can be kernelized, i.e., written in terms of the quadratic variables $X_k = \|x_k - x_*\|^2$, $G_k = \|g_k\|^2$, $GX_k = g_k^T(x_* - x_k)$ in addition to f_k and f_*. Indeed, problem (7) is equivalent to solving

$$\begin{array}{ll} \text{maximize} & 1 + 4\frac{f_k - f_*}{G_k}\frac{GX_k}{X_k} + 4\frac{(f_k - f_*)^2}{G_k X_k} \\ \text{subject to} & f_k - f_* + GX_k + \frac{1}{2L}G_k + \frac{\mu}{2(1-\frac{\mu}{L})}\left(X_k + \frac{2}{L}GX_k + \frac{1}{L^2}G_k\right) \leq 0 \\ & f_* - f_k + \frac{1}{2L}G_k + \frac{\mu}{2(1-\frac{\mu}{L})}\left(X_k + \frac{2}{L}GX_k + \frac{1}{L^2}G_k\right) \leq 0 \\ & \begin{pmatrix} X_k & GX_k \\ GX_k & G_k \end{pmatrix} \succcurlyeq 0 \end{array} \tag{8}$$

in the variables $X_k, G_k, GX_k, f_k, f_* \in \mathbb{R}$. This new problem has only five real variables but is not readily tractable because of the non-linearity in the objective. By homogeneity we can impose $X_k = 1$ without loss of generality. We introduce a step size variable γ to rewrite the problem as

$$\begin{array}{ll} \text{maximize} & \rho(\gamma) \\ \text{subject to} & \gamma \in \mathbb{R} \end{array} \tag{9}$$

where

$$\begin{array}{lll} \rho(\gamma) := & \text{max.} & 1 + 2\gamma GX_k + 2(f_k - f_*)\gamma \\ & \text{s.t.} & f_k - f_* + GX_k + \frac{1}{2L}G_k + \frac{\mu}{2(1-\frac{\mu}{L})}\left(X_k + \frac{2}{L}GX_k + \frac{1}{L^2}G_k\right) \leq 0 \\ & & f_* - f_k + \frac{1}{2L}G_k + \frac{\mu}{2(1-\frac{\mu}{L})}\left(X_k + \frac{2}{L}GX_k + \frac{1}{L^2}G_k\right) \leq 0 \\ & & \begin{pmatrix} X_k & GX_k \\ GX_k & G_k \end{pmatrix} \succcurlyeq 0 \\ & & X_k = 1,\ G_k\gamma = 2(f_k - f_*) \end{array} \tag{10}$$

which is a semidefinite program. Given γ, $\rho(\gamma)$ can thus be computed efficiently and our relaxation upper bound on the convergence rate of the method is then given by the maximum value of $\rho(\gamma)$. Note that due to the definition of the step size, we only need to study $\rho(\gamma)$ on the interval $[\frac{1}{L}, \frac{1}{\mu}]$. Figure 2 (left) plots $\rho(\gamma)$ for fixed values $\mu = 0.1$ and $L = 1$, and shows (right) the maximum value of $\rho(\gamma)$ for various condition numbers. In this experiment, the worst case convergence rates we obtained numerically appear to perfectly match the bound $(L-\mu)^2/(L+\mu)^2$.

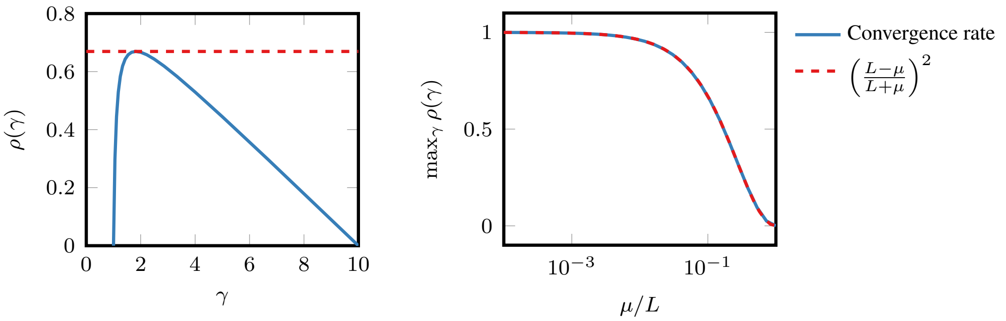

Figure 2: Left: we plot $\rho(\gamma)$, by solving (10) with $\mu = 0.1$ and $L = 1$. Right: Worst case rate $\max_\gamma\ \rho(\gamma)$, by solving (9), versus inverse condition number.

These numerical observations can in fact be proven analytically as follows. Given a target convergence rate $\rho \in [0, 1]$, we need to show that

$$\|x_{k+1} - x_*\|^2 - \rho\|x_k - x_*\|^2 \leq 0 \tag{11}$$

for all *feasible* values of $x_k, x_{k+1}, x_* \in \mathbb{R}^n$, satisfying the constraints of problem (7). In the spirit of the Putinar *positivstellensatz* used in sum of squares solutions of semi-algebraic optimization problems (Putinar, 1993; Lasserre, 2001; Parrilo, 2000), we seek to write a certificate of the validity of inequality (11) using a positively weighted sum of valid inequalities satisfied by $x_k, x_{k+1}, x_* \in \mathbb{R}^n$ in (7). Here, this means writing

$$\begin{aligned}
&\|x_{k+1} - x_*\|^2 - \rho(\gamma_k)\|x_k - x_*\|^2 = \\
&\lambda_1 \left[f(x_k) - f_* + \nabla f(x_k)^T(x_* - x_k) + \tfrac{1}{2L}\|\nabla f(x_k)\|^2 + \tfrac{\mu}{2(1-\frac{\mu}{L})}\|x_k - x_* - \tfrac{1}{L}\nabla f(x_k)\|^2 \right] \\
&+\lambda_2 \left[f_* - f(x_k) + \tfrac{1}{2L}\|\nabla f(x_k)\|^2 + \tfrac{\mu}{2(1-\frac{\mu}{L})}\|x_k - x_* - \tfrac{1}{L}\nabla f(x_k)\|^2 \right] \\
&+\lambda_3 \left[2(f(x_k) - f_*) - \gamma_k\|\nabla f(x_k)\|^2 \right] \\
&\leq 0
\end{aligned}$$

for some $\lambda_1, \lambda_2 \geq 0$, $\lambda_3 \in \mathbb{R}$, and using the fact $x_{k+1} = x_k - \gamma_k \nabla f(x_k)$ by construction. Through symbolic computations, or by trial and error, inferring a target convergence rate from optimal values of the semidefinite program, the proof consists in showing that we can pick

$$\rho(\gamma_k) = \tfrac{(\gamma_k L-1)(1-\gamma_k\mu)}{\gamma_k(L+\mu)-1}, \quad \lambda_1 = \tfrac{2\gamma_k(\gamma_k L-1)}{\gamma_k(L+\mu)-1}, \quad \lambda_2 = \tfrac{2\gamma_k(1-\gamma_k\mu)}{\gamma_k(L+\mu)-1} \quad \text{and} \quad \lambda_3 = \tfrac{\gamma_k(2-\gamma_k(L+\mu))}{\gamma_k(L+\mu)-1}.$$

In practice, the numerical solution of the semidefinite program in (10) giving $\rho(\gamma)$ can be used to greedily narrow down the list of valid inequalities required by the proof.

Note that since (9) is a semialgebraic problem, we could have used sum-of-squares techniques to prove the convergence rate. However, the multipliers and the rates are fractions in γ_k. Since one usually doesn't know in advance the form of the denominators, one needs relatively high degree polynomials in the SOS program. This means this approach suffers from the usual SOS issues of poor conditioning and scaling.

5. Numerical experiments

Numerical experiments with our algorithms are provided in Figure 3, respectively on least squares, regularized logistic regression and Lasso problems. For solving the Lasso problems, we used a proximal variant of Algorithm 2, whose details are provided in Appendix C.5. We respectively used the Sonar (Gorman and Sejnowski, 1988) and Musk (Dietterich et al., 1997) datasets.

