IOP Publishing Inverse Problems

Inverse Problems **36** (2020) 054001 (23pp) https://doi.org/10.1088/1361-6420/ab77da

Sampled limited memory methods for massive linear inverse problems

Julianne Chung[1,4], Matthias Chung[1], J Tanner Slagel[2] and Luis Tenorio[3]

[1] Department of Mathematics, Computational Modeling and Data Analytics Division, Academy of Integrated Science, Virginia Tech, Blacksburg, VA, United States of America

[2] Department of Mathematics, Virginia Tech, Blacksburg, VA, United States of America

[3] Department of Applied Mathematics and Statistics, Colorado School of Mines, Golden, CO, United States of America

E-mail: jmchung@vt.edu, mcchung@vt.edu, slagelj@vt.edu and ltenorio@mines.edu

Received 17 December 2019, revised 5 February 2020
Accepted for publication 19 February 2020
Published 24 April 2020

Abstract

In many modern imaging applications the desire to reconstruct high resolution images, coupled with the abundance of data from acquisition using ultra-fast detectors, have led to new challenges in image reconstruction. A main challenge is that the resulting linear inverse problems are massive. The size of the forward model matrix exceeds the storage capabilities of computer memory, or the observational dataset is enormous and not available all at once. Row-action methods that iterate over samples of rows can be used to approximate the solution while avoiding memory and data availability constraints. However, their overall convergence can be slow. In this paper, we introduce a sampled *limited memory* row-action method for linear least squares problems, where an approximation of the global curvature of the underlying least squares problem is used to speed up the initial convergence and to improve the accuracy of iterates. We show that this limited memory method is a generalization of the damped block Kaczmarz method, and we prove linear convergence of the expectation of the iterates and of the error norm up to a convergence horizon. Numerical experiments demonstrate the benefits of these sampled limited memory row-action methods for massive 2D and 3D inverse problems in tomography applications.

Keywords: least squares problems, row-action methods, Kaczmarz methods, randomized methods, tomography, streaming data

(Some figures may appear in colour only in the online journal)

[4] Author to whom any correspondence should be addressed.

1. Introduction

Recent advancements in imaging technology have led to many new challenging mathematical problems for image reconstruction. One problem that has gained significant interest, especially in the age of big data, is that of image reconstruction when the number of unknown parameters is huge (i.e., images with very high spatial resolution) and the size of the observation dataset is massive and possibly growing [1, 2]. In streaming data problems, not only is it undesirable to wait until all data have been collected to obtain a reconstruction, but also partial reconstructions may be needed to inform the data acquisition process (e.g., for optimal experimental design or model calibration).

We consider linear inverse problems of the form

$$\mathbf{b} = \mathbf{A}\mathbf{x}_{\text{true}} + \boldsymbol{\varepsilon}, \tag{1}$$

where $\mathbf{x}_{\text{true}} \in \mathbb{R}^n$ is the desired solution, $\mathbf{A} \in \mathbb{R}^{m\times n}$ models the forward process, $\mathbf{b} \in \mathbb{R}^m$ contains observed measurements, and $\boldsymbol{\varepsilon} \in \mathbb{R}^m$ represents additive noise. We investigate sampled limited memory row-action methods to approximate a solution to the corresponding massive least squares (LS) problem, $\min_{\mathbf{x}} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2$. For $\mathbf{A}$ with full column rank, the goal is to approximate the unique solution,

$$\mathbf{x}_{\text{LS}} = \arg\min_{\mathbf{x}} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 = (\mathbf{A}^\top\mathbf{A})^{-1}\mathbf{A}^\top\mathbf{b}. \tag{2}$$

The LS problem is ubiquitous and core to computational mathematics and statistics. However, for massive problems where both the number of unknown parameters n and the size of the observational dataset m are massive or dynamically growing, standard techniques based on matrix factorization or iterative methods based on *full* matrix-vector multiplications (e.g., Krylov subspace methods [3] or randomized methods [4, 5]) are not feasible. Problems of such nature appear more frequently and are plentiful, for instance in classification problems [6], data mining [7–9], 3D molecular structure determination [10, 11], and super-resolution imaging [12, 13].

Row-action methods such as the Kaczmarz method have emerged as an attractive alternative due to their scalability, simplicity, and quick initial convergence [14–21]. Furthermore, for ill-posed inverse problems such as in tomography reconstruction, row-action methods are widely used and exhibit regularizing properties [22, 23]. They are commonly known as algebraic reconstruction techniques [24–28]. Basically, row-action methods are iterative methods where each iteration only requires a sample of rows of $\mathbf{A}$ and $\mathbf{b}$, thus circumventing issues with memory or data access. The most widely-known row-action methods are the Kaczmarz and block Kaczmarz methods [24], where only one row or one block of rows of $\mathbf{A}$ and $\mathbf{b}$ are required at each iteration. Various extensions have been proposed to include random sampling and damping parameters (e.g., [22, 29]), and many authors have studied the convergence properties of these methods [18–21, 30]. The literature on Kaczmarz-type methods is vast, and we refer the interested reader to overviews such as [31, 32] and references therein. In section 2 we provide some connections to previous works in this area, but first we present the problem setup.

To mathematically describe the sampling process, let $\{\mathbf{W}_k\}$ be an independent and identically distributed (i.i.d.) sequence of $m \times \ell$ random matrices uniformly distributed on the set $\{\mathbf{W}^{(1)}, \ldots, \mathbf{W}^{(M)}\}$, where $\mathbf{W}^{(i)}$ are such that $\mathbb{E}\mathbf{W}_k\mathbf{W}_k{}^\top = \beta\mathbf{I}$ for some $\beta > 0$. Then, at the kth iteration we denote $\mathbf{A}_k = \mathbf{W}_k^\top\mathbf{A}$ and $\mathbf{b}_k = \mathbf{W}_k^\top\mathbf{b}$. In this paper, we focus on a row-action method called *sampled limited memory for LS* (`slimLS`), where given an arbitrary initial guess

$\mathbf{x}_0 \in \mathbb{R}^n$, the kth `slimLS` iterate is defined as

$$\mathbf{x}_k = \mathbf{x}_{k-1} - \mathbf{B}_k \mathbf{A}_k^\top \left(\mathbf{A}_k \mathbf{x}_{k-1} - \mathbf{b}_k\right), \tag{3}$$

with

$$\mathbf{B}_k = \left(\alpha_k^{-1} \mathbf{C}_k + \mathbf{M}_k^\top \mathbf{M}_k\right)^{-1} \quad \text{and} \quad \mathbf{M}_k = [\mathbf{A}_{k-r}^\top, \ldots, \mathbf{A}_k^\top]^\top. \tag{4}$$

Here $\{\mathbf{C}_k\}$ is a sequence of matrices such that $\mathbf{B}_k$ is positive definite, $\{\alpha_k\}$ is sequence of damping parameters, and the parameter $r \in \mathbb{N}_0$ is a 'memory parameter' where we define $\mathbf{A}_{k-r}$ with negative index as an empty matrix. Hence, $\mathbf{M}_k$ increases in size within the first r iterations, and the global curvature of the problem is approximated using previous samples of the data.

Various choices for $\mathbf{W}$ can be used, e.g., see [33]. Notice that when the realizations of $\mathbf{W}$ are sparse matrices, only information from a few rows of $\mathbf{A}$ is extracted at each iteration, and $\mathbf{A}_k$ contains partial information of $\mathbf{A}$. A straightforward choice of $\mathbf{W}$, which we consider here, is where $\mathbf{W}_k$ is chosen such that $\mathbf{A}_k$ contains ℓ selected rows of $\mathbf{A}$ where $\beta = 1/M$. With this sampling scheme, the `slimLS` method is a generalization of the damped block Kaczmarz method, which is obtained when $\mathbf{C}_k = \mathbf{I}$ and $r = 0$. The `slimLS` method can also be interpreted as a stochastic approximation method [34]. Using the properties of $\mathbf{W}$ described above, one can show that the LS problem in (2) is equivalent to the following stochastic optimization problem,

$$\arg\min_{\mathbf{x}} \mathbb{E}\|\mathbf{W}^\top \left(\mathbf{A}\mathbf{x} - \mathbf{b}\right)\|_2^2. \tag{5}$$

Stochastic approximation methods for (5) may have the form $\mathbf{x}_k = \mathbf{x}_{k-1} - \mathbf{B}_k \nabla f_{\mathbf{W}_k}(\mathbf{x}_{k-1})$, where $f_{\mathbf{W}_k}(\mathbf{x}) = \|\mathbf{W}_k{}^\top \left(\mathbf{A}\mathbf{x} - \mathbf{b}\right)\|_2^2$ and $\{\mathbf{B}_k\}$ is some sequence of positive semi-definite matrices. For the particular choice of $\mathbf{B}_k$ defined in (4), we see that the `slimLS` method is a stochastic approximation method. Furthermore, since samples of rows of $\mathbf{A}$ are used at each iteration, randomized row-action methods can be characterized as stochastic approximation methods applied to (5).

Although our proposed `slimLS` method can be interpreted as both a row-action and a stochastic approximation method, the main distinction of `slimLS` compared to existing methods to approximate (2) is that `slimLS` exhibits favorable initial and asymptotic convergence properties for constant and vanishing step sizes, respectively. Kaczmarz methods have fast initial convergence, but for inconsistent systems iterates converge asymptotically to a weighted LS solution rather than the desired LS solution [17]. On the other hand, stochastic gradient methods (where $\mathbf{B}_k = \alpha_k \mathbf{I}$) are guaranteed to converge asymptotically to the LS solution but can have erratic initial convergence. We show linear convergence of the expectation of `slimLS` iterates, and we prove linear convergence of the expected squared error norm up to a 'convergence horizon' for constant damping parameter. Furthermore, it can be shown that `slimLS` iterates with decaying damping parameter converge asymptotically to the LS solution [33, 35]. The power of the `slimLS` method is revealed in our numerical examples, where we demonstrate the performance of the `slimLS` method for massive and streaming tomographic reconstruction problems.

An outline of the paper is as follows. In section 2, we give an overview of previous work on row-action methods and make connections to and distinctions from existing methods. In section 3, we provide convergence results for `slimLS` iterates. Numerical results are presented in section 4, where we compare the performance of `slimLS` to existing methods, and conclusions are provided in section 5.

2. Previous works on row-action methods

Different choices of $\mathbf{B}_k$ in (3) yield different well-known row-action methods. The most computationally simple choice is $\mathbf{B}_k = \alpha_k \mathbf{I}$, which gives the standard stochastic gradient method,

$$\mathbf{x}_k = \mathbf{x}_{k-1} - \alpha_k \mathbf{A}_k^\top (\mathbf{A}_k \mathbf{x}_{k-1} - \mathbf{b}_k).$$

Under mild conditions, the stochastic gradient method converges to the LS solution [33, 35], but convergence can be very slow and depends heavily on the choice of the step size α_k [36, 37]. We remark that α_k has different interpretations depending on the scientific community. It is often referred to as the learning rate in machine learning, the step size in classical optimization, and the relaxation parameter in algebraic reconstruction techniques for tomography. Notice that for `slimLS` iterates, the damping parameter plays the role of the step size.

Stochastic Newton or stochastic quasi-Newton methods have also been proposed to accelerate convergence [6, 33, 38, 39]. For the stochastic Newton method, we can let $\mathbf{B}_k = \alpha_k \left(\mathbf{A}_k^\top \mathbf{A}_k\right)^\dagger$ in (3), and we get the block Kaczmarz method,

$$\begin{aligned} \mathbf{x}_k &= \mathbf{x}_{k-1} - \alpha_k \left(\mathbf{A}_k^\top \mathbf{A}_k\right)^\dagger \mathbf{A}_k^\top (\mathbf{A}_k \mathbf{x}_{k-1} - \mathbf{b}_k) \\ &= \mathbf{x}_{k-1} - \alpha_k \mathbf{A}_k^\dagger (\mathbf{A}_k \mathbf{x}_{k-1} - \mathbf{b}_k), \end{aligned} \tag{6}$$

where we have used the property that $\left(\mathbf{A}_k^\top \mathbf{A}_k\right)^\dagger \mathbf{A}_k{}^\top = \mathbf{A}_k^\dagger$. Thus, we see that the block Kaczmarz method is nothing more than a stochastic Newton method. Note that for $\alpha_k = 1$, we get the standard block Kaczmarz method, and linear convergence to within a convergence horizon has been shown in [19, 40]. For a decaying α_k, the block Kaczmarz method converges to the solution of a weighted LS problem, rather than to the desired LS solution [26].

