Nesterov acceleration despite very noisy gradients

Kanan Gupta
Department of Mathematics
University of Pittsburgh
kanan.g@pitt.edu

Jonathan W. Siegel
Department of Mathematics
Texas A&M University
jwsiegel@tamu.edu

Stephan Wojtowytsch
Department of Mathematics
University of Pittsburgh
s.woj@pitt.edu

Abstract

We present a generalization of Nesterov's accelerated gradient descent algorithm. Our algorithm (AGNES) provably achieves acceleration for smooth convex and strongly convex minimization tasks with noisy gradient estimates if the noise intensity is proportional to the magnitude of the gradient at every point. Nesterov's method converges at an accelerated rate if the constant of proportionality is below 1, while AGNES accommodates any signal-to-noise ratio. The noise model is motivated by applications in overparametrized machine learning. AGNES requires only two parameters in convex and three in strongly convex minimization tasks, improving on existing methods. We further provide clear geometric interpretations and heuristics for the choice of parameters.

1 Introduction

The recent success of deep learning [LeCun et al., 2015] is built on stochastic first order optimization methods such as stochastic gradient descent [LeCun et al., 1998] and ADAM [Kingma and Ba, 2014], which have enabled the large-scale training of neural networks. While such tasks are generally non-convex, accelerated first order methods for convex optimization have proved practically useful. Specifically, Nesterov [1983]'s accelerated gradient descent has become a standard training method [Sutskever et al., 2013].

Modern neural networks tend to operate in the *overparametrized* regime, i.e. the number of model parameters exceeds the number of data points to be fit [Belkin, 2021]. In this setting, minibatch gradient estimates are exact (namely, exactly 0) on the set of global minimizers since data can be interpolated exactly. Motivated by such applications, Vaswani et al. [2019] proved that Nesterov [2012]'s accelerated coordinate descent method (ACDM) achieves acceleration in (strongly) convex optimization with *multiplicative noise*, i.e. when assuming stochastic gradient estimates for which the noise intensity scales linearly with the magnitude of the gradient. Conversely, Liu and Belkin [2018] show that the original version of Nesterov [1983]'s method generally does not achieve acceleration in this setting.

Another algorithm with a similar goal is the continuized Nesterov method (CNM), which has been studied by Even et al. [2021], Berthier et al. [2021] in convex optimization (deterministic or with additive noise) and with multiplicative noise for overparametrized linear least squares regression. For a more extensive discussion of the context of our work in the literature, please see Section 2.

Vaswani et al. [2019]'s algorithm is a four parameter scheme in the strongly convex case, which reduces to a three parameter scheme in the convex case. Liu and Belkin [2018] introduce a simpler three parameter scheme, but only prove that it achieves acceleration for overparametrized *linear problems*. In this work, we demonstrate that it is possible to achieve the same theoretical guarantees as Vaswani et al. [2019] with a simpler scheme, which can be considered as a reparametrized version of Liu and Belkin [2018]'s Momentum-Added Stochastic Solver (MaSS) method. More precisely, we prove the following:

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

1. We show that Nesterov's accelerated gradient descent achieves an accelerated convergence rate, but *only with noise which is strictly smaller than the gradient in the L^2-sense*. We also show numerically that when the noise is larger than the gradient, the algorithm diverges for a choice of step size for which gradient descent remains convergent.
2. Motivated by this, we introduce a generalization of Nesterov's method, which we call Accelerated Gradient descent with Noisy EStimators (AGNES), which provably achieves acceleration *no matter how large the noise is relative to the gradient, both in the convex and strongly convex cases*.
3. When moving from NAG to AGNES, the learning rate 'bifurcates' to two parameters in order to accommodate stochastic gradient estimates. The extension requires three hyperparameters in the strongly convex case and two in the convex case.
4. We provide a transparent geometric interpretation of the AGNES parameters in terms of their scaling with problem parameters (Appendix F.3) and the continuum limit models for various scaling regimes (Appendix C).
5. We build strong intuition for the choice of hyperparameters for machine learning applications and empirically demonstrate that AGNES improves the training of CNNs relative to SGD with momentum and Nesterov's accelerated gradient descent.

2 Literature Review

Accelerated first order methods. Accelerated first order methods have been extensively studied in convex optimization. Beginning with the conjugate gradient (CG) algorithm introduced by Hestenes and Stiefel [1952], the Heavy ball method of Polyak [1964], and Nesterov [1983]'s seminal work on accelerated gradient descent, many authors have developed and analyzed accelerated first order methods for convex problems, including Beck and Teboulle [2009], Nesterov [2012, 2013], Chambolle and Dossal [2015], Kim and Fessler [2018] to name just a few.

An important line of research is to gain an understanding of how accelerated methods work. After Polyak [1964] derived the original Heavy ball method as a discretization of an ordinary differential equation, Alvarez et al. [2002], Su et al. [2014], Wibisono et al. [2016], Zhang et al. [2018], Siegel [2019], Shi et al. [2019], Muehlebach and Jordan [2019], Wilson et al. [2021], Shi et al. [2021], Suh et al. [2022], Attouch et al. [2022], Aujol et al. [2022b,a], Dambrine et al. [2022] studied accelerated first order methods from the point of view of ODEs. This perspective has facilitated the use of Lyapunov functional analysis to quantify the convergence properties. We remark that in addition to the intuition provided by differential equations, Joulani et al. [2020] and Gasnikov and Nesterov [2018] have also proposed interesting ideas for explaining and deriving accelerated first-order methods. In addition, there has been a large interest in deriving adaptive accelerated first order methods, see for instance Levy et al. [2018], Cutkosky [2019], Kavis et al. [2019].

Stochastic optimization. Robbins and Monro [1951] first introduced optimization algorithms where gradients are only estimated by a stochastic oracle. For convex optimization, Nemirovski et al. [2009], Ghadimi and Lan [2012] obtained minimax-optimal convergence rates with additive stochastic noise.

In deep learning, stochastic algorithms are ubiquitous in the training of deep neural networks, see [LeCun et al., 1998, 2015, Goodfellow et al., 2016, Bottou et al., 2018]. Here, the additive noise assumption not usually appropriate. As Wojtowytsch [2023], Wu et al. [2022a] show, the noise is of low rank and degenerates on the set of global minimizers. Stich [2019], Stich and Karimireddy [2022], Bassily et al. [2018], Gower et al. [2019], Damian et al. [2021], Wojtowytsch [2023], Zhou et al. [2020] consider various non-standard noise models and [Wojtowytsch, 2021, Zhou et al., 2020, Li et al., 2022] study the continuous time limit of stochastic gradient descent. These include noise assumptions for degenerate noise due to Bassily et al. [2018], Damian et al. [2021], Wojtowytsch [2023, 2021], low rank noise studied by Damian et al. [2021], Li et al. [2022] and noise with heavy tails explored by Zhou et al. [2020].

Acceleration with stochastic gradients. Kidambi et al. [2018] prove that there are situations in which it is impossible for any first order oracle method to improve upon SGD due to information-

theoretic lower bounds. More generally, lower bounds in the stochastic first order oracle (SFO) model were presented by Nemirovski et al. [2009] (see also [Ghadimi and Lan, 2012]).

A partial improvement on the state of the art is given by Jain et al. [2018], who present an accelerated stochastic gradient method motivated by a particular low-dimensional and strongly convex problem. Laborde and Oberman [2020] obtain faster convergence of an accelerated method under an additive noise assumption by a Lyapunov function analysis. Bollapragada et al. [2022] study an accelerated gradient method for the optimization of a strongly convex quadratic objective function with minibatch noise.

Closest to our work are Liu and Belkin [2018], Vaswani et al. [2019], Even et al. [2021], Berthier et al. [2021] who study generalizations of Nesterov's method in stochastic optimization. Liu and Belkin [2018], Even et al. [2021] obtain guarantees with noise of approximately multiplicative noise in overparametrized linear least squares problems and for general convex objective functions with additive noise and in deterministic optimization. Vaswani et al. [2019] obtain comparable guarantees for the more complicated method of Nesterov [2013].

3 Algorithm and Convergence Guarantees

3.1 Assumptions

In the remainder of this article, we consider the task of minimizing an objective function $f : \mathbb{R}^m \to \mathbb{R}$ using stochastic gradient estimates g. We assume that f, g and the initial condition x_0 satisfy:

1. The initial condition x_0 is a (potentially random) point such that $\mathbb{E}[f(x_0) + \|x_0\|^2] < \infty$.
2. f is $L-$smooth, i.e. ∇f is $L-$Lipschitz continuous with respect to the Euclidean norm.
3. There exists a probability space $(\Omega, \mathcal{A}, \mathbb{P})$ and a gradient estimator, i.e. a measurable function $g : \mathbb{R}^m \times \Omega \to \mathbb{R}^m$ such that for all $x \in \mathbb{R}^m$ the properties
   - $\mathbb{E}_\omega[g(x,\omega)] = \nabla f(x)$ (unbiased gradient oracle) and
   - $\mathbb{E}_\omega\left[\|g(x,\omega) - \nabla f(x)\|^2\right] \leq \sigma^2 \|\nabla f(x)\|^2$ (multiplicative noise scaling) hold.

A justification of the multiplicative noise scaling is given in Section 4. In the setting of machine learning, the space Ω is given by the random subsampling of the dataset. A rigorous discussion of the probabilistic foundations is given in Appendix D.

3.2 Nesterov's Method with Multiplicative Noise

First we analyze Nesterov [1983]'s accelerated gradient descent algorithm (NAG) in the setting of multiplicative noise. NAG is given by the initialization $x_0 = x'_0$ and the two-step iteration

$$x_{n+1} = x'_n - \eta g'_n, \qquad x'_{n+1} = x_{n+1} + \rho_n\big(x_{n+1} - x_n\big) = x_{n+1} + \rho_n(x'_n - \eta g'_n - x_n) \tag{1}$$

where $g'_n = g(x'_n, \omega_n)$ and the variables ω_n are iid samples from the probability space Ω, i.e. g'_n is an unbiased estimate of $\nabla f(x'_n)$. We write ρ instead of ρ_n in cases where a dependence on n is not required. We show that this scheme achieves an $O(1/n^2)$ convergence rate for convex functions but *only in the case that* $\sigma < 1$. To the best of our knowledge, this analysis is optimal.

Theorem 1 (NAG, convex case)**.** *Suppose that x_n and x'_n are generated by the time-stepping scheme (1), f and g satisfy the conditions laid out in Section 3.1, f is convex, and x^* is a point such that $f(x^*) = \inf_{x\in\mathbb{R}^m} f(x)$. If $\sigma < 1$ and the parameters are chosen such that*

$$0 < \eta \leq \frac{1-\sigma^2}{L(1+\sigma^2)}, \quad \textit{and} \quad \rho_n = \frac{n}{n+3}, \qquad \textit{then} \qquad \mathbb{E}[f(x_n) - f(x^*)] \leq \frac{2\mathbb{E}[\|x_0 - x^*\|^2]}{\eta n^2}.$$

The expectation on the right hand side is over the random initialization x_0.

The proof of Theorem 1 is given in Appendix E. Note that the constant $1/\eta$ blows up as $\sigma \nearrow 1$ and the analysis yields no guarantees for $\sigma > 1$. This mirrors numerical experiments in Section 5.

Theorem 2 (NAG, strongly convex case)**.** *In addition to the assumptions in Theorem 1, suppose that f is μ-strongly convex and the parameters are chosen such that*

$$0 < \eta \leq \frac{1-\sigma^2}{L(1+\sigma^2)} \textit{ and } \rho = \frac{1-\sqrt{\mu\eta}}{1+\sqrt{\mu\eta}}, \textit{ then } \mathbb{E}[f(x_n) - f(x^*)] \leq 2(1-\sqrt{\mu\eta})^n\, \mathbb{E}\left[f(x_0) - f(x^*)\right].$$

Just like in the convex case, the step size η decreases to zero as $\sigma \nearrow 1$, and we fail to obtain convergence guarantees for $\sigma \geq 1$. We argue in the proof of Theorem 2, given in Appendix F, that it is not possible to modify the Lyapunov sequence analysis to obtain a better rate of convergence. This motivates our introduction of the more general AGNES method below.

Notably, there cannot be a diverging lower bound for NAG since gradient descent arises in the special case $\rho = 0$, and gradient descent converges for small stepsize with multiplicative noise [Wojtowytsch, 2023]. On the other hand, Liu and Belkin [2018] show that NAG does not achieve accelerated convergence with multiplicative type noise even for quadratic strongly convex functions.

3.3 AGNES Descent algorithm

The proofs of Theorems 1 and 2 suggest that the momentum step in (1) is quite sensitive to the step size used for the gradient step, which severely restricts the step size η. We propose the Accelerated Gradient descent with Noisy EStimators (AGNES) scheme, which addresses this problem by introducing an additional parameter α in the momentum step:

$$x_0 = x_0', \qquad x_{n+1} = x_n' - \eta g_n', \qquad x_{n+1}' = x_{n+1} + \rho_n\big(x_n' - \alpha g_n' - x_n\big), \tag{2}$$

where $g_n' = g(x_n', \omega_n)$ as before. Equivalently, AGNES can be formulated as a three-step scheme with an auxiliary velocity variable v_n, initialized as $v_0 = 0$:

$$x_n' = x_n + \alpha v_n, \qquad x_{n+1} = x_n' - \eta g_n', \qquad v_{n+1} = \rho_n(v_n - g_n'). \tag{3}$$

We show that the two formulations of AGNES are equivalent in Appendix B.1. However, we find (3) more intuitive (see Appendix C for a continuous time interpretation) and easier to implement as an algorithm without storing past values of x_n. The pseudocode and a set of suggested default parameters are given in Algorithm 1.

Algorithm 1: Accelerated Gradient descent with Noisy EStimators (AGNES)

Input: f (objective/loss function), x_0 (initial point), $\alpha = 10^{-3}$ (learning rate), $\eta = 10^{-2}$ (correction step size), $\rho = 0.99$ (momentum), N (number of iterations)

$n \leftarrow 0$
$v_0 \leftarrow 0$
while $n < N$ **do**
  $g_n \leftarrow \nabla_x f(x_n)$ `// gradient estimate`
  $v_{n+1} \leftarrow \rho(v_n - g_n)$
  $x_{n+1} \leftarrow x_n + \alpha v_{n+1} - \eta g_n$
  $n \leftarrow n + 1$
end
$g_N \leftarrow \nabla_x f(x_N)$
$x_N \leftarrow x_N - \eta g_n$
Return: x_N

From (1) and (2), we note that NAG is AGNES with the special choice $\alpha = \eta$. Allowing α and η to be different helps AGNES achieve an accelerated rate of convergence for both convex and strongly convex functions, no matter how large σ is. While for gradient descent, only the product $L(1+\sigma^2)$ has to be considered, this is not the case for momentum-based schemes. We consider first the convergence rate in the convex case.

Theorem 3 (AGNES, convex case)**.** *Suppose that x_n and x_n' are generated by the time-stepping scheme (3), f and $g_n' = g(x_n', \omega_n)$ satisfy the conditions laid out in Section 3.1, f is convex, and x^* is a point such that $f(x^*) = \inf_{x\in\mathbb{R}^m} f(x)$. If the parameters are chosen such that*

$$0 < \eta < \frac{1}{L(1+\sigma^2)}, \quad \alpha = \frac{\eta}{1+\sigma^2}, \quad \rho_n = \frac{n}{n+1+a_0}, \quad \textit{for} \quad a_0 \geq \frac{2(1-\eta L)}{1-\eta L(1+\sigma^2)}, \quad \textit{then}$$

$$\mathbb{E}\big[f(x_n) - f(x^*)\big] \leq \frac{a_0^2\,\mathbb{E}\big[\,\|x_0 - x^*\|^2\big]}{2\,\alpha\,n^2}.$$

In particular, if $\eta \leq \frac{1}{L(1+2\sigma^2)}$, we may make the universal choice $a_0 = 4$, i.e. $\rho_n = \frac{n}{n+5}$. Only the parameters η, α depend on the specific problem. The proof of Theorem 3 is given in Appendix E.

There, we also present an alternative version of Theorem 3 for a different choice of parameters

$$\eta \leq \frac{1}{L(1+\sigma^2)}, \quad \alpha < \frac{\eta}{1+\sigma^2}, \quad \rho_n = \frac{n+n_0}{n+n_0+3}$$

for a potentially large $n_0 \geq \frac{2\eta\sigma^2}{\eta - \alpha(1+\sigma^2)} \geq 2\sigma^2$. The convergence guarantees are similar in both cases.

The benefit of the accelerated scheme is an improvement from a decay rate of $O(1/n)$ to the rate $O(1/n^2)$, which is optimal under the given assumptions even in the deterministic case. While the noise can be orders of magnitude larger than the quantity we want to estimate, it only affects the constants in the convergence, not the rate. We get an analogous result for strongly convex functions.

Theorem 4 (AGNES, strongly convex case)**.** *In addition to the assumptions in Theorem 3, suppose that f is μ-strongly convex and the parameters are chosen such that*

$$0 < \eta \leq \frac{1}{L(1+\sigma^2)}, \qquad \rho = \frac{1 - \sqrt{\frac{\mu\eta}{1+\sigma^2}}}{1 + \sqrt{\frac{\mu\eta}{1+\sigma^2}}}, \qquad \textit{and} \qquad \alpha = \frac{1 - \sqrt{\frac{\mu}{L}}}{1 - \sqrt{\frac{\mu}{L}} + \sigma^2}\,\eta \qquad \textit{then}$$

$$\mathbb{E}[f(x_n) - f(x^*)] \leq 2\left(1 - \sqrt{\frac{\mu\eta}{1+\sigma^2}}\right)^n \mathbb{E}[f(x_0) - f(x^*)].$$

Choosing η too small can be interpreted as overestimating L or σ. Choosing α too small (with respect to η) can be interpreted as overestimating σ. Since every L-Lipschitz function is L'-Lipschitz for $L' > L$, and since the multiplicative noise bound with constant σ implies the same bound with $\sigma' > \sigma$, exponential convergence still holds at a generally slower rate.

We note that since $|\nabla f(x)|^2 \leq 2L(f(x) - \inf f)$ (Lemma 12 in Appendix D), Theorems 3 and 4 lead to analogous convergence results for $\mathbb{E}[\nabla f(x_n)]$ as well. Due to the summability of the sequences n^{-2} and r^n for $r < 1$, we get not only convergence in expectation but also almost sure convergence. The proof is given in Appendix E.

Corollary 5. *In the setting of Theorems 3 and 4, $f(x_n) \to \inf f$ with probability 1.*

In the deterministic case $\sigma = 0$, we have $\alpha = \eta$ in both Theorems 3 and 4. In Theorem 4, the parameters coincide with the usual choice for NAG, while we opted for a simple statement in Theorem 3 which does not exactly recover the standard choice $\eta = 1/L$ and $\rho_n = n/(n+3)$. The proofs below easily cover these special cases as well. If $0 < \sigma < 1$, both AGNES and NAG converge with the same rate n^{-2} in the convex case, but the constant of NAG is always larger. In the strongly convex case, even the decay rate of NAG is slower than AGNES for $\sigma \in (0,1)$ since $1 - \sigma^2 < (1+\sigma^2)^{-1}$. We see the real power of AGNES in the stochastic setting where it converges for very high values of σ when Nesterov's method may diverge. For the optimal choice of parameters, we summarize the results in terms of the time-complexity of SGD and AGNES in Figure 1. For the related guarantee for SGD, see Theorems 17 and 22 in Appendices E and F respectively.

Remark 6 (Batching). Let us compare AGNES with two families of gradient estimators:

1. $g'_n = g(x'_n, \omega_n)$ as studied in Theorems 3 and 4.
2. A gradient estimator $g'_n := \frac{1}{n_b}\sum_{j=1}^{n_b} g(x'_n, \omega_{n,j})$ which averages multiple independent estimates to reduce the variance.

The second gradient estimator falls into the same framework with $\tilde{\Omega} = \Omega^{n_b}$ and $\tilde{\sigma}^2 = \sigma^2/n_b$. Assuming vector additions cost negligible time, optimizer steps are only as expensive as gradient evaluations. In this setting – which is often realistic in deep learning – it is appropriate to compare $\mathbb{E}[f(x_{n_b n})]$ ($n_b \cdot n$ iterations using g'_n) and $\mathbb{E}[f(X_n)]$ (n iterations with g'_n). For the strongly convex case, we note that $\left(1 - \sqrt{\frac{\mu}{L}}\frac{1}{1+\sigma^2}\right)^{n_b} \leq 1 - \sqrt{\frac{\mu}{L}}\frac{1}{1+\sigma^2/n_b}$ if and only if

$$n_b \geq \frac{\log\left(1 - \sqrt{\frac{\mu}{L}}\frac{1}{1+\sigma^2/n_b}\right)}{\log\left(1 - \sqrt{\frac{\mu}{L}}\frac{1}{1+\sigma^2}\right)} \approx \frac{\sqrt{\frac{\mu}{L}}\frac{1}{1+\sigma^2/n_b}}{\sqrt{\frac{\mu}{L}}\frac{1}{1+\sigma^2}} = \frac{1+\sigma^2}{1+\sigma^2/n_b} = \frac{1+\sigma^2}{n_b+\sigma^2}\,n_b.$$

Time complexity	Convex	μ-strongly convex
SGD	$O\left(\frac{L}{\varepsilon}(1+\sigma^2)\right)$	$O\left(\frac{L}{\mu}(1+\sigma^2)\lvert\log\varepsilon\rvert\right)$
AGNES	$O\left(\sqrt{\frac{L}{\varepsilon}}(1+\sigma^2)\right)$	$O\left(\sqrt{\frac{L}{\mu}}(1+\sigma^2)\lvert\log\varepsilon\rvert\right)$

Figure 1: The minimal n for AGNES and SGD such that $\mathbb{E}[f(x_n) - \inf f] < \varepsilon$ when minimizing an L-smooth function with multiplicative noise intensity σ in the gradient estimates and under a convexity assumption. The SGD rate of the μ-strongly convex case is achieved more generally under a PL condition with PL-constant μ. While SGD requires the optimal choice of one variable to achieve the optimal rate, AGNES requires three (two in the determinstic case).

The approximation is well-justified in the important case that $\mu \ll L$. In particular, the upper bound for non-batching AGNES is *always* favorable compared to the batching version as $n_b \in \mathbb{N}_{\geq 1}$, and the two only match for the optimal batch size $n_b = 1$. The optimal batch size for minimizing f is the largest one that can be processed in parallel without increasing the computing time for a single step. A similar argument holds for the convex case.

With a slight modification, the proof of Theorem 3 extends to the situation of convex objective functions which do not have minimizers. Such objectives arise for example in linear classification with the popular cross-entropy loss function and linearly separable data.

Theorem 7 (Convexity without minimizers)**.** *Let f be a convex objective function satisfying the assumptions in Section 3.1 and x_n be generated by the time-stepping scheme (3). Assume that η, α and ρ_n are as in Theorem 3. Then* $\liminf_{n\to\infty} \mathbb{E}[f(x_n)] = \inf_{x\in\mathbb{R}^m} f(x)$.

