

On the fast convergence of minibatch heavy ball momentum

RAGHU BOLLAPRAGADA*

*Operations Research and Industrial Engineering, The University of Texas at Austin,
204 E. Dean Keeton Street, 78712, TX, USA*

*Corresponding author: Raghu.bollapragada@utexas.edu

TYLER CHEN

*Mathematics, New York University, 251 Mercer Street, 10012, NY, USA
Computer Science and Engineering, New York University, 370 Jay Street, 11201, NY, USA*

AND

RACHEL WARD

*Mathematics, The University of Texas at Austin, 2515 Speedway, 78712, TX, USA
Computational Engineering and Sciences, The University of Texas at Austin,
201 E. 24th Street, 78712, TX, USA*

[Received on 10 June 2023; revised on 12 December 2023]

Simple stochastic momentum methods are widely used in machine learning optimization, but their good practical performance is at odds with an absence of theoretical guarantees of acceleration in the literature. In this work, we aim to close the gap between theory and practice by showing that stochastic heavy ball momentum retains the fast linear rate of (deterministic) heavy ball momentum on quadratic optimization problems, at least when minibatching with a sufficiently large batch size. The algorithm we study can be interpreted as an accelerated randomized Kaczmarz algorithm with minibatching and heavy ball momentum. The analysis relies on carefully decomposing the momentum transition matrix, and using new spectral norm concentration bounds for products of independent random matrices. We provide numerical illustrations demonstrating that our bounds are reasonably sharp.

Keywords: Momentum; Stochastic Gradient; Linear Systems; Least Squares.

1. Introduction

The success of learning algorithms trained with stochastic gradient descent (SGD) variants, dating back to the seminal AlexNet architecture (Krizhevsky *et al.*, 2012)—arguably initiating the ‘deep learning revolution’—and empirically verified comprehensively in Sutskever *et al.* (2013), emphasizes the importance of incorporating simple momentum for achieving rapid convergence in neural network learning. Although more complex momentum (or acceleration) algorithms have been proposed (Nesterov, 1983; Bubeck *et al.*, 2015; Liu & Wright, 2015; Van Scoy *et al.*, 2017; Allen-Zhu, 2018; Cyrus *et al.*, 2018; Jain *et al.*, 2018; Jin *et al.*, 2018) demonstrating faster convergence rates than plain SGD on standard classes of loss functions, Polyak’s original simple momentum update (Polyak, 1964) defies theory by remaining highly effective in practice and remains a popular choice for many applications. Despite several studies analyzing the performance of stochastic momentum methods (Flammarion & Bach, 2015; Loizou & Richtárik, 2017; Gadat *et al.*, 2018; Kidambi *et al.*, 2018; Yan *et al.*, 2018; Can *et al.*, 2019; Gitman *et al.*, 2019; Liu *et al.*, 2020; Loizou & Richtárik, 2020; Sebbouh *et al.*, 2021)

a gap persists between existing theoretical guarantees and their superior practical performance. We aim to bridge this gap by analyzing the properties of simple stochastic momentum methods in the context of quadratic optimization.

Given an $n \times d$ matrix \mathbf{A} and a length n vector \mathbf{b} , the linear least squares problem

$$\min_{\mathbf{x} \in \mathbb{R}^d} f(\mathbf{x}); \quad f(\mathbf{x}) := \frac{1}{2} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2 = \frac{1}{2} \sum_{i=1}^n |\mathbf{a}_i^\top \mathbf{x} - b_i|^2 = \sum_{i=1}^n f_i(\mathbf{x}) \quad (1.1)$$

is one of the most fundamental problems in optimization. One approach to solving (1.1) is Polyak's heavy ball momentum (HBM) (Polyak, 1964), also called 'standard' or 'classical' momentum. (HBM) updates the parameter estimate \mathbf{x}_k for the solution \mathbf{x}^* as

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \nabla f(\mathbf{x}_k) + \beta_k \mathbf{m}_k, \quad \mathbf{m}_{k+1} = \beta_k \mathbf{m}_k - \alpha_k \nabla f(\mathbf{x}_k),$$

where α_k and β_k are the step-size and momentum parameters, respectively. This is equivalent to the update

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \nabla f(\mathbf{x}_k) + \beta_k (\mathbf{x}_k - \mathbf{x}_{k-1}). \quad (\text{HBM})$$

The gradient of the objective (1.1) is easily computed to be $\nabla f(\mathbf{x}) = \mathbf{A}^\top (\mathbf{A}\mathbf{x} - \mathbf{b})$, and when $\mathbf{A}^\top \mathbf{A}$ has finite condition number $\kappa = \lambda_{\max} / \lambda_{\min}$, where λ_{\max} and λ_{\min} are the largest and smallest eigenvalues of $\mathbf{A}^\top \mathbf{A}$, (HBM) with properly chosen constant step-size and momentum parameters $\alpha_k = \alpha$ and $\beta_k = \beta$ provably attains the optimal linear rate

$$\|\mathbf{x}_k - \mathbf{x}^*\| \leq C_{\text{HBM}} \left(1 - \frac{1}{\sqrt{\kappa}}\right)^k \|\mathbf{x}_0 - \mathbf{x}^*\|, \quad C_{\text{HBM}} > 0.$$

When $\beta_k = 0$, (HBM) reduces to the standard gradient descent algorithm which, with the optimal choice of step-sizes α_k , converges at a sub-optimal linear rate

$$\|\mathbf{x}_k - \mathbf{x}^*\| \leq C_{\text{GD}} \left(1 - \frac{1}{\kappa}\right)^k \|\mathbf{x}_0 - \mathbf{x}^*\|, \quad C_{\text{GD}} > 0.$$

On large-scale problems, computing the gradient $\nabla f(\mathbf{x})$ can be prohibitively expensive. For problems such as (1.1), it is common to replace applications of the gradient with a minibatch stochastic gradient

$$\nabla f_{S_k}(\mathbf{x}) = \frac{1}{B} \sum_{j \in S_k} \frac{1}{p_j} \nabla f_j(\mathbf{x}),$$

where S_k contains B indices drawn independently with replacement from $\{1, 2, \dots, n\}$ where, at each draw, p_j is the probability that an index j is chosen. Note that this sampling strategy ensures $\mathbb{E}[\nabla f_{S_k}(\mathbf{x})] = \nabla f(\mathbf{x})$.

We denote by *minibatch-heavy ball momentum (Minibatch-HBM)* the following algorithm: starting from initial conditions $\mathbf{x}_1 = \mathbf{x}_0$, iterate until convergence

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \nabla f_{S_k}(\mathbf{x}_k) + \beta_k (\mathbf{x}_k - \mathbf{x}_{k-1}). \tag{Minibatch-HBM}$$

In the case of (1.1), the minibatch stochastic gradient can be written

$$\nabla f_{S_k}(\mathbf{x}_k) = \frac{1}{B} \sum_{j \in S_k} \frac{1}{p_j} \mathbf{a}_j (\mathbf{a}_j^\top \mathbf{x}_k - b_j), \tag{1.2}$$

where \mathbf{a}_j^\top is the j th row of \mathbf{A} and b_j is the j th entry of \mathbf{b} .

ASSUMPTION 1.1 Throughout, we will assume that, for some $\eta \geq 1$, the sampling probabilities are such that

$$\eta p_j \geq \frac{\|\mathbf{a}_j\|^2}{\|\mathbf{A}\|_F^2}, \quad j = 1, 2, \dots, n, \tag{1.3}$$

where $\|\cdot\|_F$ represents the matrix Frobenious norm.

REMARK 1.2 If rows are sampled proportionally to their squared norm $p_j = \|\mathbf{a}_j\|^2 / \|\mathbf{A}\|_F^2$, we have $\eta = 1$. If rows are sampled i.i.d. uniformly, $p_j = 1/n$ for $j = 1, \dots, n$, (1.3) is satisfied with $\eta = n \max_j \|\mathbf{a}_j\|^2 / \|\mathbf{A}\|_F^2$.

Applied to the problem (1.1), plain SGD ($B = 1, \beta = 0$) with an appropriately chosen step-size ($p_j = \|\mathbf{a}_j\|^2 / \|\mathbf{A}\|_F^2$) is equivalent to the randomized Kaczmarz (RK) algorithm if importance weighted sampling ($\eta = 1$) is used (Needell *et al.*, 2014). The standard version of RK extends the original cyclic Kaczmarz algorithm (Kaczmarz, 1937) and, as proved in Strohmer & Vershynin (2008), converges in a number of iterations scaling with $d\bar{\kappa}$, where $\bar{\kappa} = \lambda_{\text{ave}} / \lambda_{\text{min}}$ is the *smoothed* condition number of $\mathbf{A}^\top \mathbf{A}$, and λ_{ave} and λ_{min} are the average and smallest eigenvalues of $\mathbf{A}^\top \mathbf{A}$, respectively. Note the important relationship between the smoothed and the standard condition numbers:

$$\bar{\kappa} \leq \kappa \leq d\bar{\kappa}.$$

This relationship implies that, when $n \geq d\sqrt{\kappa}$, RK at least matches (up to constants) the performance of (HBM). If $\bar{\kappa} \ll \kappa$ or $n \gg d\sqrt{\kappa}$ then RK can significantly outperform (HBM), at least in terms of total number of row products.

While the number of row products of RK is reduced compared to (HBM), the number of iterations required to converge is increased. In practice, running times are not necessarily directly associated with the number of row products, and instead depend on other factors such as communication and memory access patterns. These costs often scale with the number of iterations, so it is desirable to understand whether the iteration complexity of (Minibatch-HBM) can be reduced to that of (HBM).

1.1 Contributions

We aim to make precise the observation that stochastic momentum affords acceleration in the minibatch setting. An illustration of this phenomenon is provided in Fig. 1. Our main theoretical result is a proof

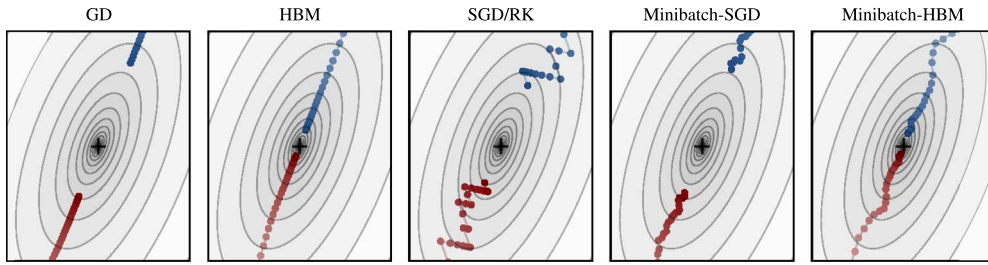


FIG. 1. Sample convergence trajectories for various iterative methods applied to a quadratic problem (1.1) with $n = 200$ and $d = 2$. Gradient descent with heavy ball momentum (HBM) allows for accelerated convergence over gradient descent (GD). SGD allows for lower per iteration costs, and the use of batching (Minibatch-SGD) reduces the variance of the iterates. While batching and momentum are often used simultaneously (Minibatch-HBM), convergence guarantees have remained elusive, even for quadratic objectives (1.1). In this paper we prove that, on such objectives, (Minibatch-HBM) converges linearly at the same rate as (HBM), provided the batch size is sufficiently large in a precise sense.

that the linear rate of convergence of (HBM) can be matched by (Minibatch-HBM), provided the batch size is larger than a critical size. Informally, this result can be summarized as follows:

THEOREM 1.3 Consider (Minibatch-HBM) applied to a strongly convex quadratic objective (1.1) with stochastic gradients (1.2) whose sampling probabilities satisfy (1.3) with parameter $\eta \geq 1$. Suppose that the minimizer \mathbf{x}^* satisfies $\mathbf{A}\mathbf{x}^* = \mathbf{b}$. Then, with the same fixed step-size and momentum parameters $\alpha_k = \alpha$, $\beta_k = \beta$ as (HBM), there exists a constant $C > 0$ such that, if κ is sufficiently large and $B \geq C\eta d \log(d)\bar{\kappa}\sqrt{\kappa}$, the (Minibatch-HBM) iterates converge in expected norm at least at a linear rate $1 - 1/\sqrt{\kappa}$.

For a more precise statement of Theorem 1.3, see Corollary 3.4.

REMARK 1.4 The bound for B does not depend directly on n , and in many situations $B \ll n$. In these cases, (Minibatch-HBM) offers a provable speedup over (HBM). Theorem 3.2 provides a more fine-grained relationship between the momentum and step-size parameters than Theorem 1.3. In particular, it shows that it is always possible to take $B < n$ and have (Minibatch-HBM) converge similar to (HBM), albeit at the cost of slowed convergence compared to the rate of convergence of (HBM) with the optimal α, β .

Owing to the equivalence between SGD on convex quadratics and the RK algorithm (Needell *et al.*, 2014) our convergence guarantees give a provable iteration complexity $\sqrt{\kappa}$ for RK-type algorithms. Our analysis method is quite general, and can be used to certify fast linear convergence for various forms of momentum beyond heavy ball momentum; in Appendix A we illustrate the generality of the approach by proving an analogous convergence result for a minibatch Nesterov's accelerated gradient method in the setting of linear regression.

1.2 Literature review

In the remainder of this section, we provide an overview of state-of-art existing results for row-sampling methods for solving (consistent) linear squares problems. A summary is given in Table 1.

