A review of the Kaczmarz method

Carl Lokrantz

Bachelor's thesis
2023:K23

Lund University

Faculty of Science
Centre for Mathematical Sciences
Numerical Analysis

CENTRUM SCIENTIARUM MATHEMATICARUM

Abstract

The Kaczmarz method is an iterative method for solving linear systems of equations. The Kaczmarz method has been around since it was developed by Kaczmarz 1937. The main idea behind the original Kaczmarz method is to orthogonally project the previous x_k onto the solution space given by a row of the system. The block Kaczmarz on the other hand orthogonally projects the previous x_k onto the solution space given by a sub system of equations. Both the original Kaczmarz method and block Kaczmarz method can only solve consistent systems, however the extended Kaczmarz method is an adaptation that makes it possible to solve inconsistent systems. We will look at both deterministic and randomized, row and block selection processes then compare them on both consistent and inconsistent systems of equations.

Acknowledgement

I am grateful for my supervisor Andreas Langer and I really appreciate how he stuck with me even through constant delays and mistakes. This would not have been possible without his constant guidance, guiding me into the right direction when solving problems and challenging me with his questions to understand it better. Thank you Andreas. I would also like to thank my younger brother Axel and my classmates for the interesting discussions throughout my studies in Lund.

Contents

1 Introduction

The Kaczmarz method is one of many methods to solve linear system of equations and thus it could be of interest to first introduce what a linear system of equation is.

Definition 1 (Linear system of equations)**.** *A Linear system of equations is a collection of $m \in \mathbf{N}$ equations with $n \in \mathbf{N}$ unknown variables*

$$a_{1;1}x^{\{1\}} + a_{1;2}x^{\{2\}} + \cdots + a_{1;n}x^{\{n\}} = b_1$$

$$a_{2;1}x^{\{1\}} + a_{2;2}x^{\{2\}} + \cdots + a_{2;n}x^{\{n\}} = b_2$$

$$\vdots$$

$$a_{m;1}x^{\{1\}} + a_{m;2}x^{\{2\}} + \cdots + a_{m;n}x^{\{n\}} = b_m,$$

where $a_{i;j} \in \mathbf{R}$ and $b_i \in \mathbf{R}$ are constants and $x^{\{j\}}$ are our unknown variables.

Remark 1. *A linear system of equations can also be denoted as a matrix vector multiplication $Ax = b$ where $A \in \mathbf{R}^{m\times n}$ and $b \in \mathbf{R}^m$. Throughout the thesis we will use this definition for linear systems of equations.*

Definition 2 (Consistent system of equations)**.** *A system of equations is considered consistent if for a given $A \in \mathbf{R}^{m\times n}$ and $b \in \mathbf{R}^m$ there exists an $x \in \mathbf{R}^n$ such that $Ax = b$. If there exists no such x then the system is said to be inconsistent.*

If the problem is inconsistent it could still be interesting finding some $\hat{x}$ that minimizes the difference $||Ax - b||_2^2$.

Definition 3 (Least squares solution). *For any linear system of equations $Ax = b$ with $A \in \mathbf{R}^{m\times n}$ and $b \in \mathbf{R}^m$ the least squares solution is defined as the set of solutions satisfying*

$$\hat{x} := \arg\min_{x\in\mathbf{R}^n} ||Ax - b||_2^2.$$

The Kaczmarz method is an iterative method, where in each iteration x_{k+1} is the solution of an equation $a_{i_k} x = b_{i_k}$ which minimizes the distance to x_k i.e.

$$x_{k+1} = \arg\min_{\{x\in\mathbf{R}^n; a_{i_k} x = b_{i_k}\}} ||x - x_k||_2^2.$$

It selects equation in each iteration cyclically passing trough all equations once before returning to the first, meaning $i_k = k + 1 \mod m$.
Given any x_0 the Kaczmarz method converges towards a solution, if our system of equations has several solutions, the Kaczmarz method will converge towards the solution minimizing the distance between x_0 and the solution space $Ax = b$ i.e.

$$\hat{x} = \arg\min_{\{x\in\mathbf{R}^n; Ax=b\}} ||x - x_0||_2^2.$$

It should also be noted that not all variations of the Kaczmarz method selects the equations in a cyclical manner.

The Kaczmarz method was first discovered 1937, initially it had few to no applications, however as computers developed so did the need for iterative methods to solve linear systems. In 1970 the first application of the Kaczmarz method occurred, Gordon, Bender and Herman used it to reconstruct images from the data given from a computed tomography [3]. In 2008 Strohmer and Vershynin showed that the rate of convergence could be bound by the condition number of A, if a random row selection process was applied instead, this caused a boom in popularity of the Kaczmarz method [12]. Which gave room for the extended variations and block variations to develop.

In this thesis we will present some of the most well known variations of the Kaczmarz method and compare them. In the tests we see that the discreet variations slightly outperform the randomized variations and what the optimal order of operations for the extended Kaczmarz method is. Initially in section 2 we define some useful notation that will be used throughout the thesis.

We then move onto section 3 where we begin by presenting the original Kaczmarz method and show why it converges as well as how it converges presented by Kaczmarz [7]. We will also introduce the rate of converges for consistent systems with A being square and full rank presented in [2]. Then we move on to the randomized Kaczmarz method where we state the Algorithm and its rate of convergence presented by Strohmer and Vershynin [12], continuing with the rate of convergence presented by Needle [8] for noisy systems. Further we will look at how the Kaczmarz method can be utilized to transform an inconsistent system into a consistent system and the extended Kaczmarz method for solving inconsistent systems directly, both presented by Zouzias [14].

Then in section 4 we show the block Kaczmarz method presented by Needle and Tropp [9] and why the block Kaczmarz method converges, furthermore that it converges to the same solution as the original Kaczmarz method. Then we use the previous result to state the Kaczmarz method for the least squares problems which was showed by Needle, Zhao and Zouzias [10]. In the end we will present a variation of the block Kaczmarz method by Yu-Qi Niu and Bing Zheng [11].

In section 5 we will construct the tests and show a couple of adaptations of the methods constructed throughout the thesis and present the results. We also implement the method to solve an image reconstruction problem. We conclude with a summary and improvements 6.

2 Definitions

Definition 4. *Let $A \in \mathbf{R}^{m\times n}$, $a_i \in \mathbf{R}^{1\times n}$ denote the i'th row of A, i.e.*

$$A = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_{m-1} \\ a_m \end{bmatrix}.$$

Definition 5. *Let $b \in \mathbf{R}^m$, $b_i \in \mathbf{R}$ denote the i'th value of b, i.e.*

$$b = \begin{bmatrix} b_1, b_2, \dots, b_{m-1}, b_m \end{bmatrix}^T$$

Definition 6 (Inner product)**.** *Consider we have two equally sized vectors $x, y \in \mathbf{R}^m$, then the inner product $\langle x, y\rangle$ is defined to be*

$$\langle x, y\rangle := x^T y.$$

Definition 7 (Vector two-norm)**.** *The two-norm of $x \in \mathbf{R}^m$ is defined as*

$$||x||_2 := \sqrt{\langle x, x\rangle}.$$

Definition 8 (Matrix Frobenius-norm)**.** *Given $A \in \mathbf{R}^{m\times n}$,*

$$A = \begin{bmatrix} a_{1;1} & a_{1;2} & \dots & a_{1;n} \\ a_{2;1} & a_{2;2} & \dots & a_{2;n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m;1} & a_{m;2} & \dots & a_{m;n} \end{bmatrix},$$

let the Frobenius norm be defined as

$$||A||_F := \left(\sum_{i=1}^{m} \sum_{j=1}^{n} |a_{i;j}|^2 \right)^{1/2}.$$

Definition 9 (Hermitian and unitary matrices)**.** *Consider we have a matrix $A \in \mathbf{R}^{m\times n}$ then it is considered hermitian if $A = A^T$ and unitary if $A^T = A^{-1}$.*

Definition 10 (Eigenvectors and eigenvalues)**.** *Let $A \in \mathbf{R}^{m\times m}$ be a square matrix then the eigenvectors are the set of vectors $\mathbf{v} \in \mathbf{C}^n \setminus \mathbf{0}$ that solve*

$$A\mathbf{v} = \lambda\mathbf{v}$$

where $\lambda \in \mathbf{C}$ is the corresponding eigenvalue.

Definition 11 (Diagonalizable)**.** *A square matrix $A \in \mathbf{R}^{m\times m}$ is said to be diagonalizable if there exists an invertible V such that*

$$A = VDV^{-1}$$

where D is a diagonal matrix with the eigenvalues of A on its diagonal.

Theorem 1 (Singular value decomposition)**.** *For every matrix $A \in \mathbf{R}^{m\times n}$ there exists a decomposition*

$$A = USV^T$$

where $U \in \mathbf{R}^{m\times m}$, $V \in \mathbf{R}^{n\times n}$ are unitary and

$$S = \begin{bmatrix} \Sigma & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}$$

with Σ being a diagonal matrix with the singular values on its diagonal

$$\Sigma = \begin{bmatrix} \sigma_1 & 0 & \dots & 0 \\ 0 & \sigma_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \sigma_r \end{bmatrix}.$$

Proof. See [5] page 330 -332 theorem 13.2 □

Definition 12 (Matrix two-norm). *Consider a matrix $A \in \mathbf{R}^{m\times n}$ then the two-norm is*

$$||A||_2 := \sigma_{\max}$$

where $\sigma_{\max}$ is the largest singular value of A. $\sigma_{\min}$ is the smallest non-zero singular value of A and

$$||A^\dagger||_2 = \frac{1}{\sigma_{\min}}.$$

Definition 13 (Induced matrix norm). *Let $A \in \mathbf{R}^{m\times n}$ and $1 \le \mu \le \infty$ then the Induced matrix norm is defined as*

$$||A||_\mu := \sup_{x\in\mathbf{R}^n\setminus\mathbf{0}} \frac{||Ax||_\mu}{||x||_\mu}.$$

Definition 14 (Moore-Penrose pseudoinverse). *The Moore-Penrose pseudoinverse of A is defined as the matrix $A^\dagger$ satisfying the following equalities*

$$AA^\dagger A = A,$$

$$A^\dagger AA^\dagger = A^\dagger,$$

$$\left(A^\dagger A\right)^T = A^\dagger A,$$

$$\left(AA^\dagger\right)^T = AA^\dagger.$$

Definition 15 (Orthogonal Projection). *Suppose we have a matrix $A \in \mathbf{R}^{m\times m}$, it is said to be an orthogonal projection if*

$$\langle Ax, y - Ay\rangle = 0$$

for any $x, y \in \mathbf{R}^m$.

Theorem 2. *We have a matrix $A \in \mathbf{R}^{m\times m}$ then it is an orthogonal projection if*

$$A^T = A$$

$$A^2 = A.$$

Proof. We have that

$$\langle Ax, y - Ay\rangle = \langle Ax, y\rangle - \langle Ax, Ay\rangle = 0 \iff \langle Ax, y\rangle = \langle Ax, Ay\rangle$$

and similarly

$$\langle x - Ax, Ay\rangle = 0 \iff \langle x, Ay\rangle = \langle Ax, Ay\rangle.$$

Meaning we have

$$\langle Ax, y\rangle = \langle x, Ay\rangle = \langle A^T x, y\rangle,$$

implying that $A^T = A$. Then from

$$\langle Ax, Ay\rangle = \langle A^T Ax, y\rangle = \langle AAx, y\rangle = \langle Ax, y\rangle$$

we get $A^2 = A$.

□

Definition 16 (Condition number). *Given a matrix $A \in \mathbf{R}^{m\times n}\setminus\{\mathbf{0}\}$ then the condition number is defined as*

$$\kappa(A) := ||A||_2||A^\dagger||_2.$$

We define a condition number that is both induced by the Frobenius norm and the two-norm i.e.,

$$\kappa(A)_F := ||A||_F||A^\dagger||_2,$$

$$\kappa(A) \le \kappa(A)_F \le \sqrt{m}\kappa(A).$$

Where the final inequality comes from

$$||A||_2 \le ||A||_F \le \sqrt{m}||A||_2.$$

Definition 17 (Linear Convergence [13]). *Suppose* $\lim_{k\to\infty} x_k = \hat{x}$. *We say that the sequence* $\{x_k\}_{k=0}^{\infty}$ *converges to* $\hat{x}$ *at least linearly if there exists a* $C \in (0, 1)$ *such that*

$$\lim_{k\to\infty} \frac{||x_{k+1} - \hat{x}||_2}{||x_k - \hat{x}||_2} = C.$$

Remark 2. *The sequence is said to have superlinear convergence if* $C = 0$*, and sublinear convergence when* $C = 1$.

Definition 18 (Positive definite [1]). *A symmetric matrix* $H \in \mathbf{R}^{n\times n}$ *is said to be positive definite if*

$$x^T H x > 0$$

for all $x \in \mathbf{R}^n \setminus \mathbf{0}$. H *is called positive semi-definite if*

$$x^T H x \geq 0$$

for all $x \in \mathbf{R}^n$.

3 Kaczmarz method

3.1 Original Kaczmarz method

We introduce the original Kaczmarz method, which was first presented by Kaczmarz in 1937 [7]. The original Kaczmarz method only solves consistent systems of equations $Ax = b$ and in the original work [7] it was limited to square matrices with full rank with all rows satisfying $||a_i||_2 = 1$.

Algorithm 1 Original Kaczmarz method [7]

procedure (A, b, x_0, N, ϵ) ▷ $A \in \mathbf{R}^{m\times n}$, $b \in \mathbf{R}^m$, $x_0 \in \mathbf{R}^n$, $N \in \mathbf{N}$, $\epsilon \in \mathbf{R}$
  A and b describes our system of equations
  x_0 is our initial point
  N determines the maximum amount of iterations
  ϵ how accurate we want to be
  repeat
    $i_k = k + 1 \mod m$
    if $||a_{i_k}^T||_2^2 \neq 0$ **then**

$$x_{k+1} = x_k + \frac{b_{i_k} - \langle a_{i_k}^T, x_k\rangle}{||a_{i_k}^T||_2^2} a_{i_k}^T$$

    else
      $x_{k+1} = x_k$
    if $k = 0 \mod m$ **and** $||Ax_{k+1} - b||_2^2 \leq \epsilon$ **then**
      return x_{k+1}
      terminate
  until $k + 1 > Nm$
return x_{k+1}
terminate

Statement 1. *If* x_{k+1} *is defined by Algorithm 1 as*

$$x_{k+1} = x_k + \frac{b_{i_k} - \langle a_{i_k}^T, x_k\rangle}{||a_{i_k}^T||_2^2} a_{i_k}^T$$

and $i_k \in \{1, 2, \ldots, m\}$*, then* x_{k+1} *solves the* i_k*'th equation*

$$a_{i_k} x_{k+1} = b_{i_k}.$$

Proof. We can write $a_{i_k} x_{k+1}$ as

$$a_{i_k} x_{k+1} = a_{i_k} x_k + a_{i_k} \frac{b_{i_k} - \langle a_{i_k}^T, x_k\rangle}{||a_{i_k}^T||_2^2} a_{i_k}^T$$

using the definition of x_{k+1}. Then we simplify it and get

$$a_{i_k} x_{k+1} = \langle a_{i_k}^T, x_k \rangle + ||a_{i_k}^T||_2^2 \frac{b_{i_k} - \langle a_{i_k}^T, x_k \rangle}{||a_{i_k}^T||_2^2} = \langle a_{i_k}^T, x_k \rangle + b_{i_k} - \langle a_{i_k}^T, x_k \rangle = b_{i_k}.$$

□

Definition 19 (Minimum norm solution)**.** *Given a linear system of equations $Ax = b$ with $A \in \mathbf{R}^{m \times n}$ and $b \in \mathbf{R}^n$ and the set of least squares solutions $\hat{\omega} = \{x \in \mathbf{R}^n; \arg\min_{x \in \mathbf{R}^n} ||Ax - b||_2^2\}$, then the minimum norm solution is*

$$\tilde{x} := \arg\min_{x \in \hat{\omega}} ||x||_2^2.$$

Theorem 3 (Normal equation [5])**.** *If we have a least squares problem $\hat{\omega} = \{x \in \mathbf{R}^n; \arg\min_{x \in \mathbf{R}^n} ||Ax - b||_2^2\}$ then we can transform our problem into a consistent problem given by the normal equation*

$$A^T A x = A^T b,$$

where the solution is $x \in \hat{\omega}$.

Proof. See [5] page 391-392 theorem 15.10. □

Lemma 1. *For any $A \in \mathbf{R}^{m \times n}$ if $A^T A y = \mathbf{0}$, then $y \in \ker(A)$.*

Proof. By the singular value decomposition we have

$$A^T A = V S^T U^T U S V^T = V S^T S V^T.$$

Now we want to show that if $V S^T S V^T y = \mathbf{0}$ then $Ay = \mathbf{0}$.
We have

$$\begin{aligned} & V S^T S V^T y = \mathbf{0} \\ \iff & V^T V S^T S V^T y = \mathbf{0} \\ \iff & S^T S V^T y = \mathbf{0} \\ \implies & \left(S^\dagger\right)^T S^T S V^T y = \mathbf{0} \end{aligned} \tag{1}$$

where the equivalences comes form V being unitary and the implication comes from the multiplication by $\left(S^\dagger\right)^T$. We can write $S \in \mathbf{R}^{m \times n}$ and $S^T \in \mathbf{R}^{n \times m}$ as a block matrices where Σ is a diagonal matrix with the singular values of A on its diagonal

$$S = \begin{bmatrix} \Sigma & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}$$

and the corresponding $\left(S^\dagger\right)^T \in \mathbf{R}^{m \times n}$

$$S^\dagger = \begin{bmatrix} \Sigma^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}.$$

Now we simplify the left hand side of the equation

$$\begin{aligned} \left(S^\dagger\right)^T S^T S V^T y &= \begin{bmatrix} \Sigma^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Sigma & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} S V^T y \\ &= \begin{bmatrix} I & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} S V^T y \\ &= S V^T y. \end{aligned} \tag{2}$$

Since we multiply both sides with U and get

$$U S V^T y = \mathbf{0}$$

then by the definition of the singular value decomposition we have

$$Ay = \mathbf{0}$$

and thus $y \in \ker(A)$.

