Proceedings of Machine Learning Research vol 117:1–22, 2020 31st International Conference on Algorithmic Learning Theory

A Tight Convergence Analysis for Stochastic Gradient Descent with Delayed Updates

Yossi Arjevani YOSSI.ARJEVANI@GMAIL.COM
New York University

Ohad Shamir OHAD.SHAMIR@WEIZMANN.AC.IL
Weizmann Institute of Science

Nathan Srebro NATI@TTIC.EDU
Toyota Technological Institute at Chicago

Editors: Aryeh Kontorovich and Gergely Neu

Abstract

We establish matching upper and lower complexity bounds for gradient descent and stochastic gradient descent on quadratic functions, when the gradients are delayed and reflect iterates from τ rounds ago. First, we show that without stochastic noise, delays strongly affect the attainable optimization error: In fact, the error can be as bad as non-delayed gradient descent ran on only $1/\tau$ of the gradients. In sharp contrast, we quantify how stochastic noise makes the effect of delays negligible, improving on previous work which only showed this phenomenon asymptotically or for much smaller delays. Also, in the context of distributed optimization, the results indicate that the performance of gradient descent with delays is competitive with synchronous approaches such as mini-batching. Our results are based on a novel technique for analyzing convergence of optimization algorithms using generating functions.

Keywords: optimization, stochastic gradient descent, delayed, asynchronous, upper bounds, lower bounds

1. Introduction

Gradient-based optimization methods are widely used in machine learning and other large-scale applications, due to their simplicity and scalability. However, in their standard formulation, they are also strongly synchronous and iterative in nature: In each iteration, the update step is based on the gradient at the current iterate, and we need to wait for this computation to finish before moving to the next iterate. For example, to minimize some function F, plain stochastic gradient descent initializes at some point $\mathbf{w}_0$, and computes iterates of the form

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \eta(\nabla F(\mathbf{w}_k) + \boldsymbol{\xi}_k) \,, \tag{1}$$

where $\nabla F(\mathbf{w}_k)$ is the gradient of F at $\mathbf{w}_k$, η is the step size and $\boldsymbol{\xi}_1, \boldsymbol{\xi}_2, \ldots$ are independent zero-mean noise terms. Unfortunately, in several important applications, a direct implementation of this is too costly. For example, consider a setting where we wish to optimize a function F using a distributed platform, consisting of several machines with shared memory. We can certainly implement gradient descent, by letting one of the machines compute the gradient at each iteration, but this is clearly wasteful, since just one machine is non-idle at any given time. Thus, it is highly desirable to use methods which parallelize the computation. One approach is to employ *mini-batch gradient* methods,

which parallelize the computation of the stochastic gradient, and their analysis is relatively well understood (e.g. Dekel et al. (2012); Cotter et al. (2011); Shamir and Srebro (2014); Takác et al. (2013)). However, these methods are still generally iterative and synchronous in nature, and hence can suffer from problems such as having to wait for the slowest machine at each iteration.

A second and popular approach is to utilize *asynchronous* gradient methods. With these methods, each update step is not necessarily based just on the gradient of the current iterate, but possibly on the gradients of earlier iterates (often called *stale updates*). For example, when optimizing a function using several machines, each machine might read the current iterate from a shared parameter server, compute the gradient at that iterate, and then update the parameters, even though other machines might have performed other updates to the parameters in the meantime. Although such asynchronous methods often work well in practice, analyzing them is much trickier than synchronous methods.

In our work, we focus on arguably the simplest possible variant of these methods, where we perform plain stochastic gradient descent on a convex function F on $\mathbb{R}^d$, with a fixed delay of $\tau > 0$ in the gradient computation:

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \eta(\nabla F(\mathbf{w}_{k-\tau}) + \boldsymbol{\xi}_k)\,, \tag{2}$$

where we assume that $\mathbf{w}_0 = \mathbf{w}_1 = \ldots = \mathbf{w}_\tau$. Compared to Eq. (1), we see that the gradient is computed with respect to $\mathbf{w}_{k-\tau}$ rather than $\mathbf{w}_k$. Already in this simple formulation, the precise effect of the delay on the convergence rate is not completely clear. For example, for a given number of iterations k, how large can τ be before we might expect a significant deterioration in the accuracy? And under what conditions? Although there exist some prior results in this direction (which we survey in the related work section below), these questions have remained largely open.

In this paper, we aim at providing a tight, finite-time convergence analysis for stochastic gradient descent with delays, focusing on the simple case where F is a convex quadratic function. Although a quadratic assumption is non-trivial, it arises naturally in problems such as least squares, and is an important case study since all smooth and convex function are locally quadratic close to their minimum (hence, our results should still hold in a local sense). In future work, we hope to show that our results are also applicable more generally.

First, we consider the case of *deterministic* delayed gradient descent (DGD, defined in Eq. (2) with $\boldsymbol{\xi}_k = \mathbf{0}$). Assuming the step size η is chosen appropriately, we prove that

$$F(\mathbf{w}_k) - F(\mathbf{w}^*) \;\leq\; 5\mu\|\mathbf{w}_0 - \mathbf{w}^*\|^2 \exp\left(-\frac{\lambda(k+1)}{10\mu(\tau+1)}\right)$$

after k iterations, over the class of λ-strongly convex μ-smooth quadratic functions with a minimum at $\mathbf{w}^*$, and

$$F(\mathbf{w}_k) - F(\mathbf{w}^*) \;\leq\; \frac{17\mu\|\mathbf{w}_0 - \mathbf{w}^*\|^2(\tau+1)}{k+1}$$

over the class of μ-smooth convex quadratic functions with minimum at $\mathbf{w}^*$. In terms of iteration complexity, the number of iterations k required to achieve a fixed optimization error of at most ϵ in the strongly convex and the convex cases is therefore

$$\mathcal{O}\left(\tau \cdot \kappa \ln\left(\frac{\mu\|\mathbf{w}_0 - \mathbf{w}^*\|^2}{\epsilon}\right)\right) \quad \text{and} \quad \mathcal{O}\left(\tau \cdot \frac{\mu\|\mathbf{w}_0 - \mathbf{w}^*\|^2}{\epsilon}\right) \tag{3}$$

respectively, where $\kappa := \mu/\lambda$ is the so-called condition number[1]. When τ is a bounded constant, these bounds match the known iteration complexity of standard gradient descent without delays Nesterov (2004). However, as τ increases, both bounds deteriorate linearly with τ. Notably, in our setting of delayed gradients, this implies that DGD is no better than a trivial algorithm, which performs a single gradient step, and then waits for τ rounds till the delayed gradient is received, before performing the next step (thus, the algorithm is equivalent to non-delayed gradient descent with k/τ gradient steps, resulting in the same linear deterioration of the iteration complexity with τ).

Despite these seemingly weak guarantees, we show that they are in fact tight in terms of τ, by proving that this linear dependence on τ is unavoidable with standard gradient-based methods (including gradient descent). The dependence on the other problem parameters in our lower bounds is a bit weaker than our upper bounds, but can be matched by an *accelerated* gradient descent procedure (see Sec. 3 for more details).

In the second part of our paper, we consider the case of *stochastic* delayed gradient descent (SDGD, defined in (2)). Assuming $\boldsymbol{\xi}_k$ satisfies $\mathbb{E}[\|\boldsymbol{\xi}_k\|^2] \leq \sigma^2$ and that the step size η is appropriately tuned, we prove that

$$\mathbb{E}\left[F(\mathbf{w}_k) - F(\mathbf{w}^*)\right] \;\leq\; \tilde{\mathcal{O}}\left(\frac{\sigma^2}{\lambda k} + \mu\|\mathbf{w}_0 - \mathbf{w}^*\|^2 \exp\left(-\frac{\lambda k}{10\mu\tau}\right)\right) . \tag{4}$$

for λ-strongly convex, μ-smooth quadratic functions with minimum at $\mathbf{w}^*$, and

$$\mathbb{E}\left[F(\mathbf{w}_k) - F(\mathbf{w}^*)\right] \;\leq\; \tilde{\mathcal{O}}\left(\frac{\|\mathbf{w}_0 - \mathbf{w}^*\|\sigma}{\sqrt{k}} + \frac{\|\mathbf{w}_0 - \mathbf{w}^*\|^2\mu\tau}{k}\right) . \tag{5}$$

for μ-smooth convex quadratic functions. In terms of iteration complexity, these correspond to

$$\tilde{\mathcal{O}}\left(\frac{\sigma^2}{\lambda\epsilon} + \tau\cdot\kappa\ln\left(\frac{\mu\|\mathbf{w}_0 - \mathbf{w}^*\|^2}{\epsilon}\right)\right) \quad \text{and} \quad \tilde{\mathcal{O}}\left(\frac{\|\mathbf{w}_0 - \mathbf{w}^*\|^2\sigma^2}{\epsilon^2} + \tau\cdot\frac{\|\mathbf{w}_0 - \mathbf{w}^*\|^2\mu}{\epsilon}\right) , \tag{6}$$

in the strongly convex and convex cases respectively, where again $\kappa := \mu/\lambda$. As in the deterministic case, when τ is a bounded constant, these bounds match the known iteration complexity bounds for standard gradient descent without delays Bubeck et al. (2015); Shamir and Zhang (2013). Moreover, these bounds match the bounds for the deterministic case in Eq. (3) when $\sigma^2 = 0$ (i.e. zero noise), as they should. However, in sharp contrast to the deterministic case, the dependence on τ in Eq. (6) is quite different: The delay τ only appears in second-order terms (as $\epsilon \to 0$), and its influence becomes negligible when ϵ is small enough. The same effect can be seen in Eq. (4) and Eq. (5): Once the number of iterations k is large enough, the first term in both bounds dominates, and τ no longer plays a role. More specifically:

- In the strongly convex case, the effect of the delay becomes negligible once the target accuracy ϵ is sufficiently smaller than $\tilde{\mathcal{O}}(\sigma^2/(\mu\tau))$, or when the number of iterations k is sufficiently larger than $\tilde{\Omega}(\tau\mu/\lambda)$. In other words, assuming the condition number μ/λ is bounded, we can have the delay τ nearly as large as the total number of iterations k (up to log-factors), without significant deterioration in the convergence rate. Note that this is a mild requirement, since if $\tau \geq k$, the algorithm receives no gradients and makes no updates.

1. Following standard convention, we use here the $\mathcal{O}$-notation to hide constants, and tilde $\tilde{\mathcal{O}}$-notation to hide constants and factors polylogarithmic in the problem parameters.