Randomized Kaczmarz. A number of improvements to the standard RK algorithm have been proposed. Liu and Wright (Liu & Wright, 2015) introduce an accelerated randomized Kaczmarz (ARK) method that, through the use of Nesterov's acceleration, can achieve a faster rate of convergence

TABLE 1 *Runtime comparisons for row-sampling iterative methods for solving a consistent linear least squares problem (1.1) to constant accuracy when $\mathbf{A} \in \mathbb{R}^{n \times d}$ for large condition number κ . Constants and a logarithmic dependence on the accuracy parameter are suppressed. Here κ and $\bar{\kappa}$ are the regular and smoothed condition numbers of $\mathbf{A}^T \mathbf{A}$. Due to practical considerations such as parallelization, data movement, caching, energy efficiency, etc., the real-world cost of an iteration does not necessarily scale linearly with the number of row products*

Algorithm	Iterations	# row prods/iter.	References
(HBM)	$\sqrt{\kappa}$	n	Hestenes <i>et al.</i> (1952), Polyak (1964), Liesen & Strakoš (2013)
SGD/RK	$d\bar{\kappa}$	1	Strohmer & Vershynin (2008), Needell <i>et al.</i> (2014)
ARK (Minibatch-HBM)	$d\sqrt{\bar{\kappa}}$ $\sqrt{\kappa}$	1 $B \gtrsim d \log(d)\bar{\kappa}\sqrt{\kappa}$	Liu & Wright (2015) (this paper)

compared to RK. However, their rate is still sub-optimal compared to the rate attained by (HBM). Moreover, ARK is less able to take advantage of potential sparsity in the data matrix \mathbf{A} than the standard RK algorithm and (Minibatch-HBM). This issue is partially addressed by a special implementation of ARK for sparse matrices, but is still of concern for particularly sparse matrices.

Minibatching in the setting of the RK has been studied extensively in ‘small’ batch regimes (Needell & Tropp, 2014; Needell & Ward, 2016; Moorman *et al.*, 2020). These works view minibatching as a way to reduce the variance of iterates and improve on the standard RK algorithm. In general, the bounds for the convergence rates for such algorithms are complicated, but can improve on the convergence rate of RK by up to a factor of B in the best case. This ‘best case’ improvement, however, can only be attained for small B ; indeed, RK reduces to standard gradient descent in the deterministic gradient limit. In contrast to these works, we study minibatching in the RK method as a *necessary* algorithmic structure for unlocking the fast convergence rate of HBM.

Minibatch-HBM. Several recent works provide theoretical convergence guarantees for (Minibatch-HBM). Loizou & Richtárik (2017, 2020) show that (Minibatch-HBM) can achieve a linear rate of convergence for solving convex linear regression problems. However, the linear rate they show is slower than the rate of (deterministic) (HBM) in the same setting. Gitman *et al.* (2019) establish local convergence guarantees for the (Minibatch-HBM) method for general strongly convex functions and appropriate choice of the parameters α_k and β_k . Liu *et al.* (2020) show that (Minibatch-HBM) converges as fast as SGD for smooth strongly convex and nonconvex functions. Under the assumption that the stochastic gradients have uniformly bounded variance, Can *et al.* (2019) provide a number of convergence guarantees for stochastic HBM. In particular, it is shown that the same fast rate of convergence can be attained for full-batch quadratic objectives with bounded additive noise. The results of Can *et al.* (2019) however do not apply to the setting of RK, where the variance of the stochastic gradients necessarily grows proportionally to the squared norm of the full gradient.

Jain *et al.* (2018) demonstrate that (Minibatch-HBM) with a batch size of $B = 1$ provably fails to achieve faster convergence than SGD. They acknowledged that the favorable empirical results of (Minibatch-HBM), such as those found in Sutskever *et al.* (2013), should be seen as an ‘artifact’ of large minibatching, where the variance of stochastic gradients is sufficiently reduced that the deterministic convergence behavior of (HBM) dominates. *In this paper, we aim to precisely quantify this observation*

by providing a characterization of the minimal batch size required for (Minibatch-HBM) to achieve fast linear convergence comparable to that of (HBM).

In concurrent work, Lee *et al.* (2022) analyze the dynamics of (Minibatch-HBM) applied to quadratic objectives corresponding to a general class of random data matrices. Their results show that when the batch size is sufficiently large, (Minibatch-HBM) converges like its deterministic counterpart, but convergence is necessarily slower for smaller batch sizes. The batch size requirement of Lee *et al.* (2022) is a factor of κ better than what we obtain (see Theorem 1.3). However, while our analysis makes no assumptions on \mathbf{A} , Lee *et al.* (2022) requires certain invariance assumptions on the singular vectors of \mathbf{A} . It would be interesting to understand whether the extra factor of κ in our bound can be improved or whether is a necessary artifact of the lack of assumptions on \mathbf{A} .

Stochastic Nesterov’s Accelerated Gradient (SNAG). Several recent works have analyzed the theoretical convergence properties of stochastic Nesterov’s accelerated gradient (SNAG) methods and their variants in both strongly convex and nonconvex settings (Ghadimi & Lan, 2016; Can *et al.*, 2019; Vaswani *et al.*, 2019; Aybat *et al.*, 2020). Can *et al.* (2019) and Aybat *et al.* (2020) demonstrate accelerated convergence guarantees of SNAG method variants to a neighborhood of the solution for problems with uniformly bounded noise. Additionally, Ghadimi & Lan (2016) provide convergence guarantees for SNAG variants in nonconvex settings. Vaswani *et al.* (2019) show that SNAG methods achieve accelerated convergence rates to the solution for over-parameterized machine learning models under the assumption of the strong gradient growth condition, where the ℓ_2 norm of the stochastic gradients is assumed to be bounded by the norm of the gradient. Our contributions imply that the strong gradient growth conditions hold in the consistent under-parameterized least squares setting for the stochastic minibatch gradient, and demonstrating that this condition implies acceleration for (Minibatch-HBM), in addition to Nesterov momentum. Acceleration techniques have also been integrated with the variance reduction techniques to achieve optimal convergence rate guarantees for finite-sum problems (Frostig *et al.*, 2015; Defazio, 2016; Allen-Zhu, 2018; Lin *et al.*, 2018; Zhou *et al.*, 2019).

Ma *et al.* (2018) established critical batch size for SGD to retain the convergence rate of deterministic gradient descent method where as in this work we establish the critical batch size for (Minibatch-HBM) to retain the convergence properties of (HBM).

1.3 Notation

We denote vectors using lower case roman letters and matrices using upper case roman letters. We use $\mathbf{x} \in \mathbb{R}^d$ to denote the variables of the optimization problem and \mathbf{x}^* to denote the minimizer of $f(\mathbf{x})$. We use $\|\cdot\|$ to represent the Euclidean norm for vectors and operator norm for matrices and $\|\cdot\|_F$ to represent the matrix Frobenious norm. Throughout, \mathbf{A} will be an $n \times d$ matrix and \mathbf{b} a length n vector. The eigenvalues of $\mathbf{A}^T\mathbf{A}$ are denoted $\lambda_1, \lambda_2, \dots, \lambda_d$, and we write λ_{\max} , λ_{\min} , and λ_{ave} for the largest, smallest, and average eigenvalue of $\mathbf{A}^T\mathbf{A}$. The regular and smoothed condition numbers κ and $\bar{\kappa}$ of $\mathbf{A}^T\mathbf{A}$ are respectively defined as $\kappa = \lambda_{\max}/\lambda_{\min}$ and $\bar{\kappa} = \lambda_{\text{ave}}/\lambda_{\min}$. All logarithms are natural, and we denote complex numbers by i and Euler’s number $2.718\dots$ by e .

2. Preliminaries

In this section we provide an overview of (HBM) analysis and important statements from random matrix theory that are used in proving the convergence of (Minibatch-HBM).

2.1 Standard analysis of heavy ball momentum for quadratics

We review the standard analysis for heavy ball momentum (HBM) in the setting of strongly convex quadratic optimization problems (1.1) (see Recht (2010)). Here and henceforth, we take $\alpha_k = \alpha$ and $\beta_k = \beta$ as constants.

First, we re-write the (HBM) updates as

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha(\mathbf{A}^\top \mathbf{A} \mathbf{x}_k - \mathbf{A}^\top \mathbf{b}) + \beta(\mathbf{x}_k - \mathbf{x}_{k-1}) \quad (2.1)$$

so, by definition, the (HBM) updates satisfy

$$\begin{aligned} \mathbf{x}_{k+1} - \mathbf{x}^* &= \mathbf{x}_k - \alpha(\mathbf{A}^\top \mathbf{A} \mathbf{x}_k - \mathbf{A}^\top \mathbf{A} \mathbf{x}^*) + \beta(\mathbf{x}_k - \mathbf{x}_{k-1}) - \mathbf{x}^* \\ &= \mathbf{x}_k - \mathbf{x}^* - \alpha \mathbf{A}^\top \mathbf{A} (\mathbf{x}_k - \mathbf{x}^*) + \beta(\mathbf{x}_k - \mathbf{x}_{k-1}) \\ &= (\mathbf{I} - \alpha \mathbf{A}^\top \mathbf{A})(\mathbf{x}_k - \mathbf{x}^*) + \beta(\mathbf{x}_k - \mathbf{x}_{k-1} - \mathbf{x}^* + \mathbf{x}^*) \\ &= ((1 + \beta)\mathbf{I} - \alpha \mathbf{A}^\top \mathbf{A})(\mathbf{x}_k - \mathbf{x}^*) - \beta(\mathbf{x}_{k-1} - \mathbf{x}^*). \end{aligned}$$

This can be written more concisely as

$$\begin{bmatrix} \mathbf{x}_{k+1} - \mathbf{x}^* \\ \mathbf{x}_k - \mathbf{x}^* \end{bmatrix} = \underbrace{\begin{bmatrix} (1 + \beta)\mathbf{I} - \alpha \mathbf{A}^\top \mathbf{A} & -\beta \mathbf{I} \\ \mathbf{I} & \mathbf{0} \end{bmatrix}}_{\mathbf{T} = \mathbf{T}(\alpha, \beta)} \begin{bmatrix} \mathbf{x}_k - \mathbf{x}^* \\ \mathbf{x}_{k-1} - \mathbf{x}^* \end{bmatrix}, \quad (2.2)$$

where \mathbf{T} is the transition matrix taking us from the error vectors at steps k and $k - 1$ to the error vectors at steps $k + 1$ and k . Repeatedly applying this relation, we find

$$\begin{bmatrix} \mathbf{x}_{k+1} - \mathbf{x}^* \\ \mathbf{x}_k - \mathbf{x}^* \end{bmatrix} = \mathbf{T}^k \begin{bmatrix} \mathbf{x}_1 - \mathbf{x}^* \\ \mathbf{x}_0 - \mathbf{x}^* \end{bmatrix} \quad (2.3)$$

from which we obtain the error bound

$$\left\| \begin{bmatrix} \mathbf{x}_{k+1} - \mathbf{x}^* \\ \mathbf{x}_k - \mathbf{x}^* \end{bmatrix} \right\| \leq \|\mathbf{T}^k\| \left\| \begin{bmatrix} \mathbf{x}_1 - \mathbf{x}^* \\ \mathbf{x}_0 - \mathbf{x}^* \end{bmatrix} \right\|.$$

The assumption $\mathbf{x}_1 = \mathbf{x}_0$ allows us to write

$$\|\mathbf{x}_{k+1} - \mathbf{x}^*\| \leq \sqrt{2} \|\mathbf{T}^k\| \|\mathbf{x}_0 - \mathbf{x}^*\|.$$

The difficulty in analyzing (HBM) compared to plain gradient descent (when $\beta = 0$) lies in the fact $\|\mathbf{T}\|^k$ need not provide a useful upper bound for $\|\mathbf{T}^k\|$. Indeed, while $\|\mathbf{T}^k\| = \|\mathbf{T}\|^k$ for symmetric matrices, this is not necessarily the case for nonsymmetric matrices. To get around this issue, it is common to bound the spectral radius $\rho(\mathbf{T}) = \max_j \{|\lambda_j|\}$ and use Gelfand's formula $\rho(\mathbf{T}) = \lim_{k \rightarrow \infty} \|\mathbf{T}^k\|^{1/k}$ to derive the rate of convergence (Recht, 2010).

To bound the spectral radius of \mathbf{T} , note that \mathbf{T} is orthogonally similar to a block diagonal matrix consisting of 2×2 components $\{\mathbf{T}_j\}$. Specifically,

$$\mathbf{U}^{-1}\mathbf{T}\mathbf{U} = \begin{bmatrix} \mathbf{T}_1 & & & \\ & \mathbf{T}_2 & & \\ & & \ddots & \\ & & & \mathbf{T}_n \end{bmatrix}, \quad \text{where } \mathbf{T}_j = \begin{bmatrix} 1 + \beta - \alpha\lambda_j & -\beta \\ 1 & 0 \end{bmatrix} \quad (2.4)$$

for each $j = 1, 2, \dots, n$, \mathbf{U} is a certain orthogonal matrix (see Recht (2010)), and $\{\lambda_j\}$ are the eigenvalues of $\mathbf{A}^\top\mathbf{A}$. For each $j = 1, 2, \dots, n$, the eigenvalues z_j^\pm of \mathbf{T}_j are easily computed to be

$$z_j^\pm := \frac{1}{2} \left(1 + \beta - \alpha\lambda_j \pm \sqrt{(1 + \beta - \alpha\lambda_j)^2 - 4\beta} \right)$$

and are therefore nonreal if and only if

$$(1 + \beta - \alpha\lambda_j)^2 - 4\beta < 0. \quad (2.5)$$

In this case, the magnitude of both the eigenvalues of \mathbf{T}_j is

$$|z_j^\pm| = \frac{1}{2} \sqrt{(1 + \beta - \alpha\lambda_j)^2 + |(1 + \beta - \alpha\lambda_j)^2 - 4\beta|} = \sqrt{\beta}.$$

Here we have used that $|(1 + \beta - \alpha\lambda_j)^2 - 4\beta| = 4\beta - (1 + \beta - \alpha\lambda_j)^2$ whenever (2.5) holds. Therefore, provided

$$(1 + \beta - \alpha\lambda_j)^2 - 4\beta < 0 \quad \text{for all } j = 1, 2, \dots, n, \quad (2.6)$$

we have that

$$\rho(\mathbf{T}) = \max\{|z_j^\pm| : j = 1, \dots, n\} = \sqrt{\beta}. \quad (2.7)$$

We would like to choose $\sqrt{\beta} = \rho(\mathbf{T})$ as small as possible subject to the condition that (2.6) holds.¹ Note that (2.6) is equivalent to the condition

$$\frac{(1 - \sqrt{\beta})^2}{\lambda_j} < \alpha < \frac{(1 + \sqrt{\beta})^2}{\lambda_j} \quad \text{for all } j = 1, 2, \dots, n,$$

which we can rewrite as

$$\frac{(1 - \sqrt{\beta})^2}{\lambda_{\min}} < \alpha < \frac{(1 + \sqrt{\beta})^2}{\lambda_{\max}}. \quad (2.8)$$

¹ If (2.6) does not hold, then the larger of $|z_j^\pm|$ will be greater than $\sqrt{\beta}$.