□

Lemma 2. *Given a* $\hat{\omega} = \{x \in \mathbf{R}^n; \arg\min_{x\in\mathbf{R}^n} ||Ax - b||_2^2\}$ *let* $\hat{x}$ *any element in* $\hat{\omega}$*, and let* $\tilde{x}$ *be the minimum norm solution to* $\hat{\omega}$*. Then* $\hat{x}$ *can be written as* $\hat{x} = \tilde{x} + y$ *where* $y \in \ker(A)$*.*

Proof. We begin as by writing $\hat{x}$ as the solution to the normal equation

$$A^T A\hat{x} = A^T b.$$

By decomposing into $\hat{x} = \tilde{x} + y$ we get

$$A^T A\tilde{x} + A^T Ay = A^T b$$

since $A^T A\tilde{x} = A^T b$ we have $A^T Ay = \mathbf{0}$ which by lemma 1 implies that $y \in \ker(A)$.

□

Lemma 3 ([5]). *If we have a linear system of equations* $\hat{\omega} = \{x \in \mathbf{R}^n; \arg\min_{x\in\mathbf{R}^n} ||Ax - b||_2^2\}$ *then* $A^\dagger b$ *a solution to the normal equation.*

Proof. Using the singular value decomposition of A we have

$$A^T AA^\dagger b = VS^T U^T USV^T VS^\dagger U^T b,$$

however we know that U and V are unitary so it simplifies to

$$A^T AA^\dagger b = VS^T SS^\dagger U^T.$$

We can write $S \in \mathbf{R}^{m\times n}$ and $S^T \in \mathbf{R}^{n\times m}$ as a block matrix where Σ is a diagonal matrix with the singular values of A on the diagonal

$$S = \begin{bmatrix} \Sigma & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}$$

and $S^\dagger \in \mathbf{R}^{n\times m}$ the corresponding matrix

$$S^\dagger = \begin{bmatrix} \Sigma^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}.$$

Thus we have

$$VSSS^\dagger U^T = V \begin{bmatrix} \Sigma & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Sigma & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Sigma^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} U^T = V \begin{bmatrix} \Sigma & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} U^T = A^T,$$

giving us $A^T AA^\dagger b = A^T b$.

□

Lemma 4. *Since* A *projects onto a linear subspace we have* $Ax =$

Proof. Since P is an orthogonal projection we know $P = P^T = P^2$. We have that

$$(I - P)^T = I^T - P^T = I - P$$

and that

$$(I - P)^2 = I^2 - 2IP + P^2 = I - P.$$

Thus we know that both P and $I - P$ are orthogonal projections. It remains to show that $Im(P)$ is orthogonal to $Im(I - P)$ given any vector $x \in \mathbf{R}^n$ we have

$$\begin{aligned} \langle Px, (I - P)x\rangle &= \langle Px, x\rangle - \langle Px, Px\rangle \\ &= (Px)^T x - (Px)^T Px \\ &= x^T P^T x - x^T P^2 x \\ &= x^T Px - x^T Px = 0. \end{aligned} \tag{3}$$

We know that $Im(P)$ and $Im(I-P)$ are subsets of $\mathbf{R}^n$ and for any $x \in \mathbf{R}^n$ we have $Px+Ix-Px = x$ resulting in $Im(P) \oplus Im(I - P) = \mathbf{R}^n$ □

Lemma 5. *For any matrix* $A \in \mathbf{R}^{m\times n}$ *we have that* $I - A^\dagger A$ *is an orthogonal projection and* $Im(I - A^\dagger A) = \ker(A)$*.*

Proof. From lemma 4 it is enough to show that $A^\dagger A$ is an orthogonal projection. We use the singular value decomposition to show that $(A^\dagger A)^T = A^\dagger A$

$$\begin{aligned}(A^\dagger A)^T &= A^T (A^\dagger)^T \\ &= VSU^T US^\dagger V^T \\ &= V \begin{bmatrix} I & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} V^T \\ &= A^\dagger A.\end{aligned} \tag{4}$$

We expand and then use the singular value decomposition to show that $(A^\dagger A)^2 = A^\dagger A$

$$\begin{aligned}(A^\dagger A)^2 &= A^\dagger A A^\dagger A \\ &= VS^\dagger U^T USV^T VS^\dagger U^T USV^T \\ &= V \begin{bmatrix} I & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} I & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} V^T \\ &= A^\dagger A.\end{aligned} \tag{5}$$

Now we want to show that $Im(I - A^\dagger A) = \ker(A)$. We begin by choosing any element $y \in \ker(A)$ then

$$(I - A^\dagger A)y = y - \mathbf{0} = y$$

which shows that $y \in Im(I - A^\dagger A)$. Consider any element $\hat{y} \in Im(I - A^\dagger A)$ which can be written as $(I - A^\dagger A)x$ where $x \in \mathbf{R}^n$. We have that

$$A\hat{y} = A(I - A^\dagger A)x = Ax - AA^\dagger Ax = Ax - Ax = \mathbf{0}$$

and thus $\hat{y} \in Im(I - A^\dagger A)$ implying that $Im(I - A^\dagger A) = \ker(A)$. □

Lemma 6. *Suppose we have a least squares problem $\hat{\omega} = \{x \in \mathbf{R}^n; \arg\min_{x \in \mathbf{R}^n} ||Ax - b||_2^2\}$ and any $\hat{x} \in \hat{\omega}$ is written as $\hat{x} = \tilde{x} + y$ where $y \in \ker(A)$ and $\tilde{x}$ is the minimum norm solution. Then $\tilde{x}$ is also orthogonal to y.*

Proof. We orthogonaly decompose $\hat{x}$ into $\hat{x} = x_1 + \bar{y}$ where $x_1 \in Im(A^\dagger A)$ and $\bar{y} \in \ker(A)$. We have that

$$\begin{aligned}A^T A x_1 &= A^T A(\hat{x} - \bar{y}) \\ &= A^T b - \mathbf{0} \\ &= A^T b\end{aligned} \tag{6}$$

showing that x_1 is a solution to normal equation.

We use both expression for $\hat{x}$ to get the following equalities

$$\hat{x} = x_1 + \bar{y} = \tilde{x} + y$$

$$\tilde{x} = x_1 + \bar{y} - y$$
$$||\tilde{x}||_2^2 = ||x_1 + \bar{y} - y||_2^2$$

however since x_1 is orthogonal to $\ker(A)$ we get

$$||\tilde{x}||_2^2 = ||x_1||_2^2 + ||\bar{y} - y||_2^2.$$

Since $||\tilde{x}||_2^2$ is the minimum norm solution we have $||x_1||_2^2 \geq ||\tilde{x}||_2^2$ and thus get

$$||\tilde{x}||_2^2 \geq ||\tilde{x}||_2^2 + ||\bar{y} - y||_2^2$$
$$0 \geq ||\bar{y} - y||_2^2$$

which only holds when $\bar{y} = y$ yielding us

$$\tilde{x} = x_1 + y - y$$

$$\tilde{x} = x_1.$$

□

Lemma 7. *If we have a linear system then the solution $A^\dagger b$ is the minimum norm solution $\tilde{x}$ in particular.*

Proof. Using the definition of the Moore-Penrose pseudoinverse we require the following equality to hold

$$A^\dagger A A^\dagger b = A^\dagger b.$$

Using lemma 3 and lemma 2 we get

$$A^\dagger A\hat{x} = \hat{x}$$
$$A^\dagger A\tilde{x} + A^\dagger Ay = \tilde{x} + y$$

since y is in the kernel of A we have $Ay = \mathbf{0}$ yielding us

$$A^\dagger A\tilde{x} = \tilde{x} + y.$$

Taking the norm gets us

$$||A^\dagger A\tilde{x}||_2^2 = ||\tilde{x} + y||_2^2.$$

Using the singular value decomposition we expand $||A^\dagger A\tilde{x}||_2^2$ to give us a bound

$$\begin{aligned} ||A^\dagger A\tilde{x}||_2^2 &= ||VS^\dagger SV^T\tilde{x}||_2^2 \\ &= \left|\left|V \begin{bmatrix} I & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} V^T\tilde{x}\right|\right|_2^2 \\ &\leq ||VIV^T\tilde{x}||_2^2 = ||\tilde{x}||_2^2. \end{aligned} \tag{7}$$

This yields

$$||\tilde{x}||_2^2 \geq ||\tilde{x} + y||_2^2$$

then from lemma 6 we get

$$||\tilde{x}||_2^2 \geq ||\tilde{x}||_2^2 + ||y||_2^2$$

however this inequality only holds when $y = 0$ meaning $A^\dagger b = \tilde{x}$. □

Theorem 4. *Letting $a \in \{\mathbf{R}^m \setminus \mathbf{0}\}$ we have*

$$a^\dagger = \frac{1}{||a||_2^2} a^T.$$

Proof. See [5] page 388. □

Statement 2. *If x_{k+1} is defined by Algorithm 1*

$$x_{k+1} = x_k + \frac{b_{i_k} - \langle a_{i_k}^T, x_k\rangle}{||a_{i_k}^T||_2^2} a_{i_k}^T$$

and ω is the set $\omega = \{x \in \mathbf{R}^n; a_{i_k}x = b_{i_k}\}$ then x_{k+1} is the orthogonal projection of x_k onto ω.

Proof. From statement 1 we have that $x_{k+1} \in \omega$ and thus is a projection. It remains to prove that the projection is orthogonal. Rearranging the equation we get

$$x_k + \frac{b_{i_k} - \langle a_{i_k}^T, x_k\rangle}{||a_{i_k}^T||_2^2} a_{i_k}^T = x_k - \frac{a_{i_k}^T a_{i_k}}{||a_{i_k}^T||_2^2} x_k + \frac{1}{||a_{i_k}^T||_2^2} a_{i_k}^T b_{i_k}.$$

We see that the final part $\frac{1}{||a_{i_k}^T||_2^2} a_{i_k}^T b_{i_k} = a_{i_k}^\dagger b$ transforms our solution onto the solution space. Thus it remains to show that

$$x_k - \frac{a_{i_k}^T a_{i_k}}{||a_{i_k}^T||_2^2} x_k$$

is an orthogonal projection. Defining

$$P := I - \frac{a_{i_k}^T a_{i_k}}{||a_{i_k}^T||_2^2}$$

we have

$$x_k - \frac{a_{i_k}^T a_{i_k}}{||a_{i_k}^T||_2^2} x_k = Px_k.$$

By showing that $P^T = P$ and $P^2 = P$ we prove that it is an orthogonal projection. We have

$$P^T = \left(I - \frac{a_{i_k}^T a_{i_k}}{||a_{i_k}^T||_2^2}\right)^T = I^T - \left(\frac{a_{i_k}^T a_{i_k}}{||a_{i_k}^T||_2^2}\right)^T$$

however both I and $a_{i_k}^T a_{i_k}$ are hermitian and thus we have

$$P^T = I - \frac{a_{i_k}^T a_{i_k}}{||a_{i_k}^T||_2^2} = P$$

and

$$\begin{aligned} P^2 &= \left(I - \frac{a_{i_k}^T a_{i_k}}{||a_{i_k}^T||_2^2}\right)\left(I - \frac{a_{i_k}^T a_{i_k}}{||a_{i_k}^T||_2^2}\right) = I - 2\frac{a_{i_k}^T a_{i_k}}{||a_{i_k}^T||_2^2} + \frac{a_{i_k}^T a_{i_k}}{||a_{i_k}^T||_2^2}\frac{a_{i_k}^T a_{i_k}}{||a_{i_k}^T||_2^2} \\ &= I - 2\frac{a_{i_k}^T a_{i_k}}{||a_{i_k}^T||_2^2} + \frac{a_{i_k}^T ||a_{i_k}^T||_2^2 a_{i_k}}{||a_{i_k}^T||_2^4} = I - 2\frac{a_{i_k}^T a_{i_k}}{||a_{i_k}^T||_2^2} + \frac{a_{i_k}^T a_{i_k}}{||a_{i_k}^T||_2^2} = I - \frac{a_{i_k}^T a_{i_k}}{||a_{i_k}^T||_2^2} = P. \end{aligned} \tag{8}$$

□

To illustrate the convergence of the Kaczmarz method we construct two consistent systems and compare the first 4 iterations. One is constructed such that the lines are relatively close to being orthogonal and thus converges relatively quickly, while the other is constructed to show the method when they are relatively far from being orthogonal and thus converges slower. The system $Ax = b$ in figure 1 is given by

$$A = \begin{bmatrix} 10 & 1 \\ 1 & 10 \end{bmatrix},$$

$$b = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

setting the initial point to

$$x_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

While the system $Ax = b$ in figure 2 is given by

$$A = \begin{bmatrix} 2 & 1 \\ 2 & 3 \end{bmatrix},$$

$$b = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

setting the initial point to

$$x_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

In the figure 1, 2 we have the solution space for all equations in our system represented by blue lines where the solution is the intersection. The red dots represents the respective points after $k \in \{0, 1, 2, 3, 4\}$ iterations, the cyanide line represents the orthogonal projection onto the solution space of the equations and the gray represents our previous projections.
Comparing the figures we see that it converges quicker in figure 1 where the solution spaces are close to orthogonal, than in figure 2.

Lemma 8 (Cauchy-Schwarz inequality [13]). *Let $a, b \in \mathbf{R}^n$ then*

$$\langle a, b \rangle^2 \leq ||a||_2^2 ||b||_2^2.$$

Proof. See [13] page 59 lemma 2.2. □

Theorem 5 (Original Kaczmarz method [7]). *If we have a consistent system of equations given by the set $\Omega = \{x \in \mathbf{R}^n; Ax = b\}$ where $A \in \mathbf{R}^{m \times n}$, all $||a_i||_2^2 \neq 0$ and $b \in \mathbf{R}^m$. Given any $x_0 \in \mathbf{R}^n$ the sequence $\{x_{k+1}\}_0^\infty$ given by Algorithm 1 converges towards the limit point $\hat{x} \in cl(\Omega)$ where $\hat{x}$ is the unique solution to $\hat{x} = \arg\min_{x \in \Omega} ||x - x_0||$.*

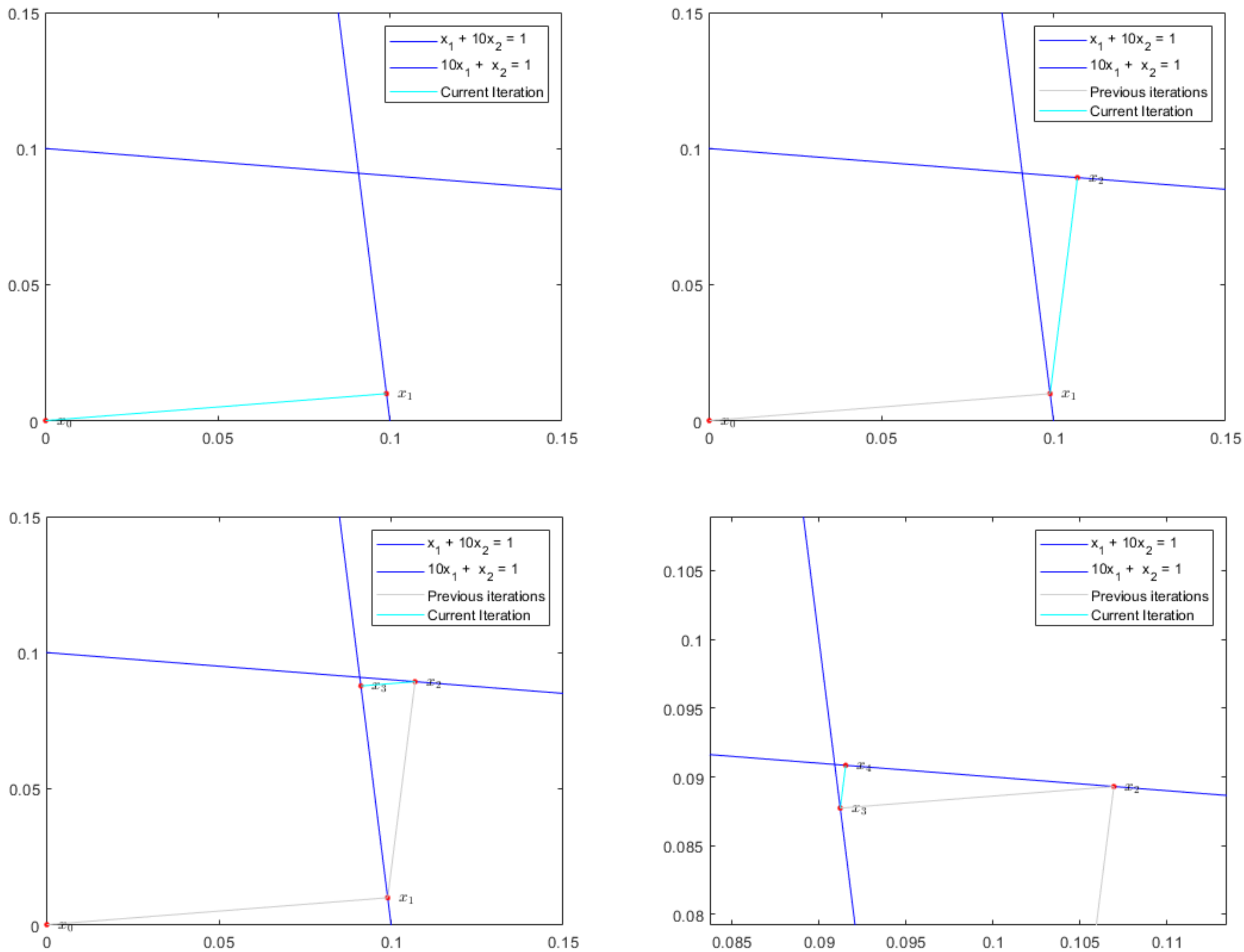

Figure 1: Illustrates the path towards the solution of Algorithm 1 of a system that has almost orthogonal solution spaces.