For the special case where $\mathbf{W}$ is a uniform random vector on the columns of the identity matrix, each iteration only requires one row of $\mathbf{A}$. More precisely, let $\mathbf{a}_i \in \mathbb{R}^{1\times n}$ denote the ith row of $\mathbf{A}$ and let τ be a random variable with uniform distribution on the set $\{1, \ldots, m\}$, then $\mathbf{W}_k{}^\top \mathbf{A} = \mathbf{a}_{\tau(k)}$. In this case, stochastic Newton iterates in (6) are identical to the randomized Kaczmarz method,

$$\mathbf{x}_k = \mathbf{x}_{k-1} - \alpha_k \frac{\mathbf{a}_{\tau(k)} \mathbf{x}_{k-1} - b_{\tau(k)}}{\|\mathbf{a}_{\tau(k)}^\top\|_2^2} \mathbf{a}_{\tau(k)}^\top, \tag{7}$$

which has been studied extensively, see e.g., [17, 21, 25–27, 41–45].

The Kaczmarz method was introduced for cyclic control (i.e., $\tau(k) = ((k-1) \bmod m) + 1$) and $\alpha_k = 1$, where for an invertible matrix $\mathbf{A}$, it was first shown in 1937 that the iterates converge to the unique solution [46]. Extensions of the Kaczmarz method to rectangular systems have been considered. Tanabe showed convergence for consistent systems in 1971 [44], and Herman provided results for Kaczmarz methods with relaxation parameters $\alpha_k \neq 1$ [47]. Various theoretical convergence properties of the Kaczmarz algorithm have been investigated for inconsistent systems. For a decaying step size (strong underrelaxation), iterates will converge to a weighted LS solution $\tilde{\mathbf{x}} = \arg\min_{\mathbf{x}} \|\mathbf{D}^{-1} (\mathbf{A}\mathbf{x} - \mathbf{b})\|_2^2$ where $\mathbf{D} \in \mathbb{R}^{m\times m}$ is a diagonal matrix with diagonal elements $d_i = \|\mathbf{a}_i\|_2^2$, see [26]. Extended Kaczmarz methods for inconsistent systems have also been proposed and investigated [48, 49].

A randomized Kaczmarz method developed for consistent overdetermined systems was shown to have an expected linear convergence rate that depends on the condition number of $\mathbf{A}$ [20, 21]. For a constant step size, these iterates will converge linearly to the weighted LS solution to within what is known as a convergence horizon, which accounts for the variance in

the iterates [17, 18]. It is worth noting that a special case of `slimLS` where $\mathbf{W}_k^\top \mathbf{A} = \mathbf{a}_{\tau(k)}$ and $\mathbf{C}_k = \mathbf{a}_{\tau(k)}\mathbf{C}^{-1}\mathbf{a}_{\tau(k)}^\top\mathbf{C} - \alpha_k \mathbf{M}_k^\top \mathbf{M}_k$ for any symmetric positive definite matrix $\mathbf{C}$ can be interpreted as a randomized Kaczmarz method with a mismatched adjoint, for which expected convergence up to a convergence horizon was shown in [50].

To address problems that arise when some rows have small norm (i.e., a small denominator in (7)), Andersen and Hansen [22] in 2014 considered a variant of the Kaczmarz method to include a damping term. They showed a connection to proximal gradient methods and provided convergence properties under cyclic control. When the blocks $\mathbf{A}_k$ are ill-conditioned, computing the search direction in (6) can become numerically unstable and a similar idea can be used. A damping term can be introduced in the sample Hessian, which leads to the damped block Kaczmarz method,

$$\mathbf{x}_k = \mathbf{x}_{k-1} - \left(\alpha_k^{-1}\mathbf{I} + \mathbf{A}_k^\top \mathbf{A}_k\right)^{-1} \mathbf{A}_k^\top \left(\mathbf{A}_k \mathbf{x}_{k-1} - \mathbf{b}_k\right). \tag{8}$$

Notice that including the damping parameter eliminates the need for a step size parameter, although one could still be included.

To speed up convergence, stochastic quasi-Newton methods use the current and any previous samples of $\mathbf{A}$ to produce a matrix $\mathbf{B}_k$ that approximates the global curvature $\left(\mathbf{A}^\top \mathbf{A}\right)^{-1}$. For general convex optimization, stochastic quasi-Newton methods that use an LBFGS type update have been introduced and analyzed [39, 51, 52]. These methods have been investigated for nonlinear problems; however, for linear problems better approximations can be obtained by exploiting the fact that the Hessian is constant. One row-action method for linear problems is the randomized recursive LS method where the kth iterate, which is given by

$$\mathbf{x}_k = \mathbf{x}_{k-1} - \left(\sum_{i=1}^{k} \mathbf{A}_i^\top \mathbf{A}_i\right)^\dagger \mathbf{A}_k^\top \left(\mathbf{A}_k \mathbf{x}_{k-1} - \mathbf{b}_k\right), \tag{9}$$

is the minimum norm solution of

$$\min_{\mathbf{x}} \left\| \begin{bmatrix} \mathbf{A}_1 \\ \vdots \\ \mathbf{A}_k \end{bmatrix} \mathbf{x} - \begin{bmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_k \end{bmatrix} \right\|_2^2. \tag{10}$$

This equivalency is shown in appendix A and implies that after sampling all M blocks exactly once, we get $\mathbf{x}_M = \mathbf{x}_{\text{LS}}$. The disadvantage is that the randomized recursive LS algorithm is not computationally feasible for very large problems because of the large linear solve required at each iteration and the cost to store $\sum_{i=1}^{k} \mathbf{A}_i^\top \mathbf{A}_i$, see [13, 53, 54]. Notice that if $\mathbf{C}_k = \sum_{i=1}^{k-r-1} \mathbf{A}_i^\top \mathbf{A}_i$ and $\alpha_k = 1$ in (3), we recover the randomized recursive LS algorithm. Thus, the `slimLS` iterates (3) can be interpreted as a limited memory variant of the recursive LS algorithm. On the other hand, if $\mathbf{C}_k = \alpha_k^{-1}\mathbf{I}_n$ the `slimLS` method can be interpreted as a generalization of the damped block Kaczmarz method.

It should be noted that other methods exist for solving very large LS problems, but many have limitations that prohibit their use for massive or streaming data problems. For example, for problems where m is small enough to allow storage of an $m \times m$ matrix, the Sherman–Woodbury identity can be used to get the exact solution [55]. In our problems, m and n are on the order of hundreds of millions. Randomized methods such as Blendenpik [4] and LSRN [56] are effective for cases where $m \gg n$ or $n \gg m$, (assuming matrix $\mathbf{A}$ has a large gap

in the singular values). Nevertheless, these methods require full matrix-vector-multiplications and do not work for streaming problems.

Next, for a specific choice of $\mathbf{W}_k$, we make a connection to the sampled limited memory Tikhonov (`slimTik`) method described in [13] to approximate a Tikhonov regularized solution,

$$\mathbf{x}_{\text{tik}} = \arg\min_{\mathbf{x}} \ \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 + \lambda^2 \|\mathbf{L}\mathbf{x}\|_2^2 = \left\| \begin{bmatrix} \mathbf{A} \\ \lambda \mathbf{L} \end{bmatrix} \mathbf{x} - \begin{bmatrix} \mathbf{b} \\ \mathbf{0} \end{bmatrix} \right\|_2^2, \tag{11}$$

where $\lambda \in \mathbb{R}^+$ is the regularization parameter and the regularization matrix $\mathbf{L} \in \mathbb{R}^{n \times n}$ is invertible[5]. In particular, let $\mathbf{W}_k$ be defined as in section 1 and define random variable

$$\widetilde{\mathbf{W}}_k = \begin{bmatrix} \mathbf{W}_k & \mathbf{0}_{m \times n} \\ \mathbf{0}_{n \times \ell} & \frac{1}{\sqrt{M}} \mathbf{I} \end{bmatrix},$$

where $\widetilde{\mathbf{W}}_k$ has the property that $\mathbb{E}\widetilde{\mathbf{W}}_k \widetilde{\mathbf{W}}_k^\top = \frac{1}{M}\mathbf{I}$. Then, `slimLS` applied to (11) with $\mathbf{C}_k = \mathbf{L}^\top \mathbf{L}$ gives iterates,

$$\mathbf{x}_k = \mathbf{x}_{k-1} - \left(\left(\alpha_k^{-1} + \frac{r\lambda^2}{M} \right) \mathbf{L}^\top \mathbf{L} + \mathbf{M}_k^\top \mathbf{M}_k \right)^{-1} \left(\mathbf{A}_k^\top \left(\mathbf{A}_k \mathbf{x}_{k-1} - \mathbf{b}_k \right) + \frac{\lambda^2}{M} \mathbf{L}^\top \mathbf{L} \mathbf{x}_{k-1} \right), \tag{12}$$

which are equivalent to `slimTik` iterates with a fixed regularization parameter. The significance of this equivalence is that the analysis and results that we present in the next section can be extended to the Tikhonov problem. It should be noted that a good choice of λ may not be known in advance and must be estimated. Methods to update the Tikhonov regularization parameter within the `slimTik` method have been studied in [13], but a theoretical analysis for such cases remains a topic of ongoing research.

3. Convergence properties of `slimLS`

In this section we analyze the convergence properties of the `slimLS` method. In particular, we will show that it exhibits favorable initial convergence properties without the memory burden of having to save all previous samples (e.g., as in randomized recursive LS [13, 58]). This is possible because `slimLS` iterates can utilize previous samples to better approximate the Hessian $\mathbf{A}^\top \mathbf{A}$. We show that for a fixed damping parameter $\alpha > 0$, memory level $r = 0$, and $\mathbf{C}_k = \mathbf{I}$, `slimLS` iterates exhibit linear convergence of the expectation of the iterates and linear convergence of the L^2-error up to a convergence horizon. This type of analysis is essential for understanding stochastic approximation methods [16, 18, 19, 21, 54], and it may reveal potential trade-offs between solution accuracy and speed of convergence based on the damping parameter. For example, such analyses have been proved for the Kaczmarz method (for vanishing step size) and for the block Kaczmarz method (for step size one) [16, 18, 19, 21], but to the best of our knowledge results have not been proved for the damped block Kaczmarz method. For clarity of presentation, all proofs have been relegated to appendix B.

The following definitions will be used throughout the paper. We will use the functions $\lambda_{\min}(\cdot)$ and $\lambda_{\max}(\cdot)$ that provide the smallest and largest eigenvalues of a matrix, and write

[5] This assumption is not required [57] but is used here for notational simplicity.

$$A_{\min} = \min_k \left\{ \lambda_{\min}(\mathbf{A}^\top \mathbf{W}^{(i)} \big(\mathbf{W}^{(i)}\big)^\top \mathbf{A}) > 0 \right\} \quad \text{and} \quad A_{\max} = \max_k \lambda_{\max}(\mathbf{A}^\top \mathbf{W}^{(i)} \big(\mathbf{W}^{(i)}\big)^\top \mathbf{A}),$$

where the minimum is across all of the M different realizations of $\mathbf{W}_k$ that lead to a positive minimum eigenvalue, while the maximum is across all of the M realizations. For a fixed $\alpha > 0$ we define the matrices

$$\mathbf{B}_k(\alpha) = \alpha\big(\mathbf{I} + \alpha\, \mathbf{A}_k^\top \mathbf{A}_k\big)^{-1} \quad \text{and} \quad \mathbf{B} = \mathbb{E}\, \mathbf{B}_k(\alpha)\mathbf{A}_k^\top \mathbf{A}_k = \mathbf{I} - \mathbb{E}\, \mathbf{B}_k(\alpha)/\alpha.$$

For simplicity we will often write $\mathbf{B}_k$ instead of $\mathbf{B}_k(\alpha)$. It is clear that $\mathbf{B}$ is symmetric positive semi-definite. In fact, it is positive definite with $\|\mathbf{B}\|_2 < 1$ when $\mathbf{A}$ has full column-rank (see lemma B.1), in which case we define

$$\widehat{\mathbf{x}} = \arg\min_{\mathbf{x}} \|\mathbf{B}\mathbf{x} - \mathbb{E}\, \mathbf{B}_k \mathbf{A}_k^\top \mathbf{b}_k\|_2^2 = \mathbf{B}^{-1} \mathbb{E}\, \mathbf{B}_k \mathbf{A}_k^\top \mathbf{b}_k. \tag{13}$$

Note that all of the expectations in this paper are understood to be with respect to the joint distribution of the sequence $\{\mathbf{W}_k\}$ conditional on the noise.