Minimizing in β gives the condition $(1 - \sqrt{\beta})^2/\lambda_{\min} = \alpha = (1 + \sqrt{\beta})^2/\lambda_{\max}$ from which we determine

$$\sqrt{\alpha^*} = \frac{2}{\sqrt{\lambda_{\max}} + \sqrt{\lambda_{\min}}} \quad \text{and} \quad \sqrt{\beta^*} = \frac{\alpha(\lambda_{\max} - \lambda_{\min})}{4} = \frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1}. \tag{2.9}$$

As noted at the start of this section, this gives an asymptotic rate of convergence $\sqrt{\beta}$.

2.2 A closer look at the quadratic case

To derive a bound for stochastic (HBM) at finite k , it is desired to understand the eigendecomposition of the matrix \mathbf{T} from (2.4) more carefully. Thus, we might aim to diagonalize \mathbf{T} as

$$\mathbf{T} = \mathbf{UCDC}^{-1}\mathbf{U}^{-1} = (\mathbf{UC})\mathbf{D}(\mathbf{UC})^{-1}, \tag{2.10}$$

where \mathbf{U} is the previously described orthogonal matrix rotating \mathbf{T} into block diagonal form (2.4) and \mathbf{D} and \mathbf{C} are block-diagonal matrices with 2×2 blocks $\{\mathbf{D}_j\}$ and $\{\mathbf{C}_j\}$ where, for each $j = 1, \dots, d$, \mathbf{T}_j is diagonalized as $\mathbf{T}_j = \mathbf{C}_j\mathbf{D}_j\mathbf{C}_j^{-1}$. Given such a factorization, we would have $\mathbf{T}^k = (\mathbf{UC})\mathbf{D}^k(\mathbf{UC})^{-1}$. Then, since that \mathbf{U} is unitary, we would obtain the bound

$$\|\mathbf{T}^k\| \leq M(\alpha, \beta)\|\mathbf{D}\|^k = M(\alpha, \beta)(\sqrt{\beta})^k,$$

where $M(\alpha, \beta) = \|\mathbf{C}\|\|\mathbf{C}^{-1}\| = \|\mathbf{UC}\|\|\mathbf{UC}^{-1}\|$ is the condition number of the eigenvector matrix \mathbf{UC} .

However, if α and β are chosen as in (2.9), \mathbf{T} is defective (that is, does not have a complete basis of eigenvectors) and no such diagonalization exists.² To avoid this issue, we perturb the choices of α and β , and define, for some $\gamma \in (0, \lambda_{\min})$,

$$L = \lambda_{\max} + \gamma \quad \text{and} \quad \ell = \lambda_{\min} - \gamma. \tag{2.11}$$

Taking

$$\sqrt{\alpha} = \frac{2}{\sqrt{L} + \sqrt{\ell}} \quad \text{and} \quad \sqrt{\beta} = \frac{\alpha(L - \ell)}{4} = \frac{\sqrt{L/\ell} - 1}{\sqrt{L/\ell} + 1} \tag{2.12}$$

ensures that (2.6) holds and that \mathbf{T} is diagonalizable. Indeed, since we can write $z_j^\pm = a_j \pm ib_j$ for $a_j, b_j \in \mathbb{R}$ with $b_j \neq 0$, it is easily verified that the (up to a scaling of the eigenvectors) eigendecomposition $\mathbf{T}_j\mathbf{C}_j = \mathbf{C}_j\mathbf{D}_j$ for \mathbf{T}_j is

$$\mathbf{T}_j \begin{bmatrix} a_j + ib_j & a_j - ib_j \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} a_j + ib_j & a_j - ib_j \\ 1 & 1 \end{bmatrix} \begin{bmatrix} a_j + ib_j & \\ & a_j - ib_j \end{bmatrix}. \tag{2.13}$$

² In particular, the blocks \mathbf{T}_j corresponding to the smallest and largest eigenvalues each have only a single eigenvector.

We clearly have $\|\mathbf{D}\| = \max_j |z_j^\pm| = \sqrt{\beta}$, so the (HBM) iterates satisfy the convergence guarantee

$$\|\mathbf{x}_{k+1} - \mathbf{x}^*\| \leq \sqrt{2}M(\alpha, \beta) \left(\frac{\sqrt{L/\ell} - 1}{\sqrt{L/\ell} + 1} \right)^k \|\mathbf{x}_0 - \mathbf{x}^*\|. \tag{2.14}$$

Note that $M(\alpha, \beta)$, L and ℓ each depend on λ_{\max} , λ_{\min} and γ . The dependency of $M(\alpha, \beta)$ on these values is through \mathbf{C} , which, up to a unitary scaling by \mathbf{U} , is the eigenvector matrix of the transition matrix \mathbf{T} . We can bound $M(\alpha, \beta)$ by the following lemma.

LEMMA 2.1 For any $\gamma \in (0, \lambda_{\min})$ set $\ell = \lambda_{\min} - \gamma$ and $L = \lambda_{\max} + \gamma$ and choose $\alpha = 4/(\sqrt{L} + \sqrt{\ell})^2$ and $\sqrt{\beta} = \alpha(L - \ell)/4$. Let $M(\alpha, \beta) = \|\mathbf{UC}\| \|(\mathbf{UC})^{-1}\|$, where \mathbf{UC} is the eigenvector matrix for \mathbf{T} . Then,

$$M(\alpha, \beta) \leq \frac{4}{\alpha\sqrt{\gamma(\gamma + \lambda_{\max} - \lambda_{\min})}}.$$

Proof. In order to bound $M(\alpha, \beta)$, we note that the block diagonal structure of \mathbf{C} implies that $\|\mathbf{C}\| = \max\{\|\mathbf{C}_j\| : j = 1, \dots, d\}$ and $\|\mathbf{C}^{-1}\| = \max\{\|\mathbf{C}_j^{-1}\| : j = 1, \dots, d\}$.

By construction, the specified values of α and β ensure that $(1 + \beta - \alpha\lambda_j)^2 < 4\beta$ for all $i = 1, 2, \dots, d$, i.e. condition (2.5). This implies $|z_j^\pm| = \sqrt{\beta}$, so we easily compute

$$\|\mathbf{C}_j\|^2 \leq \|\mathbf{C}_j\|_{\mathbb{F}}^2 = |z_j^+|^2 + |z_j^-|^2 + 1 + 1 = 2\beta + 2 \leq 4.$$

By direct computation, we find

$$\mathbf{C}_j^{-1} = \begin{bmatrix} a_j + ib_j & a_j - ib_j \\ 1 & 1 \end{bmatrix}^{-1} = \frac{1}{2ib_j} \begin{bmatrix} 1 & -a_j + ib_j \\ -1 & a_j + ib_j \end{bmatrix}.$$

To bound $\|\mathbf{C}_j^{-1}\|$ we first note that the condition (2.5) is also equivalent to

$$\ell = \frac{(1 - \sqrt{\beta})^2}{\alpha} < \lambda_j < \frac{(1 + \sqrt{\beta})^2}{\alpha} = L,$$

which implies that

$$4\beta - (1 + \beta - \alpha\lambda_j)^2 = \alpha^2(\lambda_j - \ell)(L - \lambda_j) \geq \alpha^2\gamma(L - \lambda_{\min}).$$

Since $L \geq \lambda_{\max}$ we therefore have the bound

$$\|\mathbf{C}_j^{-1}\|^2 \leq \|\mathbf{C}_j^{-1}\|_{\mathbb{F}}^2 = \frac{\|\mathbf{C}_j\|_{\mathbb{F}}^2}{4|b_j|^2} = \frac{2(1 + \beta)}{4\beta - (1 + \beta - \alpha\lambda_j)^2} \leq \frac{4}{\alpha^2\gamma(L - \lambda_{\min})}.$$

The result follows by combining the above expressions. □

2.3 Lemmas from nonasymptotic random matrix theory

Before we prove our main result, we need to introduce tools from nonasymptotic random matrix theory which are crucial components of the proof.

PROPOSITION 2.2 Consider a finite sequence $\{\mathbf{W}_k\}$ of independent random matrices with common dimension $d_1 \times d_2$. Assume that

$$\mathbb{E}[\mathbf{W}_i] = \mathbf{0} \quad \text{and} \quad \|\mathbf{W}_i\| \leq W \quad \text{for each index } i$$

and introduce the random matrix

$$\mathbf{Z} = \mathbf{W}_1 + \dots + \mathbf{W}_k.$$

Let $v(\mathbf{Z})$ be the matrix variance statistic of the sum:

$$v(\mathbf{Z}) = \max \left\{ \left\| \sum_i \mathbb{E}[\mathbf{W}_i \mathbf{W}_i^T] \right\|, \left\| \sum_i \mathbb{E}[\mathbf{W}_i^T \mathbf{W}_i] \right\| \right\}.$$

Then,

$$\begin{aligned} \mathbb{E}[\|\mathbf{Z}\|] &\leq \sqrt{2v(\mathbf{Z}) \log(d_1 + d_2)} + \frac{1}{3}W \log(d_1 + d_2), \\ \sqrt{\mathbb{E}[\|\mathbf{Z}\|^2]} &\leq \sqrt{2ev(\mathbf{Z}) \log(d_1 + d_2)} + 4eW \log(d_1 + d_2). \end{aligned}$$

The bound on $\mathbb{E}[\|\mathbf{Z}\|]$ is Theorem 6.1.1 in Tropp (2015), and the bound on $\sqrt{\mathbb{E}[\|\mathbf{Z}\|^2]}$ follows from equation 6.1.6 in Tropp (2015) and the fact $\sqrt{\mathbb{E}[\max_i \|\mathbf{W}_i\|^2]} \leq W$. Equation 6.1.6 in Tropp (2015) comes from applying Theorem A.1 in Chen et al. (2012) to the Hermitian dilation of \mathbf{Z} . Under the stated conditions, the logarithmic dependence on the dimension is necessary (Tropp, 2015).

We will also use a theorem on products of random matrices from Huang et al. (2021):

PROPOSITION 2.3 (Corollary 5.4 in Huang et al. (2021)). Consider an independent sequence of $d \times d$ random matrices $\mathbf{X}_1, \dots, \mathbf{X}_k$, and form the product

$$\mathbf{Z} = \mathbf{X}_k \mathbf{X}_{k-1} \dots \mathbf{X}_1.$$

Assume $\|\mathbb{E}[\mathbf{X}_i]\| \leq q_i$ and $\mathbb{E}[\|\mathbf{X}_i - \mathbb{E}\mathbf{X}_i\|^2]^{1/2} \leq \sigma_i q_i$ for $i = 1, \dots, k$. Let $Q = \prod_{i=1}^n q_i$ and $v = \sum_{i=1}^k \sigma_i^2$. Then,

$$\mathbb{E}[\|\mathbf{Z}\|] \leq Q \exp \left(\sqrt{2v \max\{2v, \log(d)\}} \right).$$

3. Main results

We are now prepared to analyze (Minibatch-HBM) applied to strongly convex least squares problems of the form (1.1). We begin by considering the case of consistent linear systems, i.e. systems for which

\mathbf{b} is in the column span of \mathbf{A} . In Section 3.1 we then provide an analogous result for inconsistent least squares problems.

We begin with a useful technical lemma that bounds the batch size required to ensure that a certain random matrix is near its expectation.