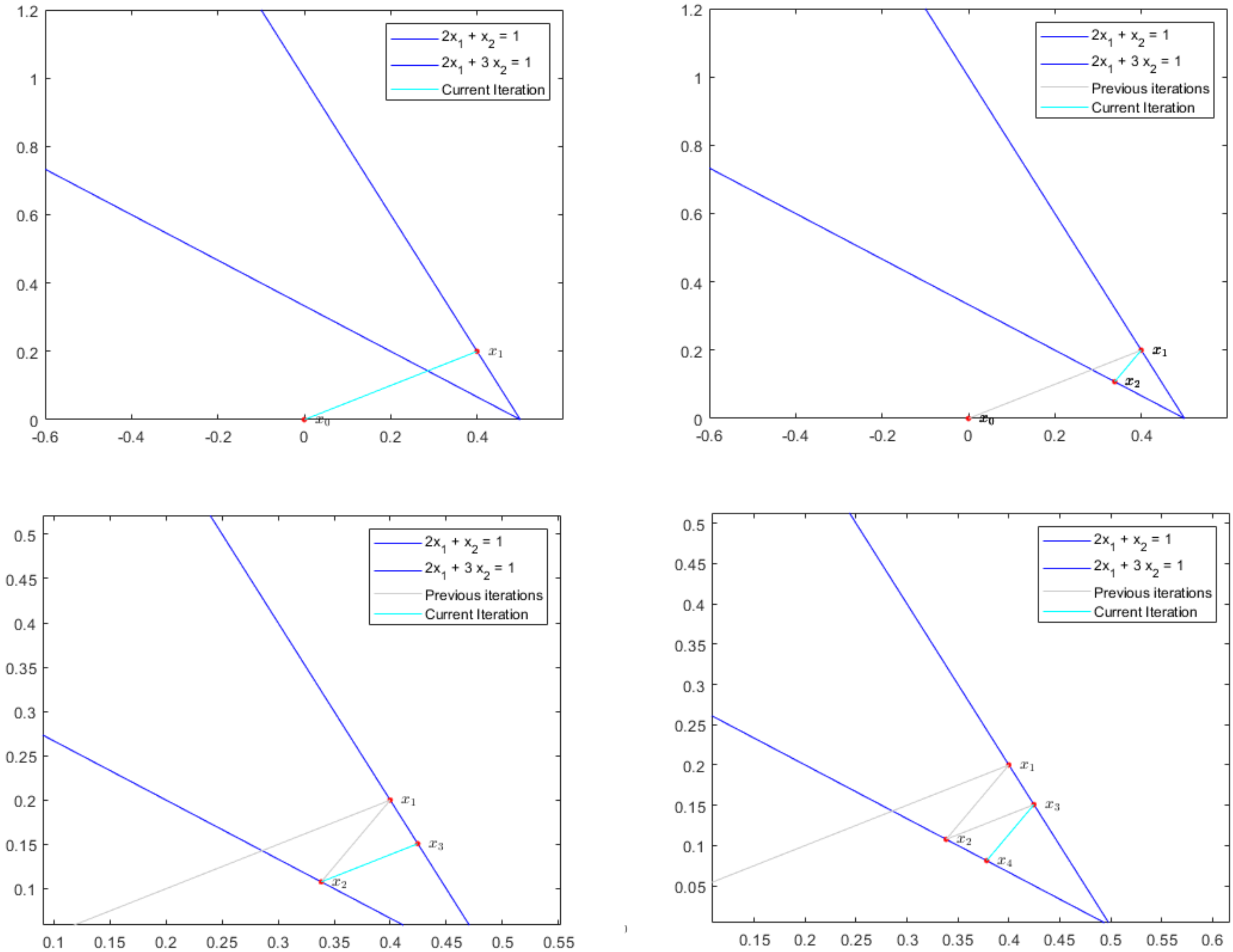

Figure 2: Illustrates the path towards the solution of Algorithm 1 of a system that has quite far from orthogonal solution spaces.

Proof. To show that $\{x_{k+1}\}_0^\infty$ converges towards the limit point $\hat{x} \in \Omega$ we begin by stating

$$\begin{aligned} ||x_{k+1} - \hat{x}||_2^2 &= \left|\left|(x_k - \hat{x}) + \left(\frac{b_{i_k} - \langle a_{i_k}^T, x_k\rangle}{||a_{i_k}^T||_2^2} a_{i_k}^T\right)\right|\right|_2^2 \\ &= ||x_k - \hat{x}||_2^2 + \left|\left|\frac{b_{i_k} - \langle a_{i_k}^T, x_k\rangle}{||a_{i_k}^T||_2^2} a_{i_k}^T\right|\right|_2^2 + 2\left\langle (x_k - \hat{x}), \left(\frac{b_{i_k} - \langle a_{i_k}^T, x_k\rangle}{||a_{i_k}^T||_2^2} a_{i_k}^T\right)\right\rangle. \end{aligned} \tag{9}$$

Removing the scalar from the inner product yields

$$||x_{k+1} - \hat{x}||_2^2 = ||x_k - \hat{x}||_2^2 + \left|\left|\frac{b_{i_k} - \langle a_{i_k}^T, x_k\rangle}{||a_{i_k}^T||_2^2} a_{i_k}^T\right|\right|_2^2 + 2\frac{b_{i_k} - \langle a_{i_k}^T, x_k\rangle}{||a_{i_k}^T||_2^2}\left\langle (x_k - \hat{x}), a_{i_k}^T\right\rangle.$$

Taking a closer look at the scalar $\langle (x_k - \hat{x}), a_{i_k}^T\rangle$ and separating the inner products yields

$$\left\langle (x_k - \hat{x}), a_{i_k}^T\right\rangle = \langle x_k, a_{i_k}^T\rangle - \langle \hat{x}, a_{i_k}^T\rangle.$$

Using the consistency of the system $A\hat{x} = b$ to say that $\langle \hat{x}, a_{i_k}^T\rangle = b_{i_k}$ which gives us

$$\left\langle (x_k - \hat{x}), a_{i_k}^T\right\rangle = \langle x_k, a_{i_k}^T\rangle - b_{i_k} = -\left(b_{i_k} - \langle x_k, a_{i_k}^T\rangle\right).$$

Moving back to the original equation we get

$$\begin{aligned} ||x_{k+1} - \hat{x}||_2^2 &= ||x_k - \hat{x}||_2^2 + \left|\left|\frac{b_{i_k} - \langle a_{i_k}^T, x_k\rangle}{||a_{i_k}^T||_2^2} a_{i_k}^T\right|\right|_2^2 + 2\frac{b_{i_k} - \langle a_{i_k}^T, x_k\rangle}{||a_{i_k}^T||_2^2}\left(-\left(b_{i_k} - \langle x_k, a_{i_k}^T\rangle\right)\right) \\ &= ||x_k - \hat{x}||_2^2 + \left|\left|\frac{b_{i_k} - \langle a_{i_k}^T, x_k\rangle}{||a_{i_k}^T||_2^2} a_{i_k}^T\right|\right|_2^2 - 2\frac{(b_{i_k} - \langle a_{i_k}^T, x_k\rangle)^2}{||a_{i_k}^T||_2^2}. \end{aligned} \tag{10}$$

Expanding and then simplifying the second term yields

$$\left|\left|\frac{b_{i_k} - \langle a_{i_k}^T, x_k\rangle}{||a_{i_k}^T||_2^2} a_{i_k}^T\right|\right|_2^2 = \frac{\left(b_{i_k} - \langle a_{i_k}^T, x_k\rangle\right)^2}{||a_{i_k}^T||_2^4}||a_{i_k}^T||_2^2 = \frac{\left(b_{i_k} - \langle a_{i_k}^T, x_k\rangle\right)^2}{||a_{i_k}^T||_2^2}.$$

This allows us to simplify the original equation to

$$\begin{aligned} ||x_{k+1} - \hat{x}||_2^2 &= ||x_k - \hat{x}||_2^2 + \frac{\left(b_{i_k} - \langle a_{i_k}^T, x_k\rangle\right)^2}{||a_{i_k}^T||_2^2} - 2\frac{\left(b_{i_k} - \langle a_{i_k}^T, x_k\rangle\right)^2}{||a_{i_k}^T||_2^2} \\ &= ||x_k - \hat{x}||_2^2 - \frac{\left(b_{i_k} - \langle a_{i_k}^T, x_k\rangle\right)^2}{||a_{i_k}^T||_2^2}. \end{aligned} \tag{11}$$

Now that we have an expression for $||x_{k+1} - \hat{x}||_2^2$ we want to use definition 17 to show convergence. We have

$$\begin{aligned} \lim_{k\to\infty} \frac{||x_{k+1} - \hat{x}||_2^2}{||x_k - \hat{x}||_2^2} &= \lim_{k\to\infty} \frac{||x_k - \hat{x}||_2^2 - \frac{\left(b_{i_k} - \langle a_{i_k}^T, x_k\rangle\right)^2}{||a_{i_k}^T||_2^2}}{||x_k - \hat{x}||_2^2} \\ &= \lim_{k\to\infty} 1 - \frac{\left(b_{i_k} - \langle a_{i_k}^T, x_k\rangle\right)^2}{||a_{i_k}^T||_2^2||x_k - \hat{x}||_2^2} \end{aligned} \tag{12}$$

then we have $b_{i_k} = \langle a_{i_k}^T, \hat{x}\rangle$ thus we can write it as

$$\begin{aligned} \lim_{k\to\infty} \frac{||x_{k+1} - \hat{x}||_2^2}{||x_k - \hat{x}||_2^2} &= \lim_{k\to\infty} 1 - \frac{\left(\langle a_{i_k}^T, \hat{x}\rangle - \langle a_{i_k}^T, x_k\rangle\right)^2}{||a_{i_k}^T||_2^2||x_k - \hat{x}||_2^2} \\ &= \lim_{k\to\infty} 1 - \frac{\left(\langle a_{i_k}^T, \hat{x} - x_k\rangle\right)^2}{||a_{i_k}^T||_2^2||x_k - \hat{x}||_2^2}. \end{aligned} \tag{13}$$

Since

$$\lim_{k\to\infty} \frac{\left(\langle a_{i_k}^T, \hat{x} - x_k\rangle\right)^2}{||a_{i_k}^T||_2^2||x_k - \hat{x}||_2^2} \geq 0$$

we can bound

$$\lim_{k\to\infty} 1 - \frac{\left(\langle a_{i_k}^T, \hat{x} - x_k\rangle\right)^2}{||a_{i_k}^T||_2^2||x_k - \hat{x}||_2^2} \leq 1.$$

Then by using lemma 8 we can bound

$$\lim_{k\to\infty} \frac{\left(\langle a_{i_k}^T, \hat{x} - x_k\rangle\right)^2}{||a_{i_k}^T||_2^2||x_k - \hat{x}||_2^2} \leq \lim_{k\to\infty} \frac{||a_{i_k}^T||_2^2||\hat{x} - x_k||_2^2}{||a_{i_k}^T||_2^2||x_k - \hat{x}||_2^2} = 1$$

and thus get

$$0 \leq \lim_{k\to\infty} 1 - \frac{\left(\langle a_{i_k}^T, \hat{x} - x_k\rangle\right)^2}{||a_{i_k}^T||_2^2||x_k - \hat{x}||_2^2}.$$

Implying by definition 17 that the Algorithm converges towards some $\hat{x}$.

Defining $\tilde{x} := \arg\min_{x\in\Omega} ||x - x_0||_2^2$ allows us to write the solution $\hat{x} = \lim_{k\to\infty} x_{k+1}$ as $\hat{x} := \tilde{x} + y$. We want to show that $\hat{x} \in \Omega$ if and only if $y \in \ker(A)$, we have

$$||A\hat{x} - b||_2^2 = ||A\tilde{x} + Ay - b||_2^2$$

then since $\tilde{x} \in \Omega$ we get

$$||A\tilde{x} + Ay - b||_2^2 = ||Ay||_2^2$$

and $||Ay||_2^2 = \mathbf{0}$ if and only if $y \in \ker(A)$. Now we want to show that $\hat{x} = \tilde{x}$ by showing that $\hat{x} - x_0$ is orthogonal to y. We have that

$$\begin{aligned}\langle \hat{x} - x_0, y\rangle &= \lim_{k\to\infty} \langle x_{k+1} - x_0, y\rangle \\ &= \lim_{k\to\infty} \left\langle \sum_{j=0}^{k} \frac{b_{i_j} - \langle a_{i_j}^T, x_j\rangle}{||a_{i_j}^T||_2^2} a_{i_j}^T, y\right\rangle \\ &= \lim_{k\to\infty} \sum_{j=0}^{k} \frac{b_{i_j} - \langle a_{i_j}^T, x_j\rangle}{||a_{i_j}^T||_2^2} \left\langle a_{i_j}^T, y\right\rangle\end{aligned} \tag{14}$$

however $\langle a_{i_j}^T, y\rangle = 0$ since $y \in \ker(A)$ and thus we get

$$\langle \hat{x} - x_0, y\rangle = 0.$$

□

Remark 3. *Furthermore any consistent system with some* $||a_i||_2^2 = 0$ *can be transformed into a system that is solvable by the Kaczmarz method. Consider we have a row* $||a_i||_2^2 = 0$ *then if the system is consistent we have* $b_i = 0$*, which means that the i'th row solution space is the entire set* $x \in \mathbf{R}^n$*. Thus we can omit the row without loss of generality and solve a system given by* $A \in \mathbf{R}^{m-1\times n}$ *and* $b \in \mathbf{R}^{m-1}$ *using Algorithm 1.*

Remark 4. *It is also important to notice that the order of row choice does not necessarily matter, as long as* $\{x_{k+1}\}_p^\infty$ *passes trough all rows for any p. Later in the thesis we will present other row choice methods.*

So the Kaczmarz method can solve all consistent systems, however if we consider an inconsistent system the Kaczmarz method will not converge to any solution $\hat{x}$. We get the solution to a new equation in each iteration and for inconsistent systems the intersection of all equations is empty. This is illustrated in figure 3 where we have an overdetermined inconsistent system given by

$$A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ -1 & 1 \end{bmatrix},$$

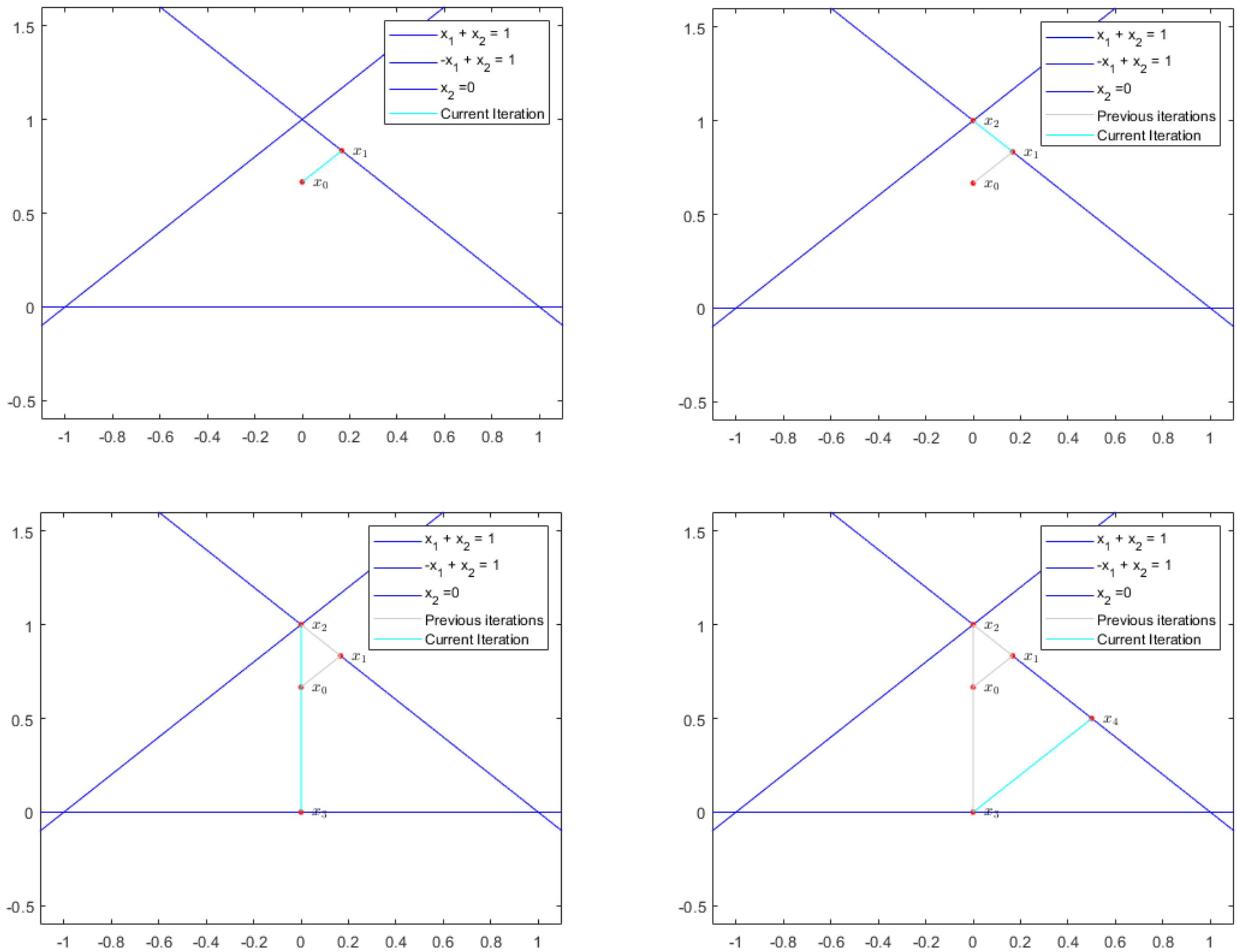

Figure 3: Shows a over determined inconsistent system and Algorithm 1 inability to reach the least square solution. Interestingly enough we choose x_0 to be the least squares solution of the system.

$$b = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$$

with an initial point $x_0 = \arg\min_x ||Ax - b||_2^2$.
We are also interested in the rate of convergence of the original Kaczmarz method. Galantai [2] provides a proof of the rate of convergence for the original Kaczmarz method, however the rate of convergence in the proof is limited to non singular A.

