"OVIDIUS" UNIVERSITY OF CONSTANŢA
DOCTORAL SCHOOL OF APPLIED SCIENCES
MATHEMATICS DOCTORAL PROGRAM

DOCTORAL THESIS

SUMMARY

ITERATIVE PROJECTION ALGORITHMS FOR LINEAR LEAST SQUARES PROBLEMS

Scientific coordinator
Prof. univ. dr. Constantin POPA

Ph.D. Student
Ioana PANTELIMON

CONSTANŢA
2013

Preface

In the field of numerical linear algebra, various applications such as image reconstruction and processing give rise to large, sparse and ill-conditioned linear systems of equations. Due to inexact measurements or unit roundoff, these problems are not always consistent, i.e., the right hand side is not contained in the range of the system matrix. When this is the case, one may solve a corresponding linear least squares problem. Iterative algorithms, although computationally expensive, usually provide better results in the presence of noise or incomplete data.

A general iterative method which encompasses Kaczmarz (see, e.g., [49]), Cimmino (see, e.g., [31]), Diagonal Weighting (see, e.g., [42]) or Landweber (see, e.g., [30, 43]) algorithms was proposed in [31] to solve the linear least squares problem. The authors proved that an extended version of the general method converges even in the inconsistent case to the solution set of the aforementioned linear least squares problem.

When solving an image reconstruction problem, previous knowledge concerning the original image may lead to various constraining strategies. If available, a priori information about the original image, such as smoothness or that it belongs to a certain closed convex set, may be used to improve the reconstruction quality. In constraining iterative processes, the algorithmic operator of the iterative process is pre-multiplied by a constraining operator at each iterative step. This enables the constrained algorithm, besides solving the original problem, also to find an approximation that incorporates some prior knowledge about the exact solution. This approach has been useful in image reconstruction and other image processing situations when a single constraining operator was used [7, 11, 14, 24, 25, 26, 28, 47, 50].

Another topic of interest is the performance improvement of iterative algorithms. For example, in [45] the authors proposed a procedure of transforming the classical Kaczmarz algorithm into a direct method.

According to these aspects, the thesis is organised as follows.

In the first chapter we present some preliminary results and the general method proposed in [31].

In Chapter 2 we study constraining of iterative processes by a family of operators rather than by a single operator. In Section 2.1 we start with a brief presentation of the constrained general iterative algorithm from [31] and the corresponding main results. We propose a new adaptive iteration-dependent constraining procedure which employs a family of strictly nonexpansive idempotent functions. Under some supplementary assumptions we prove the convergence of the family constrained general algorithm. We also give an example of a family of box projection functions which satisfy our additional hypotheses. In Section 2.2, in a more general setting, for a family of strictly nonexpansive operators with nonempty common fixed points set and a supplementary condition, appropriate in the context of image reconstruction problems, we adapt some results from [13] for our purpose. We prove the convergence of an iterative process determined by this family of strictly nonexpansive operators. We present the *family-constrained algorithm* (FCA) and prove that the series expansion methods and the smoothing matrices used in [13] obey all our hypotheses. Furthermore, we show that the general iterative method [31] is itself an algorithmic operator of the form required for the convergence of the FCA Algorithm and that the earlier mentioned example of a family of nonlinear constraining operators satisfies our assumptions.

In Chapter 3 we try to establish convergence for extended block versions of Kaczmarz and Jacobi projection methods. Our purpose is to show that these algorithms are special cases of the extended general iterative process, which was given in [31]. Replacing the inverse operator with the Moore-Penrose pseudoinverse, we generalize previous results from [41].

The fourth chapter is dedicated to the study of an acceleration technique based on the idea of adding row and column directions for projection to the linear system. In the paper [45], an extended version of the classical Kaczmarz algorithm was shown to yield in only one iteration a solution of our linear least squares problem. Unfortunately, as any direct method applied to large sparse matrices, this algorithm usually determines a considerable increase of the fill-in percentage in our data. In order to overcome this difficulty, in the present paper we propose a modified version of this direct Kaczmarz algorithm in which the transformations applied to the system matrix try to conserve the initial sparsity structure. These transformations are done via clustering using Jaccard and Hamming distances. The modified Kaczmarz algorithm is no more a direct method, but we obtain an acceleration of convergence with respect to the classical Kaczmarz algorithm.

In the last chapter we design an adaptive algorithm that uses an iteration-dependent family of constraining functions for some numerical experiments of image reconstruction on Tomographic Particle Image Velocimetry (see, e.g., [35] and [34]). The results are illustrated for different levels of perturbation. We also present numerical results concerning the method of accelerating the Kaczmarz iterative scheme using supplementary directions for projection while preserving the sparsity pattern of the original problem via clustering. We further investigate methods of calculating the minimum number of clusters needed to keep the sparsity of our data under a given threshold.

The original contributions which are presented in this thesis are contained in the following papers.

[10] Y. Censor, **I. Pantelimon**, and C. Popa. Family constraining of iterative algorithms. *Numerical Algorithms*, pages 1–16, 2013. doi: 10.1007/s11075-013-9736-5

[33] **I. Pantelimon** and C. Popa. Constraining by a family of strictly nonexpansive idempotent functions with applications in image reconstruction. *BIT Numerical Mathematics*, 53(2):527–544, 2013. doi: 10.1007/s10543-012-0414-0

[36] **I. Pomparău**. On the acceleration of Kaczmarz projection algorithm. Presented at the 84th Annual Meeting of the International Association of Applied Mathematics and Mechanics, 18–22 March, 2013, Novi Sad, Serbia. Submitted for publication in *Proceedings in Applied Mathematics and Mechanics*, 2013

[37] **I. Pomparău**. On an accelerated version of the Kaczmarz algorithm with a priori clustering. In *Topics in mathematical modelling of life science problems*, pages 61–76. Matrix Rom, Bucureşti, 2013. Presented at the Nineth Workshop on Mathematical Modeling of Environmental and Life Sciences Problems, 1–4 November, 2012, Constanţa, România

[38] **I. Pomparău** and C. Popa. Supplementary projections for the acceleration of Kaczmarz algorithm. Submitted for publication, 2013

[39] **I. Pomparău** and C. Popa. Weaker assumptions for convergence of extended block Kaczmarz and Jacobi projection algorithms. Submitted for publication, 2013

1 Background

1.1 Preliminaries

Many real world problems lead after certain discretization steps to solving systems of linear equations of the form $Ax = b$, for an $m \times n$ matrix A and the vector $b \in \mathbb{R}^m$. If modelling or measurement errors occur, then the system may become inconsistent and we arrive at solving a linear least squares problem: find $x \in \mathbb{R}^n$ such that

$$\|Ax - b\| = \min\{\|Az - b\|, z \in \mathbb{R}^n\}. \tag{1}$$

We will further use the notations $A^T, A^\dagger, A_i, A^j, \mathcal{R}(A), \mathcal{N}(A),\ P_V,\ x_{LS}$ and $LSS(A; b)$ for the transpose, the (unique) Moore-Penrose pseudoinvese, i-th row, j-th column, range and null space of A, the projection onto a vector subspace V, the (unique) minimal norm solution of problem (1) and the set of all solutions of (1); also $\langle\cdot,\cdot\rangle$, $\parallel z \parallel$ and $\parallel M \parallel$ will denote the Euclidean scalar product, the Euclidean norm of the vector z and the spectral norm of the matrix M. All the vectors appearing in the paper will be considered as column vectors. We will assume in the rest of the paper that

$$A_i \neq 0, A^j \neq 0,\ \text{for all}\ i \in \{1, 2, \dots m\},\ \text{and}\ j \in \{1, 2, \dots n\}.$$

1.2 A general iterative projection method

The following general projection-based iterative solver for the linear least squares problem (1) was proposed in [31].