Theorem 3.1. *Let $\mathbf{A} \in \mathbb{R}^{m\times n}$ have full column-rank. For arbitrary initial vector $\mathbf{x}_0 \in \mathbb{R}^n$ and damping parameter $\alpha > 0$, define*

$$\mathbf{x}_k = \mathbf{x}_{k-1} - \mathbf{B}_k(\alpha)\, \mathbf{A}_k^\top\, (\mathbf{A}_k \mathbf{x}_{k-1} - \mathbf{b}_k)$$

Then:

(*a*) $\mathbb{E}\, \mathbf{x}_k \to \widehat{\mathbf{x}}$, *or more precisely,*

$$\|\mathbb{E}\, \mathbf{x}_k - \widehat{\mathbf{x}}\|_2 \leqslant \rho^k \|\mathbf{x}_0 - \widehat{\mathbf{x}}\|_2,$$

where $\rho = \|\mathbb{E}\, \mathbf{B}_k(\alpha)/\alpha\|_2 < 1$.

(*b*) *The L^2-error around $\widehat{\mathbf{x}}$ can be bounded by*

$$\mathbb{E}\|\mathbf{x}_k - \widehat{\mathbf{x}}\|_2^2 \leqslant (1-2c)^k\, \|\mathbf{x}_0 - \widehat{\mathbf{x}}\|_2^2 + \alpha^2 c^{-1} \sigma^2, \tag{14}$$

where $0 < 1 - 2c < 1$, with $c = \lambda_{\min}(\mathbf{B})\,/(1 + \alpha A_{\max})$ and $\sigma = \mathbb{E}\|\mathbf{A}_k^\top(\mathbf{A}_k\widehat{\mathbf{x}} - \mathbf{b}_k)\|_2$.

The first part of the theorem shows that as $k \to \infty$, $\mathbf{x}_k$ is an asymptotically unbiased estimator of $\widehat{\mathbf{x}}$ with a linear convergence rate. The second part shows linear convergence of the L^2-error up to a convergence horizon. For the case where $\alpha \to 0$, the constant in the first term of (14) approaches one, indicating a slowing linear convergence rate, while the second term in (14) goes to zero, i.e., the convergence horizon gets smaller. This is because $\alpha^2/\lambda_{\min}(\mathbf{B}) \to 0$ as $\alpha \to 0$, since $\lambda_{\min}(\mathbf{B}) > 0$ as shown in lemma B.1. In general the value for σ will depend on various factors including the amount of noise in $\mathbf{b}$, the definition of the blocks $\mathbf{A}_k$ and $\mathbf{b}_k$, and the value of α.

Having shown that $\mathbf{x}_k$ converges to $\widehat{\mathbf{x}}$ in L^2 as $k \to \infty$ and $\alpha \to 0$, the next question is how much $\widehat{\mathbf{x}}$ differs from the LS solution $\mathbf{x}_{\text{LS}}$. To answer this question we let $\mathbf{P}_{\mathbf{A}} = \mathbf{A}(\mathbf{A}^\top\mathbf{A})^{-1}\mathbf{A}^\top$ and $\mathbf{Q}_{\mathbf{A}} = \mathbf{I} - \mathbf{P}_{\mathbf{A}}$ be, respectively, the orthogonal projections onto the column space of $\mathbf{A}$ and its orthogonal complement. We then have the following equivalent definition of $\widehat{\mathbf{x}}$:

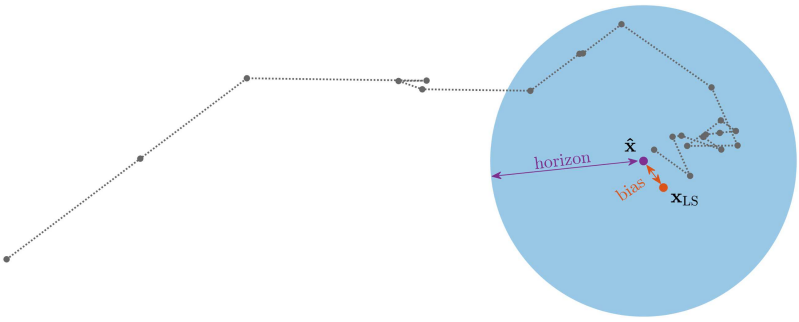


Figure 1. Illustration of the convergence horizon for `slimLS`. The gray dotted line contains the iterates $\mathbf{x}_k$ and the convergence horizon is denoted by the blue disk. By theorem 3.1, the `slimLS` iterates converge to within a convergence horizon of $\widehat{\mathbf{x}}$. $\mathbf{x}_{\mathrm{LS}}$ is provided for comparison with $\widehat{\mathbf{x}}$ (see lemma 3.2).

$$\begin{aligned}\widehat{\mathbf{x}} &= \arg\min_{\mathbf{x}} \|\mathbb{E}\,\mathbf{B}_k\mathbf{A}_k^\top\mathbf{A}_k\,(\mathbf{x}-\mathbf{x}_{\mathrm{LS}}) - \mathbb{E}\,\mathbf{B}_k\mathbf{A}_k^\top\mathbf{W}_k^\top\mathbf{Q}_{\mathbf{A}}\mathbf{b})\|_2^2 \\ &= \mathbf{x}_{\mathrm{LS}} + \mathbf{B}^{-1}\,(\,\mathbb{E}\,\mathbf{B}_k\mathbf{A}_k^\top\mathbf{W}_k^\top\,)\,\mathbf{Q}_{\mathbf{A}}\mathbf{b}. \qquad (15)\end{aligned}$$

In particular, $\widehat{\mathbf{x}} = \mathbf{x}_{\mathrm{LS}}$ when $\mathbf{b}$ belongs to the column space of $\mathbf{A}$, and in this case, it is easy to see that the constant $\sigma = 0$ in (14). The following result provides a bound for $\|\widehat{\mathbf{x}} - \mathbf{x}_{\mathrm{LS}}\|_2$.

Lemma 3.2. *If $\mathbf{A} \in \mathbb{R}^{m\times n}$ has full column-rank, then*

$$\|\widehat{\mathbf{x}} - \mathbf{x}_{LS}\|_2 \leqslant \alpha\,\frac{M(1+\alpha A_{\min})A_{\max}}{(1+\alpha A_{\max})A_{\min}}\,C\,\|\mathbf{Q}_{\mathbf{A}}\mathbf{b}\|_2,$$

where $C = \mathbb{E}\,\|\mathbf{A}_k^\top\mathbf{W}_k\|_2 + \|(\mathbf{A}^\top\mathbf{A})^{-1}\|_2\,\|\mathbf{A}\|_2\,\mathbb{E}\,\|\mathbf{A}_k^\top\mathbf{A}_k\|_2$.

It is important to notice the relationship between the damping parameter α and the upper bound in lemma 3.2. The bound is smaller for smaller values of α, which makes sense in light of the asymptotic property that $\mathbf{x}_k \to \mathbf{x}_{\mathrm{LS}}$ a.s. for a decaying damping parameter α (see [33]). However, there is a trade-off between the convergence rate and the precision of iterates (i.e., the bias) that depends on α: as $\alpha \to 0$, we get more accurate approximations, i.e., $\widehat{\mathbf{x}} \to \mathbf{x}_{\mathrm{LS}}$ in L^2. On the other hand, for larger α the convergence is faster at the cost of a larger convergence horizon.

In summary, we have shown that with a fixed damping parameter, the `slimLS` iterates will converge in L^2 linearly to within a horizon of $\widehat{\mathbf{x}}$, and the expected value of the iterates converges to $\widehat{\mathbf{x}}$. A pictorial illustration of this convergence behavior is provided in figure 1.

This trade-off between quick initial convergence that comes with using a constant damping parameter at the cost of solution accuracy has been observed in related stochastic optimization methods in the literature, see e.g., [22, 59]. For row-action methods theoretical results supporting the observed quick initial convergence behavior can be found in [30]. It has also been observed that more accurate solutions can be obtained using a decaying damping parameter, but then convergence can be quite slow (sub-linear) [60, 61]. Thus, it is often practical to use a constant damping parameter to obtain quick initial convergence and then switch to a decaying parameter to obtain higher accuracy. It may be desirable to provide convergence results for more general sequences $\{\mathbf{C}_k\}$. However, our proofs rely on various properties of the sequence

$\mathbf{B}_k$ which are fulfilled for $\mathbf{C}_k = \mathbf{I}$ (see lemma B.1). Determining a wider class of sequences is a topic of future work.

4. Numerical results

In this section we first illustrate the convergence behavior of our proposed `slimLS` method in a small simulation study. The goal of the first experiment is to illustrate how different memory levels and damping parameters affect the convergence of `slimLS`. We also compare `slimLS` to existing row-action methods and provide a numerical investigation into the sensitivity toward the damping parameter/step size. Then we discuss some of the computational considerations when solving massive problems and investigate the performance of our method on very large tomographic reconstruction problems.

4.1. An illustrative example

In the first numerical experiment we use a smaller example to illustrate some of the key features of the `slimLS` method. We let $\mathbf{A} \in \mathbb{R}^{1000\times 100}$ have random entries from a standard normal distribution. We further assume that $\mathbf{x}_{\text{true}} = [1, \ldots, 1]^\top$ and the simulated observations $\mathbf{b}$ are generated as in (1) where $\boldsymbol{\epsilon}$ is white noise with noise level 1%; that is, $\|\boldsymbol{\epsilon}\|_2/\|\mathbf{A}\mathbf{x}_{\text{true}}\|_2 = 0.01$. The matrix $\mathbf{A}$ is assumed to be sampled in $M = 100$ blocks, each with block size $\ell = 10$, which corresponds to sampling matrices $\mathbf{W}^{(i)} = \left[\mathbf{0}_{100\times 10(i-1)}, \mathbf{I}, \mathbf{0}_{100\times 10(100-i)}\right]^\top$.

First we illustrate the convergence behavior investigated in theorem 3.1 for different constant damping parameters α. In figure 2 we provide relative reconstruction errors computed as $\|\mathbf{x}_k - \widehat{\mathbf{x}}\|_2/\|\widehat{\mathbf{x}}\|_2$ for various damping factors from $\alpha = 0.001$ to $\alpha = 10$ on a log–log scale. We repeatedly run `slimLS` with random sampling for 100 epochs and with memory level $r = 0$. We observe that larger values of α have favorable initial convergence, but then stabilize at a larger relative error. On the other hand, smaller values of α have a slower initial convergence, but to a smaller relative error. This illustrates the trade-off between fast initial convergence and solution accuracy, as discussed in section 3. Furthermore, for various α we provide the relative difference between $\widehat{\mathbf{x}}$ and $\mathbf{x}_{\text{LS}}$ in figure 3. We notice that for small α the relative difference is within machine precision while slowly increasing for $\alpha > 10^{-1}$.

Next, we investigate how the initial convergence is affected by the choice of the memory parameter r. Here, we fix $\alpha = 1$ and choose memory levels $r = 0, 2, 4, 6$, and 8. We run our `slimLS` method for 20 iterations and provide the median relative reconstruction errors for 100 repeated runs in figure 4. The errors are computed with respect to the LS solution, i.e., $\|\mathbf{x}_k - \mathbf{x}_{\text{LS}}\|_2/\|\mathbf{x}_{\text{LS}}\|_2$. We empirically observe that with higher memory levels we get faster initial convergence.

We also provide a comparison of `slimLS` with $r = 0$ to other row-action methods, including the sampled or batch gradient `sg` method and the online limited memory BFGS `olbfgs` method [52] with memory level 10 (i.e., storing the 10 previously computed sampled gradient vectors). In particular, we are interested in the sensitivity of the algorithms with respect to the step size. For different constant step sizes/damping factors from $\alpha = 10^{-5}$ to $\alpha = 10^3$, we computed the reconstruction error norm at $k = 100$ (i.e., corresponding to one epoch) relative to $\widehat{\mathbf{x}}$. Although reconstruction errors could be computed relative to $\mathbf{x}_{\text{LS}}$, we note that the relative error between $\mathbf{x}_{\text{LS}}$ and $\widehat{\mathbf{x}}$ is negligible (see figure 3). In figure 5 we provide the median relative reconstruction error norms along with the 5–95th percentiles after repeating the experiment 100 times. Notice that the plot is on a log–log scale. We observe that the `sg` method has just a tiny 'window' of α's for which results have small relative reconstruction errors. The window for `olbfgs` is larger and is centered around step size $\alpha = 1$, which is expected, see [62].