Lemma 9 (Meany [2]). *Let $x_i \in \mathbf{R}^n$ and $||x_i||_2 = 1$ for $i = 1, ..., k$, $k \leq n$, $X_k = [x_1, ..., x_k]$ and $Q_k = \prod_{j=1}^{k} I - x_j x_j^T$, then*

$$||Q_k||_2 \leq \sqrt{1 - \det(X_k^T X_k)}.$$

Proof. See [2] page 72 theorem 2.77 in book □

Lemma 10 (Arithmetic-geometric mean [5]). *For any non-negative real numbers C_i we have*

$$\left(\prod_{i=1}^{m} C_i\right)^{1/m} \leq \frac{1}{m}\sum_{i=1}^{m} C_i.$$

Proof. See [[5] page 259-260 theorem 10.8] □

Lemma 11. *If we have a diagonalizable matrix $A \in \mathbf{R}^{m\times m}$ then the trace $tr(A) = tr(D)$ and determinant $det(A) = det(D)$ where $A = VDV^{-1}$.*

Theorem 6 (Original Kaczmarz method; rate of convergence [2]). *If we have square consistent system with full rank $\Omega = \{x \in \mathbf{R}^m; Ax = b\}$ where $A \in \mathbf{R}^{m\times m}$ and $b \in \mathbf{R}^m$ then we can bound the expected rate of convergence of the sequence $\{x_{k+1}\}_{k=0}^{\infty}$ given by Algorithm 1 to $\hat{x}$. Defining Y as*

$$Y_m := [y_1, y_2, \ldots, y_m] := \left[\frac{a_1^T}{||a_1^T||_2}, \frac{a_2^T}{||a_2^T||_2}, \ldots, \frac{a_m^T}{||a_m^T||_2}\right],$$

the rate of convergence every $k + 1 = 0$ mod m can be given by

$$||x_{k+1} - \hat{x}||_2^2 \leq \left(1 - \det(Y_m^T Y_m)\right)^{(k+1)/m} ||(x_0 - \hat{x})||_2^2.$$

Proof. Using the consistency to represent $\hat{x}$ as

$$\hat{x} = \hat{x} + \frac{b_{i_k} - \langle a_{i_k}^T, \hat{x}\rangle}{||a_{i_k}^T||_2^2} a_{i_k}^T.$$

Expressing the difference between x_{k+1} and $\hat{x}$ then moving the constant over

$$x_{k+1} - \hat{x} = (x_k - \hat{x}) - \frac{\langle a_{i_k}^T, x_k - \hat{x}\rangle}{||a_{i_k}^T||_2^2} a_{i_k}^T.$$

We are looking to get an expression where we can use lemma 9 and thus write it as a matrix vector multiplication

$$x_{k+1} - \hat{x} = (I - \frac{a_{i_k}^T a_{i_k}}{||a_{i_k}^T||_2^2})(x_k - \hat{x}).$$

Repeating this process gives us

$$\begin{aligned} x_{k+1} - \hat{x} &= \prod_{j=0}^{m-1}\left(I - \frac{a_{i_j}^T a_{i_j}}{||a_{i_j}^T||_2^2}\right)(x_{k+1-m} - \hat{x}) \\ &= \left(\prod_{j=0}^{m-1}(I - \frac{a_{i_j}^T a_{i_j}}{||a_{i_j}^T||_2^2})\right)^{(k+1)/m} (x_0 - \hat{x}). \end{aligned} \tag{15}$$

Using the definition 13 for the matrix norm we get

$$||x_{k+1} - \hat{x}||_2^2 = \left|\left|\left(\prod_{i=0}^{m-1}(I - \frac{a_i^T a_i}{||a_i^T||_2^2})\right)^{(k+1)/m} (x_0 - \hat{x})\right|\right|_2^2 \leq \left(\left|\left|\prod_{i=0}^{m-1}(I - \frac{a_i^T a_i}{||a_i^T||_2^2})\right|\right|_2^2\right)^{(k+1)/m} ||(x_0 - \hat{x})||_2^2.$$

We now see that

$$\frac{a_i^T a_i}{||a_i^T||_2^2} = \frac{a_i^T}{||a_i^T||_2} \frac{a_i}{||a_i^T||_2} = y_i y_i^T$$

from the definition earlier. We now use the assumption that A is non singular and apply theorem 9. To make it clearer we choose to define $Q_m := \prod_{i=0}^{m-1}(I - y_i y_i^T)$

$$||x_{k+1} - \hat{x}||_2^2 = \left(||Q_m||_2^2\right)^{(k+1)/m} ||(x_0 - \hat{x})||_2^2 \leq \left(1 - \det(Y_m^T Y_m)\right)^{(k+1)/m} ||(x_0 - \hat{x})||_2^2.$$

It remains to show that $0 \leq (1 - \det(Y_m^T Y_m)) \leq 1$ we show that $0 < \det(Y_m^T Y_m) \leq 1$. First to show $0 < \det(Y_m^T Y_m)$ it is enough to show that $Y_m^T Y_m$ is positive definite. By definition 18 we need to show

$$c^T (Y_m^T Y_m) c > 0$$

for any $c \in \mathbf{R}^m \setminus \mathbf{0}$. Since A has full rank it means that Y_m has full rank thus we have that $Y_m c \in \mathbf{R}^m \setminus \mathbf{0}$ which implies that

$$||Y_m c||_2^2 > 0.$$

Now it remains to show that $\det(Y_m^T Y_m) \leq 1$, all the values in diagonal of $Y_m^T Y_m$ are

$$\frac{||a_i||_2^2}{||a_i||_2^2} = 1$$

and thus the trace

$$\sum_{i=1}^{m} \frac{||a_i||_2^2}{||a_i||_2^2} = m.$$

Considering the eigenvalues λ_i of $Y_m^T Y_m$ we have

$$m = \sum_{i=1}^{m} \lambda_i.$$

Using lemma 10 we get

$$\left(\prod_{i=1}^{m} \lambda_i\right)^{1/m} \leq \frac{1}{m} \left(\sum_{i=1}^{m} \lambda_i\right)$$

however we have $m = \sum_{i=1}^{m} \lambda_i$ and thus get

$$\prod_{i=1}^{m} \lambda_i \leq 1$$

and we have that $\det(Y_m^T Y_m) = \prod_{i=1}^{m} \lambda_i$ yielding $\det(Y_m^T Y_m) \leq 1$. $\square$

3.2 Randomized Kaczmarz method

The randomized Kaczmarz method [12] uses the same projection scheme as the original Kaczmarz method however selects rows randomly more precisely with a probability given by $||a_i||_2^2/||A||_F^2$.
The randomized Kaczmarz method was the first variation of the Kaczmarz method with an expected

Algorithm 2 Randomized Kaczmarz method [12]

1: **procedure** (A, b, x_0, N, ϵ) ▷ $A \in \mathbf{R}^{m\times n}$, $b \in \mathbf{R}^m$, $x_0 \in \mathbf{R}^n$, $N \in \mathbf{N}$, $\epsilon \in \mathbf{R}$
2: **repeat**
3: Choose row i_k with a probability given by $||a_i||_2^2/||A||_F^2$
4:
$$x_{k+1} = x_k + \frac{b_{i_k} - \langle a_{i_k}^T, x_k\rangle}{||a_{i_k}^T||_2^2} a_i^T$$
5: **if** $k = 0 \bmod m$ **and** $||Ax_{k+1} - b||_2^2 \leq \epsilon$ **then**
6: **return** x_{k+1}
7: **until** $k + 1 > Nm$
8: **return** x_{k+1}

rate of convergence that could be bound after each iteration using the Frobenius induced condition number. This made it clearer to compare the Kaczmarz method to other solvers for linear system of equations. Strohmer and Vershynin compares the randomized Kaczmarz method to the CGLS (conjugate gradient least squares) method in their article [12]. It was proven that it required less computational complexity than the CGLS method on a overdetermined, consistent and normally distributed system with $m > 3n$.
Needle purposed a rate of convergence 2010 [8] that describes the rate of convergence up until we get close to the solution. Thus showing how the Kaczmarz method handles noisy system of equations.

Definition 20 (Expectation)**.** *Given a distribution function $f(x)$ we have that the expectation $\mathbf{E}$ of a random variable X is given by*

$$\mathbf{E}[X] = \sum_{x=-\infty}^{\infty} x f(x)$$

or

$$\mathbf{E}[X] = \int_{-\infty}^{\infty} x f(x) dx$$

where $f(x)$ is the probability of each outcome.

Definition 21 (Noise)**.** *Given a system of equations $A \in \mathbf{R}^{m\times n}$ and $b \in \mathbf{R}^m$. The noise is then set to be smallest $r \in \mathbf{R}^m$ such that the system of equations becomes consistent $Ax = b + r$.*

Theorem 7 (Randomized Kaczmarz method; rate of convergence [12])**.** *If we have a consistent system $A \in \mathbf{R}^{m\times n}$ and $b \in \mathbf{R}^m$ then we can bound the expected rate of convergence to $\hat{x}$ of Algorithm 2 by a rate given by $0 \leq (1 - \kappa(A)_F^{-2}) < 1$ i.e.*

$$\mathbf{E}\Big[||x_{k+1} - \hat{x}||_2^2\Big] \leq (1 - \kappa(A)_F^{-2})^{k+1} ||x_0 - \hat{x}||_2^2.$$

Theorem 8 (Randomized Kaczmarz method rate of convergence [8] for noisy systems)**.** *If we have a noisy system of equations and $A \in \mathbf{R}^{m\times n}$, $x \in \mathbf{R}^n$, $b \in \mathbf{R}^m$ and $r \in \mathbf{R}^m$. Then we can bound the expected rate of convergence of Algorithm 2 to the least squares solution $\hat{x}$ by*

$$\mathbf{E}\Big[||x_{k+1} - \hat{x}||_2^2\Big] \leq (1 - \kappa(A)_F^{-2})^{k+1} ||x_0 - \hat{x}||_2^2 + \rho^2 \kappa(A)_F.$$

Where ρ is defined as

$$\rho := \max_i \frac{|r_i|}{||a_{i_k}||_2}.$$

3.3 Extended randomized Kaczmarz method

Furthering our interest of the Kaczmarz method applied to inconsistent systems, we desire to find the least squares solution $\arg\min_x ||Ax - b||_2$ by first transforming the problem into a consistent problem. Zouzias proposed 2013 [14] to instead solve a consistent problem $Ax = b - z$, where z is approximated using the randomized Kaczmarz method to solve $A^T z = \mathbf{0}$ setting $z^{(0)} = b$.

Definition 22. *For a matrix $A \in \mathbf{R}^{m\times n}$, $a_{(j)} \in \mathbf{R}^m$ is defined by the j'th column of A i.e.*

$$A = [a_{(1)}, a_{(2)}, ..., a_{(n-1)}, a_{(n)}].$$

Algorithm 3 Randomized Kaczmarz transform for least squares problems [14]

procedure (A, b, M, ϵ) ▷ $A \in \mathbf{R}^{m\times n}$, $b \in \mathbf{R}^m$, $M \in \mathbf{N}, \epsilon \in \mathbf{R}$
  $z^{(0)} = b$
  repeat
    Choose column j_k with a probability given by $||a_{(j)}||_2^2/||A||_F^2$.

$$z^{(k+1)} = z^{(k)} - \frac{\langle a_{(j_k)}, z^{(k)}\rangle}{||a_{(j_k)}||_2^2} a_{(j_k)}$$

    if $k = 0 \bmod n$ **and** $||A^T z^{(k+1)}||_2^2 \leq \epsilon$ **then**
      return $z^{(k+1)}$
  until $k + 1 > Mn$
return $z^{(k+1)}$

Statement 3. *Given any inconsistent system $A \in \mathbf{R}^{m\times n}$, $b \in \mathbf{R}^m$, $x \in \mathbf{R}^n$ and $\min_{x\in\mathbf{R}^n} ||Ax - b||_2^2 \neq 0$ we can transform it into a consistent system $\min_{x\in\mathbf{R}^n} ||Ax - (b - z^{(k+1)})||_2^2 = 0$ letting $z^{(k+1)} \in \mathbf{R}^m$ be defined by Algorithm 3.*

Proof. To show there exists an $x \in \mathbf{R}^n$ such that $Ax = b - z^{(k+1)}$ we begin by writing Ax as a sum of column vectors

$$Ax = \sum_{j=1}^{m} x_j a_{(j)}$$

defining x_j as the j'th element of x. Now we investigate $b - z^{(k+1)}$, first we rewrite $z^{(k+1)}$ using the definition for $z^{(k+1)}$

$$z^{(k+1)} = b - \sum_{p=0}^{k} \frac{\langle a_{(j_p)}, z^{(p)}\rangle}{||a_{(j_p)}||_2^2} a_{(j_p)}$$

defining $a_{(j_p)}$ as the randomly chosen column on the p'th iteration of Algorithm 3. It remains to show that there exists an x such that

$$\sum_{j=0}^{m} x_j a_{(j)} = \sum_{p=0}^{k} \frac{\langle a_{(j_p)}, z^{(p)}\rangle}{||a_{(j_p)}||_2^2} a_{(j_p)}$$

holds. We group the terms on the right hand side such that all randomly selected $a_{(j_p)} = a_{(j)}$ are grouped together,

$$\sum_{p=0}^{k} \frac{\langle a_{(j_p)}, z^{(p)}\rangle}{||a_{(j_p)}||_2^2} a_{(j_p)} = \sum_{j=1}^{m} \left(\sum \frac{\langle a_{(j)}, z^{(p_j)}\rangle}{||a_{(j)}||_2^2} \right) a_{(j)}$$

where $\sum \frac{\langle a_{(j)}, z^{(p_j)}\rangle}{||a_{(j)}||_2^2}$ is the sum of all the terms when $a_{(j_p)} = a_{(j)}$. We have

$$\sum_{j=1}^{m} x_j a_{(j)} = \sum_{j=0}^{m} \left(\sum \frac{\langle a_{(j)}, z^{(p_j)}\rangle}{||a_{(j)}||_2^2} \right) a_{(j)}$$

and we see that there exists at least one solution x letting $x_j = \sum \frac{\langle a_{(j)}, z^{(p_j)}\rangle}{||a_{(j)}||_2^2}$. □

Statement 4. *Given a system of equations $A \in \mathbf{R}^{m\times n}$, $x \in \mathbf{R}^n$ and $b \in \mathbf{R}^m$, let $\hat{z}$ be any point in the set $\omega = \{z \in \mathbf{R}^m; A^T z = \mathbf{0}\}$, then $\arg\min_{x\in\mathbf{R}^n} ||Ax - b||_2^2 = \arg\min_{x\in\mathbf{R}^n} ||Ax - (b - \hat{z})||_2^2$.*

Proof. We have that

$$\hat{x} \in \arg\min_{x \in \mathbf{R}^n} ||Ax - b||_2^2 \iff A^T A\hat{x} = A^T b$$

however since $A^T \hat{z} = \mathbf{0}$ we get

$$A^T A\hat{x} = A^T b \iff A^T A\hat{x} = A^T(b - \hat{z}) \iff \hat{x} \in \arg\min_{x \in \mathbf{R}^n} ||Ax - (b - \hat{z})||_2^2$$

□

By statement 3 and statement 4 we know that Algorithm 3 creates a sequence of $\{z^{(k+1)}\}_0^\infty$ that transforms the problem into a consistent problem and converges towards a $\hat{z}$ that has the same least squares solution. Then by applying the randomized Kaczmarz method we get the extended Kaczmarz method which by theorem 8 converges towards the solution.