In the experiments, when no analytical version of f_* was available (for logistic regression and Lasso), we used ad hoc methods to obtain higher precision estimates of f_*. As previously discussed, a fundamental next step is to incorporate successive refinements of a lower bound on f_* (a first step in this direction is for example (Hazan and Kakade, 2019)). One should notice that vanilla Polyak steps without momentum actually perform very well when they apply (see Appendix C.6 for a discussion on the performances of vanilla Polyak steps). We believe that modifying the accelerated Polyak so that it also adapts to the Lipschitz constant could make it more competitive, but the current state of the proofs does not allow it yet.

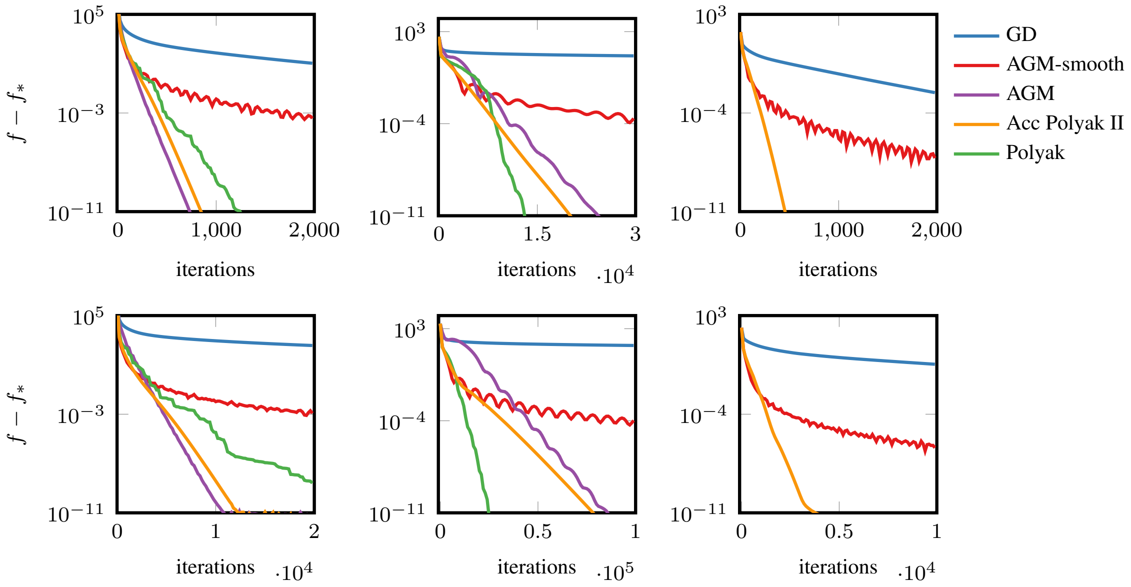

Figure 3: Top: Sonar dataset. Bottom: Musk dataset. Left: Least squares. Middle: Logistic regression with Tikhonov regularization (regularization parameter 10^{-3}). Right: LASSO (regularization parameter 1). For Polyak steps the best iterate is displayed. No tuning in any of the methods.

6. Conclusion and perspectives

We provided a momentum version of the Polyak steps, with an accelerated linear convergence rate. When f_* is available, this method is easy to implement and requires no tuning at all. On the way, we illustrated the methodology that was used for obtaining those rates, for the special case of a gradient method with Polyak steps. This methodology relies on the recent developments on performance estimation problems (Drori and Teboulle, 2014; Taylor et al., 2017), which we adapted for studying our adaptive methods.

One of the main questions that remains open is to understand whether there exists a way to get the same convergence guarantees without using f_*. The robustness result of Lemma 3 is reassuring in the sense that a misspecified f_* cannot break the algorithm (albeit worsening the convergence rate). We are confident that ideas introduced by Hazan and Kakade (2019) for Polyak steps could be used for our algorithm as well, and could potentially allow dealing with unknown f_* at a reasonable cost. However it still appears as an unnatural trick that adds complexity to the method.

Let us mention that the problem of designing theoretically supported adaptive methods is an open question. We managed to design (Variant II), for which we used our methodology—to find a method that would use Polyak steps to make the primal gap decrease linearly at each iterations—, but designing adaptive accelerated methods appeared as much more daunting task.

Finally, we note that regular Polyak steps do not enjoy a known (working) proximal extension. On the contrary, our results suggest that its accelerated counterparts do work with proximal operators (for minimizing composite objective functions with a non-smooth term). Therefore, developing the theory in this direction is another natural next step.

Codes The code used to obtain Figures 2-4-3 and to verify proofs is available at https://github.com/mathbarre/PerformanceEstimationPolyakSteps.

Acknowledgments

The authors thank Konstantin Mishchenko and Yura Malitsky for insightful discussions on Polyak steps, and comments on a preliminary version of this work. The authors also thank three anonymous reviewers for their constructive feedbacks on the manuscript.

MB acknowledges support from an AMX fellowship. AT acknowledges support from the European Research Council (grant SEQUOIA 724063). AA is at CNRS & département d'informatique, École normale supérieure, UMR CNRS 8548, 45 rue d'Ulm 75005 Paris, France, INRIA and PSL Research University. AA acknowledges support from the French government under management of Agence Nationale de la Recherche as part of the "Investissements d'avenir" program, reference ANR-19-P3IA-0001 (PRAIRIE 3IA Institute), the ML & Optimisation joint research initiative with the fonds AXA pour la recherche and Kamet Ventures, as well as a Google focused award.

References

Yossi Arjevani and Ohad Shamir. On the iteration complexity of oblivious first-order optimization algorithms. In *International Conference on Machine Learning*, pages 908–916, 2016.

Hilal Asi and John C. Duchi. Stochastic (approximate) proximal point methods: Convergence, optimality, and adaptivity. *SIAM Journal on Optimization*, 29(3):2257–2290, 2019.

Nikhil Bansal and Anupam Gupta. Potential-function proofs for gradient methods. *Theory of Computing*, 15(1):1–32, 2019.

Jonathan Barzilai and Jonathan M. Borwein. Two-point step size gradient methods. *IMA journal of numerical analysis*, 8(1):141–148, 1988.

Jérôme Bolte, Aris Daniilidis, and Adrian Lewis. The lojasiewicz inequality for nonsmooth subanalytic functions with applications to subgradient dynamical systems. *SIAM Journal on Optimization*, 17(4):1205–1223, 2007.

Jérôme Bolte, Trong Phong Nguyen, Juan Peypouquet, and Bruce W. Suter. From error bounds to the complexity of first-order descent methods for convex functions. *Mathematical Programming*, 165(2):471–507, 2017.

Stephen Boyd, Lin Xiao, and Almir Mutapcic. Subgradient methods. *lecture notes of EE392o, Stanford University, Autumn Quarter*, 2004:2004–2005, 2003.

Richard H. Byrd, Jorge Nocedal, and Ya-Xiang Yuan. Global convergence of a class of quasi-Newton methods on convex problems. *SIAM Journal on Numerical Analysis*, 24(5):1171–1190, 1987.

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.

Thomas G. Dietterich, Richard H. Lathrop, and Tomás Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial intelligence*, 89(1-2):31–71, 1997.

Yoel Drori and Marc Teboulle. Performance of first-order methods for smooth convex minimization: a novel approach. *Mathematical Programming*, 145(1-2):451–482, 2014.

Olivier Fercoq and Zheng Qu. Restarting accelerated gradient methods with a rough strong convexity estimate. *arXiv preprint arXiv:1609.07358*, 2016.

Roger Fletcher. Practical methods of optimization, 1987.

Roger Fletcher. On the Barzilai-Borwein method. In *Optimization and control with applications*, pages 235–256. Springer, 2005.

Roger Fletcher and Colin M. Reeves. Function minimization by conjugate gradients. *The computer journal*, 7(2):149–154, 1964.

Paul R. Gorman and Terrence J. Sejnowski. Analysis of hidden units in a layered network trained to classify sonar targets. *Neural Networks*, 1:75, 1988.

William W. Hager and Hongchao Zhang. A survey of nonlinear conjugate gradient methods. *Pacific journal of Optimization*, 2(1):35–58, 2006.

Elad Hazan and Sham Kakade. Revisiting the Polyak step size. *arXiv preprint arXiv:1905.00313*, 2019.