General Iterative Algorithm (GEN)

Initialization. $x^0 \in \mathbb{R}^n$;

Iterative step.

$$x^{k+1} = Qx^k + Rb, \tag{2}$$

with Q and R matrices of dimensions $n \times n$ and $n \times m$, respectively, which satisfy the following properties:

$$Q + RA = I, \tag{3}$$

$$\forall y \in \mathbb{R}^m,\ Ry \in \mathcal{R}(A^T), \tag{4}$$

$$\text{if}\ \tilde{Q} = QP_{\mathcal{R}(A^T)}\ \text{then}\ \parallel \tilde{Q} \parallel\ <\ 1. \tag{5}$$

Remark 1 *The above described general formulation includes almost all projection based algorithms in computer tomography, like Kaczmarz, Cimmino, Jacobi Projective, Diagonal Weighting (see, e.g., [49], [31], [15] and [42], respectively). For details and proofs of this statement see [43].*

Theorem 1 *([31, Theorem 2.2]) If (3)-(5) hold, for any $x^0 \in \mathbb{R}^n$, the sequence $(x^k)_{k\geq 0}$ generated by* (2) *converges and*

$$\lim_{k\to\infty} x^k = P_{\mathcal{N}(A)}(x^0) + x_{LS} + \Delta, \quad \text{with} \quad \Delta = (I - \tilde{Q})^{-1} R P_{\mathcal{N}(A^T)}(b). \tag{6}$$

Remark 2 *If the problem* (1) *is consistent, we have $\Delta = 0$ and the limit in* (6) *is an element of $S(A;b) = \{x \in \mathbb{R}^n | \ Ax = b\}$, for all $x^0 \in \mathbb{R}^n$. In the inconsistent case, $\|\Delta\|$ represents the distance between the limit point in* (6) *and the set $LSS(A;b)$ (for details see, e.g., [44]).*

For the matrices Q, R and U of dimensions $n \times n$, $n \times m$ and $m \times m$, respectively, the authors defined in [31] the following extended general iterative method.

Extended General Algorithm (EGEN)

Initialization. $x^0 \in R^n$ is arbitrary and $y^0 = b$.

Iterative step. For every $k \geq 0$,

$$y^{k+1} = U y^k, \tag{7}$$

$$b^{k+1} = b - y^{k+1}, \tag{8}$$

$$x^{k+1} = Q x^k + R b^{k+1}. \tag{9}$$

When Q and R satisfy (3)-(5) and U verifies the following general assumptions

$$\text{if } x \in \mathcal{N}(A^T) \text{ then } Ux = x, \tag{10}$$

$$\text{if } x \in \mathcal{R}(A) \text{ then } Ux \in \mathcal{R}(A), \tag{11}$$

$$\text{if } \tilde{U} = U P_{\mathcal{R}(A)} \text{ then } \parallel \tilde{U} \parallel \ < \ 1, \tag{12}$$

the next convergence result was proved.

Theorem 2 *[31, Theorem 2.6] Let us suppose that the matrices Q and R satisfy equations (3)-(5) and for U the properties (10)-(12) hold. Then, for any $x^0 \in \mathbb{R}^n$, the sequence $(x^k)_{k\geq 0}$ generated with the algorithm EGEN converges and*

$$\lim_{k\to\infty} x^k = P_{\mathcal{N}(A)}(x^0) + x_{LS}. \tag{13}$$

2 Family constraining of iterative algorithms

This chapter is about *constraining of iterative processes* which has the following meaning. When dealing with a real-world problem it is sometimes the case that we have some prior knowledge about features of the solution that is being sought after. If possible, such prior knowledge may be formulated as an additional constraint and added to the original problem formulation. But sometimes, when we have already at our disposal a “good” algorithm for solving the original problem without such an additional constraint, it is beneficial to modify the algorithm, rather than the problem, so that it will, in some way, “take care” of the additional constraint (or constraints) without loosing its ability to generate (finitely or asymptotically) a solution to the original problem. This is called constraining of the original iterative algorithm. Given an (algorithmic) operator $\Gamma : R^n \to R^n$ between Euclidean spaces, the original iterative process may have the form

$$x^{k+1} = \Gamma(x^k), \text{ for all } k \geq 0, \tag{14}$$

under various assumptions on Γ. Constraining such an algorithm with a family of operators means that we desire to use instead of (14) the iterative process

$$x^{k+1} = C_k\Gamma(x^k), \text{ for all } k \geq 0, \tag{15}$$

where $\{C_k\}_{k=0}^{\infty}$ is a family of operators $C_k : R^n \to R^n$, henceforth called the *constraining operators*.

Our purpose is to study the possibility to constrain an algorithm with a family of operators and to analyze the asymptotic behavior of such family-constrained algorithms. We extend earlier results on this topic that were limited to a single constraining operator, i.e., $C_k = C$ for all $k \geq 0$, see, e.g., [7, 11, 14, 24, 25, 26, 28, 47, 50].

2.1 Constraining the GEN method using a family of strictly nonexpansive idempotent functions

In the linear least squares problem (1), the matrix A is in general large, sparse, ill-conditioned and rank deficient, leading to the existence of an infinite solution set. The minimal norm solution is usually provided by classical iterative solvers for (1). Unfortunately, x_{LS} is not always close to the solution that we are looking for, say x^{ex}. The well known connection between them is (see, e.g., [43])

$$x^{ex} = P_{\mathcal{N}(A)}(x^{ex}) + x_{LS}. \tag{16}$$

Therefore, one may require to find a better approximation of the exact solution than the one provided by x_{LS}. For this reason apriori information about the x^{ex} solution may be used, if possible, in the form of additional constraints to the problem (1).

In general, for actual applications, one can consider (see, e.g., [20]) that the original image has the components included in some previous known intervals, i.e., $x_i^{ex} \in [a_i, b_i]$, with $[a_i, b_i] \subset \mathbb{R}$ for all $i \in \{1, 2, \ldots, n\}$. If this is the case one may force the i-th component of the approximation to be in the interval $[a_i, b_i]$ at each step of an iterative method. In the paper [28] the authors applied this idea to the classical Kaczmarz iterative algorithm by considering a general constraining function $C : \mathbb{R}^n \longrightarrow \mathbb{R}^n$ with a closed image $\mathrm{Im}(C) = \{y = Cx, x \in \mathbb{R}^n\} \subset \mathbb{R}^n$ and the properties

$$\|Cx - Cy\| \leq \|x - y\|, \tag{17}$$

$$\text{if} \quad \|Cx - Cy\| = \|x - y\| \quad \text{then} \quad Cx - Cy = x - y, \tag{18}$$

$$\text{if} \quad y \in \mathrm{Im}(C) \quad \text{then} \quad y = Cy. \tag{19}$$

We say that an operator $C : \mathbb{R}^n \longrightarrow \mathbb{R}^n$ is *strictly nonexpansive (SNE)* if it obeys conditions (17) and (18).

Remark 3 *An example of a function for which (17)-(19) hold is the orthogonal projection operator onto the box* $[a, b] = [a_1, b_1] \times \cdots \times [a_n, b_n] \subset \mathbb{R}^n$, *defined by*

$$(Cx)_i = \begin{cases} x_i, & \text{if} \quad x_i \in [a_i, b_i] \\ a_i, & \text{if} \quad x_i < a_i \\ b_i, & \text{if} \quad x_i > b_i. \end{cases} \tag{20}$$

In [31], for a function $C : \mathbb{R}^n \longrightarrow \mathbb{R}^n$ with a closed image $\mathrm{Im}(C) \subset \mathbb{R}^n$ and the properties (17)-(19), the authors proposed a constrained GEN algorithm. But practical applications show that a “uniform” constraining procedure (with the same constraining function in any iteration) is not always efficient; sometimes it is necessary to use an adaptive iteration-dependent constraining procedure. For example, in the box constraining case we have to change the interval $[a_i, b_i]$ (“acting” on i-th pixel’s value of the approximate image) in each iteration, in order to better “catch” the appropriate value for that pixel.

In this section we try to accomplish this task by using a family of constraining functions. We consider for any iteration $k \geq 0$ a function $C_k : \mathbb{R}^n \longrightarrow \mathbb{R}^n$ with a closed image $\mathrm{Im}(C_k) \subset \mathbb{R}^n$ and the properties (17)-(19).

2.1.1 Using a family of constraining functions

Let $(C_k)_{k\geq 0}$ be a family of constraining functions for which the properties (17)-(19) and the following assumption hold

(A_1) $V = \bigcap_{k\geq 0} \mathrm{Im}(C_k) \neq \emptyset$.

If we suppose that $\mathcal{V}^*_\infty = \{y \in V,\ y - \Delta \in LSS(A;b)\} \neq \emptyset$ and define for each $k \geq 0$, $\mathcal{V}^*_k = \{z \in \mathrm{Im}(C_k),\ z - \Delta \in LSS(A;b)\}$, we have that $\mathcal{V}^*_\infty = \bigcap_{k\geq 0} \mathcal{V}^*_k$.

We propose the following algorithm.