Algorithm 4 Extended randomized Kaczmarz method [14]

procedure (A, b, x_0, N, ϵ) ▷ $A \in \mathbf{R}^{m \times n}$, $b \in \mathbf{R}^m$, $x_0 \in \mathbf{R}^n$, $N \in \mathbf{N}$, $\epsilon \in \mathbf{R}$
  $z^{(0)} = b$
  repeat
    Choose column j_k with a probability given by $||a_{(j)}||_2^2/||A||_F^2$.

$$z^{(k+1)} = z^{(k)} - \frac{\langle a_{(j_k)}, z^{(k)} \rangle}{||a_{(j_k)}||_2^2} a_{(j_k)}$$

    Choose row i_k with a probability given by $||a_i||_2^2/||A||_F^2$.

$$x_{k+1} = x_k + \frac{(b_{i_k} - z_{i_k}^{(k+1)}) - \langle a_{i_k}^T, x_k \rangle}{||a_{i_k}^T||_2^2} a_{i_k}^T$$

    if $k = 0 \mod \min(m, n)$ **and** $||A^T z^{(k+1)}||_2^2 \leq \epsilon$ **and** $||Ax_{k+1} - (b - z^{k+1})||_2^2 \leq \epsilon$ **then**
      return x_{k+1}
  until $k + 1 > N \min(m, n)$
return x_{k+1}

Theorem 9 (Extended randomized Kaczmarz method [14])**.** *If we have a system of equations $A \in \mathbf{R}^{m \times n}$ and $b \in \mathbf{R}^m$ we can bound the expected rate of convergence to $\hat{x}$ of Algorithm 4 by*

$$\mathbf{E}\Big[||x_{k+1} - \hat{x}||_2^2\Big] \leq (1 - \kappa(A)_F^{-2})^{(k+1)/2}(1 + 2\kappa(A)^2)||x_0 - \hat{x}||_2^2.$$

4 Block Kaczmarz method

4.1 Block Kaczmarz method

The idea behind the block Kaczmarz is to apply several row conditions at each iteration step to approach the solution quicker. The block Kaczmarz method divide $A \in \mathbf{R}^{n\times m}$ into blocks $A_\tau \in \mathbf{R}^{t_\tau \times n}$ where before starting the iterations. Then selects block A_τ in each iteration with uniform distribution.

Algorithm 5 Block Kaczmarz method [9]

procedure (A, b, x_0, N, ϵ) ▷ $A \in \mathbf{R}^{m\times n}$, $b \in \mathbf{R}^m$, $x_0 \in \mathbf{R}^n$, $N \in \mathbf{N}$, $\epsilon \in \mathbf{R}$
    Construct the blocks $A_\tau \in \mathbf{R}^{t_\tau \times n}$ and the corresponding $b_\tau \in \mathbf{R}^{t_\tau}$
    Let t be the amount of blocks A_τ created
    repeat
        Choose A_{τ_k} with uniform distribution

$$x_{k+1} = x_k + A_{\tau_k}^\dagger (b_{\tau_k} - A_{\tau_k} x_k)$$

        if $k = 0 \bmod t$ **and** $||Ax_{k+1} - b||_2^2 \leq \epsilon$ **then**
            return x_{k+1}
    until $k + 1 > Nt$
return x_{k+1}

Theorem 10 ([6]). *The system $Ax = b$ is consistent if and only if $AA^\dagger b = b$.*

Proof. See [6] for proof. □

Statement 5. *If x_{k+1} is defined by Algorithm 5*

$$x_{k+1} = x_k + A_{\tau_k}^\dagger (b_{\tau_k} - A_{\tau_k} x_k)$$

then x_{k+1} solves the previous equation $A_{\tau_k} x_{k+1} = b_{\tau_k}$.

Proof. By expanding $A_{\tau_k} x_{k+1}$ we get,

$$A_{\tau_k} x_{k+1} = A_{\tau_k} \left(x_k + A_{\tau_k}^\dagger (b_{\tau_k} - A_{\tau_k} x_k) \right) = A_{\tau_k} x_k + A_{\tau_k} A_{\tau_k}^\dagger b_{\tau_k} - A_{\tau_k} A_{\tau_k}^\dagger A_{\tau_k} x_k$$

then by definition 14 we have $A_{\tau_k} A_{\tau_k}^\dagger A_{\tau_k} = A_{\tau_k}$, yielding us

$$A_{\tau_k} x_{k+1} = A_{\tau_k} A_{\tau_k}^\dagger b_{\tau_k}.$$

We use the assumption that the system is consistent, which implies that the equations $A_{\tau_k} x = b_{\tau_k}$ are consistent as well. It allows us to use theorem 10 resulting in

$$A_{\tau_k} x_{k+1} = A_{\tau_k} A_{\tau_k}^\dagger b_{\tau_k} = b_{\tau_k}.$$

□

Statement 6. *If x_{k+1} is defined by Algorithm 5 and ω is the set $\omega = \{x \in \mathbf{R}^n; A_\tau x = b_\tau\}$ then x_{k+1} is the orthogonal projection of x_k onto ω.*

Proof. From statement 5 we have that $x_{k+1} \in \omega$, now it remains to show that it is an orthogonal projection. Rearranging the equation we get

$$x_k + A_{\tau_k}^\dagger (b_{\tau_k} A_{\tau_k} x_k) = x_k - A_{\tau_k}^\dagger A_{\tau_k} x_k + A_{\tau_k}^\dagger b_{\tau_k}.$$

From lemma 7 we see that $A_{\tau_k}^\dagger b_{\tau_k}$ is the minimum norm transform to our solution space. It remains to show that $x_k - A_{\tau_k}^\dagger A_{\tau_k} x_k$ is an orthogonal projection. By defining

$$P := I - A_{\tau_k}^\dagger A_{\tau_k}$$

we have

$$x_k - A_{\tau_k}^\dagger A_{\tau_k} x_k = P x_k.$$

By showing that $P^2 = P$ and $P^T = P$ we prove that it is an orthogonal projection. By expanding P^2 we get

$$\begin{aligned} P^2 &= (I - A^{\dagger}_{\tau_k} A_{\tau_k})(I - A^{\dagger}_{\tau_k} A_{\tau_k}) \\ &= I - 2A^{\dagger}_{\tau_k} A_{\tau_k} + A^{\dagger}_{\tau_k} A_{\tau_k} A^{\dagger}_{\tau_k} A_{\tau_k}, \end{aligned} \tag{16}$$

then from definition 14 we get

$$P^2 = I - 2A^{\dagger}_{\tau_k} A_{\tau_k} + A^{\dagger}_{\tau_k} A_{\tau_k} = P.$$

We now expand P^T and get

$$\begin{aligned} P^T &= (I - A^{\dagger}_{\tau_k} A_{\tau_k})^T \\ &= I - (A^{\dagger}_{\tau_k} A_{\tau_k})^T, \end{aligned} \tag{17}$$

however by definition 14 we have

$$P^T = I - A^{\dagger}_{\tau_k} A_{\tau_k} = P.$$

□

Remark 5. *Since each iteration of Algorithm 5 orthogonally projects onto the solution space given by $\omega = \{x \in \mathbf{R}^n; A_\tau x = b_\tau\}$, we have that every iteration step is equal to applying Algorithm 1 on ω. Which by remark 4 implies that Algorithm 5 converges towards the same solution as Algorithm 1.*

Definition 23 (Paving parameters [9, 10])**.** *To describe the rate of convergence we need to introduce the paving parameters. Let $T = [A_{\tau_1}, A_{\tau_2}, \dots, A_{\tau_t}]$ then define α and β such that*

$$\alpha \leq \sigma_{\min}(A_\tau)$$

$$\beta \geq \sigma_{\max}(A_\tau)$$

holds for all $A_\tau \in T$.

The rate of convergence of Algorithm 5 for a noisy system $Ax = b + r$ was described 2014 by Needle [9].

Theorem 11 (Block Kaczmarz method rate of convergence for noisy systems [9])**.** *Let $A \in \mathbf{R}^{m \times n}$, $b \in \mathbf{R}^m$ and $r \in \mathbf{R}^m$ define the noisy linear system given by $Ax = b + r$. Then Algorithm 5 converges towards the solution $\hat{x}$ with an expected rate of convergence given by*

$$\mathbf{E}\Big[||x_{k+1} - \hat{x}||_2^2\Big] \leq \left(1 - \frac{\sigma^2_{\min}(A)}{\beta t}\right)^{k+1} ||x_0 - \hat{x}||_2^2 + \frac{\beta}{\alpha} \frac{||r||_2^2}{\sigma^2_{\min}(A)}.$$

4.2 Extended block Kaczmarz method

Now we are interested in using the block Kaczmarz method to solve inconsistent systems similarly to how we did with the randomized Kaczmarz method.

Algorithm 6 Block Kaczmarz method tranform for least squares problems

1: **procedure** (A^T, b, N, ϵ) ▷ $A^T \in \mathbf{R}^{n \times m}$, $b \in \mathbf{R}^m$, $N \in \mathbf{N}$, $\epsilon \in \mathbf{R}$
2: Construct the blocks $A^T_\xi \in \mathbf{R}^{c_\xi \times m}$
3: Let c be the amount of blocks created A^T_ξ
4: $z^{(0)} = b$
5: **repeat**
6: Choose $A^T_{\xi_k}$ with uniform distribution
7:
$$z^{(k+1)} = z^{(k)} - (A^T_{\xi_k})^\dagger A^T_{\xi_k} z^{(k)}$$
8: **if** $k = 0 \bmod c$ **and** $||A^T z^{(k+1)} - b||_2^2 \leq \epsilon$ **then**
9: **return** $z^{(k+1)}$
10: **until** $k + 1 > Nc$
11: **return** $z^{(k+1)}$

Statement 7. *We can transform any inconsistent system* $\min_{x\in R^n} ||Ax - b||_2^2 \neq 0$ *into a consistent system* $\min_{x\in \mathbf{R}^n} ||Ax - (b - z^{(k+1)})||_2^2 = 0$ *letting* $z^{(k+1)}$ *be defined by Algorithm 6.*

Proof. This follows directly from statement 3 and remark 5. □

Statement 8. *Let* $\hat{z}$ *be any point in the set* $\omega = \{z \in \mathbf{R}^m; A^T z = \mathbf{0}\}$*, then* $\arg\min_{x\in\mathbf{R}^n} ||Ax-b||_2^2 = \arg\min_{x\in\mathbf{R}^n} ||Ax - (b + \hat{z})||_2^2$.

Proof. This follows directly from statement 4 and remark 5 □

Algorithm 7 Extended block Kaczmarz method [10]

procedure (A, b, x_0, N, ϵ) ▷ $A \in \mathbf{R}^{m\times n}$, $b \in \mathbf{R}^m$, $x_0 \in \mathbf{R}^n$, $N \in \mathbf{N}, \epsilon \in \mathbf{R}$
  Construct the blocks $A_\xi^T \in \mathbf{R}^{c_\xi \times m}$
  Construct the blocks $A_\tau \in \mathbf{R}^{t_\tau \times n}$ and the corresponding $b_\tau \in \mathbf{R}^{t_\tau}$
  Let c be the amount of blocks created A_ξ^T
  Let t be the amount of blocks created A_τ
  $z^{(0)} = b$
  repeat
    Choose $A_{\xi_k}^T$ with uniform distribution

$$z^{(k+1)} = z^{(k)} - (A_{\xi_k}^T)^\dagger A_{\xi_k}^T z^{(k)}$$

    Choose A_{τ_k} with uniform distribution
    Construct $z_{\tau_k}^{(k+1)}$ using the corresponding rows to the block A_τ

$$x_{k+1} = x_k + A_{\tau_k}^\dagger \left((b_{\tau_k} - z_{\tau_k}^{(k+1)}) - A_{\tau_k} x_k \right)$$

    if $k = 0 \bmod \min(c, t)$ **and** $||A^T z^{(k+1)}||_2^2 \leq \epsilon$ **and** $||Ax_{k+1} - (b - z^{k+1})||_2^2 \leq \epsilon$ **then**
      return x_{k+1}
  until $k + 1 > N \min(c, t)$
return $z^{(k+1)}$

Theorem 12 (Extended block Kaczmarz method rate of convergence [10]). *Let $A \in \mathbf{R}^{m\times n}$ and $b \in \mathbf{R}^m$, define the least squares problem* $\arg\min_{x\in\mathbf{R}^n} ||Ax - b||_2^2$*. Then Algorithm 7 converges towards the solution $\hat{x}$ with an expected rate of convergence given by*

$$\mathbf{E}\Big[||x_{k+1} - \hat{x}||_2^2\Big] \leq \left(1 - \frac{\sigma_{\min}^2(A)}{\beta t}\right)^{k+1} ||x_0 - \hat{x}||_2^2 + \left(\left(1 - \frac{\sigma_{\min}^2(A)}{\beta t}\right)^{(k+1)/2} + \left(1 - \frac{\sigma_{\min}^2(A)}{\hat{\beta}\hat{t}}\right)^{(k+1)/2}\right) \frac{\beta d||b - \hat{z}||_2^2}{\alpha\sigma_{\min}^2(A)} \tag{18}$$

where α and β are the paving parameters of the blocks A_τ and $\hat{\beta}$ is the paving parameter of the blocks A_ξ^T.

4.3 Greedy block Kaczmarz method

The idea behind the greedy block Kaczmarz method is to create the block A_τ in each iteration such that the distance between x_{k+1} and x_k is maximized, meaning that x_{k+1} is as close to orthogonal as it can to x_k. The greedy Kaczmarz method presented by Yu-Qi Niu [11] approximates the block that gives the maximum distance in each iteration by finding the maximum distance row

$$\max_{i\in\{1,2,\ldots,m\}} \frac{|b_i - \langle a_i^T, x_k\rangle|^2}{||a_i^T||_2}$$

and then let the permutation τ be defined as the combination of all rows satisfying the condition

$$\frac{|b_i - \langle a_i^T, x_k\rangle|^2}{||a_i^T||_2} \geq \eta \left(\max_{i\in\{1,2,\ldots,m\}} \frac{|b_i - \langle a_i^T, x_k\rangle|^2}{||a_i^T||_2}\right)$$

where η is a relaxation parameter.

Algorithm 8 Greedy block Kaczmarz method [11]

---

procedure $(A, b, x_0, N, \eta, \epsilon)$ $\quad\triangleright A \in \mathbf{R}^{m\times n}$, $b \in \mathbf{R}^m$, $x_0 \in \mathbf{R}^n$, $N \in \mathbf{N}$, $0 < \eta \leq 1$, $\epsilon \in \mathbf{R}$

  repeat

    Calculate

$$\epsilon = \left(\max_{i\in\{1,2,\ldots,m\}} \frac{|b_i - \langle a_i^T, x_k\rangle|^2}{||a_i^T||_2}\right)$$

    Construct the block A_{τ_k} and b_{τ_k} as the matrix with all rows
    i that satisfies the following condition

$$\frac{|b_i - \langle a_i^T, x_k\rangle|^2}{||a_i^T||_2} \geq \eta\epsilon$$

$$x_{k+1} = x_k + A_{\tau_k}^\dagger (b_{\tau_k} - A_{\tau_k} x_k)$$

    if $Ax_{k+1} \leq \epsilon$ **then**
      return x_{k+1}
  until $k + 1 > N$
return x_{k+1}

Theorem 13 (Greedy block Kaczmarz method [11]). *Let $A \in \mathbf{R}^{m\times n}$ and $b \in \mathbf{R}^m$ define a consistent system, then we can bound the rate of convergence of Algorithm 8 to the solution $\hat{x}$ by*

$$||x_{k+1} - \hat{x}||_2^2 \leq \left(1 - \delta_k(\eta)\kappa(A)_F^{-2}\right)^{k+1} ||x_0 - \hat{x}||_2^2,$$

defining

$$\delta_k(\eta) := \eta \frac{||A||_F^2}{||A||_F^2 - ||A_{\tau_{k-1}}||_F^2} \frac{||A_{\tau_k}||_F^2}{\sigma_{\max}^2(A_{\tau_k})}$$

for all $k \geq 1$ and defining δ_0 as

$$\delta_0 := \eta \frac{||A_{\tau_k}||_F^2}{\sigma_{\max}^2(A_{\tau_k})}.$$

5 Implementation

The tests where all implemented on Matlab and the computer used for the tests runs on windows 10 with a intel core i7-7700hq and 2.80 GHz (8 CPUs).

5.1 Tests on consistent problems

5.1.1 Consistent tests

We want to compare the different variations of the Kaczmarz method for consistent problems Algorithm 1, 2, 5, 8. We include two Algorithms for consistent problems to the tests.

Algorithm 9 Uniform randomized Kaczmarz method [9]

procedure (A, b, x_0, N, ϵ) ▷ $A \in \mathbf{R}^{m\times n}$, $b \in \mathbf{R}^m$, $x_0 \in \mathbf{R}^n$, $N \in \mathbf{N}$, $\epsilon \in \mathbf{R}$
  repeat
    Choose row i_k randomly with a uniform distribution

$$x_{k+1} = x_k + \frac{b_{i_k} - \langle a_{i_k}^T, x_k\rangle}{||a_{i_k}^T||_2^2} a_{i_k}^T$$

    if $k = 0 \mod m$ **and** $||Ax_{k+1} - b||_2^2 \leq \epsilon$ **then**
      return x_{k+1}
  until $k + 1 > Nm$
return x_{k+1}

Algorithm 10 Deterministic block Kaczmarz method

procedure (A, b, x_0, N, ϵ) ▷ $A \in \mathbf{R}^{m\times n}$, $b \in \mathbf{R}^m$, $x_0 \in \mathbf{R}^n$, $N \in \mathbf{N}$, $\epsilon \in \mathbf{R}$
  Construct the blocks A_τ and the corresponding b_τ
  Let t be the amount of blocks created A_τ
  repeat
    Choose $\tau_k = k + 1 \mod t$

$$x_{k+1} = x_k + A_{\tau_k}^\dagger (b_{\tau_k} - A_{\tau_k} x_k)$$

    if $k = 0 \mod t$ **and** $||Ax_{k+1} - b||_2^2 \leq \epsilon$ **then**
      return x_{k+1}
  until $k + 1 > Nt$
return x_{k+1}

Algorithm 9 is the randomized Kaczmarz method presented in [9]. While Algorithm 10 is Algorithm 5 with the deterministic block selection processes found in Algorithm 1.