Magnus R. Hestenes and Eduard Stiefel. Methods of conjugate gradients for solving linear systems. *Journal of research of the National Bureau of Standards*, 49(6):409–436, 1952.

Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 795–811. Springer, 2016.

Jean-Bernard Lasserre. Global optimization with polynomials and the problem of moments. *SIAM Journal on Optimization*, 11(3):796–817, 2001.

Laurent Lessard, Benjamin Recht, and Andrew Packard. Analysis and design of optimization algorithms via integral quadratic constraints. *SIAM Journal on Optimization*, 26(1):57–95, 2016.

Qihang Lin and Lin Xiao. An adaptive accelerated proximal gradient method and its homotopy continuation for sparse optimization. In *ICML*, pages 73–81, 2014.

Dong C. Liu and Jorge Nocedal. On the limited memory BFGS method for large scale optimization. *Mathematical programming*, 45(1-3):503–528, 1989.

Yura Malitsky and Konstantin Mishchenko. Adaptive gradient descent without descent. *arXiv preprint arXiv:1910.09529*, 2019.

Angelia Nedic and Dimitri P. Bertsekas. Incremental subgradient methods for nondifferentiable optimization. *SIAM Journal on Optimization*, 12(1):109–138, 2001.

Yurii Nesterov. A method of solving a convex programming problem with convergence rate $O(1/k^2)$. *Soviet Mathematics Doklady*, 27(2):372–376, 1983.

Yurii Nesterov. Gradient methods for minimizing composite functions. *Mathematical Programming*, 140(1):125–161, 2013.

Yurii Nesterov. *Lectures on convex optimization*, volume 137. Springer, 2018.

Brendan O'Donoghue and Emmanuel Candes. Adaptive restart for accelerated gradient schemes. *Foundations of computational mathematics*, 15(3):715–732, 2015.

Pablo A. Parrilo. *Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization*. PhD thesis, California Institute of Technology, 2000.

Boris T. Polyak. The conjugate gradient method in extremal problems. *USSR Computational Mathematics and Mathematical Physics*, 9(4):94–112, 1969.

Boris T. Polyak. *Introduction to optimization*. Optimization Software, New York, 1987.

Mihai Putinar. Positive polynomials on compact semi-algebraic sets. *Indiana University Mathematics Journal*, 42(3):969–984, 1993.

William Rodi and Randall L Mackie. Nonlinear conjugate gradients algorithm for 2-d magnetotelluric inversion. *Geophysics*, 66(1):174–187, 2001.

Vincent Roulet and Alexandre d'Aspremont. Sharpness, restart and acceleration. In *Advances in Neural Information Processing Systems*, pages 1119–1129, 2017.

Katya Scheinberg, Donald Goldfarb, and Xi Bai. Fast first-order methods for composite convex optimization with backtracking. *Foundations of Computational Mathematics*, 14(3):389–417, 2014.

Adrien Taylor and Francis Bach. Stochastic first-order methods: non-asymptotic and computer-aided analyses via potential functions. In *Proceedings of the Thirty-Second Conference on Learning Theory (COLT)*, volume 99, pages 2934–2992. PMLR, 2019.

Adrien Taylor, Bryan Van Scoy, and Laurent Lessard. Lyapunov functions for first-order methods: Tight automated convergence guarantees. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, volume 80, pages 4897–4906. PMLR, 2018.

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(1-2): 307–345, 2017.

Stefan Volkwein. Nonlinear conjugate gradient methods for the optimal control of laser surface hardening. *Optimization Methods and Software*, 19(2):179–199, 2004.

Jing Zhao, Edwin AH Vollebregt, and Cornelis W Oosterlee. A fast nonlinear conjugate gradient based method for 3d concentrated frictional contact problems. *Journal of Computational Physics*, 288:86–100, 2015.

Appendix A. Proof of Proposition 1

Proof. For proving the desired result, it is only necessary to consider a single iteration of Algorithm 1 with (Variant I). We use the following (in)equalities obtained from Lemma 10:

- smoothness and strong convexity between x_k and x_*, with multiplier $\lambda_1 = \frac{2\gamma_k(\gamma_k L-1)}{\gamma_k(L+\mu)-1}$:

$$f(x_k) - f_* + \nabla f(x_k)^T(x_* - x_k) + \tfrac{1}{2L}\|\nabla f(x_k)\|^2 + \tfrac{\mu}{2(1-\frac{\mu}{L})}\|x_k - x_* - \tfrac{1}{L}\nabla f(x_k)\|^2 \leq 0,$$

- smoothness and strong convexity between x_* and x_k, with multiplier $\lambda_2 = \frac{2\gamma_k(1-\gamma_k\mu)}{\gamma_k(L+\mu)-1}$:

$$f_* - f(x_k) + \tfrac{1}{2L}\|\nabla f(x_k)\|^2 + \tfrac{\mu}{2(1-\frac{\mu}{L})}\|x_k - x_* - \tfrac{1}{L}\nabla f(x_k)\|^2 \leq 0,$$

- definition of the step-size policy, with multiplier $\lambda_3 = \frac{\gamma_k(2-\gamma_k(L+\mu))}{\gamma_k(L+\mu)-1}$:

$$2(f(x_k) - f_*) - \gamma_k\|\nabla f(x_k)\|^2 = 0.$$

Given that $\lambda_1, \lambda_2 \geq 0$ (since $\frac{1}{L} \leq \gamma_k \leq \frac{1}{\mu}$), the following weighted sum is a valid inequality:

$$\begin{aligned} 0 \geq &\lambda_1 \left[f(x_k) - f_* + \nabla f(x_k)^T(x_* - x_k) + \tfrac{1}{2L}\|\nabla f(x_k)\|^2 + \tfrac{\mu}{2(1-\frac{\mu}{L})}\|x_k - x_* - \tfrac{1}{L}\nabla f(x_k)\|^2\right] \\ &+ \lambda_2 \left[f_* - f(x_k) + \tfrac{1}{2L}\|\nabla f(x_k)\|^2 + \tfrac{\mu}{2(1-\frac{\mu}{L})}\|x_k - x_* - \tfrac{1}{L}\nabla f(x_k)\|^2\right] \\ &+ \lambda_3 \left[2(f(x_k) - f_*) - \gamma_k\|\nabla f(x_k)\|^2\right]. \end{aligned}$$

Using the fact that $x_{k+1} = x_k - \gamma_k \nabla f(x_k)$, this weighted sum can be reformulated exactly as

$$\|x_{k+1} - x_*\|^2 - \rho(\gamma_k)\|x_k - x_*\|^2 \leq 0$$

(one can verify that both expressions are equal) with $\rho(\gamma) = \frac{(\gamma L-1)(1-\gamma\mu)}{\gamma(L+\mu)-1}$. Therefore, after N iterations, we get

$$\|x_N - x_*\|^2 \leq \left(\prod_{i=0}^{N-1} \rho(\gamma_i)\right)\|x_0 - x_*\|^2.$$

In addition, distance to optimality decreases, in the worst-case, with rate $\max_\gamma \rho(\gamma)$, with

$$\tfrac{(L-\mu)^2}{(L+\mu)^2} = \max\left\{\rho(\gamma) \mid \tfrac{1}{L} \leq \gamma \leq \tfrac{1}{\mu}\right\}.$$

because $\rho(\gamma)$ is a concave function of γ on the interval $[\frac{1}{L}, \frac{1}{\mu}]$, as $\rho''(\gamma) = -\frac{2L\mu}{(\gamma(L+\mu)-1)^3} \leq 0$, whose maximum is attained at $\gamma_* = \frac{2}{L+\mu}$. Note that substituting the expression of γ_k inside the interpolation inequalities, instead of using it as an independent equality constraints, yields a considerably less tractable result. ■

Appendix B. Proof of Proposition 2

Proof. Let us consider a single iteration of Algorithm 1, with step sizes (Variant II). The proof is a consequence of the following combination of inequalities obtained from Lemma 10:

- smoothness and strong convexity between x_k and x_*, with multiplier $\lambda_1 = \gamma_k\mu(L\gamma_k - 1)$:

$$f(x_k) - f_* + \nabla f(x_k)^T(x_* - x_k) + \tfrac{1}{2L}\|\nabla f(x_k)\|^2 + \tfrac{\mu}{2(1-\frac{\mu}{L})}\|x_k - x_* - \tfrac{1}{L}\nabla f(x_k)\|^2 \leq 0,$$

- smoothness and strong convexity between x_{k+1} and x_*, with multiplier $\lambda_2 = \gamma_k\mu$:

$$\begin{aligned} f(x_{k+1}) - f_* + \nabla f(x_{k+1})^T(x_* - x_{k+1}) + \tfrac{1}{2L}\|\nabla f(x_{k+1})\|^2 \\ + \tfrac{\mu}{2(1-\frac{\mu}{L})}\|x_{k+1} - x_* - \tfrac{1}{L}\nabla f(x_{k+1})\|^2 \leq 0, \end{aligned}$$

- smoothness and strong convexity between x_{k+1} and x_k, with multiplier $\lambda_3 = 1 - \gamma_k\mu$:

$$\begin{aligned} f(x_{k+1}) - f(x_k) + \nabla f(x_{k+1})^T(x_k - x_{k+1}) + \tfrac{1}{2L}\|\nabla f(x_{k+1}) - \nabla f(x_k)\|^2 \\ + \tfrac{\mu}{2(1-\frac{\mu}{L})}\|x_{k+1} - x_k - \tfrac{1}{L}(\nabla f(x_{k+1}) - \nabla f(x_k))\|^2 \leq 0, \end{aligned}$$

- definition of the step-size policy, with multiplier $\lambda_4 = \frac{\gamma_k}{2}((L+\mu)\gamma_k - 2)$:

$$(2L^2\gamma_k - 4L)(f(x_k) - f_*) + \|\nabla f(x_k)\|^2 = 0.$$

Given that $\lambda_1, \lambda_2, \lambda_3 \geq 0$ (due to $\frac{1}{L} \leq \gamma_k \leq \frac{2-\frac{\mu}{L}}{L}$), the following weighted sum is a valid inequality:

$$\begin{aligned} 0 \geq \lambda_1 & \left[f(x_k) - f_* + \nabla f(x_k)^T(x_* - x_k) + \tfrac{1}{2L}\|\nabla f(x_k)\|^2 + \tfrac{\mu}{2(1-\frac{\mu}{L})}\|x_k - x_* - \tfrac{1}{L}\nabla f(x_k)\|^2\right] \\ +\lambda_2 & \Big[f(x_{k+1}) - f_* + \nabla f(x_{k+1})^T(x_* - x_{k+1}) + \tfrac{1}{2L}\|\nabla f(x_{k+1})\|^2 \\ & + \tfrac{\mu}{2(1-\frac{\mu}{L})}\|x_{k+1} - x_* - \tfrac{1}{L}\nabla f(x_{k+1})\|^2\Big] \\ +\lambda_3 & \Big[f(x_{k+1}) - f(x_k) + \nabla f(x_{k+1})^T(x_k - x_{k+1}) + \tfrac{1}{2L}\|\nabla f(x_{k+1}) - \nabla f(x_k)\|^2 \\ & + \tfrac{\mu}{2(1-\frac{\mu}{L})}\|x_{k+1} - x_k - \tfrac{1}{L}(\nabla f(x_{k+1}) - \nabla f(x_k))\|^2\Big] \\ +\lambda_4 & \left[(2L^2\gamma_k - 4L)(f(x_k) - f_*) + \|\nabla f(x_k)\|^2\right]. \end{aligned}$$

Using the expression $x_{k+1} = x_k - \gamma_k\nabla f(x_k)$ (without substituting the expression of γ_k, whose value is encoded through the last equality of the list), this weighted sum can be rewritten exactly as

$$\begin{aligned} 0 \geq & f(x_{k+1}) - f_* - \rho(\gamma_k)(f(x_k) - f_*) \\ & + \tfrac{1}{2(L-\mu)}\left\|\nabla f(x_{k+1}) - L\mu\gamma_k(x_k - x_*) + (\gamma_k(L+\mu) - 1)\nabla f(x_k)\right\|^2 \end{aligned}$$

with $\rho(\gamma) = (L\gamma - 1)\,(L\gamma(3 - \gamma(L+\mu)) - 1)$ which, in turns, give

$$\begin{aligned} f(x_{k+1}) - f_* \leq& \rho(\gamma_k)(f(x_k) - f_*) \\ &- \tfrac{1}{2(L-\mu)} \left\| \nabla f(x_{k+1}) - L\mu\gamma_k(x_k - x_*) + (\gamma_k(L+\mu) - 1)\nabla f(x_k) \right\|^2 \\ \leq& \rho(\gamma_k)(f(x_k) - f_*). \end{aligned}$$

Therefore, after N iterations, we get

$$f(x_N) - f_* \leq \left(\prod_{i=0}^{N-1} \rho(\gamma_i) \right) (f(x_0) - f_*).$$

Finally, the worst-case convergence rate is $\max_\gamma \rho(\gamma)$ on the interval $[\frac{1}{L}, \frac{2-\mu/L}{L}]$, for which

$$\tfrac{(L-\mu)^2}{(L+\mu)^2} = \max \left\{ \rho(\gamma) \,\middle|\, \tfrac{1}{L} \leq \gamma \leq \tfrac{2-\mu/L}{L} \right\}.$$

The proof follows from the following steps:

- First, on the boundaries of the interval: (i) $\rho(\frac{1}{L}) = 0$ and (ii) $\rho(\frac{2-\frac{\mu}{L}}{L}) = \frac{(L-\mu)^4}{L^4} \leq \frac{(L-\mu)^2}{(L+\mu)^2}$.
- Secondly, in the interior of the interval: $\rho'(\gamma) = L(3L\gamma - 2)(2 - (L+\mu)\gamma)$ is zero at $\gamma_* = \frac{2}{L+\mu}$ (inside the interval).
- Therefore $\rho(\gamma_*) = \frac{(L-\mu)^2}{(L+\mu)^2}$ and this is the maximum on the interval.

■

Appendix C. Proof of § 3

C.1. Proof of Lemma 3

Proof. In this section, we use $\rho = 1 - \mu/L$. The proof consists in combining the following inequalities obtained from Lemma 10:

- smoothness and strong convexity between x_k and y_k with multiplier $\lambda_1 = \rho$:

$$\begin{aligned} f(x_k) - f(y_k) + \nabla f(x_k)^T(y_k - x_k) + \tfrac{1}{2L}\|\nabla f(x_k) - \nabla f(y_k)\|^2 \\ + \tfrac{\mu}{2(1-\frac{\mu}{L})}\|x_k - y_k - \tfrac{1}{L}(\nabla f(x_k) - \nabla f(y_k))\|^2 \leq 0, \end{aligned}$$

- smoothness and strong convexity between y_{k+1} and x_* with multiplier $\lambda_2 = 1 - \rho$:

$$\begin{aligned} f(y_{k+1}) - f_* + \nabla f(y_{k+1})^T(x_* - y_{k+1}) + \tfrac{1}{2L}\|\nabla f(y_{k+1})\|^2 \\ + \tfrac{\mu}{2(1-\frac{\mu}{L})}\|y_{k+1} - x_* - \tfrac{1}{L}\nabla f(y_{k+1})\|^2 \leq 0, \end{aligned}$$

- smoothness and strong convexity between y_{k+1} and x_k with multiplier $\lambda_3 = \rho$:

$$\begin{aligned} f(y_{k+1}) - f(x_k) + \nabla f(y_{k+1})^T(x_k - y_{k+1}) + \tfrac{1}{2L}\|\nabla f(y_{k+1}) - \nabla f(x_k)\|^2 \\ + \tfrac{\mu}{2(1-\frac{\mu}{L})}\|y_{k+1} - x_k - \tfrac{1}{L}(\nabla f(y_{k+1}) - \nabla f(x_k))\|^2 \leq 0. \end{aligned}$$