- In the convex case, the effect of the delay becomes negligible once the target accuracy ϵ is sufficiently smaller than $\tilde{\mathcal{O}}(\sigma^2/(\mu\tau))$, or when the number of iterations k is sufficiently larger than $\tilde{\Omega}((\|\mathbf{w}_0 - \mathbf{w}^*\|\mu\tau/\sigma)^2)$. Compared to the strongly convex case, here the regime is the same in terms of ϵ, but the regime in terms of k is more restrictive: We need k to scale quadratically (rather than linearly) with τ. Thus, the maximal delay τ with no performance deterioration is order of $\sqrt{k}$.

Finally, it is interesting to compare our bounds to those of *mini-batch* stochastic gradient descent (SGD), which can be seen as a synchronous gradient-based method to cope with delays, especially in distributed optimization and learning problems Dekel et al. (2012); Cotter et al. (2011); Agarwal and Duchi (2011). In mini-batch SGD, each update step is performed only after accumulating and averaging a mini-batch of b stochastic gradients, all with respect to the same point:

$$\forall k \in \{0, b, 2b, \ldots\}, \quad \mathbf{w}_{k+b} = \mathbf{w}_k - \eta \cdot \frac{1}{b} \sum_{i=0}^{b-1} \left(\nabla F(\mathbf{w}_k) + \xi_{k+i}\right) \ ,$$

Although the algorithm makes an update only every b stochastic gradient computations, the averaging reduces the stochastic noise, and helps speed up convergence. Moreover, this can be seen as a particular type of algorithm with delayed updates (with the delay correspond to b), as we use $\nabla F(\mathbf{w}_k)$ to compute iterate $\mathbf{w}_{k+b}$. The important difference is that it is an inherently synchronous method, that waits for all b stochastic gradients to be computed before performing an update step. Remarkably, the bounds we proved above for delayed SGD are essentially identical to those known for mini-batch SGD, with the delay τ replaced by the mini-batch size b (at least in the convex case where mini-batch SGD has been more thoroughly analyzed). This indicates that an asynchronous method like delayed SGD can potentially match the performance of synchronous methods like mini-batch SGD, even without requiring synchronization – an important practical advantage.

Analyzing gradient descent with delays is notoriously tricky, due to the dependence of the updates on iterates produced many iterations ago. The technique we introduce for deriving our upper bounds is primarily based on *generating functions*, and might be useful for studying other optimization algorithms. We discuss this approach more thoroughly in Section 2. The rest of the paper is devoted mostly to presenting the formal theorems and an explanation of how they are derived (with technical details relegated to the supplementary material).

Related Work

There is a huge literature on asynchronous versions of gradient-based methods (see for example the seminal book Bertsekas and Tsitsiklis (1989)), including treating the effect of delay. However, most of these do not consider the setting we study here. For example, there has been much recent interest in asynchronous algorithms, in a model where there is a delay in updating individual *coordinates* in a shared parameter vector (e.g., the Hogwild! algorithm of Recht et al. (2011), or more recently Mania et al. (2015); Leblond et al. (2018)). Of course, this is a different model than ours, where the updates use a full gradient vector. Other works (such as Sirb and Ye (2016)) focus on a setting where different agents in a network can perform local communication, which is again a different model than ours. Yet other works focus on sharp but asymptotic results, and do not provide guarantees after a fixed number k of iterations (e.g., Chaturapruek et al. (2015)).

Moving closer to our setting, Nedić et al. (2001) showed convergence for delayed gradient descent, with the result implying an $\sqrt{\tau/k}$ convergence rate for convex functions. A similar bound

on average regret has been shown in an adversarial online learning setting, for general convex functions, and this bound is known to be optimal Joulani et al. (2013). These results differ from our setting, in that they consider possibly non-smooth functions, in which the dependence on k is no better than $1/\sqrt{k}$ even without delays and no noise, and where the delay τ always plays a significant role. In contrast, we focus here on smooth functions, where rates better than $1/\sqrt{k}$ are possible, and where the effect of τ is more subtle. In Feyzmahdavian et al. (2014), the authors study a setting very similar to ours in the deterministic case, and manage to prove a linear convergence rate, but for a less standard algorithm, different than the one we study here (with iterates of the form $\mathbf{w}_{t+1} = \mathbf{w}_{t-\tau} - \nabla F(\mathbf{w}_{t-\tau})$). Perhaps the works closest to ours are Agarwal and Duchi (2011); Feyzmahdavian et al. (2016), which study stochastic gradient descent with delayed gradients. Moreover, they consider a setting more general than ours, where the delay at each iteration is any integer up to τ (rather than fixed τ), and the functions are not necessarily quadratic. On the flip side, their bounds are significantly weaker. For example, for smooth convex functions and an appropriate step size, (Agarwal and Duchi, 2011, Corollary 1) show a bound of

$$\mathcal{O}\left(\frac{\sigma}{\sqrt{k}} + \frac{\tau^2+1}{\sigma^2 k}\right).$$

in terms of k, τ, σ. Note that this bound is vacuous in the deterministic or near-deterministic case (where $\sigma^2 \approx 0$), and is weaker than our bounds. With a different choice of the step size, it is possible to get a non-vacuous bound even if $\sigma^2 \to 0$, but the dependence on τ becomes even stronger. Feyzmahdavian et al. (2016) improve the bound to

$$\mathcal{O}\left(\frac{\sigma}{\sqrt{k}} + \frac{\tau^2+1}{k}\right) \quad \text{and} \quad \mathcal{O}\left(\frac{\sigma^2}{k} + \frac{\tau^4+1}{k^2}\right).$$

in the convex and strongly convex case respectively. Even if $\sigma^2 = 0$, the iteration complexity is $\mathcal{O}(\tau^2/\epsilon)$ and $\mathcal{O}(\tau^2/\sqrt{\epsilon})$, and implies a quadratic dependence on τ (whereas in our bounds the scaling is linear). When σ^2 is positive, the effect of delay on the bound is negligible only up to $\tau = \mathcal{O}(\sqrt[4]{(k)})$ (in contrast to $\tilde{\mathcal{O}}(\sqrt{k})$ or even $\tilde{\mathcal{O}}(k)$ in our bounds). We note that there are several other works which study a similar setting (such as Sra et al. (2015)), but do not result in bounds which improve on the above. Finally, we note that Langford et al. (2009) attempt to show that for stochastic gradient descent with delayed updates, the dependence on the delay τ is negligible after sufficiently many iterations. Unfortunately, as pointed out in Agarwal and Duchi (2011), the analysis contains a bug which make the results invalid.

2. Framework and the Generating Functions Approach

Throughout, we will assume that F is a convex quadratic function specified by

$$F(\mathbf{w}) := \frac{1}{2}\mathbf{w}^\top A \mathbf{w} + \mathbf{l}^\top \mathbf{w} + c, \tag{7}$$

where $A \in \mathbb{R}^{d\times d}$ is a positive semi-definite matrix whose eigenvalues $a_1, \ldots, a_d$ are in $[0, \mu]$ (where μ is the smoothness parameter), $\mathbf{l} \in \mathbb{R}^d$ and $c \in \mathbb{R}$. To make the optimization problem meaningful, we further assume that F is bounded from below, which implies that it has some minimizer $\mathbf{w}^* \in \mathbb{R}^d$ at

which the gradient vanishes (for completeness, we provide a proof in Lemma 7 in the supplementary material). Letting $\mathbf{e}_k = \mathbf{w}_k - \mathbf{w}^*$, it is easily verified that

$$F(\mathbf{w}_k) - F(\mathbf{w}^*) = \frac{1}{2}\left\|\sqrt{A}(\mathbf{w} - \mathbf{w}^*)\right\|^2 = \frac{1}{2}\left\|\sqrt{A}\mathbf{e}_k\right\|^2, \tag{8}$$

so our goal will be to analyze the dynamics of $\mathbf{e}_k$.

To explain our technique, consider the iterates of DGD on the function F, which can be written as $\mathbf{w}_{k+1} = \mathbf{w}_k - \eta\nabla F(\mathbf{w}_{k-\tau}) = \mathbf{w}_k - \eta(A\mathbf{w}_{k-\tau} + \mathbf{l})$. Since $\nabla F(\mathbf{w}^*) = 0$, we have $\mathbf{w}^* = \mathbf{w}^* - \eta(A\mathbf{w}^* + \mathbf{l})$, by which it follows that the error term $\mathbf{e}_k = \mathbf{w}_k - \mathbf{w}^*$, satisfies the recursion $\mathbf{e}_{k+1} = \mathbf{e}_k - \eta A\mathbf{e}_{k-\tau}$, and (by definition of the algorithm) $\mathbf{e}_0 = \mathbf{e}_1 = \ldots = \mathbf{e}_\tau$. By some simple arguments, our analysis then boils down to bounding the elements of the scalar-valued version of this sequence, namely

$$\begin{aligned} b_0 = \cdots = b_\tau &\in \mathbb{R}, \\ b_{k+1} = b_k - \alpha b_{k-\tau},\ k &\geq \tau, \end{aligned} \tag{9}$$

for some integer $\tau \geq 0$ and non-negative real number $\alpha \geq 0$. To analyze this sequence, we rely on tools from the area of generating functions, which have proven very effective in studying growth rates of sequences in many areas of mathematics. We now turn to briefly describe these functions and our approach (for general surveys on generating functions, see e.g., Wilf (2005); Flajolet and Sedgewick (2009); Stanley (1986)).

Generally speaking, generating functions are formal power series associated with infinite sequences of numbers. Concretely, given a sequence (b_k) of numbers in a ring R, we define the corresponding generating function as a formal power series in z, defined as $f(z) = \sum_{k=0}^{\infty} b_k z^k$. The set of all formal power series in z over R is denoted by $R[[z]]$. Moreover, given two power series defined by sequences (a_k) and (c_k), we can define their addition as the power series corresponding to $(a_k + c_k)$, and their multiplication as the coefficients of the Cauchy product of the power series, namely $(\sum_k a_k z^k)(\sum_k c_k z^k) = \sum_k (\sum_{l=0}^{k} a_l c_{k-l}) z^k$. In particular, over the reals, $\mathbb{R}[[z]]$ endowed with addition and multiplication is a commutative ring, and the set of matrices with elements in $\mathbb{R}[[z]]$ (with the standard addition and multiplication operations) forms a matrix algebra, denoted by $\mathcal{M}(\mathbb{R}[[z]])$. We will often use the fact that any matrix, whose entries are power series with scalar coefficients, can also be written as a power series with matrix-valued coefficients: More formally, $\mathcal{M}(\mathbb{R}[[z]])$ is naturally identified with the ring of formal power series with real matrix coefficients $\mathcal{M}(\mathbb{R})[[z]]$. To extract the coefficients of a given $M(z) \in \mathcal{M}(R[[z]])$, we shall use the conventional bracket notation $[z^k]M(z)$, defined to be a matrix whose entries are the k'th coefficients of the respective formal power series.