The proof and more details are given in Appendix E. For completeness, we consider the case of non-convex optimization in Appendix G. As a limitation, we note that multiplicative noise is well-motivated in machine learning for global minimizers, but not at generic critical points.

3.4 Geometric Interpretation

Let us briefly discuss the parameter choices in Theorem 4. As we consider larger σ for fixed μ and L, the decay factor ρ moves closer to 1. This slows the 'forgetting' of past gradients in v_n, allowing us to better average out stochastic noise. The price we pay is computing with more outdated gradients, slowing convergence. Our choice balances these effects.

In AGNES, ρ inadvertently also governs magnitude of the momentum variable v_n, which scales as $(1-\rho)^{-1}$ for objective functions with constant gradient and $n \gg 1$. To compensate, we choose α smaller compared to η when σ (and thus $(1-\rho)^{-1}$) is large. Nevertheless, the effect of the momentum step does not decrease. For further details, see Appendix F.3.

For further interpretability, we obtain several ODE and SDE continuous time descriptions of AGNES in Appendix C.

4 Motivation for Multiplicative Noise

In supervised learning applications, the learning task often corresponds to minimizing a risk or loss function $\mathcal{R}(w) = \frac{1}{N}\sum_{i=1}^N \ell\big(h(w,x_i),y_i\big) =: \frac{1}{N}\sum_{i=1}^N \ell_i(w)$, where $h : \mathbb{R}^m \times \mathbb{R}^d \to \mathbb{R}^k$, $(w,x) \mapsto h(w,x)$ and $\ell : \mathbb{R}^k \times \mathbb{R}^k \to [0,\infty)$ are a parametrized function of weights w and data x and a loss function measuring compliance between $h(w,x_i)$ and y_i respectively.[1] Safran and Shamir [2018], Chizat and Bach [2018], Du et al. [2018] show that working in the overparametrized regime $m \gg N$ simplifies the optimization process and Belkin et al. [2019, 2020] illustrate that it facilitates generalization to previously unseen data. Cooper [2019] shows that fitting N constraints with m parameters typically leads to an $m-N$-dimensional submanifold $\mathcal{M}$ of the parameter space $\mathbb{R}^m$

[1] Both ℓ and $\mathcal{R}$ are commonly called a 'loss function' in the literature. To distinguish between the two, we will borrow the terminology of statistics and refer to $\mathcal{R}$ as the risk functional and ℓ as the loss function. The notation L, which is often used in place of $\mathcal{R}$, is reserved for the Lipschitz constant in this work.

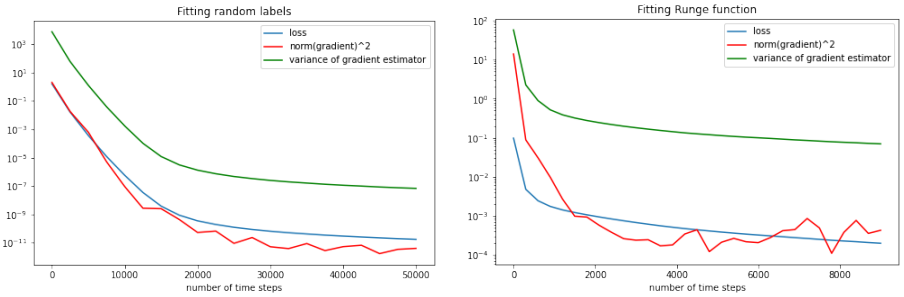

Figure 2: To be able to quantify the gradient noise exactly, we choose relatively small models and data sets. **Left:** A ReLU network with four hidden layers of width 250 is trained by SGD to fit random labels y_i (drawn from a 2-dimensional standard Gaussian) at $1,000$ random data points x_i (drawn from a 500-dimensional standard Gaussian). The variance σ^2 of the gradient estimators is $\sim 10^5$ times larger than the loss function and $\sim 10^6$ times larger than the parameter gradient. This relationship is stable over approximately ten orders of magnitude. **Right:** A ReLU network with two hidden layers of width 50 is trained by SGD to fit the Runge function $1/(1+x^2)$ on equispaced data samples in the interval $[-8, 8]$. Also here, the variance in the gradient estimates is proportional to both the loss function and the magnitude of the gradient.

such that all given labels y_i are fit exactly by $h(w, \cdot)$ at the data points x_i for $w \in \mathcal{M}$, i.e. $\mathcal{R} \equiv 0$ on the smooth set of minimizers $\mathcal{M} = \mathcal{R}^{-1}(\{0\})$.

If N is large, it is computationally expensive to evaluate the gradient $\nabla\mathcal{R}(w) = \frac{1}{N}\sum_{i=1}^N \nabla\ell_i$ of the risk function $\mathcal{R}$ exactly and we commonly resort to stochastic estimates

$$g = \frac{1}{n_b}\sum_{i\in I_b} \nabla\ell_i(w) = \frac{1}{n_b}\sum_{i\in I_b}\sum_{j=1}^{k} (\partial_{h_j}\ell)\big(h(w,x_i), y_i\big)\, \nabla_w h_j(w, x_i),$$

where $I_b \subseteq \{1, \dots, N\}$ is a subsampled collection of n_b data points (a batch or minibatch). Minibatch gradient estimates are very different from the stochasticity we encounter e.g. in statistical mechanics:

1. The covariance matrix $\Sigma = \frac{1}{N}\sum_{i=1}^N \big(\nabla\ell_i - \nabla\mathcal{R}\big) \otimes \big(\nabla\ell_i - \nabla\mathcal{R}\big)$ of the gradient estimators $\nabla\ell_i$ has low rank $N \ll m$.
2. Assume specifically that ℓ is a loss function which satisfies $\ell(y, y) = 0$ for all $y \in \mathbb{R}^k$, such as the popular ℓ^2-loss function $\ell(h, y) = \|h - y\|^2$. Then $\nabla\ell_i(w) = 0$ for all $i \in \{1, \dots, N\}$ and all $w \in \mathcal{M} = \mathcal{R}^{-1}(0)$. In particular, minibatch gradient estimates are exact on $\mathcal{M}$.

The following Lemma makes the second observation precise in the overparameterized regime and bounds the stochasticity of mini-batch estimates more generally.

Lemma 8 (Noise intensity)**.** *Assume that $\ell(h, y) = \|h - y\|^2$ and $h : \mathbb{R}^m \times \mathbb{R}^d \to \mathbb{R}^k$ satisfies $\|\nabla_w h(w, x_i)\|^2 \leq C\big(1 + \|w\|\big)^p$ for some $C, p > 0$ and all $w \in \mathbb{R}^m$ and $i = 1, \dots, N$. Then for all $w \in \mathbb{R}^m$:*

$$\frac{1}{N}\sum_{i=1}^N \big\|\nabla\ell_i - \nabla\mathcal{R}\big\|^2 \;\leq\; 4C^2\,(1 + \|w\|)^{2p}\,\mathcal{R}(w).$$

Lemma 8 is proved in Appendix H. It is a modification of [Wojtowytsch, 2023, Lemma 2.14] for function models which are locally, but not globally Lipschitz-continuous in the weights w, such as deep neural networks with smooth activation function. The exponent p may scale with network depth.

Lemma 8 describes the variance of a gradient estimator which uses a random index $i \in \{1, \dots, N\}$ and the associated gradient $\nabla\ell_i$ is used to approximate $\nabla\mathcal{R}$. If a batch I_b of n_b indices is selected randomly with replacement, then the variance of the estimates scales in the usual way:

$$\mathbb{E}_{I_b}\left[\left\|\frac{1}{n_b}\sum_{i\in I_b}\nabla\ell_i - \nabla\mathcal{R}\right\|^2\right] \leq \frac{4C^2(1 + \|w\|)^{2p}}{n_b}\,\mathcal{R}(w). \tag{4}$$

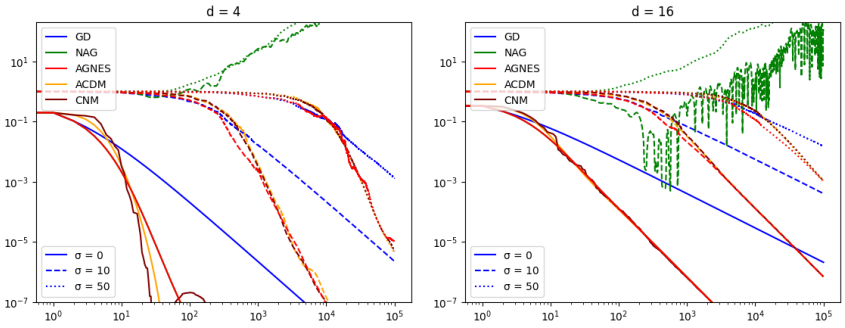

Figure 3: We plot $\mathbb{E}\big[f_d(x_n)\big]$ on a loglog scale for SGD (blue), AGNES (red), NAG (green), ACDM (orange) and CNM (maroon) with $d = 4$ (left) and $d = 16$ (right) for noise levels $\sigma = 0$ (solid line), $\sigma = 10$ (dashed) and $\sigma = 50$ (dotted). The initial condition is $x_0 = 1$ in all simulations. Means are computed over 200 runs. After an initial plateau, AGNES, CNM and ACDM significantly outperform SGD in all settings, while NAG (green) diverges if σ is large. The length of the initial plateau increases with σ.

As noted by Wu et al. [2019, 2022b], $\mathcal{R}$ and $\|\nabla\mathcal{R}\|^2$ often behave similarly in overparametrized deep learning. We illustrate this in Figure 2 together with Lemma 8. Heuristically, we therefore replaced (4) by a more manageable assumption akin to $\mathbb{E}[\frac{1}{N}\sum_{i=1}^{N}\|\nabla\ell_i - \nabla\mathcal{R}\|^2] \leq \sigma^2\|\nabla\mathcal{R}\|^2$ in Section 3.1. The setting where the signal-to-noise ratio (the quotient of estimate variance and true magnitude) is $\Omega(1)$ is often referred to as 'multiplicative noise', as it resembles the noise generated by estimates of the form $g = (1+\sigma Z)\nabla\mathcal{R}$, where $Z \sim \mathcal{N}(0,1)$. When the objective function is L-smooth and satisfies a PL condition (see e.g. [Karimi et al., 2016]), both scaling assumptions are equivalent.

5 Numerical Experiments

5.1 Convex optimization

We compare the optimization algorithms for the family of objective functions

$$f_d : \mathbb{R} \to \mathbb{R}, \qquad f_d(x) = \begin{cases} |x|^d & \text{if } |x| < 1 \\ 1 + d(|x| - 1) & \text{else} \end{cases}$$

for $d \geq 2$ with gradient estimators $g = (1+\sigma N)f'(x)$, where N is a unit normal random variable. The functions are convex and their derivatives are Lipschitz-continuous with $L = d(d-1)$. Various trajectories are compared for different values of d and σ in Figure 3. We run AGNES with the parameters $\alpha = \frac{\eta}{1+\sigma^2}$, $\eta = \frac{1}{L(1+2\sigma^2)}$, $\rho_n = \frac{n}{n+5}$ derived above and SGD with the optimal step size $\eta = \frac{1}{L(1+\sigma^2)}$ (see Lemmas 16 and 17). For NAG, we select $\eta = \frac{1}{L(1+\sigma^2)}$ and $\rho_n = \frac{n}{n+3}$. We present a similar experiment in the strongly convex case in Appendix A.

We additionally compare to two other methods of accelerated gradient descent which were recently proposed for multiplicative noise models: The ACDM method of Nesterov [2012], Vaswani et al. [2019], and the continuized Nesterov method (CNM) of Even et al. [2021], Berthier et al. [2021] with the proposed parameters. In this simple setting where all constants are known, AGNES, ACDM and CNM perform comparably in the long run and on average.

5.2 Neural network regression

We generated $n = 100,000$ 12-dimensional random vectors. Using a fixed, randomly initialized neural network f^* (with 10 hidden layers, each with width 10, and output dimension 1), we produced labels $y_i = f^*(x_i)$. The resulting dataset was split into 90% training and 10% testing data. We then trained identically initialized copies of a larger neural network (15 hidden layers, each with width 15) using Adam, NAG, SGD with momentum, and AGNES to minimize the mean-squared error (MSE) loss.

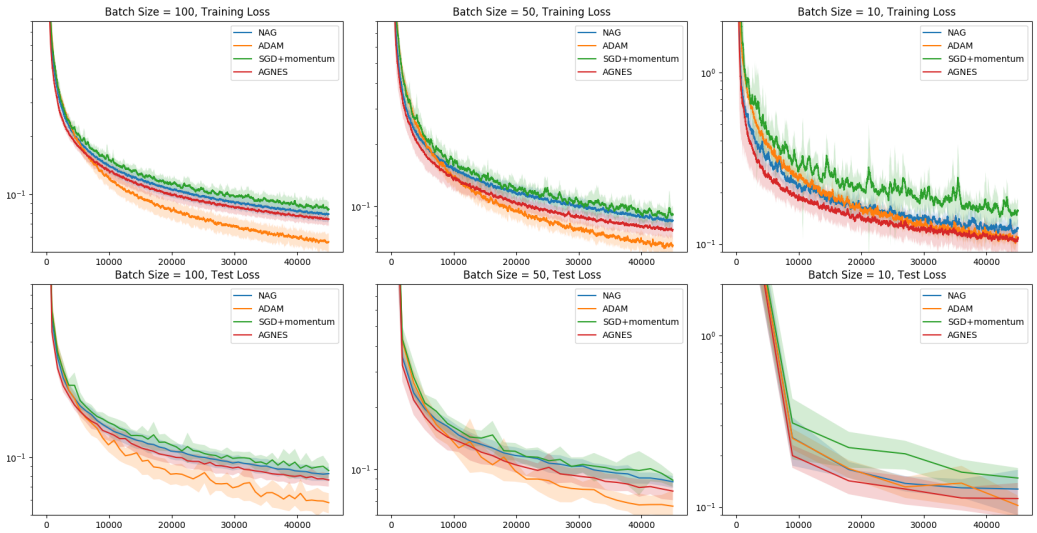

Figure 4: We report the training loss as a running average with decay rate 0.99 (top row) and test loss (bottom row) for batch sizes 100 (left column), 50 (middle column), and 10 (right column) in the setting of Section 5.2. The horizontal axis represents the number of optimizer steps. The performance gap between AGNES and other algorithms widens for smaller batch sizes, where the gradient estimates are more stochastic and the two different parameters α, η add the most benefit.

We selected the learning rate 10^{-3} for Adam as it performed poorly at higher or lower rates 10^{-2} and 10^{-4}. For AGNES, NAG, and SGD, based on initial exploratory experiments, we used a learning rate of 10^{-4}, a momentum value of 0.99, and for AGNES, a correction step size $\eta = 10^{-3}$. The experiment was repeated 10 times each for batch sizes 100, 50, and 10, and run for 45,000 optimizer steps each time. The average loss and standard deviation for each algorithm are reported in Figure 4. The results show that AGNES performs better than SGD and NAG for all batch sizes. With large batch size, Adam performs well with default hyperparameters. The performance of AGNES relative to other algorithms especially improves as the batch size decreases.

5.3 Image classification

We trained ResNet-34 [He et al., 2016] with batch sizes 50 and 10, and ResNet-50 with batch size 50 on the CIFAR-10 image dataset [Krizhevsky et al., 2009] with standard data augmentation (normalization, random crop, and random flip) using Adam, SGD with momentum, NAG, and AGNES. The model implementations were based on [Liu, 2017]. Each algorithm was provided an identically initialized model and the experiment was repeated 5 times for 50 epochs each. The averages and standard deviations of training loss and test accuracy are reported in Figure 5. We used the same initial learning rate 10^{-3} for all the algorithms, which was dropped to 10^{-4} after 25 epochs. A momentum value of 0.99 was used for SGD, NAG, and AGNES and a constant correction step size $\eta = 10^{-2}$ was used for AGNES.

AGNES reliably outperforms SGD and NAG both in terms of training loss and test accuracy. The gap in performance appears to increase as model size increases or batch size decreases, suggesting that AGNES primarily excels in situations where gradients are harder to estimate accurately. For the sake of completeness, we include Adam with default hyperparameters as a comparison.

In congruence with convergence guarantees from convex optimization, grid search suggests that α is the primary learning rate and η should be chosen larger than α. We tried NAG and Adam with higher learning rates 10^{-2} and 10^{-1} as well to ensure a fair comparison with AGNES, but found that they become unstable or perform worse for larger learning rates in our experiments. The AGNES default parameters $\alpha = 10^{-3}, \eta = 10^{-2}, \rho = 0.99$ in Algorithm 1 give consistently strong performance on different models but can be further tuned to improve performance. While the numerical experiments we performed support our theoretical predictions, we acknowledge that our focus lies on theoretical guarantees and we did not test these predictions over a broad set of benchmark problems.

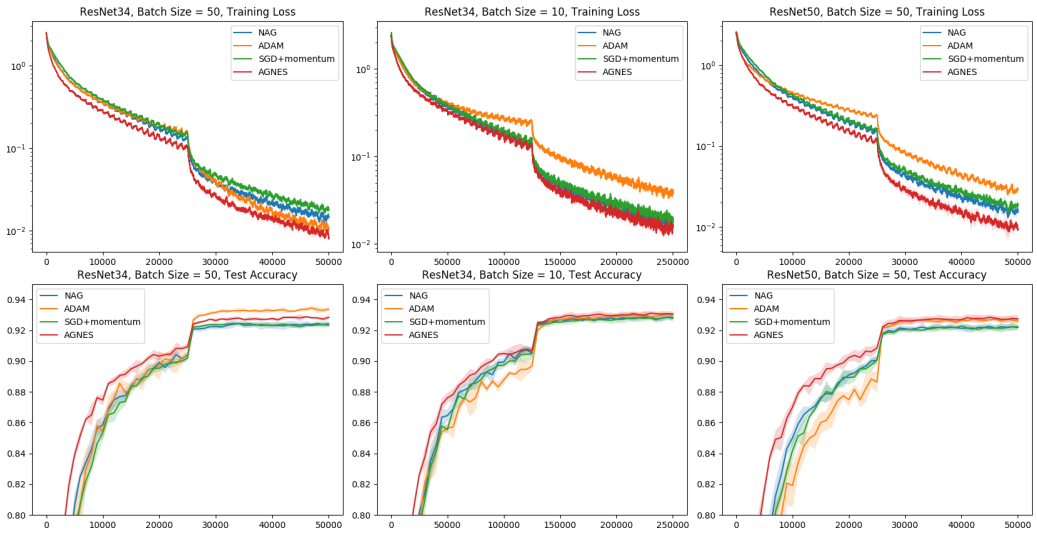

Figure 5: We report the training loss as a running average with decay rate 0.99 (top row) and test accuracy (bottom row) for ResNet-34 trained on CIFAR-10 with batch sizes 50 (left column) and 10 (middle column), and ResNet-50 trained with batch size 50 (right column). The performance of AGNES with the proposed hyperparameters is stable over the changes in model and batch size.

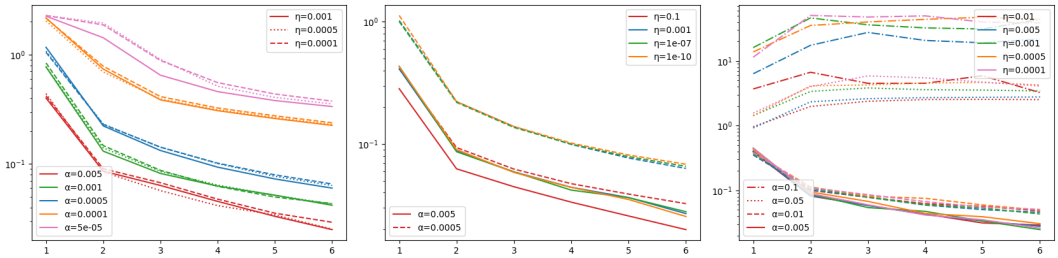

Figure 6: We report the average training loss after each epoch for six epochs for training LeNet-5 on MNIST with AGNES for various combinations of the hyperparameters α and η to illustrate that α is the algorithm's primary learning rate. **Left:** For a given α (color coded), the difference in the trajectory for the three values of η (line style) is marginal. On the other hand, choosing α well significantly affects performance. **Middle:** For any given α, the largest value of η performs much better than the other three values which have near-identical performance. Nevertheless, the worst performing value of η with well chosen $\alpha = 5 \cdot 10^{-3}$ performs better than the best performing value of η with $\alpha = 5 \cdot 10^{-4}$. **Right:** When α is too large, the loss increases irrespective of the value of η.

We present a more thorough comparison of NAG and AGNES with various parameter selections in Figure 8 in Appendix A. With default parameters or minimal parameter tuning, AGNES reliably achieves superior performance compared to NAG (training loss) and smoother curves, suggesting more stable behavior (test accuracy).

5.4 Hyperparameter comparison

We tried various combinations of AGNES hyperparameters α and η to train LeNet-5 on the MNIST dataset to determine which hyperparameter has a greater impact on training. With a fixed batch size of 60 and a momentum value $\rho = 0.99$, we trained independent copies of the model for 6 epochs for each combination of the hyperparameters. The average training loss over the epoch was recorded after each epoch. The results are reported in Figure 6. We see that α has the largest impact on the rate of decay of the loss, which establishes it as the 'primary learning rage'. If α is too small, the algorithm converges slowly and if α is too large, it diverges. If α is chosen correctly, a good choice of the correction step size η (which can be orders of magnitude larger than α) further accelerates convergence, but η cannot compensate for a poor choice of α.

Acknowledgements

Portions of this research were conducted with the advanced computing resources provided by Texas A&M High Performance Research Computing. This research was also supported in part by the University of Pittsburgh Center for Research Computing, RRID:SCR_022735, through the resources provided. Specifically, this work used the H2P cluster, which is supported by NSF award number OAC-2117681.

References

Felipe Alvarez, Hedy Attouch, Jérôme Bolte, and Patrick Redont. A second-order gradient-like dissipative dynamical system with Hessian-driven damping: Application to optimization and mechanics. *Journal de mathématiques pures et appliquées*, 81(8):747–779, 2002.

Hedy Attouch, Zaki Chbani, Jalal Fadili, and Hassan Riahi. First-order optimization algorithms via inertial systems with hessian driven damping. *Mathematical Programming*, pages 1–43, 2022.

J. F. Aujol, Ch. Dossal, and A. Rondepierre. Convergence rates of the heavy-ball method under the łojasiewicz property. *Mathematical Programming*, 2022a. doi: 10.1007/s10107-022-01770-2. URL https://doi.org/10.1007/s10107-022-01770-2.

J.-F. Aujol, Ch. Dossal, and A. Rondepierre. Convergence rates of the heavy ball method for quasi-strongly convex optimization. *SIAM Journal on Optimization*, 32(3):1817–1842, 2022b. doi: 10.1137/21M1403990. URL https://doi.org/10.1137/21M1403990.