Given that $\lambda_1, \lambda_2, \lambda_3 \geq 0$, the following weighted sum is a valid inequality

$$\begin{aligned}0 \geq & \lambda_1 \Big[f(x_k) - f(y_k) + \nabla f(x_k)^T (y_k - x_k) + \tfrac{1}{2L} \|\nabla f(x_k) - \nabla f(y_k)\|^2 \\ & + \tfrac{\mu}{2(1-\frac{\mu}{L})} \|x_k - y_k - \tfrac{1}{L}(\nabla f(x_k) - \nabla f(y_k))\|^2 \Big] \\ & + \lambda_2 \Big[f(y_{k+1}) - f_* + \nabla f(y_{k+1})^T (x_* - y_{k+1}) + \tfrac{1}{2L} \|\nabla f(y_{k+1})\|^2 \\ & + \tfrac{\mu}{2(1-\frac{\mu}{L})} \|y_{k+1} - x_* - \tfrac{1}{L} \nabla f(y_{k+1})\|^2 \Big] \\ & + \lambda_3 \Big[f(y_{k+1}) - f(x_k) + \nabla f(y_{k+1})^T (x_k - y_{k+1}) + \tfrac{1}{2L} \|\nabla f(y_{k+1}) - \nabla f(x_k)\|^2 \\ & + \tfrac{\mu}{2(1-\frac{\mu}{L})} \|y_{k+1} - x_k - \tfrac{1}{L}(\nabla f(y_{k+1}) - \nabla f(x_k))\|^2 \Big],\end{aligned}$$

which can be reformulated exactly, using the notation

$$\begin{aligned}V(x,y) &= f(y) - f_* + \tfrac{L-\mu}{2} \|x - y\|^2 \\ y_{k+1} &= x_k - \tfrac{1}{L} \nabla f(x_k) \\ x_{k+1} &= y_{k+1} + \beta_k (y_{k+1} - y_k)\end{aligned}$$

along with the expression of ρ, in the form

$$\begin{aligned}0 \geq & V(x_{k+1}, y_{k+1}) - \rho V(x_k, y_k) \\ & + \tfrac{1}{2(L-\mu)} \|(1-\rho)(\nabla f(x_k) - L(x_k - x_*)) + \nabla f(y_{k+1})\|^2 \\ & + \tfrac{\rho}{2(L-\mu)} \|\nabla f(y_k) - \nabla f(x_k) + \mu(x_k - y_k)\|^2 \\ & + \tfrac{(1-\beta^2)\rho}{2L} \|\nabla f(x_k) + L(y_k - x_k)\|^2.\end{aligned}$$

Therefore, using the assumption $\beta_k \in [0,1]$, we finally arrive to the desired

$$V(x_{k+1}, y_{k+1}) \leq \rho V(x_k, y_k).$$

■

C.2. Proof of Lemma 5

Proof. In this setting, we write $\rho(x) = \frac{1}{1+\frac{x}{L}}$. The proof consists in the following combination of inequalities obtained from Lemma 10:

- smoothness and convexity between y_{k+1} and x_k with multiplier $\lambda_1 = \rho(\tilde{\mu}_k)$:

$$f(y_{k+1}) - f(x_k) + \nabla f(y_{k+1})^T (x_k - y_{k+1}) + \tfrac{1}{2L} \|\nabla f(x_k) - \nabla f(y_{k+1})\|^2 \leq 0,$$

- convexity between x_k and y_k with multiplier $\lambda_2 = \rho(\tilde{\mu}_k)$:
$$f(x_k) - f(y_k) + \nabla f(x_k)^T(y_k - x_k) \leq 0,$$
- definition of $\tilde{\mu}_k$ with multiplier $\lambda_3 = \frac{1-\rho(\tilde{\mu}_k)}{2\tilde{\mu}_k}$:
$$2\tilde{\mu}_k(f(y_{k+1}) - f_*) - \|\nabla f(y_{k+1})\|^2 \leq 0$$
(we use an inequality so that it also holds for $\tilde{\mu}_k = \min\{\tilde{\mu}_{k-1}, \frac{\|\nabla f(y_{k+1})\|^2}{2(f(y_{k+1})-f_*)}\}$).

The weighted sum is a valid inequality given that $\lambda_1, \lambda_2, \lambda_3 \geq 0$:

$$\begin{aligned} 0 \geq \lambda_1 & \left[f(y_{k+1}) - f(x_k) + \nabla f(y_{k+1})^T(x_k - y_{k+1}) + \tfrac{1}{2L}\|\nabla f(x_k) - \nabla f(y_{k+1})\|^2\right] \\ & + \lambda_2 \left[f(x_k) - f(y_k) + \nabla f(x_k)^T(y_k - x_k)\right] \\ & + \lambda_3 \left[2\tilde{\mu}_k(f(y_{k+1}) - f_*) - \|\nabla f(y_{k+1})\|^2\right], \end{aligned}$$

which can be reformulated exactly, using the notation

$$\begin{aligned} V(x,y) &= f(y) - f_* + \tfrac{L}{2}\|x-y\|^2 \\ y_{k+1} &= x_k - \tfrac{1}{L}\nabla f(x_k) \\ x_{k+1} &= y_{k+1} + \beta_k(y_{k+1} - y_k) \\ \beta_k &= \frac{\sqrt{L} - \sqrt{\tilde{\mu}_k}}{\sqrt{L} + \sqrt{\tilde{\mu}_k}} \end{aligned}$$

along with the expression for $\rho(x)$, in the form

$$\begin{aligned} 0 \geq & V(x_{k+1}, y_{k+1}) - \rho(\tilde{\mu}_k)V(x_k, y_k) \\ & + \frac{\left(4L^2\sqrt{\frac{\tilde{\mu}_k}{L}} - L\left(\tilde{\mu}_k - 2\tilde{\mu}_k\sqrt{\frac{\tilde{\mu}_k}{L}}\right) - \tilde{\mu}_k^2\right)}{2L^2(L+\tilde{\mu}_k)\left(\sqrt{\frac{\tilde{\mu}_k}{L}}+1\right)^2}\|\nabla f(x_k) + L(y_k - x_k)\|^2, \end{aligned}$$

which, in turns, is equivalent to

$$\begin{aligned} V(x_{k+1}, y_{k+1}) \leq & \rho(\tilde{\mu}_k)V(x_k, y_k) - \frac{\left(4L^2\sqrt{\frac{\tilde{\mu}_k}{L}} - L\left(\tilde{\mu}_k - 2\tilde{\mu}_k\sqrt{\frac{\tilde{\mu}_k}{L}}\right) - \tilde{\mu}_k^2\right)}{2L^2(L+\tilde{\mu}_k)\left(\sqrt{\frac{\tilde{\mu}_k}{L}}+1\right)^2}\|\nabla f(x_k) + L(y_k - x_k)\|^2, \\ \leq & \rho(\tilde{\mu}_k)V(x_k, y_k) \end{aligned}$$

where the inequality follows from the sign of the term we removed, so it remains to show that

$$4L^2\sqrt{\tfrac{\tilde{\mu}_k}{L}} - L\left(\tilde{\mu}_k - 2\tilde{\mu}_k\sqrt{\tfrac{\tilde{\mu}_k}{L}}\right) - \tilde{\mu}_k^2 \geq 0 \quad \forall \tilde{\mu}_k \in [0, L].$$

Indeed, evaluating the sign of the previous expression boils down to study that of $g(x) = 4\sqrt{x} - (x - 2x\sqrt{x}) - x^2$ on $[0,1]$, which follows from:

$$g(x) \geq 3\sqrt{x} - x\sqrt{x} \geq 0 \quad \forall x \in [0,1].$$

■

C.3. Proof of Lemma 8

Proof. Our statement follows from a weighted sum of inequalities obtained from Lemma 10:

- smoothness and strong convexity between y_{k+1} and x_k, with multiplier $\lambda_1 = 1$:

$$\begin{aligned} f(y_{k+1}) - f(x_k) + \nabla f(y_{k+1})^T(x_k - y_{k+1}) + \tfrac{1}{2L}\|\nabla f(y_{k+1}) - \nabla f(x_k)\|^2 \\ + \tfrac{\mu}{2(1-\frac{\mu}{L})}\|y_{k+1} - x_k - \tfrac{1}{L}(\nabla f(y_{k+1}) - \nabla f(x_k))\|^2 \leq 0, \end{aligned}$$