Returning to Eq. (9), we write (b_k) as a formal power series denoted by $f(z)$, and proceed as follows,

$$\begin{aligned} f(z) &= \sum_{k=0}^{\tau} b_k z^k + \sum_{k=\tau+1}^{\infty} (b_{k-1} - \alpha b_{k-\tau-1}) z^k = \sum_{k=0}^{\tau} b_k z^k + \sum_{k=\tau+1}^{\infty} b_{k-1} z^k - \alpha \sum_{k=\tau+1}^{\infty} b_{k-\tau-1} z^k \\ &= \sum_{k=0}^{\tau} b_k z^k + z\left(f(z) - \sum_{k=0}^{\tau-1} b_k z^k\right) - \alpha z^{\tau+1} f(z) = b_0 + (z - \alpha z^{\tau+1}) f(z)\,. \end{aligned} \tag{10}$$

Denoting

$$\pi_\alpha(z) \;:=\; 1 - z + \alpha z^{\tau+1}$$

and rearranging terms gives

$$f(z) = \frac{b_0}{\pi_\alpha(z)} \quad \Longrightarrow \quad b_k = [z^k]f(z) = [z^k]\frac{b_0}{\pi_\alpha(z)} \tag{11}$$

(by a well-known fact, $\pi_\alpha(z)$ is invertible in $\mathbb{R}[[z]]$, as its constant term 1 is trivially invertible in $\mathbb{R}$ – see surveys mentioned above). We now see that the problem of bounding the coefficients (b_k) is reduced to that of estimating the coefficients of the rational function $1/\pi_\alpha(z)$, written as a power series. Note that for the analogous problem where the elements of the sequence are vectors $(\mathbf{b}_k)_{k=0}^\infty$ and the factor α is replaced by αA for some square matrix A, the same derivation as above yields $\sum_{k=0}^\infty \mathbf{b}_k z^k = (I - z + \alpha A z)^{-1}\mathbf{b}_0$ (likewise, $I - z + Az$ is invertible in $\mathcal{M}(\mathbb{R})[[z]]$ as its constant term I is invertible in $\mathcal{M}(\mathbb{R})$).

To estimate the coefficients of $1/\pi_\alpha(z)$, we form the corresponding partial fraction decomposition. First, we note that as a polynomial of degree $\tau+1$, $\pi_\alpha(z)$ has $\tau+1$ roots $\zeta_1, \ldots, \zeta_{\tau+1}$ (possibly complex-valued, and all non-zero since $\pi_\alpha(0) = 1$ for any $\alpha \in \mathbb{R}$). Assuming α is chosen so that all the roots are distinct (equivalently, $\pi'_\alpha(\zeta_i) \neq 0$, for $i \in [\tau+1]$), we have by a standard derivation

$$\frac{1}{\pi_\alpha(z)} = \sum_{i=1}^{\tau+1} \frac{1}{\pi'_\alpha(\zeta_i)(z-\zeta_i)} = \sum_{i=1}^{\tau+1} \frac{-1}{\pi'_\alpha(\zeta_i)\zeta_i} \cdot \frac{1}{1-\frac{z}{\zeta_i}} = \sum_{i=1}^{\tau+1} \frac{-1}{\pi'_\alpha(\zeta_i)\zeta_i} \sum_{k=0}^{\infty} \left(\frac{z}{\zeta_i}\right)^k.$$

Thus,

$$[z^k]\left(\frac{1}{\pi_\alpha(z)}\right) = \sum_{i=1}^{\tau+1} \frac{-1}{\pi'_\alpha(\zeta_i)\zeta_i^{k+1}}\,. \tag{12}$$

To bound the magnitude of $1/\zeta_i$ and $\pi'_\alpha(\zeta_i)$, we invoke the following lemma, whose proof (in the supplementary material) relies on standard tools from complex analysis:

Lemma 1 *Let $\alpha \in (0, 1/20(\tau+1)]$, and assume $|\zeta_1| \leq |\zeta_2| \leq \cdots \leq |\zeta_{\tau+1}|$, then*

1. *ζ_1 is a real scalar satisfying $1/\zeta_1 \leq 1-\alpha$, and for $i > 1$, $|1/\zeta_i| \leq 1 - \frac{3}{2(\tau+1)}$.*
2. *$|\pi'_\alpha(\zeta_i)| > 1/2$, for any $i \in [\tau+1]$.*

With this lemma at hand, we have

$$\left|[z^k]\left(\frac{1}{\pi_\alpha(z)}\right)\right| \leq 2(1-\alpha)^{k+1} + 2\tau\left(1 - \frac{3}{2(\tau+1)}\right)^{k+1} \leq 2(1-\alpha)^{k+1}\left(1 + \tau\exp\left(-\frac{k+1}{\tau+1}\right)\right)\,,$$

where the last inequality is due to Lemma 9 (provided in the supplementary material). Moreover, one can use elementary arguments to show that $|[z^k]1/\pi_\alpha(z)| \leq 1$ for any $k \geq 0$, as long as $\alpha \in [0, 1/\tau]$ (see Lemma 6 in the supplementary material). Overall, for any $\tau \geq 0$, we have

$$\begin{cases} \left|[z^k]\left(\frac{1}{\pi_\alpha(z)}\right)\right| \leq 1 & 0 \leq k \leq (\tau+1)\ln(2(\tau+1)) - 1, \\ \left|[z^k]\left(\frac{1}{\pi_\alpha(z)}\right)\right| \leq 3(1-\alpha)^{k+1} & k \geq (\tau+1)\ln(2(\tau+1)), \end{cases} \tag{13}$$

which, using Eq. (11), gives the desired bounds on the elements (b_k) defined in Eq. (9).

We would like to remark at this point that a common technique of studying the growth rate of sequences in the context of optimization, such as (b_k), is to express the dynamics in a higher-dimensional space, over which the task boils down to bounding the norm of powers of a given matrix (e.g., Flammarion and Bach (2015); O'Neill and Wright (2017); Arjevani et al. (2016)). In more detail, first Eq. (9) is re-expressed over $\mathbb{R}^{\tau+1}$ as follows

$$\mathbf{b}_0 = \begin{pmatrix} b_0 \\ \vdots \\ b_\tau \end{pmatrix} \in \mathbb{R}^{\tau+1},\ \mathbf{b}_k = M\mathbf{b}_{k-1},\ k \geq 1, \quad \text{where } M = \begin{pmatrix} 0 & 1 & & & \\ & 0 & 1 & & \\ & & \ddots & \ddots & \\ -\alpha, & 0 & \dots & 0 & 1 \end{pmatrix}.$$

One then proceeds by forming a convenient way of iterating M which typically involves bounding the corresponding eigenvalues and using a suitable matrix decomposition. Here, M forms a companion matrix. In particular, the spectral decomposition theorem for normal matrices does not apply. Using Jordan normal form instead, results in Vandermonde transition matrices, whose condition number behaves poorly in general and forms a major obstacle for the convergence analysis (e.g., Pan (2016)). Compared to this technique, the generating functions-based approach offers a more accessible way of controlling the non-asymptotic growth rate of (b_k) by relying on other types of quantities which seem to be more tangible for the problem at hand (e.g., roots of the derivative of the characteristic polynomial, as demonstrated above). To the best of our knowledge, this is the first time generating functions are used in the context of optimization for deriving converge rates.

3. Deterministic Delayed Gradient Descent

We start by analyzing the convergence of DGD for λ-strongly convex and μ-smooth quadratic functions, where the eigenvalues of A are assumed to lie in $[\lambda, \mu]$ for some $\mu \geq \lambda > 0$.

Following the same line of the derivation as in Eq. (10), we obtain $\mathbf{e}(z) = (I - Iz + \eta A z^{\tau+1})^{-1}\mathbf{e}_0$. Letting $[d] := \{1, 2, \dots, d\}$, it follows that for any $k \geq (\tau+1)\ln(2(\tau+1))$,

$$\begin{aligned} \|\mathbf{e}_k\| = \|[z^k]\left((I - Iz + \eta A z^{\tau+1})^{-1}\mathbf{e}_0\right)\| &\overset{(a)}{=} \|[z^k]\left((I - Iz + \eta A z^{\tau+1})^{-1}\right)\mathbf{e}_0\| \\ &\overset{(b)}{\leq} \max_{i\in[d]} \left|[z^k]\frac{1}{\pi_{\eta a_i}(z)}\right| \|\mathbf{e}_0\| \overset{(c)}{\leq} 3\max_{i\in[d]}(1-\eta a_i)^{k+1}\|\mathbf{e}_0\| \overset{(d)}{\leq} 3(1-\eta\lambda)^{k+1}\|\mathbf{e}_0\|, \end{aligned} \tag{14}$$

where (a) follows by the linearity of the bracket operation $[z^k]$, (b) by eigendecomposition of A (that reveals that the spectral norm of a matrix polynomial equals the absolute value of the same polynomial in one of its eigenvalues), (c) by Ineq. (13) for $\eta\mu \in (0, 1/(20(\tau+1))]$, and (d) by the fact that $a_i \geq \lambda$ for all i. By Eq. (8) and the fact that all the eigenvalues of A are at most μ, we have the following bound:

Theorem 2 *For any delay $\tau \geq 0$ and $k \geq (\tau+1)\ln(2(\tau+1))$, running DGD with step size $\eta \in (0, 1/(20\mu(\tau+1))]$ on a μ-smooth, λ-strongly convex quadratic function yields*

$$F(\mathbf{w}_k) - F(\mathbf{w}^*) \leq 5\mu\left(1-\eta\lambda\right)^{2(k+1)}\|\mathbf{w}_0 - \mathbf{w}^*\|^2.$$

In particular, setting $\eta = \Omega(1/\mu\tau)$, we get that

$$F(\mathbf{w}_k) - F(\mathbf{w}^*) \ \leq\ 5\mu\|\mathbf{w}_0 - \mathbf{w}^*\|^2 \exp\left(-\Omega\left(\frac{k\lambda}{\mu\tau}\right)\right).$$

Note that the assumption that $k \geq (\tau+1)\ln(2(\tau+1))$ is very mild, since if $k \leq \tau$ then the algorithm trivially makes no updates after k rounds.