Family Constrained General Algorithm (CGENk)

Initialization. $x^0 \in \mathrm{Im}(C_0)$;

Iterative step. for any $k \geq 0$

$$x^{k+1} = C_{k+1}[Qx^k + Rb]. \tag{21}$$

We further consider two additional hypotheses on the family $(C_k)_{k\geq 0}$ of constraining functions.

(A_2) The sets $V_k = \mathrm{Im}(C_k)$ are closed;

(A_3) For any $\ell \in \mathbb{N}^*$ there exists $k(\ell) \geq \ell$ such that for any $y \in \mathcal{V}^*_\infty$ and any $k \geq k(\ell)$ we have the inequality

$$\|C_{k+1}[Qx^k + Rb] - y\| \quad \leq \quad \|C_\ell[Qx^k + Rb] - y\|.$$

Theorem 3 *In the above stated hypotheses, the sequence $(x^k)_{k\geq 0}$ generated by (21) converges and*

$$\lim_{k\to\infty} x^k \in \mathcal{V}^*_\infty. \tag{22}$$

2.1.2 A special family of constraining functions which satisfies the supplementary hypotheses

Recall the example shown in Remark 3. For every $k \geq 0$, consider the box $[a^k, b^k] = [a^k_1, b^k_1] \times \cdots \times [a^k_n, b^k_n] \subset \mathbb{R}^n$ and the metric projection operator onto the k-th box $[a^k, b^k]$. It is known that for an operator defined as in (20) the properties (17)-(19) and (A_2) hold and we will now develop a sufficient condition for a family of such operators to satisfy assumptions (A_1) and (A_3).

Proposition 1 *For a family $(C_k)_{k\geq 0}$ of metric projection operators onto the k-th box $[a^k, b^k]$ as defined in (20) and two matrices Q and R of dimensions $n \times n$ and $n \times m$, respectively, for which the properties (3)-(5) hold with the additional property that for all $\ell \geq 0$, there exists $k(\ell) \geq \ell$ such that:*

$$\mathrm{Im}(C_{k+1}) \subset \mathrm{Im}(C_\ell), \forall k \geq k(\ell), \tag{23}$$

the assumptions (A_1) and (A_3) are true.

The metric projection operators, like those in (20), are frequently used for constraining purposes in image reconstruction problems that are formulated according to (1). As mentioned at the beginning of this section, the idea of using iteration independent constraints was previously examined. Our purpose is to explore a procedure of adapting the constraining function at each step of the algorithm to obtain a better approximation of the scanned image. The meaning of (23) in practice is that the image of every constraining function should be built from a priori knowledge to contain the exact solution (the original image), however, $\{\mathrm{Im}(C_k)\}_{k=0}^{\infty}$ should not necessarily be a decreasing nested sequence.

2.2 A more general setting for constraining iterative processes

In this section we introduce a family of strictly nonexpansive operators $\{C_k\}_{k=0}^{\infty}$ and prove the convergence of the family-constrained algorithms in a more general setting. The algorithm constraining approach is successfully applied to problems of image restoration, to smoothing in image reconstruction from projections (see also [20, Subsection 12.3]), and to constraining of linear iterative processes in general.

2.2.1 Convergence for a Family of Strictly Nonexpansive Operators

We will prove in this subsection that, under two special hypotheses, an iterative scheme which employs a family of strictly nonexpansive operators converges to a common fixed point.

For a family $\{T_k\}_{k=0}^{\infty}$ of strictly nonexpansive operators we define the fixed points sets and their intersection by

$$\mathrm{Fix}(T_k) = F_k := \{x \in R^n \mid T_k(x) = x\} \text{ and } F := \cap_{k=0}^{\infty} F_k, \tag{24}$$

respectively, and assume that

$$F \neq \emptyset. \tag{25}$$

Consider the algorithm

$$x^0 \in R^n \text{ and } x^{k+1} = T_{k+1}(x^k),\ \forall\, k \geq 0, \tag{26}$$

Condition 1 *Let $\{T_k\}_{k=0}^{\infty}$ be a family of strictly nonexpansive operators for which (25) holds. If $\{x^k\}_{k=0}^{\infty}$ is any sequence, given by (26), then for every $\ell \geq 0$, there exists an index $k(\ell) \geq 0$ such that*

$$\left\| T_{k+1}(x^k) - z \right\| \leq \left\| T_\ell(x^k) - z \right\|, \tag{27}$$

for all $z \in F$ and all $k \geq k(\ell)$.

Theorem 4 *Let $\{T_k\}_{k=0}^{\infty}$ be a family of strictly nonexpansive operators for which (25) and Condition 1 holds. Any sequence $\{x^k\}_{k=0}^{\infty}$, generated by (26), converges to an element of F.*

Remark 4 *Replacing the strict nonexpansivity of the operators $\{T_k\}_{k=0}^{\infty}$ with the assumption that they belong to the wider class of paracontracting operators (see [18, Definition 1]), the results stated in Theorem 4 still hold.*

2.2.2 The Family-Constrained Algorithm (FCA)

Many iterative algorithms are of, or can be cast into, the form of *one-step stationary iterations* (see, e.g., [32, Chapter 10]). If $\Gamma : R^n \to R^n$ and $C_k : R^n \to R^n$, with $k \geq 0$, are strictly nonexpansive, we define the operators $T_k : R^n \to R^n$ by

$$T_k(x) := C_k \Gamma(x),\ \text{for all } k \geq 0, \tag{28}$$

and prove that they are also strictly nonexpansive.

Proposition 2 *For any $k \geq 0$, an operator T_k as in (28), in which Γ and C_k are strictly nonexpansive, has the following properties:*

$$\|T_k(x) - T_k(y)\| \leq \|x - y\|,\ \text{for all } x, y \in R^n, \tag{29}$$

and

$$\textit{if } \|T_k(x) - T_k(y)\| = \|x - y\|, \textit{ then } T_k(x) - T_k(y) = \Gamma(x) - \Gamma(y) = x - y. \tag{30}$$

For $\{T_k\}_{k=0}^{\infty}$ defined according to (28), with $\{C_k\}_{k=0}^{\infty}$ and Γ strictly nonexpansive, the iterative process (26) may be written as a constrained algorithm.

The Family-Constrained Algorithm (FCA)

Initialization. $x^0 \in R^n$ is arbitrary.

Iterative step. For every $k \geq 0$, g*iven the current iterate* x^k calculate the next iterate x^{k+1} by

$$x^{k+1} = C_{k+1}\Gamma(x^k). \tag{31}$$

Proposition 2 and Theorem 4 yield that if assumptions (25) and Condition 1 hold, then any sequence generated by the FCA algorithm converges to an element of F.

Definition 1 *[13, Definition 1] Let $\mathcal{F}_1$ be the set of continuous operators $\Gamma : R^n \to R^n$ that satisfy*

$$\|\Gamma(x) - \Gamma(y)\| \leq \|x - y\|, \textit{ for all } x, y \in R^n. \tag{32}$$

and

$$\begin{aligned} &\textit{If } \|\Gamma(x) - \Gamma(y)\| = \|x - y\|, \textit{ then} \\ &\quad \Gamma(x) - \Gamma(y) = x - y \textit{ and } \langle x - y, \Gamma(y) - y\rangle = 0. \end{aligned} \tag{33}$$

Definition 2 *[13, Definition 2] Let $\mathcal{F}_2$ be the set of operators $\Gamma \in \mathcal{F}_1$ with the property that for all $S \in R^{n \times n}$ the function $g : R^n \to R$ defined by $g(x) := \|x - S\Gamma(x)\|^2$ attains its unconstrained global minimum.*

Proposition 3 *The family $\{C_k\}_{k=0}^{\infty}$ with $C_k = S$, for all $k \geq 0$, where S is a symmetric, stochastic, with positive diagonal, matrix, is strictly nonexpansive.*

2.2.3 Solving The Linear Least Squares Problem

We will prove in the sequel that the general iterative method GEN [31], presented in Section 1.2, is strictly nonexpansive and, moreover, belongs to $\mathcal{F}_2$.

Proposition 4 *When A and b are as in (1) and the matrices Q, R and A have the properties (3)–(5), then the affine operator $\Gamma : R^n \to R^n$ defined by*

$$\Gamma(\cdot) := Q(\cdot) + Rb \tag{34}$$

belongs to $\mathcal{F}_2$.