Raef Bassily, Mikhail Belkin, and Siyuan Ma. On exponential convergence of SGD in non-convex over-parametrized learning. *CoRR*, abs/1811.02564, 2018. URL http://arxiv.org/abs/1811.02564.

Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009.

Mikhail Belkin. Fit without fear: remarkable mathematical phenomena of deep learning through the prism of interpolation. *Acta Numerica*, 30:203–248, 2021.

Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854, 2019.

Mikhail Belkin, Daniel Hsu, and Ji Xu. Two models of double descent for weak features. *SIAM Journal on Mathematics of Data Science*, 2(4):1167–1180, 2020.

Raphaël Berthier, Francis Bach, Nicolas Flammarion, Pierre Gaillard, and Adrien Taylor. A continuized view on nesterov acceleration. *arXiv preprint arXiv:2102.06035*, 2021.

Raghu Bollapragada, Tyler Chen, and Rachel Ward. On the fast convergence of minibatch heavy ball momentum. *arXiv preprint arXiv:2206.07553*, 2022.

Léon Bottou, Frank E Curtis, and Jorge Nocedal. Optimization methods for large-scale machine learning. *Siam Review*, 60(2):223–311, 2018.

Antonin Chambolle and Charles H Dossal. On the convergence of the iterates of FISTA. *Journal of Optimization Theory and Applications*, 166(3):25, 2015.

Lénaïc Chizat and Francis Bach. On the global convergence of gradient descent for over-parameterized models using optimal transport. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper/2018/file/a1afc58c6ca9540d057299ec3016d726-Paper.pdf.

Y. Cooper. The loss landscape of overparameterized neural networks, 2019. URL https://openreview.net/forum?id=SyezvsC5tX.

Ashok Cutkosky. Anytime online-to-batch, optimism and acceleration. In *International conference on machine learning*, pages 1446–1454. PMLR, 2019.

Marc Dambrine, Ch Dossal, Bénédicte Puig, and Aude Rondepierre. Stochastic differential equations for modeling first order optimization methods. *HAL preprint hal-03630785*, 2022.

Alex Damian, Tengyu Ma, and Jason D. Lee. Label noise SGD provably prefers flat global minimizers. In *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=x2TMPhseWAW.

Jelena Diakonikolas and Michael I Jordan. Generalized momentum-based methods: A hamiltonian perspective. *SIAM Journal on Optimization*, 31(1):915–944, 2021.

Simon S Du, Xiyu Zhai, Barnabas Poczos, and Aarti Singh. Gradient descent provably optimizes over-parameterized neural networks. *arXiv:1810.02054 [cs.LG]*, 2018.

Mathieu Even, Raphaël Berthier, Francis Bach, Nicolas Flammarion, Pierre Gaillard, Hadrien Hendrikx, Laurent Massoulié, and Adrien Taylor. A continuized view on Nesterov acceleration for stochastic gradient descent and randomized gossip. *arXiv preprint arXiv:2106.07644*, 2021.

Alexander Vladimirovich Gasnikov and Yu E Nesterov. Universal method for stochastic composite optimization problems. *Computational Mathematics and Mathematical Physics*, 58:48–64, 2018.

Saeed Ghadimi and Guanghui Lan. Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization i: A generic algorithmic framework. *SIAM Journal on Optimization*, 22(4):1469–1492, 2012.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

Robert Mansel Gower, Nicolas Loizou, Xun Qian, Alibek Sailanbayev, Egor Shulgin, and Peter Richtárik. Sgd: General analysis and improved rates. In *International conference on machine learning*, pages 5200–5209. PMLR, 2019.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

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.

Prateek Jain, Sham M Kakade, Rahul Kidambi, Praneeth Netrapalli, and Aaron Sidford. Accelerating stochastic gradient descent for least squares regression. In *Conference On Learning Theory*, pages 545–604. PMLR, 2018.

Richard Jordan, David Kinderlehrer, and Felix Otto. The variational formulation of the fokker–planck equation. *SIAM Journal on Mathematical Analysis*, 29(1):1–17, 1998. doi: 10.1137/S0036141096303359. URL https://doi.org/10.1137/S0036141096303359.

Pooria Joulani, Anant Raj, Andras Gyorgy, and Csaba Szepesvári. A simpler approach to accelerated optimization: iterative averaging meets optimism. In *International conference on machine learning*, pages 4984–4993. PMLR, 2020.

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

Ali Kavis, Kfir Y Levy, Francis Bach, and Volkan Cevher. Unixgrad: A universal, adaptive algorithm with optimal guarantees for constrained optimization. *Advances in neural information processing systems*, 32, 2019.

Rahul Kidambi, Praneeth Netrapalli, Prateek Jain, and Sham Kakade. On the insufficiency of existing momentum schemes for stochastic optimization. In *2018 Information Theory and Applications Workshop (ITA)*, pages 1–9. IEEE, 2018.

Donghwan Kim and Jeffrey A Fessler. Another look at the fast iterative shrinkage/thresholding algorithm (fista). *SIAM Journal on Optimization*, 28(1):223–250, 2018.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

A. Klenke. *Probability Theory: A Comprehensive Course*. Universitext. Springer London, 2013. ISBN 978-1-4471-5361-0.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Maxime Laborde and Adam Oberman. A lyapunov analysis for accelerated gradient methods: from deterministic to stochastic case. In *International Conference on Artificial Intelligence and Statistics*, pages 602–612. PMLR, 2020.

Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. doi: 10.1109/5.726791.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

Kfir Y Levy, Alp Yurtsever, and Volkan Cevher. Online adaptive methods, universality and acceleration. *Advances in neural information processing systems*, 31, 2018.

Zhiyuan Li, Tianhao Wang, and Sanjeev Arora. What happens after SGD reaches zero loss? –a mathematical framework. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=siCt4xZn5Ve.

Chaoyue Liu and Mikhail Belkin. Accelerating sgd with momentum for over-parameterized learning. *arXiv preprint arXiv:1810.13395*, 2018.

Kuang Liu. Train cifar10 with pytorch. https://github.com/kuangliu/pytorch-cifar, 2017. [Online; accessed 16-May-2024].

Ilya Loshchilov and Frank Hutter. Fixing weight decay regularization in adam, 2018. URL https://openreview.net/forum?id=rk6qdGgCZ.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Michael Muehlebach and Michael Jordan. A dynamical systems perspective on nesterov acceleration. In *International Conference on Machine Learning*, pages 4656–4662. PMLR, 2019.

Arkadi Nemirovski, Anatoli Juditsky, Guanghui Lan, and Alexander Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on optimization*, 19(4):1574–1609, 2009.

Yurii Nesterov. A method for solving the convex programming problem with convergence rate $o(1/k^2)$. *Dokl. Akad. Nauk SSSR*, 269:543–547, 1983.

Yurii Nesterov. Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM Journal on Optimization*, 22(2):341–362, 2012.

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

Mark A Peletier. Variational modelling: Energies, gradient flows, and large deviations. *arXiv preprint arXiv:1402.1990*, 2014.

Boris T Polyak. Some methods of speeding up the convergence of iteration methods. *Ussr computational mathematics and mathematical physics*, 4(5):1–17, 1964.

Herbert Robbins and Sutton Monro. A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407, 1951.

Itay Safran and Ohad Shamir. Spurious local minima are common in two-layer relu neural networks. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 4430–4438. PMLR, 2018. URL http://proceedings.mlr.press/v80/safran18a.html.

Bin Shi, Simon S Du, Weijie Su, and Michael I Jordan. Acceleration via symplectic discretization of high-resolution differential equations. *Advances in Neural Information Processing Systems*, 32, 2019.

Bin Shi, Simon S Du, Michael I Jordan, and Weijie J Su. Understanding the acceleration phenomenon via high-resolution differential equations. *Mathematical Programming*, pages 1–70, 2021.

Jonathan W Siegel. Accelerated first-order methods: Differential equations and Lyapunov functions. *arXiv preprint arXiv:1903.05671*, 2019.

Jonathan W Siegel and Stephan Wojtowytsch. A qualitative difference between gradient flows of convex functions in finite-and infinite-dimensional Hilbert spaces. *arXiv preprint arXiv:2310.17610*, 2023.

Sebastian U. Stich. Unified Optimal Analysis of the (Stochastic) Gradient Method. *arXiv e-prints*, art. arXiv:1907.04232, July 2019. doi: 10.48550/arXiv.1907.04232.

Sebastian U. Stich and Sai Praneeth Karimireddy. The error-feedback framework: Better rates for sgd with delayed gradients and compressed updates. *J. Mach. Learn. Res.*, 21(1), jun 2022. ISSN 1532-4435.

Weijie Su, Stephen Boyd, and Emmanuel Candes. A differential equation for modeling nesterov's accelerated gradient method: theory and insights. *Advances in neural information processing systems*, 27, 2014.

Jaewook J Suh, Gyumin Roh, and Ernest K Ryu. Continuous-time analysis of accelerated gradient methods via conservation laws in dilated coordinate systems. In *International Conference on Machine Learning*, pages 20640–20667. PMLR, 2022.

Ilya Sutskever, James Martens, George Dahl, and Geoffrey Hinton. On the importance of initialization and momentum in deep learning. In *International conference on machine learning*, pages 1139–1147. PMLR, 2013.

Sharan Vaswani, Francis Bach, and Mark Schmidt. Fast and faster convergence of sgd for over-parameterized models and an accelerated perceptron. In *The 22nd international conference on artificial intelligence and statistics*, pages 1195–1204. PMLR, 2019.

Andre Wibisono, Ashia C Wilson, and Michael I Jordan. A variational perspective on accelerated methods in optimization. *proceedings of the National Academy of Sciences*, 113(47):E7351–E7358, 2016.

Ashia C Wilson, Ben Recht, and Michael I Jordan. A lyapunov analysis of accelerated methods in optimization. *The Journal of Machine Learning Research*, 22(1):5040–5073, 2021.

Stephan Wojtowytsch. Stochastic gradient descent with noise of machine learning type. Part II: Continuous time analysis. *arXiv:2106.02588 [cs.LG]*, 2021.

Stephan Wojtowytsch. Stochastic gradient descent with noise of machine learning type. Part I: Discrete time analysis. *Journal of Nonlinear Science*, 33, 2023.

Lei Wu, Mingze Wang, and Weijie J Su. The alignment property of sgd noise and how it helps select flat minima: A stability analysis. In *Advances in Neural Information Processing Systems*, 2022a.

Xiaoxia Wu, Simon S Du, and Rachel Ward. Global convergence of adaptive gradient methods for an over-parameterized neural network. *arXiv preprint arXiv:1902.07111*, 2019.

Xiaoxia Wu, Yuege Xie, Simon Shaolei Du, and Rachel Ward. Adaloss: A computationally-efficient and provably convergent adaptive gradient method. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 8691–8699, 2022b.

Jingzhao Zhang, Aryan Mokhtari, Suvrit Sra, and Ali Jadbabaie. Direct runge-kutta discretization achieves acceleration. *Advances in neural information processing systems*, 31, 2018.

Pan Zhou, Jiashi Feng, Chao Ma, Caiming Xiong, Steven Hoi, and E. Weinan. Towards theoretically understanding why sgd generalizes better than adam in deep learning. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

Appendix

A Additional experiments

A.1 Numerical experiments for AGNES in smooth strongly convex optimization

We compare SGD and AGNES for the family of objective functions

$$f_{\mu,L} : \mathbb{R}^2 \to \mathbb{R}, \qquad f_{\mu,L}(x) = \frac{\mu}{2}\, x_1^2 + \frac{L}{2}\, x_2^2.$$

We considered several stochastic estimators with multiplicative gradient scaling such as

- collinear noise $g = (1 + \sigma N)\nabla f(x)$, where N is one-dimensional standard normal.
- isotropic noise $g = \nabla f(x) + \frac{\sigma \|\nabla f(x)\|}{\sqrt{d}}\, N$, where N is a d-dimensional unit Gaussian.
- Gaussian noise with standard variation $\sigma\|\nabla f(x)\|$ only in the direction orthogonal to $\nabla f(x)$.
- Gaussian noise with standard variation $\sigma\|\nabla f(x)\|$ only in the direction of the fixed vector $v = (1, 1)/\sqrt{2}$.
- Noise of the form $\nabla f(x) + \sqrt{1 + \sigma^2 \|\nabla f(x)\|^2}\, N\, v$ where $v = (1, 1)/\sqrt{2}$ and a variable N which takes values 1 or -1 with probability $\frac{1}{2}\, \frac{\sigma^2 \|\nabla f(x)\|^2}{1+\sigma^2 \|\nabla f(x)\|^2}$ each; $N = 0$ otherwise. In this setting, the noise remains macroscopically large at the global minimum, but the probability of encountering noise becomes small.

Numerical results were found to be comparable for all settings on a long time-scale, but the geometry of trajectories may change in the early stages of optimization depending on the noise structure.

For collinear and isotropic noise, the results obtained for $f_{\mu,L}$ on $\mathbb{R}^2$ were furthermore found comparable (albeit not identical) to simulations with a quadratic form on $\mathbb{R}^d$ with $d = 10$ and

- $(d-1)$ eigenvalues $= \mu$ and one eigenvalue $= L$
- $(d-1)$ eigenvalues $= L$ and one eigenvalue $= \mu$
- eigenvalues equi-spaced between μ and L.

The evolution of $\mathbb{E}[f(x_n)]$ for different values of σ and $L \geq \mu \equiv 1$ is considered for both SGD and AGNES in Figure 7.

The objective functions are $\mu = 1$-convex and L-smooth. We use the optimal parameters α, η, ρ derived above for AGNES and the optimal step size $\eta = \frac{1}{L(1+\sigma^2)}$ for SGD. The mean of the objective value at each iteration is computed over 1,000 samples for each optimizer.

A.2 Extensively comparing against NAG

We ran additional experiments testing a much wider range of hyperparameters for NAG for the task of classifying images from CIFAR-10. The results, presented in Figure 8 indicate that AGNES outperforms NAG with little fine-tuning of the hyperparameters.

We trained ResNet-34 using batch size of 50 for 40 epochs using NAG with learning rate in $\{8 \cdot 10^{-5}, 10^{-4}, 2 \cdot 10^{-4}, 5 \cdot 10^{-4}, 8 \cdot 10^{-4}, 10^{-3}, 2 \cdot 10^{-3}, 5 \cdot 10^{-3}, 8 \cdot 10^{-3}, 10^{-2}, 2 \cdot 10^{-2}, 5 \cdot 10^{-2}, 8 \cdot 10^{-2}, 10^{-1}, 2 \cdot 10^{-1}, 5 \cdot 10^{-1}\}$ and momentum value in $\{0.2, 0.5, 0.8, 0.9, 0.99\}$. These 80 combinations of hyperparameters for NAG were compared against AGNES with the default hyperparameters suggested $\alpha = 10^{-3}$ (learning rate), $\eta = 10^{-2}$ (correction step), and $\rho = 0.99$ (momentum) as well as AGNES with a slightly smaller learning rate $5 \cdot 10^{-4}$ (with the other two hyperparameters being the same).

AGNES consistently achieved a lower training loss as well as a better test accuracy faster than any combination of NAG hyperparameters tested. The same random seed was used each time to ensure a fair comparison between the optimizers. Overall, AGNES remained more stable and while other versions of NAG occasionally achieved a higher classification accuracy in certain epochs.

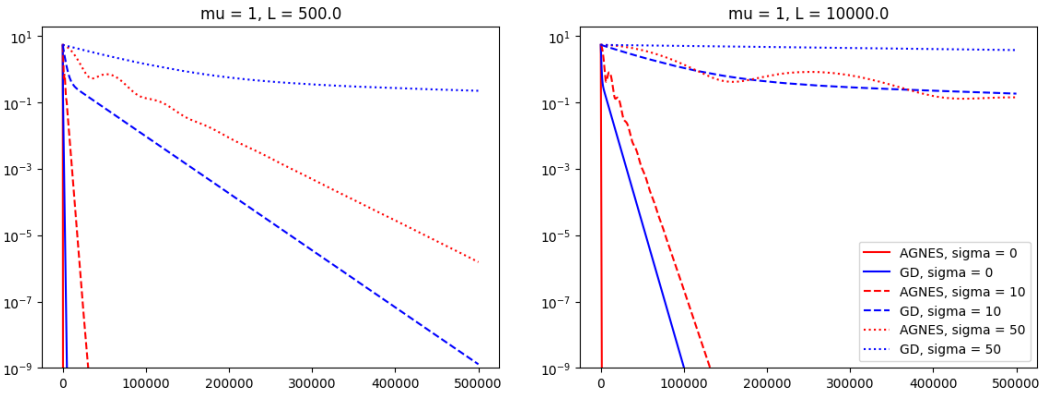

Figure 7: We compare AGNES (red) and SGD (blue) for the optimization of $f_{\mu,L}$ with $\mu = 1$ and $L = 500$ (left) / $L = 10^4$ (right) for different noise levels $\sigma = 0$ (solid line), $\sigma = 10$ (dashed) and $\sigma = 50$ (dotted). In all cases, AGNES improves significantly upon SGD. The noise model used is isotropic Gaussian, but comparable results are obtained for different versions of multiplicatively scaling noise.

B Multiple versions of the AGNES scheme

B.1 Equivalence of the two formulations of AGNES

Lemma 9. *The two formulations of the AGNES time-stepping scheme (2) and (3) produce the same sequence of points.*

Proof. We consider the three-step formulation (3),

$$v_0 = 0, \quad x'_n = x_n + \alpha v_n, \qquad x_{n+1} = x'_n - \eta g'_n, \qquad v_{n+1} = \rho_n(v_n - g'_n),$$

and use it to derive (2) by eliminating the velocity variable v_n. If $v_0 = 0$, then $x'_0 = x_0$. From the definition x'_n, we get $\alpha v_n = x'_n - x_n$. Substituting this into the definition of v_{n+1},

$$v_{n+1} = \rho_n \left(\frac{x'_n - x_n}{\alpha} - g'_n \right).$$

Then using this expression for v_{n+1} to compute x'_{n+1},

$$\begin{aligned} x'_{n+1} &= x_{n+1} + \alpha v_{n+1} \\ &= x_{n+1} + \alpha\rho_n \left(\frac{x'_n - x_n}{\alpha} - g'_n \right) \\ &= x_{n+1} + \rho_n(x'_n - \alpha g'_n - x_n). \end{aligned}$$

Together with the definition of x_{n+1} and the initialization $x'_0 = x_0$, this is exactly the two-step formulation (2) of AGNES. □

B.2 Equivalence of AGNES and MaSS

After the completion of this work, we learned of Liu and Belkin [2018]'s Momentum-Added Stochastic Solver (MaSS) method, which generates sequences according to the iteration

$$x_{n+1} = x'_n - \eta_1 g'_n, \qquad x'_{n+1} = (1+\gamma)x_{n+1} - \gamma x_n + \eta_2 g'_n$$

where g'_n is an estimate for $\nabla f(x'_n)$. This is a version of AGNES with the choice $\eta = \eta_1$ and the momentum step

$$\begin{aligned} x'_{n+1} &= x_{n+1} + \rho(x'_n - \alpha g'_n - x_n) \\ &= x_{n+1} + \rho(x_{n+1} + \eta g'_n - \alpha g'_n - x_n) \\ &= (1+\rho)x_{n+1} - \rho x_n + (\eta - \alpha)\rho g'_n, \end{aligned}$$

i.e. MaSS coincides with AGNES for $\gamma = \rho$ and $\eta_2 = (\eta - \alpha)\rho$.

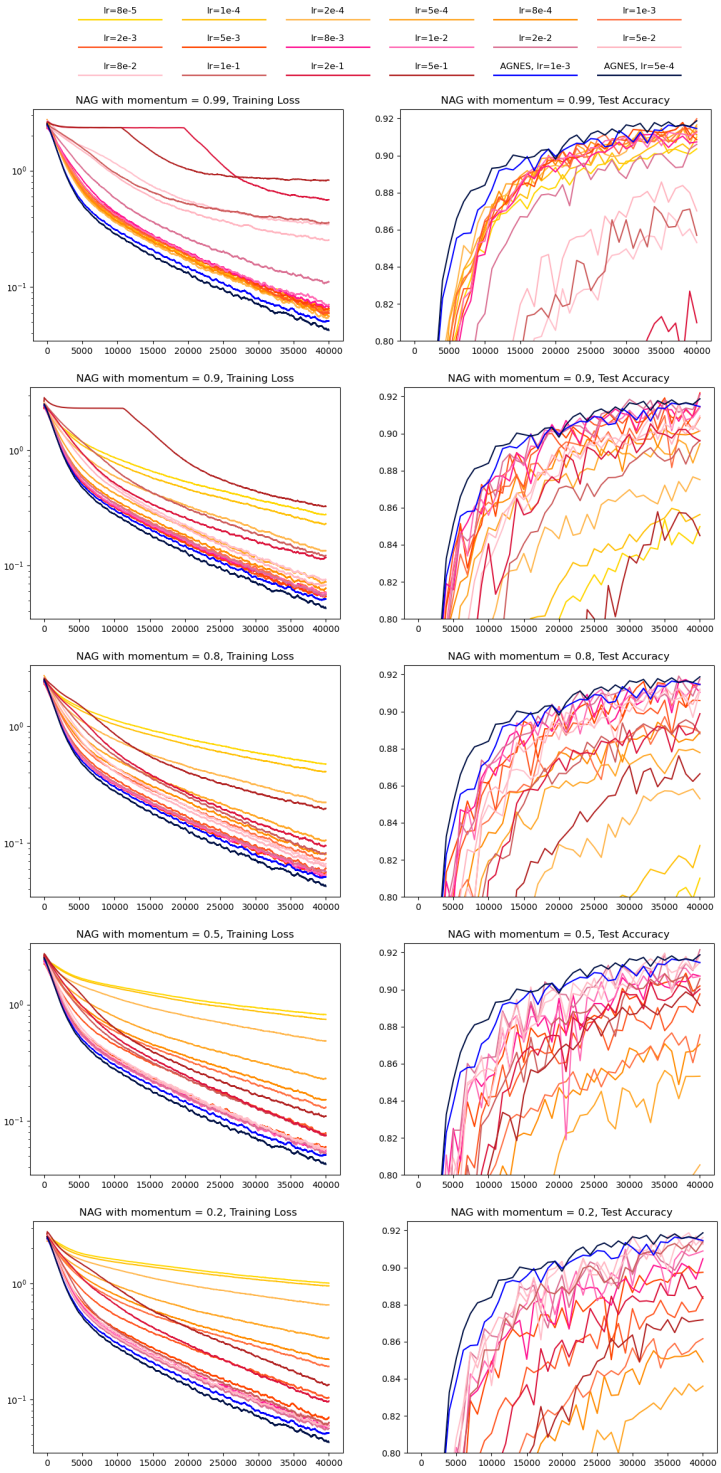

Figure 8: We trained ResNet34 on CIFAR-10 with batch size 50 for 40 epochs using NAG. Training losses are reported as a running average with decay rate 0.999 in the left column and test accuracy after every epoch is reported in the right column. Each row represents a specific value of momentum used for NAG (from top to bottom: 0.99, 0.9, 0.8, 0.5, and 0.2) with learning rates ranging from $8 \cdot 10^{-5}$ to 0.5. These hyperparameter choices for NAG were compared against AGNES with the default hyperparameters suggested $\alpha = 10^{-3}$ (learning rate), $\eta = 10^{-2}$ (correction step), and $\rho = 0.99$ (momentum) as well as AGNES with a slightly smaller learning rate $5 \cdot 10^{-4}$ (with $\rho = 0.99$, $\eta = 10^{-2}$ as well). The same two training trajectories with AGNES are shown in all the plots in shades of blue. The horizontal axes represent the number of optimizer steps.