We now turn to analyze the case of μ-smooth convex quadratic functions, where the eigenvalues of the matrix A are assumed to lie in $[0, \mu]$. Following the same derivation as in Ineq. (14) and using Ineq. (8), we have for any $k \geq (\tau+1)\ln(2(\tau+1))$ and $\eta \in (0, 1/(20\mu(\tau+1))]$,

$$\begin{aligned} F(\mathbf{w}_k) - F(\mathbf{w}^*) &= \frac{1}{2}\|\sqrt{A}\mathbf{e}_k\|^2 = \frac{1}{2}\|\sqrt{A}[z^k]\left((I - Iz + \eta A z^{\tau+1})^{-1}\right)\mathbf{e}_0\|^2 \\ &\overset{(a)}{\leq} \frac{1}{2}\left(3\max_{i\in[d]}\sqrt{a_i}(1-\eta a_i)^{k+1}\right)^2\|\mathbf{e}_0\|^2 \overset{(b)}{\leq} \frac{9}{4e\eta(k+1)}\|\mathbf{e}_0\|^2\,, \end{aligned} \tag{15}$$

where $e = 2.718...$ is Euler's number, (a) is by the fact that the spectral norm of a matrix polynomial equals the absolute value of the same polynomial in one of its eigenvalues, and (b) is by the fact that $\sqrt{a_i}(1-\eta a_i)^{k+1} \leq 1/\sqrt{2e\eta(k+1)}$ for any $i \in [d]$ (see Lemma 11 in the supplementary material). We have thus arrived at the following bound for the convex case:

Theorem 3 *For any delay $\tau \geq 0$ and $k \geq (\tau+1)\ln(2(\tau+1))$, running DGD with step size $\eta \in (0, 1/(20\mu(\tau+1))]$ on a μ-smooth convex quadratic function yields*

$$F(\mathbf{w}_k) - F(\mathbf{w}^*) \leq \frac{9}{4e\eta(k+1)}\|\mathbf{w}_0 - \mathbf{w}^*\|^2\,.$$

In particular, if we set $\eta = \Omega(1/\mu\tau)$, we get that

$$F(\mathbf{w}_k) - F(\mathbf{w}^*) \leq \mathcal{O}\left(\frac{\mu\tau\|\mathbf{w}_0 - \mathbf{w}^*\|^2}{k}\right)\,.$$

As discussed in the introduction, the theorems above imply that a delay of τ increases the iteration complexity by a factor of τ. We now turn to present lower bounds which imply that this linear dependence on τ is unavoidable, for a large family of gradient-based algorithms (of which gradient descent is a special case). Specifically, we will consider any iterative algorithm producing iterates $\mathbf{w}_0, \mathbf{w}_1, \ldots$ which satisfies the following:

$$\mathbf{w}_0 = \ldots = \mathbf{w}_\tau = \mathbf{0} \quad \text{and} \quad \forall k \geq t,\ \mathbf{w}_{k+1} \in \text{span}\{\nabla F(\mathbf{w}_0), \nabla F(\mathbf{w}_1), \ldots, \nabla F(\mathbf{w}_{k-\tau})\}\,. \tag{16}$$

This is a standard assumption in proving optimization lower bounds (see Nesterov (2004)), and is satisfied by most standard gradient-based methods, and in particular our DGD algorithm. We also note that this algorithmic assumption can be relaxed at the cost of a more involved proof, similar to Nemirovsky and Yudin (1983); Woodworth and Srebro (2016) in the non-delayed case.

Theorem 4 *Consider any algorithm satisfying Eq. (16). Then the following holds for any $k \geq \tau+1$ and sufficiently large dimensionality d:*

- *There exists a μ-smooth, λ-strongly convex function F over $\mathbb{R}^d$, such that*

$$F(\mathbf{w}_k) - F(\mathbf{w}^*) \;\geq\; \frac{\lambda}{4}\exp\left(-\frac{5k}{\left(\sqrt{\mu/\lambda}-1\right)(\tau+1)}\right)\|\mathbf{w}_0 - \mathbf{w}^*\|^2\,.$$

- *There exists a μ-smooth, convex quadratic function F over $\mathbb{R}^d$, such that*

$$F(\mathbf{w}_k) - F(\mathbf{w}^*) \;\geq\; \frac{\mu(\tau+1)^2\|\mathbf{w}_0 - \mathbf{w}^*\|^2}{45k^2} \,.$$

The proof of the theorem is very similar to standard optimization lower bounds for gradient-based methods without delays (e.g., Nesterov (2004); Lan and Zhou (2018)), and is presented in the supplementary material. In fact, our main contribution is to recognize that the proof technique easily extends to incorporate delays.

In terms of iteration complexity, these bounds correspond to $\Omega\left(\tau \cdot \sqrt{\mu/\lambda} \cdot \ln\left(\lambda\|\mathbf{w}_0 - \mathbf{w}^*\|^2/\epsilon\right)\right)$ in the strongly convex case, and $\Omega\left(\tau \cdot \sqrt{\mu\|\mathbf{w}_0 - \mathbf{w}^*\|^2/\epsilon^2}\right)$ in the convex case, which show that the linear dependence on τ is inevitable. The dependence on the other problem parameters is somewhat better than in our upper bounds, but this is not just an artifact of the analysis: In our delayed setting, the lower bounds can be matched by running *accelerated* gradient descent (AGD) Nesterov (2004), where each time we perform an accelerated gradient descent step, and then stay idle for τ iterations till we get the gradient of the current point. Overall, we perform k/τ accelerated gradient steps, and can apply the standard analysis of AGD to get an iteration complexity which is τ times the iteration complexity of AGD without delays. These match the lower bounds above up to constants. We believe it is possible to prove a similar upper bound for AGD performing an update with a delayed gradient at every iteration (like our DGD procedure), but the analysis is more challenging than for plain gradient descent, and we leave it to future work.

4. Stochastic Delayed Gradient Descent

In this section, we study the case of noisy gradient updates (namely, SDGD), in which the influence of the delay is quite different than in the noiseless case. Instantiating SDGD for quadratic $F(\mathbf{w})$ (defined in (7)) results in the following update rule

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \eta\nabla F(\mathbf{w}_{k-\tau} + \epsilon_k) = \mathbf{w}_k - \eta(A\mathbf{w}_{k-\tau} + \mathbf{l} + \boldsymbol{\xi}_k) \,, \tag{17}$$

where $\boldsymbol{\xi}_k,\ k \geq 0$ are independent zero-mean noise terms satisfying $\mathbb{E}[\|\boldsymbol{\xi}_t\|^2] \leq \sigma^2$. As before, in terms of the error term $\mathbf{e}_k = \mathbf{w}_k - \mathbf{w}^*$, Eq. (17) reads as $\mathbf{e}_{k+1} = \mathbf{e}_k - \eta A\mathbf{e}_{k-\tau} - \eta\boldsymbol{\xi}_k$. Given a realization of $(\boldsymbol{\xi}_k)$, we denote its associated formal power series by $g(z) \coloneqq \sum_{k=\tau}^{\infty} \boldsymbol{\xi}_k z^k$. As before, we get that the formal power series of the error terms $(\mathbf{e}_k)$ satisfies $\mathbf{e}(z) = (I - Iz + \eta A z^{\tau+1})^{-1}(\mathbf{e}_0 - \eta g(z))$. We can now bound the error terms by extracting the corresponding coefficients of $\mathbf{e}(z)$. Letting $D \coloneqq (I - Iz + \eta A z^{\tau+1})^{-1}$, we have for any $k \geq (\tau+1)\ln(2(\tau+1))$,

$$\begin{aligned}
2 \cdot \mathbb{E}[F(\mathbf{w}_k) - F(\mathbf{w}^*)] = \mathbb{E}\left[\left\|\sqrt{A}\mathbf{e}_k\right\|^2\right] &= \mathbb{E}\left[\left\|\sqrt{A}\,[z^k]\,(D(\mathbf{e}_0 - \eta g(z)))\right\|^2\right] \\
&\overset{(a)}{=} \left\|\sqrt{A}\,[z^k]D\mathbf{e}_0\right\|^2 + \eta^2\mathbb{E}\left[\left\|\sqrt{A}\,[z^k]\,(Dg(z))\right\|^2\right] \\
&\overset{(b)}{=} \|\sqrt{A}\,[z^k]D\mathbf{e}_0\|^2 + \eta^2\mathbb{E}\left[\|\sqrt{A}\sum_{i=0}^{k}\left([z^i]D\right)\boldsymbol{\xi}_{k-i}\|^2\right] \\
&\overset{(c)}{\leq} \|\sqrt{A}\,[z^k]D\|^2\|\mathbf{e}_0\|^2 + \eta^2\sigma^2\sum_{i=0}^{k}\left\|\sqrt{A}\,[z^i]D\right\|^2 \,,
\end{aligned} \tag{18}$$

where (a) follows by the linearity of the bracket operation $[z^k]$ and the assumption that $\mathbb{E}[\boldsymbol{\xi}_k] = 0$ for all k (hence $\mathbb{E}[g(z)] = 0$), (b) follows by the Cauchy product for formal power series, and (c) by the hypothesis that $\boldsymbol{\xi}_k$ are independent and satisfy $\mathbb{E}[\|\boldsymbol{\xi}_k\|^2] \leq \sigma^2$ for all k. We then upper bound both terms, building on Ineq. (13) (see the supplementary material for a full derivation), resulting in the following theorem:

Theorem 5 *Assuming the step η satisfies $\eta \in (0, \frac{1}{20\mu(\tau+1)}]$, and $k \geq (\tau+1)\ln(2(\tau+1))$, the following holds for SDGD:*

- *For λ-strongly convex, μ-smooth quadratic convex functions, $\mathbb{E}[F(\mathbf{w}_k) - F(\mathbf{w}^*)]$ is at most*

$$5\mu \exp(-2\eta\lambda(k+1))\|\mathbf{w}_0 - \mathbf{w}^*\|^2 \; + \; \frac{\eta^2\sigma^2}{2}\left(\mu(\tau+1)\ln(2(\tau+1)) + \frac{1 + e + \ln(\frac{1}{\eta\lambda})}{e\eta}\right) .$$

  In particular, by tuning η appropriately,

$$\mathbb{E}\left(F(\mathbf{w}_k) - F(\mathbf{w}^*)\right) \; \leq \; \tilde{\mathcal{O}}\left(\frac{\sigma^2}{\lambda k} + \mu\|\mathbf{e}_0\|^2 \exp\left(-\frac{\lambda k}{10\mu\tau}\right)\right) .$$

- *For μ-smooth quadratic convex functions, $\mathbb{E}[F(\mathbf{w}_k) - F(\mathbf{w}^*)]$ is at most*

$$\frac{9\|\mathbf{w}_0 - \mathbf{w}^*\|^2}{4e\eta(k+1)} \; + \; \eta^2\sigma^2\left(\mu(\tau+1)\ln(2(\tau+1)) + \frac{9}{2e\eta}(1 + \ln(k+1))\right) .$$

  In particular, by tuning η appropriately,

$$\mathbb{E}\left(F(\mathbf{w}_k) - F(\mathbf{w}^*)\right) \; \leq \; \tilde{\mathcal{O}}\left(\frac{\|\mathbf{w}_0 - \mathbf{w}^*\|\sigma}{\sqrt{k}} + \frac{\|\mathbf{w}_0 - \mathbf{w}^*\|^2\mu\tau}{k}\right) .$$

As discussed in the introduction in detail, the theorem implies that the effect of τ is negligible once k is sufficiently large.