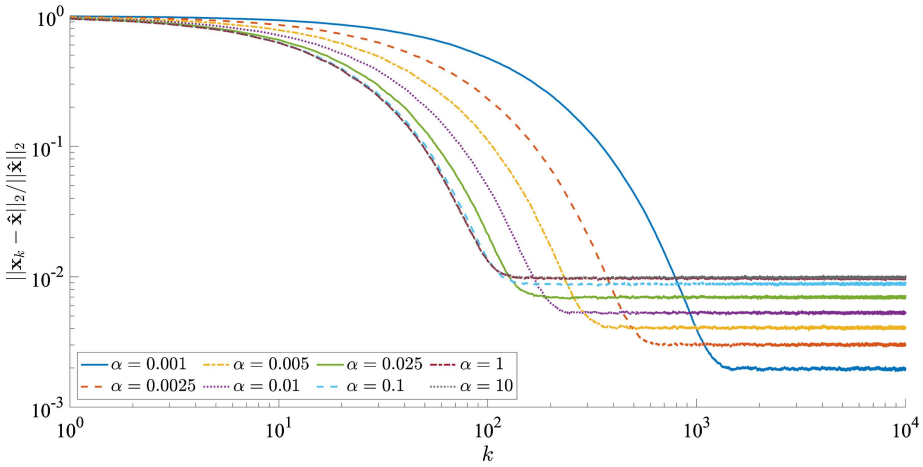


Figure 2. Comparison of median reconstruction errors, over 100 runs and relative to $\widehat{\mathbf{x}}$, for different (fixed) damping parameters α in `slimLS` on a log–log scale.

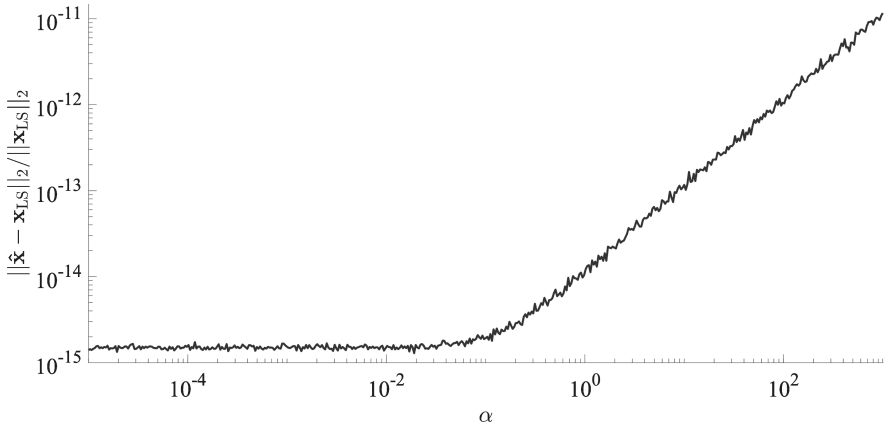


Figure 3. Relative difference between the weighted LS solution $\widehat{\mathbf{x}}$ and the desired LS solution $\mathbf{x}_{\mathrm{LS}}$ for various α.

However, compared to `sg` and `olbfgs`, the `slimLS` method provides good reconstructions for a much wider range of damping factors, which is a very attractive property of the `slimLS` method.

4.2. Computational considerations

Recall that the `slimLS` method can be interpreted as a row-action method, which by construction alleviates many of the computational bottlenecks (i.e., data access and memory requirements per iteration are significantly reduced). However, for many realistic problems where n is on the order of millions or billions (e.g., in tomography), the computational cost of each iteration can still be large. We remark that a noteworthy distinction of our numerical investigations

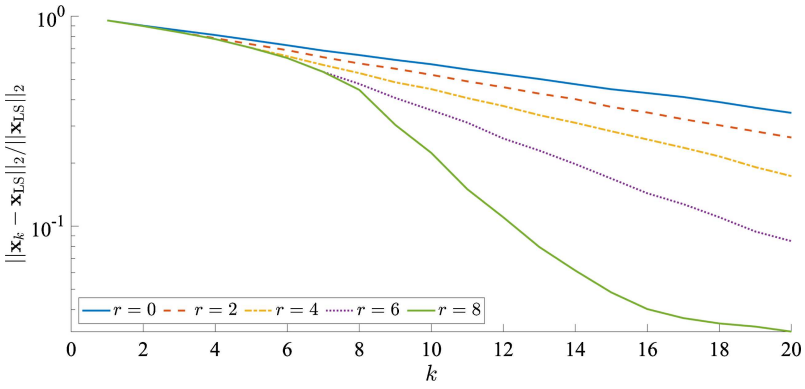


Figure 4. Comparison of median reconstruction errors, over 100 runs and relative to $\mathbf{x}_{\mathrm{LS}}$, for different memory levels in `slimLS` with $\alpha = 1$.

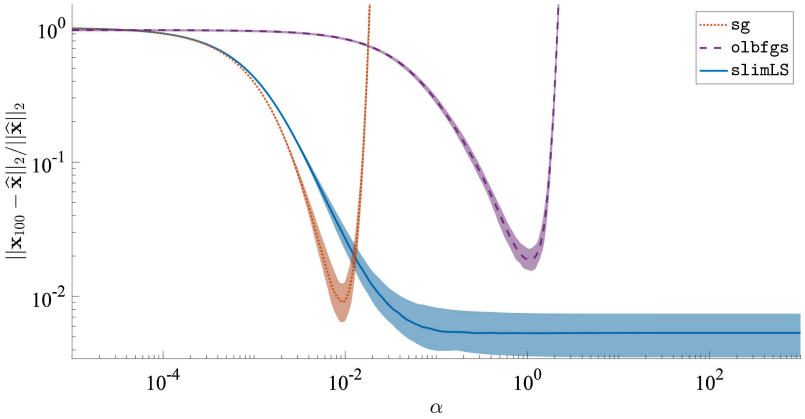


Figure 5. Comparison of median reconstruction errors along with the 5–95th percentiles over 100 runs and relative to $\widehat{\mathbf{x}}$ for different (fixed) step sizes in `slimLS`, sampled gradient, and online limited memory BFGS. All results correspond to accessing one epoch of the data.

compared to previously published works on row-action methods, such as [19, 21, 22], is that we consider very large imaging problems with hundreds of millions of unknowns and focus on the initial convergence (one epoch) rather than the 'asymptotic' convergence (hundreds of epochs) behavior. Next we address some considerations with respect to computational cost and comparisons with other methods.

The `slimLS` iterates can be written as $\mathbf{x}_k = \mathbf{x}_{k-1} - \mathbf{s}_k$, where

$$\mathbf{s}_k = \left(\alpha_k^{-1}\mathbf{C}_k + \mathbf{M}_k^\top \mathbf{M}_k\right)^{-1} \mathbf{A}_k^\top \left(\mathbf{A}_k \mathbf{x}_{k-1} - \mathbf{b}_k\right),$$

where $\mathbf{M}_k = [\mathbf{A}_{k-r}^\top, \ldots, \mathbf{A}_k^\top]^\top$. Thus, each iteration consists of two main steps,

(*a*) Accessing model block $\mathbf{A}_k$ and corresponding data $\mathbf{b}_k$, and

(*b*) Computing the update step $\mathbf{s}_k$.

The computational costs for the first step are often overlooked, but since data are usually stored on a hard-drive, data access can be time-consuming. Furthermore, depending on the application, constructing the corresponding matrix block $\mathbf{A}_k$ can also be computationally expensive. In data-streaming problems one may not have control over when and which blocks of data become available at any given time.

For the second step and for symmetric $\mathbf{C}_k$, solving for $\mathbf{s}_k$ can be done efficiently by first noticing that $\mathbf{s}_k$ is the solution to the LS problem,

$$\mathbf{s}_k = \arg\min_{\mathbf{s}} \left\| \begin{bmatrix} \mathbf{A}_{k-r} \\ \vdots \\ \mathbf{A}_{k-1} \\ \mathbf{A}_k \\ \frac{1}{\sqrt{\alpha_k}}\mathbf{L}_k \end{bmatrix} \mathbf{s} - \begin{bmatrix} \mathbf{0} \\ \vdots \\ \mathbf{0} \\ \mathbf{A}_k\mathbf{x}_{k-1} - \mathbf{b}_k \\ \mathbf{0} \end{bmatrix} \right\|_2^2, \tag{16}$$

where $\mathbf{C}_k = \mathbf{L}_k^\top \mathbf{L}_k$. Hence, any efficient LS solver that exploits the structure in (16) can be used to compute $\mathbf{s}_k$. Here, we utilize LSQR for damped LS, see [63], where a very efficient implementation is available if $\mathbf{L}_k = \mathbf{I}$. It is worth mentioning that another LS reformulation can be made to solve for $\mathbf{x}_k$ directly, where the right-hand side becomes dense and will depend on α_k and $\mathbf{L}_k$. That is,

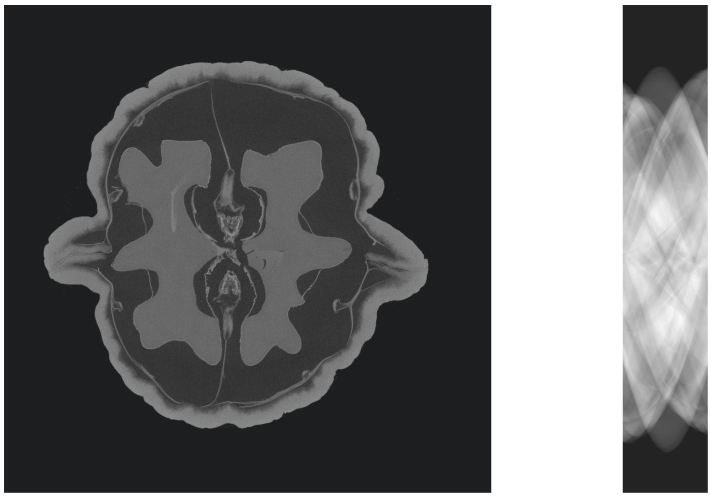
$$\mathbf{x}_k = \arg\min_{\mathbf{x}} \left\| \begin{bmatrix} \mathbf{A}_{k-r} \\ \vdots \\ \mathbf{A}_{k-1} \\ \mathbf{A}_k \\ \frac{1}{\sqrt{\alpha_k}}\mathbf{L}_k \end{bmatrix} \mathbf{x} - \begin{bmatrix} \mathbf{A}_{k-r}\mathbf{x}_{k-1} \\ \vdots \\ \mathbf{A}_{k-1}\mathbf{x}_{k-1} \\ \mathbf{b}_k \\ \frac{1}{\sqrt{\alpha_k}}\mathbf{L}_k\mathbf{x}_{k-1} \end{bmatrix} \right\|_2^2. \tag{17}$$

Next, we remark further on the choice of α_k. For the `sg` method—as illustrated in figure 5—an acceptable constant step size α_k is hard to come by and is often chosen such that $\alpha_k \ll 1$. For `olbfgs` we choose the 'natural' constant step size $\alpha_k = 1$, with a possible exception at the early iterations. Since `olbfgs` is equivalent to `sg` in the first iteration, `olbfgs` may suffer from large step sizes while building up its memory. To compensate for this we can ramp up the value of α_k in early iterations, e.g., $\alpha_k = \frac{k\alpha}{r+1}$ for the first $r+1$ iterations for fixed α where r is the memory parameter.

In many structured problems and in particular for tomography problems—where each block matrix corresponds to a single projection image (i.e., one angle)—$\mathbf{A}_k$ is extremely sparse with the number of non-zero elements in $\mathbf{A}_k$ on the order of n. Note that in some cases, matrix $\mathbf{A}_k$ does not even need to be constructed, but function evaluations can be used within iterative methods [64]. Hence, for extremely large-scale problems where memory storage becomes an issue, `slimLS` can take advantage of any sparsity or structure. In contrast, the `olbfgs` method requires storing two vectors of length n for each memory level, and these vectors are likely dense so the storage becomes cumbersome for large n.