- smoothness and strong convexity between x_k and x_*, with multiplier $\lambda_2 = 1 - \rho$:

$$\begin{aligned} f(x_k) - f_* + \nabla f(x_k)^T(x_* - x_k) + \tfrac{1}{2L}\|\nabla f(x_k)\|^2 \\ + \tfrac{\mu}{2(1-\frac{\mu}{L})}\|x_k - x_* - \tfrac{1}{L}\nabla f(x_k)\|^2 \leq 0, \end{aligned}$$

- convexity between x_k and y_k, with multiplier $\lambda_3 = \rho$:

$$f(x_k) - f(y_k) + \nabla f(x_k)^T(y_k - x_k) \leq 0.$$

The weighted sum is a valid inequality given that $\lambda_1, \lambda_2, \lambda_3 \geq 0$:

$$\begin{aligned} 0 \geq \lambda_1 \Bigg[& f(y_{k+1}) - f(x_k) + \nabla f(y_{k+1})^T(x_k - y_{k+1}) + \tfrac{1}{2L}\|\nabla f(y_{k+1}) - \nabla f(x_k)\|^2 \\ & + \tfrac{\mu}{2(1-\frac{\mu}{L})}\|y_{k+1} - x_k - \frac{1}{L}(\nabla f(y_{k+1}) - \nabla f(x_k))\|^2 \Bigg] \\ +\lambda_2 \Bigg[& f(x_k) - f_* + \nabla f(x_k)^T(x_* - x_k) + \tfrac{1}{2L}\|\nabla f(x_k)\|^2 \\ & + \tfrac{\mu}{2(1-\frac{\mu}{L})}\|x_k - x_* - \tfrac{1}{L}\nabla f(x_k)\|^2 \Bigg] \\ +\lambda_3 & \left[f(x_k) - f(y_k) + \nabla f(x_k)^T(y_k - x_k) \right]. \end{aligned}$$

This inequality can be reformulated using the notations

$$\begin{aligned} V(x, y) &= f(y) - f_* + \tfrac{L}{2}\|\tfrac{1}{\sqrt{\rho}}(x - x_*) - \sqrt{\rho}(y - x_*)\|^2 \\ y_{k+1} &= x_k - \tfrac{1}{L}\nabla f(x_k) \\ x_{k+1} &= y_{k+1} + \beta_k(y_{k+1} - y_k) \\ \beta &= \beta_k \end{aligned}$$

in the form

$$\begin{aligned} 0 \geq & V(x_{k+1}, y_{k+1}) - \rho V(x_k, y_k) + \tfrac{1}{2(L-\mu)}\|\nabla f(y_{k+1})\|^2 + \tfrac{1-\rho}{2L}\|\nabla f(x_k)\|^2 \\ & + \tfrac{L\left(\rho^3-\beta^2\right)}{2\rho}\|(y_k - x_*) + \tfrac{\beta\rho-\beta(\beta+1)+\rho^2}{\beta^2-\rho^3}(x_k - x_*) + \tfrac{\beta^2-\beta\rho+\beta-\rho^2}{\beta^2 L-L\rho^3}\nabla f(x_k)\|^2 \\ & + \tfrac{L^2(1-\rho)\left(\frac{\mu}{L}\rho(2\beta\rho-\beta(\beta+2)+\rho)+(\rho-1)(\beta-\rho)^2\right)}{2(\rho^3-\beta^2)(L-\mu)}\|x_k - x_* - \tfrac{1}{L}\nabla f(x_k)\|^2. \end{aligned}$$

It is then direct to reach

$$\begin{aligned} V(x_{k+1}, y_{k+1}) \leq & \rho V(x_k, y_k) - \tfrac{1}{2(L-\mu)}\|\nabla f(y_{k+1})\|^2 - \tfrac{1-\rho}{2L}\|\nabla f(x_k)\|^2 \\ & - \tfrac{L\left(\rho^3-\beta^2\right)}{2\rho}\|(y_k - x_*) + \tfrac{\beta\rho-\beta(\beta+1)+\rho^2}{\beta^2-\rho^3}(x_k - x_*) + \tfrac{\beta^2-\beta\rho+\beta-\rho^2}{\beta^2 L - L\rho^3}\nabla f(x_k)\|^2 \\ & - \tfrac{L^2(1-\rho)\left(\frac{\mu}{L}\rho(2\beta\rho-\beta(\beta+2)+\rho)+(\rho-1)(\beta-\rho)^2\right)}{2(\rho^3-\beta^2)(L-\mu)}\|x_k - x_* - \tfrac{1}{L}\nabla f(x_k)\|^2 \\ \leq & \rho V(x_k, y_k), \end{aligned}$$

where we used the facts that the following coefficients were nonnegative (proofs below) on the domain of interest:

- $\frac{1}{2(L-\mu)} \geq 0$ (clear from the assumption $\mu \leq L$),
- $\frac{1-\rho}{L} \geq 0$ (clear from $\rho \leq 1$),
- $\frac{L\left(\rho^3-\beta^2\right)}{2\rho} \geq 0$ follows from $\left(\rho^3 - \beta^2\right) \geq 0$, proved below,
- $\frac{L^2(1-\rho)\left(\frac{\mu}{L}\rho(2\beta\rho-\beta(\beta+2)+\rho)+(\rho-1)(\beta-\rho)^2\right)}{2(\rho^3-\beta^2)(L-\mu)} \geq 0$ follows from previous points along with

  $$\tfrac{\mu}{L}\rho(2\beta\rho - \beta(\beta+2) + \rho) + (\rho-1)(\beta-\rho)^2 \geq 0,$$

  which is alo proved below.

The missing proofs are as follow. First, let us define $\kappa := \frac{\mu}{L} \in [0,1]$, the (inverse) condition number, and recall that we want to prove the expressions above to be nonnegative when $\rho = \frac{1}{1+\kappa^{3/4}}$ and $\beta_- \leq \beta \leq \beta_+$ with $\beta_- = \frac{\sqrt{1-\sqrt[4]{\kappa}}}{\sqrt{1+\sqrt[4]{\kappa}}}$ and $\beta_+ = \frac{\sqrt{1-\sqrt{\kappa}}}{\sqrt{1+\sqrt{\kappa}}}$.

- To show that $\rho^3 - \beta^2 \geq 0$, let us remark that the expression is a second order polynomial in the variable β with negative curvature. Therefore, its minimum values are achieved on the boundary of the interval, and it is sufficient to show $\rho^3 - \beta_-^2 \geq 0$ and $\rho^3 - \beta_+^2 \geq 0$ for establishing our claim. For the case $\beta = \beta_-$, we get:

  $$\rho^3 - \beta_-^2 = \frac{\kappa^{1/4}\left(4-8\kappa^{1/4}+9\sqrt{\kappa}-4\kappa^{3/4}-4\kappa+9\kappa^{5/4}-8\kappa^{3/2}+4\kappa^{7/4}-\kappa^2\right)}{\left(1+\kappa^{1/4}\right)^3\left(1-\kappa^{1/4}+\sqrt{\kappa}\right)^3},$$

  and we need to show that $\left(4 - 8\kappa^{1/4} + 9\sqrt{\kappa} - 4\kappa^{3/4} - 4\kappa + 9\kappa^{5/4} - 8\kappa^{3/2} + 4\kappa^{7/4} - \kappa^2\right)$ is non negative for all $\kappa \in [0,1]$. For showing that, we perform the change of variable $x \leftarrow \kappa^{1/4}$ (which is invertible since $\kappa \in [0,1]$), and study the polynomial

  $$p_1(x) = -x^8 + 4x^7 - 8x^6 + 9x^5 - 4x^4 - 4x^3 + 9x^2 - 8x + 4,$$

such that

$$\begin{aligned}
p_1(x) &\geq 3x^7 - 8x^6 + 9x^5 - 4x^4 - 4x^3 + 9x^2 - 8x + 4 \\
&= 3x^7 - 8x^6 + 9x^5 - 4x^4 - 4x^3 + 5x^2 + 4(1-x)^2 \\
&\geq 3x^7 - 8x^6 + 9x^5 - 4x^4 - 4x^3 + 5x^2 \\
&\geq 3x^7 - 8x^6 + 9x^5 - 4x^4 + x^3 \\
&= 3x^7 - 8x^6 + 5x^5 + x^3(2x-1)^2 \\
&\geq x^5(3x^2 - 8x + 5) \\
&= x^5(1-x)(5-3x) \\
&\geq 0,
\end{aligned}$$