C Continuous time interpretation of AGNES

For better interpretability, we consider the continuous time limit of the AGNES algorithm. Similar ODE analyses of accelerated first order methods have been considered by many authors, including Su et al. [2014], Siegel [2019], Wilson et al. [2021], Attouch et al. [2022], Aujol et al. [2022a], Zhang et al. [2018], Dambrine et al. [2022].

Consider the time-stepping scheme

$$v_0 = 0, \quad x_n' = x_n + \gamma_1 v_n, \qquad x_{n+1} = x_n' - \eta g_n', \qquad v_{n+1} = \rho_n(v_n - \gamma_2 g_n'), \tag{5}$$

which reduces to AGNES as in (3) with the choice of parameters $\gamma_1 = \alpha, \gamma_2 = 1$. For the derivation of continuous time dynamics, we show that the same scheme arises with the choice $\gamma_1 = \gamma_2 = \sqrt{\alpha}$.

Lemma 10. *Let $\rho \in (0,1)$ and $\eta > 0$ parameters. Assume that γ_1, γ_2 and $\tilde\gamma_1, \tilde\gamma_2$ are parameters such that $\tilde\gamma_1\tilde\gamma_2 = \gamma_1\gamma_2$. Consider the sequences $(\tilde x_n, \tilde x_n', \tilde v_n)$ and (x_n, x_n', v_n) generated by the time stepping scheme (5) with parameters $(\rho, \eta, \tilde\gamma_1, \tilde\gamma_2)$ and $(\rho, \eta, \gamma_1, \gamma_2)$ respectively. If $x_0 = \tilde x_0$ and $\gamma_1 v_0 = \tilde\gamma_1\, \tilde v_0$, then $x_n = \tilde x_n$, $x_n' = \tilde x_n'$ and $\tilde\gamma_1\, \tilde v_n = \gamma_1\, v_n$ for all $n \in \mathbb{N}$.*

Proof. We proceed by mathematical induction on n. For $n = 0$, the claim holds by the hypotheses of the lemma. For the inductive hypothesis, we suppose that $x_n = \tilde x_n$ and $\gamma_1 v_n = \tilde\gamma_1 \tilde v_n$ and prove the claim for $n+1$. Note that since $x_n' = x_n + \gamma_1 v_n$, it automatically follows that $x_n' = \tilde x_n'$. This implies that

$$x_{n+1} = x_n' - \eta g_n' = \tilde x_n' - \eta g_n' = \tilde x_{n+1}.$$

Considering the velocity term,

$$\gamma_1 v_{n+1} = \rho_n(\gamma_1 v_n - \gamma_1\gamma_2 g_n') = \rho_n(\tilde\gamma_1\tilde v_n - \tilde\gamma_1\tilde\gamma_2 g_n') = \tilde\gamma_1\rho_n(\tilde v_n - \tilde\gamma_2 g_n') = \tilde\gamma_1\tilde v_{n+1}.$$

Thus $x_{n+1} = \tilde x_{n+1}$ and $\gamma_1 v_{n+1} = \tilde\gamma_1\tilde v_{n+1}$. The induction can therefore be continued. □

Consider the choice of parameters in Theorem 4 by

$$\eta = \frac{1}{L(1+\sigma^2)}, \quad \alpha = \frac{1-\sqrt{\mu/L}}{1-\sqrt{\mu/L}+\sigma^2}\eta \approx \frac{\eta}{1+\sigma^2}, \quad \rho = \frac{\sqrt{L}(1+\sigma^2)-\sqrt{\mu}}{\sqrt{L}(1+\sigma^2)+\sqrt{\mu}}$$

if $\mu \ll L$. We denote $h := \frac{1}{\sqrt{L}(1+\sigma^2)}$ and note that

$$\gamma_1 = \gamma_2 = \sqrt{\alpha} \approx h, \qquad \eta = (1+\sigma^2)h^2 = \frac{h}{\sqrt{L}}$$

and

$$\rho = 1 - 2\frac{\sqrt{\mu}}{\sqrt{L}(1+\sigma^2)+\sqrt{\mu}} = 1 - 2\sqrt{\mu}\,\frac{\sqrt{L}(1+\sigma^2)}{\sqrt{L}(1+\sigma^2)+\sqrt{\mu}}h \approx 1 - 2\sqrt{\mu}h.$$

Depending on which interpretation of η we select, we obtain a different continuous time limit. First, consider the deterministic case $\sigma = 0$. Then

$$\begin{aligned}\begin{pmatrix} x_{n+1} \\ v_{n+1}\end{pmatrix} &= \begin{pmatrix} x_n \\ v_n\end{pmatrix} + h\begin{pmatrix} v_n - h\nabla f(x_n + hv_n) \\ -2\sqrt{\mu}\, v_n - (1-\sqrt{\mu}h)\nabla f(x_n + hv_n)\end{pmatrix} \\ &= \begin{pmatrix} x_n \\ v_n\end{pmatrix} + h\begin{pmatrix} v_n \\ -2\sqrt{\mu}\, v_n - \nabla f(x_n)\end{pmatrix} + O(h^2)\end{aligned}$$

Keeping f fixed and taking $h \to 0$, this is a time-stepping scheme for the coupled ODE system

$$\begin{pmatrix} \dot x \\ \dot v\end{pmatrix} = \begin{pmatrix} v \\ -2\sqrt{\mu}\, v - \nabla f(x)\end{pmatrix}.$$

Differentiating the first equation and using the system ODEs to subsequently eliminate v from the expression, we observe that

$$\ddot x = \dot v = -2\sqrt{\mu}\, v - \nabla f(x) = -2\sqrt{\mu}\,\dot x - \nabla f(x),$$

i.e. we recover the heavy ball ODE. The alternative interpretation $\eta = h/\sqrt{L}$ can be analized equivalently and leads to a system

$$\begin{pmatrix}\dot{x}\\ \dot{v}\end{pmatrix} = \begin{pmatrix} v - \frac{1}{\sqrt{L}}\nabla f(x) \\ -2\sqrt{\mu}\, v - \nabla f(x)\end{pmatrix}.$$

which corresponds to a second order ODE

$$\begin{aligned}\ddot{x} &= -\dot{v} - \frac{1}{\sqrt{L}} D^2 f(x)\,\dot{x} \\ &= -2\sqrt{\mu} v - \nabla f(x) - \frac{1}{\sqrt{L}} D^2 f(x)\,\dot{x} \\ &= -2\sqrt{\mu}\left(\dot{x} + \frac{1}{\sqrt{L}}\nabla f(x)\right) - \nabla f(x) - \frac{1}{\sqrt{L}} D^2 f(x)\,\dot{x} \\ &= -\left(2\sqrt{\mu}\, I_{m\times m} + \frac{1}{\sqrt{L}}\, D^2 f(x)\right)\dot{x} - \left(1 + 2\sqrt{\frac{\mu}{L}}\right)\nabla f(x).\end{aligned}$$

This continuum limit is not a simple heavy-ball ODE, but rather a system with adaptive friction modelled by Hessian damping. A related Newton/heavy ball hybrid dynamical system was studied in greater detail by Alvarez et al. [2002]. For L-smooth functions, the ℓ^2-operator norm of $D^2 f(x)$ satisfies $\|D^2 f(x)\| \leq L$, i.e. the additional friction term can be as large as $\sqrt{L}$ in directions corresponding to high eigenvalues of the Hessian. This provides significant regularization in directions which would otherwise be notably underdamped.

Following Appendix F.3, we maintain that the scaling

$$\frac{\eta(1-\rho)}{\alpha} = 2\sqrt{\frac{\mu}{L}}$$

is 'natural' as we vary η, α, ρ. The same fixed ratio is maintained for the scaling choice $\eta = h/\sqrt{L}$ as

$$\frac{\eta(1-\rho)}{\alpha} = \frac{h/\sqrt{L}\cdot 2\sqrt{\mu}\, h}{h^2} = 2\sqrt{\frac{\mu}{L}}.$$

Indeed, numerical experiments in Section A suggest that such regularization may be observed in practice as high eigenvalues in the quadratic map do not exhibit 'underdamped' behavior. We therefore believe that Hessian dampening is the potentially more instructive continuum description of AGNES. A similar analysis can be conducted in the stochastic case with the scaling

$$\gamma_1 = \gamma_2 = h, \quad \eta \in \left\{(1+\sigma^2)h^2,\ \frac{h}{\sqrt{L}}\right\}, \quad \rho = 1 - 2\sqrt{\mu}h$$

for large σ. We incorporate noise as

$$g'_n = (1 + \sigma N_n)\nabla f(x'_n)$$

and write

$$\begin{aligned}\begin{pmatrix}x_{n+1}\\ v_{n+1}\end{pmatrix} &= \begin{pmatrix}x_n\\ v_n\end{pmatrix} + h\begin{pmatrix} v_n - (1+\sigma^2)h\nabla g'_n \\ -2\sqrt{\mu}\, v_n - (1-\sqrt{\mu}h)\big(1+\sigma N_n\big)\nabla f(x_n + h v_n)\end{pmatrix} \\ &= \begin{pmatrix}x_n\\ v_n\end{pmatrix} + h\begin{pmatrix} v_n \\ -2\sqrt{\mu}\, v_n - \nabla f(x_n) - \sqrt{h}\,\frac{\sigma\sqrt{h}}{2}\, N_n \nabla f(x)\end{pmatrix} + O(h^2)\end{aligned}$$

which can be viewed as an approximation of the coupled ODE/SDE system

$$\begin{pmatrix}\mathrm{d}x\\ \mathrm{d}v\end{pmatrix} = \begin{pmatrix} v\,\mathrm{d}t \\ \big(-2\sqrt{\mu}\, v - \nabla f(x)\big)\,\mathrm{d}t + \sigma\sqrt{h}\,\mathrm{d}B\cdot\nabla f(x)\end{pmatrix}$$

under moment bounds on the noise N_n. The precise noise type depends on the assumptions on the covariance structure of N_n – noise can point only in gradient direction or be isotropic on the entire space. For small h, the dynamics become deterministic. Again, an alternative continuous time limit is

$$\begin{pmatrix}\mathrm{d}x\\ \mathrm{d}v\end{pmatrix} = \begin{pmatrix} (v - \nabla f(x)/\sqrt{L})\,\mathrm{d}t + \frac{\sigma\sqrt{h}}{\sqrt{L}}\,\mathrm{d}B\cdot\nabla f(x) \\ \big(-2\sqrt{\mu}\, v - \nabla f(x)\big)\,\mathrm{d}t + \sigma\sqrt{h}\,\mathrm{d}B\cdot\nabla f(x)\end{pmatrix}$$

if η is scaled towards zero as $h/\sqrt{L}$. The first limiting structure is recoverd in the limit $L \to \infty$. Notably, the noise in the first equation is expected to be non-negligible if $\sigma \gg \sqrt{L}$. A similar analysis can be conducted in the convex case, noting that

$$\frac{n+n_0}{n+n_0+3} = 1 - \frac{3}{n+n_0+3} = 1 - \frac{3}{(n+n_0+3)h}\,h$$

where $(n+n_0+3)h$ roughly corresponds to the time t in the continuous time setting.

D Background material and auxiliary results

In this appendix, we gather a few auxiliary results that will be used in the proofs below. We believe that these will be familiar to the experts and can be skipped by experienced readers.

D.1 A brief review of L-smoothness and (strong) convexity

Recall that if a function f is L-smooth, then

$$f(y) \le f(x) + \nabla f(x)\cdot(y-x) + \frac{L}{2}\|x-y\|^2. \tag{6}$$

For convex functions, this is in fact equivalent to ∇f being L-Lipschitz.

Lemma 11. *If f is convex and differentiable and satisfies (6), then $\|\nabla f(x) - \nabla f(y)\| \le L\|x-y\|$ for all x and y.*

Proof. Setting $y = x - \frac{1}{L}\nabla f(x)$ in (6) implies that $f(x) - \inf_z f(z) \ge \frac{1}{2L}\|\nabla f(x)\|^2$. Applying this to the modified function $f_y(x) = f(x) - \nabla f(y)\cdot(x-y)$, which is still convex and satifies (6), we get

$$\begin{aligned} f_y(x) - \inf_z f_y(z) = f_y(x) - f_y(y) &= f(x) - \nabla f(y)\cdot(x-y) - f(y) \\ &\ge \frac{1}{2L}\|\nabla f_y(x)\|^2 = \frac{1}{2L}\|\nabla f(x) - \nabla f(y)\|^2. \end{aligned}$$

Note that here we have used the convexity to conclude that $\inf_z f_y(z) = f_y(y)$, i.e. that f_y is minimized at y, since by construction $\nabla f_y(y) = 0$ (this is the only place where we use convexity!). Swapping the role of x and y, adding these inequalities, and applying Cauchy-Schwartz we get

$$\frac{1}{L}\|\nabla f(x) - \nabla f(y)\|^2 \le (\nabla f(x) - \nabla f(y))\cdot(x-y) \le \|\nabla f(x) - \nabla f(y)\|\|x-y\|,$$

which implies the result. □

From the first order strong convexity condition,

$$f(y) \ge f(x) + \nabla f(x)\cdot(y-x) + \frac{\mu}{2}\|x-y\|^2,$$

we deduce the more useful formulation $\nabla f(x)\cdot(x-y) \ge f(x) - f(y) + \frac{\mu}{2}\|x-y\|^2$. The convex case arises as the special case $\mu = 0$. We note a special case of these conditions when one of the points is a minimizer of f.

Lemma 12. *If f is an L-smooth function and x^* is a point such that $f(x^*) = \inf_{x\in\mathbb{R}^m} f(x)$ then for any $x \in \mathbb{R}^m$,*

$$f(x) - f(x^*) \le \frac{L}{2}\|x - x^*\|^2.$$

Similarly, if f is differentiable and μ-strongly convex then for any $x \in \mathbb{R}^m$,

$$\frac{\mu}{2}\|x - x^*\|^2 \le f(x) - f(x^*).$$

Proof. This follows from the two first order conditions stated above by noting that $\nabla f(x^*) = 0$ if x^* is a minimizer of f. □

Additionally, L-smooth functions which are bounded from below satisfy the inequality

$$\|\nabla f\|^2 \leq 2L\,(f - \inf f).$$

Intuitively, if the gradient is large at a point, then we reduce f quickly by walking in the gradient direction. The L-smoothness condition prevents the gradient from decreasing quickly along our path. Thus if the gradient is larger than a threshold at a point where f is close to $\inf f$, then the inequality $f \geq \inf f$ would be violated.

Let us record a modified gradient descent estimate, which is used only in the non-convex case. The difference to the usual estimate is that the gradient is evaluated at the terminal point of the interval rather than the initial point.

Lemma 13. *For any x, v and α: If f is L-smooth, then*

$$f(x + \alpha v) \leq f(x) + \alpha \nabla f(x + \alpha v) \cdot v + \frac{L\alpha^2}{2}\|v\|^2.$$

Note that if f is convex, this follows immediately from (6) and the convexity condition $(\nabla f(y) - \nabla f(x)) \cdot (y - x) \geq 0$.

Proof. The proof is essentially identical to the standard decay estimate. We compute

$$\begin{aligned}
f(x) &= f(x + \alpha v) - \int_0^\alpha \frac{d}{dt} f(x + tv)\mathrm{d}t \\
&= f(x + \alpha v) - \int_0^\alpha \left[\nabla f(x + \alpha v) + \left\{\nabla f(x + tv) - \nabla f(x + \alpha v)\right\}\right] \cdot v \,\mathrm{d}t \\
&\geq f(x + \alpha v) - \nabla f(x + \alpha v) \cdot v - \int_0^\alpha L\,(\alpha - t)\|v\|^2 \,\mathrm{d}t \\
&= f(x + \alpha v) - \alpha\, \nabla f(x + \alpha v) \cdot v - \frac{L\alpha^2}{2}\|v\|^2.
\end{aligned}$$

□

D.2 Stochastic processes, conditional expectations, and a decrease property for SGD

Now, we turn towards a very brief review of the stochastic process theory used in the analysis of gradient descent type algorithms. Recall that $(\Omega, \mathcal{A}, \mathbb{P})$ is a probablity space from which we draw elements ω_n for gradient estimates $g(x'_n, \omega_n)$ (AGNES) or $g(x_n, \omega_n)$ (SGD). We consider x_0 as a random variable on $\mathbb{R}^m$ with law $\mathbb{Q}$. Let us introduce the probability space $(\widehat{\Omega}, \widehat{\mathcal{A}}, \widehat{\mathbb{P}})$ where

1. $\widehat{\Omega} = \mathbb{R}^d \times \prod_{n\in\mathbb{N}} \Omega$,
2. $\widehat{\mathcal{A}}$ is the cylindrical/product σ-algebra on $\widehat{\Omega}$, and
3. $\widehat{\mathbb{P}} = \mathbb{Q} \times \bigotimes \mathbb{P}$.

The product σ-algebra and product measure are objects suited to events which are defined using only *finitely many* variables in the product space. A more detailed introduction can be found in [Klenke, 2013, Example 1.63]. We furthermore define the filtration $\{\mathcal{F}_n\}_{n\in\mathbb{N}}$ where $\mathcal{F}_n$ is the σ-algebra generated by sets of the form

$$B \times \prod_{i=1}^n A_i \times \prod_{i\in\mathbb{N}} \Omega, \qquad B \subseteq \mathbb{R}^m \text{ Borel}, \quad A_i \in \mathcal{A}.$$

In particular, $\bigcup_{n\in\mathbb{N}} \mathcal{F}_n \subseteq \sigma\left(\bigcup_{n\in\mathbb{N}} \mathcal{F}_n\right) = \widehat{\mathcal{A}}$ and, examining the time-stepping scheme, it is immediately apparent that x_n, x'_n, v_n are $\mathcal{F}_n$-measurable random variables on $\widehat{\Omega}$. In particular, they are $\mathcal{A}$-measurable. Alternatively, we can consider $\mathcal{F}_n$ as the σ-algebra generated by the random variables $x_1, x'_1, \ldots, x_n, x'_n$, i.e. all the information that is known after intialization and taking n gradient steps. All probabilities in the main article are with respect to $\widehat{\mathbb{P}}$.

Recall that conditional expectations are a technical tool to capture the stochasticity in a random variable X which can be predicted from another random quantity Y. This allows us to quantify the

randomness in the gradient estimators g'_n which comes from the fact that x_n is a random variable (not known ahead of time) and which randomness comes from the fact that on top of the inherent randomness due to e.g. initialization, we do not compute exact gradients. In particular, even at run time when x_n is known, there is additional noise in the estimators g'_n in our setting due to the selection of ω_n.

In the next Lemma, we recall two important properties of conditional expectations.

Lemma 14. *[Klenke, 2013, Theorem 8.14] Let g and h be $\mathcal{A}$-measurable random variables on a probability space $(\Omega, \mathcal{A}, \mathbb{P})$ and $\mathcal{F} \subseteq \mathcal{A}$ be a $\sigma-$algebra. Then the conditional expectations $\mathbb{E}[g \mid \mathcal{F}]$ and $\mathbb{E}[h \mid \mathcal{F}]$ satisfy the following properties:*

1. *(linearity) $\mathbb{E}[\alpha g + \beta h \mid \mathcal{F}] = \alpha\,\mathbb{E}[g] + \beta\,\mathbb{E}[h \mid \mathcal{F}]$ for all $\alpha, \beta \in \mathbb{R}$*
2. *(tower identity) $\mathbb{E}[\mathbb{E}[g \mid \mathcal{F}]] = \mathbb{E}[g]$*
3. *If g is $\mathcal{F}-$measurable then $\mathbb{E}[gh \mid \mathcal{F}] = g\,\mathbb{E}[h \mid \mathcal{F}]$. In particular, $\mathbb{E}[g \mid \mathcal{F}] = g$*

For a more thorough introduction to filtrations and conditional expectations, see e.g. [Klenke, 2013, Chapter 8]. $\mathbb{E}[g'_n|\mathcal{F}_n]$ is the mean of g'_n if all previous steps are already known.

Lemma 15. *Suppose g'_n, x_n, and x'_n satisfy the assumptions laid out in Section 3.1, then the following statements hold*

1. $\mathbb{E}\big[g'_n|\mathcal{F}_n\big] = \nabla f(x'_n)$
2. $\mathbb{E}\big[\|g'_n - \nabla f(x'_n)\|^2\big] \leq \sigma^2\,\mathbb{E}\big[\|\nabla f(x'_n)\|^2\big]$.
3. $\mathbb{E}[\|g'_n\|^2] = (1+\sigma^2)\mathbb{E}[\|\nabla f(x'_n)\|^2]$
4. $\mathbb{E}[\nabla f(x'_n) \cdot g'_n] = \mathbb{E}[\|\nabla f(x'_n)\|^2]$

Proof. **First and second claim.** This follows from Fubini's theorem.

Third claim. The third result then follows by an application of the tower identity with $\mathcal{F}_n$, expanding the square of the norm as a dot product, and then using the linearity of conditional expectation:

$$\begin{aligned}
\mathbb{E}\left[\|g'_n\|^2\right] &= \mathbb{E}\left[\mathbb{E}\left[\|g'_n\|^2 \mid \mathcal{F}_n\right]\right] \\
&= \mathbb{E}\left[\mathbb{E}\left[\|g'_n - \nabla f(x'_n)\|^2 + 2g \cdot \nabla f(x'_n) - \|\nabla f(x'_n)\|^2 \mid \mathcal{F}_n\right]\right] \\
&= \mathbb{E}\left[\mathbb{E}\left[\|g'_n - \nabla f(x'_n)\|^2 \mid \mathcal{F}_n\right] + 2\mathbb{E}\left[g \cdot \nabla f(x'_n) \mid \mathcal{F}_n\right] - \mathbb{E}\left[\|\nabla f(x'_n)\|^2 \mid \mathcal{F}_n\right]\right] \\
&\leq \mathbb{E}\left[\sigma^2 \nabla f(x'_n) + 2\|\nabla f(x'_n)\|^2 - \|\nabla f(x'_n)\|^2\right] \\
&= (1+\sigma^2)\mathbb{E}\left[\|\nabla f(x'_n)\|^2\right].
\end{aligned}$$

Fourth claim. For the fourth result, we observe that since f is a deterministic function and x'_n is $\mathcal{F}_n$-measurable, $\nabla f(x'_n)$ is also measurable with respect to the σ-algebra. Then using the tower identity followed by the third property in Lemma 14,

$$\begin{aligned}
\mathbb{E}\left[\nabla f(x'_n) \cdot g'_n\right] &= \mathbb{E}\left[\mathbb{E}\left[\nabla f(x'_n) \cdot g'_n \mid \mathcal{F}_n\right]\right] \\
&= \mathbb{E}\left[\nabla f(x'_n) \cdot \mathbb{E}\left[g'_n \mid \mathcal{F}_n\right]\right] \\
&= \mathbb{E}\left[\nabla f(x'_n) \cdot \nabla f(x'_n)\right] \\
&= \mathbb{E}\left[\|\nabla f(x'_n)\|^2\right].
\end{aligned}$$

□

As a consequence, we note the following decrease estimate.