4.3. Large-scale tomographic reconstruction

Next we present two numerical experiments that demonstrate the performance of `slimLS` for solving massive tomography reconstruction problems. Tomography has become very important in many applications, including medical imaging, seismic imaging, and atmospheric imaging



(a) (b)

Figure 6. Two-dimensional tomography problem with missing wedge of 60°. True image of a walnut slice is provided in (a) and the observed sinogram in (b) corresponds to angles between -60° and $+60^\circ$ at 0.3 degree steps.

[25, 42]. The goal in computerized tomography is to reconstruct the interior of an object given observed, exterior measurements. However, recent advances in detector technology have led to faster scan speeds and the collection of massive amounts of data. Furthermore, in dynamic or streaming data scenarios (e.g., in microCT reconstruction), partial reconstructions are needed to inform the data acquisition process. The `slimLS` method can be used to address both of these problems. The first example we consider is a very large, limited angle 2D tomography reconstruction problem that is underdetermined, and the second example is a 3D streaming reconstruction problem that is ultimately overdetermined.

For ill-posed problems such as tomography, semiconvergence of iterative methods is a concern whereby early iterates converge quickly to a good approximation of the solution but later iterates become contaminated with errors. Iterative regularization techniques (i.e., early termination of the iterative process) are often used to obtain a reasonable solution [57]. For row-action methods applied to tomography problems, semi convergence properties have been investigated [23], but due to notoriously slow convergence (after fast initial convergence), the ill effects of semiconvergence tend to appear only after multiple epochs of the data. Thus, we do not include additional regularization in the following results.

4.3.1. Two-dimensional limited-angle tomography. First we consider a parallel-beam x-ray tomography example, where the true image represents a cross-section of a walnut. In [65] the authors provide an image reconstruction computed from 1200 projections using filtered back projection. Our 'ground truth' image provided in figure 6(a) is a cleaned image of the filtered back projection reconstruction. For this problem, $\mathbf{x}_{\text{true}} \in \mathbb{R}^{2296^2}$, and we simulate observations by taking 400 projections at angles between -60° and $+60^\circ$ at 0.3 degree steps, with 2296 rays for each angle. This can be interpreted as a missing data problem where we have a missing wedge of 60°. In this example, $\mathbf{A} \in \mathbb{R}^{400\cdot 2296 \times 2296^2}$. The observed sinogram provided in figure 6(b)was generated as in (1) with noise level 0.01.

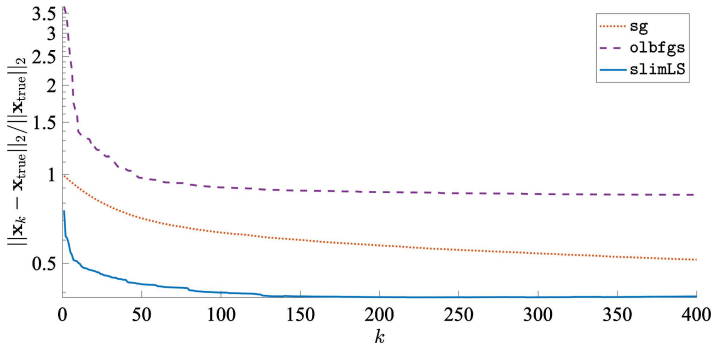


Figure 7. Relative reconstruction error norms at each iteration of `sg`, `olbfgs`, and `slimLS`.

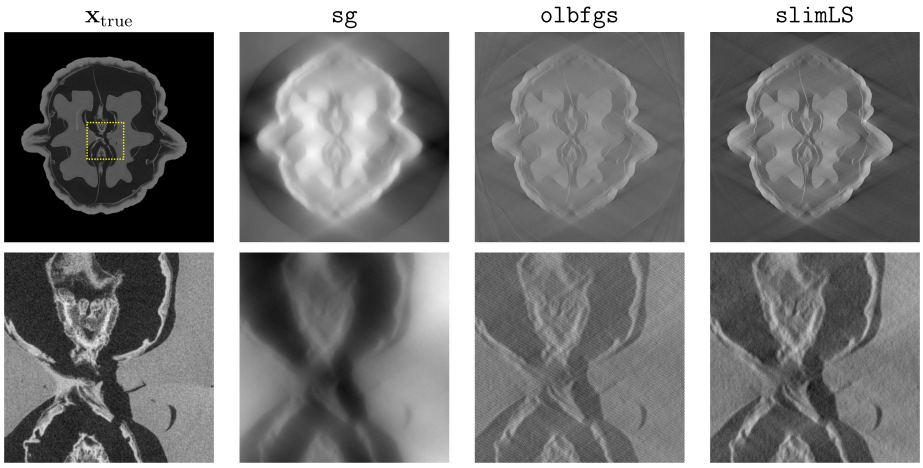


Figure 8. Image reconstructions for `sg`, `olbfgs` and `slimLS`, along with true image and with subimages provided below. The area of the subimage is denoted by the enclosed dotted region on the true image.

Here $\mathbf{W}^\top\mathbf{A}$ is random cyclic uniformly chosen from 400 blocks of size 2296×2296^2. We apply the `slimLS` method with memory level $r = 2$ and ramped-up damping parameter with $\alpha = 1$. Relative reconstruction errors computed as $\|\mathbf{x}_k - \mathbf{x}_{\text{true}}\|_2 / \|\mathbf{x}_{\text{true}}\|_2$ are provided in figure 7. We provide comparisons to the `sg` method with step size $\alpha_k = 10^{-5}$ and the `olbfgs` method with memory level 20 and ramped-up step size with $\alpha = 1$. We found that both methods required a small initial step size to prevent reconstruction errors from getting very large.

Image reconstructions and subimages, along with the true image, are provided in figure 8. We observe that after one epoch of the data, the `slimLS` reconstruction contains sharper details and fewer artifacts than the `sg` and `olbfgs` reconstructions. As described in section 4.2, it is difficult to provide a fair comparison of methods, especially in terms of the memory level and the step length. Careful tuning of the step length for `sg` and obfgs can lead to reconstructions that are similar in quality to the `slimLS` reconstruction, but that is very time consuming especially for these massive problems. Also, as observed in section 4.1, there may be only a

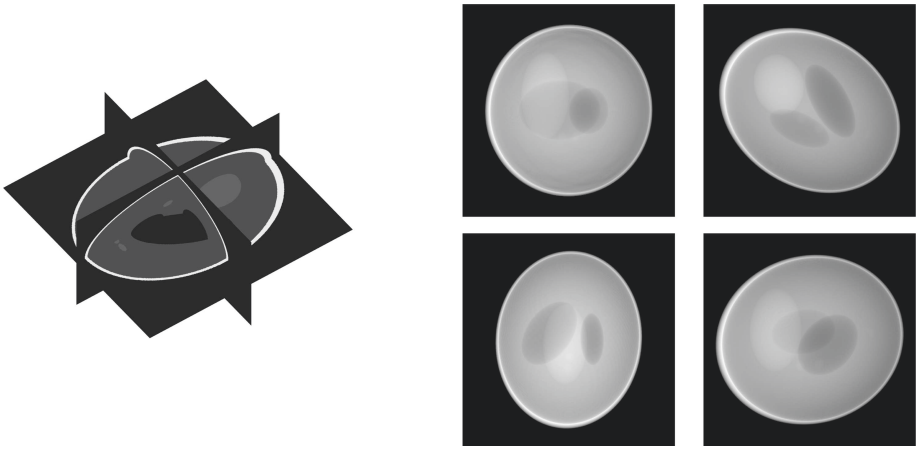


Figure 9. For the 3D tomography example, we provide orthogonal slices of the true image phantom (left) along with four of the observed projection images (right).

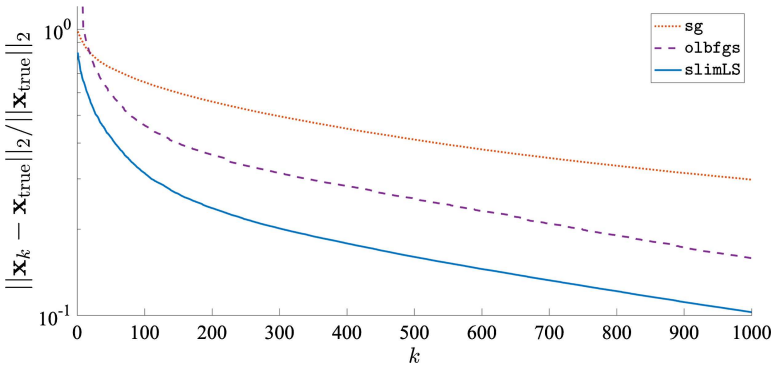


Figure 10. Relative reconstruction errors per iteration of `sg`, `olbfgs`, and `slimLS` for the 3D tomography example.

small window of good values. If good parameters are known in advance, `olbfgs` can produce good solutions, but if they are not known *a priori*, then poor results, or even divergence of the relative reconstruction errors, were often observed.

4.3.2. Three-dimensional streaming tomography. Next we demonstrate the power of `slimLS` for three-dimensional tomography reconstruction. For very large problems where data access is a computational bottleneck and for problems where data is being streamed and partial reconstructions are needed, row-action methods are the only feasible option. Here we show that sampled limited memory methods can be a good alternative.

In this problem setup, the true image is a $511 \times 511 \times 511$ modified 3D Shepp–Logan phantom. We generate 1000 projection images of size 511×511 taken from random directions, which are samples from the uniform distribution over the unit sphere. The raytracing matrix $\mathbf{A}$ is of size $261\,121\,000 \times 133\,432\,831$ and is certainly never constructed, but matrix blocks are generated using a modification of the tomobox code [66] which represents parallel beam

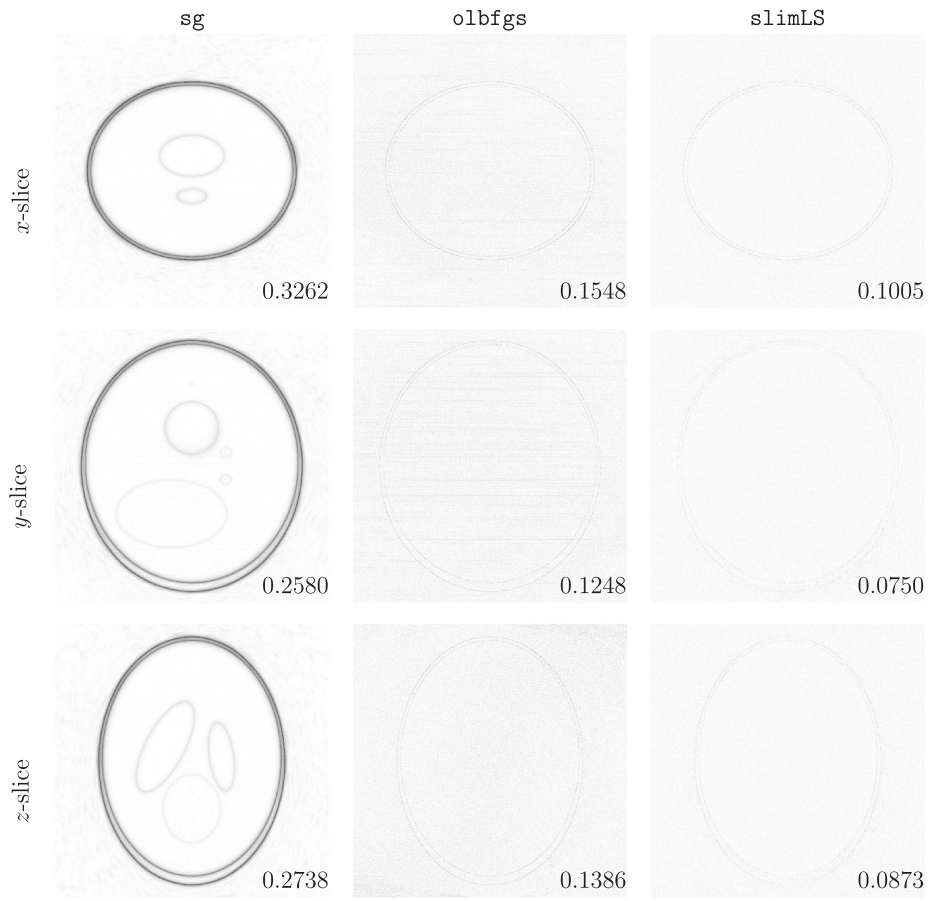


Figure 11. For the 3D tomographic reconstruction problem, we provide x, y, and z slices of absolute error images for `sg`, `olbfgs`, and `slimLS` reconstructions after one epoch of the data. The relative reconstruction error norm for each slice is provided in the bottom right corner of each image.

tomography. White noise was added to the projection images such that the noise level over all projection images is 0.001. Orthogonal slices of the true 3D phantom, along with four of the observed, projection images are provided in figure 9.