LEMMA 3.1 Define $\mathbf{W}_j = B^{-1}(-p_j^{-1} \mathbf{a}_j \mathbf{a}_j^\top + \mathbf{A}^\top \mathbf{A})$ and let

$$\mathbf{W} = \sum_{j \in S} \mathbf{W}_j,$$

where S is a list of B indices each chosen independently according to (1.3). Then, $\sqrt{\mathbb{E}[\|\mathbf{W}\|^2]} \leq \delta$ provided

$$B \geq 8\eta \log(2d) \max \{ \|\mathbf{A}\|_F^2 \|\mathbf{A}\|^2 \delta^{-2}, (4\|\mathbf{A}\|_F^4 \delta^{-2})^{1/2} \}.$$

Proof. Since the sampling probabilities satisfy (1.3), $\|\mathbf{a}_j \mathbf{a}_j^\top\| = \|\mathbf{a}_j\|^2 \leq \eta p_j \|\mathbf{A}\|_F^2$. Then, since $\eta \geq 1$,

$$\|\mathbf{W}_j\| \leq \frac{1}{B} \left(\frac{1}{p_j} \|\mathbf{a}_j \mathbf{a}_j^\top\| + \|\mathbf{A}\|^2 \right) \leq \frac{\eta \|\mathbf{A}\|_F^2 + \|\mathbf{A}\|^2}{B} \leq \frac{2\eta \|\mathbf{A}\|_F^2}{B}.$$

Next, observe that

$$\mathbf{W}_j^\top \mathbf{W}_j = \frac{1}{B^2} \left(\frac{\|\mathbf{a}_j\|^2}{p_j^2} \mathbf{a}_j \mathbf{a}_j^\top - \frac{1}{p_j} \mathbf{a}_j \mathbf{a}_j^\top \mathbf{A}^\top \mathbf{A} - \frac{1}{p_j} \mathbf{A}^\top \mathbf{A} \mathbf{a}_j \mathbf{a}_j^\top + (\mathbf{A}^\top \mathbf{A})^2 \right). \tag{3.1}$$

Using that $\mathbb{E}[(p_j)^{-1} \mathbf{a}_j \mathbf{a}_j^\top] = \mathbf{A}^\top \mathbf{A}$ and $\|\mathbf{a}_j\|^2 \leq \eta p_j \|\mathbf{A}\|_F^2$, we find that

$$\mathbb{E}[\mathbf{W}_j^\top \mathbf{W}_j] = \frac{1}{B^2} \left(\sum_{i=1}^n \frac{\|\mathbf{a}_i\|^2}{p_i} \mathbf{a}_i \mathbf{a}_i^\top - (\mathbf{A}^\top \mathbf{A})^2 \right) \leq \frac{1}{B^2} (\eta \|\mathbf{A}\|_F^2 \mathbf{A}^\top \mathbf{A} - (\mathbf{A}^\top \mathbf{A})^2).$$

Here we write $\mathbf{M}_1 \preceq \mathbf{M}_2$ if $\mathbf{M}_2 - \mathbf{M}_1$ is positive semi-definite. Note that $\mathbf{M}_1 \preceq \mathbf{M}_2$ implies the largest eigenvalue of \mathbf{M}_2 is greater than the largest eigenvalue of \mathbf{M}_1 . Therefore, using that $\mathbf{0} \preceq \mathbb{E}[\mathbf{W}_j^\top \mathbf{W}_j]$ followed by the fact $\mathbf{0} \preceq \mathbf{A}^\top \mathbf{A} \preceq \|\mathbf{A}\|_F^2 \mathbf{I}$,

$$\|\mathbb{E}[\mathbf{W}_j^\top \mathbf{W}_j]\| \leq \frac{1}{B^2} \|(\eta \|\mathbf{A}\|_F^2 \mathbf{I} - \mathbf{A}^\top \mathbf{A}) \mathbf{A}^\top \mathbf{A}\| \leq \frac{\eta \|\mathbf{A}\|_F^2 \|\mathbf{A}\|^2}{B^2}.$$

Thus, since \mathbf{W}_j is symmetric and the samples in S are i.i.d., we obtain a bound for the variance statistic

$$v(\mathbf{W}) = \left\| \sum_{j \in S} \mathbb{E}[\mathbf{W}_j^\top \mathbf{W}_j] \right\| \leq \sum_{j \in S} \left\| \mathbb{E}[\mathbf{W}_j^\top \mathbf{W}_j] \right\| \leq \frac{\eta \|\mathbf{A}\|_F^2 \|\mathbf{A}\|^2}{B}. \tag{3.2}$$

Together with (3.1) and (3.2), Theorem 2.2 implies

$$\sqrt{\mathbb{E}[\|\mathbf{W}\|^2]} \leq \left(\frac{2e\eta\|\mathbf{A}\|_{\mathbb{F}}^2\|\mathbf{A}\|^2\log(2d)}{B} \right)^{1/2} + \frac{4e(2\eta\|\mathbf{A}\|_{\mathbb{F}}^2)\log(2d)}{B}. \tag{3.3}$$

The first term is bounded by $\delta/2$ when

$$B \geq 8e\eta\|\mathbf{A}\|_{\mathbb{F}}^2\|\mathbf{A}\|^2\log(2d)\delta^{-2}$$

whereas the second term is bounded by $\delta/2$ when

$$B \geq 16e\eta\|\mathbf{A}\|_{\mathbb{F}}^2\log(2d)\delta^{-1}.$$

The result follows by taking the max of these quantities. □

Our main result is the following theorem.

THEOREM 3.2 Consider (Minibatch-HBM) applied to a strongly convex quadratic objective (1.1) with stochastic gradients (1.2) whose sampling probabilities satisfy (1.3). Fix parameters α and β satisfying $(1 - \sqrt{\beta})^2/\lambda_{\min} < \alpha < (1 + \sqrt{\beta})^2/\lambda_{\max}$. For any $k^* > 1$ choose

$$B \geq 16e\eta\log(2d) \max \left\{ \frac{\|\mathbf{A}\|_{\mathbb{F}}^2\|\mathbf{A}\|^2\alpha^2M(\alpha, \beta)^2k^*}{\beta\log(k^*)}, \left(\frac{2\|\mathbf{A}\|_{\mathbb{F}}^4\alpha^2M(\alpha, \beta)^2k^*}{\beta\log(k^*)} \right)^{1/2} \right\}.$$

Then, for all $k > 0$, assuming that the minimizer \mathbf{x}^* satisfies $\mathbf{A}\mathbf{x}^* = \mathbf{b}$, the (Minibatch-HBM) iterates satisfy

$$\mathbb{E}[\|\mathbf{x}_k - \mathbf{x}^*\|] \leq \sqrt{2}M(\alpha, \beta) \max\{d, (k^*)^{k/k^*}\}(\sqrt{\beta})^k\|\mathbf{x}_0 - \mathbf{x}^*\|,$$

where $M(\alpha, \beta)$ is the condition number of eigenvector matrix \mathbf{UC} for \mathbf{T} defined in (2.10).

REMARK 3.3 For $k \leq k^*$, $(k^*)^{k/k^*} \leq \max\{e, k\}$. Thus, (Minibatch-HBM) converges at a rate nearly $\sqrt{\beta}$ for $k \leq \max\{d, k^*\}$. For $k > \max\{d, k^*\}$, the convergence is still linear for k^* sufficiently large, but at a rate slower than $\sqrt{\beta}$. Specifically, (Minibatch-HBM) converges at a rate $(\sqrt{\beta})^{1-\delta}$ where $\delta = 2\log(k^*)/(k^*\log(1/\beta))$.

Proof. Due to assumption of consistency, we have that $\mathbf{A}\mathbf{x}^* = \mathbf{b}$. Therefore, we can write the minibatch gradient (1.2) as

$$\nabla f_{S_k}(\mathbf{x}_k) = \frac{1}{B} \sum_{j \in S_k} \frac{1}{p_j} \mathbf{a}_j \mathbf{a}_j^\top (\mathbf{x}_k - \mathbf{x}^*). \tag{3.4}$$

Define the random matrix

$$\mathbf{M}_{S_k} = \frac{1}{B} \sum_{j \in S_k} \frac{1}{p_j} \mathbf{a}_j \mathbf{a}_j^\top$$

and note that $\mathbb{E}[\mathbf{M}_{S_k}] = \mathbf{A}^\top \mathbf{A}$. Then, analogously to (2.2), the (Minibatch-HBM) iterates satisfy the recurrence

$$\begin{bmatrix} \mathbf{x}_{k+1} - \mathbf{x}^* \\ \mathbf{x}_k - \mathbf{x}^* \end{bmatrix} = \underbrace{\begin{bmatrix} (1 + \beta)\mathbf{I} - \alpha\mathbf{M}_{S_k} & -\beta\mathbf{I} \\ \mathbf{I} & \mathbf{0} \end{bmatrix}}_{\mathbf{Y}_{S_k} = \mathbf{Y}_{S_k}(\alpha, \beta)} \begin{bmatrix} \mathbf{x}_k - \mathbf{x}^* \\ \mathbf{x}_{k-1} - \mathbf{x}^* \end{bmatrix},$$

where \mathbf{Y}_{S_k} is the stochastic transition matrix at iteration k . After k iterations, the error satisfies

$$\left\| \begin{bmatrix} \mathbf{x}_{k+1} - \mathbf{x}^* \\ \mathbf{x}_k - \mathbf{x}^* \end{bmatrix} \right\| \leq \|\mathbf{Y}_{S_k} \mathbf{Y}_{S_{k-1}} \cdots \mathbf{Y}_{S_1}\| \left\| \begin{bmatrix} \mathbf{x}_1 - \mathbf{x}^* \\ \mathbf{x}_0 - \mathbf{x}^* \end{bmatrix} \right\|,$$

so our goal is to bound the norm of the random matrix $\mathbf{Y}_{S_k} \mathbf{Y}_{S_{k-1}} \cdots \mathbf{Y}_{S_1}$.

This is a product of random matrices, so we may hope to apply Proposition 2.3. However, while $\mathbb{E}[\mathbf{Y}_{S_i}] = \mathbf{T}$, where \mathbf{T} is the deterministic transition matrix (2.2), $\|\mathbb{E}[\mathbf{Y}_{S_i}]\|$ is not necessarily bounded by $\sqrt{\beta}$. Thus, to apply Proposition 2.3 we will instead consider

$$\mathbf{Z}_k = (\mathbf{UC})^{-1} \mathbf{Y}_{S_k} \mathbf{Y}_{S_{k-1}} \cdots \mathbf{Y}_{S_1} (\mathbf{UC}) = \mathbf{X}_{S_k} \mathbf{X}_{S_{k-1}} \cdots \mathbf{X}_{S_1}, \tag{3.5}$$

where $\mathbf{X}_{S_i} = (\mathbf{UC})^{-1} \mathbf{Y}_{S_i} (\mathbf{UC})$ and \mathbf{U} and \mathbf{C} are the matrices from (2.10). Then, as desired,

$$\|\mathbb{E}[\mathbf{X}_{S_i}]\| = \|(\mathbf{UC})^{-1} \mathbb{E}[\mathbf{Y}_{S_i}] (\mathbf{UC})\| = \|(\mathbf{UC})^{-1} \mathbf{T} (\mathbf{UC})\| = \|\mathbf{D}\| = \sqrt{\beta}.$$

Thus, if we can guarantee that the variances $\{\sqrt{\mathbb{E}[\|\mathbf{X}_{S_i} - \mathbb{E}[\mathbf{X}_{S_i}]\|^2]}\}$ are not too large, we can apply Proposition 2.3 to obtain a rate similar to (HBM).

Towards this end, note that

$$\mathbf{Y}_{S_i} - \mathbb{E}[\mathbf{Y}_{S_i}] = \sum_{j \in S_i} \frac{\alpha}{B} \begin{bmatrix} -p_j^{-1} \mathbf{a}_j \mathbf{a}_j^\top + \mathbf{A}^\top \mathbf{A} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} = \alpha \sum_{j \in S_i} \begin{bmatrix} \mathbf{W}_j & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix},$$

where \mathbf{W}_j is as in Lemma 3.1. This and the fact that $\|\mathbf{X}_{S_i} - \mathbb{E}[\mathbf{X}_{S_i}]\| = \|(\mathbf{UC})^{-1} (\mathbf{Y}_{S_i} - \mathbb{E}[\mathbf{Y}_{S_i}]) (\mathbf{UC})\|$ implies

$$\sqrt{\mathbb{E}[\|\mathbf{X}_{S_i} - \mathbb{E}[\mathbf{X}_{S_i}]\|^2]} \leq M(\alpha, \beta) \sqrt{\mathbb{E}[\|\mathbf{Y}_{S_i} - \mathbb{E}[\mathbf{Y}_{S_i}]\|^2]} \leq \alpha M(\alpha, \beta) \sqrt{\mathbb{E}[\|\mathbf{W}\|^2]}, \tag{3.6}$$

where $\mathbf{W} = \sum_{j \in S_i} \mathbf{W}_j$. Using Lemma 3.1 and (3.6), we have $\sqrt{\mathbb{E}[\|\mathbf{X}_{S_i} - \mathbb{E}[\mathbf{X}_{S_i}]\|^2]} \leq \delta$ provided that the batch size B satisfies

$$B \geq 8\epsilon\eta \log(2d) \max \{ \|\mathbf{A}\|_{\mathbb{F}}^2 \|\mathbf{A}\|^2 \alpha^2 M(\alpha, \beta)^2 \delta^{-2}, (4\|\mathbf{A}\|_{\mathbb{F}}^4 \alpha^2 M(\alpha, \beta)^2 \delta^{-2})^{1/2} \}.$$

Applying Proposition 2.3 to the product (3.5) with the parameters

$$q_i = \sqrt{\beta}, \quad \sigma_i = \delta/\sqrt{\beta} \quad \text{and} \quad v = \sum_{i=1}^k \sigma_i^2 = k\delta^2/\beta$$

gives the bound

$$\mathbb{E}[\|\mathbf{Z}_k\|] \leq (\sqrt{\beta})^k \exp\left(\sqrt{2v \max\{2v, \log(d)\}}\right).$$

Set $\delta^2 = \beta \log(k^*)/(2k^*)$ so that $2v = k \log(k^*)/k^*$. This gives the desired expression for B . Moreover, we then have

$$\mathbb{E}[\|\mathbf{Z}_k\|] \leq (\sqrt{\beta})^k \exp(\max\{2v, \log(d)\}) = (\sqrt{\beta})^k \max\{(k^*)^{k/k^*}, d\}.$$

Thus, we find that

$$\mathbb{E}[\|\mathbf{Y}_{S_k} \mathbf{Y}_{S_{k-1}} \cdots \mathbf{Y}_{S_1}\|] = \mathbb{E}[\|(\mathbf{UC})\mathbf{Z}_k(\mathbf{UC})^{-1}\|] \leq M(\alpha, \beta)(\sqrt{\beta})^k \max\{(k^*)^{k/k^*}, d\},$$

giving the desired bound for the iterates. □

The expressions for the required batch size in Theorem 3.2 is somewhat complicated, but can be simplified in the large condition number limit.