Lemma 16. *Suppose that f, x'_n, and $g'_n = g(x'_n, \omega_n)$ satisfy the conditions laid out in Section 3.1, then*

$$\mathbb{E}\big[f(x'_n - \eta g'_n)\big] \leq \mathbb{E}\big[f(x'_n)\big] - \eta\left(1 - \frac{L(1+\sigma^2)\eta}{2}\right)\mathbb{E}\left[\|\nabla f(x'_n)\|^2\right].$$

Proof. Using L-smoothness of f,

$$f(x_n' - \eta g_n') \leq f(x_n') - \eta g_n' \cdot \nabla f(x_n') + \frac{L\eta^2}{2} \|g_n'\|^2 .$$

Then taking the expectation and using the results of the previous lemma,

$$\begin{aligned}\mathbb{E}\left[f(x_n' - \eta g_n')\right] &\leq \mathbb{E}[f(x_n')] - \eta \mathbb{E}\left[\|\nabla f(x_n')\|^2\right] + \frac{L\eta^2}{2}(1+\sigma^2)\mathbb{E}\left[\|\nabla f(x_n')\|^2\right] \\ &\leq \mathbb{E}\big[f(x_n')\big] - \eta\left(1 - \frac{L(1+\sigma^2)\eta}{2}\right)\mathbb{E}\left[\|\nabla f(x_n')\|^2\right]\end{aligned}$$

□

In particular, if $\eta \leq \frac{1}{L(1+\sigma^2)}$, then

$$\mathbb{E}\big[f(x_n' - \eta g_n')\big] \leq \mathbb{E}\big[f(x_n')\big] - \frac{\eta}{2}\mathbb{E}\left[\|\nabla f(x_n')\|^2\right].$$

E Convergence proofs: convex case

E.1 Gradient Descent (GD)

We first present a convergence result for stochastic gradient descent for convex functions with multiplicative noise scaling. To the best of our knowledge, convergence proofs for this type of noise which degenerates at the global minimum have been given by Bassily et al. [2018], Wojtowytsch [2023] under a Polyak-Lojasiewicz (or PL) condition (which holds automatically in the strongly convex case), but not for functions which are merely convex. We note that, much like AGNES, SGD achieves the same rate of convergence in stochastic convex optimization with multiplicative noise as in the deterministic case (albeit with a generally much larger constant). In particular, SGD with multiplicative noise is more similar to deterministic gradient descent than to SGD with additive noise in this way.

Analyses of SGD with non-standard noise under various conditions are given by Stich and Karimireddy [2022], Stich [2019].

Theorem 17 (GD, convex case)**.** *Assume that f is a convex function and that the assumptions laid out in Section 3.1 are satisfied. If the sequence x_n is generated by the gradient descent scheme*

$$g_n = g(x_n, \omega_n), \qquad x_{n+1} = x_n - \eta g_n, \qquad \eta \leq \frac{1}{L(1+\sigma^2)},$$

then for any $x^* \in \mathbb{R}^m$ and any $n_0 \geq 1 + \sigma^2$,

$$\mathbb{E}[f(x_n) - f(x^*)] \leq \frac{\eta n_0 \,\mathbb{E}[f(x_0) - f(x^*)] + \frac{1}{2}\,\mathbb{E}\big[\|x_0 - x^*\|^2\big]}{\eta(n + n_0)}.$$

In particular, if $\eta = \frac{1}{L(1+\sigma^2)}$, $n_0 = 1 + \sigma^2$, and x^* is a point such that $f(x^*) = \inf_{x\in\mathbb{R}^m} f(x)$, then

$$\mathbb{E}[f(x_n) - f(x^*)] \leq \frac{L(1+\sigma^2)\,\mathbb{E}\big[\|x_0 - x^*\|^2\big]}{2(n + 1 + \sigma^2)}.$$

Proof. Let $n_0 \geq 0$ and consider the Lyapunov sequence

$$\mathscr{L}_n = \mathbb{E}\left[\eta(n+n_0)\big(f(x_n) - \inf f\big) + \frac{1}{2}\|x_n - x^*\|^2\right]$$

We find that

$$\begin{aligned}
\mathscr{L}_{n+1} &= \mathbb{E}\left[\eta(n+n_0+1)\big\{f(x_n-\eta g_n)-\inf f\big\}+\frac{1}{2}\left\|x_n-\eta g_n-x^*\right\|^2\right] \\
&\leq \mathbb{E}\Bigg[\eta(n+n_0+1)\left\{f(x_n)-\frac{\eta}{2}\|\nabla f(x_n)\|^2-\inf f\right\} \\
&\qquad +\frac{1}{2}\left\|x_n-x^*\right\|^2-\eta\left(x_n-x^*\right)\cdot g_n+\frac{\eta^2}{2}\left\|g_n\right\|^2\Bigg] \\
&= \mathbb{E}\Bigg[\eta(n+n_0)\big\{f(x_n)-\inf f\big\}+\frac{1}{2}\|x_n-x^*\|^2+f(x_n)-\inf f+\eta\nabla f(x_n)\cdot(x^*-x_n) \\
&\qquad -\frac{\eta^2(n+n_0)}{2}\|\nabla f(x_n)\|^2+\frac{\eta^2}{2}\left\|g_n\right\|^2\Bigg] \\
&\leq \mathscr{L}_n+0-\frac{\eta^2}{2}\big(n+n_0-(1+\sigma^2)\big)\,\mathbb{E}\big[\|\nabla f(x_n)\|^2\big]
\end{aligned}$$

by the convexity of f. The result therefore holds if n_0 is chosen large since

$$\mathbb{E}[f(x_n)-f(x^*)] \leq \frac{\mathscr{L}_n}{\eta(n+n_0)} \leq \frac{\mathscr{L}_0}{\eta(n+n_0)} = \frac{\eta n_0\,\mathbb{E}[f(x_0)-f(x^*)]+\frac{1}{2}\,\mathbb{E}\big[\|x_0-x^*\|^2\big]}{\eta(n+n_0)}.$$

If x^* is a minimizer of f then the last claim in the theorem follows by using the upper bound $f(x_0)-f(x^*) \leq \frac{L}{2}\|x_0-x^*\|^2$ from Lemma 12 and substituting $\eta = \frac{1}{L(1+\sigma)^2}, n_0 = 1+\sigma^2$. □

E.2 AGNES and NAG

The proofs of Theorems 1 and 3 in this section are constructed in analogy to the simplest setting of deterministic continuous-time optimization. As noted by Su et al. [2014], Nesterov's time-stepping scheme can be seen as a non-standard time discretization of the heavy ball ODE

$$\begin{cases} \ddot{x} & = -\frac{3}{t}\,\dot{x}-\nabla f(x) & t>0 \\ \dot{x} & = 0 & t=0 \\ x & = x_0 & t=0 \end{cases}$$

with a decaying friction coefficient. The same is true for AGNES, which reduces to Nesterov's method in the determinstic case. Taking the derivative and exploiting the first-order convexity condition, we see that the *Lyapunov function*

$$\mathscr{L}(t) := t^2\big(f(x(t))-f(x^*)\big)+\frac{1}{2}\left\|t\dot{x}+2\big(x(t)-x^*\big)\right\|^2 \tag{7}$$

is decreasing in time along the heavy ball ODE, see e.g. [Su et al., 2014, Theorem 3]. Here x^* is a minimizer of the convex function f. In particular

$$f(x(t))-f(x^*) \leq \frac{\mathscr{L}(t)}{t^2} \leq \frac{\mathscr{L}(0)}{t^2} = \frac{2\left\|x_0-x^*\right\|^2}{t^2}.$$

To prove Theorems 1 and 3, we construct an analogue to $\mathscr{L}$ in (7). Note that $\alpha v_n = x'_n - x_n$ is a discrete analogue of the velocity $\dot{x}$ in the continuous setting. Both the proofs follow the same outline. Since Nesterov's algorithm is a special case of AGNES, we first prove Theorem 3. We present the Lyapunov sequence in a fairly general form, which allows us to reuse calculations for both proofs and suggests the optimality of our approach for Nesterov's original algorithm.

For details on the probalistic set-up and useful properties of gradient estimators, see Appendix D.2. Let us recall the two-step formulation of AGNES, which we use for the proof,

$$x_0 = x'_0, \qquad x_{n+1} = x'_n - \eta g'_n, \qquad x'_{n+1} = x_{n+1}+\rho_n\big(x'_n-\alpha g'_n-x_n\big). \tag{2}$$

We first prove the alternative version mentioned after Theorem 3 in the main text. Both proofs proceed initially identically and only diverge in Step 3. The reader interested mainly in Theorem 3 is invited to read the first two steps of the proof of Theorem 18 and then skip ahead to the proof of Theorem 3 below.

Theorem 18 (AGNES, convex case, n_0 version)**.** *Suppose that x_n and x'_n are generated by the time-stepping scheme (3), f and $g'_n = g(x'_n, \omega_n)$ satisfy the conditions laid out in Section 3.1, f is convex, and x^* is a point such that $f(x^*) = \inf_{x\in\mathbb{R}^m} f(x)$. If the parameters are chosen such that*

$$\eta \le \frac{1}{L(1+\sigma^2)}, \qquad \alpha < \frac{\eta}{1+\sigma^2}, \qquad n_0 \ge \frac{2\sigma^2\eta}{\eta - \alpha(1+\sigma^2)}, \qquad \rho_n = \frac{n+n_0}{n+n_0+3},$$

then

$$\mathbb{E}\big[f(x_n) - f(x^*)\big] \le \frac{(\alpha n_0 + 2\eta)n_0\,\mathbb{E}\big[f(x_0) - \inf f\big] + 2\,\mathbb{E}\big[\,\|x_0 - x^*\|^2\big]}{\alpha\,(n+n_0)^2}.$$

In particular, if $\alpha \le \frac{\eta}{1+2\sigma^2}$ then it suffices to choose $n_0 \ge 2\eta/\alpha \ge 2(1+2\sigma^2)$.

Proof. **Set-up.** Mimicking the continuous time model in (7), we consider the Lyapunov sequence given by

$$\mathscr{L}_n = P(n)\,\mathbb{E}\left[f(x_n) - f(x^*)\right] + \frac{1}{2}\mathbb{E}\left[\left\|b(n)(x'_n - x_n) + a(n)(x'_n - x^*)\right\|^2\right]$$

where $P(n)$ some function of n, $a(n) = a_0 + a_1 n$, and $b(n) = b_0 + b_1 n$ for some coefficients a_0, a_1, b_0, b_1. Our goal is to choose these in such a way that $\mathscr{L}_n$ is a decreasing sequence.

Step 1. If we denote the first half of the Lyapunov sequence as $\mathscr{L}^1_n = P(n)\mathbb{E}\left[f(x_n) - f(x^*)\right]$, then

$$\begin{aligned}\mathscr{L}^1_{n+1} - \mathscr{L}^1_n &= P(n+1)\mathbb{E}[f(x_{n+1}) - f(x^*)] - P(n)\mathbb{E}[f(x_n) - f(x^*)] \\ &\le (P(n+1)+k)\mathbb{E}[f(x_{n+1}) - f(x^*)] - P(n)\,\mathbb{E}[f(x_n) - f(x^*)],\end{aligned}$$

where k is a positive constant that can be chosen later to balance out other terms. Using Lemma 15,

$$\begin{aligned}\mathscr{L}^1_{n+1} - \mathscr{L}^1_n &\le (P(n+1)+k)\mathbb{E}\left[f(x'_n) - c_{\eta,\sigma,L}\left\|\nabla f(x'_n)\right\|^2 - f(x^*)\right] - P(n)\mathbb{E}[f(x_n) - f(x^*)] \\ &= P(n)\mathbb{E}\left[f(x'_n) - f(x_n)\right] + (P(n+1)+k-P(n))\,\mathbb{E}[f(x'_n) - f(x^*)] \\ &\quad - (P(n+1)+k)c_{\eta,\sigma,L}\mathbb{E}[\left\|\nabla f(x'_n)\right\|^2]\end{aligned}$$

where $c_{\eta,\sigma,L} = \eta\left(1 - \frac{L(1+\sigma^2)\eta}{2}\right)$. Using convexity,

$$\begin{aligned}\mathscr{L}^1_{n+1} - \mathscr{L}^1_n &\le P(n)\mathbb{E}\left[\nabla f(x'_n)\cdot(x'_n - x_n)\right] + (P(n+1)+k-P(n))\,\mathbb{E}[\nabla f(x'_n)\cdot(x'_n - x^*)] \\ &\quad - (P(n+1)+k)c_{\eta,\sigma,L}\mathbb{E}[\left\|\nabla f(x'_n)\right\|^2]. \qquad (8)\end{aligned}$$

Step 2. We denote

$$w_n = b(n)(x'_n - x_n) + a(n)(x'_n - x^*)$$

and use the definition of x'_{n+1} from (2),

$$\begin{aligned}w_{n+1} &= b(n+1)(x'_{n+1} - x_{n+1}) + a(n+1)(x'_{n+1} - x^*) \\ &= b(n+1)\rho_n\left(x'_n - \alpha g'_n - x_n\right) + a(n+1)\left(x_{n+1} + \rho_n(x'_n - \alpha g'_n - x_n) - x^*\right) \\ &= (b(n+1) + a(n+1))\rho_n\left(x'_n - \alpha g'_n - x_n\right) + a(n+1)(x_{n+1} - x^*).\end{aligned}$$

We will choose

$$\rho_n = \frac{b(n)}{b(n+1) + a(n+1)},$$

such that the expression becomes

$$\begin{aligned}w_{n+1} &= b(n)\left(x'_n - \alpha g'_n - x_n\right) + a(n+1)(x_{n+1} - x^*) \\ &= b(n)\left(x'_n - \alpha g'_n - x_n\right) + (a_0 + a_1 n + a_1)(x'_n - \eta g'_n - x^*) \\ &= w_n + a_1(x'_n - x^*) - (\alpha b(n) + \eta a(n+1))g'_n.\end{aligned}$$

Then

$$\begin{aligned}\frac{1}{2}\left\|w_{n+1}\right\|^2 - \frac{1}{2}\left\|w_n\right\|^2 &= w_n\cdot(w_{n+1} - w_n) + \frac{1}{2}\left\|w_{n+1} - w_n\right\|^2 \\ &= w_n\cdot(a_1(x'_n - x^*) - (\alpha b(n) + \eta a(n+1))g'_n) \\ &\quad + \frac{1}{2}\left\|a_1(x'_n - x^*) - (\alpha b(n) + \eta a(n+1))g'_n\right\|^2.\end{aligned}$$

We want the terms in this expression to balance the terms in $\mathscr{L}^1_{n+1} - \mathscr{L}^1_n$, so we choose $a_1 = 0$, i.e. $a(n) = a_0$ is a constant. This implies,

$$\begin{aligned}
\mathbb{E}\left[\frac{1}{2}\left\|w_{n+1}\right\|^2 - \frac{1}{2}\left\|w_n\right\|^2\right] &= \mathbb{E}\left[-(\alpha b(n) + \eta a_0) w_n \cdot g'_n + \frac{1}{2}(\alpha b(n) + \eta a_0)^2 \left\|g'_n\right\|^2\right] \\
&\leq -(\alpha b(n) + \eta a_0)\mathbb{E}[w_n \cdot \nabla f(x'_n)] + \frac{1}{2}(\alpha b(n) + \eta a_0)^2(1+\sigma^2)\mathbb{E}[\|\nabla f(x'_n)\|^2] \\
&= -(\alpha b(n) + \eta a_0) b(n) \mathbb{E}[(x'_n - x_n) \cdot \nabla f(x'_n)] \\
&\quad - (\alpha b(n) + \eta a_0) a_0 \mathbb{E}[(x'_n - x^*) \cdot \nabla f(x'_n)] \\
&\quad + \frac{1}{2}(\alpha b(n) + \eta a_0)^2(1+\sigma^2)\mathbb{E}[\|\nabla f(x'_n)\|^2]. \qquad (9)
\end{aligned}$$

Step 3. Combining the estimates (8) and (9) from the last two steps,

$$\begin{aligned}
\mathscr{L}_{n+1} - \mathscr{L}_n \leq\ & (P(n) - (\alpha b(n) + \eta a_0) b(n))\, \mathbb{E}\left[\nabla f(x'_n) \cdot (x'_n - x_n)\right] \\
& + (P(n+1) + k - P(n) - (\alpha b(n) + \eta a_0) a_0)\, \mathbb{E}[\nabla f(x'_n) \cdot (x'_n - x^*)] \\
& + \left(\frac{1}{2}(\alpha b(n) + \eta a_0)^2(1+\sigma^2) - (P(n+1) + k) c_{\eta,\sigma,L}\right) \mathbb{E}[\|\nabla f(x'_n)\|^2].
\end{aligned}$$

Since $\|\nabla f(x'_n)\|^2 \geq 0$ and $\nabla f(x'_n) \cdot (x'_n - x^*) \geq f(x'_n) - f(x^*) \geq 0$, we require the coefficients of these two terms to be non-positive and the coefficient of $\nabla f(x'_n) \cdot (x'_n - x_n)$ to be zero. That gives us the following system of inequalities,

$$P(n) = (\alpha b(n) + \eta a_0) b(n) \qquad (10)$$

$$P(n+1) + k - P(n) \leq (\alpha b(n) + \eta a_0) a_0 \qquad (11)$$

$$\frac{1}{2}(\alpha b(n) + \eta a_0)^2(1+\sigma^2) \leq (P(n+1) + k)\eta \left(1 - \frac{L(1+\sigma^2)\eta}{2}\right). \qquad (12)$$

Step 4. Now we can choose values that will satisfy the above system of inequalities. We substitute $a_0 = 2, b_1 = 1, b_0 = n_0$, and $k = 2\eta - \alpha$. From (10), we get $P(n) = (\alpha(n + n_0) + 2\eta)(n + n_0)$. Next, we observe that

$$P(n+1) = P(n) + \alpha + 2\alpha(n + n_0) + 2\eta.$$

Then (11) holds because

$$\begin{aligned}
P(n+1) + k - P(n) &= \alpha + 2\alpha(n + n_0) + 2\eta + 2\eta - \alpha \\
&= 2(\alpha(n + n_0) + 2\eta) \\
&= (\alpha b(n) + \eta a_0) a_0.
\end{aligned}$$

We now choose η to satisfy $\eta \leq \frac{1}{L(1+\sigma^2)}$, which ensures that $\frac{\eta}{2} \leq \eta\left(1 - \frac{L(1+\sigma^2)\eta}{2}\right)$. Consequently, for (12), it suffices to ensure that

$$(\alpha b(n) + \eta a_0)^2(1+\sigma^2) \leq (P(n+1) + k)\eta,$$

which is equivalent to showing that the polynomial,

$$\begin{aligned}
q(z) = &\left(\alpha z^2 + 2\eta z + \alpha + 2\alpha z + 2\eta + 2\eta - \alpha\right)\eta \\
&- \left(\alpha^2 z^2 + 4\eta^2 + 4\alpha\eta z\right)(1+\sigma^2),
\end{aligned}$$

is non-negative for all $z \geq n_0$. $q(z)$ simplifies to

$$q(z) = \alpha(\eta - \alpha(1+\sigma^2))z^2 + 2\eta(\eta + \alpha - 2\alpha(1+\sigma^2))z - 4\eta^2\sigma^2.$$

To guarantee that q is non-negative for $z = n + n_0 \geq n_0$, we require that

1. the leading order coefficient is strictly positive[2] and

[2] In principle, it would suffice if the quadratic coefficient vanished and the linear coefficient were strictly positive. However, if $\eta - \alpha(1+\sigma^2) = 0$, then the coefficient of the linear term is negative since $\eta + \alpha - 2\alpha(1+\sigma^2) = -\alpha\sigma^2 < 0$. We therefore require the coefficient of the quadratic term to be positive.

2. $n_0 \geq 0$, $q(n_0) \geq 0$.

Since $q(0) < 0$ and q is quadratic, this suffices to guarantee that q is increasing on $[n_0, \infty)$. The first condition reduces to the fact that

$$\eta - \alpha(1+\sigma^2) > 0.$$

We can find the minimal admissible value of n_0 by the quadratic formula. We first consider the term *outside* the square root:

$$-\frac{2\eta(\eta+\alpha-2\alpha(1+\sigma^2))}{2\alpha(\eta-\alpha(1+\sigma^2))} = -\frac{\eta}{\alpha}\left(1 - \frac{\alpha\,\sigma^2}{\eta-\alpha(1+\sigma^2)}\right)$$

and thus

$$\begin{aligned}
n_0 &\geq -\frac{\eta}{\alpha}\left(1 - \frac{\alpha\,\sigma^2}{\eta-\alpha(1+\sigma^2)}\right) + \sqrt{\frac{\eta^2}{\alpha^2}\left(1 - \frac{\alpha\,\sigma^2}{\eta-\alpha(1+\sigma^2)}\right)^2 + \frac{4\eta^2\sigma^2}{\alpha(\eta-\alpha(1+\sigma^2))}} \\
&= \frac{\eta}{\alpha}\left\{\sqrt{1 - 2\frac{\alpha\sigma^2}{\eta-\alpha(1+\sigma^2)} + \left(\frac{\alpha\sigma^2}{\eta-\alpha(1+\sigma^2)}\right)^2 + 4\,\frac{\alpha\sigma^2}{\eta-\alpha(1+\sigma^2)}} - \left(1 - \frac{\alpha\,\sigma^2}{\eta-\alpha(1+\sigma^2)}\right)\right\} \\
&= \frac{\eta}{\alpha}\left\{\sqrt{\left(1 + \frac{\alpha\,\sigma^2}{\eta-\alpha(1+\sigma^2)}\right)^2} - \left(1 - \frac{\alpha\,\sigma^2}{\eta-\alpha(1+\sigma^2)}\right)\right\} \\
&= \frac{\eta}{\alpha}\,\frac{2\alpha\sigma^2}{\eta-\alpha(1+\sigma^2)} \\
&= \frac{2\eta\,\sigma^2}{\eta-\alpha(1+\sigma^2)}.
\end{aligned}$$

In particular, in the deterministic case $\sigma = 0$, the choice $n_0 = 0$ is admissible. Notably, we require $n_0 \geq 2\sigma^2\,\frac{\eta}{\eta} = 2\sigma^2$. Furthermore, if $\alpha \leq \frac{\eta}{1+2\sigma^2}$, then

$$\frac{2\sigma^2\eta}{\eta-\alpha(1+\sigma^2)} \leq \frac{2\sigma^2\eta}{\alpha\sigma^2} = \frac{2\eta}{\alpha},$$

so it suffices to choose $n_0 \geq 2\eta/\alpha$ in this case.