Acknowledgments

This research is partially supported by an NSF/BSF grant no. 2016741.

References

Alekh Agarwal and John C Duchi. Distributed delayed stochastic optimization. In *Advances in Neural Information Processing Systems*, pages 873–881, 2011.

Yossi Arjevani, Shai Shalev-Shwartz, and Ohad Shamir. On lower and upper bounds in smooth and strongly convex optimization. *Journal of Machine Learning Research*, 17(126):1–51, 2016.

Dimitri P Bertsekas and John N Tsitsiklis. *Parallel and distributed computation: numerical methods*, volume 23. Prentice hall Englewood Cliffs, NJ, 1989.

Sébastien Bubeck et al. Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, 2015.

Sorathan Chaturapruek, John C Duchi, and Christopher Ré. Asynchronous stochastic convex optimization: the noise is in the noise and sgd don't care. In *Advances in Neural Information Processing Systems*, pages 1531–1539, 2015.

Andrew Cotter, Ohad Shamir, Nati Srebro, and Karthik Sridharan. Better mini-batch algorithms via accelerated gradient methods. In *Advances in neural information processing systems*, pages 1647–1655, 2011.

Ofer Dekel, Ran Gilad-Bachrach, Ohad Shamir, and Lin Xiao. Optimal distributed online prediction using mini-batches. *Journal of Machine Learning Research*, 13(Jan):165–202, 2012.

Hamid Reza Feyzmahdavian, Arda Aytekin, and Mikael Johansson. A delayed proximal gradient method with linear convergence rate. In *Machine Learning for Signal Processing (MLSP), 2014 IEEE International Workshop on*, pages 1–6. IEEE, 2014.

Hamid Reza Feyzmahdavian, Arda Aytekin, and Mikael Johansson. An asynchronous mini-batch algorithm for regularized stochastic optimization. *IEEE Transactions on Automatic Control*, 61 (12):3740–3754, 2016.

Philippe Flajolet and Robert Sedgewick. *Analytic combinatorics*. cambridge University press, 2009.

Nicolas Flammarion and Francis Bach. From averaging to acceleration, there is only a step-size. In *Conference on Learning Theory*, pages 658–695, 2015.

Pooria Joulani, Andras Gyorgy, and Csaba Szepesvári. Online learning under delayed feedback. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, pages 1453–1461, 2013.

Guanghui Lan and Yi Zhou. An optimal randomized incremental gradient method. *Mathematical programming*, 171(1-2):167–215, 2018.

John Langford, Alex J Smola, and Martin Zinkevich. Slow learners are fast. *Advances in Neural Information Processing Systems*, 22:2331–2339, 2009.

Rémi Leblond, Fabian Pedregosa, and Simon Lacoste-Julien. Improved asynchronous parallel optimization analysis for stochastic incremental methods. *The Journal of Machine Learning Research*, 19(1):3140–3207, 2018.

Horia Mania, Xinghao Pan, Dimitris Papailiopoulos, Benjamin Recht, Kannan Ramchandran, and Michael I Jordan. Perturbed iterate analysis for asynchronous stochastic optimization. *arXiv preprint arXiv:1507.06970*, 2015.

A Nedić, Dimitri P Bertsekas, and Vivek S Borkar. Distributed asynchronous incremental subgradient methods. *Studies in Computational Mathematics*, 8(C):381–407, 2001.

AS Nemirovsky and DB Yudin. Problem complexity and method efficiency in optimization. 1983. *Willey-Interscience, New York*, 1983.

Yurii Nesterov. *Introductory lectures on convex optimization*, volume 87. Springer Science & Business Media, 2004.

Michael O'Neill and Stephen J Wright. Behavior of accelerated gradient methods near critical points of nonconvex functions. *arXiv preprint arXiv:1706.07993*, 2017.

Victor Y Pan. How bad are vandermonde matrices? *SIAM Journal on Matrix Analysis and Applications*, 37(2):676–694, 2016.

Benjamin Recht, Christopher Re, Stephen Wright, and Feng Niu. Hogwild: A lock-free approach to parallelizing stochastic gradient descent. In *Advances in neural information processing systems*, pages 693–701, 2011.

Ohad Shamir and Nathan Srebro. Distributed stochastic optimization and learning. In *Communication, Control, and Computing (Allerton), 2014 52nd Annual Allerton Conference on*, pages 850–857. IEEE, 2014.

Ohad Shamir and Tong Zhang. Stochastic gradient descent for non-smooth optimization: Convergence results and optimal averaging schemes. In *International Conference on Machine Learning*, pages 71–79, 2013.

Benjamin Sirb and Xiaojing Ye. Decentralized consensus algorithm with delayed and stochastic gradients. *arXiv preprint arXiv:1604.05649*, 2016.

Suvrit Sra, Adams Wei Yu, Mu Li, and Alexander J Smola. Adadelay: Delay adaptive distributed stochastic convex optimization. *arXiv preprint arXiv:1508.05003*, 2015.

Richard P Stanley. Enumerative combinatorics. vol. i, the wadsworth & brooks/cole mathematics series, wadsworth & brooks, 1986.

Martin Takác, Avleen Singh Bijral, Peter Richtárik, and Nati Srebro. Mini-batch primal and dual methods for svms. In *ICML (3)*, pages 1022–1030, 2013.

Herbert S Wilf. *generatingfunctionology*. AK Peters/CRC Press, 2005.

Blake E Woodworth and Nati Srebro. Tight complexity bounds for optimizing composite objectives. In *Advances in neural information processing systems*, pages 3639–3647, 2016.

Appendix A. Proof of Lemma 1

Recall that $\pi_\alpha(z) = 1 - z + \alpha z^{\tau+1}$, and its roots, denoted by ζ_i, are ordered such that $|\zeta_1| \le |\zeta_2| \le \cdots \le |\zeta_{\tau+1}|$. In order to bound from above the magnitude of $1/\zeta_i$, we analyze a related polynomial $p_\alpha(z) = z^{\tau+1}\pi_a(1/z)$ which takes the following explicit form

$$p_\alpha(z) \;=\; z^{\tau+1} - z^\tau + \alpha \;=\; (z-1)z^\tau + \alpha.$$

The roots of p_α are precisely $1/\zeta_i$ (note that, $\pi_a(0) = 1 \neq 0$, hence $\zeta_i \neq 0,\ i \in \{1,\dots,\tau+1\}$). Thus, bounding from above (below) the magnitude of the roots of $p_\alpha(z)$ gives an upper (lower) bound for $|1/\zeta_i|$.

We first establish that for any $\alpha \in \left(0, \frac{1}{20(\tau+1)}\right]$, p_α has a real-valued root in $\left(1 - \frac{1}{2(\tau+1)}, 1-\alpha\right]$. Indeed, for any such α, we have on the one hand,

$$p_\alpha(1-\alpha) \;=\; -\alpha(1-\alpha)^\tau + \alpha \;=\; \alpha\left(1 - (1-\alpha)^\tau\right) \;\ge\; 0,$$

and on the other hand (using the fact that $(1 - 1/2x)^x \ge 1/2$ for all $x \ge 1$),

$$\begin{aligned} p_\alpha\left(1 - \frac{1}{2(\tau+1)}\right) &= -\frac{1}{2(\tau+1)}\left(1 - \frac{1}{2(\tau+1)}\right)^\tau + \alpha \\ &= -\frac{1}{2(\tau+1)}\left(\left(1 - \frac{1}{2(\tau+1)}\right)^{\tau+1}\right)^{\frac{\tau}{\tau+1}} + \alpha \\ &\le -\frac{1}{2(\tau+1)}\left(\frac{1}{2}\right)^{\frac{\tau}{\tau+1}} + \alpha < -\frac{1}{20(\tau+1)} + \alpha \le 0, \end{aligned} \tag{19}$$

so by continuity of p_z, we get that a real-valued root exists in $\left(1 - \frac{1}{2(\tau+1)}, 1-\alpha\right]$.

Next, we show that τ non-dominant roots of p_α are of absolute value smaller than $R = 1 - \frac{3}{2(\tau+1)}$. To this end, we invoke Rouché's theorem, which states that for any two holomorphic functions f, g in some region $K \subseteq \mathbb{C}$ with closed contour ∂K, if $|g(z)| < |f(z)|$ for any $z \in \partial K$, then f and $f+g$ have the same number of zeros (counted with multiplicity) inside K. In particular, choosing $f(z) = -z^\tau$, $g(z) = z^{\tau+1} + \alpha$ and $K = \{z : |z| \le R\}$, it follows that if $|z^{\tau+1} + \alpha| < |-z^\tau|$ for all z such that $|z| = R$, then $f+g$ (which equals our polynomial p_α) has the same number of zeros as $f = -z^\tau$ inside K (namely, exactly τ). However, since p_α is a degree $\tau+1$ polynomial, it has exactly $\tau+1$ roots, so the only root of absolute value larger than R is the real-valued one we found earlier. It remains to verify the condition $|z^{\tau+1} + \alpha| < |-z^\tau|$ for all z such that $|z| = R$. For that, it is sufficient to show that $|z^{\tau+1}| + \alpha < |z^\tau|$ for all such z, or equivalently, $R^\tau > \alpha + R^{\tau+1}$.

$$\begin{aligned} R^\tau \;&=\; \left(1 - \frac{3}{2(\tau+1)}\right)^\tau \;=\; \left(1 - \frac{3}{2(\tau+1)}\right)^\tau - \left(1 - \frac{3}{2(\tau+1)}\right)^{\tau+1} + \left(1 - \frac{3}{2(\tau+1)}\right)^{\tau+1} \\ &=\; \frac{3}{2(\tau+1)}\left(1 - \frac{3}{2(\tau+1)}\right)^\tau + R^{\tau+1}. \end{aligned}$$

By the inequality $1 - 1/(x+1) \ge \exp(-1/x)$ (see Lemma 8 below), we have

$$1 - \frac{3}{2(\tau+1)} \ge \exp\left(\frac{-1}{2/3\tau - 1/3}\right) \quad\Longrightarrow\quad \left(1 - \frac{3}{2(\tau+1)}\right)^\tau \ge \exp\left(\frac{-\tau}{2/3\tau - 1/3}\right)$$

It is straightforward to verify that

$$\frac{-\tau}{2/3\tau - 1/3} \geq -3,\ \tau \geq 1,$$

implying that

$$\begin{aligned} R^\tau &= \frac{3}{2(\tau+1)}\left(1 - \frac{3}{2(\tau+1)}\right)^\tau + R^{\tau+1} \\ &\geq \frac{3}{2e^3(\tau+1)} + R^{\tau+1} \\ &> \frac{1}{20(\tau+1)} + R^{\tau+1} \ \geq\ \alpha + R^{\tau+1}\,, \end{aligned}$$

where in the last inequality we used the assumption that $\alpha \in \left(0, \frac{1}{20(\tau+1)}\right]$. As mentioned earlier, the roots of p_α are exactly the reciprocals of the roots of π_α, therefore we conclude

$$\left|\frac{1}{\zeta_i}\right| \leq 1 - \frac{3}{2(\tau+1)},\ i \in [\tau]. \tag{20}$$