COROLLARY 3.4 Fix $c \in (0, 2)$. There exist parameters α and β and a constant $C > 0$ (depending on c) such that, for all κ sufficiently large, the (Minibatch-HBM) iterates converge in expected norm at least at a linear rate $1 - c/\sqrt{\kappa}$ provided that $B \geq C\eta d \log(d)\bar{\kappa}\sqrt{\kappa}$.

Proof. Suppose $\gamma = c_1\lambda_{\min}$ for $c_1 \in (0, 1)$. Then, using that the definitions of L and ℓ from (2.11) imply that $L/\ell = (\lambda_{\max} + c_1\lambda_{\min})/(\lambda_{\min} - c_1\lambda_{\min}) = \kappa/(1 - c_1) + c_1/(1 - c_1)$, we have

$$\sqrt{\beta} = \frac{\sqrt{L/\ell} - 1}{\sqrt{L/\ell} + 1} = 1 - \frac{2}{\sqrt{L/\ell} + 1} = 1 - \frac{2\sqrt{1 - c_1}}{\sqrt{\kappa} + c_1 + \sqrt{1 - c_1}}.$$

Now, set $k^*/\log(k^*) = \sqrt{\kappa}/c_2$ for some $c_2 > 0$. Then, for

$$1 - \delta = 1 - \frac{\log(k^*)}{k^* \log(1/\sqrt{\beta})} = 1 - \frac{c_2}{\sqrt{\kappa} \log(1/\sqrt{\beta})} = 1 - \frac{c_2}{2\sqrt{1 - c_1}} + \mathcal{O}(\kappa^{-1})$$

we have

$$(k^*)^{1/k^*} \sqrt{\beta} = (\sqrt{\beta})^{1-\delta} = 1 - \frac{2\sqrt{1 - c_1} - c_2}{\sqrt{\kappa}} + \mathcal{O}(\kappa^{-1}).$$

Therefore, if we take $c_1 = 1 - ((c + 6)/8)^2$ and $c_2 = (2 - c)/4$ we have that, for κ sufficiently large,

$$(k^*)^{1/k^*} \sqrt{\beta} = 1 - \frac{(c + 2)/2}{\sqrt{\kappa}} + o(1) \leq 1 - \frac{c}{\sqrt{\kappa}}.$$

Using Lemma 2.1 we have that

$$(\alpha\lambda_{\min}M(\alpha, \beta))^2 \leq \frac{4(\lambda_{\min})^2}{\gamma(\gamma + \lambda_{\max} - \lambda_{\min})} \leq \frac{\lambda_{\min}}{\gamma} \frac{4\lambda_{\min}}{\lambda_{\max} - \lambda_{\min}} = \mathcal{O}(\kappa^{-1}).$$

This, with the fact that $\beta = \mathcal{O}(1)$, implies

$$\frac{\|\mathbf{A}\|_{\mathbb{F}}^2 \|\mathbf{A}\|^2 \alpha^2 M(\alpha, \beta)^2 k^*}{\beta \log(k^*)} = \frac{(d\bar{\kappa}\lambda_{\min})(\kappa\lambda_{\min})\alpha^2 M(\alpha, \beta)^2 (4\sqrt{\kappa})}{\beta} = \mathcal{O}(d\bar{\kappa}\sqrt{\kappa})$$

and

$$\left(\frac{\|\mathbf{A}\|_{\mathbb{F}}^4 \alpha^2 M(\alpha, \beta)^2 k^*}{\beta \log(k^*)} \right)^{1/2} = \left(\frac{(d\bar{\kappa}\lambda_{\min})^2 \alpha^2 M(\alpha, \beta)^2 (4\sqrt{\kappa})}{\beta} \right)^{1/2} = \mathcal{O}(d\bar{\kappa}/\sqrt[4]{\kappa}).$$

Thus, the bound on B becomes $B \geq \mathcal{O}(\eta d \log(d)\bar{\kappa}\sqrt{\kappa})$. □

3.1 Inconsistent least squares problems

Our results can be extended to inconsistent systems. On such systems, the stochastic gradients at the optimal point \mathbf{x}^* need not equal zero, even though $\nabla f(\mathbf{x}^*) = \mathbf{0}$. As a result, stochastic gradient methods will only converge to within a *convergence horizon* of the minimizer \mathbf{x}^* .

REMARK 3.5 As shown in (Needell (2010), Theorem 2.1), the RK iterates converge to within an expected radius $\sqrt{d\bar{\kappa}}\sigma$ of the least squares solution, where $\sigma = \max_i |r_i|/\|\mathbf{a}_i\|$. The minibatch-RK algorithm from Moorman *et al.* (2020) improves the convergence horizon by roughly a factor of \sqrt{B} .

THEOREM 3.6 In the setting of Theorem 3.2, define $\mathbf{r} = \mathbf{Ax}^* - \mathbf{b}$ and $\sigma = \max_i |r_i|/\|\mathbf{a}_i\|$. Let the batch size B satisfy the conditions in Theorem 3.2. Then, provided k^* is chosen so that $\delta = 2 \log(k^*)/(k^* \log(1/\beta)) < 1$, the (Minibatch-HBM) iterates satisfy

$$\mathbb{E}[\|\mathbf{x}_k - \mathbf{x}^*\|] \leq \sqrt{2}M(\alpha, \beta) \max\{d, (k^*)^{k/k^*}\}(\sqrt{\beta})^k \|\mathbf{x}_0 - \mathbf{x}^*\| + R,$$

where

$$R \leq \frac{\alpha M(\alpha, \beta)(d+1)}{1 - (\sqrt{\beta})^{1-\delta}} \left(\left(\frac{2\eta \|\mathbf{A}\|_{\mathbb{F}}^2 \log(d+1) \|\mathbf{r}\|^2}{B} \right)^{1/2} + \frac{\eta \|\mathbf{A}\|_{\mathbb{F}}^2 \log(d+1) \sigma}{3B} \right).$$

REMARK 3.7 The term in R containing $\|\mathbf{r}\|$ scales with $1/\sqrt{B}$ whereas the term containing σ scales with $1/B$. Thus, when B is large, the term in R involving σ becomes small relative to the term involving $\|\mathbf{r}\|$.

REMARK 3.8 In the large κ limit considered in Corollary 3.4, $k^*/\log(k^*) = \mathcal{O}(\sqrt{\kappa})$ and $1/(1 - (\sqrt{\beta})^{1-\delta}) = \mathcal{O}(\sqrt{\kappa})$. Thus, $(k^* + 1)/(1 - (\sqrt{\beta})^{1-\delta}) = \mathcal{O}(\kappa)$.

Proof of Theorem 3.6. Since $\mathbf{b} = \mathbf{A}\mathbf{x}^* - \mathbf{r}$, our stochastic gradients for inconsistent systems satisfy

$$\nabla f_{S_k}(\mathbf{x}_k) = \frac{1}{|S_k|} \sum_{i \in S_k} \left(\frac{1}{p_i} \mathbf{a}_i \mathbf{a}_i^\top (\mathbf{x}_k - \mathbf{x}^*) + \frac{r_i}{p_i} \mathbf{a}_i \right) = \mathbf{M}_{S_k} (\mathbf{x}_k - \mathbf{x}^*) + \mathbf{r}_{S_k},$$

where \mathbf{M}_{S_k} is as in the proof of Theorem 3.2, and define

$$\mathbf{r}_{S_k} = \frac{1}{|S_k|} \sum_{i \in S_k} \frac{r_i}{p_i} \mathbf{a}_i.$$

We therefore find the iterates satisfy the update formula

$$\begin{bmatrix} \mathbf{x}_{k+1} - \mathbf{x}^* \\ \mathbf{x}_k - \mathbf{x}^* \end{bmatrix} = \underbrace{\begin{bmatrix} (1 + \beta)\mathbf{I} - \alpha\mathbf{M}_{S_k} & -\beta\mathbf{I} \\ \mathbf{I} & \mathbf{0} \end{bmatrix}}_{\mathbf{Y}_{S_k} = \mathbf{Y}_{S_k}(\alpha, \beta)} \begin{bmatrix} \mathbf{x}_k - \mathbf{x}^* \\ \mathbf{x}_{k-1} - \mathbf{x}^* \end{bmatrix} + \alpha \begin{bmatrix} \mathbf{r}_{S_k} \\ \mathbf{0} \end{bmatrix}.$$

Thus, after k iterations, the error satisfies

$$\begin{bmatrix} \mathbf{x}_{k+1} - \mathbf{x}^* \\ \mathbf{x}_k - \mathbf{x}^* \end{bmatrix} \leq \left(\prod_{i=1}^k \mathbf{Y}_{S_i} \right) \begin{bmatrix} \mathbf{x}_1 - \mathbf{x}^* \\ \mathbf{x}_0 - \mathbf{x}^* \end{bmatrix} + \alpha \sum_{j=1}^k \left(\prod_{i=j+1}^k \mathbf{Y}_{S_i} \right) \begin{bmatrix} \mathbf{r}_{S_j} \\ \mathbf{0} \end{bmatrix}.$$

The first term is identical to the case $\mathbf{r} = 0$, so we have

$$\frac{R}{\alpha} \leq \mathbb{E} \left[\left\| \sum_{j=1}^k \left(\prod_{i=j+1}^k \mathbf{Y}_{S_i} \right) \begin{bmatrix} \mathbf{r}_{S_j} \\ \mathbf{0} \end{bmatrix} \right\| \right] \leq \sum_{j=1}^k \mathbb{E} \left[\left\| \left(\prod_{i=j+1}^k \mathbf{Y}_{S_i} \right) \right\| \right] \mathbb{E} \left[\left\| \begin{bmatrix} \mathbf{r}_{S_j} \\ \mathbf{0} \end{bmatrix} \right\| \right].$$

Here we have used the triangle inequality and definition of operator norm followed by the linearity of expectation and independence of the minibatch draws across iterations.

As in the proof of Theorem 3.2,

$$\mathbb{E} \left[\left\| \mathbf{Y}_{S_k} \mathbf{Y}_{S_{k-1}} \cdots \mathbf{Y}_{S_{j+1}} \right\| \right] \leq M(\alpha, \beta) \max\{k - j, (k^*)^{(k-j)/k^*}\} (\sqrt{\beta})^{k-j}.$$

Therefore, assuming all minibatch draws $\{S_j\}$ are identically distributed, we have

$$\begin{aligned} \frac{R}{\alpha} &\leq \left(\sum_{j=1}^k M(\alpha, \beta) \max\{d, (k^*)^{(k-j)/k^*}\} (\sqrt{\beta})^{k-j} \right) \mathbb{E}[\|\mathbf{r}_{S_1}\|] \\ &\leq M(\alpha, \beta) \left(\sum_{j=0}^{k-1} \max\{d, (k^*)^{j/k^*}\} (\sqrt{\beta})^j \right) \mathbb{E}[\|\mathbf{r}_{S_1}\|] \\ &\leq M(\alpha, \beta) \left(\sum_{j=0}^{k-1} d(\sqrt{\beta})^j + \sum_{j=0}^{k-1} (k^*)^{j/k^*} (\sqrt{\beta})^j \right) \mathbb{E}[\|\mathbf{r}_{S_1}\|] \\ &= M(\alpha, \beta) \left(d \frac{1 - (\sqrt{\beta})^k}{1 - \sqrt{\beta}} + \frac{1 - (k^*)^{k/k^*} (\sqrt{\beta})^k}{1 - (k^*)^{1/k^*} \sqrt{\beta}} \right) \mathbb{E}[\|\mathbf{r}_{S_1}\|]. \end{aligned}$$

Now, note that, by assumption,³ $(k^*)^{1/k^*} \sqrt{\beta} = (\sqrt{\beta})^{1-\delta} < 1$. Thus,

$$d \frac{1 - (\sqrt{\beta})^k}{1 - \sqrt{\beta}} + \frac{1 - ((k^*)^{1/k^*} \sqrt{\beta})^k}{1 - (k^*)^{1/k^*} \sqrt{\beta}} \leq \frac{d + 1}{1 - (\sqrt{\beta})^{1-\delta}}.$$

It remains to bound $\mathbb{E}[\|\mathbf{r}_{S_1}\|]$, and to do so we again turn to the matrix Bernstein inequality. Define the sum

$$\mathbf{Z} = \sum_{i \in S_k} \frac{r_i}{p_i} \mathbf{a}_i = \mathbf{X}_1 + \cdots + \mathbf{X}_B$$

and note that, with $\sigma = \max_i |r_i| / \|\mathbf{a}_i\|$, and the assumption (1.3) on the sampling probabilities,

$$\|\mathbf{X}_i\| \leq \frac{\|\mathbf{a}_i\| |r_i|}{\|\mathbf{a}_i\|^2 / (\eta \|\mathbf{A}\|_F^2)} \leq \eta \sigma \|\mathbf{A}\|_F^2.$$

By direct computation we also observe that

$$\left\| \sum_{i \in S_k} \mathbb{E}[\mathbf{X}_i \mathbf{X}_i^\top] \right\| \leq \left\| \sum_{i \in S_k} \mathbb{E} \left[\frac{\eta \|\mathbf{A}\|_F^2 r_i^2}{\|\mathbf{a}_i\|^2 p_i} \mathbf{a}_i \mathbf{a}_i^\top \right] \right\| = \eta B \|\mathbf{A}\|_F^2 \sum_{i=1}^n r_i^2 = \eta B \|\mathbf{A}\|_F^2 \|\mathbf{r}\|^2$$

and, since $\left\| \sum_{i \in S_k} r_i^2 \mathbf{a}_i \mathbf{a}_i^\top / \|\mathbf{a}_i\|^2 \right\| \leq \sum_{i \in S_k} r_i^2$, that

$$\left\| \sum_{i \in S_k} \mathbb{E}[\mathbf{X}_i \mathbf{X}_i^\top] \right\| \leq \left\| \sum_{i \in S_k} \mathbb{E} \left[\frac{\eta \|\mathbf{A}\|_F^2 r_i^2}{\|\mathbf{a}_i\|^2 p_i} \mathbf{a}_i \mathbf{a}_i^\top \right] \right\| \leq \eta B \|\mathbf{A}\|_F^2 \sum_{i=1}^n r_i^2 = \eta B \|\mathbf{A}\|_F^2 \|\mathbf{r}\|^2.$$

³ Without this assumption we still have a (messy) bound for R .