5.1.2 Underdetermined with $\sigma_{\max}/\sigma_{\min} = 10^5$ *figure. 4*

To construct A we first constructed a matrix $\tilde{A} \in R^{300\times 5000}$ with uniformly distributed values $[0, 1]$. Then we took the singular value decomposition and changed the singular values such that $\frac{1}{100000} \leq \sigma \leq 1$. Then constructed a solution $\tilde{x} \in R^{5000\times 1}$ with uniformly distributed values $[0, 1]$. We then calculate $b = Ax$ and set $x_0 = \mathbf{0}$. For the single row variations Algorithm 1, 2, 9 we set $N = 10$. For the block variations that do not create blocks iteratively Algorithm 5, 10 we create blocks of size 10 and set $N = 100$. For the greedy Kaczmarz method Algorithm 8 we set $N = 3000$ and $\mu = 0, 8$. For all randomized variations we take the average of 10 runs. All variations will use $\epsilon = 0.001$.

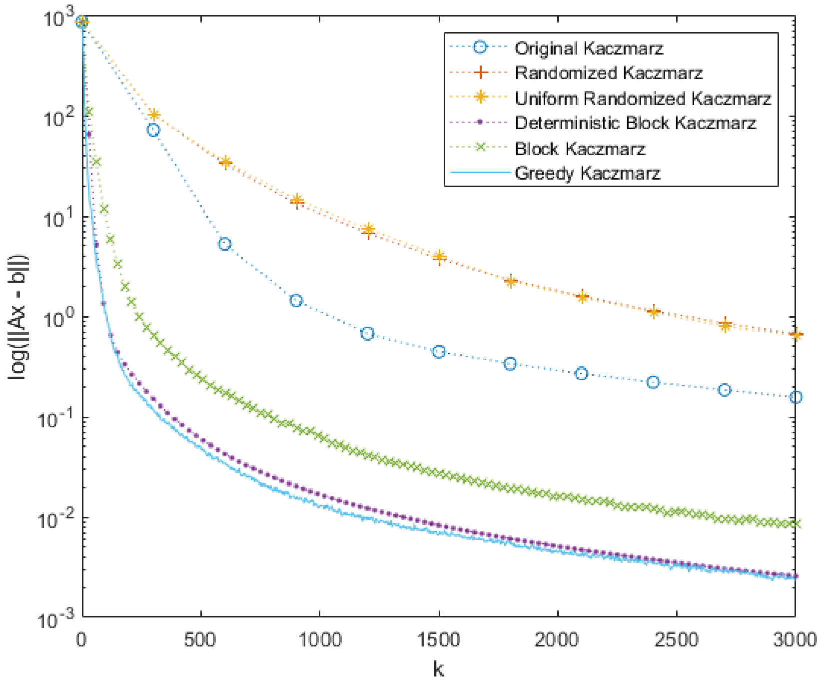

Figure 4: Semilog plot of the Kaczmarz methods on the system described in section 5.1.2 with $||Ax - b||_2^2$ on the y axis and k on the x axis.

5.1.3 Overdetermined with $\sigma_{\max}/\sigma_{\min} = 10^5$ *figure. 5*

To construct A we first constructed a matrix $\tilde{A} \in R^{5000\times300}$ with uniformly distributed values $[0,1]$. Then we took the singular value decomposition and changed the singular values such that $\frac{1}{100000} \leq \sigma \leq 1$. Then constructed a solution $\tilde{x} \in R^{300\times1}$ with uniformly distributed values $[0,1]$. We then calculate $b = Ax$ and set $x_0 = \mathbf{0}$. For the single row variations Algorithm 1, 2, 9 we set $N = 10$. For the block variations that do not create blocks iteratively Algorithm 5, 10 we create blocks of size 10 and set $N = 100$. For the greedy Kaczmarz method Algorithm 8 we set $N = 3000$ and $\mu = 0,8$. For all randomized variations we take the average of 10 runs. All variations will use $\epsilon = 0.001$.

5.1.4 Underdetermined with $\sigma_{\max}/\sigma_{\min} = 1 + 10^{-1}$ *figure. 6*

To construct A we first constructed a matrix $\tilde{A} \in R^{300\times5000}$ with uniformly distributed values $[0,1]$. Then we took the singular value decomposition and changed the singular values such that $1 \leq \sigma \leq 1 + 10^{-1}$. Then constructed a solution $\tilde{x} \in R^{5000\times1}$ with uniformly distributed values $[0,1]$. We then calculate $b = Ax$ and set $x_0 = \mathbf{0}$. For the single row variations Algorithm 1, 2, 9 we set $N = 10$. For the block variations that do not create blocks iteratively Algorithm 5, 10 we create blocks of size 10 and set $N = 100$. For the greedy Kaczmarz method Algorithm 8 we set $N = 3000$ and $\mu = 0,8$. For all randomized variations we take the average of 10 runs. All variations will use $\epsilon = 0.001$.

5.1.5 Overdetermined with $\sigma_{\max}/\sigma_{\min} = 1 + 10^{-1}$ *figure. 7*

To construct A we first constructed a matrix $\tilde{A} \in R^{5000\times300}$ with uniformly distributed values $[0,1]$. Then we took the singular value decomposition and changed the singular values such that $1 \leq \sigma \leq 1 + 10^{-1}$. Then constructed a solution $\tilde{x} \in R^{300\times1}$ with uniformly distributed values $[0,1]$. We then calculate $b = Ax$ and set $x_0 = \mathbf{0}$. For the single row variations Algorithm 1, 2, 9 we set $N = 10$. For the block variations that do not create blocks iteratively Algorithm 5, 10 we create blocks of size 10 and set $N = 100$. For the greedy Kaczmarz method Algorithm 8 we set $N = 3000$ and $\mu = 0,8$. For all randomized variations we take the average of 10 runs. All variations will use $\epsilon = 0.001$.

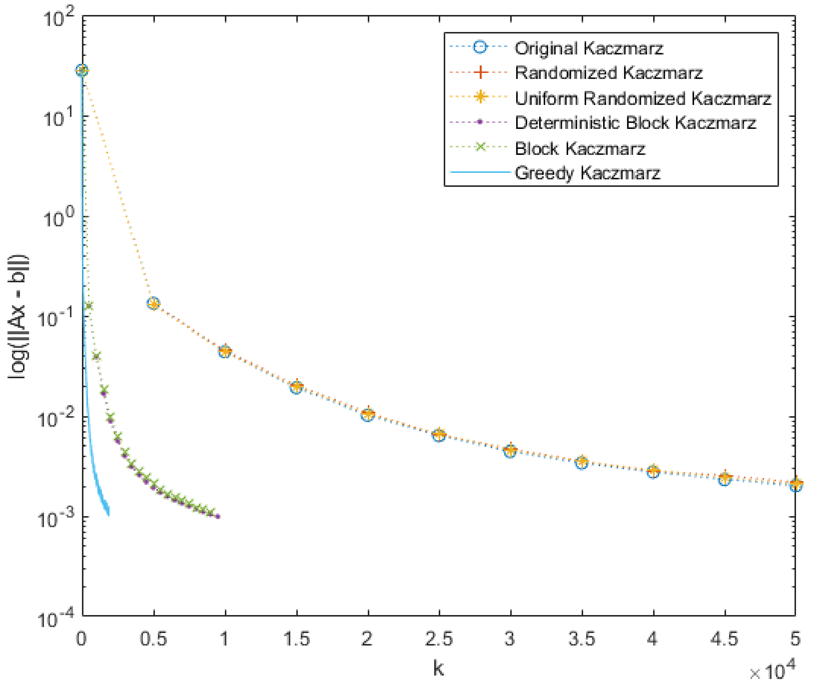

Figure 5: Semilog plot of the Kaczmarz methods on the system described in section 5.1.3 with $||Ax - b||_2^2$ on the y axis and k on the x axis.

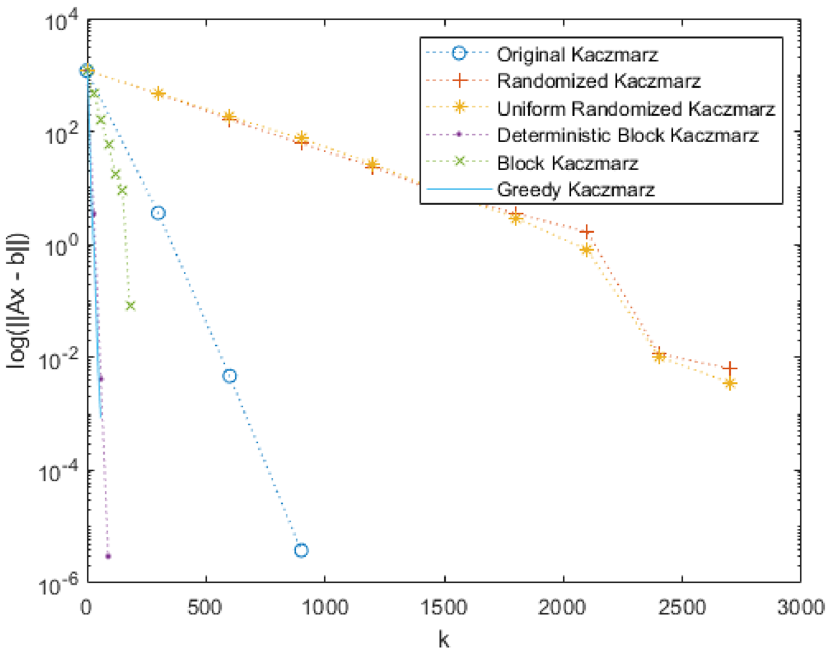

Figure 6: Semilog plot of the Kaczmarz methods on the system described in section 5.1.4 with $||Ax - b||_2^2$ on the y axis and k on the x axis.

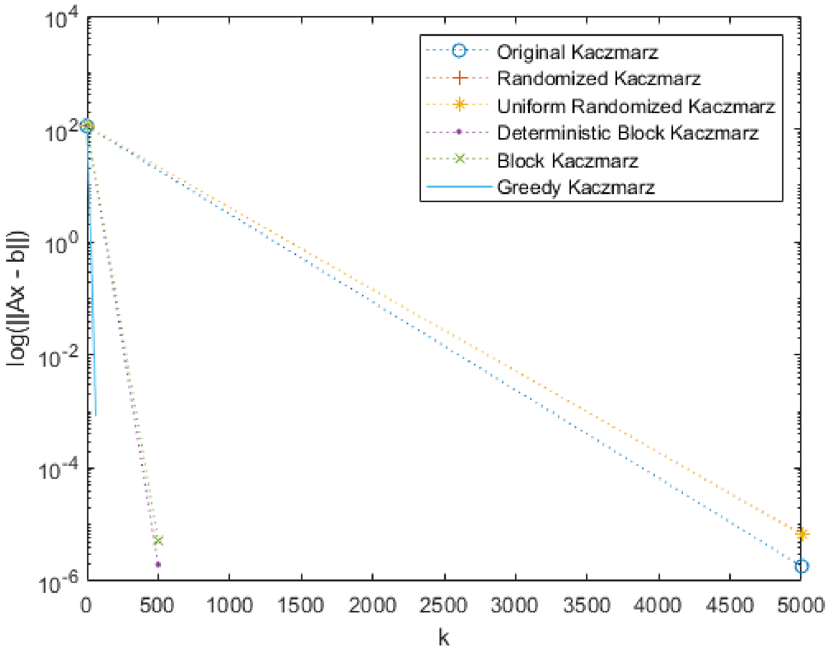

Figure 7: Semilog plot of the Kaczmarz methods on the system described in section 5.1.5 with $||Ax - b||_2^2$ on the y axis and k on the x axis.

5.1.6 Conclusion

From all the figures 4, 5, 6, 7 we can see that the more rows we evaluate in each iteration the less iterations we need to converge which is in line with the result in [9], however it is important to realize that the amount of rows evaluated in each iteration also affects the computational time. In the tests conducted it also seems as if the deterministic variations Algorithm 1, 10 outperform the randomized variations Algorithm 2, 9, 5 when the same amount of rows were evaluated in each iteration, on underdetermined systems figure 4, 6.

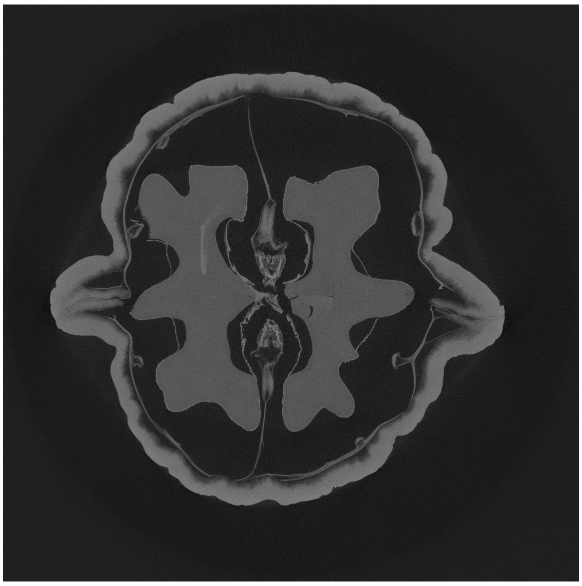

Figure 8: Ground truth of the problem described in section 5.2.1 with data from [4].

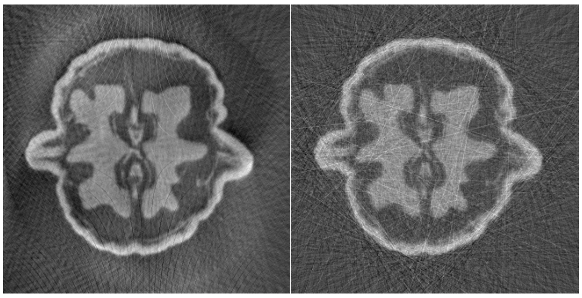

Figure 9: Reconstruction of the problem described in section 5.2.1 with data from [4].
Left: Reconstruction using Algorithm 1.
Right: Reconstruction using Algorithm 2.

5.2 Image reconstruction

5.2.1 Image reconstruction *image. 9, 10, 11*

In 1970 the first application of the Kaczmarz method occurred, Gordon, Bender and Herman used it to reconstruct images from the data given from a computed tomography [3]. Thus we want to tests the different variations of the Kaczmarz method on an image reconstruction problem. The reconstruction problem is sourced from `http://www.fips.fi/dataset.php` [4], we use the most pixelated walnut as our reconstruction problem. The largest sample is a system of equations given by $A \in \mathbf{R}^{39360 \times 107584}$, $b \in \mathbf{R}^{39360}$ and $x \in \mathbf{R}^{107584}$. It is ill-conditioned, underdetermined and noisy, however we expect the noise to be relatively small. We will use Algorithm 1, 2, 9, 5, 10, 8 to reconstruct the image. For the Algorithm 5, 10 we will use a block size of 60. For Algorithm 8 we will use $\eta = 0.8$. All Algorithms 1, 2, 9, 5, 10, 8 will let $N = 1$.

5.2.2 Conclusion

The images 9, 10, 11 of Algorithm 1, 2, 9, 5, 10 seem to relatively well reconstruct the image 8. Unfortunately Algorithm 8 is less clear which seem to run counter the results in figure 4, 5, 6, 7 however this is for mostly due to the amount of iterations the Algorithm performs, as in the image 10 it only performs one iteration as we choose to evaluate the same amount of rows for all Algorithms 1 2, 9, 10, 5, 8.

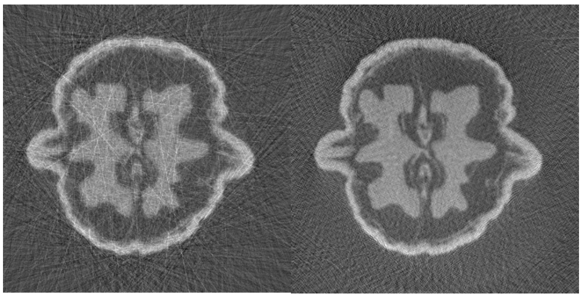

Figure 10: Reconstruction of the problem described in section 5.2.1 with data from [4].
Left: Reconstruction using Algorithm 9.
Right: Reconstruction using Algorithm 10.

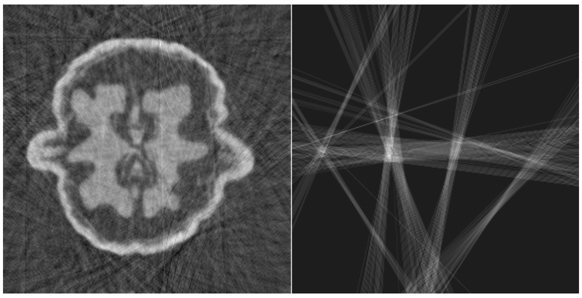

Figure 11: Reconstruction of the problem described in section 5.2.1 with data from [4].
Left: Reconstruction using Algorithm 5.
Right: Reconstruction using Algorithm 8.