Remark 5 *The FCA algorithm, with T_k as in (28), $C_k = I$, Γ as in (34) with Q, R as in (3)-(5) includes the Kaczmarz (see, e.g., [49]), Cimmino (see, e.g., [31]) and Diagonal Weighting (see, e.g., [42]) algorithms (for details and proofs of this statement see [43]). We prove in the following result that another such example is the Landweber method (see, e.g., [30, 43]).*

Proposition 5 *Let $\{\omega_k\}_{k=0}^{\infty} \subset R^n$ have the property that there exists a real ϵ such that $0 < \epsilon \leq \omega_k \leq \frac{2}{\rho(A)^2} - \epsilon$, where $\rho(A)$ denotes the spectral norm of A. For any $x^0 \in R^n$ and $k \geq 0$ the Landweber iteration is defined by*

$$x^{k+1} = (I - \omega_k A^T A)x^k + \omega_k A^T b. \tag{35}$$

If we denote $I - \omega_k A^T A$ by T_k and $\omega_k A^T$ by R_k, then, for every $k \geq 0$, the properties (3)-(5) hold.

Lemma 1 *Let $\mathrm{Fix}(\Gamma)$ be the fixed points set of the operator Γ defined by (34), with Q and R matrices of dimensions $n \times n$ and $n \times m$, respectively, having the properties (3)–(5). The following property then holds*

$$\mathrm{Fix}(\Gamma) = \{x + \Delta \mid x \in LSS(A; b)\}, \text{ with } \Delta = (I - \tilde{Q})^{-1} R P_{\mathcal{N}(A^T)}(b). \tag{36}$$

Lemma 2 *Let $\{C_k\}_{k=0}^{\infty}$ be a family of metric projection operators onto the k-th box $[a_k, b_k] \subset R^n$, as defined in (20) and assume that $\mathcal{V}_k^* \neq \emptyset$ for all $k \geq 0$. If for every $\ell \geq 0$ there exists a $k(\ell) \geq \ell$ such that it obeys (23), i.e.,*

$$Im(C_{k+1}) \subseteq Im(C_\ell), \text{ for all } k \geq k(\ell),$$

then the infinite intersection set

$$\mathcal{V}_\infty^* := \cap_{k=0}^{\infty} \mathcal{V}_k^*, \tag{37}$$

is not empty.

Proposition 6 *For a family* $\{C_k\}_{k=0}^{\infty}$ *of box constraining operators like those in (20) with the properties* $\mathcal{V}_k^* \neq \emptyset$ *and (23), and an operator* Γ *defined by (34), with* Q *and* R *matrices having the properties (3)–(5), the assumption (25) and Condition 1 are satisfied.*

In conclusion, according to Proposition 4, Lemma 1, Proposition 6 and Theorem 4, we may solve the linear least squares problem (1) using Algorithm 2.2.2 with Γ defined by (34), when Q and R matrices have the properties (3)–(5) and a family $\{C_k\}_{k=0}^{\infty}$ of box constraining operators like those in (20) satisfying the properties $\mathcal{V}_k^* \neq \emptyset$ and (23).

3 Weaker assumptions for convergence of extended block Kaczmarz and Jacobi projection algorithms

In this chapter we are interested to approximate a solution of the linear least squares problem (1), in the inconsistent case, using block-type Kaczmarz and Jacobi projection methods.

Extended block versions of these two algorithms were introduced in [41] and convergence was proved under nonsingularity assumptions on matrices resulted from row and column block decompositions.

In this chapter we show that, after replacing the inverse operator with the Moore-Penrose pseudoinverse, the earlier results on this topic, see [41, Theorem 3.4 and Theorem 6.7], still hold without the nonsingularity hypotheses.

3.1 Previous results on this topic

We consider block row decompositions of the matrix A and corresponding vector b. In this respect, let $p \geq 2$, $1 \leq m_i \leq m$, with $i \in \{1, 2, \ldots, p\}$, such that $m_1 + m_2 + \cdots + m_p = m$,

$$A = \begin{bmatrix} A_1 \\ A_2 \\ \vdots \\ A_p \end{bmatrix} \text{ and } b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_p \end{bmatrix}, \tag{38}$$

where A_i are $m_i \times n$ matrices and $b_i \in \mathbb{R}^{m_i}$.

Similarly, for $q \geq 2$ and $n = n_1 + n_2 + \cdots + n_q$, with $1 \leq n_j < n$ for any $j \in \{1, 2, \ldots, q\}$, the block column decomposition of A is given by

$$A^T = \begin{bmatrix} B_1 \\ B_2 \\ \vdots \\ B_q \end{bmatrix}, \tag{39}$$

where B_j are $n_j \times m$ real matrices.

In [41] the author introduced extended block versions of the Kaczmarz and Jacobi with relaxation parameters algorithms. They proved that under the hypotheses

$$\det\left(A_i A_i^T\right) \neq 0, \ \forall\, i \in \{1, 2, \ldots, p\} \tag{40}$$

and

$$\det\left(B_j B_j^T\right) \neq 0, \ \forall\, j \in \{1, 2, \ldots, q\} \tag{41}$$

these methods converge to an element of the linear least square solutions set of the problem (1).

Let the linear applications $f_0^i(b; \cdot), F_0(b; \cdot) : \mathbb{R}^n \to \mathbb{R}^n$ be defined by

$$f_0^i(b; x) = x + A_i^T (A_i A_i^T)^{-1} (b_i - A_i x), \ \forall\, i \in \{1, 2, \ldots, p\}, \tag{42}$$

$$F_0(b; x) = \left(f_0^1 \circ f_0^2 \circ \cdots \circ f_0^p\right)(b; x) \tag{43}$$

and the linear mapping

$$\Phi_0 = \prod_{j=1}^{q} \left(I - A_i^T (A_i A_i^T)^{-1} A_i\right). \tag{44}$$

The following algorithm was introduced in [41].

Extended Block Kaczmarz Algorithm(EBK)

Initialization: $x^0 \in R^n$ is arbitrary and $y^0 = b$.

Iterative step For every $k \geq 0$,

$$y^{k+1} = \Phi_0 y^k; \ b^{k+1} = b - y^{k+1}; \ x^{k+1} = F_0(b^{k+1}; x^k). \tag{45}$$

Now, for the real parameters $\omega, \alpha \neq 0$, consider

$$Q_0^\omega = I - \omega \sum_{i=1}^{p} A_i^T (A_i A_i^T)^{-1} A_i, \tag{46}$$

$$R_0^\omega = \omega \mathrm{col} \left[\; A_1^T(A_1A_1^T)^{-1} \mid A_2^T(A_2A_2^T)^{-1} \mid \; \ldots \; \mid A_p^T(A_pA_p^T)^{-1} \; \right] \tag{47}$$

and

$$\Phi_0^\alpha = I - \alpha \sum_{j=1}^{q} B_j^T(B_jB_j^T)^{-1}B_j. \tag{48}$$

An extended version of the Jacobi method was given in [41].

Extended Block Jacobi Algorithm with Relaxation Parameters (EBJRP)

Initialization: $x^0 \in R^n$ is arbitrary and $y^0 = b$.

Iterative step For every $k \geq 0$,

$$y^{k+1} = \Phi_0^\alpha y^k; \; b^{k+1} = b - y^{k+1}; \; x^{k+1} = Q_0^\omega x^k + R_0^\omega b^{k+1}. \tag{49}$$

Unfortunately, in real examples, conditions of the type (40)-(41) are usually not true or hard to verify. Following the considerations from [41], we will show that if we use the Moore-Penrose pseudoinverse rather than the inverse operator, the convergence results remain true without the assumptions (40)-(41). This is accomplished by proving that the two algorithms are particular cases of the extended general projection method EGEN [45], described in Section 1.2.

3.2 EBK and EBJRP algorithms as special cases of the EGEN method

In the rest of the paper we will denote by f^i, F, Φ, Q^ω, R^ω and Φ^α, the linear operators f_0^i, F_0, Φ_0, Q_0^ω, R_0^ω and Φ_0^α defined according to (42)-(44) and (46)-(48), respectively, in which we replaced the inverse mapping with the Moore-Penrose pseudoinverse.