hence finally $\rho^3 - \beta_-^2 \geq 0$. For the case $\beta = \beta_+$, we obtain:

$$\rho^3 - \beta_+^2 = \frac{\sqrt{\kappa}\left(4 - 3\kappa^{1/4} + 6\kappa^{3/4} - 3\kappa - 3\kappa^{5/4} + 6\kappa^{3/2} - \kappa^{7/4} - 3\kappa^2 + 2\kappa^{9/4} - \kappa^{11/4}\right)}{\left(1+\kappa^{1/4}\right)^3\left(1+\sqrt{\kappa}\right)^2\left(1-\kappa^{1/4}+\sqrt{\kappa}\right)^3},$$

and we need to show that

$$\left(4 - 3\kappa^{1/4} + 6\kappa^{3/4} - 3\kappa - 3\kappa^{5/4} + 6\kappa^{3/2} - \kappa^{7/4} - 3\kappa^2 + 2\kappa^{9/4} - \kappa^{11/4}\right)$$

is nonnegative for all $\kappa \in [0,1]$. After changing variable $x \leftarrow \kappa^{1/4}$ (which is invertible since $\kappa \in [0,1]$), we study the polynomial

$$p_2(x) = -x^{11} + 2x^9 - 3x^8 - x^7 + 6x^6 - 3x^5 - 3x^4 + 6x^3 - 3x + 4$$

such that

$$\begin{aligned}
p_2(x) &\geq x^9 - 3x^8 - x^7 + 6x^6 - 3x^5 - 3x^4 + 6x^3 - 3x + 4 \\
&\geq x^9 - 3x^8 - x^7 + 6x^6 - 3x^5 - 3x^4 + 6x^3 + 1 \\
&\geq x^9 - 3x^8 - x^7 + 6x^6 + 1 \\
&\geq x^9 + 2x^6 + 1 \\
&\geq 0,
\end{aligned}$$

hence $\rho^3 - \beta_+^2 \geq 0$.

- Similarly, the expression $p_3(\kappa) = \big(\kappa\rho(2\beta\rho - \beta(\beta+2) + \rho) + (\rho-1)(\beta-\rho)^2\big)$ is also a second order polynomial in β, with leading coefficient

$$-(1-\rho) - \kappa\rho \leq -(1-\rho) \leq 0.$$

Therefore, this quadratic function is also concave and we only need to verify the inequality on the boundary of the interval $[\beta_-, \beta_+]$. In the case $\beta = \beta_-$, we get:

$$p_3(\beta_-) = \frac{\left(1-\sqrt{\kappa}+\kappa^{3/4}\right)\kappa^{7/4}}{\left(1+\kappa^{1/4}\right)^3\left(1-\kappa^{1/4}+\sqrt{\kappa}\right)^3} \geq 0.$$

For case $\beta = \beta_+$, we obtain:

$$p_3(\beta_+) = \frac{\kappa^{3/2}\left(4-7\kappa^{1/4}+4\sqrt{\kappa}+5\kappa^{3/4}-7\kappa+3\kappa^{5/4}+2\kappa^{3/2}-\kappa^{7/4}+\kappa^2\right)}{\left(1+\kappa^{1/4}\right)^3\left(1+\sqrt{\kappa}\right)^2\left(1-\kappa^{1/4}+\sqrt{\kappa}\right)^3},$$

and we need to show that $\left(\kappa^2 - \kappa^{7/4} + 2\kappa^{3/2} + 3\kappa^{5/4} - 7\kappa + 5\kappa^{3/4} + 4\sqrt{\kappa} - 7\sqrt[4]{\kappa} + 4\right)$ is nonnegative for $\kappa \in [0,1]$. We change variables $x \leftarrow \kappa^{1/4}$ (which is invertible since $\kappa \in [0,1]$), and study the polynomial

$$p_4(x) = x^8 - x^7 + 2x^6 + 3x^5 - 7x^4 + 5x^3 + 4x^2 - 7x + 4$$

on the interval $[0,1]$:

$$\begin{aligned} p_4(x) &= x^8 - x^7 + 2x^6 + 3x^5 - 7x^4 + 5x^3 + x + 4(1-x)^2 \\ &\geq x^3(x^5 - x^4 + 2x^3 + 3x^2 - 7x + 5) \\ &= x^3(x^5 - x^4 + 2x^3 - x^2 + x + 1 + 4(1-x)^2) \\ &\geq x^3(x^5 + x^3 + 1 + 4(1-x)^2) \\ &\geq 0, \end{aligned}$$

hence $p_3(\beta_+) \geq 0$, which concludes the proof.

■

C.4. Proof of Proposition 9

Proof. The case $m = 0$ results from Lemma 8 applied recursively and the case $m = \infty$ result from Proposition 6. In the following we consider that $m \in [1, N]$. Then for $(y_k, x_k)_{k \in [m+1,N]}$,

$$\frac{\sqrt{L} - \sqrt[4]{L\mu}}{\sqrt{L} + \sqrt[4]{L\mu}} \leq \beta_{k-1} \leq \frac{\sqrt{L} - \sqrt{\mu}}{\sqrt{L} + \sqrt{\mu}}$$

and Lemma 8 applies so

$$f(y_N) - f_* \leq \rho_1^{N-m}\left(\frac{L}{2}\|\frac{1}{\sqrt{\rho_1}}(x_m - x_*) - \sqrt{\rho_1}(y_m - x_*)\|^2 + f(y_m) - f_*\right)$$

and we have

$$\begin{aligned} &\frac{L}{2}\|\frac{1}{\sqrt{\rho_1}}(x_m - x_*) - \sqrt{\rho_1}(y_m - x_*)\|^2 + f(y_m) - f_* \\ &= \frac{L}{2}\left(\frac{1}{\rho_1} - 1\right)\|x_m - x_*\|^2 - \frac{L}{2}(1-\rho_1)\|y_m - x_*\|^2 + \frac{L}{2}\|x_m - y_m\|^2 + f(y_m) - f_* \\ &\leq \frac{L}{2}\left(\frac{1}{\rho_1} - 1\right)\|x_m - x_*\|^2 + \frac{L}{2}\|x_m - y_m\|^2 + f(y_m) - f_* \\ &\leq \frac{L}{2}\left(\frac{1}{\rho_1} - 1\right)(\|x_m - y_m\| + \|y_m - x_*\|)^2 + \frac{L}{2}\|x_m - y_m\|^2 + f(y_m) - f_* \\ &\leq \left(\frac{1}{\rho_1} - 1\right)\left(\sqrt{\frac{L}{2}}\|x_m - y_m\| + \sqrt{\frac{L}{2\mu}}\sqrt{f(y_m) - f_*}\right)^2 + \frac{L}{2}\|x_m - y_m\|^2 + f(y_m) - f_* \end{aligned}$$

We can now apply Corollary 7. From the definition of m, we have

$$2(f(y_k) - f_*) \leq \frac{1}{\sqrt{L\mu}} \|\nabla f(y_k)\|^2 \text{ for all } k \in [1, m].$$

Therefore, by denoting $\rho_2 = \left(1 + \sqrt{\frac{\mu}{L}}\right)^{-1}$, we have the following inequalities

$$\frac{L}{2}\|x_m - y_m\|^2 + f(y_m) - f_* \leq \rho_2^m (f(x_0) - f_*),$$

$$\sqrt{\frac{L}{2}}\|x_m - y_m\| \leq \rho_2^{m/2}\sqrt{f(x_0) - f_*},$$

$$\sqrt{\frac{L}{2\mu}}\sqrt{f(y_m) - f_*} \leq \sqrt{\frac{L}{2\mu}}\rho_2^{m/2}\sqrt{f(x_0) - f_*},$$

which leads to

$$\frac{L}{2}\|\frac{1}{\sqrt{\rho_1}}(x_m - x_*) - \sqrt{\rho_1}(y_m - x_*)\|^2 + f(y_m) - f_*$$
$$\leq \left(\left(\frac{1}{\rho_1} - 1\right)\left(1 + \sqrt{\frac{L}{2\mu}}\right)^2 + 1\right) \rho_2^m (f(x_0) - f_*),$$

reaching the desired result. ■

C.5. Proximal variants

A natural extension of smooth and strongly convex optimization is the case composite optimization

$$\min_{x \in \mathbb{R}^n} \{F(x) \equiv f(x) + h(x)\},$$

where $f \in \mathcal{F}_{\mu,L}$ and $h \in \mathcal{F}_{0,\infty}$ is a proper convex function with proximal operator available.