We run the `slimLS` method with $r = 0$ and damping factor $\alpha_k = 1$. After one epoch of the data (i.e., the cost of accessing all data once), we compare the `slimLS` reconstruction to the stochastic gradient and online LBFGS reconstructions. For `sg`, we use step length $\alpha_k = 0.0001$, and for `olbfgs`, we use $\alpha_k = 1$ and memory level 10. Relative reconstruction error norms per iteration are provided in figure 10. Notice that even in early iterations, where only a small fraction of the data has been accessed, `slimLS` reconstructions have smaller relative reconstruction errors than `sg` and `olbfgs`.

We computed absolute error images, which correspond to absolute values of the difference between the reconstruction and the true image, and we provide three slices (in the x, y, and z direction) for `sg`, `olbfgs`, and `slimLS` in figure 11. The absolute errors are provided in

inverted color map so that black corresponds to large errors. All images use the same color map. We observe that absolute error images for `sg` are significantly worse than those from `olbfgs` and `slimLS` reconstructions, with absolute error images for `slimLS` having fewer artifacts.

5. Conclusions

In this paper, we investigate sampled limited memory methods for solving massive inverse problems, such as those that arise in modern 2D and 3D tomography applications. Limited memory row-action methods are relevant in scenarios where *full* matrix-vector-multiplications with coefficient matrix $\mathbf{A}$ are not possible or too computationally expensive. This includes problems where $\mathbf{A}$ is so large that it does not fit in computer memory and problems where the data is being streamed. The main theoretical contribution is that, contrary to existing row-action and stochastic approximation methods, the `slimLS` method has both favorable initial and asymptotic convergence properties. Additional benefits of `slimLS` include faster initial convergence due to the use of information from previous iterates and convergence for a wider range of step length or damping parameters. We provide theoretical convergence results, and numerical examples from massive tomography reconstruction problems show the potential impact of these methods.

Acknowledgments

We gratefully acknowledge support by the National Science Foundation under grants NSF DMS 1723048 (L Tenorio), NSF DMS 1723005 (M Chung, J Chung) and NSF DMS 1654175 (J Chung). J Chung and M Chung would also like to acknowledge the Alexander von Humboldt Foundation for their generous support. The authors would like to thank the anonymous reviewers for pointing out important connections in the literature.

Appendix A. Randomized recursive LS method

In this section, we show that the kth iterate of the randomized recursive LS method defined in (9) is the minimum norm solution of LS problem in (10). We prove this by induction. For $k = 1$ and with $\mathbf{x}_0 = \mathbf{0}$, (9) yields $\mathbf{x}_1 = \mathbf{A}_1^{\dagger}\mathbf{b}_1$. Now let us assume that

$$\mathbf{x}_{k-1} = \left(\sum_{i=1}^{k-1} \mathbf{A}_i^{\top}\mathbf{A}_i\right)^{\dagger} \begin{bmatrix} \mathbf{A}_1 \\ \vdots \\ \mathbf{A}_{k-1} \end{bmatrix}^{\top} \begin{bmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_{k-1} \end{bmatrix}, \tag{A.1}$$

then from (9), we get

$$\begin{aligned} \mathbf{x}_k &= \mathbf{x}_{k-1} - \left(\sum_{i=1}^{k} \mathbf{A}_i^{\top}\mathbf{A}_i\right)^{\dagger} \mathbf{A}_k^{\top}\left(\mathbf{A}_k\mathbf{x}_{k-1} - \mathbf{b}_k\right) \\ &= \left(\sum_{i=1}^{k} \mathbf{A}_i^{\top}\mathbf{A}_i\right)^{\dagger} \left[\left(\sum_{i=1}^{k} \mathbf{A}_i^{\top}\mathbf{A}_i\right)\mathbf{x}_{k-1} - \mathbf{A}_k^{\top}\left(\mathbf{A}_k\mathbf{x}_{k-1} - \mathbf{b}_k\right)\right] \end{aligned} \tag{A.2}$$

where we are using the fact that $\mathbf{x}_{k-1} = \left(\sum_{i=1}^{k} \mathbf{A}_i^\top \mathbf{A}_i\right)^\dagger \left(\sum_{i=1}^{k} \mathbf{A}_i^\top \mathbf{A}_i\right) \mathbf{x}_{k-1}$ since $\mathbf{x}_{k-1} \in \mathcal{R}\left(\sum_{i=1}^{k-1} \mathbf{A}_i^\top \mathbf{A}_i\right) \subseteq \mathcal{R}\left(\sum_{i=1}^{k} \mathbf{A}_i^\top \mathbf{A}_i\right)$. A similar argument can be used to show that $\left(\sum_{i=1}^{k-1} \mathbf{A}_i^\top \mathbf{A}_i\right)\left(\sum_{i=1}^{k-1} \mathbf{A}_i^\top \mathbf{A}_i\right)^\dagger \begin{bmatrix} \mathbf{A}_1 \\ \vdots \\ \mathbf{A}_{k-1} \end{bmatrix}^\top \begin{bmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_{k-1} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_1 \\ \vdots \\ \mathbf{A}_{k-1} \end{bmatrix}^\top \begin{bmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_{k-1} \end{bmatrix}$ since $\begin{bmatrix} \mathbf{A}_1 \\ \vdots \\ \mathbf{A}_{k-1} \end{bmatrix}^\top \begin{bmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_{k-1} \end{bmatrix} \in \mathcal{R}\left(\sum_{i=1}^{k-1} \mathbf{A}_i^\top \mathbf{A}_i\right)$, and we arrive at

$$\mathbf{x}_k = \left(\sum_{i=1}^{k} \mathbf{A}_i^\top \mathbf{A}_i\right)^\dagger \left[\left(\sum_{i=1}^{k-1} \mathbf{A}_i^\top \mathbf{A}_i\right) \mathbf{x}_{k-1} + \mathbf{A}_k^\top \mathbf{b}_k\right] = \left(\sum_{i=1}^{k} \mathbf{A}_i^\top \mathbf{A}_i\right)^\dagger \begin{bmatrix} \mathbf{A}_1 \\ \vdots \\ \mathbf{A}_k \end{bmatrix}^\top \begin{bmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_k \end{bmatrix}.$$

Appendix B. Proofs for section 3

We continue to use the notation and assumptions from sections 1 and 3.

Lemma B.1. *If $\mathbf{A} \in \mathbb{R}^{m\times n}$ has full column-rank, then for any fixed $\alpha > 0$:*

(*a*) $\|\mathbb{E}\mathbf{B}_k/\alpha\|_2 < 1$ *with upper bound*

$$\|\mathbb{E}\,\mathbf{B}_k/\alpha\|_2 \leqslant \frac{M^{\mathrm{o}}}{M} + \frac{M - M^{\mathrm{o}}}{M(1+\alpha A_{\min})} < 1,$$

where M^{o} is the number of eigenvalues $\lambda_{\min}(\mathbf{A}^\top \mathbf{W}^{(i)}(\mathbf{W}(i))^\top \mathbf{A})$ equal to zero over the M different $\mathbf{W}^{(i)}$,

(*b*) $\mathbf{B}$ *is symmetric positive definite,*

(*c*) $\|\mathbf{B}_k \mathbf{A}_k^\top \mathbf{A}_k\|_2 \leqslant \alpha A_{\max}/(1+\alpha A_{\max})$, *and*

(*d*) $0 < \alpha A_{\min}/[M(1+\alpha A_{\min})] \leqslant \lambda_{\min}(\mathbf{B}) \leqslant \lambda_{\max}(\mathbf{B}) < (1+\alpha A_{\max})/2$.

Proof.

(a) For simplicity we use the notation $\widetilde{\mathbf{A}}_i = (\mathbf{W}^{(i)})^\top \mathbf{A}$. By Jensen's inequality

$$\|\mathbb{E}\,\mathbf{B}_k/\alpha\|_2 = \|\mathbb{E}\,(\mathbf{I} + \alpha\,\mathbf{A}_k^\top \mathbf{A}_k)^{-1}\|_2 \leqslant \mathbb{E}\,\|(\mathbf{I} + \alpha\,\mathbf{A}_k^\top \mathbf{A}_k)^{-1}\|_2 \leqslant 1.$$

The last inequality becomes an equality only if $\|(\mathbf{I} + \alpha\,\mathbf{A}_k^\top \mathbf{A}_k)^{-1}\|_2 = 1$ a.s., in which case $\lambda_{\max}(\mathbf{A}_k^\top \mathbf{A}_k) = 0$ across all realizations of $\mathbf{W}_k$. That is, $\|(\mathbf{I} + \alpha\,\widetilde{\mathbf{A}}_i^\top \widetilde{\mathbf{A}}_i)^{-1}\|_2 = 1$ for all i. Furthermore, an eigenvector $\mathbf{v}$ for the largest eigenvalue $\lambda = 1$ of $\mathbb{E}\,(\mathbf{I} + \alpha\,\mathbf{A}_k^\top \mathbf{A}_k)^{-1}$ is also an eigenvector of all $(\mathbf{I} + \alpha\,\widetilde{\mathbf{A}}_i^\top \widetilde{\mathbf{A}}_i)^{-1}$ with eigenvalue 1. That is, $\widetilde{\mathbf{A}}_i^\top \widetilde{\mathbf{A}}_i\,\mathbf{v} = \mathbf{0}$ for all i. This is not possible because $\mathbb{E}\,\mathbf{A}_k^\top \mathbf{A}_k = \beta\,\mathbf{A}^\top \mathbf{A}$. Hence, $\|\mathbf{B}_k/\alpha\|_2 < 1$. This implies that $M^{\mathrm{o}} < M$ and the upper bound in (a) follows again from $\|\mathbb{E}\,\mathbf{B}_k/\alpha\|_2 \leqslant \mathbb{E}\,\|(\mathbf{I} + \alpha\,\mathbf{A}_k^\top \mathbf{A}_k)^{-1}\|_2$.

(b) It is clear that $\mathbf{B}$ is symmetric positive semi-definite because $\mathbf{B} = \mathbf{I} - \mathbb{E}\,\mathbf{B}_k/\alpha$. It is positive definite by (a).

(c) The upper bound follows from the identity $\mathbf{B}_k \mathbf{A}_k^\top \mathbf{A}_k = \mathbf{I} - (\mathbf{I} + \alpha\,\mathbf{A}_k^\top \mathbf{A}_k)^{-1}$.

(d) From (c),

$$\lambda_{\max}(\mathbf{B}) \leqslant \frac{\alpha A_{\max}}{1+\alpha A_{\max}} < \frac{1+\alpha A_{\max}}{2}.$$

The lower bound follows from (a):

$$\begin{aligned}\lambda_{\min}(\mathbf{B}) \geqslant 1 - \mathbb{E}\,\|\mathbf{B}_k/\alpha\|_2 &\geqslant 1 - \frac{M^{\mathrm{o}}}{M} - \frac{M-M^{\mathrm{o}}}{M(1+\alpha A_{\min})} \\ &= \left(\frac{M-M^{\mathrm{o}}}{M}\right)\frac{\alpha A_{\min}}{1+\alpha A_{\min}} \geqslant \frac{\alpha A_{\min}}{M(1+\alpha A_{\min})}.\end{aligned}$$

□

Proof of theorem 3.1. We use $(\mathcal{F}_k)$ to denote the natural filtration induced by the sequence $\{\mathbf{W}_k\}$: $\mathcal{F}_k = \sigma(\mathbf{W}_j,\ j \leqslant k)$.