For every $i \in \{1, 2, \ldots, p\}$ and $j \in \{1, 2, \ldots, q\}$, we define the matrices

$$P_i = I - A_i^T(A_iA_i^T)^\dagger A_i, \; \phi_j = I - B_j^T(B_jB_j^T)^\dagger B_j, \tag{50}$$

$$\bar{P}_i = A_i^T(A_iA_i^T)^\dagger A_i, \; \bar{\phi}_j = B_j^T(B_jB_j^T)^\dagger B_j. \tag{51}$$

We will now consider the corresponding linear mappings

$$\Delta_i = A_i^T(A_iA_i^T)^\dagger, \; Q_i = P_1P_2\ldots P_i, \; \forall \; i \in \{1, 2, \ldots, p\}, \tag{52}$$

$$Q = P_1P_2\ldots P_p, \tag{53}$$

$$R = \mathrm{col}\,[\ \Delta_1 \mid Q_1\Delta_2 \mid \ \ldots \ \mid Q_{p-1}\Delta_p\ ]\,, \tag{54}$$

$$\Phi_j = \phi_1\phi_2\ldots\phi_j,\ \forall\ j \in \{1, 2, \ldots, q\}, \tag{55}$$

$$\tilde{Q}^\omega = Q^\omega P_{\mathcal{R}}(A^T) \text{ and } \tilde{\Phi}^\alpha = \Phi^\alpha P_{\mathcal{R}}(A). \tag{56}$$

Consequently, we have

$$\Phi = \phi_1\phi_2\ldots\phi_q,\ Q^\omega = I - \omega\sum_{i=1}^{p}\bar{P}_i, \tag{57}$$

and

$$R^\omega = \omega\mathrm{col}\,[\ \Delta_1 \mid \Delta_2 \mid \ \ldots \ \mid \Delta_p\ ]\,,\ \Phi^\alpha = I - \alpha\sum_{j=1}^{q}\bar{\phi}_j. \tag{58}$$

The following theorem ensures the convergence of the extended block Kaczmarz algorithm defined using the Moore-Penrose pseudoinverse, without the assumptions (40)-(41).

Theorem 5 *If Q, R and Φ are linear applications defined according to (53), (54) and (57), respectively, then*
(i) we have the equality

$$F(b; x) = Qx + Rb, \tag{59}$$

(ii) Q and R satisfy (3)-(5),
(iii) for the matrix Φ the properties (10)-(12) hold.

In the case of the EBJRP algorithm we will confirm similar statements. The assumptions (5) and (12) will be proved using results from [15, 41].

Theorem 6 *The following properties are true*
(i) Q^ω and R^ω satisfy (3)-(4),
(ii) for the matrix Φ^α the assumptions (10)-(11) hold.

Corollary 1 *[41] If $0 < \omega < \frac{2}{\rho(E)}$ and $0 < \alpha < \frac{2}{\rho(D)}$, where $D = \frac{1}{\alpha}(I - \Phi^\alpha)$ and $E = \frac{1}{\omega}R^\omega A$, then $\tilde{Q}^\omega$ and $\tilde{\Phi}^\alpha$ satisfy the assumptions (5) and (12), respectively.*

From Theorem 6 and Corollary 1 it results that the EBJRP algorithm is a particular case of the EGEN method.

4 Supplementary projections for the acceleration of Kaczmarz algorithm

In the earlier work [45] the authors observed that the Extended Kaczmarz algorithm computes a solution of (1) in a single iteration, provided that the matrix A satisfies special hypotheses. Hence, they studied a procedure of transforming the original problem by adding an appropriate set of directions for projection, built as linear combinations of the rows and columns of the system matrix. Although the obtained extended matrices will verify the desired hypotheses, one disadvantage is that they will usuallly have a high fill-in percentage compared with the original matrix A.

Our purpose is to explore the possibility of adding a subset of supplementary directions for projection that keep the sparsity of the extended matrices near the initial level of A. Because in this way we will lose the direct solver property of the Kaczmarz algorithm we will try to get for it a better convergence speed.

4.1 The Direct Extended Kaczmarz algorithm

In [45], the authors introduced a Direct Extended Kaczmarz projection solver for (1) and proved is convergence in the inconsistent case. The idea is to transform the original problem by constructing new row and column directions for projections as follows. For all $i \in \{1, 2, \dots m\}$ and $j \in \{1, 2, \dots n\}$, let $P_i : \mathbb{R}^n \to \mathbb{R}^n$, $\phi_j, \mathbb{R}^m \to \mathbb{R}^m$, be the applications defined by $P_i(x) = x - \frac{x^T A_i}{\|A_i\|^2} A_i$, $\phi_j(x) = x - \frac{x^T A^j}{\|A^j\|^2} A^j$. Consider the sets of direction vectors $d_1, d_2, \dots d_{m-1} \in \mathbb{R}^n$ and $\delta_1, \delta_2, \dots, \delta_{n-1} \in \mathbb{R}^m$ defined ([45]) by

$$\begin{aligned} d_{m-1} &= P_m(A_{m-1}), \\ d_{m-2} &= P_{m-1} P^{d_{m-1}} P_m(A_{m-2}), \\ &\dots\dots \\ d_1 &= P_2 P^{d_2} P_3 \dots P_{m-1} P^{d_{m-1}} P_m(A_1), \end{aligned} \tag{60}$$

$$\begin{aligned} \delta_{n-1} &= \phi_n(A^{n-1}), \\ \delta_{n-2} &= \phi_{n-1} \phi^{\delta_{n-1}} \phi_n(A^{n-2}), \\ &\dots\dots \\ \delta_1 &= \phi_2 \phi^{\delta_2} \phi_3 \dots \phi_{n-1} \phi^{\delta_{n-1}} \phi_n(A^1), \end{aligned} \tag{61}$$

where the linear mappings $P^{d_i} : \mathbb{R}^n \to \mathbb{R}^n$ and $\phi^{\delta_j} : \mathbb{R}^m \to \mathbb{R}^m$ are defined by $P^{d_i}(x) = x - \frac{x^T d_i}{\|d_i\|^2} d_i$, $\phi^{\delta_j}(x) = x - \frac{x^T \delta_j}{\|\delta_j\|^2} \delta_j$. We then define the $(2m-1) \times n$ and $m \times (2n-1)$ matrices $\widehat{A}$ and $\bar{A}$ as follows

$$\widehat{A} = \text{col} \left[\; A_1 \mid d_1 \mid A_2 \mid d_2 \mid \; \dots \; \mid d_{m-1} \mid A_m \; \right]^T, \tag{62}$$

$$\bar{A} = \text{col} \left[\; A^1 \mid \delta^1 \mid \delta^2 \mid \delta^2 \mid \; \dots \; \mid \delta^{n-1} \mid A^n \; \right]. \tag{63}$$

For any $i \in \{1, 2, \dots m-1\}$, each direction d_i is (from its construction) a linear combination of the rows $A_i, A_{i+1}, \dots A_m$ (for more details see, e.g., [45]). Therefore, there exist scalars $\beta^i_i, \beta^i_{i+1}, \dots \beta^i_m \in \mathbb{R}$ such that $d_i = \sum_{k=i}^m \beta^i_k A_k$. We then set $b(d_i) = \sum_{k=i}^m \beta^i_k b_k$ and

$$\widehat{b} = (b_1, b(d_1), b_2, b(d_2), b_3, \dots b(d_{m-1}), b_m)^T. \tag{64}$$

Unfortunately, the DEK method, although direct, has the disadvantage that, even for a sparse matrix A, usually yields a large fill-in percentage of $\widehat{A}$ and $\bar{A}$.

4.2 Accelerating the convergence of the Kaczmarz method in the consistent case

In this section we focus our attention on finding a procedure of adding row directions for projection to (1) for which both the performance of the Kaczmarz iterative methods ([49, 40]) is improved and the sparsity of the new obtained matrix is conserved. The idea is to perform an agglomerative clustering of its rows and columns according to their sparsity similarity. This may be measured with one of the distances *Jaccard* or *Hamming* (for more details concerning these distances and a wide variety of usage examples see, e.g., [46]). We propose the following algorithm.

The Modified Kaczmarz Algorithm with Clustering (MKC)

Clustering. Using one of the distances *Jaccard* or *Hamming* compute a clustering of the matrix A rows with maximum nc clusters; we denote by m_k the number of rows in the k-th cluster and let $\{A^k_1, A^k_2, \dots A^k_{m_k}\}$ be its set of rows. Let the $m_k \times n$ and $m \times n$ matrices B_k and A^{nc}_c, respectively, defined by

$$B_k = \begin{pmatrix} (A^k_1)^T \\ (A^k_2)^T \\ \vdots \\ (A^k_{m_1})^T \end{pmatrix} \text{ and } A^{nc}_c = \begin{pmatrix} B_1 \\ B_2 \\ \vdots \\ B_{nc} \end{pmatrix}. \tag{65}$$

The matrix A_c^{nc} is obtained from a permutation of the original matrix rows. Apply the same permutation on b, to obtain b_c^{nc}. We have that $S(A;b) = S(A_c^{nc}; b_c^{nc})$.