Step 5. We have shown that the Lyapunov sequence,

$$\mathscr{L}_n = ((n+n_0)\alpha+2\eta)(n+n_0)\mathbb{E}\left[f(x_n) - f(x^*)\right] + \frac{1}{2}\mathbb{E}\left[\left\|(n+n_0)(x'_n - x_n) + 2(x'_n - x^*)\right\|^2\right],$$

is monotone decreasing. It follows that

$$\mathbb{E}\big[f(x_n) - f(x^*)\big] \leq \frac{\mathscr{L}_n}{P(n)} \leq \frac{\mathscr{L}_0}{P(n)} \leq \frac{\mathbb{E}\big[(n_0\alpha+2\eta)n_0\left(f(x_0) - f(x^*)\right) + 2\left\|x_0 - x^*\right\|^2\big]}{\alpha\,(n+n_0)^2}.$$

If $2\eta \leq \alpha n_0$, we get

$$\mathbb{E}\big[f(x_n) - f(x^*)\big] \leq \frac{2\alpha n_0^2\,\mathbb{E}\big[f(x_0) - \inf f\big] + 2\,\mathbb{E}\big[\left\|x_0 - x^*\right\|^2\big]}{\alpha\,(n+n_0)^2}.$$

Finally, if $\eta = \frac{1}{L(1+\sigma^2)}$, $\alpha = \frac{1}{L(1+\sigma^2)(1+2\sigma^2)}$, and $n_0 = 2(1+2\sigma^2)$, then using Lemma 12, the expression above simplifies to

$$\mathbb{E}\big[f(x_n) - f(x^*)\big] \leq \frac{2L(1+2\sigma^2)(3+5\sigma^2)\mathbb{E}\left[\left\|x_0 - x^*\right\|^2\right]}{n^2}. \qquad \square$$

Remark 19. Note that SGD arises as a special case of this analysis if we consider $\alpha = 0, n_0 \geq 2\sigma^2$ since $P(n)$ is a *linear* polynomial in this case.

Remark 20. Note that the proof of Theorem 18 implies more generally that

$$\mathscr{L}_{n+1} \leq \mathscr{L}_n - q(n+n_0)\, E\big[\|\nabla f(x'_n)\|^2\big],$$

even if n_0 is not chosen such that $q(n+n_0) \geq 0$ for all n. However, $q(n+n_0) \geq 0$ for all *sufficiently large* $n \in \mathbb{N}$, i.e. $\mathscr{L}_n$ decreases eventually (assuming that $\mathscr{L}_n < \infty$ for all finite n) . More precisely, for given η, α if

$$n + n_0 \geq n^* := \left\lceil \frac{\eta\sigma^2}{\eta - \alpha(1+\sigma^2)} \right\rceil, \qquad \text{then} \quad \mathscr{L}_n \leq \frac{\mathscr{L}_{n^*}}{\alpha\,(n+n_0)^2}.$$

Thus a poor choice of n_0 will not prevent convergence, but it may delay it.

We now prove the version of this result stated in the main text, for which $\rho_n = \frac{n}{n+a_0+1}$ with $a_0 > 2$, i.e. with slightly more friction.

Theorem 3 (AGNES, convex case)**.** *Suppose that x_n and x'_n are generated by the time-stepping scheme (3), f and $g'_n = g(x'_n, \omega_n)$ satisfy the conditions laid out in Section 3.1, f is convex, and x^* is a point such that $f(x^*) = \inf_{x\in\mathbb{R}^m} f(x)$. If the parameters are chosen such that*

$$0 < \eta < \frac{1}{L(1+\sigma^2)}, \quad \alpha = \frac{\eta}{1+\sigma^2}, \quad \rho_n = \frac{n}{n+1+a_0}, \quad \textit{for} \quad a_0 \geq \frac{2(1-\eta L)}{1-\eta L(1+\sigma^2)}, \quad \textit{then}$$

$$\mathbb{E}\big[f(x_n) - f(x^*)\big] \leq \frac{a_0^2\, \mathbb{E}\big[\,\|x_0 - x^*\|^2\big]}{2\,\alpha\, n^2}.$$

Proof. The proof for this version of Theorem 3 is identical to the proof of Theorem 18 until Step 3, after which we take an alternate approach. Let us recall the expression we got in the beginning of Step 3.

Step 3. We want to show that the bound on the right hand side of the inequality

$$\begin{aligned}\mathscr{L}_{n+1} - \mathscr{L}_n \leq\ & (P(n) - (\alpha b(n) + \eta a_0)b(n))\, \mathbb{E}\left[\nabla f(x'_n)\cdot(x'_n - x_n)\right] \\ & + (P(n+1) + k - P(n) - (\alpha b(n) + \eta a_0)a_0)\, \mathbb{E}[\nabla f(x'_n)\cdot(x'_n - x^*)] \\ & + \left(\frac{1}{2}(\alpha b(n) + \eta a_0)^2(1+\sigma^2) - (P(n+1)+k)c_{\eta,\sigma,L}\right)\mathbb{E}[\|\nabla f(x'_n)\|^2]\end{aligned} \tag{13}$$

is non-positive. Using convexity and L-smoothness in the form of [Wojtowytsch, 2023, Lemma B.1], we get the inequality

$$\nabla f(x'_n)\cdot(x'_n - x^*) \geq f(x'_n) - f(x^*) \geq \frac{1}{2L}\,\|\nabla f(x'_n)\|^2\,,$$

which allows us to combine the second and third line in (13), assuming that the coefficient in the second line is non-positive. If this is the case, then the entire right hand side of (13) is bounded from above by

$$\begin{aligned}&(P(n) - (\alpha b(n) + \eta a_0)b(n))\, \mathbb{E}\left[\nabla f(x'_n)\cdot(x'_n - x_n)\right] \\ &\quad + \Bigg\{ \{P(n+1) + k - P(n) - (\alpha b(n) + \eta a_0)a_0\}\,\frac{1}{2L} \\ &\qquad + \left(\frac{1}{2}\big(\alpha b(n) + \eta a_0\big)^2(1+\sigma^2) - (P(n+1)+k)c_{\eta,\sigma,L}\right)\Bigg\}\mathbb{E}[\|\nabla f(x'_n)\|^2].\end{aligned}$$

As $\mathbb{E}\left[\nabla f(x'_n)\cdot(x'_n - x_n)\right]$ does not have a sign, we choose to set its coefficient to zero, and we require both the coefficient in the second line of (13) and the coefficient of $\mathbb{E}[\|\nabla f(x'_n)\|^2]$ in the combined version to be non-positive. Noting that $c_{\eta,\sigma,L} \geq \eta/2$ if $\eta \leq 1/L(1+\sigma^2)$, this leads to the system of inequalities

$$P(n) = (\alpha b(n) + \eta a_0)b(n) \tag{14}$$

$$P(n+1) + k - P(n) \leq (\alpha b(n) + \eta a_0)a_0 \tag{15}$$

$$P(n+1) + k - P(n) - (\alpha b(n) + \eta a_0)a_0 \leq L\big((P(n+1)+k)\eta - (\alpha b(n) + \eta a_0)^2(1+\sigma^2)\big). \tag{16}$$

In principle, this approach is more general than that of Theorem 3 as we do not require two terms to be individually non-positive, but only one of them and their weighted sum. In the proof of Theorem 3, a similar role was played by the parameter k, which allowed to shift a small positive term between expressions.

Step 4. Now we can choose the parameters and variables so as to satisfy the inequalities above. We begin by setting $b_1 = 1, b_0 = 0, k = 0$, and choosing α, η, a_0 as in the theorem statement. Using (14) as the definition of $P(n)$, we note that

$$P(n+1) - P(n) = 2\alpha n + \alpha + \eta a_0.$$

Thus, (15) simplifies to

$$2\alpha n + \alpha + \eta a_0 \leq a_0(\alpha n + \eta a_0),$$

which holds since $a_0 \geq 2$ and $\eta \geq \alpha$. The right hand side of (16) simplifies to

$$\begin{aligned} L\Big(\eta(n+1)(\alpha(n+1)+\eta a_0) - L\big(\alpha n + \eta a_0\big)^2(1+\sigma^2)\Big) \\ &= L\left(\left\{\eta\alpha - (1+\sigma^2)\alpha^2\right\} n^2 + \left\{\eta(2\alpha+\eta a_0) - 2\eta a_0\alpha(1+\sigma^2)\right\} n - \eta^2 a_0^2(1+\sigma^2) + \eta(\alpha+\eta a_0)\right) \\ &= L\left(\left\{\eta(2\alpha+\eta a_0) - 2\eta a_0\alpha(1+\sigma^2)\right\} n - \eta^2 a_0^2(1+\sigma^2) + \eta(\alpha+\eta a_0)\right), \end{aligned}$$

where the last equality holds since $\alpha = \eta/(1+\sigma^2)$. Thus for (16) to hold, it suffices that

$$2\alpha n + \alpha + \eta a_0 - a_0(\alpha n + \eta a_0) \leq L\left(\left\{\eta(2\alpha+\eta a_0) - 2\eta a_0\alpha(1+\sigma^2)\right\} n - \eta^2 a_0^2(1+\sigma^2) + \eta(\alpha+\eta a_0)\right),$$

which is equivalent to

$$\left\{\alpha(2-a_0) - L\eta(2\alpha+\eta a_0) + 2L\eta a_0\alpha(1+\sigma^2)\right\} n + \left\{\alpha + \eta a_0 - a_0^2\eta + L\eta^2 a_0^2(1+\sigma^2) - L\eta(\alpha+\eta a_0)\right\} \leq 0. \tag{17}$$

A linear polynomial is non-negative for all $n \geq 0$ if and only if both of its coefficients are. The leading order coefficient in (17) is

$$\begin{aligned} \alpha(2-a_0) - L\eta(2\alpha+\eta a_0) + 2L\eta a_0\alpha(1+\sigma^2) &= \alpha(2-a_0) - L\eta(2\alpha+\eta a_0) + 2L\eta^2 a_0 \\ &= 2\alpha - 2L\eta\alpha + a_0(-\alpha + L\eta^2) \\ &= \frac{\eta}{1+\sigma^2}\big(2(1-L\eta) + a_0(L\eta(1+\sigma^2) - 1)\big), \end{aligned}$$

which is non-positive if and only if $a_0 \geq \frac{2(1-L\eta)}{1-L\eta(1+\sigma^2)}$. We remark that it is this part of the computation that forces us to choose η strictly smaller than $\frac{1}{L(1+\sigma^2)}$. In the deterministic case $\sigma = 0$, we would encounter no such limitation as the term would be automatically zero for $\eta = 1/L$. Finally, we consider the constant term in (17) and use the fact that $1 < a_0$ and $\alpha \leq \eta$

$$\begin{aligned} \alpha + \eta a_0 - a_0^2\eta + L\eta^2 a_0^2(1+\sigma^2) - L\eta(\alpha+\eta a_0) &= (\alpha+\eta a_0)(1-L\eta) + a_0^2\eta(L\eta(1+\sigma^2) - 1) \\ &\leq 2\eta a_0(1-L\eta) + a_0^2\eta(L\eta(1+\sigma^2) - 1) \\ &\leq \eta a_0\left(2(1-L\eta) + a_0(L\eta(1+\sigma^2) - 1)\right) \\ &\leq 0, \end{aligned}$$

using again that $a_0 \geq 2\frac{1-L\eta}{1-L\eta(1+\sigma^2)}$. This shows that $\mathscr{L}_{n+1} \leq \mathscr{L}_n$.

Step 5. The conclusion again follows as in the proof of Theorem 18. □

In addition to convergence in expectation, we get almost sure convergence as well.

Corollary 5. *In the setting of Theorems 3 and 4, $f(x_n) \to \inf f$ with probability 1.*

The same is of course true for Theorem 18.

Proof. The conclusion follows by standard arguments from the fact that the sequence of expectations $\mathbb{E}[f(x_n) - \inf f]$ is summable: By the previous argument, the estimate

$$\mathbb{E}\big[\big|f(x_n) - f(x^*)\big|\big] = \mathbb{E}\big[f(x_n) - f(x^*)\big] \leq \frac{C}{n^2}$$

holds for some $C > 0$. Since

$$\begin{aligned}\mathbb{P}\left(\lim_{n\to\infty} f(x_n) \neq \inf f\right) &= \mathbb{P}\left(\limsup_{n\to\infty} |f(x_n) - \inf f| > 0\right) \\ &= \mathbb{P}\left(\bigcup_{k=1}^{\infty}\left\{\limsup_{n\to\infty} |f(x_n) - \inf f| > \frac{1}{k}\right\}\right) \\ &\leq \sum_{k=1}^{\infty} \mathbb{P}\left(\limsup_{n\to\infty} |f(x_n) - \inf f| > \frac{1}{k}\right),\end{aligned}$$

it suffices to show that $\mathbb{P}\left(\limsup_{n\to\infty} |f(x_n) - \inf f| > \varepsilon\right) = 0$ for any $\varepsilon > 0$. We further note that for any $N \in \mathbb{N}$ we have

$$\begin{aligned}\mathbb{P}\left(\limsup_{n\to\infty} |f(x_n) - \inf f| > \varepsilon\right) &\leq \mathbb{P}\left(\exists\, n \geq N \text{ s.t. } |f(x_n) - \inf f| > \varepsilon\right) \\ &= \mathbb{P}\left(\bigcup_{n=N}^{\infty} \{|f(x_n) - \inf f| > \varepsilon\}\right) \\ &\leq \sum_{n=N}^{\infty} \mathbb{P}\left(|f(x_n) - \inf f| > \varepsilon\right) \\ &\leq \sum_{n=N}^{\infty} \frac{\mathbb{E}\big[|f(x_n) - \inf f|\big]}{\varepsilon} \\ &\leq \frac{C}{\varepsilon} \sum_{n=N}^{\infty} \frac{1}{n^2}\end{aligned}$$

by Markov's inequality. As the series over n^{-2} converges, the expression on the right can be made arbitrarily small by choosing N sufficiently large. Thus the quantity on the left must be zero, which concludes the proof. In the strongly convex case, the series $\sum_{n=1}^{\infty}\left(1 - \sqrt{\frac{\mu}{L}}\frac{1}{1+\sigma^2}\right)^n$ converges and thus the same argument applies there as well. □

Next we turn to NAG. Let us recall the statement of Theorem 1.

Theorem 1 (NAG, convex case)**.** *Suppose that x_n and x'_n are generated by the time-stepping scheme (1), f and g satisfy the conditions laid out in Section 3.1, f is convex, and x^* is a point such that $f(x^*) = \inf_{x\in\mathbb{R}^m} f(x)$. If $\sigma < 1$ and the parameters are chosen such that*

$$0 < \eta \leq \frac{1-\sigma^2}{L(1+\sigma^2)}, \quad \textit{and} \quad \rho_n = \frac{n}{n+3}, \qquad \textit{then} \qquad \mathbb{E}[f(x_n) - f(x^*)] \leq \frac{2\mathbb{E}[\|x_0 - x^*\|^2]}{\eta n^2}.$$

The expectation on the right hand side is over the random initialization x_0.

Proof. We consider a Lyapunov sequence of the same form as before,

$$\mathscr{L}_n = P(n)\mathbb{E}\left[f(x_n) - f(x^*)\right] + \frac{1}{2}\mathbb{E}\left[\|b(n)(x'_n - x_n) + a(n)(x'_n - x^*)\|^2\right]$$

where $P(n)$ is some function of n, $a(n) = a_0 + a_1 n$, and $b(n) = b_0 + b_1 n$.

Since Nesterov's algorithm is a special case of AGNES, after substituting $\alpha = \eta$, the analysis in steps 1, 2, and 3 of the proof of Theorem 3 remains valid. With that substitution, we get the following system of inequalities corresponding to step 3,

$$P(n) = \eta(b(n) + a_0)b(n) \tag{18}$$

$$P(n+1) + k - P(n) \leq \eta(b(n) + a_0)a_0 \tag{19}$$

$$\frac{\eta^2}{2}(b(n) + a_0)^2(1+\sigma^2) \leq (P(n+1) + k)\eta\left(1 - \frac{L(1+\sigma^2)\eta}{2}\right). \tag{20}$$

Using the definition of $P(n)$ from (18), (20) is equivalent to

$$(1+\sigma^2) \leq \frac{2(b_1 n + b_1 + b_0 + a_0 + k)(b_1 n + b_0)\left(1 - \frac{L(1+\sigma^2)\eta}{2}\right)}{(b_1 n + b_0 + a_0)^2}$$

which should still hold in limit as $n \to \infty$,

$$\begin{aligned}(1+\sigma^2) &\leq \lim_{n\to\infty} \frac{2(b_1 n + b_1 + b_0 + a_0 + k)(b_1 n + b_0)\left(1 - \frac{L(1+\sigma^2)\eta}{2}\right)}{(b_1 n + b_0 + a_0)^2} \\ &= 2\left(1 - \frac{L(1+\sigma^2)\eta}{2}\right).\end{aligned}$$

This implies

$$\eta \leq \frac{1-\sigma^2}{L(1+\sigma^2)}.$$

We can choose $a_0 = 2$, $b(n) = n$, and $k = \eta$. Then (18) implies that $P(n) = \eta n(n+2)$. (19) holds because

$$P(n+1) + k - P(n) = \eta(2n+4) = \eta(b(n) + a_0)a_0$$

and (20) holds because

$$\begin{aligned}\frac{\eta}{2}(b(n)+a_0)^2(1+\sigma^2) &= \frac{\eta(n+2)^2(1+\sigma^2)}{2} \\ &\leq \frac{\eta((n+1)(n+3)+1)(1+\sigma^2)}{2} \\ &= (P(n+1)+k)\left(1 - \frac{L(1+\sigma^2)\eta}{2}\right).\end{aligned}$$

We have shown that the Lyapunov sequence

$$\mathscr{L}_n = \eta n(n+2)\mathbb{E}[f(x_n) - f(x^*)] + \frac{1}{2}\mathbb{E}[\|n(x'_n - x_n) + 2(x'_n - x^*)\|^2],$$

where $\eta \leq \frac{1-\sigma^2}{L(1+\sigma^2)}$, is monotonically decreasing. It follows that

$$\eta n(n+2)\mathbb{E}[f(x_n) - f(x^*)] \leq \mathscr{L}_n \leq \mathscr{L}_0 = 2\mathbb{E}[\|x_0 - x^*\|^2]. \qquad \square$$

We emphasize again that this analysis works only if $\sigma < 1$. The condition that $\eta \leq \frac{1-\sigma^2}{L(1+\sigma^2)}$ is imposed by (20) and does not depend on any specific choice of a_0, b_0, or b_1. On the other hand, (18) forces the rate of convergence to be inversely proportional to η. This means that as σ approaches 1, the step size η decreases to zero, and the rate of convergence blows up to infinity. On the other hand, as the proof of Theorem 3 shows, AGNES does not suffer from this problem. Having an additional parameter enables AGNES to converge even if the noise σ is arbitrarily large.

Let us point out how the same techniques used in Theorem 3 can be adapted to prove convergence $f(x_n) \to \inf f$, even if a global minimizer does not exist. We recall the main statement.

Theorem 7 (Convexity without minimizers)**.** *Let f be a convex objective function satisfying the assumptions in Section 3.1 and x_n be generated by the time-stepping scheme (3). Assume that η, α and ρ_n are as in Theorem 3. Then* $\liminf_{n\to\infty} \mathbb{E}[f(x_n)] = \inf_{x\in\mathbb{R}^m} f(x)$.

Proof. The first step follows along the same lines as the proof of Theorem 18 with minor modifications. Note that we did not use the minimizing property of x^* except for Step 5.2. Assume for the moment that $\inf f > -\infty$.

Assume first that $\varepsilon := \liminf_{n\to\infty} \mathbb{E}[f(x_n)] - \inf f > 0$. Select x^* such that $f(x^*) < \inf f + \varepsilon/4$ and define the Lyapunov sequence $\mathscr{L}_n$ just as in the proof of Theorem 3 with the selected point x^*.

We distinguish between two situations. First, assume that n satisfies $\mathbb{E}\big[f(x'_n)\big] \geq f(x^*)$. In this case we find that also $\mathbb{E}\big[f(x_{n+1}) \leq \mathbb{E}\big[f(x'_n)\big] \leq f(x^*)$.

On the other hand, assume that $\mathbb{E}\big[f(x_n')\big] \geq f(x^*)$ for $n = 0, \dots, N$. In that case, the proof of Theorem 3 still applies, meaning that $\mathbb{E}[f(x_N)]$ cannot remain larger than $f(x^*)+\varepsilon/2$ indefinitely. In either case, we find that there exists $N \in \mathbb{N}$ such that $\mathbb{E}\big[f(x_N)\big] \leq f(x^*) + \varepsilon/2 < \liminf_{n\to\infty} \mathbb{E}[f(x_n)]$.

Note that the proof of Theorem 3 applies with $n' \geq n_0$ as a starting point and a non-zero initial velocity v_n. The argument therefore shows that, for every $n' \in \mathbb{N}$ there exists $N \in \mathbb{N}$ such that $\mathbb{E}[f(x_N)] \leq \liminf_{n\to\infty} \mathbb{E}[f(x_n)]$. Inserting the definition of the lower limit, we have reached a contradiction. □

We conjecture that the statement holds with the limit in place of the lower limit, but that it is impossible to guarantee a rate of convergence $O(n^{-\beta})$ for any $\beta > 0$ in this setting. When following this strategy, the key question is how far away the point x^* must be chosen. For very flat functions such as

$$f_\alpha : \mathbb{R} \to \mathbb{R}, \qquad f_\alpha(x) = \begin{cases} x^{-\alpha} & x > 1 \\ 1 + \alpha(1-x) & x \leq 1, \end{cases}$$

x^* may be very far away from the initial point x_0, and the rate of decay can be excrutiatingly slow if minimizers do not exist. For an easy example, we turn to the continuous time model. The solution to the heavy ball ODE

$$\begin{cases} x'' &= -\frac{3}{t}\, x' - f_\alpha'(x) & t > 1 \\ x &= 1 & t = 1 \\ x' &= -\beta & t = 1 \end{cases}$$

is given by

$$x(t) = \left(\frac{4\,(3+\alpha)}{\alpha(2+\alpha)^2}\right)^{\frac{2}{2+\alpha}}\, t^{\frac{2}{2+\alpha}}$$

for $\beta = \frac{2}{2+\alpha}\left(\frac{4\,(3+\alpha)}{\alpha(2+\alpha)^2}\right)^{\frac{2}{2+\alpha}} > 0$. Ignoring the complicated constant factor, we see that

$$f_\alpha(x(t)) = x(t)^{-\alpha} \sim t^{-\frac{2\alpha}{2+\alpha}},$$

the decay rate can be as close to zero as desired for α close to zero, and indeed Siegel and Wojtowytsch [2023] show that no rate of decay can be guaranteed even beyond the situation of algebraic rates. For comparison, the solution of the gradient flow equation

$$\begin{cases} z' &= -f_\alpha'(z) & t > 0 \\ z &= 1 & t = 0 \end{cases} \qquad \text{is given by } z(t) = \big(1+\alpha(2+\alpha)t\big)^{\frac{1}{2+\alpha}} \quad \Rightarrow \quad f_\alpha(z(t)) \sim t^{-\frac{\alpha}{2+\alpha}}.$$

Thus, while both the heavy ball ODE and the gradient flow can be made arbitrarily slow in this setting, the heavy ball remains much faster in comparison.