Therefore, applying Theorem 2.2 we obtain the bound

$$\mathbb{E}[\|\mathbf{r}_{S_k}\|] = \frac{1}{B} \mathbb{E}[\|\mathbf{Z}\|] \leq \sqrt{\frac{2\eta\|\mathbf{r}\|^2\|\mathbf{A}\|_F^2 \log(d+1)}{B}} + \frac{\eta\sigma\|\mathbf{A}\|_F^2 \log(d+1)}{3B}.$$

Combining everything gives the desired result. \square

4. Numerical results

We conduct numerical experiments on quadratic objectives (1.1) to illustrate the performance of (Minibatch-HBM). Throughout this section, we use the value

$$B^* = \frac{16e\|\mathbf{A}\|_F^2\|\mathbf{A}\|^2 \log(2d)\alpha^2}{\beta \log(1/\beta)}$$

as a *heuristic* for the batch size needed to observe linear convergence at a rate similar to that of (HBM). This value is obtained from Theorem 3.2 by making several simplifying assumptions. Specifically, we drop the dependence on $M(\alpha, \beta)$ which results from a change of basis followed by a return to the original basis (which we believe is likely an artifact of our analysis approach) and replace $k^*/\log(k^*)$ by $1/\log(1/\beta) = \mathcal{O}(\sqrt{\kappa})$.

In all experiments we use $n = 10^6$, $d = 10^2$ and set $\gamma = \lambda_{\min}/10^3$. Each experiment is repeated 100 times and the median and 5th to 95th percentile range for each algorithm/parameter choice are plotted.

4.1 Row-norm and uniform sampling

In this example, we study the dependence of (Minibatch-HBM) on the sampling probabilities. Our bounds are sharpest when $p_j \propto \|\mathbf{a}_j\|^2$, but in practice it is common to use uniform sampling $p_j = 1/n$ to avoid the need for computing row-norms which requires accessing the entire data matrix. We take $\mathbf{A} = \mathbf{D}\mathbf{G}$, where \mathbf{G} is an $n \times d$ matrix whose entries are independently 1 with probability 1/10 or 0 with probability 9/10 and \mathbf{D} is an $n \times n$ diagonal matrix whose diagonal entries are 1 with probability 9/10 and 10 with probability 1/10. Thus, uniform sampling probabilities $p_j = 1/n$ satisfy (1.3) provided $\eta \geq n \max_j \|\mathbf{a}_j\|^2 / \|\mathbf{A}\|_F^2 \approx 23$. We use a planted solution \mathbf{x}^* with i.i.d. standard normal entries.

In Fig. 2 we show the convergence of (Minibatch-HBM) with row norm sampling and uniform sampling at several values of B . As expected, row norm sampling works better than uniform sampling for a fixed value of B . However, since the norms of rows are not varying too significantly, the convergence rates are still comparable. See Needell *et al.* (2014) for a further discussion on sampling probabilities in the context of RK and SGD.

4.2 Sensitivity to batch size

The fact (Minibatch-HBM) exhibits accelerated convergence is an artifact of batching, and we expect different behavior at different batch sizes. In fact, we have already observed this phenomenon on the previous example. In Theorem 3.2, we provide an upper bound on the required batch size depending on spectral properties of $\mathbf{A}^\top \mathbf{A}$ such as $\bar{\kappa} = \lambda_{\text{ave}}/\lambda_{\min}$ and $\kappa = \lambda_{\max}/\lambda_{\min}$. To study whether the stated dependence on such quantities is reasonable, we construct a series of synthetic problems with prescribed spectra.

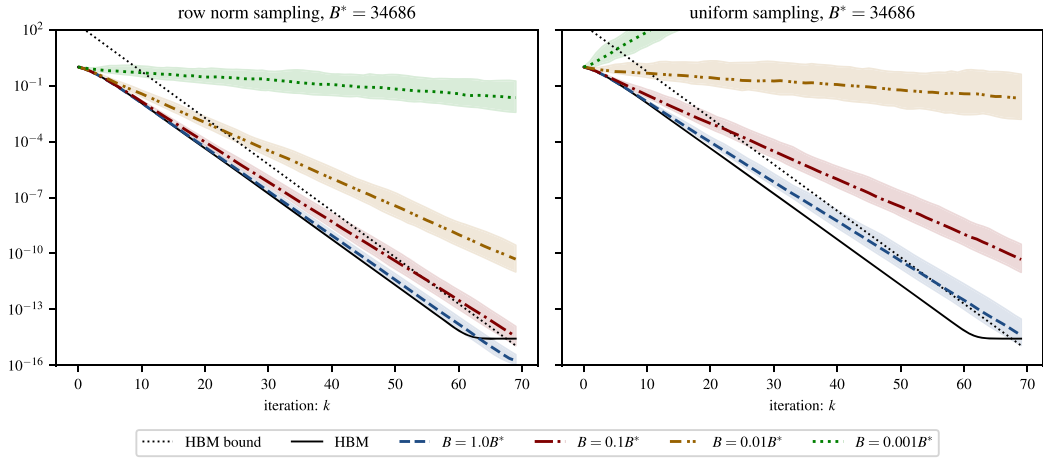


FIG. 2. Median and 5th to 95th percentile error norm $\|\mathbf{x}_k - \mathbf{x}^*\|$ of (Minibatch-HBM) for row norm sampling and uniform sampling at varying values of batch size B . For reference, we also show the convergence of (HBM) and the (HBM) bound (2.14).

We construct problems $\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^T$ by choosing the singular vectors \mathbf{U} and \mathbf{V} uniformly at random, and selecting singular values $\{\sigma_i\}$ with exponential or algebraic decay. For exponential decay we use the squared singular values⁴

$$\sigma_j^2 = 1 + \left(\frac{j-1}{d-1}\right) (\kappa - 1)\rho^{d-j}, \quad j = 1, 2, \dots, d, \tag{4.1}$$

and for algebraic decay we use the squared singular values

$$\sigma_j^2 = 1 + \left(\frac{j-1}{d-1}\right)^\rho (\kappa - 1), \quad j = 1, 2, \dots, d. \tag{4.2}$$

In both cases, the condition number of the $\mathbf{A}^T\mathbf{A}$ is κ and ρ determines how fast the singular values of \mathbf{A} decay. We again use a planted solution \mathbf{x}^* with i.i.d. standard normal entries.

In Figs 3 and 4 we report the results of our experiments. Here we consider consistent equations with condition numbers $\kappa = 10, 30, 100$. For each value of κ , we generate two problems according to (4.1) with $\rho = 0.1$ and $\rho = 0.8$ and two problems according to (4.2) with $\rho = 2$ and $\rho = 1$. In Fig. 3 we run (Minibatch-HBM) (row norm sampling) with $B = cB^*$ for $c = 10^{-3}, 10^{-2}, 10^{-1}, 10^0$ and show the convergence as a function of the iterations k . In Fig. 4 we run (Minibatch-HBM) for a fixed number of iterations, and show the convergence as a function of the batch size B . This empirically illustrates that when the batch size is near B^* , the rate of convergence is nearly that of (HBM).

⁴ This spectrum is often referred to as the ‘model problem’ in numerical analysis and is commonly used to study the convergence of iterative methods for solving linear systems of equations (Strakos, 1991; Strakos & Greenbaum, 1992).

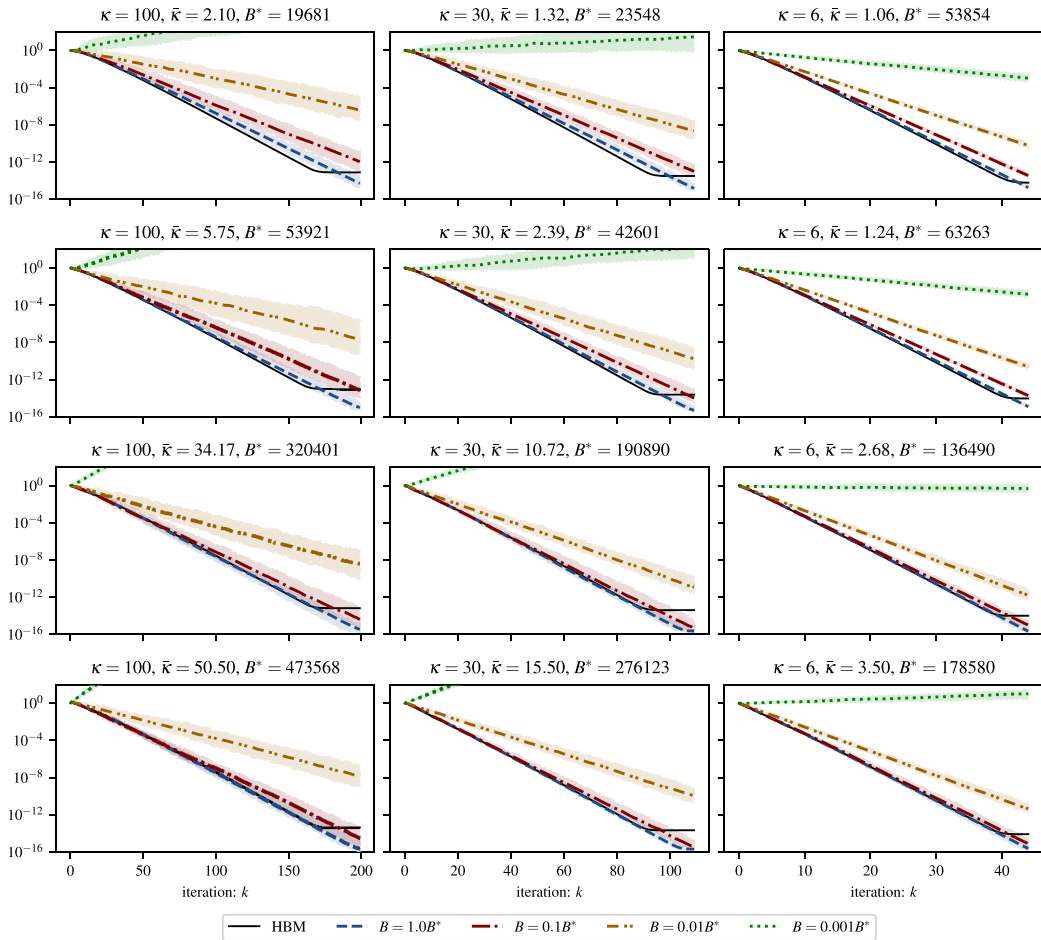


FIG. 3. Median and 5th to 95th percentile error norm $\|x_k - x^*\|$ of (Minibatch-HBM) for varying values of batch size B on problems with a range of κ and $\bar{\kappa}$. For reference, we also show the convergence of (HBM).

4.3 Inconsistent systems

For inconsistent systems, stochastic gradients at the optimum will not be zero and convergence is only possible to within the so-called *convergence horizon*. In this experiment we sample A as in the previous experiment using (4.1) with $\kappa = 50$ and $\rho = 0.5$. We take $b = Ax + \epsilon$, where x is has i.i.d. standard normal entries and ϵ is drawn uniformly from the hypersphere of radius 10^{-5} . Thus, the minimum residual norm $\|b - Ax^*\|$ is around 10^{-5} . We use several different values of B .