Matrix transformation. For every $k \in \{1, 2, \dots nc\}$ compute the whole set of direction vectors $d_1^k, d_2^k, \dots d_{m_k-1}^k \in \mathbb{R}^n$ according to (60) for B_k to obtain the matrix $\widehat{B}^k$ according to (62). We will now define the new matrix.

$$\widehat{A_c^{nc}} = \begin{pmatrix} \widehat{B}^1 \\ \widehat{B}^2 \\ \vdots \\ \widehat{B}^{nc} \end{pmatrix}. \tag{66}$$

The corresponding vector $\widehat{b_c^{nc}}$ is calculated from b_c^{nc} as in (64) for each row direction added to A_c^{nc}.

Iterative Kaczmarz algorithm. Use the Kaczmarz algorithm to solve the problem (equivalent to (1)): find $x \in \mathbb{R}^n$ such that

$$\left\| \widehat{A_c^{nc}} x - \widehat{b_c^{nc}} \right\| = \min\{ \left\| \widehat{A_c^{nc}} z - \widehat{b_c^{nc}} \right\|, z \in \mathbb{R}^n \}.$$

Remark 6 *When $nc = m$, the MKC algorithm is actually the classical Kaczmarz iterative method applied to (1). For a single cluster the whole set of directions is added to the original matrix and the MKC algorithm becomes Direct Kaczmarz, which converges in a single iteration to a solution of (1)(for details see [45, Theorem 3]).*

4.3 An extended algorithm for the inconsistent case

In the inconsistent case we will apply the reasoning of the previous section. We will obtain independently two different transformations of the matrices A and A^T, denoted by $\widehat{A}_0$ and $\bar{A}_0$, and introduce a modified version of the Extended Kaczmarz (EK) algorithm. However, $\widehat{A}_0^T \neq \bar{A}_0$, and the new algorithm will not be equivalent to the EK method.

For arbitrarily fixed scalars $\beta_1, \beta_2, \dots, \beta_n \in \mathbb{R}$, $\alpha_1, \alpha_2, \dots, \alpha_n \in \mathbb{R}$ and the indices $i_0 \in \{2, 3, \dots, m\}$ and $j_0 \in \{2, 3, \dots, n\}$ we define the vectors d, δ by

$$d = \sum_{i=1}^{m} \beta_i A_i, \;\; \delta = \sum_{i=1}^{n} \alpha_i A^i \tag{67}$$

and the matrices of dimensions $(m+1)\times n$ and $m\times(n+1)$, respectively,

$$\widehat{A}_0^T = \mathrm{col}\,[A_1,\ldots,A_{i_0-1},d,A_{i_0},\ldots,A_m], \tag{68}$$

$$\bar{A}_0 = \mathrm{col}\left[(A^1),\ldots,(A^{j_0-1}),\delta,(A^{j_0}),\ldots,(A^n)\right]. \tag{69}$$

To any vector $z=(z_1,z_2,\ldots,z_m)^T\in\mathbb{R}^m$ we associate the vector $\widehat{z}\in\mathbb{R}^{m+1}$ given by

$$\widehat{z} = (z_1,z_2,\ldots,z_{i_0-1},\sum_{i=1}^m \beta_i z_i, z_{i_0},\ldots,z_m)^T. \tag{70}$$

Consider the linear applications $P^d:\mathbb{R}^n\to\mathbb{R}^n$, $\phi^\delta:\mathbb{R}^m\to\mathbb{R}^m$, $\widehat{Q}_0:\mathbb{R}^n\to\mathbb{R}^n$, $\bar{\Phi}_o:\mathbb{R}^m\to\mathbb{R}^m$ defined by $P^d(x)=x-\frac{x^Td}{\|d\|^2}d$, $\phi^\delta(x)=x-\frac{x^T\delta}{\|\delta\|^2}\delta$,

$$\widehat{Q}_0 = (P_1\circ\cdots\circ P_{i_0}\circ P^d\circ P_{i_0+1}\circ\cdots\circ P_m), \tag{71}$$

$$\bar{\Phi}_o = (\phi_1\circ\cdots\circ\phi_{j_0}\circ\phi^\delta\circ\phi_{j_0+1}\circ\cdots\circ\phi_n), \tag{72}$$

and the $n\times(m+1)$ matrix $\widehat{R}_0=\mathrm{col}\left[\widehat{R}_0^1,\widehat{R}_0^2,\ldots,\widehat{R}_0^{m+1}\right]$ defined by (see [45])

$$\begin{aligned}
\widehat{R}_0^k &= \frac{1}{\|A_k\|^2}P_1P_2\ldots P_{k-1}(A_k),\ \text{for all } k\in\{1,2,\ldots,i_0-1\},\\
\widehat{R}_0^{i_0} &= \frac{1}{\|d\|^2}P_1P_2\ldots P_{i_0-1}(d),\\
\widehat{R}_0^k &= \frac{1}{\|A_{k-1}\|^2}P_1P_2\ldots P_{i_0-1}P^d\ldots P_{k-2}(A_{k-1}),\ \text{for all } k\in\{i_0+1,\ldots,m\}.
\end{aligned} \tag{73}$$

The Extended Kaczmarz Modified Algorithm(EKM)

Initialization. $x^0\in R^n$ is arbitrary and $y^0=b$.

Iterative step. For every $k\geq 0$, compute

$$y^{k+1}=\bar{\Phi}_0(y^k);\ x^{k+1}=\widehat{Q}_0x^k+\widehat{R}_0(\widehat{b-y^{k+1}}). \tag{74}$$

Theorem 7 *The sequences $(x^k)_{k\geq0}$ and $(y^k)_{k\geq0}$ generated with the algorithm EKM converge and*

$$\lim_{k\to\infty}y^k=P_{\mathcal{N}(A^T)}(b),\ \lim_{x\to\infty}x^k=P_{\mathcal{N}(A)}(x^0)+x_{LS}. \tag{75}$$

Theorem 8 *For any $m \times n$ matrix A, any vector $b \in \mathbb{R}^m$, let the $(m+1) \times n$ matrix $\widehat{A}_0$ be defined as in (68), the vector $\widehat{b} \in \mathbb{R}^{(m+1)}$ as in (70) and the applications $\widehat{Q}_0 : \mathbb{R}^n \to \mathbb{R}^n$, $\widehat{R}_0 : \mathbb{R}^{(m+1)} \to \mathbb{R}^n$, $\bar{\Phi}_0 : \mathbb{R}^m \to \mathbb{R}^m$ be defined according to (71), (73), and (72), respectively. If*

$$\begin{aligned} &\widehat{Q}_0(A_i) = 0, \ \forall \ i \in \{1, 2, \ldots, m\}, \ \widehat{Q}_0(d) = 0, \\ &\bar{\Phi}_0(A^j) = 0, \ \forall \ j \in \{1, 2, \ldots, n\}, \ \bar{\Phi}_0(\delta) = 0, \end{aligned} \tag{76}$$

then the Extended Kaczmarz Modified Algorithm converges in one iteration to a solution of (1).

When (1) is inconsistent we propose the following algorithm

The Modified Extended Kaczmarz Algorithm with Clustering (MEKC)

Clustering. Perform agglomerative clustering on the columns of the matrix A with maximum ncc clusters using one of the distances *Jaccard* or *Hamming*. For any cluster k with $k \in \{1, 2, \ldots ncc\}$, let $A_k^1, A_k^2, \ldots A_k^{n_k}$ be its set of columns, where n_k is the size of the cluster. Let the $n_k \times m$ and $n \times m$ matrices B^k and A^c_{ncc}, respectively, defined by

$$B^k = \text{col}\left[A_k^1, A_k^2, \ldots, A_k^{n_k}\right], \tag{77}$$

$$A^c_{ncc} = \text{col}\left[B^1, B^2, \ldots, B^{nc}\right]. \tag{78}$$

The matrix A^c_{ncc} is obtained from a permutation of the original matrix columns. Using one of the distances *Jaccard* or *Hamming* compute a clustering of the matrix A rows with maximum ncr clusters to obtain the matrix A_c^{ncr} according to (65) and apply the same permutation on b, to obtain b_c^{ncr}.