(a) Using the recursion for $\mathbf{x}_k$, we obtain

$$\begin{aligned}\mathbb{E}\left[\mathbf{x}_k - \widehat{\mathbf{x}} | \mathcal{F}_{k-1}\right] &= \mathbf{x}_{k-1} - \widehat{\mathbf{x}} - \mathbb{E}\left[\mathbf{B}_k\mathbf{A}_k^\top(\mathbf{A}_k\mathbf{x}_{k-1} - \mathbf{b}_k) \,|\, \mathcal{F}_{k-1}\right] \\ &= \mathbf{x}_{k-1} - \widehat{\mathbf{x}} - \mathbf{B}\left(\mathbf{x}_{k-1} - \mathbf{B}^{-1}\mathbb{E}\mathbf{B}_k\mathbf{A}_k^\top\mathbf{b}_k\right) \\ &= (\mathbf{I}-\mathbf{B})\left(\mathbf{x}_{k-1} - \widehat{\mathbf{x}}\right) = (\mathbb{E}\,\mathbf{B}_k/\alpha)\,\left(\mathbf{x}_{k-1} - \widehat{\mathbf{x}}\right),\end{aligned}$$

and therefore

$$\mathbb{E}\,\mathbf{x}_k - \widehat{\mathbf{x}} = (\mathbb{E}\,\mathbf{B}_k/\alpha)(\mathbb{E}\,\mathbf{x}_{k-1} - \widehat{\mathbf{x}}) = (\mathbb{E}\,\mathbf{B}_k/\alpha)^k(\mathbf{x}_0 - \widehat{\mathbf{x}}),$$

where the last equality comes from the fact that $\mathbf{A}_k$ are i.i.d. Using lemma B.1(a) we get

$$\|\mathbb{E}\mathbf{x}_k - \widehat{\mathbf{x}}\|_2 \leqslant \|\mathbb{E}\,\mathbf{B}_k/\alpha\|_2^k\|\mathbf{x}_0 - \widehat{\mathbf{x}}\|_2 \to 0.$$

Thus, $\mathbb{E}\,\mathbf{x}_k \to \widehat{\mathbf{x}}$ linearly.

(b) Next we show that $\mathbf{x}_k$ converges linearly to a convergence horizon of $\widehat{\mathbf{x}}$. The recursion of $\mathbf{x}_k$ leads to

$$\|\mathbf{x}_k - \widehat{\mathbf{x}}\|_2^2 = \|\mathbf{x}_{k-1} - \widehat{\mathbf{x}}\|_2^2 - 2\left(\mathbf{x}_{k-1} - \widehat{\mathbf{x}}\right)^\top\mathbf{B}_k\mathbf{A}_k^\top(\mathbf{A}_k\mathbf{x}_{k-1} - \mathbf{b}_k) + \|\mathbf{B}_k\mathbf{A}_k^\top(\mathbf{A}_k\mathbf{x}_{k-1} - \mathbf{b}_k)\|_2^2.$$

We find an upper bound for the last term using lemma B.1(a):

$$\begin{aligned}\|\mathbf{B}_k\mathbf{A}_k^\top(\mathbf{A}_k\mathbf{x}_{k-1} - \mathbf{b}_k)\|_2^2 &= \|\mathbf{B}_k\mathbf{A}_k^\top\mathbf{A}_k(\mathbf{x}_{k-1} - \widehat{\mathbf{x}}) + \mathbf{B}_k\mathbf{A}_k^\top(\mathbf{A}_k\widehat{\mathbf{x}} - \mathbf{b}_k)\|_2^2 \\ &\leqslant 2\|\mathbf{B}_k\mathbf{A}_k^\top\mathbf{A}_k(\mathbf{x}_{k-1} - \widehat{\mathbf{x}})\|_2^2 + 2\|\mathbf{B}_k\mathbf{A}_k^\top(\mathbf{A}_k\widehat{\mathbf{x}} - \mathbf{b}_k)\|_2^2 \\ &\leqslant 2\|\mathbf{B}_k\mathbf{A}_k^\top\mathbf{A}_k(\mathbf{x}_{k-1} - \widehat{\mathbf{x}})\|_2^2 + 2\alpha^2\|\mathbf{A}_k^\top(\mathbf{A}_k\widehat{\mathbf{x}} - \mathbf{b}_k)\|_2^2,\end{aligned}$$

and by lemma B.1(c)

$$\|\mathbf{B}_k\mathbf{A}_k^\top\mathbf{A}_k(\mathbf{x}_{k-1} - \widehat{\mathbf{x}})\|_2^2 \leqslant \frac{\alpha A_{\max}}{1+\alpha A_{\max}}(\mathbf{x}_{k-1} - \widehat{\mathbf{x}})^\top\mathbf{B}_k\mathbf{A}_k^\top\mathbf{A}_k(\mathbf{x}_{k-1} - \widehat{\mathbf{x}}).$$

The last two bounds yield

$$\|\mathbf{x}_k - \widehat{\mathbf{x}}\|_2^2 \leqslant \|\mathbf{x}_{k-1} - \widehat{\mathbf{x}}\|_2^2 - 2\left(\mathbf{x}_{k-1} - \widehat{\mathbf{x}}\right)^\top \mathbf{B}_k \mathbf{A}_k^\top (\mathbf{A}_k \mathbf{x}_{k-1} - \mathbf{b}_k) + \frac{2\alpha A_{\max}}{1 + \alpha A_{\max}} \left(\mathbf{x}_{k-1} - \widehat{\mathbf{x}}\right)^\top \mathbf{B}_k \mathbf{A}_k^\top \mathbf{A}_k \left(\mathbf{x}_{k-1} - \widehat{\mathbf{x}}\right) + 2\alpha^2 \|\mathbf{A}_k^\top (\mathbf{A}_k \widehat{\mathbf{x}} - \mathbf{b}_k)\|_2^2,$$

whose conditional expectation and lemma B.1(d) give us

$$\begin{aligned}\mathbb{E}\left[\|\mathbf{x}_k - \widehat{\mathbf{x}}\|_2^2 \,|\, \mathcal{F}_{k-1}\right] &\leqslant \|\mathbf{x}_{k-1} - \widehat{\mathbf{x}}\|_2^2 - 2\left(\mathbf{x}_{k-1} - \widehat{\mathbf{x}}\right)^\top \mathbf{B} \left(\mathbf{x}_{k-1} - \widehat{\mathbf{x}}\right) \\ &\quad + \frac{2\alpha A_{\max}}{1 + \alpha A_{\max}} \left(\mathbf{x}_{k-1} - \widehat{\mathbf{x}}\right)^\top \mathbf{B} \left(\mathbf{x}_{k-1} - \widehat{\mathbf{x}}\right) + 2\alpha^2 \mathbb{E}\|\mathbf{A}_k^\top (\mathbf{A}_k \widehat{\mathbf{x}} - \mathbf{b}_k)\|_2^2 \\ &= \|\mathbf{x}_{k-1} - \widehat{\mathbf{x}}\|_2^2 - \frac{2}{1 + \alpha A_{\max}} \left(\mathbf{x}_{k-1} - \widehat{\mathbf{x}}\right)^\top \mathbf{B} \left(\mathbf{x}_{k-1} - \widehat{\mathbf{x}}\right) \\ &\quad + 2\alpha^2\, \mathbb{E}\,\|\mathbf{A}_k^\top (\mathbf{A}_k \widehat{\mathbf{x}} - \mathbf{b}_k)\|_2^2 \\ &\leqslant (1 - 2c)\, \|\mathbf{x}_{k-1} - \widehat{\mathbf{x}}\|_2^2 + 2\alpha^2 \mathbb{E}\|\mathbf{A}_k^\top (\mathbf{A}_k \widehat{\mathbf{x}} - \mathbf{b}_k)\|_2^2,\end{aligned}$$

where $c = \lambda_{\min}(\mathbf{B})/(1 + \alpha A_{\max})$. Then, the expected squared norm of the error can be bounded using the fact that $0 < 1 - 2c < 1$ by lemma B.1(d)

$$\begin{aligned}\mathbb{E}\|\mathbf{x}_k - \widehat{\mathbf{x}}\|_2^2 &\leqslant (1 - 2c)^k \mathbb{E}\|\mathbf{x}_0 - \widehat{\mathbf{x}}\|_2^2 + 2\alpha^2 \mathbb{E}\|\mathbf{A}_k^\top (\mathbf{A}_k \widehat{\mathbf{x}} - \mathbf{b}_k)\|_2^2 \sum_{i=0}^{k-1} (1 - 2c)^i \\ &\leqslant (1 - 2c)^k \|\mathbf{x}_0 - \widehat{\mathbf{x}}\|_2^2 + \alpha^2 c^{-1} \mathbb{E}\|\mathbf{A}_k^\top (\mathbf{A}_k \widehat{\mathbf{x}} - \mathbf{b}_k)\|_2^2.\end{aligned}$$

□

Proof of lemma 3.2. By (15), $\widehat{\mathbf{x}} - \mathbf{x}_{\text{LS}} = \mathbf{B}^{-1}\mathbf{C}\mathbf{Q}_\mathbf{A}\mathbf{b}$, where

$$\mathbf{C} = \mathbb{E}\,\mathbf{B}_k \mathbf{A}_k^\top \mathbf{W}_k^\top.$$

Since $\mathbf{Q}_\mathbf{A}$ projects onto the orthogonal complement of the column space of $\mathbf{A}$, we have

$$\mathbf{B}^{-1}\mathbf{C}\mathbf{Q}_\mathbf{A} = (\mathbf{B}^{-1}\mathbf{C} - (\mathbf{A}^\top \mathbf{A})^{-1}\mathbf{A}^\top)\mathbf{Q}_\mathbf{A},$$

and also using lemma B.1(c)

$$\begin{aligned}\|\mathbf{B}^{-1}\mathbf{C} - (\mathbf{A}^\top \mathbf{A})^{-1}\mathbf{A}^\top\|_2 &= \|\mathbf{B}^{-1}\mathbb{E}\,(\mathbf{B}_k - \alpha\mathbf{I})\mathbf{A}_k^\top \mathbf{W}_k^\top + \alpha\mathbf{B}^{-1}\mathbb{E}\,\mathbf{A}_k^\top \mathbf{W}_k^\top - (\mathbf{A}^\top \mathbf{A})^{-1}\mathbf{A}^\top\|_2 \\ &\leqslant \|\mathbf{B}^{-1}\|_2 \|\mathbb{E}\,\alpha\mathbf{B}_k \mathbf{A}_k^\top \mathbf{A}_k \mathbf{A}_k^\top \mathbf{W}_k^\top\|_2 + \|\alpha\beta\mathbf{B}^{-1}\mathbf{A}^\top - (\mathbf{A}^\top \mathbf{A})^{-1}\mathbf{A}^\top\|_2 \\ &\leqslant \frac{\alpha^2 A_{\max}\, \|\mathbf{B}^{-1}\|_2}{1 + \alpha A_{\max}} \mathbb{E}\,\|\mathbf{A}_k^\top \mathbf{W}_k^\top\|_2 + \|\alpha\beta\mathbf{B}^{-1} - (\mathbf{A}^\top \mathbf{A})^{-1}\|_2\, \|\mathbf{A}\|_2.\end{aligned}$$

Furthermore,

$$\|\alpha\beta\mathbf{B}^{-1} - \left(\mathbf{A}^\top \mathbf{A}\right)^{-1}\|_2 \leqslant \|\mathbf{B}^{-1}\|_2 \|\alpha\beta\mathbf{A}^\top \mathbf{A} - \mathbf{B}\|_2 \|\left(\mathbf{A}^\top \mathbf{A}\right)^{-1}\|_2$$

and

$$\|\alpha\beta\mathbf{A}^\top \mathbf{A} - \mathbf{B}\|_2 = \|\mathbb{E}(\alpha\mathbf{I} - \mathbf{B}_k)\mathbf{A}_k^\top \mathbf{A}_k\|_2 \leqslant \frac{\alpha^2 A_{\max}}{1 + \alpha A_{\max}} \mathbb{E}\|\mathbf{A}_k^\top \mathbf{A}_k\|_2.$$