The results of the experiment are shown in Fig. 5. For larger batch sizes, the convergence horizon of (Minibatch-HBM) gets smaller. In particular, when the batch size is increased by a factor of 100, the convergence horizon is decreased by about a factor of 10. This aligns with the intuition that the convergence horizon should depend on \sqrt{B} . For reference, we also show the convergence for standard RK. Note RK requires more iterations to converge, although each iteration involves significantly less

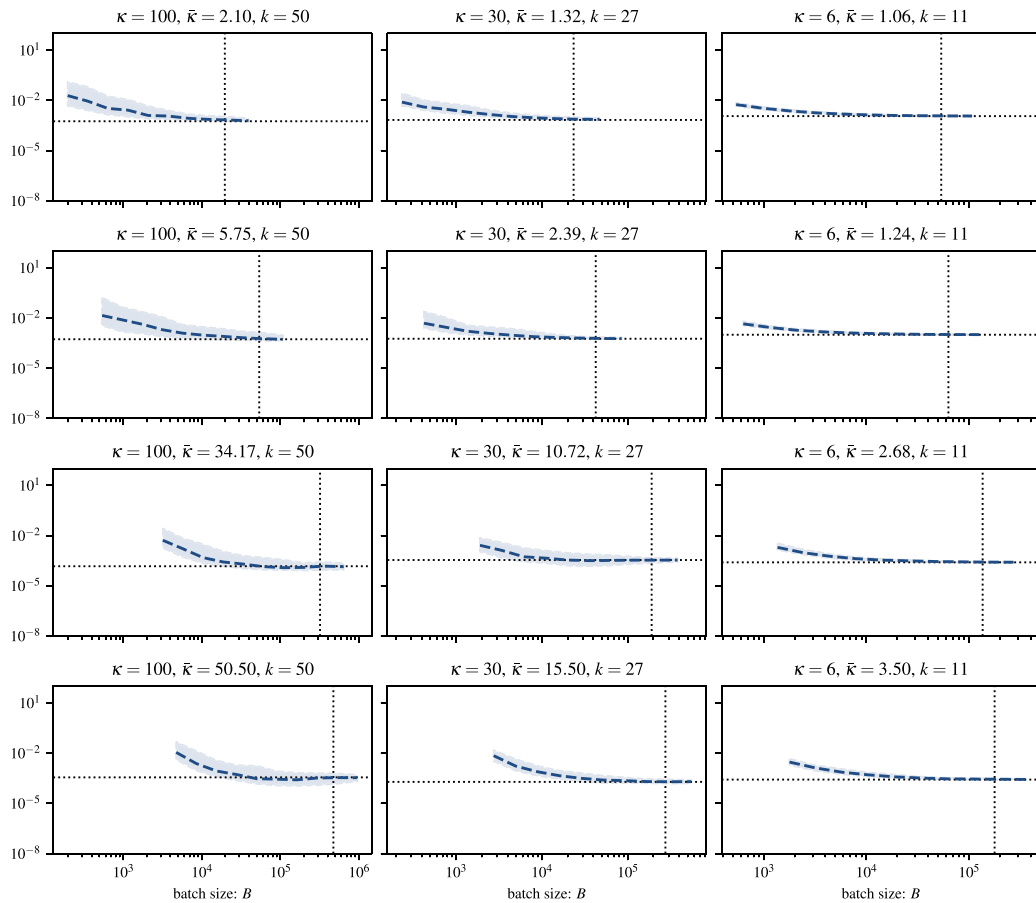


FIG. 4. Median and 5th to 95th percentile error norm $\|\mathbf{x}_k - \mathbf{x}^*\|$ of (Minibatch-HBM) for varying values of batch size B on problems with a range of κ and $\bar{\kappa}$. The horizontal dotted lines indicate the accuracy of (HBM), and the vertical dotted lines indicate $B = B^*$.

computation.⁵ We also note that, while the error in the iterates stagnates at different points, the value of the objective function is quite similar in all cases, nearly matching the residual norm of the true least squares solution.

4.4 Computational tomography

One of the most prevalent applications of Kaczmarz-like methods is in tomographic image reconstruction, notably in medical imaging. In X-ray tomography (e.g. CT scans), X-rays are passed through an object and the intensity of resulting X-ray beam is measured. This process is repeated for numerous known orientations of the X-ray beams relative to the object of interest. With a sufficient number of

⁵ While (Minibatch-HBM) uses significantly more floating point operations than RK, the runtime to fixed accuracy (for all batch sizes tested) was actually significantly lower than RK due to vectorized operations. Since runtime is highly system dependent, we do not provide actual timings.

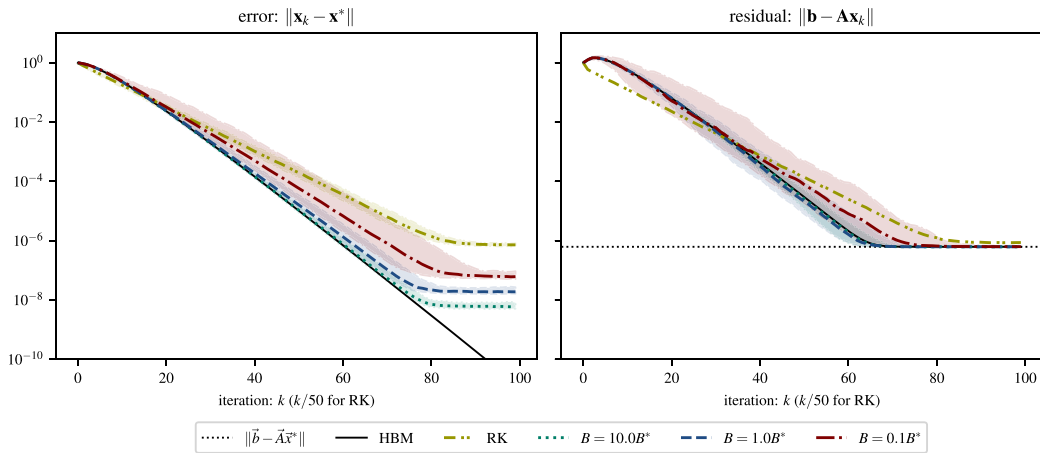


FIG. 5. Median and 5th to 95th percentile error norm and residual norm of (Minibatch-HBM) for varying values of batch size B on an inconsistent problem. For reference, we also show the convergence of (HBM) as well as optimal residual norm.

measurements, it is possible to reconstruct a ‘slice’ of the interior of the object of interest. In theory, this reconstruction involves solving a large, sparse consistent linear system.

In this example we consider the performance of (Minibatch-HBM) on a tomography problem corresponding to a parallel beam geometry scanner with 128 sensor pixels. Measurements are taken for 360 degrees of rotation at half-degree increments, totaling 720 measurements. The goal is to reconstruct a 64×64 pixel image. This results in a measurement matrix of dimensions $(720 \cdot 128) \times (64 \cdot 64) = 92160 \times 4096$, which we construct using the ASTRA toolbox (van Aarle *et al.*, 2015).

We employ a planted solution of a walnut, which we aim to recover from the resulting measurement data. Uniform sampling is employed due to the roughly similar norms of all rows. The step-size α and momentum parameter β are chosen so that (HBM) converges at a reasonable rate and so that $B^* = 407$ is not too large relative to n . In Fig. 6 we report the convergence of (HBM) and (Minibatch-HBM), along with the resulting images recovered by the algorithms after $k = 500$ iterations.

5. Conclusion

We provided a first analysis of accelerated convergence of minibatch heavy ball momentum method for quadratics using standard choices of the momentum step-sizes. Our proof method involves a refined quantitative analysis of the convergence proof for (deterministic) heavy ball momentum, combined with matrix concentration results for sums and products of independent matrices. Our proof technique is general, and also can be used to verify the accelerated convergence of a minibatch version of Nesterov’s acceleration for quadratics, using the constant step-size parameters suggested in Nesterov’s original paper. An interesting direction for future work is to combine the simple minibatch momentum algorithm with an adaptive gradient update such as AdaGrad (McMahan & Streeter, 2010; Duchi *et al.*, 2011; Ward *et al.*, 2019; Défossez *et al.*, 2020), to potentially learn the momentum parameters α_k and β_k adaptively. Such an analysis would also shed light on the convergence of ADAM (Kingma & Adam, 2014), an extension of momentum which combines adaptive gradients and momentum in a careful way to achieve state-of-the-art performance across various large-scale optimization problems.

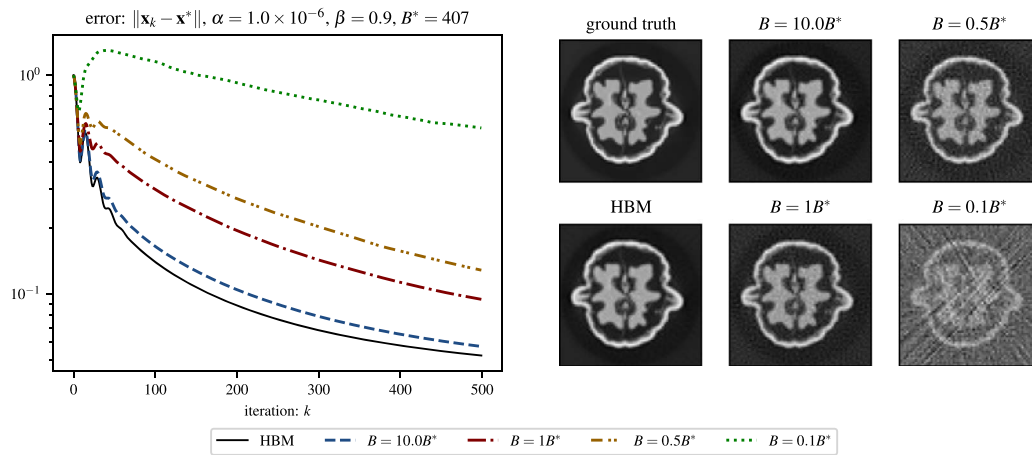


FIG. 6. Error $\|x_k - x^*\|$ of (Minibatch-HBM) on a problem from computational tomography at varying batch sizes, and the resulting images of the interior of a walnut recovered after $k = 500$ iterations.

Acknowledgements

We thank Qijia Jiang, Stephen Wright and Qian Zuo for helpful comments during the preparation of this manuscript.

Funding

R.B. was supported by NSF DMS 2324643. T.C. was supported by NSF DGE 1762114. R.W. was partially supported by AFOSR MURI FA9550-19-1-0005, NSF DMS 1952735, NSF HDR 1934932 and NSF CCF 2019844.

REFERENCES

- ALLEN-ZHU, Z. (2018) Katyusha: The first direct acceleration of stochastic gradient methods. *J. Mach. Learn. Res.*, **18**, 1–51.
- AYBAT, N. S., FALLAH, A., GURBUZBALABAN, M. & OZDAGLAR, A. (2020) Robust accelerated gradient methods for smooth strongly convex functions. *SIAM J. Optim.*, **30**, 717–751.
- BUBECK, S., LEE, Y. T. & SINGH, M. (2015) A geometric alternative to Nesterov’s accelerated gradient descent. arXiv preprint, arXiv:1506.08187.
- CAN, B., GURBUZBALABAN, M. & ZHU, L. (2019) Accelerated linear convergence of stochastic momentum methods in Wasserstein distances. *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research* (K. Chaudhuri & R. Salakhutdinov eds). PMLR, pp. 891–901. A6
- CHEN, R. Y., GITTENS, A. & TROPP, J. A. (2012) The masked sample covariance estimator: an analysis using matrix concentration inequalities. *Infer. Inference*, **1**, 2–20.
- CYRUS, S., HU, B., VAN SCOY, B. & LESSARD, L. (2018) A robust accelerated optimization algorithm for strongly convex functions. *2018 Annual American Control Conference (ACC)*. IEEE, pp. 1376–1381.
- DEFAZIO, A. (2016) A simple practical accelerated method for finite sums. *Adv. Neural Inform. Process. Syst.*, **29**, 676–684.
- DÉFOSSEZ, A., BOTTU, L., BACH, F. & USUNIER, N. (2020) A simple convergence proof of adam and adagrad.

- DUCHI, J., HAZAN, E. & SINGER, Y. (2011) Adaptive subgradient methods for online learning and stochastic optimization. *J. Mach. Learn. Res.*, **12**, 2121–2159.
- FLAMMARION, N. & BACH, F. (2015) From averaging to acceleration, there is only a step-size. *Conference on Learning Theory*. PMLR, pp. 658–695.
- FROSTIG, R., GE, R., KAKADE, S. & SIDFORD, A. (2015) Un-regularizing: approximate proximal point and faster stochastic algorithms for empirical risk minimization. *International Conference on Machine Learning*. PMLR, pp. 2540–2548.
- GADAT, S., PANLOUP, F. & SAADANE, S. (2018) Stochastic heavy ball. *Electr. J. Stat.*, **12**, 461–529.
- GHADIMI, S. & LAN, G. (2016) Accelerated gradient methods for nonconvex nonlinear and stochastic programming. *Math. Program.*, **156**, 59–99.
- GITMAN, I., LANG, H., ZHANG, P. & LIN, X. (2019) Understanding the role of momentum in stochastic gradient methods. *Adv. Neural Inform. Process. Syst.*, **32**, 9601–9611.
- HESTENES, M. R. & STIEFEL, E. (1952) Methods of conjugate gradients for solving linear systems. *J. Res. Natl. Bur. Stand.*, **49**, 409–436.
- HUANG, D., NILES-WEED, J., TROPP, J. A. & WARD, R. (2021) Matrix concentration for products. *Found. Comput. Math.*, **22**, 1–33.
- JAIN, P., KAKADE, S. M., KIDAMBI, R., NETRAPALLI, P. & SIDFORD, A. (2018) Accelerating stochastic gradient descent for least squares regression. *Conference on Learning Theory*. PMLR, pp. 545–604.
- JIN, C., NETRAPALLI, P. & JORDAN, M. I. (2018) Accelerated gradient descent escapes saddle points faster than gradient descent. *Conference on Learning Theory*. PMLR, pp. 1042–1085.
- KACZMARZ, S. M. (1937) Angenäherte auflösung von systemen linearer gleichungen. **35**, 355–357.
- KIDAMBI, R., NETRAPALLI, P., JAIN, P. & KAKADE, S. (2018) On the insufficiency of existing momentum schemes for stochastic optimization. *The 2018 Information Theory and Applications Workshop (ITA)*. IEEE, pp. 1–9.
- KINGMA, D. P. & ADAM, J. B. (2014) A method for stochastic optimization. arXiv preprint, arXiv:1412.6980.
- KRIZHEVSKY, A., SUTSKEVER, I. & HINTON, G. E. (2012) Imagenet classification with deep convolutional neural networks. *Adv. Neural Inform. Process. Syst.*, **25**, 1097–1105.
- LEE, K., CHENG, A., PAQUETTE, E. & PAQUETTE, C. (2022) Trajectory of mini-batch momentum: Batch size saturation and convergence in high dimensions. *Adv. Neural Inform. Process. Syst.*, **35**, 36944–36957.
- LIESEN, J. & STRAKOŠ, Z. (2013) Krylov subspace methods: principles and analysis. *Numerical Mathematics and Scientific Computation*, 1st ed edn. Oxford: Oxford University Press.
- LIN, H., MAIRAL, J. & HARCHAOU, Z. (2018) Catalyst acceleration for first-order convex optimization: from theory to practice. *J. Mach. Learn. Res.*, **18**, 7854–7907.
- LIU, J. & WRIGHT, S. J. (2015) An accelerated randomized Kaczmarz algorithm. *Math. Comput.*, **85**, 153–178.
- LIU, Y., GAO, Y. & YIN, W. (2020) An improved analysis of stochastic gradient descent with momentum. *Adv. Neural Inform. Process. Syst.*, **33**, 18261–18271.
- LOIZOU, N. & RICHTÁRIK, P. (2017) Linearly convergent stochastic heavy ball method for minimizing generalization error. arXiv preprint, arXiv:1710.10737.
- LOIZOU, N. & RICHTÁRIK, P. (2020) Momentum and stochastic momentum for stochastic gradient, Newton, proximal point and subspace descent methods. *Comput. Optim. Appl.*, **77**, 653–710.
- MA, S., BASSILY, R. & BELKIN, M. (2018) The power of interpolation: Understanding the effectiveness of SGD in modern over-parametrized learning. *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research* (J. Dy & A. Krause eds). PMLR, pp. 3325–3334.
- MCMAHAN, B. & STREETER, M. (2010) Adaptive bound optimization for online convex optimization. *COLT*. Madison, WI: Omnipress, pp. 244–256.
- MOORMAN, J. D., TU, T. K., MOLITOR, D. & NEEDELL, D. (2020) Randomized Kaczmarz with averaging. *BIT Numer. Math.*, **61**, 337–359.
- NEEDELL, D. (2010) Randomized Kaczmarz solver for noisy linear systems. *BIT Numer. Math.*, **50**, 395–403.