Matrix transformations. For every $k \in \{1, 2, \ldots ncc\}$ construct the direction vectors $\delta_k^1, \delta_k^2, \ldots, \delta_k^{n_k} \in \mathbb{R}^n$ from (61) for the column block B^k and compute $\bar{B^k}$ as in (63). Consider the new matrix.

$$\bar{A}^c_{ncc} = \text{col}\left[\bar{B^1}, \bar{B^2}, \ldots, \bar{B^{ncc}}\right]. \tag{79}$$

Calculate for ncr clusters the matrix $\widehat{A_c^{ncr}}$ as in (66) and compute the corresponding $\widehat{b_c^{ncr}}$.

Extended Kaczmarz Modified algorithm. Calculate an approximation of a solution for (1) using EKM algorithm with respect to the new matrices $\bar{A}^c_{ncc}$ and $\widehat{A_c^{nc}}$.

5 Numerical experiments

5.1 Constraining techniques

In this section we try to develop and analyze an algorithm for solving problem (1) based on prior information about the exact solution. The purpose of our numerical experiments is solely to demonstrate that the use of a family of adaptively built constraining functions gives better results than the use of a single one. For this reason we study a simplified version of a real world problem. The problem in question is to reconstruct a solution vector, corresponding to the exact image, known to be only composed of zeros and ones. We concentrate on finding the number and the approximate location of the nonzero components. In this respect, we apply to the classical Kaczmarz projection method a family of constraining functions, defined in an adaptive manner.

5.1.1 Defining an adaptive version of the general algorithm CGENk

We define the following family constrained algorithm, where the operator $K : \mathbb{R}^n \longrightarrow \mathbb{R}^n$ stands for an iteration step of the Kacmzarz method.

Family Constrained Kaczmarz Algorithm (FCK)

Initialization. Let $x^0 = 0$, n_0 a given positive integer, ϵ_0 and δ two real positive numbers and C_k the metric projection operator onto the box $[a^k, b^k] = [a_1^k, b_1^k] \times \cdots \times [a_n^k, b_n^k] \subset \mathbb{R}^n$ (see (20)).

Step I. For every $k \in \{0, 1, \ldots, n_0 - 1\}$ let $[a^k, b^k] = [0, 1]^n$ and

$$x^{k+1} = C_{k+1}(Kx^k). \tag{80}$$

Step II. Let $x^0 = x^{n_0}$ and for every $k > n_0$ compute $\tilde{x}^k$ as

$$\tilde{x}^k = Kx^{k-1}. \tag{81}$$

For every $i \in \{1, 2, \ldots, n\}$, if $\tilde{x}_i^k \leq \epsilon_k$ then $[a_i^k, b_i^k] = [0, 10^{-4}]$; else $[a_i^k, b_i^k] = [a_i^{k-1}, b_i^{k-1}]$.

Then compute x^k as

$$x^k = C_k(\tilde{x}^k), \tag{82}$$

and update ϵ_k to ϵ_{k+1} by $\epsilon_{k+1} = \epsilon_k + \delta$.

The values of the parameters n_0, ϵ_0 and δ are chosen according to the reconstruction problem, through systematic tests. We built our family of constraining functions to satisfy condition (23). The algorithm is composed of two steps and starts with the null vector as an initial approximation. During the first step, for a fixed number of iterations, we use a constant constraining function, which is the metric projection operator onto the box $[a, b] = [0, 1]^n$. The idea of this step is to keep the approximation vector in the expected range and also to let the "ones" - points to "grow".

In the second step we consider the fact that, if after a sufficient number of iterations the value at a point is still small enough, it probably means that in the exact solution, its true value is zero. At each iteration we apply Kaczmarz and we "measure" the value of every pixel. If it is smaller than a given threshold ϵ_k, then the corresponding interval in the box will be set to $[0, 10^{-4}]$, such that the value of the pixel to be constrained to be very close to 0. We chose $[0, 10^{-4}]$ in favor of the trivial $[0, 0]$ because we observed experimentally that better results are obtained when the interior of the constraining intervals is not empty. In order to accelerate the convergence of the algorithm more components are assumed to be zero with every step. This is done by increasing the threshold with δ before the next iteration.

In the following two sections we study a particular image reconstruction inverse problem discussed in [35] and [34]. This problem arises from 3D Tomographic Particle Image Velocimetry (TomoPIV), which is an optical method for measuring velocities of fluids.

5.1.2 Solving an image reconstruction problem

5.1.3 Numerical results

5.2 Adding directions for projection

In the following section we investigate numerical results concerning the MKC and MEKC algorithms for two different sparse matrices.

5.2.1 Some case studies

5.2.2 Preserving data sparsity under a given threshold

When dealing with an inconsistent problem (1) and a large, ill-conditioned and sparse matrix A, usually iterative methods give a reduced time performance, while the direct algorithms lead to the loss of the sparsity property of the system. Using the MEKC algorithm, as the number of rows or columns clusters increases, the fill-in percentage decreases. However, the convergence rate of the iterative process also decreases. Therefore, the problem is to find small values for ncr and ncc such that the number of nonzero elements of $\widehat{A_c^{ncr}}$ and $\bar{A}^c_{ncc}$, respectively, is under a given threshold.

Let us consider that the linear least squares problem (1) is consistent. For a fixed distance and $threshold$ we are interested in finding a minimum ncr^* such that $\mathrm{nz}(\widehat{A^{ncr^*}}) < threshold$.

Solution. Our solution is to perform a binary search of ncr^* on the ordered sequence $\mathrm{nz}(\widehat{A}) \geq \mathrm{nz}(\widehat{A^2}) \geq \cdots \geq \mathrm{nz}(\widehat{A^{m-1}}) \geq \mathrm{nz}(A)$, where $\mathrm{nz}(M)$ denotes the number of nonzero elements of a matrix M:

– start with $ncr = \frac{m+1}{2}$ and compute $\mathrm{nz}(\widehat{A^{ncr}})$;

– if $\mathrm{nz}(\widehat{A^{ncr}}) > threshold$ continue the binary search in the right subsequence $\mathrm{nz}(\widehat{A^{ncr+1}}) \geq \cdots \geq \mathrm{nz}(\widehat{A^{m-1}}) \geq \mathrm{nz}(A)$;

– else, ncr is a candidate for ncr^* and continue search for a smaller one in $\mathrm{nz}(\widehat{A}) \geq \mathrm{nz}(\widehat{A^2}) \geq \cdots \geq \mathrm{nz}(\widehat{A^{ncr-1}})$;

– stop when reached a void subsequence; ncr^* is the last found number of clusters for which $\mathrm{nz}(\widehat{A^{ncr}}) < threshold$.

Remark 7 *At each step of the binary search we compute the sequence of directions* **only** *until the threshold is reached.*

After finding ncr^* we want to compare the classical K algorithm with our MKC($dist$, ncr^*).

Solution. Evaluate the number of flops (floating-point additions, subtractions, multiplications, or divisions) in these algorithms.

Method	Number of flops
Direct Kaczmarz	$3 * \mathrm{nz}(\widehat{A}) + 2 * (2 * m - 1)$
MKC($dist$, ncr^*)	$\Big(3 * \mathrm{nz}(\widehat{A^{ncr^*}}) + 2 * (2 * m - ncr^*)\Big) * niter_{ncr^*}$
Kaczmarz	$(3 * \mathrm{nz}(A) + 2 * m) * niter_K$

Our results show that when sparsity threshold constraints prevent us from applying Direct Kaczmarz, the (MKC) algorithm provides a more efficient alternative to the classical Kaczmarz method. The same concepts may also be applied in the inconsistent case.

References

[1] A. Aleyner and Y. Censor. Best approximation to common fixed points of a semigroup of nonexpansive operators. *J. Nonlinear Convex Anal.*, 6:137–151, 2005.

[2] H. Bauschke. The approximation of fixed points of compositions of nonexpansive mappings in hilbert space. *J. Math. Anal. Appl.*, 202: 150–159, 1996.

[3] H. Bauschke and J. Borwein. On projection algorithms for solving convex feasibility problems. *SIAM Rev.*, 38:367–426, 1996.

[4] H. Bauschke and P.L. Combettes. *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Springer, New York, 2011.