F Convergence proofs: strongly convex case

F.1 Gradient Descent

Bassily et al. [2018], Wojtowytsch [2023] analyze stochastic gradient descent under the PL condition

$$\mu\big(f(x) - \inf f\big) \leq \frac{1}{2}\,\|\nabla f(x)\|^2 \qquad \forall\, x \in \mathbb{R}^m \tag{21}$$

and the noise scaling assumption

$$\mathbb{E}_\omega\big[\|g(x,\omega) - \nabla f(x)\|^2\big] \leq \sigma\big(f(x) - \inf f\big)$$

motivated by Lemma 8. The assumption is equivalent to multiplicative noise scaling within a constant since every L-smooth function which satisfies a PL condition satisfies

$$2\mu\,\big(f(x) - \inf f\big) \leq \|\nabla f(x)\|^2 \leq 2L\,\big(f(x) - \inf f\big).$$

For completeness, we provide a statement and proof directly in the multiplicative noise scaling regime which attains the optimal constant.

Additionally, we note that strong convexity implies the PL condition. The PL condition holds in many cases where convexity is false, e.g.

$$f(x,y) = (y - \sin x)^2, \qquad \|\nabla f\|^2 \geq |\partial_y f|^2 = 4\, f.$$

The set of minimizers $\{(x,y) : y = \sin x\}$ is non-convex, so f cannot be convex. While this result is well-known to the experts, we have been unable to locate a reference and hence provide a proof.

Lemma 21. *Assume that $f : \mathbb{R}^m \to \mathbb{R}$ is μ-strongly convex and C^2-smooth. Then f satisfies the PL-condition with constant $\mu > 0$.*

Proof. Let $x, y \in \mathbb{R}^d$. Strong convexity combined with the Cauchy-Schwartz inequality means that

$$\begin{aligned} f(x) - f(y) \leq -\langle \nabla f(x), y - x\rangle - \frac{\mu}{2}\|x-y\|^2 &\leq \|\nabla f(x)\|\|y-x\| - \frac{\mu}{2}\|x-y\|^2 \\ &\leq \max_{z\in\mathbb{R}} \|\nabla f(x)\| z - \frac{\mu}{2} z^2 \\ &= \frac{1}{2\mu}\|\nabla f(x)\|^2. \end{aligned} \tag{22}$$

Since this is true for $y = x^*$, the result follows. □

Several results in this vein are also collected in [Karimi et al., 2016, Theorem 2] together with additional generalizations of convexity, but with a suboptimal implication (μ-strongly convex & L-smooth) $\Rightarrow \mu/L$-PL. The additional implication (convexity & PL) $\Rightarrow$ strong convexity can also be found there.

Theorem 22 (GD, PL condition)**.** *Assume that f satisfies the PL-condition (21) and that the assumptions laid out in Section 3.1 are satisfied. Let x_n be the sequence generated by the gradient descent scheme*

$$g_n = g(x_n, \omega_n), \qquad x_{n+1} = x_n - \eta g_n', \qquad \eta \leq \frac{1}{L(1+\sigma^2)},$$

where $\omega_1, \omega_2, \ldots$ are elements of Ω which are drawn independently of each other and the initial condition x_0. Then the estimate

$$\mathbb{E}\left[f(x_n) - \inf_{x\in\mathbb{R}^m} f(x)\right] \leq \big(1 - \mu\eta\big)^n\, \mathbb{E}\big[f(x_0) - \inf f\big]$$

holds for any $n \in \mathbb{N}$. Additionally, the sequence x_n converges to a limiting random variable x_∞ almost surely and in L^2 such that $f(x_\infty) \equiv \inf f$ almost surely.

Proof. We denote

$$\mathscr{L}_n := \mathbb{E}\big[f(x_n) - \inf f\big]$$

and compute by Lemma 16 that

$$\begin{aligned} \mathscr{L}_{n+1} &\leq \mathbb{E}\left[f(x_n) - \frac{\eta}{2}\|\nabla f(x_n)\|^2 - \inf f\right] \\ &\leq \mathbb{E}\left[f(x_n) - \mu\eta\big(f(x_n) - f(x^*)\big) - \inf f\right] \\ &= \big(1 - \mu\eta\big)\mathscr{L}_n. \end{aligned}$$

The proof of almost sure convergence is identical to the corresponding argument in [Wojtowytsch, 2023, Theorem 2.2] and similar in spirit to that of Corollary 5. □

As usual, the optimal step-size is $\eta = \frac{1}{L(1+\sigma^2)}$ as used in Figure 1.

F.2 AGNES and NAG

Just like the convex case, we first prove Theorem 4 and set up the Lyapunov sequence with variable coefficients that can be chosen as per the time-stepping scheme. The continuous time analogue in this case is the heavy-ball ODE

$$\begin{cases} \ddot{x} &= -2\sqrt{\mu}\,\dot{x} - \nabla f(x) & t > 0 \\ \dot{x} &= 0 & t = 0 \\ x &= x_0 & t = 0 \end{cases}$$

For μ-strongly convex f, a simple calculation shows that the Lyapunov function

$$\mathscr{L}(t) = f(x(t)) - f(x^*) + \frac{1}{2}\left\|\dot{x} + \sqrt{\mu}\big(x(t) - x^*\big)\right\|^2$$

satisfies $\mathscr{L}'(t) \leq -\sqrt{\mu}\,\mathscr{L}(t)$ and thus

$$f(x(t)) - f(x^*) \leq \mathscr{L}(t) \leq e^{-\sqrt{\mu}\,t}\mathscr{L}(0) = e^{-\sqrt{\mu}\,t}\left(f(x_0) - f(x^*) + \frac{\mu}{2}\left\|x_0 - x^*\right\|^2\right).$$

See for instance [Siegel, 2019, Theorem 1] for details.

Here, we state and prove a slightly generalized version of Theorem 4 in the main text. While we assumed an optimal choice of parameters in the main text, we allow for a suboptimal selection here.

Theorem 4 (AGNES, strongly convex case – general version)**.** *In addition to the assumptions in Theorem 3, suppose that f is μ-strongly convex and that*

$$0 < \eta \leq \frac{1}{L(1+\sigma^2)}, \qquad 0 < \psi \leq \sqrt{\frac{\eta}{1+\sigma^2}}, \qquad \rho = \frac{1-\sqrt{\mu}\psi}{1+\sqrt{\mu}\psi}, \qquad \alpha = \frac{\psi - \eta\sqrt{\mu}}{1-\sqrt{\mu}\psi}\,\psi,$$

then

$$\mathbb{E}\big[f(x_n) - f(x^*)\big] \leq \big(1 - \sqrt{\mu}\,\psi\big)^n\,\mathbb{E}\left[f(x_0) - f(x^*) + \frac{\mu}{2}\left\|x_0 - x^*\right\|^2\right].$$

Note that $\mathbb{E}\left[f(x_0) - f(x^*) + \frac{\mu}{2}\left\|x_0 - x^*\right\|^2\right] \leq 2\mathbb{E}\left[f(x_0) - f(x^*)\right]$ due to Lemma 12. A discussion about the set of admissible parameters is provided after the proof. We note several special cases here.

1. If ψ is selected optimally as $\sqrt{\eta/(1+\sigma^2)}$ for η, the order of decay is $1 - \sqrt{\frac{\mu\eta}{1+\sigma^2}}$, strongly resembling Theorem 3.
2. If additionally $\eta = 1/(L(1+\sigma^2))$ is chosen optimally, then we recover the decay rate $1 - \sqrt{\mu/L}\,/(1+\sigma^2)$ claimed in the main text.
3. We recover the gradient descent algorithm with the choice $\alpha = 0$ which is achieved for $\psi = \eta\sqrt{\mu}$. This selection is admissible in our analysis since

   $$\sqrt{\mu\eta} \leq \sqrt{\frac{\mu}{L}\,\frac{1}{1+\sigma^2}} \leq \sqrt{\frac{1}{1+\sigma^2}} \quad \Rightarrow \quad \eta\sqrt{\mu} \leq \sqrt{\frac{\eta}{1+\sigma^2}}.$$

   As expected, the constant of decay is $1 - \sqrt{\mu}\,\psi = 1 - \mu\eta$, as achieved in Theorem 22. In this sense, our analysis of AGNES interpolates fully between the optimal AGNES scheme (a NAG-type scheme in the deterministic case) and (stochastic) gradient descent. However, this proof only applies in the strongly convex setting, but not under a mere PL assumption.
4. If $\mu < L$ – i.e. if $f(x) \not\equiv A + \mu\|x - x^*\|^2$ for some $A \in \mathbb{R}$ and $x_0 \in \mathbb{R}^m$ – then we can choose $0 < \psi < \sqrt{\mu}\,\eta$, corresponding to $\alpha < 0$. In this case, the gradient step is sufficiently strong to compensate for momentum taking us in the wrong direction. Needless to say, this is a terrible idea and the rate of convergence is worse than that of gradient descent.

Proof. **Set-up.** Consider the Lyapunov sequence

$$\mathscr{L}_n = \mathbb{E}\big[f(x_n) - f(x^*)\big] + \frac{1}{2}\,\mathbb{E}\left[\left\|b(x'_n - x_n) + a(x'_n - x^*)\right\|^2\right]$$

for constants b, a to be chosen later. We want to show that there exists some decay factor $0 < \delta < 1$ such that $\mathscr{L}_{n+1} \leq \delta\mathscr{L}_n$.

Step 1. Let us consider the first term. Note that

$$\begin{aligned}\mathbb{E}\big[f(x_{n+1})\big] &= \mathbb{E}\big[f(x'_n - \eta g'_n)\big] \\ &\leq \mathbb{E}\big[f(x'_n)\big] - c_{\eta,\sigma,L}\mathbb{E}\big[\|\nabla f(x'_n)\|^2\big]\end{aligned}$$

where $c_{\eta,\sigma,L} = \eta\left(1 - \frac{L\eta(1+\sigma^2)}{2}\right) \geq \eta/2$ if $\eta \leq \frac{1}{L(1+\sigma^2)}$.

Step 2. We now turn to the second term and use the definition of x'_{n+1} from (2),

$$\begin{aligned} b(x'_{n+1} - x_{n+1}) + a(x'_{n+1} - x^*) &= b\rho(x'_n - \alpha g'_n - x_n) + a(x_{n+1} + \rho(x'_n - \alpha g'_n - x_n) - x^*) \\ &= (b+a)\rho(x'_n - \alpha g'_n - x_n) + a(x'_n - \eta g'_n - x^*) \\ &= (b+a)\rho(x'_n - x_n) + a(x'_n - x^*) - ((b+a)\rho\alpha + \eta a) g'_n. \end{aligned}$$

To simplify notation, we introduce two new dependent variables:

$$c := (b+a)\rho, \qquad \psi := (b+a)\rho\alpha + \eta a = \alpha c + \eta a.$$

With these variables, we have

$$b\big(x'_{n+1} - x_{n+1}\big) + a\big(x'_{n+1} - x^*\big) = c\,(x'_n - x_n) + a(x'_n - x^*) - \psi g'_n.$$

Taking expectation of the square, we find that

$$\begin{aligned} &\mathbb{E}\left[\big\|b\big(x'_{n+1} - x_{n+1}\big) + a\big(x'_{n+1} - x^*\big)\big\|^2\right] \\ &\quad = c^2\,\mathbb{E}\big[\|x'_n - x_n\|^2\big] + 2ac\,\mathbb{E}\big[(x'_n - x_n)\cdot(x'_n - x^*)\big] + a^2\mathbb{E}\big[\|(x'_n - x^*)\|^2\big] \\ &\qquad - 2c\psi\,\mathbb{E}\big[g'_n\cdot(x'_n - x_n)\big] - 2a\psi\,\mathbb{E}\big[g'_n\cdot(x'_n - x^*)\big] + \psi^2\,\mathbb{E}\big[\|g'_n\|^2\big] \\ &\quad \le c^2\,\mathbb{E}\big[\|x'_n - x_n\|^2\big] + 2ac\,\mathbb{E}\big[(x'_n - x_n)\cdot(x'_n - x^*)\big] + a^2\mathbb{E}\big[\|(x'_n - x^*)\|^2\big] \\ &\qquad - 2c\psi\,\mathbb{E}\big[\nabla f(x'_n)\cdot(x'_n - x_n)\big] - 2a\psi\,\mathbb{E}\big[\nabla f(x'_n)\cdot(x'_n - x^*)\big] + \psi^2(1+\sigma^2)\,\mathbb{E}\big[\|\nabla f(x'_n)\|^2\big] \end{aligned}$$

Step 3. We now use strong convexity to deduce that

$$\begin{aligned} &\mathbb{E}\left[\big\|b\big(x'_{n+1} - x_{n+1}\big) + a\big(x'_{n+1} - x^*\big)\big\|^2\right] \\ &\quad \le c^2\,\mathbb{E}\big[\|x'_n - x_n\|^2\big] + 2ac\,\mathbb{E}\big[(x'_n - x_n)\cdot(x'_n - x^*)\big] + a^2\mathbb{E}\big[\|(x'_n - x^*)\|^2\big] \\ &\qquad - 2c\psi\mathbb{E}\left[f(x'_n) - f(x_n) + \frac{\mu}{2}\|x'_n - x_n\|^2\right] - 2a\psi\,\mathbb{E}\left[f(x'_n) - f(x^*) + \frac{\mu}{2}\|x'_n - x^*\|^2\right] \\ &\qquad + \psi^2(1+\sigma^2)\,\mathbb{E}\big[\|\nabla f(x'_n)\|^2\big] \\ &\quad = (c^2 - c\psi\mu)\,\mathbb{E}\big[\|x'_n - x_n\|^2\big] + 2ac\,\mathbb{E}\big[(x'_n - x_n)\cdot(x'_n - x^*)\big] + \big(a^2 - a\psi\mu\big)\,\mathbb{E}\big[\|x'_n - x^*\|^2\big] \\ &\qquad - 2c\psi\mathbb{E}\left[f(x'_n) - f(x_n)\right] - 2a\psi\,\mathbb{E}\left[f(x'_n) - f(x^*)\right] + \psi^2(1+\sigma^2)\,\mathbb{E}\big[\|\nabla f(x'_n)\|^2\big]. \end{aligned}$$

Step 4. We now add the estimates of Steps 1 and 3:

$$\begin{aligned} \mathscr{L}_{n+1} &= \mathbb{E}\left[f(x_{n+1}) - f(x^*) + \frac{1}{2}\|b(x'_{n+1} - x_{n+1}) + a(x'_{n+1} - x^*)\|^2\right] \\ &\le (1 - c\psi - a\psi)\,\mathbb{E}\big[f(x'_n)\big] + c\psi\,\mathbb{E}\big[f(x_n)\big] - (1 - a\psi)\,\mathbb{E}\big[f(x^*)\big] \\ &\quad + \frac{1}{2}(c^2 - c\psi\mu)\,\mathbb{E}\big[\|x'_n - x_n\|^2\big] + ac\,\mathbb{E}\big[(x'_n - x_n)\cdot(x'_n - x^*)\big] \\ &\quad + \frac{1}{2}\left(a^2 - a\psi\mu\right)\,\mathbb{E}\big[\|x'_n - x^*\|^2\big] + \left(\frac{\psi^2(1+\sigma^2)}{2} - c_{\eta,\sigma,L}\right)\,\mathbb{E}\big[\|\nabla f(x'_n)\|^2\big] \end{aligned}$$

We require the coefficient of $\mathbb{E}[f(x'_n)]$ to be zero, i.e. $1 - a\psi = c\psi$, so the inequality simplifies to

$$\begin{aligned} \mathscr{L}_{n+1} &\le c\psi\,\mathbb{E}\big[f(x_n) - f(x^*)\big] + \frac{1}{2}(c^2 - c\psi\mu)\,\mathbb{E}\big[\|x'_n - x_n\|^2\big] \\ &\quad + ac\,\mathbb{E}\big[(x'_n - x_n)\cdot(x'_n - x^*)\big] + \frac{1}{2}\left(a^2 - a\psi\mu\right)\,\mathbb{E}\big[\|x'_n - x^*\|^2\big] \\ &\quad + \left(\frac{\psi^2(1+\sigma^2)}{2} - c_{\eta,\sigma,L}\right)\,\mathbb{E}\big[\|\nabla f(x'_n)\|^2\big]. \end{aligned}$$

The smallest decay factor we can get at this point is the coefficient of $\mathbb{E}[f(x_n) - f(x^*)]$. So we hope to show that $\mathscr{L}_{n+1} \le c\psi\mathscr{L}_n$, which leads to the following system of inequalities on comparing it

with the coefficients in the upper bound that we obtained in the previous step,

$$c = (b+a)\rho \tag{23}$$
$$\psi = \alpha c + \eta a \tag{24}$$
$$(c+a)\psi = 1 \tag{25}$$
$$c^2 - c\psi\mu \le c\psi b^2 \tag{26}$$
$$ac = c\psi ab \tag{27}$$
$$a^2 - a\psi\mu \le c\psi a^2 \tag{28}$$
$$\frac{(1+\sigma^2)\psi^2}{2} \le \eta\left(1 - \frac{L(1+\sigma^2)\eta}{2}\right) \tag{29}$$

Step 5. Now we try to choose constants such that the system of inequalities holds. We assume that $\eta \le \frac{1}{L(1+\sigma^2)}$. Then since $\frac{\eta}{2} \le \eta\left(1 - \frac{L(1+\sigma^2)\eta}{2}\right)$, for (29) it suffices that $(1+\sigma^2)\psi^2 \le \eta$, i.e.

$$\psi \le \sqrt{\frac{\eta}{1+\sigma^2}}.$$

Note that (27) implies $\psi = 1/b$ and substituting that into (25), we get $c = b - a$. Using this, (28) is equivalent to

$$a^2 - a\psi\mu \le c\psi a^2 = (\frac{1}{\psi} - a)\psi a^2 = a^2 - a^2\psi,$$

which holds with equality if $a = \sqrt{\mu}$. (26) holds because

$$c - \psi\mu = b - a - \psi\mu \le b = \psi b^2,$$

if $\mu, \psi > 0$. Finally (23) implies

$$\rho = \frac{b-a}{b+a} = \frac{1-\sqrt{\mu}\psi}{1+\sqrt{\mu}\psi},$$

and (24) implies

$$\alpha = \frac{\frac{1}{b} - \eta a}{b-a} = \frac{\psi^2 - \eta\sqrt{\mu}\psi}{1-\sqrt{\mu}\psi}.$$

With these choices of parameters, $\mathscr{L}_{n+1} \le c\psi\mathscr{L}_n = (1-\sqrt{\mu}\psi)\mathscr{L}_n$, and thus

$$\begin{aligned}
\mathbb{E}[f(x_n) - f(x^*)] &\le (1-\sqrt{\mu}\psi)^n \mathscr{L}_0 \\
&= (1-\sqrt{\mu}\psi)^n \mathbb{E}\left[f(x_0) - f(x^*) + \frac{\mu}{2}\left\|x_0 - x^*\right\|^2\right] \\
&\le 2(1-\sqrt{\mu}\psi)^n \mathbb{E}\left[f(x_0) - f(x^*)\right],
\end{aligned}$$

where we have used Lemma 12 for strong convexity in the last step. When the parameters are chosen optimally, i.e. $\eta = \frac{1}{L(1+\sigma^2)}$ and $\psi = \sqrt{\frac{\eta}{1+\sigma^2}} = \frac{1}{\sqrt{L}(1+\sigma^2)}$, we get ρ, α and the convergence rate as stated in the theorem. □

We focus on the meaningful case in which $\sqrt{\mu}\psi > 0$. As discussed in Section 3, for given f, g we can replace L, σ by larger values L', σ' and μ by a smaller value μ'. Let us briefly explore the effect of these substitutions. The parameter range described in this version of Theorem 4 can be understood as a three parameter family of AGNES parameters η, α, ρ parametrized by η, ψ, μ' and constraints given by L, μ, σ as

$$D := \left\{(\eta, \psi, \mu') \;\middle|\; 0 < \eta \le \frac{1}{L(1+\sigma^2)},\ 0 < \psi \le \sqrt{\frac{\eta}{1+\sigma^2}},\ 0 < \mu' \le \mu\right\}.$$

The parameter map is given by

$$(\eta, \psi, \mu') \mapsto (\eta, \rho, \alpha) = \left(\eta,\ \frac{1-\sqrt{\mu'}\psi}{1+\sqrt{\mu'}\psi},\ \frac{\psi - \eta\sqrt{\mu'}}{1-\sqrt{\mu'}\,\psi}\,\psi\right).$$

We can converesely obtain $\sqrt{\mu'}\psi$ from ρ since the function $z \mapsto (1-z)/(1+z)$ is its own inverse and thus $\sqrt{\mu'}\psi = \frac{1-\rho}{1+\rho}$. In particular, in terms of the algorithms parameters, the decay rate is

$$1 - \sqrt{\mu'}\psi = 1 - \frac{1-\rho}{1+\rho} = \frac{2\rho}{1+\rho}.$$

Furthermore, we see that

$$\alpha = \frac{\psi^2 - \eta\sqrt{\mu'}\psi}{1-\sqrt{\mu'}\psi} = \frac{\psi^2 - \eta\frac{1-\rho}{1+\rho}}{1-\frac{1-\rho}{1+\rho}} = \frac{1+\rho}{2\rho}\left(\psi^2 - \eta\frac{1-\rho}{1+\rho}\right) \quad \Leftrightarrow \quad \psi = \sqrt{\frac{2\rho}{1+\rho}\alpha + \eta\frac{1-\rho}{1+\rho}}$$

since $\psi > 0$. Thus, at the cost of a more complicated representation, we could work directly in the parameter variables rather than using the auxiliary quantities ψ, μ'. In particular, both the parameter map and its inverse are continuous on D and its image respectively. Hence,despite the rigid appearance of the parameter selection in Theorem 4, there exists an open set of admissible parameters η, α, ρ for which we obtain exponentially fast convergence.

We provide a more general version of Theorem 2 as well. Just as in the convex case, as $\sigma \nearrow 1$, the step size η decreases to zero and the theorem fails to guarantee convergence for $\sigma > 1$.