We now turn to bound $|\pi'_\alpha(\zeta_i)|$ from above. By definition, any root of π_a satisfies $\alpha\zeta_i^{\tau+1} - \zeta_i + 1 = 0$. Thus, $\alpha\zeta_i^\tau = \frac{\zeta_i - 1}{\zeta_i}$ (note that as mentioned in the first part of the proof, $\zeta_i \neq 0$). This, in turn, gives

$$\begin{aligned} \pi'_\alpha(\zeta_i) = \alpha(\tau+1)\zeta_i^\tau - 1 &= \frac{(\tau+1)(\zeta_i - 1)}{\zeta_i} - 1 \\ &= \frac{(\tau+1)(\zeta_i - 1) - \zeta_i}{\zeta_i} \\ &= \frac{(\tau+1)\zeta_i - (\tau+1) - \zeta_i}{\zeta_i} \\ &= \frac{\tau\zeta_i - (\tau+1)}{\zeta_i} = \tau - \frac{(\tau+1)}{\zeta_i} = (\tau+1)\left(\frac{\tau}{\tau+1} - \frac{1}{\zeta_i}\right). \end{aligned} \tag{21}$$

In the previous parts of the proof, we showed that the distance from any root of p_α to the contour $\{z \mid |z| = 1 - 1/(\tau+1)\}$ is bounded from below by $\frac{1}{2(\tau+1)}$ (Ineq. 19 and Ineq. 20), therefore

$$|\pi'_\alpha(\zeta_i)| = (\tau+1)\left|1 - \frac{1}{\tau+1} - \frac{1}{\zeta_i}\right| \geq \frac{\tau+1}{2(\tau+1)} = \frac{1}{2},\ i = 1, \ldots, \tau+1\,,$$

thus concluding the proof.

Appendix B. Technical Lemmas

Lemma 6 *For any $\alpha \in [0, 1/\tau]$ and $k \geq 0$, it holds that $|[z^k]1/\pi_\alpha(z)| \leq 1$.*

Proof Recall that by Eq. (11), $b_k = [z^k]\frac{b_0}{\pi_\alpha(z)}$. Therefore, suffices it to prove that (b_k) (defined in 9) with $b_0 = 1$ and $\alpha \in [0, 1/\tau]$, satisfies $|b_k| \leq 1$ for any $k \geq 0$.

For the sake of simplicity, we slightly extend (b_k) to the negative indices by defining $b_{-\tau} = b_{-\tau+1} = \cdots = b_{-1} = 1$. We proceed by full induction. The base case holds trivially by the definition of the initial conditions of b_k. For the induction step, suppose that $|b_0|, \ldots, |b_k| \leq 1$. We have $b_{k+1} = b_k - \alpha b_{k-\tau}$, and therefore

$$b_{k+1} = (1-\alpha)b_k + \alpha(b_k - b_{k-\tau}) = (1-\alpha)b_k + \alpha \sum_{i=k-\tau}^{k-1} (b_{i+1} - b_i).$$

Using the recurrence relation again, this equals

$$(1-\alpha)b_k + \alpha \sum_{i=k-\tau}^{k-1} (-\alpha b_{i-\tau}) = (1-\alpha)b_k - \alpha \left(\alpha \sum_{i=k-2\tau}^{k-\tau-1} b_i \right).$$

By the induction hypothesis, this equals $(1-\alpha)b_k + \alpha r_k$, where $|r_k| \leq \alpha\tau \leq 1$. Thus, b_{k+1} is a weighted average of b_k and r_k which are both in $[-1, +1]$ by the induction hypothesis and the above, implying that we must have $b_{k+1} \in [-1, +1]$ as well. Thus, proving the induction step. ■

Lemma 7 *Let $F(\mathbf{w}) \coloneqq \frac{1}{2}\mathbf{w}^\top A\mathbf{w} + \mathbf{b}^\top \mathbf{w},\ A \in \mathbb{R}^{d\times d}, \mathbf{b} \in \mathbb{R}^d$ be a convex quadratic function defined over $\mathbb{R}^d$. If F is bounded from below, then F has a minimizer at which the gradient vanishes.*

Proof Since F is convex and twice differentiable, A is positive semidefinite. In particular, we have $\mathbb{R}^d = \ker(A) \oplus \mathrm{im}(A)$ (namely, the direct sum of the null space and the image space of A). Thus, $\mathbf{b}$ can be expressed as a sum of two orthogonal vectors $\mathbf{b} = \mathbf{b}^\perp + \bar{\mathbf{b}}$, where $\mathbf{b}^\perp \in \ker(A)$ and $\bar{\mathbf{b}} \in \mathrm{im}(A)$. For any $\alpha \in \mathbb{R}$, we have

$$F(\alpha\mathbf{b}^\perp) = \frac{1}{2}((\mathbf{b}^\perp)^\top A\mathbf{b}^\perp)\alpha^2 + \alpha\mathbf{b}^\top\mathbf{b}^\perp = \alpha\|\mathbf{b}^\perp\|^2.$$

By the hypothesis, F is bounded from below, hence $\mathbf{b}^\perp$ must vanish (otherwise we can take $\alpha \to -\infty$ and make F as negative as we wish). In particular, $\mathbf{b} = \bar{\mathbf{b}} \in \mathrm{im}(A)$. Let $\mathbf{y} \in \mathbb{R}^d$ be such that $A\mathbf{y} = \mathbf{b}$, then $\nabla F(-\mathbf{y}) = A(-\mathbf{y}) + \mathbf{b} = 0$. Lastly, F is convex, therefore $-\mathbf{y}$ must be a (global) minimizer, thus concluding the proof. ■

Lemma 8 *For any $x > 0$, it holds that $1 - 1/(x+1) \geq \exp(-1/x)$.*

Proof Since $(\ln(1+x))' = 1/(1+x) > 0$ for any $x > -1$, it follows by the mean-value theorem that for any $x > 0$

$$\ln(1+x) = \ln(1+x) - \ln(1) = \frac{1}{1+\xi}x,$$

for some $\xi \in (0, x)$, hence $\ln(1+x) \leq x$ for any $x > 0$. In particular, for any $x > 0$ we have

$$\ln\left(1 + \frac{1}{x}\right) \leq \frac{1}{x} \implies \ln\left(\frac{x}{x+1}\right) \geq \frac{-1}{x}.$$

Taking the exponent of both sides yields the desired lower bound. ■

Lemma 9 *Let $\tau \geq 0$. If $\alpha \in (0, 1/(20(\tau+1)]$ then*

$$\left(\frac{1-\frac{3}{2(\tau+1)}}{1-\alpha}\right)^{k+1} \leq \exp\left(-\frac{k+1}{\tau+1}\right).$$

In particular, for $k \geq (\tau+1)\ln(2(\tau+1)) - 1$, we have

$$1 + \tau\left(\frac{1-\frac{3}{2(\tau+1)}}{1-\alpha}\right)^{k+1} \leq 3/2. \tag{22}$$

Proof

$$\left(\frac{1-\frac{3}{2(\tau+1)}}{1-\alpha}\right)^{k+1} = \left(\frac{1-\alpha+\alpha-\frac{3}{2(\tau+1)}}{1-\alpha}\right)^{k+1} = \left(1+\frac{\alpha-\frac{3}{2(\tau+1)}}{1-\alpha}\right)^{k+1} \leq \left(1+\alpha-\frac{3}{2(\tau+1)}\right)^{k+1},$$

where the latter inequality follows from that fact that $\alpha < \frac{1}{20(\tau+1)} < \frac{3}{2(\tau+1)}$. Now,

$$\begin{aligned}\left(1+\alpha-\frac{3}{2(\tau+1)}\right)^{k+1} &\leq \exp\left((k+1)(\alpha-\frac{3}{2(\tau+1)})\right) \leq \exp\left((k+1)\left(\frac{1}{20(\tau+1)}-\frac{3}{2(\tau+1)}\right)\right) \\ &= \exp\left(-\frac{k+1}{\tau+1}\left(\frac{3}{2}-\frac{1}{20}\right)\right) \leq \exp\left(-\frac{k+1}{\tau+1}\right).\end{aligned}$$

Lastly, to derive Ineq. 22, we have

$$1+\tau\left(\frac{1-\frac{3}{2(\tau+1)}}{1-\alpha}\right)^{k+1} \leq 1+(\tau+1)\exp\left(-\frac{k+1}{\tau+1}\right) = 1+\exp\left(\ln(\tau+1)-\frac{k+1}{\tau+1}\right) \leq 1+1/2,$$

where the last inequality by the assumption $k \geq (\tau+1)\ln(2(\tau+1)) - 1$.

■

Appendix C. Proof of Thm. 4

The proof technique is based on a construction, first presented in (Nesterov, 2004, Section 2.1.2), which has been proven effective in various settings of optimization since then.

First, we address the strongly convex case. Given $\mu > \lambda > 0$, we consider the following function (devised by Lan and Zhou (2018)):

$$F(\mathbf{w}) := \frac{\mu(\kappa-1)}{4}\left(\frac{1}{2}\langle A\mathbf{w},\mathbf{w}\rangle - \langle \boldsymbol{\epsilon}_1,\mathbf{w}\rangle\right) + \frac{\lambda}{2}\|\mathbf{w}\|^2, \tag{23}$$

where $\kappa = \mu/\lambda$ as before, $\boldsymbol{\epsilon}_1$ denotes the first unit vector, and A is a $d \times d$ matrix defined as follows

$$A = \begin{pmatrix} 2 & -1 & 0 & 0 & \cdots & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & \cdots & 0 & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & 0 & \cdots & -1 & 2 & -1 \\ 0 & 0 & 0 & 0 & \cdots & 0 & -1 & \frac{\sqrt{\kappa}+1}{\sqrt{\kappa}+3} \end{pmatrix}. \tag{24}$$