[5] A. Ben-Israel and T. Greville. *Generalized Inverses: Theory and Applications, Second Edition*. Springer-Verlag, New York, 2003.

[6] A. Björk. *Numerical Methods for Least Squares Problems*. SIAM, Philadelphia, 1996.

[7] C. Byrne. A unified treatment of some iterative algorithms in signal processing and image reconstruction. *Inverse Probl.*, 20:103–120, 2004.

[8] A. Cegielski. *Iterative Methods for Fixed Point Problems in Hilbert Spaces, Lecture Notes in mathematics 2057*. Springer-Verlag, Berlin, Heidelberg, 2012.

[9] Y. Censor and A.Z. Stavros. *Parallel optimization: theory, algorithms and applications, "Numer. Math. and Sci. Comp." Series*. Oxford Univ. Press, New York, 1997.

[10] Y. Censor, **I. Pantelimon**, and C. Popa. Family constraining of iterative algorithms. *Numerical Algorithms*, pages 1–16, 2013. doi: 10.1007/s11075-013-9736-5.

[11] C. Chaux, J.C. Pesquet, and N. Pustelnik. Nested iterative algorithms for convex constrained image recovery problems. *SIAM J. Imaging Sci.*, 2:730–762, 2009.

[12] P.L. Combettes. *Fejér-monotonicity in convex optimization*, volume 2, pages 106–114. Kluwer, Boston, 2001.

[13] A.R. De Pierro and A. Iusem. On the asymptotic behavior of some alternate smoothing series expansion iterative methods. *Linear Algebra Appl.*, 130:3–24, 1990.

[14] B. Eicke. Iteration methods for convexly constrained ill-posed problems in hilbert space. *Numer. Funct. Anal. Optim.*, 13:413–429, 2007.

[15] T. Elfing. Block-iterative methods for consistent and inconsistent linear equations. *Numer. Math.*, 35:1–12, 1980.

[16] T. Elfing. A stationary iterative pseudoinverse algorithm. *BIT*, 38(2): 275–282, 1998.

[17] T. Elfing, T. Nikazad, and P.C. Hansen. Semi-convergence and relaxation parameters for a class of sirt algorithms. *Electron. Trans. Numer. Anal.*, 37:321–336, 2010.

[18] L. Elsner, I. Koltracht, and M. Neumann. Convergence of sequential and asynchronous nonlinear paracontractions. *Numer. Math.*, 62:305–319, 1992.

[19] G. Gan, C. Ma, and J. Wu. *Data Clustering: Theory, Algorithms and Applications*. Society for Industrial and Applied Mathematics, Philadelphia, 2007.

[20] G.T. Herman. *Image Reconstruction from Projections. The Fundamentals of Computerized Tomography*. Academic Press, New York, 1980.

[21] G.T. Herman. *The Fundamentals of Computerized Tomography: Image Reconstruction from Projections, 2nd edition*. Springer-Verlag, London, UK, 2009.

[22] S.A. Hirstoaga. Iterative selection methods for common fixed point problems. *J. Math. Anal. Appl*, 324:1020–1035, 2006.

[23] R.A. Horn and C.R. Johnson. *Matrix Analysis*. Cambridge University Press, New York, NY, USA, 1990.

[24] A. Katsaggelos, J. Biemond, R. Mersereau, and R. Schafer. A general formulation of constrained iterative restoration algorithms. *IEEE Int. Conf. on Acoustics, Speech, and Signal Process.*, 10:700–703, 1985.

[25] A.K. Katsaggelos, J. Biemond, R. Mersereau, and R. Schafer. A regularized iterative image restoration algorithm. *IEEE Trans. Signal Process.*, 39:914–929, 1991.

[26] S. Kawata and O. Nalcioglu. Constrained iterative reconstruction by the conjugate gradient method. *IEEE Trans. on Med. Imaging*, 4:65–71, 1985.

[27] K.C. Kiwiel and B. Ł opuch. Surrogate projection methods for finding fixed points of firmly nonexpansive mappings. *SIAM J. Optim.*, 7:1084–1102, 1997.

[28] I. Koltracht and P. Lancaster. Constraining strategies for linear iterative processes. *IMA J. Numer. Anal.*, 10:555–567, 1990.

[29] I. Koltracht, P. Lancaster, and D. Smith. The structure of some matrices arising in tomography. *Linear Algebra Appl.*, 130:193–218, 1990.

[30] L. Landweber. An iterative formula for fredholm integral equations of the first kind. *Amer. J. Math.*, 73:615–624, 1951.

[31] A. Nicola, S. Petra, C. Popa, and C. Schnörr. On a general extending and constraining procedure for linear iterative methods. *Int. J. Comput. Math.*, 89:231–253, 2012.

[32] J.M. Ortega and W.C. Rheinboldt. *Iterative Solution of Nonlinear Equations in Several variables.* Academic Press, New York, NY, USA, 1970.

[33] **I. Pantelimon** and C. Popa. Constraining by a family of strictly nonexpansive idempotent functions with applications in image reconstruction. *BIT Numerical Mathematics*, 53(2):527–544, 2013. doi: 10.1007/s10543-012-0414-0.

[34] S. Petra and C. Schnörr. TomoPIV meets compressed sensing. *Pure Math. Appl.*, 20:49–76, 2009.

[35] S. Petra, C. Popa, and C. Schnörr. Accelerating constrained sirt with applications in tomographic particle image reconstruction, 2009. URL `http://www.ub.uni-heidelberg.de/archiv/9477`. Preprint.

[36] **I. Pomparău**. On the acceleration of Kaczmarz projection algorithm. Presented at the 84th Annual Meeting of the International Association of Applied Mathematics and Mechanics, 18–22 March, 2013, Novi Sad, Serbia. Submitted for publication in *Proceedings in Applied Mathematics and Mechanics*, 2013.

[37] **I. Pomparău**. On an accelerated version of the Kaczmarz algorithm with a priori clustering. In *Topics in mathematical modelling of life science problems*, pages 61–76. Matrix Rom, Bucureşti, 2013. Presented at the Nine-th Workshop on Mathematical Modeling of Environmental and Life Sciences Problems, 1–4 November, 2012, Constanţa, România.

[38] **I. Pomparău** and C. Popa. Supplementary projections for the acceleration of Kaczmarz algorithm. Submitted for publication, 2013.

[39] **I. Pomparău** and C. Popa. Weaker assumptions for convergence of extended block Kaczmarz and Jacobi projection algorithms. Submitted for publication, 2013.

[40] C. Popa. Least-squares solution of overdetermined inconsistent linear systems using Kaczmarz relaxation. *Intern. J. Comp. Math.*, 55:79–89, 1995.

[41] C. Popa. Extensions of block-projections methods with relaxation parameters to inconsistent and rank-deficient least-squares problems. *BIT*, 38(1):151–176, 1998.

[42] C. Popa. Extended and constrained diagonal weighting algorithm with application to inverse problems in image reconstruction. *Inverse Probl.*, 26, 2010. doi: 10.1088/0266-5611/26/6/065004.

[43] C. Popa. *Projection algorithms - classical results and developments. Applications to image reconstruction.* Lambert Academic Publishing - AV Akademikerverlag GmbH & Co. KG, Saarbrücken, Germany, 2012.

[44] C. Popa and R. Zdunek. Kaczmarz extended algorithm for tomographic image reconstruction from limited-data. *Math. Comput. Simulation*, 65: 579–598, 2004.

[45] C. Popa, T. Preclik, H. Köstler, and U. Rüde. On Kaczmarz's projection iteration as a direct solver for linear least squares problems. *Linear Algebra and its Applications*, 436:389–404, 2012.

[46] A. Rajaraman and J.D. Ullman. *Mining of Massive Datasets*. Cambridge University Press, New York, 2012.

[47] R.W. Schafer, R.M. Mersereau, and M.A. Richards. Constrained iterative restoration algorithms. *Proc. of the IEEE*, 69:432–450, 1981.

[48] G.F. Simmons. *Introduction to Topology and Modern Analysis*. McGraw-Hill Book Company, New York, NY, USA, 1963.

[49] K. Tanabe. Projection method for solving a singular system of linear equations and its applications. *Numer. Math.*, 17:203–214, 1971.

[50] D.C. Youla and H. Webb. Image restoration by the method of convex projections: Part 1–theory. *IEEE Trans. on Med. Imaging*, 1:81–94, 1982.