- NEEDELL, D., SREBRO, N. & WARD, R. (2014) Stochastic gradient descent, weighted sampling, and the randomized Kaczmarz algorithm. *NIPS'14*. Cambridge, MA, USA: MIT Press, pp. 1017–1025.
- NEEDELL, D. & TROPP, J. A. (2014) Paved with good intentions: Analysis of a randomized block Kaczmarz method. *Linear Algebra Appl.*, **441**, 199–221.
- NEEDELL, D. & WARD, R. (2016) Batched stochastic gradient descent with weighted sampling. *International Conference Approximation Theory*. Cham, Switzerland: Springer, pp. 279–306.
- NESTEROV, Y. (1983) A method of solving a convex programming problem with convergence rate $o(1/k^2)$. *Soviet Math. Doklady*, **27**, 372–376.
- NESTEROV, Y. (2013) *Introductory Lectures on Convex Optimization: A Basic Course*. New York, NY: Springer.
- POLYAK, B. T. (1964) Some methods of speeding up the convergence of iteration methods. *USSR Comput. Math. Math. Phys.*, **4**, 1–17.
- RECHT, B. (2010) Cs726-lyapunov analysis and the heavy ball method. Department of Computer Sciences. University of Wisconsin–Madison.
- SEBBOUH, O., GOWER, R. M. & DEFAZIO, A. (2021) Almost sure convergence rates for stochastic gradient descent and stochastic heavy ball. *Conference on Learning Theory*. PMLR, pp. 3935–3971.
- STRAKOS, Z. (1991) On the real convergence rate of the conjugate gradient method. *Linear Algebra Appl.*, **154–156**, 535–549.
- STRAKOS, Z. & GREENBAUM, A. (1992) *Open Questions in the Convergence Analysis of the Lanczos Process for the Real Symmetric Eigenvalue Problem*. University of Minnesota.
- STROHMER, T. & VERSHYNIN, R. (2008) A randomized Kaczmarz algorithm with exponential convergence. *J. Fourier Anal. Appl.*, **15**, 262–278.
- SUTSKEVER, I., MARTENS, J., DAHL, G. & HINTON, G. (2013) On the importance of initialization and momentum in deep learning. In *International Conference on Machine Learning*, pp. 1139–1147. PMLR.
- TROPP, J. A. (2015) An introduction to matrix concentration inequalities. *Found. Trends Mach. Learn.*, **8**, 1–230.
- VAN AARLE, W., PALENSTIJN, W. J., DE BEENHOUWER, J., ALTANTZIS, T., BALS, S., BATENBURG, K. J. & SIJBERS, J. (2015) The ASTRA toolbox: A platform for advanced algorithm development in electron tomography. *Ultramicroscopy*, **157**, 35–47.
- VAN SCOY, B., FREEMAN, R. A. & LYNCH, K. M. (2017) The fastest known globally convergent first-order method for minimizing strongly convex functions. *IEEE Control Syst. Lett.*, **2**, 49–54.
- VASWANI, S., BACH, F. & SCHMIDT, M. (2019) Fast and faster convergence of sgd for over-parameterized models and an accelerated perceptron. *The 22nd International Conference on Artificial Intelligence and Statistics*. PMLR, pp. 1195–1204.
- WARD, R., WU, X. & BOTTOU, L. (2019) Adagrad stepsizes: Sharp convergence over nonconvex landscapes. *International Conference on Machine Learning*. PMLR, pp. 6677–6686.
- YAN, Y., YANG, T., LI, Z., LIN, Q. & YANG, Y. (2018) A unified analysis of stochastic momentum methods for deep learning. arXiv preprint, arXiv:1808.10396.
- ZHOU, K., DING, Q., SHANG, F., CHENG, J., LI, D. & LUO, Z.-Q. (2019) Direct acceleration of saga using sampled negative momentum. In *The 22nd International Conference on Artificial Intelligence and Statistics*. PMLR, pp. 1602–1610.

Appendix A. Analysis of Nesterov’s acceleration for quadratics

Another common approach to accelerating the convergence of gradient descent is Nesterov’s accelerated gradient descent (NAG). (NAG) uses iterates

$$\mathbf{x}_{k+1} = \mathbf{y}_k - \alpha_k \nabla f(\mathbf{y}_k), \quad \mathbf{y}_{k+1} = \mathbf{x}_{k+1} + \beta_k (\mathbf{x}_{k+1} - \mathbf{x}_k)$$

or equivalently,

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \beta(\mathbf{x}_k - \mathbf{x}_{k-1}) - \alpha(\mathbf{A}^\top \mathbf{A}(\mathbf{x}_k + \beta(\mathbf{x}_k - \mathbf{x}_{k-1})) - \mathbf{A}^\top \mathbf{b}). \tag{NAG}$$

Therefore, a computation analogous to the above computation for (HBM) shows that the (NAG) iterates satisfy the transition relation

$$\begin{bmatrix} \mathbf{x}_{k+1} - \mathbf{x}^* \\ \mathbf{x}_k - \mathbf{x}^* \end{bmatrix} = \underbrace{\begin{bmatrix} (1 + \beta)(\mathbf{I} - \alpha \mathbf{A}^\top \mathbf{A}) & -\beta(\mathbf{I} - \alpha \mathbf{A}^\top \mathbf{A}) \\ \mathbf{I} & \mathbf{0} \end{bmatrix}}_{\mathbf{T}=\mathbf{T}(\alpha,\beta)} \begin{bmatrix} \mathbf{x}_k - \mathbf{x}^* \\ \mathbf{x}_{k-1} - \mathbf{x}^* \end{bmatrix}. \tag{A.1}$$

Again, \mathbf{T} is unitarily similar to a block diagonal matrix whose blocks are

$$\mathbf{T}_j = \begin{bmatrix} (1 + \beta)(1 - \alpha \lambda_j) & -\beta(1 - \alpha \lambda_j) \\ 1 & 0 \end{bmatrix},$$

and it is easy to see the eigenvalues of \mathbf{T}_j are

$$z_j^\pm := \frac{1}{2} \left((1 + \beta)(1 - \alpha \lambda_j) \pm \sqrt{(1 + \beta)^2(1 - \alpha \lambda_j)^2 - 4\beta(1 - \alpha \lambda_j)} \right).$$

Rather than aiming to optimize the parameters α and β as we did for (HBM), we will simply use the standard choices of parameters suggested in Nesterov (2013):

$$\alpha = \frac{1}{L} \quad \text{and} \quad \beta = \frac{\sqrt{L/\ell} - 1}{\sqrt{L/\ell} + 1}.$$

By direct computation, we find

$$\frac{4\beta}{(1 + \beta)^2} = 1 - \frac{1}{L/\ell},$$

which implies

$$(1 + \beta)^2(1 - \alpha \lambda_j)^2 \leq 4\beta(1 - \alpha \lambda_j)$$

and therefore that

$$|z_j^\pm| = \sqrt{\beta(1 - \alpha \lambda_j)} \leq \sqrt{\beta(1 - \alpha \ell)} = 1 - \frac{1}{\sqrt{L/\ell}}.$$

Thus, the (NAG) iterates satisfy the convergence guarantee

$$\frac{\|\mathbf{x}_{k+1} - \mathbf{x}^*\|}{\|\mathbf{x}_0 - \mathbf{x}^*\|} \leq \sqrt{2}M(\alpha, \beta) \left(1 - \frac{1}{\sqrt{L/\ell}} \right)^k,$$

where $M(\alpha, \beta)$ is the eigenvector condition number of the transition matrix \mathbf{T} (note that this value is different from the value for (HBM) bounded in Lemma 2.1).

We now provide a bound, analogous to Theorem 3.2, for minibatch-NAG, the algorithm obtained by using the stochastic gradients (1.2) in (NAG).

THEOREM A.1 Set $\ell = \lambda_{\min}$, $L = \lambda_{\max}$ and define $\alpha = 1/L$ and $\beta = (\sqrt{L/\ell} - 1)/(\sqrt{L/\ell} + 1)$. For any $k^* > 0$ choose

$$B \geq 16\epsilon\eta \log(2d) \max \left\{ \frac{5\|\mathbf{A}\|_F^2 \|\mathbf{A}\|^2 \alpha^2 K^2 k^*}{\beta \log(k^*)}, \left(\frac{10\|\mathbf{A}\|_F^4 \alpha^2 K^2 k^*}{\beta \log(k^*)} \right)^{1/2} \right\}.$$

Then, for all $k > 0$, assuming that the minimizer \mathbf{x}^* satisfies $\mathbf{A}\mathbf{x}^* = \mathbf{b}$, the (Minibatch-HBM) iterates satisfy

$$\mathbb{E}[\|\mathbf{x}_k - \mathbf{x}^*\|] \leq \sqrt{2}M(\alpha, \beta) \max\{d, (k^*)^{k/k^*}\} \left(1 - \frac{1}{\sqrt{L/\ell} + 1}\right)^k \|\mathbf{x}_0 - \mathbf{x}^*\|.$$

The proof of Theorem A.1 is almost the same as the proof identical to the proof of Theorem 3.2, so we skip repeated parts.

Proof. For NAG we have that

$$\begin{aligned} \mathbf{Y}_{S_i} - \mathbb{E}[\mathbf{Y}_{S_i}] &= \sum_{j \in S_i} \frac{1}{B} \begin{bmatrix} (1 + \beta)\alpha(-p_j^{-1} \mathbf{a}_j \mathbf{a}_j^\top + \mathbf{A}^\top \mathbf{A}) & -\beta\alpha(-p_j^{-1} \mathbf{a}_j \mathbf{a}_j^\top + \mathbf{A}^\top \mathbf{A}) \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \\ &= \sum_{j \in S_i} \frac{\alpha}{B} \begin{bmatrix} -p_j^{-1} \mathbf{a}_j \mathbf{a}_j^\top + \mathbf{A}^\top \mathbf{A} & \\ & \mathbf{0} \end{bmatrix} \begin{bmatrix} (1 + \beta)\mathbf{I} & -\beta\mathbf{I} \\ \mathbf{I} & \mathbf{0} \end{bmatrix} \\ &= \alpha \left(\sum_{j \in S_i} \begin{bmatrix} \mathbf{W}_j & \\ & \mathbf{0} \end{bmatrix} \right) \begin{bmatrix} (1 + \beta)\mathbf{I} & -\beta\mathbf{I} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}. \end{aligned}$$

Note that, since $\sqrt{\beta} \leq 1$,

$$\left\| \begin{bmatrix} (1 + \beta)\mathbf{I} & -\beta\mathbf{I} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \right\| = \left\| \begin{bmatrix} (1 + \beta) & -\beta \\ 0 & 0 \end{bmatrix} \right\| = \sqrt{1 + 2\beta + 2\beta^2} \leq \sqrt{5}.$$

Thus, using the submultiplicativity of the operator norm, analogous to (3.6), we have that

$$\sqrt{\mathbb{E}[\|\mathbf{X}_{S_i} - \mathbb{E}[\mathbf{X}_{S_i}]\|^2]} \leq \sqrt{M(\alpha, \beta) \mathbb{E}[\|\mathbf{Y}_{S_i} - \mathbb{E}[\mathbf{Y}_{S_i}]\|^2]} \leq \sqrt{5}\alpha M(\alpha, \beta) \sqrt{\mathbb{E}[\|\mathbf{W}\|^2]}.$$

Again, using Lemma 3.1 and (3.6), we see that $\sqrt{\mathbb{E}\|\mathbf{X}_{S_i} - \mathbb{E}\mathbf{X}_{S_i}\|^2} \leq \delta$ provided that the batch size B satisfies

$$B \geq 8\epsilon\eta \log(2d) \max \{5\|\mathbf{A}\|_F^2 \|\mathbf{A}\|^2 \alpha^2 M(\alpha, \beta)^2 \delta^{-2}, (20\|\mathbf{A}\|_F^4 \alpha^2 M(\alpha, \beta)^2 \delta^{-2})^{1/2}\}.$$

Using the same choice of $\delta^2 = \beta \log(k^*)/(2k^*)$ we get the desired bound. □