It can be easily verified that F is μ-smooth and λ-strongly convex function. Moreover, by (Lan and Zhou, 2018, Lemma 8), it follows that the minimizer of f is $\mathbf{w}^* = (q, q^2, \ldots, q^d)$ where $q = (\sqrt{\kappa} - 1)/(\sqrt{\kappa} + 1)$. In particular, if $\mathbf{w} \in \mathbb{R}^d$ is a vector whose all non-zero entries are located in the first m coordinates, where m is such that $d \geq m/2 + \log(1/2)/\log(q^2)$, then

$$\frac{\|\mathbf{w}\|^2}{\|\mathbf{w}^*\|^2} \geq \frac{\sum_{i=m+1}^{d} q^{2i}}{\sum_{i=1}^{d} q^{2i}} = q^{2(m+1)} \frac{1 - q^{2(d-m-1)}}{1 - q^{2d}} \geq \frac{1}{2} q^{2(m+1)} \geq \frac{1}{2} \exp\left(-\frac{4(m+1)}{\sqrt{\kappa} - 1}\right), \quad (25)$$

where the last two inequalities follow from (Lan and Zhou, 2018, Lemma 9.b) and Lemma 8, respectively. Therefore, by bookkeeping which entries of the iterates are non-zero, we can bound from below the distance to the minimizer. To this end, we will need the following lemma which, based on the tridiagonal structure of the Hessian of F, determines the non-zero entries:

Lemma 10 *Let $F : \mathbb{R}^d \to \mathbb{R}$ be a convex quadratic function specified as follows $F(\mathbf{w}) := \frac{c}{2}\mathbf{w}^\top A \mathbf{w} + d\boldsymbol{\epsilon}_1^\top \mathbf{w}$, where A is a tridiagonal matrix and c, d are real scalars. Assuming that the iterates produced by a given optimization algorithm satisfy $\mathbf{w}_0 = \cdots = \mathbf{w}_\tau = 0$ and*

$$\forall k \geq \tau,\ \mathbf{w}_{k+1} \in \mathit{span}\{\nabla F(\mathbf{w}_0), \nabla F(\mathbf{w}_1), \ldots, \nabla F(\mathbf{w}_{k-\tau})\},$$

then $\mathbf{w}_k \in \mathit{span}\{\boldsymbol{\epsilon}_0, \boldsymbol{\epsilon}_1, \ldots, \boldsymbol{\epsilon}_{\lfloor k/(\tau+1) \rfloor}\}$ for all $k \geq 0$ (where $\boldsymbol{\epsilon}_0$ denotes the vector of all zeros, and $\boldsymbol{\epsilon}_i$ denote the i'th standard unit vector).

Proof

First, note that, given a vector $\mathbf{w} \in \mathbb{R}^d$, such that $\mathbf{w} \in \text{span}\{\boldsymbol{\epsilon}_0, \boldsymbol{\epsilon}_1, \ldots, \boldsymbol{\epsilon}_m\}$ for some $m \geq 0$, we have

$$\nabla F(\mathbf{w}) = cA\mathbf{w} + d\boldsymbol{\epsilon}_1.$$

Since the entries of $\mathbf{w}$ are all zero start from the $m+1$ coordinate, $cA\mathbf{w}$ is a linear combination of the first m columns of A. Being a A tridiagonal matrix, it follows that all the entries of $cA\mathbf{w}$ are zero, except for its first $m+1$ coordinates, that is, $cA\mathbf{w} \in \text{span}\{\boldsymbol{\epsilon}_0, \boldsymbol{\epsilon}_1 \ldots, \boldsymbol{\epsilon}_{m+1}\}$. Together, $\nabla F(\mathbf{w}) = cA\mathbf{w} + d\boldsymbol{\epsilon}_1 \in \text{span}\{\boldsymbol{\epsilon}_1 \ldots, \boldsymbol{\epsilon}_{m+1}\}$.

We proceed by full induction. For $k = 0, \ldots, \tau$, the claim holds trivially. Now, assume the claim holds for all $i \leq k$, where $k \geq \tau$, we show that the claim holds for $k+1$. By the induction hypothesis, $\mathbf{w}_i \in \text{span}\{\boldsymbol{\epsilon}_0, \boldsymbol{\epsilon}_1, \ldots, \boldsymbol{\epsilon}_{\lfloor i/(\tau+1) \rfloor}\}$ for all $i \leq k$. Therefore, by the first part of the proof, we have, $\nabla F(\mathbf{w}_i) \in \text{span}\{\boldsymbol{\epsilon}_1, \boldsymbol{\epsilon}_2, \ldots, \boldsymbol{\epsilon}_{\lfloor i/(\tau+1) \rfloor + 1}\}$ for all $i \leq k$, by which we conclude that $\text{span}\{\nabla F(\mathbf{w}_0), \nabla F(\mathbf{w}_1), \ldots, \nabla F(\mathbf{w}_{k-\tau})\} \subseteq \text{span}\{\boldsymbol{\epsilon}_1, \boldsymbol{\epsilon}_2, \ldots, \boldsymbol{\epsilon}_{\lfloor (k-\tau)/(\tau+1) \rfloor + 1}\}$. Thus, by the linear span assumption, it follows that

$$\mathbf{w}_{k+1} \in \text{span}\{\boldsymbol{\epsilon}_1, \boldsymbol{\epsilon}_2, \ldots, \boldsymbol{\epsilon}_{\lfloor (k-\tau)/(\tau+1) \rfloor + 1}\}. \quad (26)$$

Observing that,

$$\lfloor (k-\tau)/(\tau+1) \rfloor + 1 = \lfloor (k-\tau)/(\tau+1) + 1 \rfloor = \lfloor (k+1)/(\tau+1) \rfloor,$$

concludes the proof. ■

Overall, by Lemma 10, the k'th iterate w_k, has all its entries zero, expect for (possibly) the first $\lfloor k/(\tau+1)\rfloor$ first coordinates. By Ineq. 25, for any $\tau+1 \leq k \leq 2\left(d - \frac{\log(1/2)}{2\log(q)}\right)$, we then have

$$\begin{aligned}\frac{\|\mathbf{w}_k\|^2}{\|\mathbf{w}^*\|^2} &\geq \frac{1}{2}\exp\left(-\frac{4(\lfloor k/(\tau+1)\rfloor+1)}{\sqrt{\kappa}-1}\right) \geq \frac{1}{2}\exp\left(-\frac{4(k/(\tau+1)+1)}{\sqrt{\kappa}-1}\right) \\ &\geq \frac{1}{2}\exp\left(-\frac{5k}{(\sqrt{\kappa}-1)(\tau+1)}\right).\end{aligned}$$

For the convex case, we use a construction (devised by Nesterov (2004)) similar to that of the strongly convex case. Let $\mu > 0$ be fixed and consider the following function

$$F_k(\mathbf{w}) := \frac{\mu}{4}\left(\frac{1}{2}\langle A_k\mathbf{w},\mathbf{w}\rangle - \langle \boldsymbol{\epsilon}_1,\mathbf{w}\rangle\right),$$

where A_k is a $d\times d$ matrix defined as follows

$$A = \begin{pmatrix} 2 & -1 & 0 & 0 & \cdots & 0 & 0 & 0 & \\ -1 & 2 & -1 & 0 & \cdots & 0 & 0 & 0 & \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & 0_{k,d-k} \\ 0 & 0 & 0 & 0 & \cdots & -1 & 2 & -1 & \\ 0 & 0 & 0 & 0 & \cdots & 0 & -1 & 2 & \\ & & & & 0_{d-k,k} & & & & 0_{d-k,d-k} \end{pmatrix},$$

where $0_{m,n}$ is an $m\times n$ zero matrix. Given an iteration number k such that $\tau+1 \leq k \leq \frac{1}{2}(d-1)(\tau+1)$, we take our function F to be $F_{2\lfloor\frac{k}{\tau+1}\rfloor+1}(\mathbf{w})$. Using Lemma 10, the only (possibly) non-zero entries of the k'th iterate $\mathbf{w}_k$ are the first $\lfloor k/(\tau+1)\rfloor$ coordinates. Thus, following the same lines of proof as in (Nesterov, 2004, Theorem 2.1.6) yields

$$\frac{F(\mathbf{w}_k)-F(\mathbf{w}^*)}{\|\mathbf{w}^*\|^2} \geq \frac{3L}{32(k/(\tau+1)+1)^2} \geq \frac{L(\tau+1)^2}{45k^2}.$$

Appendix D. Proof of Thm. 5

We will first state and prove the following auxiliary lemma:

Lemma 11 *The following holds for any $\eta > 0$:*

- *For any $k \geq 1$,*

$$\max_{\{a\,:\,0<a<1/\eta\}} a(1-\eta a)^k \leq \frac{1}{e\eta k},$$

  where $e = 2.718...$ is Euler's number. In particular, $\sum_{i=0}^{k}\max_{\{a\,:\,0<a<1/\eta\}} a(1-\eta a)^{2(i+1)} \leq \frac{1}{2e\eta}H_k \leq \frac{1}{2e\eta}(1+\ln(k+1))$, where H_k denotes the k'th harmonic number.

- *If, in addition, we assume that $a > \lambda$ for some constant $\lambda > 0$, then*

$$\sum_{i=0}^{k}\max_{\{a\,:\,\lambda<a<1/\eta\}} a(1-\eta a)^{2(i+1)} \leq \frac{1+e+\ln(\frac{1}{\eta\lambda})}{e\eta}.$$

Proof By the well-known inequality $1 + x \leq \exp(x),\ x \in \mathbb{R}$, and since for the domain over which we optimize it holds that $1 - \eta a > 0$, we have for any $k \geq 1$

$$a(1 - \eta a)^k \leq a \exp(-\eta a k).$$

Let us denote the latter by $\psi(a) \coloneqq a \exp(-\eta a k)$, and derive for it the desired upper bound.

Taking the derivative of ψ and setting to zero, gives

$$(1 - a\eta k) \exp(-\eta a k) = 0.$$

Therefore, the only stationary point of ψ is $a^* = \frac{1}{\eta k}$. Since ψ' is positive for $a < a^*$ and negative for $a > a^*$, it follows that a^* is a global maximum, at which the value of ψ is $\frac{1}{e\eta k}$, concluding the first part of the proof.