5.3 Test on inconsistent problems

5.3.1 Inconsistent tests

To solve inconsistent problems we can either use Algorithm 4, 7 or first Algorithm 3, 6 then apply any solver for consistent systems.

We construct the same matrices as in section 5.1.1 except that the matrices are not necessarily consistent. Then we will compare $||Ax - b||_2$ of Algorithm 4, 11, 7, 12, 2, 5 and the termination criterion of Algorithm 4, 11, 7, 12.

5.3.2 Extended underdetermined with $\sigma_{\max}/\sigma_{\min} = 10^5$ *figure. 12, 13, 14*

To construct A we first constructed a matrix $\tilde{A} \in \mathbf{R}^{5000\times 300}$ with uniformly distributed values $[0, 1]$. Then we took the singular value decomposition and changed the singular values such that $\frac{1}{100000} \leq \sigma \leq 1$. Then we construct $b \in \mathbf{R}^{5000\times 1}$ with uniformly distributed values $[0, 1]$. For the single row Algorithms 4, 11, 2 we set $N = 10$. For the block Algorithms 7, 12, 5 we create blocks of size 10 and set $N = 100$. We take the average of 10 runs. All Algorithms 4, 11, 7, 12, 2, 5 will use $\epsilon = 0.00001$.

Algorithm 11 Transform then solve randomized Kaczmarz method [3, 2]

procedure (A, b, x_0, N) $\triangleright$ $A \in \mathbf{R}^{m\times n}$, $b \in \mathbf{R}^m$, $x_0 \in \mathbf{R}^n$, $N \in \mathbf{N}$
  $z^{(0)} = b$
  repeat
    Choose column j_p with a probability given by $||a_{(j)}||_2^2/||A||_F^2$.

$$z^{(p+1)} = z^{(p)} - \frac{\langle a_{(j_p)}, z^{(p)}\rangle}{||a_{(j_p)}||_2^2} a_{(j_p)}$$

    if $k = 0 \bmod \min(m, n)$ **and** $||A^T z^{(p+1)}||_2^2 \le \epsilon$ **then**
      Terminate loop
  until $k + 1 > Nn$
  repeat
    Choose row i_k with a probability given by $||a_i||_2^2/||A||_F^2$

$$x_{k+1} = x_k + \frac{(b_{i_k} z_{i_k}^{p+1}) - \langle a_{i_k}^T, x_k\rangle}{||a_{i_k}^T||_2^2} a_{i_k}^T$$

    if $k = 0 \bmod \min(m, n)$ **and** $||Ax_{k+1} - (b - z^{(p+1)})||_2^2 \le \epsilon$ **then**
      return x_{k+1}
  until $k + 1 > Nm$
return x_{k+1}

Algorithm 12 Transform then solve block Kaczmarz method [6, 5]

procedure (A, b, x_0, N) $\triangleright$ $A \in \mathbf{R}^{m\times n}$, $b \in \mathbf{R}^m$, $x_0 \in \mathbf{R}^n$, $N \in \mathbf{N}$
  Construct the blocks $A_\xi^T \in \mathbf{R}^{c_\xi \times m}$
  Construct the blocks $A_\tau \in \mathbf{R}^{t_\tau \times n}$ and the corresponding $b_\tau \in \mathbf{R}^{t_\tau}$
  Let c be the amount of blocks A_ξ^T created
  Let t be the amount of blocks A_τ created
  $z^{(0)} = b$
  repeat
    Choose ξ_p with uniform distribution

$$z^{(p+1)} = z^{(p)} - (A_{\xi_p}^T)^\dagger A_{\xi_p}^T z^{(p)}$$

    if $k = 0 \bmod \min(c, t)$ **and** $||A^T z^{(p+1)}||_2^2 \le \epsilon$ **then**
      Terminate loop
  until $p + 1 > Nc$
  repeat
    Choose τ_k with uniform distribution

$$x_{k+1} = x_k + A_{\tau_k}^\dagger \left((b_{\tau_k} - z_{\tau_k}^{(p+1)}) - A_{\tau_k} x_k \right)$$

    if $k = 0 \bmod \min(c, t)$ **and** $||Ax_{k+1} - (b - z^{(p+1)})||_2^2 \le \epsilon$ **then**
      return x_{k+1}
  until $k + 1 > Nt$
return x_{k+1}

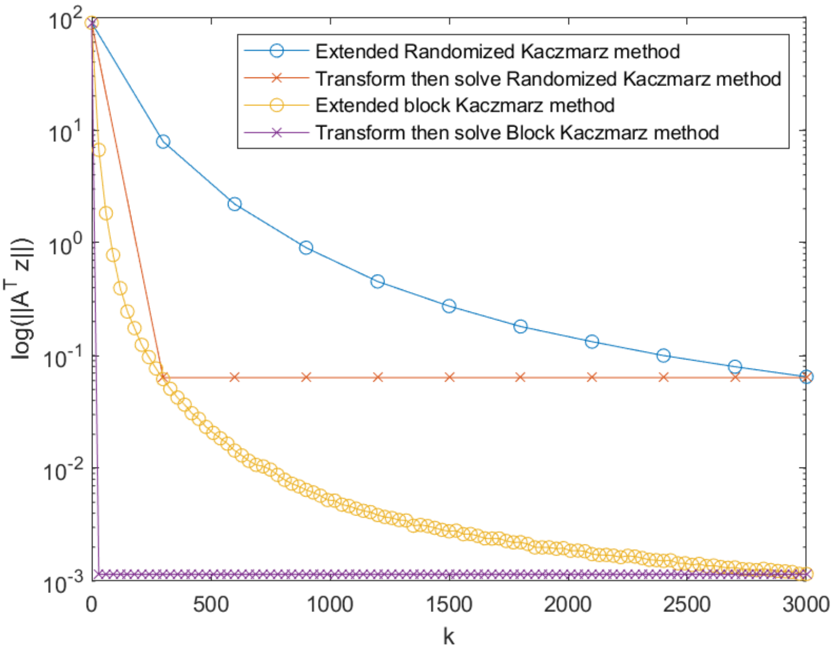

Figure 12: Semilog plot of the extended Kaczmarz methods on the system described in section 5.3.2 with $||A^T z^{(p)}||_2^2$ or $||A^T z^{(k)}||_2^2$ on the y axis and k on the x axis.

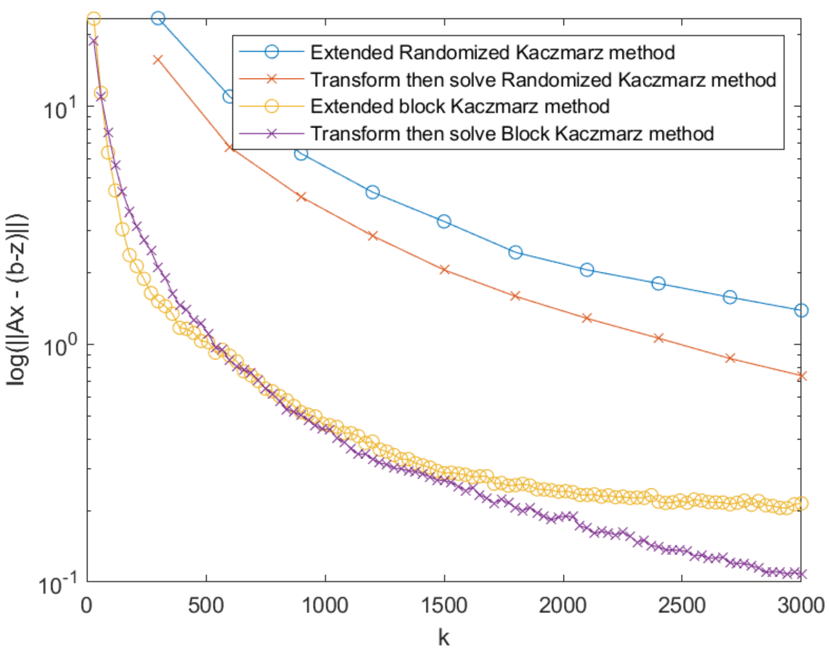

Figure 13: Semilog plot of the extended Kaczmarz methods on the system described in section 5.3.2 with $||A^T x_k - (b - z^{(p)})||_2^2$ or $||A^T x_k - (b - z^{(k)})||_2^2$ on the y axis and k on the x axis.

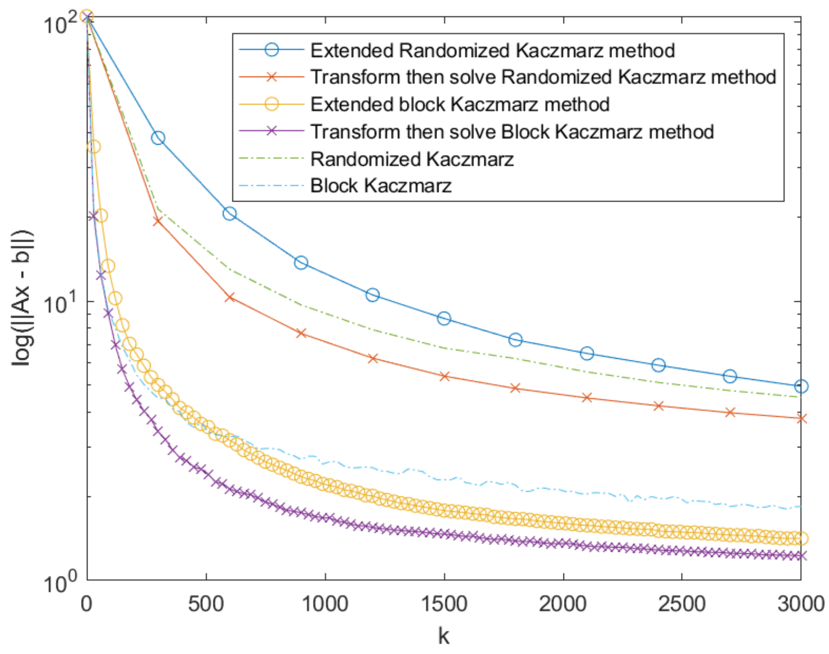

Figure 14: Semilog plot of the extended Kaczmarz methods on the system described in section 5.3.2 with $||A^T x_k - b||_2^2$ on the y axis and k on the x axis.

5.3.3 Extended overdetermined with $\sigma_{\max}/\sigma_{\min} = 10^5$ *figure. 15, 16, 17*

To construct A we first constructed a matrix $\tilde{A} \in \mathbf{R}^{300\times 5000}$ with uniformly distributed values $[0,1]$. Then we took the singular value decomposition and changed the singular values such that $\frac{1}{100000} \leq \sigma \leq 1$. Then we construct $b \in \mathbf{R}^{300\times 1}$ with uniformly distributed values $[0,1]$. For the single row Algorithms 4, 11, 2 we set $N = 10$. For the block Algorithms 7, 12, 5 we create blocks of size 10 and set $N = 100$. We take the average of 10 runs. All Algorithms 4, 11, 7, 12, 2, 5 will use $\epsilon = 0.00001$.

5.3.4 Extended underdetermined with $\sigma_{\max}/\sigma_{\min} = 1 + 10^{-1}$ *figure. 18, 19, 20*

To construct A we first constructed a matrix $\tilde{A} \in \mathbf{R}^{5000\times 300}$ with uniformly distributed values $[0,1]$. Then we took the singular value decomposition and changed the singular values such that $1 \leq \sigma \leq 1 + 10^{-1}$. Then we construct $b \in \mathbf{R}^{5000\times 1}$ with uniformly distributed values $[0,1]$. For the single row Algorithms 4, 11, 2 we set $N = 10$. For the block Algorithms 7, 12, 5 we create blocks of size 10 and set $N = 100$. We take the average of 10 runs. All Algorithms 4, 11, 7, 12, 2, 5 will use $\epsilon = 0.00001$.

5.3.5 Extended overdetermined with $\sigma_{\max}/\sigma_{\min} = 1 + 10^{-1}$ *figure. 21, 22, 23*

To construct A we first constructed a matrix $\tilde{A} \in \mathbf{R}^{300\times 5000}$ with uniformly distributed values $[0,1]$. Then we took the singular value decomposition and changed the singular values such that $1 \leq \sigma \leq 1 + 10^{-1}$. Then we construct $b \in \mathbf{R}^{5000\times 1}$ with uniformly distributed values $[0,1]$. For the single row Algorithms 4, 11, 2 we set $N = 10$. For the block Algorithms 7, 12, 5 we create blocks of size 10 and set $N = 100$. We take the average of 10 runs. All Algorithms 4, 11, 7, 12, 2, 5 will use $\epsilon = 0.00001$.

5.3.6 Conclusion

In the tests for section 5.3 we see that Algorithm 4, 11, 7, 12 converge to some value in figure 17, 23, 20, 20 which from the Theorem 9, 12 can be assumed to be the least squares solution minimizing the distance to x_0. This is further illustrated by Algorithm 2, 5 ping-ponging further away from the solution the larger the value Algorithm 4, 11, 7, 12 converges towards. The main part I wanted to illustrate with the tests in section 5.3 was however how a change in order of operations could impact

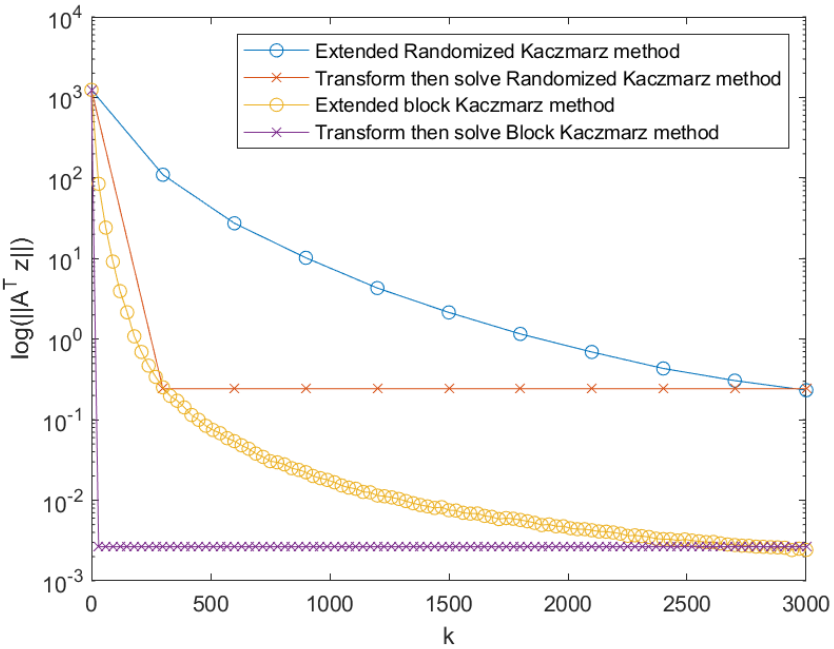

Figure 15: Semilog plot of the extended Kaczmarz methods on the system described in section 5.3.3 with $||A^T z^{(p)}||_2^2$ or $||A^T z^{(k)}||_2^2$ on the y axis and k on the x axis.

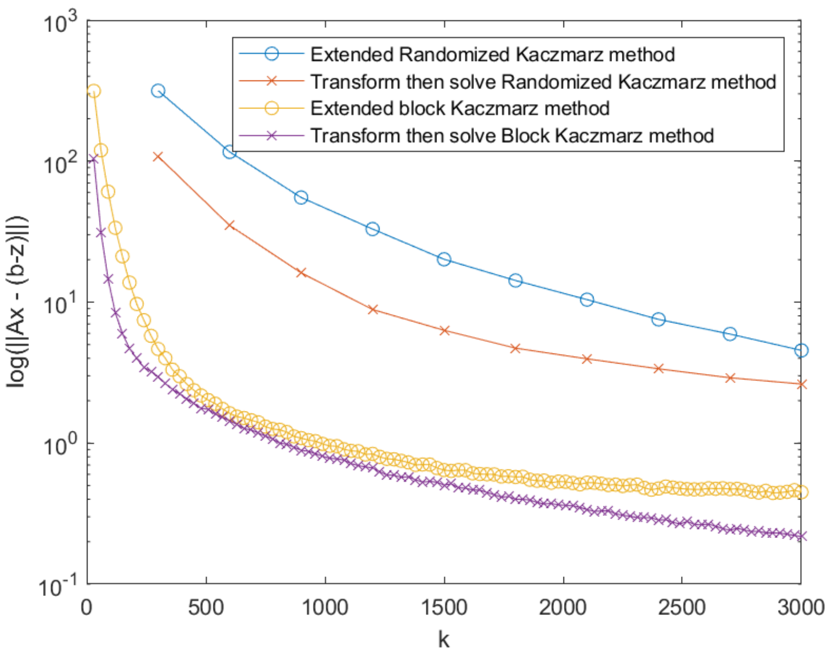

Figure 16: Semilog plot of the extended Kaczmarz methods on the system described in section 5.3.3 with $||A^T x_k - (b - z^{(p)})||_2^2$ or $||A^T x_k - (b - z^{(k)})||_2^2$ on the y axis and k on the x axis.