Algorithm 3 Proximal accelerated gradient method

```
**Input:** $x_0 \in \mathbb{R}^n$, $f_* \in \mathbb{R}$, $L$ smoothness constant.
$y_0 = x_0$,
**for** $k \geq 0$ **do**
    $y_{k+1} = \mathrm{prox}_{h/L}\left(x_k - \frac{1}{L}\nabla f(x_k)\right)$
    compute $\tilde{\mu}_k$ and $\beta_k = \frac{\sqrt{L} - \sqrt{\tilde{\mu}_k}}{\sqrt{L} + \sqrt{\tilde{\mu}_k}}$
    $x_{k+1} = y_{k+1} + \beta_k(y_{k+1} - y_k)$
**end for**
**Output:** $y_{k+1}$
```

We used the proximal version of AGM with constant momentum. It is of the same form as Algorithm 2 but the gradient step is combined with a proximal step. We extended our estimate $\tilde{\mu}_k$ the following way. Given $F = f + h$ where $f \in \mathcal{F}_{\mu,L}$ and h a proper convex function that is proximable, $\tilde{\mu}_k = \frac{\mathcal{D}(y_{k+1}, L)}{2(F(y_{k+1}) - F_*)}$ where $\mathcal{D}(x, L) = -2L \min_y \left[\langle \nabla f(x), y - x\rangle + \frac{L}{2}\|x - y\|^2 + h(y) - h(x)\right]$. Notice that when $h = 0$ the previous formula is exactly (Acc. Variant I). Also, when they are well defined these estimates still belong to $[\mu, L]$ (Karimi et al., 2016).

C.6. Study of standard Polyak steps

From numerical experiments, we noticed that (Variant I) was actually typically performing only slightly better than vanilla gradient descent. From a worst-case point of view, this is expected. However, our experiments (see Figure 1-3) suggest that regular Polyak steps (Polyak) actually perform much better than one could expect from its worst-case guarantees.

In this section, we provide a tentative explanation of this behavior, through experiments on a toy example. Figure 4 (top) was obtained by running the methods on a least squares problem (we used a rescaled version of the Sonar dataset, with regularity parameters $L = 1$ and $\mu = 0.01$).

Similar in spirit as in Figure 2 (left), we provide, in Figure 4, the worst-case ratio of $\|x_{k+1} - x_*\|^2/\|x_k - x_*\|^2$ (by solving (6) numerically for regular Polyak steps). One can observe that the worst case rate (using distances to optimum as the criterion) is slightly worse than that of (Variant I) (note that this rate can be improved through the use of refined Lyapunov functions).

In Figure 4, we provide the distributions of step size magnitudes observed through the optimization process on the toy example. One can notice that the distribution does not fully concentrate around the worst-case value (the value of γ that achieves the worst-case) for (Polyak). A large proportion of effective step size values are even located in regions of fast convergence. On the contrary, for (Variant I), the distribution is much more concentrated around its worst-case. Those distributions strongly suggest that worst case analyses might not be the best way to explain the good practical behaviors of such adaptive methods.

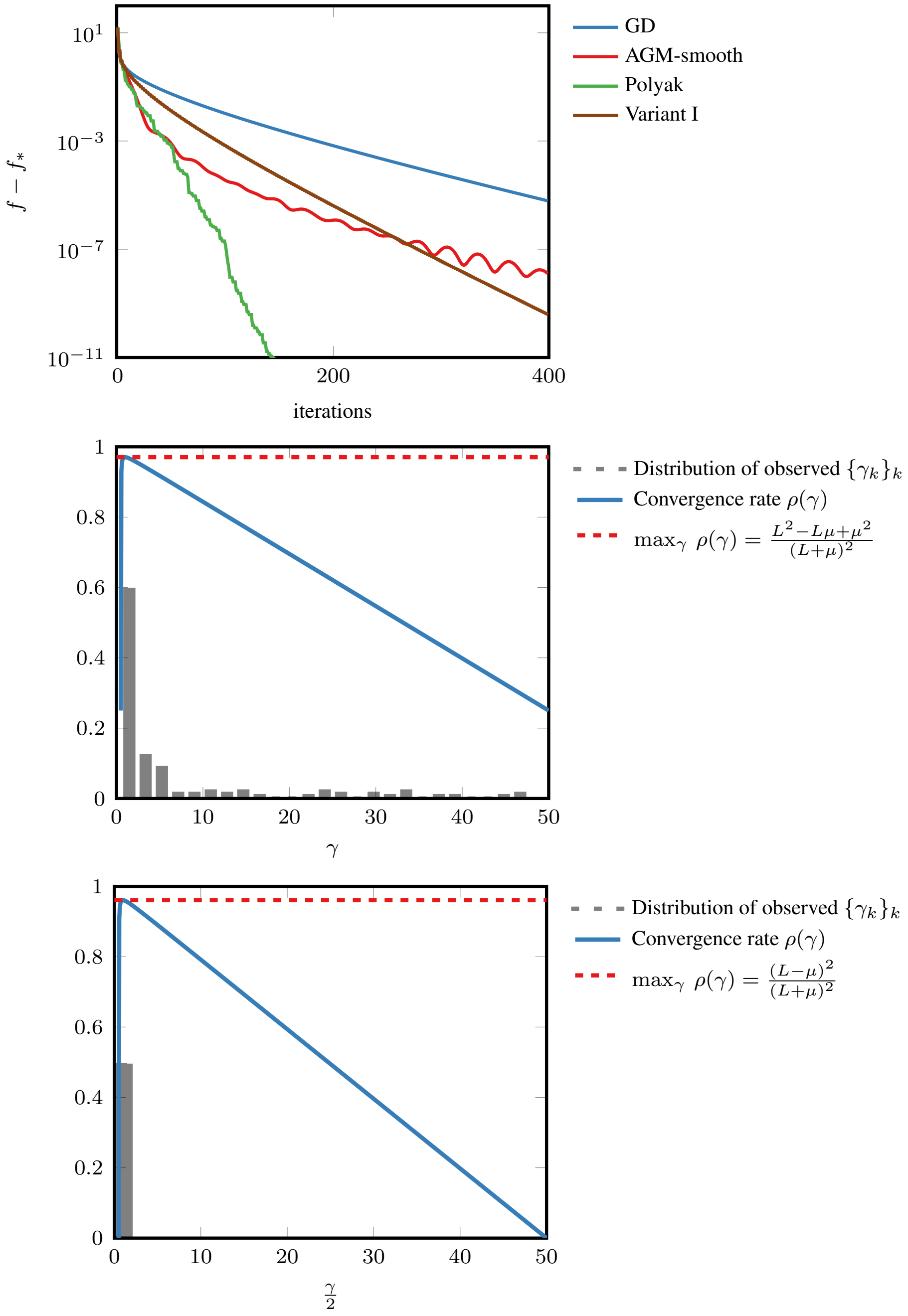

Figure 4: Top: Least squares on rescaled Sonar dataset ($L = 1$ and $\mu = 0.01$). Middle: $\rho(\gamma)$ for (Polyak) (blue)—computed numerically following the methodology of § 4 with fixed $L = 1$ and $\mu = 0.01$. Distribution of effective step size magnitudes (black) used throughout the 150 iterations of (Polyak) appearing in (top). Bottom: $\rho(\gamma)$ for (Variant I) (blue)—with $L = 1$ and $\mu = 0.01$. Distribution of effective step size magnitudes (black) used throughout the 400 iterations of (Variant I) appearing in (top).