Now, let $\lambda > 0$. Since, the only maximizer of ψ is at $a = \frac{1}{\eta k}$, if $\lambda \geq \frac{1}{2\eta(i+1)}$, or equivalently $i \geq \frac{1}{2\eta\lambda} - 1$, then $\max_{\{a\,:\,\lambda<a<1/\eta\}} a(1-\eta a)^{2(i+1)} \leq \lambda(1-\eta\lambda)^{2(i+1)}$. Therefore,

$$\begin{aligned}
\sum_{i=1}^{k} \max_{\{a\,:\,\lambda<a<1/\eta\}} a(1-\eta a)^{2(i+1)} &\leq \sum_{i=1}^{\lfloor \frac{1}{2\eta\lambda}-1 \rfloor} \max_{\{a\,:\,\lambda<a<1/\eta\}} a(1-\eta a)^{2(i+1)} \\
&\quad + \sum_{i=\lceil \frac{1}{2\eta\lambda}-1 \rceil}^{k} \max_{\{a\,:\,\lambda<a<1/\eta\}} a(1-\eta a)^{2(i+1)} \\
&\leq \sum_{i=1}^{\lfloor \frac{1}{2\eta\lambda}-1 \rfloor} \frac{1}{e\eta k} + \sum_{i=\lceil \frac{1}{2\eta\lambda}-1 \rceil}^{k} \lambda(1-\eta\lambda)^{2(i+1)} \\
&\leq \frac{1}{e\eta}(1 + \ln(\frac{1}{2\eta\lambda})) + \frac{1}{\eta} \\
&\leq \frac{1 + e + \ln(\frac{1}{\eta\lambda})}{e\eta}
\end{aligned}$$

■

We now turn to prove Thm. 5 itself. By Ineq. 18 we have

$$2\mathbb{E}[F(\mathbf{w}_k) - F(\mathbf{w}^*)] \leq \|\sqrt{A}[z^k]D\|^2 \|\mathbf{e}_0\|^2 + \eta^2\sigma^2 \sum_{i=0}^{k} \left\|\sqrt{A}[z^i]D\right\|^2. \tag{27}$$

We will bound each of the terms above separately. Assuming $\eta \in \left(0, \frac{1}{20\mu(\tau+1)}\right]$ we have by Ineq. 14 and Ineq. 13,

$$\begin{aligned}
\|\sqrt{A}[z^k]D\|^2 &= \|\sqrt{A}[z^k]\left((I - Iz + \eta A z^{\tau+1})^{-1}\right)\|^2 \\
&\leq \max_{i\in[d]} \left|\sqrt{a_i}[z^k]\frac{1}{\pi_{\eta a_i}(z)}\right|^2 \\
&\leq \begin{cases} \max_{i\in[d]} a_i & 0 \leq k \leq (\tau+1)\ln(2(\tau+1)) - 1, \\ 9\max_{i\in[d]} a_i(1-\alpha)^{2(k+1)} & k \geq (\tau+1)\ln(2(\tau+1)), \end{cases}
\end{aligned} \tag{28}$$

Thus, for the first term, assuming $k \geq (\tau+1)\ln(2(\tau+1))$, we have

$$\begin{aligned}\|\sqrt{A}[z^k]D\|^2 &\leq 9\max_{i\in[d]} a_i(1-\eta a_i)^{2(k+1)} \leq 9\mu\max_{i\in[d]}(1-\eta a_i)^{2(k+1)} \\ &\leq 9\mu\exp(-2\eta\lambda(k+1)).\end{aligned} \tag{29}$$

Bounding the second term in Ineq. 27 is somewhat more involved and requires seperating into the two regimes stated in Ineq. 28:

$$\begin{aligned}\sum_{i=0}^{k}\left\|\sqrt{A}[z^i]D\right\|^2 &\leq \sum_{i=0}^{\lceil(\tau+1)\ln(2(\tau+1))\rceil-1}\left\|\sqrt{A}[z^i]D\right\|^2 + \sum_{i=\lceil(\tau+1)\ln(2(\tau+1))\rceil}^{k}\left\|\sqrt{A}[z^i]D\right\|^2 \\ &\leq \mu(\tau+1)\ln(2(\tau+1)) + 9\sum_{i=0}^{k}\max_{i\in[d]} a_i(1-\eta a_i)^{2(i+1)}\end{aligned} \tag{30}$$

We proceed by considering the strongly convex case and the convex case separately. For the strongly convex case we have by Lemma 11

$$\begin{aligned}\sum_{i=0}^{k}\left\|\sqrt{A}[z^i]D\right\|^2 &\leq \mu(\tau+1)\ln(2(\tau+1)) + 9\sum_{i=0}^{k}\max_{i\in[d]} a_i(1-\eta a_i)^{2(i+1)} \\ &\leq \mu(\tau+1)\ln(2(\tau+1)) + \frac{1+e+\ln(\frac{1}{\eta\lambda})}{\eta}.\end{aligned}$$

Together with Ineq. 27 and Ineq. 29, this implies that for $k \geq (\tau+1)\ln(2(\tau+1))$,

$$\begin{aligned}2\mathbb{E}[F(\mathbf{w}_k) - F(\mathbf{w}^*)] &\leq \|\sqrt{A}[z^k]D\|^2\|\mathbf{e}_0\|^2 + \eta^2\sigma^2\sum_{i=0}^{k}\left\|\sqrt{A}[z^i]D\right\|^2 \\ &\leq 9\mu\exp(-2\eta\lambda(k+1))\|\mathbf{e}_0\|^2 + \eta^2\sigma^2\left(\mu(\tau+1)\ln(2(\tau+1)) + \frac{1+e+\ln(\frac{1}{\eta\lambda})}{e\eta}\right),\end{aligned}$$

resulting in the first bound stated in the theorem. To get the second bound, we show how to optimally tune the step size η (up to log factors). Ignoring the log factors, the bound above is

$$\mathbb{E}\left(F(\mathbf{w}_k) - F(\mathbf{w}^*)\right) \;\leq\; \tilde{\mathcal{O}}\left(\mu\|\mathbf{e}_0\|^2\exp(-2\eta\lambda k) + \eta^2\sigma^2\left(\mu\tau + \frac{1}{\eta}\right)\right).$$

Moreover, since we assume that $\eta \leq \mathcal{O}(1/\mu\tau)$, we get that $\mu\tau$ is dominated (up to constants) by $1/\eta$, so we can simplify the above to

$$\mathbb{E}\left(F(\mathbf{w}_k) - F(\mathbf{w}^*)\right) \;\leq\; \tilde{\mathcal{O}}\left(\mu\|\mathbf{e}_0\|^2\exp(-2\eta\lambda k) + \eta\sigma^2\right). \tag{31}$$

We now consider three cases:

- If $0 \leq \frac{\ln(\lambda\mu\|\mathbf{e}_0\|^2 k/\sigma^2)}{2\lambda k} \leq \frac{1}{20(\mu\tau)}$, we can pick $\eta = \frac{\ln(\lambda\mu\|\mathbf{e}_0\|^2 k/\sigma^2)}{2\lambda k}$, and get that Eq. (31) is

$$\tilde{\mathcal{O}}\left(\frac{\sigma^2}{\lambda k}\right) \;=\; \tilde{\mathcal{O}}\left(\mu\|\mathbf{e}_0\|^2\exp\left(-\frac{\lambda k}{10\mu\tau}\right) + \frac{\sigma^2}{\lambda k}\right)$$

- If $\frac{\ln(\lambda\mu\|\mathbf{e}_0\|^2k/\sigma^2)}{2\lambda k} < 0$, it follows that $\mu\|\mathbf{e}_0\|^2 \leq \frac{\sigma^2}{\lambda k}$. In that case, we pick $\eta = 0$, and get that Eq. (31) is

$$\tilde{\mathcal{O}}\left(\mu\|\mathbf{e}_0\|^2\right) \;\leq\; \tilde{\mathcal{O}}\left(\frac{\sigma^2}{\lambda k}\right) \;=\; \tilde{\mathcal{O}}\left(\mu\|\mathbf{e}_0\|^2\exp\left(-\frac{\lambda k}{10\mu\tau}\right)+\frac{\sigma^2}{\lambda k}\right).$$

- If $\frac{\ln(\lambda\mu\|\mathbf{e}_0\|^2k/\sigma^2)}{2\lambda k} > \frac{1}{20(\mu\tau)}$, we pick $\eta = \frac{1}{20(\mu\tau)}$, and get that Eq. (31) is

$$\tilde{\mathcal{O}}\left(\mu\|\mathbf{e}_0\|^2\exp\left(-\frac{\lambda k}{10\mu\tau}\right)+\frac{\sigma^2}{\mu\tau}\right) \;\leq\; \tilde{\mathcal{O}}\left(\mu\|\mathbf{e}_0\|^2\exp\left(-\frac{\lambda k}{10\mu\tau}\right)+\frac{\sigma^2}{\lambda k}\right).$$

Collecting the three cases above, we get a bound of

$$\tilde{\mathcal{O}}\left(\mu\|\mathbf{e}_0\|^2\exp\left(-\frac{\lambda k}{10\mu\tau}\right)+\frac{\sigma^2}{\lambda k}\right)$$

as required.

For the convex case, we have by Ineq. 27, Ineq. 28 and Lemma 11, that for $k \geq (\tau+1)\ln(2(\tau+1))$

$$\begin{aligned}&\mathbb{E}[F(\mathbf{w}_k) - F(\mathbf{w}^*)]\\ &\leq \frac{9}{4e\eta(k+1)}\|\mathbf{e}_0\|^2 + \frac{\eta^2\sigma^2}{2}\left(\mu(\tau+1)\ln(2(\tau+1)) + \frac{9}{2e\eta}(1+\ln(k+1))\right),\end{aligned}$$

resulting in the third bound in the theorem. To get the fourth bound, we now show how to optimally tune the step size η (up to log factors). Ignoring the log factors, the bound above is

$$\tilde{\mathcal{O}}\left(\frac{\|\mathbf{e}_0\|^2}{\eta k} + \eta^2\sigma^2\left(\mu\tau + \frac{1}{\eta}\right)\right).$$

As in the strongly convex case, since we assume $\eta \leq \mathcal{O}(1/(\mu\tau)$, we can simplify the above to

$$\tilde{\mathcal{O}}\left(\frac{\|\mathbf{e}_0\|^2}{\eta k} + \eta\sigma^2\right).$$

We now consider two cases:

- If $\frac{\|\mathbf{e}_0\|}{\sigma\sqrt{k}} \leq \frac{1}{20(\mu\tau)}$, we choose $\eta = \frac{\|\mathbf{e}_0\|}{\sigma\sqrt{k}}$, and get

$$\tilde{\mathcal{O}}\left(\frac{\|\mathbf{e}_0\|\sigma}{\sqrt{k}}\right) \;=\; \tilde{\mathcal{O}}\left(\frac{\|\mathbf{e}_0\|^2\mu\tau}{k} + \frac{\|\mathbf{e}_0\|\sigma}{\sqrt{k}}\right).$$

- If $\frac{\|\mathbf{e}_0\|}{\sigma\sqrt{k}} > \frac{1}{20(\mu\tau)}$, we choose $\eta = \frac{1}{20(\mu\tau)}$, and get

$$\tilde{\mathcal{O}}\left(\frac{\|\mathbf{e}_0\|^2\mu\tau}{k} + \frac{\sigma^2}{\mu\tau}\right) \;\leq\; \tilde{\mathcal{O}}\left(\frac{\|\mathbf{e}_0\|^2\mu\tau}{k} + \frac{\|\mathbf{e}_0\|\sigma}{\sqrt{k}}\right).$$

Collecting the two cases above, we get a bound of

$$\tilde{\mathcal{O}}\left(\frac{\|\mathbf{e}_0\|^2\mu\tau}{k} + \frac{\|\mathbf{e}_0\|\sigma}{\sqrt{k}}\right)$$

as required.