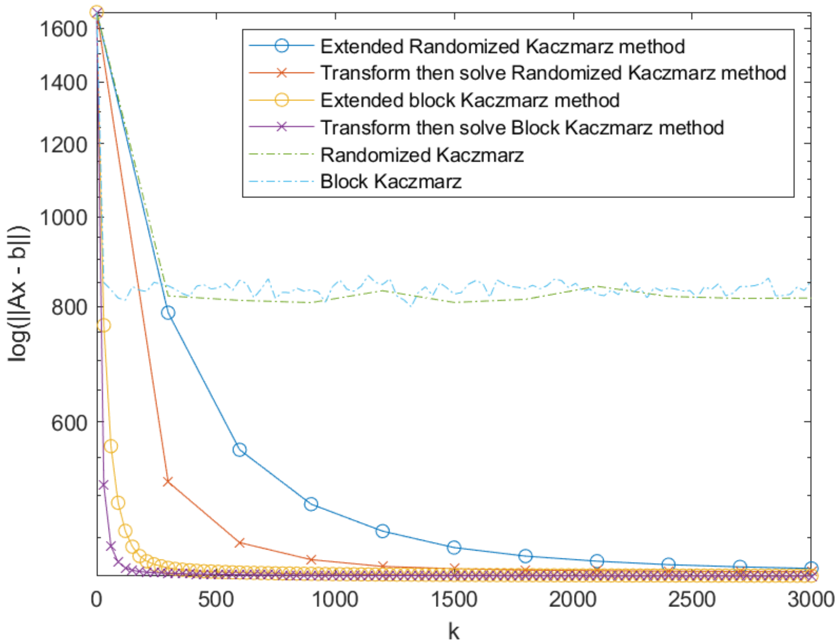

Figure 17: Semilog plot of the extended Kaczmarz methods on the system described in section 5.3.3 with $||A^T x_k - b||_2^2$ on the y axis and k on the x axis.

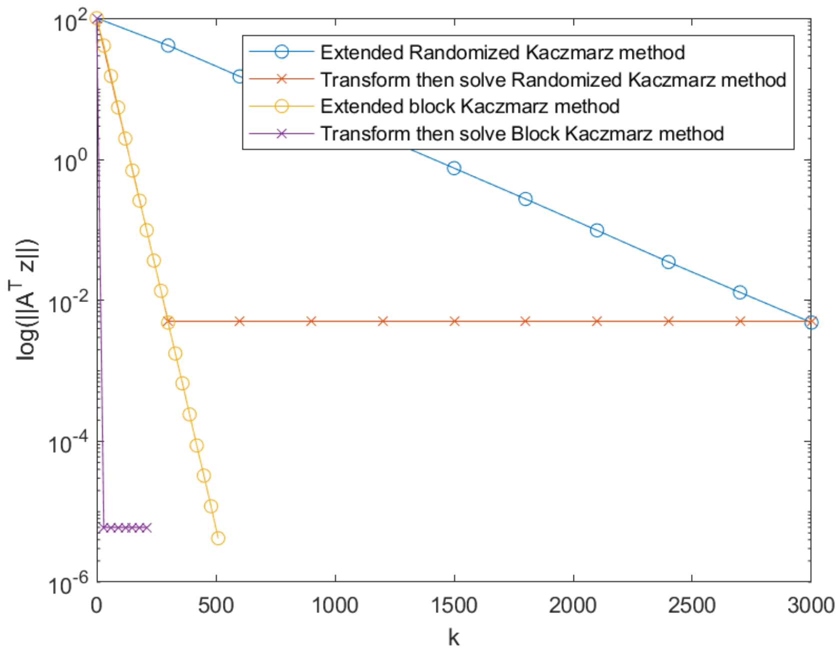

Figure 18: Semilog plot of the extended Kaczmarz methods on the system described in section 5.3.4 with $||A^T z^{(p)}||_2^2$ or $||A^T z^{(k)}||_2^2$ on the y axis and k on the x axis.

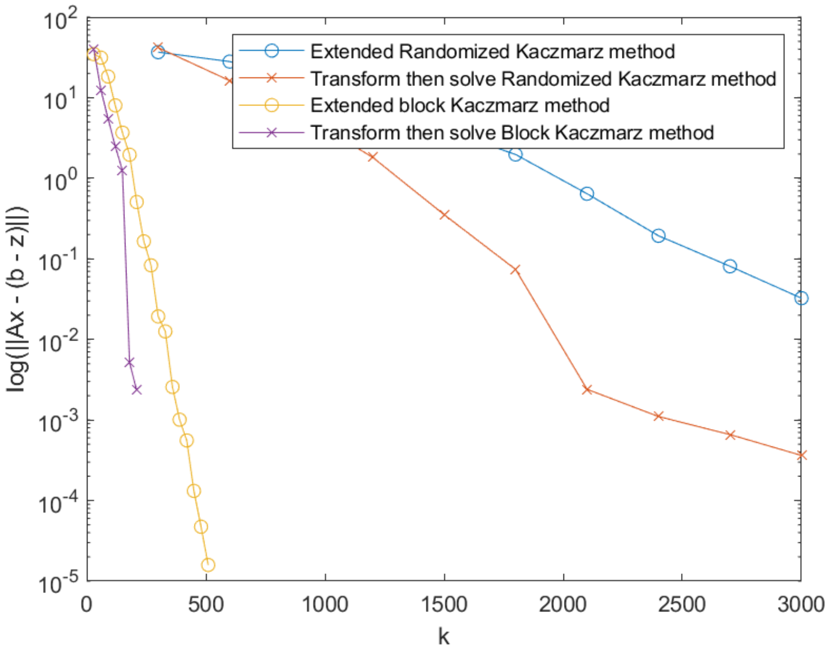

Figure 19: Semilog plot of the extended Kaczmarz methods on the system described in section 5.3.4 with $||A^T x_k - (b - z^{(p)})||_2^2$ or $||A^T x_k - (b - z^{(k)})||_2^2$ on the y axis and k on the x axis.

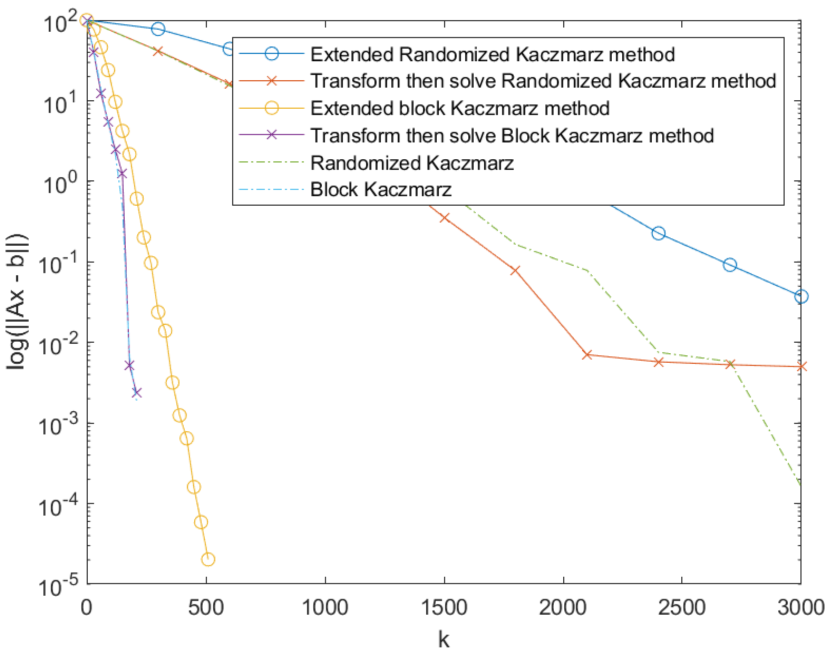

Figure 20: Semilog plot of the extended Kaczmarz methods on the system described in section 5.3.4 with $||A^T x_k - b||_2^2$ on the y axis and k on the x axis.

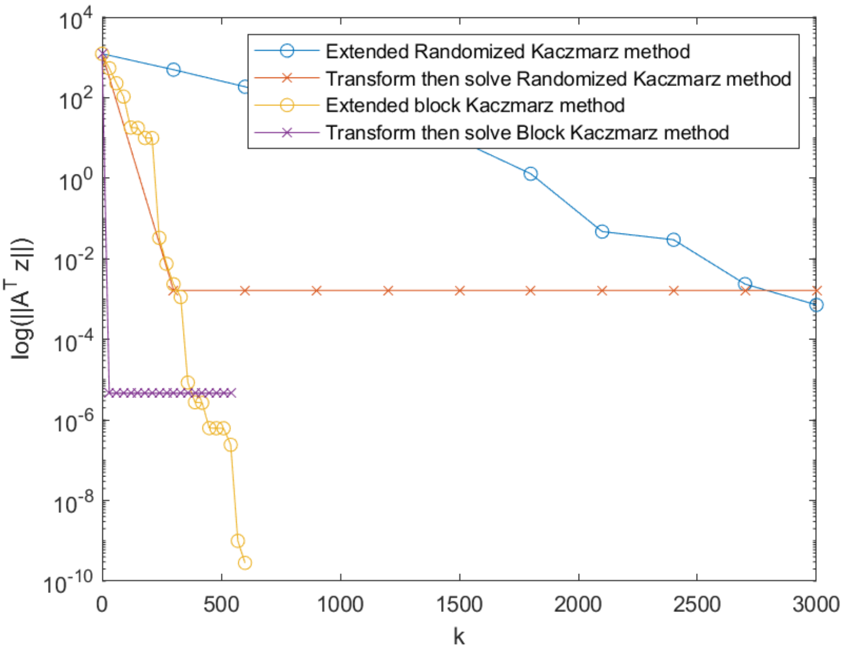

Figure 21: Semilog plot of the extended Kaczmarz methods on the system described in section 5.3.5 with $||A^T z^{(p)}||_2^2$ or $||A^T z^{(k)}||_2^2$ on the y axis and k on the x axis.

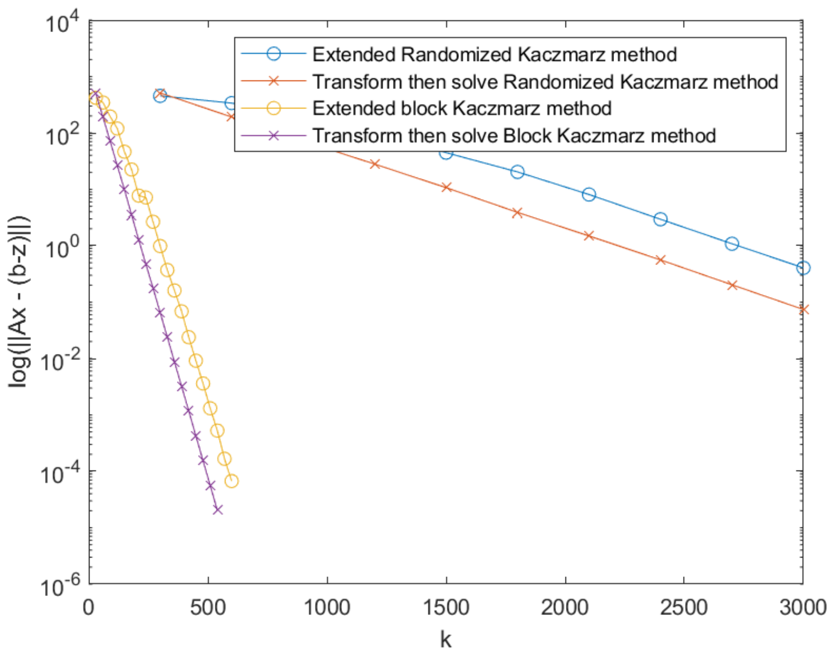

Figure 22: Semilog plot of the extended Kaczmarz methods on the system described in section 5.3.5 with $||A^T x_k - (b - z^{(p)})||_2^2$ or $||A^T x_k - (b - z^{(k)})||_2^2$ on the y axis and k on the x axis.

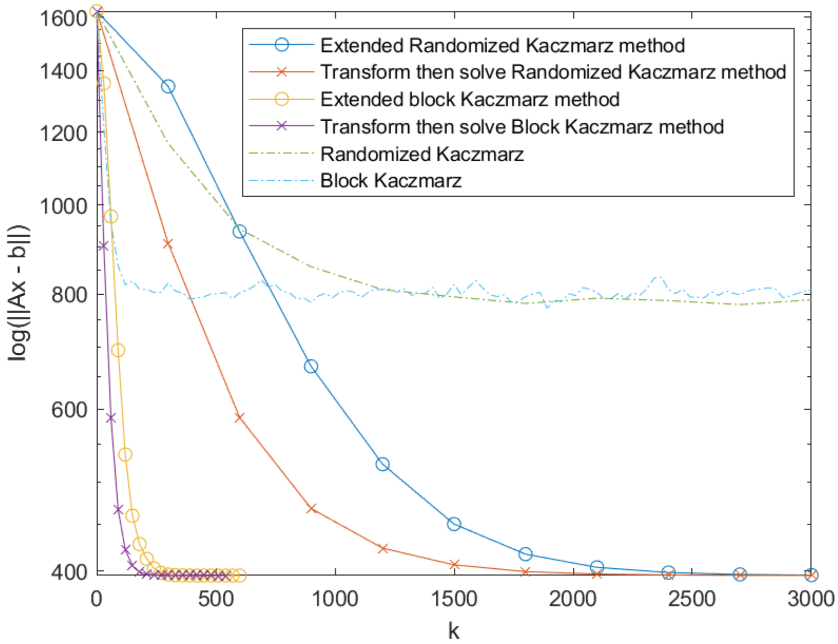

Figure 23: Semilog plot of the extended Kaczmarz methods on the system described in section 5.3.5 with $||A^T x_k - b||_2^2$ on the y axis and k on the x axis.

the convergence. In figure 15, 12, 21, 18 we can see the difference of order of operation of Algorithm 4, 11, 7, 12, with respect to x_{k+1}, however it should be acknowledged that with respect to $x^{(z+1)}$ the convergence does not differ. This change in order of operation is best illustrated by looking at both figure 17, 23, 20, 20 and 16, 22, 19, 19 where Algorithm 11 consistently outperforms Algorithm 4 and Algorithm 12 outperforms Algorithm 7 by a small margin. This is a rather intuitive result since in Algorithm 11, 12 we first transform our system of equations into the desired system of equations and then solve it, while in Algorithm 4, 7 we converge towards a new system of equations in each iteration.

6 Summary and Improvements

Initially we presented different variations of the Kaczmarz method, most of which had randomized row or block selection schemes. In our tests the cyclic row or block selection schemes seemed to perform slightly better especially on underdetermined systems, however we do not know if any improvement in the randomized variations could be found if we force the Algorithms to choose every row or block in each cycle like Needle suggested in [9]. The cyclic randomized variation only seems plausibly applied if the rows are selected with a distribution that does not require any calculations in each iteration like the uniform distribution and since our results for these sort of matrices was very similar between Algorithm 9 and Algorithm 2 it seems rather interchangeable for this sort of matrices.
It could also be interesting to see if the difference on required iterations on the systems presented in section 5.1.2, 5.1.4 is due to rank of the matrix or the ratio $\frac{n}{m}$.
We also saw Algorithm 2, 5 handle the underdetermined systems relatively well. This could be due to construction of the presented in section 5.3.2, 5.3.4 is likelier to construct system with small noise. It could thus be interesting to see the result on a underdetermined system with rather large noise.

References

[1] Lars-Christer Böiers. *Mathematical methods of optimization.* Studentlitteratur AB, 2010.

[2] Aurél Galántai. *Projectors and projection methods*, volume 6. Springer Science & Business Media, 2003.

[3] Richard Gordon, Robert Bender, and Gabor T Herman. Algebraic reconstruction techniques (art) for three-dimensional electron microscopy and x-ray photography. *Journal of theoretical Biology*, 29(3):471–481, 1970.

[4] Keijo Hämäläinen, Lauri Harhanen, Aki Kallonen, Antti Kujanpää, Esa Niemi, and Samuli Siltanen. Tomographic x-ray data of a walnut. *arXiv preprint arXiv:1502.04064*, 2015.

[5] Anders Holst and Victor Ufnarovski. *Matrix Theory.* Studentlitteratur, 2014.

[6] Menila James. The generalised inverse. *The Mathematical Gazette*, 62(420):109–114, 1978.

[7] Stefan Karczmarz. Angenäherte Auflösung von systemen linearer gleichungen. *Bull. Int. Acad. Pol. Sic. Let., Cl. Sci. Math. Nat.*, pages 355–357, 1937.

[8] Deanna Needell. Randomized kaczmarz solver for noisy linear systems. *BIT Numerical Mathematics*, 50(2):395–403, 2010.

[9] Deanna Needell and Joel A Tropp. Paved with good intentions: analysis of a randomized block kaczmarz method. *Linear Algebra and its Applications*, 441:199–221, 2014.

[10] Deanna Needell, Ran Zhao, and Anastasios Zouzias. Randomized block kaczmarz method with projection for solving least squares. *Linear Algebra and its Applications*, 484:322–343, 2015.

[11] Yu-Qi Niu and Bing Zheng. A greedy block kaczmarz algorithm for solving large-scale linear systems. *Applied Mathematics Letters*, 104:106294, 2020.

[12] Thomas Strohmer and Roman Vershynin. A randomized kaczmarz algorithm with exponential convergence. *Journal of Fourier Analysis and Applications*, 15(2):262–278, 2009.

[13] Endre Süli and David F Mayers. *An introduction to numerical analysis.* Cambridge university press, 2003.

[14] Anastasios Zouzias and Nikolaos M Freris. Randomized extended kaczmarz for solving least squares. *SIAM Journal on Matrix Analysis and Applications*, 34(2):773–793, 2013.

Bachelor's Theses in Mathematical Sciences 2023:K23
ISSN 1654-6229

LUNFNA-4050-2023

Numerical Analysis
Centre for Mathematical Sciences
Lund University
Box 118, SE-221 00 Lund, Sweden

http://www.maths.lu.se/