Using the upper bound for $\|\mathbf{B}^{-1}\|$ from lemma B.1(d) we can now write

$$\|\mathbf{B}^{-1}\mathbf{C} - (\mathbf{A}^\top\mathbf{A})^{-1}\mathbf{A}^\top\|_2 \leqslant \frac{\alpha^2 A_{\max}}{1+\alpha A_{\max}} \|\mathbf{B}^{-1}\|_2 \, C \leqslant \frac{\alpha A_{\max}}{1+\alpha A_{\max}} \frac{M(1+\alpha A_{\min})}{A_{\min}} \, C$$

where $C = \mathbb{E}\,\|\mathbf{A}_k^\top\mathbf{W}_k\|_2 + \|(\mathbf{A}^\top\mathbf{A})^{-1}\|_2\,\|\mathbf{A}\|_2\,\mathbb{E}\,\|\mathbf{A}_k^\top\mathbf{A}_k\|_2$. The upper bound finally follows from

$$\|\widehat{\mathbf{x}} - \mathbf{x}_{\mathrm{LS}}\|_2 = \|\mathbf{B}^{-1}\mathbf{C}\mathbf{Q}_{\mathbf{A}}\mathbf{b}\|_2 \leqslant \|\mathbf{B}^{-1}\mathbf{C} - (\mathbf{A}^\top\mathbf{A})^{-1}\mathbf{A}^\top\|_2\,\|\mathbf{Q}_{\mathbf{A}}\mathbf{b}\|_2.$$

□

ORCID iDs

Julianne Chung https://orcid.org/0000-0002-6760-4736
Matthias Chung https://orcid.org/0000-0001-7822-4539

References

[1] Parkinson D Y, Pelt D M, Perciano T, Ushizima D, Krishnan H, Barnard H S, MacDowell A A and Sethian J 2017 *Machine Learning for Micro-Tomography* (Developments in x-ray tomography XI, volume 10391) (International Society for Optics and Photonics) pp 103910J
[2] Parkinson D Y, Pacold J I, Gross M, McDougall T D, Jones C, Bows J, Hamilton I, Smiles D E, De Santis S and Ratti A *et al* 2018 *Achieving Fast High-Resolution 3D Imaging by Combining Synchrotron X-ray Micro CT, Advanced Algorithms, and High Performance Data Management* (Image Sensing Technologies: Materials, Devices, Systems, and Applications V106560S, vol 10656) (International Society for Optics and Photonics) https://doi.org/10.1117.12.2307272
[3] Paige C C and Saunders M A 1982 LSQR: an algorithm for sparse linear equations and sparse least squares *ACM Trans. Math. Softw.* **8** 43–71
[4] Avron H, Maymounkov P and Blendenpik S T 2010 Supercharging LAPACK's least-squares solver *SIAM J. Sci. Comput.* **32** 1217–36
[5] Halko N, Martinsson P G and Tropp J A 2011 Finding structure with randomness: probabilistic algorithms for constructing approximate matrix decompositions *SIAM Rev.* **53** 217–88
[6] Bottou L and Cun Y L 2004 Large scale online learning *Advances in Neural Information Processing Systems* pp 217–24
[7] Hand D J, Blunt G, Kelly M G and Adams N M *et al* 2000 Data mining for fun and profit *Stat. Sci.* **15** 111–31
[8] Rajaraman A and Ullman J D 2011 *Mining of Massive Datasets* (New York: Cambridge University Press)
[9] Zeng X Q and Li G Z 2014 Incremental partial least squares analysis of big streaming data *Pattern Recognit.* **47** 3726–35
[10] Boumal N, Bendory T, Lederman R R and Singer A 2018 Heterogeneous multi reference alignment: a single pass approach 52nd Annual Conf. on Information Sciences and Systems (CISS) (2018) (IEEE) pp 1–6
[11] Levin E, Bendory T, Boumal N, Kileel J and Singer A 2018 3D *ab initio* modeling in cryo-EM by autocorrelation analysis IEEE 15th Int. Symp. on Biomedical Imaging (ISBI 2018) (2018) (IEEE) pp 1569–73
[12] Chung J, Haber E and Nagy J 2006 Numerical methods for coupled super-resolution *Inverse Problems* **22** 1261
[13] Slagel J T, Chung J, Chung M, Kozak D and Tenorio L 2019 Sampled Tikhonov regularization for large linear inverse problems *Inverse Problems* **35** 114008
[14] Censor Y, Herman G T and Jiang M 2009 A note on the behavior of the randomized Kaczmarz algorithm of Strohmer and Vershynin *J. Fourier Anal. Appl.* **15** 431–6

[15] Eldar Y C and Needell D 2011 Acceleration of randomized Kaczmarz method via the Johnson–Lindenstrauss lemma *Numer. Algorithms* **58** 163–77
[16] Gower R M and Richtárik P 2015 Randomized iterative methods for linear systems *SIAM J. Matrix Anal. Appl.* **36** 1660–90
[17] Needell D 2010 Randomized Kaczmarz solver for noisy linear systems *BIT Numer. Math.* **50** 395–03
[18] Needell D, Srebro N and Ward R 2016 Stochastic gradient descent, weighted sampling, and the randomized Kaczmarz algorithm *Math. Program.* **155** 549–73
[19] Needell D and Tropp J A 2014 Paved with good intentions: analysis of a randomized block Kaczmarz method *Linear Algebr. Appl.* **441** 199–221
[20] Strohmer T and Vershynin R 2009 Comments on the randomized Kaczmarz method *J. Fourier Anal. Appl.* **15** 437–40
[21] Strohmer T and Vershynin R 2009 A randomized Kaczmarz algorithm with exponential convergence *J. Fourier Anal. Appl.* **15** 262–78
[22] Andersen M S and Hansen P C 2014 Generalized row-action methods for tomographic imaging *Numer. Algorithms* **67** 121–44
[23] Elfving T, Hansen P C and Nikazad T 2014 Semi-convergence properties of Kaczmarz's method *Inverse Problems* **30** 055007
[24] Gordon R, Bender R and Herman G T 1970 Algebraic reconstruction techniques (ART) for three-dimensional electron microscopy and x-ray photography *J. Theor. Biol.* **29** 471–81
[25] Natterer F 2001 *The Mathematics of Computerized Tomography* (Philadelphia, PA: SIAM)
[26] Censor Y, Eggermont P B and Gordon D 1983 Strong underrelaxation in Kaczmarz's method for inconsistent systems *Numer. Math.* **41** 83–92
[27] Hanke M and Niethammer W 1990 On the acceleration of Kaczmarz's method for inconsistent linear systems *Linear Algebr. Appl.* **130** 83–98
[28] Herman G T 2009 *Fundamentals of Computerized Tomography: Image Reconstruction from Projections* (Berlin: Springer)
[29] Elfving T 1980 Block-iterative methods for consistent and inconsistent linear equations *Numer. Math.* **35** 1–2
[30] Jiao Y, Jin B and Lu X 2017 Preasymptotic convergence of randomized Kaczmarz method *Inverse Problems* **33** 125012
[31] Censor Y and Zenios S A 1997 *Parallel Optimization: Theory, Algorithms, and Applications* (Oxford: Oxford University Press)
[32] Escalante R and Raydan M 2011 *Alternating Projection Methods* (Philadelphia, PA: SIAM)
[33] Chung J, Chung M, Slagel J T and Tenorio L 2017 Stochastic Newton and quasi-Newton methods for large linear least-squares problems (arXiv:1702.07367)
[34] Shapiro A, Dentcheva D and Ruszczyński A 2009 *Lectures on Stochastic Programming: Modeling and Theory* (Philadelphia, PA: SIAM)
[35] Bottou L 1998 Online learning and stochastic approximations *Online Learning in Neural Networks* (Cambridge: Cambridge University Press) ch 2 pp 9–42
[36] Schaul T, Zhang S and Cun Y L 2013 No more pesky learning rates Int. Conf. on Machine Learning pp 343–51
[37] Zeiler M D 2012 ADADELTA: an adaptive learning rate method (arXiv:1212.5701)
[38] Gower R M and Richtárik P 2017 Randomized quasi-Newton updates are linearly convergent matrix inversion algorithms *SIAM J. Matrix Anal. Appl.* **38** 1380–409
[39] Byrd R H, Hansen S L, Nocedal J and Singer Y 2016 A stochastic quasi-Newton method for large-scale optimization *SIAM J. Optim.* **26** 1008–31
[40] Needell D, Zhao R and Zouzias A 2015 Randomized block Kaczmarz method with projection for solving least squares *Linear Algebr. Appl.* **484** 322–43
[41] Feichtinger H G, Cenker C, Mayer M, Steier H and Strohmer T 1992 New variants of the POCS method using affine subspaces of finite co-dimension with applications to irregular sampling *Visual Communications and Image Processing'92* vol 1818 (International Society for Optics and Photonics) pp 299–311
[42] Herman G T and Meyer L B 1993 Algebraic reconstruction techniques can be made computationally efficient (positron emission tomography application) *IEEE Trans. Med. Imaging* **12** 600–9
[43] Whitney T M and Meany R K 1967 Two algorithms related to the method of steepest descent *SIAM J. Numer. Anal.* **4** 109–18
[44] Tanabe K 1971 Projection method for solving a singular system of linear equations and its applications *Numer. Math.* **17** 203–14

[45] Zouzias A and Freris N M 2013 Randomized extended Kaczmarz for solving least squares *SIAM J. Matrix Anal. Appl.* **34** 773–93
[46] Kaczmarz S 1937 Angenäherte Auflösung von Systemen linearer Gleichungen *Bulletin International de l'Académie Polonaise des Sciences et des Lettres. Classe des Sciences Mathématiques et Naturelles. Série A, Sciences Mathématiques* **35** 335–57
[47] Herman G T, Lent A and Lutz P H 1978 Relaxation methods for image reconstruction *Commun. ACM* **21** 152–8
[48] Popa C 1998 Extensions of block-projections methods with relaxation parameters to inconsistent and rank-deficient least-squares problems *BIT Numer. Math.* **38** 151–76
[49] Popa C and Zdunek R 2004 Kaczmarz extended algorithm for tomographic image reconstruction from limited-data *Math. Comput. Simul.* **65** 579–98
[50] Lorenz D A, Rose S and Schöpfer F 2018 The randomized Kaczmarz method with mismatched adjoint *BIT Numer. Math.* **58** 1079–98
[51] Gower R M, Goldfarb D and Richtarik P 2016 Stochastic block BFGS: squeezing more curvature out of data Int. Conf. on Machine Learning pp 1869–78
[52] Mokhtari A and Ribeiro A 2015 Global convergence of online limited memory BFGS *J. Mach. Learn. Res.* **16** 3151–81
[53] Chen Y, Dong G, Han J, Pei J, Wah B W and Wang J 2006 Regression cubes with lossless compression and aggregation *IEEE Trans. Knowl. Data Eng.* **18** 1585–99
[54] Kushner H and Yin G G 2003 *Stochastic Approximation and Recursive Algorithms and Applications* vol 35 (Berlin: Springer)
[55] Egidi N and Maponi P 2006 A Sherman–Morrison approach to the solution of linear systems *J. Comput. Appl. Math.* **189** 703–18
[56] Meng X, Saunders M A and LSRN M W M 2014 A parallel iterative solver for strongly over- or underdetermined systems *SIAM J. Sci. Comput.* **36** C95–118
[57] Hansen P C 2010 *Discrete Inverse Problems: Insight and Algorithms* (Philadelphia, PA: SIAM)
[58] Björck A 1996 *Numerical Methods for Least Squares Problems* (Philadelphia, PA: SIAM)
[59] Benveniste A, Wilson S S, Métivier M and Priouret P 2012 *Adaptive Algorithms and Stochastic Approximations* (New York: Springer)
[60] Bottou L and Cun Y L 2005 On-line learning for very large data sets *Appl. Stoch Model Bus. Ind.* **21** 137–51
[61] Nemirovski A, Juditsky A, Lan G and Shapiro A 2009 Robust stochastic approximation approach to stochastic programming *SIAM J. Optim.* **19** 1574–609
[62] Bottou L, Curtis F E and Nocedal J 2018 Optimization methods for large-scale machine learning *SIAM Rev.* **60** 223–311
[63] Paige C C and Saunders M A 1982 Algorithm 583, LSQR: sparse linear equations and least-squares problems *ACM Trans. Math. Softw.* **8** 195–209
[64] Golub G H and Van Loan C F 2012 *Matrix Computations* vol 3 (Baltimore, MA: JHU Press)
[65] Hämäläinen K, Harhanen L, Kallonen A, Kujanpää A, Niemi E and Siltanen S 2015 Tomographic x-ray data of a walnut (arXiv:1502.04064)
[66] Jorgensen J Tomobox https://mathworks.com/matlabcentral/fileexchange/28496-tomobox?s=prof_contriblnk (accessed September 2019)