Theorem 2 (NAG, strongly convex case)**.** *In addition to the assumptions in Theorem 1, suppose that f is μ-strongly convex and the parameters are chosen such that*

$$0 < \eta \leq \frac{1-\sigma^2}{L(1+\sigma^2)} \text{ and } \rho = \frac{1-\sqrt{\mu\eta}}{1+\sqrt{\mu\eta}}, \text{ then } \mathbb{E}[f(x_n) - f(x^*)] \leq 2(1-\sqrt{\mu\eta})^n\, \mathbb{E}\left[f(x_0) - f(x^*)\right].$$

Proof. Consider the Lyapunov sequence

$$\mathscr{L}_n = \mathbb{E}\left[f(x_n) - f(x^*)\right] + \frac{1}{2}\mathbb{E}\left[\left\|b(x_n' - x_n) + a(x_n' - x^*)\right\|^2\right],$$

where a and b are to be determined later. Since NAG is a special case of AGNES with $\alpha = \eta$, the first four steps are identical to the proof of Theorem 4. We get the following system of inequalities,

$$c = (a+b)\rho \tag{30}$$
$$\psi = \eta(a+c) \tag{31}$$
$$(c+a)\psi = 1 \tag{32}$$
$$c^2 - c\psi\mu \leq b^2\psi \tag{33}$$
$$a^2 - a\psi\mu \leq a^2\psi \tag{34}$$
$$ac = abc\psi \tag{35}$$
$$\frac{\psi^2(1+\sigma^2)}{2} \leq \eta\left(1 - \frac{L\eta(1+\sigma^2)}{2}\right) \tag{36}$$

Substituting (31) into (32), we get $(a+c)^2 = \frac{1}{\eta}$ and $\psi = \eta/\sqrt{\eta} = \sqrt{\eta}$. Thus (35) simplifies to

$$\frac{1-\sigma^2}{2} \leq 1 - \frac{L\eta(1+\sigma^2)}{2},$$

which is equivalent to

$$\eta \leq \frac{1-\sigma^2}{L(1+\sigma^2)}.$$

From (35), $b = 1/\psi = 1/\sqrt{\eta}$. The rest of the inequalities can be verified to work with $a = \sqrt{\mu}, c = b - a, \rho = \frac{b-a}{b+a}$. This shows that $\mathscr{L}_{n+1} \leq c\psi\mathscr{L}_n = (1-\sqrt{\mu\eta})\mathscr{L}_n$. Finally, we get

$$\begin{aligned}\mathbb{E}[f(x_n) - f(x^*)] &\leq (1-\sqrt{\mu\eta})^n\, \mathbb{E}\left[f(x_0) - f(x^*) + \frac{\mu}{2}\left\|x_0 - x^*\right\|^2\right] \\ &\leq 2(1-\sqrt{\mu\eta})^n\, \mathbb{E}\left[f(x_0) - f(x^*)\right].\end{aligned}$$

□

F.3 On the role of momentum parameters

Two different AGNES parameters are associated with momentum: α and ρ. In this section, we disentangle their respective contributions to keeping AGNES stable for highly stochastic noise.

For simplicity, first consider the case $f : \mathbb{R} \to \mathbb{R}$, $f(x) = x$ and $g(x) = (1 + \sigma N)\, f'(x)$ where N is a standard normal random variable. Then

$$v_{n+1} = \rho(v_n - g'_n) = \cdots = -\rho \sum_{i=0}^{n} \rho^{n-i} g'_i.$$

since $v_0 = 0$. In particular, we note that

$$\mathbb{E}[v_{n+1}] = -\rho \sum_{i=1}^{n} \rho^{n-i} \mathbb{E}[g'_i] = -\rho \sum_{i=1}^{n} \rho^{n-i} = -\rho \frac{1 - \rho^{n+1}}{1 - \rho}.$$

and

$$\begin{aligned}\mathbb{E}\left[\left| v_{n+1} - \left(-\rho \frac{1-\rho^{n+1}}{1-\rho}\right)\right|^2\right] &= \rho^2 \mathbb{E}\left[\left|\sum_{i=0}^{n} \rho^{n-i}(g'_i - 1)\right|^2\right] = \sigma^2\rho^2 \sum_{i=0}^{n} \rho^{2(n-i)} \mathbb{E}\left[|g'_i - 1|^2\right] \\ &= \sigma^2\rho^2 \sum_{i=0}^{n} \rho^{2(n-i)} = \sigma^2\rho^2 \, \frac{1-\rho^{2(n+1)}}{1-\rho^2}\end{aligned}$$

due to the independence of different gradient estimators between time steps. In particular, we see that

1. as ρ becomes closer to 1, the eventual magnitude of the velocity variable increases as $\lim_{n\to\infty} \mathbb{E}\|v_n\| = \frac{\rho}{1-\rho}$.
2. as ρ becomes closer to 1, the eventual variance of the velocity variable increases as $\lim_{n\to\infty} \mathbb{E}\left[\|v_n - \mathbb{E}[v_n]\|^2\right] = \frac{\rho^2}{1-\rho^2}$.
3. the noise in the normalized velocity estimate asymptotically satisfies

$$\lim_{n\to\infty} \mathbb{E}\left[\left\| \frac{v_n - \mathbb{E}[v_n]}{\mathbb{E}[\|v_n\|]} \right\|^2\right] = \sigma^2 \frac{(1-\rho)^2}{1-\rho^2} = \sigma^2 \frac{(1-\rho)^2}{(1-\rho)(1+\rho)} = \sigma^2 \frac{1-\rho}{1+\rho}$$

Thus, if ρ is closer to 1, both the magnitude and the variance of the velocity variable increase, but the the relative importance of noise approaches zero as $\rho \to 1$. This is not surprising – if ρ is close to 1, the sequence ρ^n decays much slower than if ρ is small. Gradient estimates from different times enter at a similar scale and cancellations can occur easily. As the influence of past gradients remains large, we say that the momentum variable has a 'long memory'.

Of course, when minimizing a non-linear function f, the gradient is not constant, and we face a trade-off:

1. A long memory allows us to cancel random oscillations in the gradient estimates more easily.
2. A long memory also means we compute with more out-of-date gradient estimates from points much further in the past along the trajectory.

Naturally, the relative importance of the first point increases with the stochasticity σ of the gradient estimates. Even if the gradient evaluations are deterministic, we benefit from integrating historic information gained throughout the optimization process, but the rate at which we 'forget' outdated information is much higher.

Thus the parameter ρ corresponds to the rate at which we forget old information. It also impacts the magnitude of the velocity variable. The parameter α compensates for the scaling of v_n with $1/(1-\rho)$. We can think of ρ as governing the rate at which we forget past gradients, and α as a measure of the confidence with which we integrate past gradient information into time-steps for x.

Let us explore this relationship in strongly convex optimization. In Theorem 4, the optimal choice of hyper-parameters is given by $\eta = \frac{1}{L(1+\sigma^2)}$ and

$$\alpha = \frac{1-\sqrt{\mu/L}}{1-\sqrt{\mu/L}+\sigma^2}\eta, \qquad \rho = \frac{\sqrt{L}\,(1+\sigma^2)-\sqrt{\mu}}{\sqrt{L}(1+\sigma^2)+\sqrt{\mu}} = 1 - \frac{2\sqrt{\mu}}{\sqrt{L}(1+\sigma^2)+\sqrt{\mu}}.$$

Let us consider the simplified regime $\mu \ll L$ in which

$$\alpha \approx \frac{\eta}{1+\sigma^2}, \qquad \rho \approx 1 - 2\sqrt{\frac{\mu}{L}}\,\frac{1}{1+\sigma^2} \quad \Rightarrow \quad \frac{\alpha}{1-\rho} = \frac{\eta}{2\sqrt{\mu/L}}.$$

In particular, we note: The larger σ, the closer ρ is to 1, i.e. the longer the memory we keep. The relative importance of the momentum step compared to the gradient step, on the other hand, remains constant, depending only on the 'condition number' L/μ.

We note that also in the convex case, high stochasticity forces n_0 to be large, meaning that ρ_n is always close to 1. Notably for generic non-convex objective functions, it is unclear that past gradients along the trajectory would carry useful information, as there is no discernible geometric relationship between gradients at different points. This mirrors an observation of Appendix G, just after Theorem 23.

G AGNES in non-convex optimization

We consider the case of non-convex optimization. In the deterministic setting, momentum methods for non-convex optimization have recently been studied by Diakonikolas and Jordan [2021]. We note that the algorithm may perform worse than stochastic gradient descent, but that for suitable parameters, the performance is comparable to that of SGD within a constant factor.

Theorem 23 (Non-convex case)**.** *Assume that f satisfies the assumptions laid out in Section 3.1. Let η, α, ρ be such that*

$$\eta \le \frac{1}{L(1+\sigma^2)}, \qquad \alpha < \frac{\eta}{1+\sigma^2}, \qquad (L\alpha+1)\rho^2 \le 1.$$

Then

$$\min_{0\le i\le n} \mathbb{E}\big[\|\nabla f(x_i)\|^2\big] \le \frac{2\,\mathbb{E}\left[f(x_0) - \inf f + \frac{1}{\alpha\rho^2}\|v_0\|^2\right]}{(n+1)\,(\eta-\alpha(1+\sigma^2))}.$$

If $v_0 = 0$, the bound is minimal for gradient descent (i.e. $\alpha = 0$) since the decay factor $\varepsilon = \eta - \alpha(1+\sigma^2)$ is maximal.

Proof. Consider

$$\mathscr{L}_n = \mathbb{E}\left[f(x_n) + \frac{\lambda}{2}\|x'_n - x_n\|^2\right].$$

for a parameter $\lambda > 0$ to be fixed later. We have

$$\begin{aligned}
\mathbb{E}\big[f(x_{n+1})\big] &\le \mathbb{E}\big[f(x'_n)\big] - \frac{\eta}{2}\,\mathbb{E}\big[\|\nabla f(x'_n)\|^2\big] \\
&\le \mathbb{E}\left[f(x_n) + \nabla f(x'_n)\cdot(x'_n - x_n) + \frac{L\,\alpha^2}{2}\|v_n\|^2 - \frac{\eta}{2}\,\|\nabla f(x'_n)\|^2\right] \\
\mathbb{E}\big[\|x'_{n+1} - x_{n+1}\|^2\big] &= \rho^2\mathbb{E}\big[\|(x'_n - x_n)\|^2 - 2\alpha\,(x'_n - x_n)\cdot\, g'_n + \alpha^2\,\|g'_n\|^2\big]
\end{aligned}$$

by Lemmas 13 and 16. We deduce that

$$\begin{aligned}
\mathscr{L}_{n+1} &\le \mathbb{E}\big[f(x_n)\big] + \big(1-\lambda\alpha\rho^2\big)\,\mathbb{E}\big[\nabla f(x'_n)\cdot(x'_n - x_n)\big] + \frac{L+\lambda\rho^2}{2}\,\mathbb{E}\big[\|x'_n - x_n\|^2\big] \\
&\qquad + \frac{\lambda\rho^2\alpha\cdot\alpha(1+\sigma^2)-\eta}{2}\,\mathbb{E}\big[\|\nabla f(x'_n)\|^2\big] \\
&\le \mathscr{L}_n + \frac{\lambda\rho^2\alpha\cdot\alpha(1+\sigma^2)-\eta}{2}\,\mathbb{E}\big[\|\nabla f(x'_n)\|^2\big]
\end{aligned}$$

under the conditions

$$1 - \lambda\alpha\rho^2 = 0, \qquad L + \lambda\rho^2 \leq \lambda.$$

The first condition implies that $\lambda = (\alpha\rho^2)^{-1}$, so the second one reduces to

$$(1-\rho^2)\lambda = \frac{1-\rho^2}{\rho^2\alpha} \geq L \quad \Leftrightarrow \quad 1-\rho^2 \geq L\rho^2\alpha \quad \Leftrightarrow \quad 1 \geq (1+L\alpha)\rho^2.$$

Finally, we consider the last equation. If

$$\varepsilon := \eta - \lambda\rho^2\alpha \cdot \alpha(1+\sigma^2) = \eta - \alpha(1+\sigma^2) > 0,$$

then we find that

$$\mathbb{E}\left[f(x_0) + \frac{1}{\alpha\rho^2}\left\|v_0\right\|^2 - \inf f\right] \geq \mathscr{L}_1 - \mathscr{L}_{n+1} = \sum_{i=0}^{n}(\mathscr{L}_i - \mathscr{L}_{i+1}) \geq \frac{\varepsilon}{2}\sum_{i=1}^{n}\mathbb{E}\big[\left\|\nabla f(x_i)\right\|^2\big]$$

and hence

$$\min_{0\leq i\leq n}\mathbb{E}\big[\left\|\nabla f(x_i)\right\|^2\big] \leq \frac{1}{n+1}\sum_{i=1}^{n}\mathbb{E}\big[\left\|\nabla f(x_i)\right\|^2\big] \leq \frac{2\mathbb{E}\left[f(x_0) - \inf f + \frac{1}{\alpha\rho^2}\left\|v_0\right\|^2\right]}{\varepsilon(n+1)}. \qquad \square$$

H Proof of Lemma 8: Scaling intensity of minibatch noise

In this appendix, we provide theoretical justification for the multiplicative noise scaling regime considered in this article. Recall our main statement:

Lemma 8 (Noise intensity)**.** *Assume that $\ell(h,y) = \|h-y\|^2$ and $h : \mathbb{R}^m \times \mathbb{R}^d \to \mathbb{R}^k$ satisfies $\|\nabla_w h(w,x_i)\|^2 \leq C\big(1+\|w\|\big)^p$ for some $C, p > 0$ and all $w \in \mathbb{R}^m$ and $i = 1,\ldots,N$. Then for all $w \in \mathbb{R}^m$:*

$$\frac{1}{N}\sum_{i=1}^{N}\left\|\nabla\ell_i - \nabla\mathcal{R}\right\|^2 \;\leq\; 4C^2\,(1+\|w\|)^{2p}\,\mathcal{R}(w).$$

Proof. Since $\nabla\mathcal{R} = \frac{1}{n}\sum_{i=1}^{n}\nabla\ell_i$, we observe that

$$\frac{1}{n}\sum_{i=1}^{n}\left\|\nabla\ell_i - \nabla\mathcal{R}\right\|^2 \leq \frac{1}{n}\sum_{i=1}^{n}\left\|\nabla\ell_i\right\|^2$$

as the average of a quantity is the unique value which minimizes the mean square discrepancy: $\mathbb{E}X = \operatorname{argmin}_{a\in\mathbb{R}}\mathbb{E}\big[|X-a|^2\big]$. We further find by Hölder's inequality that

$$\begin{aligned}
\frac{1}{n}\sum_{i=1}^{n}\left\|\nabla\ell_i\right\|^2 &= \frac{1}{n}\sum_{i=1}^{n}\left\|\sum_{j=1}^{k}2\big(h_j(w,x_i)-y_{i,j}\big)\nabla_w h_j(w,x_i)\right\|^2 \\
&\leq \frac{4}{n}\sum_{i=1}^{n}\left(\sum_{j=1}^{k}\big(h_j(w,x_i)-y_{i,j}\big)^2\right)\left(\sum_{j=1}^{k}\left\|\nabla_w h_j(w,x_i)\right\|_2^2\right) \\
&= \frac{4}{n}\sum_{i=1}^{n}\left\|h(w,x_i)-y_i\right\|_2^2\left\|\nabla_w h(w,x_i)\right\|^2 \\
&\leq 4C^2\big(1+\|w\|^2\big)^{2p}\,\frac{1}{n}\sum_{i=1}^{n}\left\|h(w,x_i)-y_i\right\|_2^2 \\
&= 4C^2\big(1+\|w\|^2\big)^{2p}\,\mathcal{R}(w).
\end{aligned}$$

$\square$

I Implementation aspects

We discuss some implementation in this section. All the code used for the experiments in the paper has been provided in the supplementary materials. The experiments in section Section 5 and Appendix A were run on Google Colab for compute time less than an hour. The experiments in Section 5.2 were run on a laptop CPU with compute time less than an hour. The experiments in Sections 5.3 and 5.4 were run on a single current generation GPU in a local cluster for up to 50 hours. An additional compute of no more than 200 hours on a single GPU was used for experiments which were ultimately not used in the submitted version.

I.1 The last iterate

All neural-network based experiments were performed using the PyTorch library. Gradient-based optimizers in PyTorch and TensorFlow are implemented in such a way that gradients are computed outside of the optimizer and the point returned by an optimizer step is the point for the next gradient evaluation. This strategy facilitates the manual manipulation of gradients by scaling, clipping or masking to train only a subset of the network parameters.

The approach is theoretically justified for SGD. Guarantees for NAG and AGNES, on the other hand, are given for $f(x_n)$ rather than $f(x'_n)$, i.e. not at the point where the gradient is evaluated. A discrepancy arises between theory and practice.[3] In Algorithm 1, this discrepancy is resolved by taking a final gradient descent step in the last time step and returning the sequence x'_n at intermediate steps. In our numerical experiments, we did not include the final gradient descent step. Skipping the gradient step in particular allows for an easier continuation of simulations beyond the initially specified stopping time, if so desired. We do not anticipate major differences under realistic circumstances. This can be justified analytically in convex and strongly convex optimization, at least for a low learning rate.

Lemma 24. *If $\eta < \frac{1}{3L}$, then*

$$\mathbb{E}\big[f(x'_n) - f(x^*)\big] \leq \frac{\mathbb{E}[f(x_{n+1}) - f(x^*)]}{1 - 3L\eta}.$$

Proof. By essentially the same proof as Lemma 16, we have

$$\mathbb{E}\left[f(x'_n) - \frac{3\eta}{2}\|\nabla f(x'_n)\|^2\right] \leq \mathbb{E}\big[f(x_{n+1})\big] \leq \mathbb{E}\left[f(x'_n) - \frac{\eta}{2}\|\nabla f(x'_n)\|^2\right],$$

since the correction term to linear approximation is bounded by the L-Lipschitz continuity of ∇f both from above and below. Recall furthermore that

$$\|\nabla f(x)\|^2 \leq 2L\left(f(x) - f(x^*)\right)$$

for all L-smooth functions. Thus

$$(1 - 3L\eta)\mathbb{E}\big[f(x'_n) - f(x^*)\big] \leq \mathbb{E}\left[f(x'_n) - \frac{3\eta}{2}\|\nabla f(x'_n)\|^2\right] \leq \mathbb{E}\big[f(x_{n+1})\big].$$

In particular, if $1 - 3L\eta > 0$, then

$$\mathbb{E}\big[f(x'_n) - f(x^*)\big] \leq \frac{1}{1 - 3L\eta}\,\mathbb{E}[f(x_{n+1}) - f(x^*)]. \qquad \square$$

The condition $\eta < 1/(3L)$ is guaranteed if the stochastic noise scaling satisfies $\sigma > \sqrt{2}$ since then $1 - 3L\eta \geq 1 - \frac{3}{1+\sigma^2}$. For $\eta = 1/((1+\sigma^2)L$, we than find that

$$\mathbb{E}\big[f(x'_n) - f(x^*)\big] \leq \frac{\mathbb{E}[f(x_{n+1}) - f(x^*)]}{1 - \frac{3}{1+\sigma^2}} = \frac{\sigma^2 + 1}{\sigma^2 - 2}\,\mathbb{E}[f(x_{n+1}) - f(x^*)].$$

[3] For instance, the implementations of NAG in PyTorch and Tensorflow return x'_n rather than x_n.

I.2 Weight decay

Weight decay is a machine learning tool which controls the magnitude of the coefficients of a neural network. In the simplest SGD setting, weight decay takes the form of a modified update step

$$x_{n+1} = (1 - \lambda\eta)x_n - \eta g_n$$

for $\lambda > 0$. A gradient flow is governed by (1) an energy to be minimized and (2) an energy dissipation mechanism [Peletier, 2014]. It is known that different energy/dissipation pairings may induce the same dynamics – for instance, Jordan et al. [1998] show that the heat equation is both the L^2-gradient flow of the Dirichlet energy and the Wasserstein gradient flow of the entropy function.

In this language, weight decay can be interpreted in two different ways:

1. We minimize a modified objective function $x \mapsto f(x) + \frac{\lambda}{2}\|x\|^2$ which includes a Tikhonov regularizer. The gradient estimates are stochastic for f and deterministic for the regularizer. This perspective corresponds to including weight decay as part of the *energy*.
2. We dynamically include a confinement into the optimizer which pushes back against large values of x_n. This perspective corresponds to including weight decay as part of the *dissipation*.

In GD, both perspectives lead to the same optimization algorithm. In advanced minimizers, the two perspectives no longer coincide. For Adam, Loshchilov and Hutter [2018, 2019] initiated a debate on the superior strategy of including weight decay. We note that the two strategies do not coincide for AGNES, but do not comment on the superiority of one over the other:

1. Treating weight decay as a dynamic property of the optimizer leads to an update rule like
$$x'_n = x_n + \alpha v_n, \qquad v_{n+1} = \rho\big(v_n - g'_n\big), \qquad x_{n+1} = (1 - \lambda\eta)x'_n - \eta g'_n.$$
2. Treating weight decay as a component of the objective function to be minimized leads to the update rule
$$x'_n = x_n + \alpha v_n, \qquad v_{n+1} = \rho\left(v_n - g'_n - \lambda\, x'_n\right), \qquad x_{n+1} = (1 - \lambda\eta)x'_n - \eta g'_n.$$

In our numerical experiments, we choose the second approach, viewing weight decay as a property of the objective function rather than the dissipation. This coincides with the approach taken by the SGD (and SGD with momentum) optimizer as well as Adam (but not AdamW).

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We wrote the abstract and introduction with the goal to summarize our main contributions accurately and precisely.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We compare the algorithm proposed to commonly used methods both in convex optimization and deep learning. We dedicate Section 3 to the derivation of the noise modelling assumption and illustrate the heuristics which are being made.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: All assumptions are summarized in Section 3.1. Wherever additional assumptions are made, they are stated clearly in the proof. Complete and correct proofs for all the lemmas and theorems are provided in the appendices.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: All code from experiments is provided in the supplementary materials.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: All code as well as the synthetically generated data used for the regression experiments are provided in the supplementary materials.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: All experimental settings are described in the article and its supplementary materials. They can also be inferred in the code provided.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: All experiments were repeated multiple times. We provide means and standard deviations over all runs.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The information on compute resources is provided in the appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The work presented here is primarily theoretical. No human subjects were involved. The datasets used are standard benchmark datasets (MNIST, CIFAR-10) or purely synthetic. No direct social consequences are anticipated.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: The main contribution of the work is an algorithm for smooth convex optimization. Foundational as the topic at large may be in various fields, it is impossible to link directly to societal impact.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The main contribution of the work is theoretical and no data or models with a high risk for misuse are produced.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We use the MNIST and CIFAR-10 datasets, which are cited accurately. We also use an implementation of ResNets, for which we cite the GitHub repository and reproduce the license terms in the code provided.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.

- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: No new assets are released.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.