Math. & Comput. Sci., 6(2) 2025, 112–124

Online: ISSN 2717-2708

Mathematics and Computational Sciences

Journal Homepage: http://mcs.qut.ac.ir/mcs

An overview of iterative methods based on orthogonal projections

Touraj Nikazad[a,*], Mona Khakzad[a]

[a] *Applied Mathematics Department, School of Mathematics and computer Sciences, Iran University of Science and Technology, Tehran, Iran.*

Abstract

This paper investigates the linear feasibility problem (LFP), which plays a fundamental role in image reconstruction, especially in applications such as computed tomography and signal processing. The goal is to find a point in the intersection of a finite collection of convex sets defined by linear constraints. We provide a structured overview and comparison of existing orthogonal projection-based iterative methods for solving LFPs, including sequential, simultaneous, and block-iterative algorithms. While these methods have been studied individually in the literature, our work highlights their theoretical underpinnings, practical performance, and convergence properties in a unified framework. We also revisit and refine known convergence theorems, discussing their assumptions and implications in the context of real-world reconstruction problems. The novelty of this study lies in its comprehensive synthesis of algorithmic strategies along with a critical analysis of their relative strengths, limitations, and applicability. This work aims to clarify the landscape of projection methods for LFPs and to guide the selection or development of more effective reconstruction techniques in practice.

Keywords: Convex feasibility problem, Projections, Iterative methods.

2020 MSC: 33C45, 49-XX

1. Introduction

Many problems can be modeled as the following system of linear equations

$$Ax = b, \tag{1.1}$$

where $b \in \mathbb{R}^m$ and $A \in \mathbb{R}^{m \times n}$. For example, computerized tomography and magnetic resonance imaging (MRI), which are two widely used techniques in medical imaging, can be modeled using equation (1.1)[19, 25, 31]. The main problem in image reconstruction is to reconstruct the image x from the observed data b. The use of direct methods is practically impossible due to the ill-posed nature of the problem and the high-dimensional, noisy data [10, 23]. For this reason, iterative methods, such as the Algebraic Reconstruction Technique (ART), were developed for image reconstruction [5].

*Corresponding author

Email addresses: tnikazad@iust.ac.ir (Touraj Nikazad), khakzad.mona@yahoo.com (Mona Khakzad)

doi: 10.30511/mcs.2025.2050369.1288

Received: 10 January 2025 Accepted: 21 June 2025

Due to the practical challenges in X-ray computerized tomography imaging of the human body, the demand for obtaining an acceptable approximate solution in a short time has increased. Iterative algorithms have gained importance due to their excellent performance in many cases [11, 34]. In recent years, iterative algorithms for solving the linear system (1.1), which is a special case of the Convex Feasibility Problem (CFP), have been developed [2, 11].

In the field of medical imaging, the first row-action iterative method used in CT imaging was the ART method [15, 22], which has recently been applied to MRI as well [18, 31]. In terms of classification, the ART method belongs to the category of sequential algorithms. The Cimmino algorithm is an example of a synchronous iterative algorithms [12] developed for solving linear systems. The Component Averaging (CAV) algorithm is one of the most well-known synchronous iterative methods. To improve the speed of convergence, this method uses oblique projections instead of orthogonal projections [9].

Block iterative methods have been extensively studied in various research works with different applications, such as [1, 3, 7, 8, 28, 29]. In block methods, the problem is first partitioned into several blocks. This approach divides the problem into smaller sub-problems, making it highly suitable for problems with many convex sets. Block algorithms are generally categorized into sequential block and synchronous block methods. For example, the algorithm proposed by Aharoni and Censor is a sequential block algorithm in which the Cimmino algorithm is applied within each block [1]. In [13], the algorithm proposed by Elfving is an example of a synchronous block algorithm.

In this paper, we provide a structured overview of iterative methods based on orthogonal projections, which are among the most fundamental and widely used approaches for solving the convex feasibility problem (CFP). While many such methods exist, their classification, practical differences, and comparative strengths are often scattered across the literature. Our goal is to offer a clear and concise presentation of these methods, emphasizing their underlying principles, convergence behavior, and suitability for different problem settings, particularly in large-scale applications such as medical imaging. By organizing the methods thematically and highlighting their advantages and limitations, we aim to support researchers and practitioners in selecting appropriate algorithms for their needs. The paper is organized as follows: In Section 2, we introduce the convex and linear feasibility problems, with a focus on medical imaging as a key application. Section 3 presents the definitions and theoretical background on orthogonal projections, followed by a detailed discussion of various iterative methods and their characteristics. The final section concludes the paper.

We use the following notations throughout the paper. Let $\langle x, y\rangle$ denote the Euclidean inner product and $\|x\|$ the corresponding Euclidean norm. The Frobenius norm of a matrix A is denoted by $\|A\|_F$. We use $A^\dagger$, $R(A)$, and $\rho(A)$ to denote the MoorePenrose inverse, the range (column space) of a matrix A, and the spectral radius of A, respectively. The identity matrix of appropriate size is denoted by I. The generalized scaled condition number is defined as $\kappa(A) = \|A\|_F\|A^\dagger\|$.

2. Convex feasibility problem and its applications

An important problem in mathematics and physical sciences is finding a point in the intersection of several closed and convex sets. This problem is known as the convex feasibility problem (CFP).

Definition 2.1. Let X be a Hilbert space, and let $C_m, \cdots, C_2, C_1$ be closed and convex subsets of X, such that $C = \bigcap_{i=1}^m C_i \neq \emptyset$. The problem of finding a point $x^* \in X$ such that $x^* \in C$ is called the convex feasibility problem.

This problem has a long and rich history in applied mathematics, dating back at least to the 19th century. Comprehensive information about the CFP can be found in references [2] and [6]. The CFP has numerous applications in mathematics and physical sciences, including medical imaging and radiotherapy (tomography), electron microscopy, signal processing, minimizing nonsmooth convex functions, statistics (linear prediction theory), partial differential equations (e.g., the Dirichlet problem), and more.

A specific and very important special case of the CFP is presented as follows

$$C_i = \{x \in \mathbb{R}^n | g_i(x) \leqslant 0\} \quad (2.1)$$

where g_i are convex functions. If in (2.1) all the functions g_i's are hyperplanes, i.e., $g_i(x) = \langle x, a_i \rangle - b_i$, where a_i is the i-th row of the matrix A in (1.1), then the resulting problem is called the linear feasibility problem (LFP). It can be defined in matrix form as follows.

Definition 2.2. Let $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$. The problem of finding $x^* \in \mathbb{R}^n$ such that $Ax^* \leqslant b$ is called the linear feasibility problem.

In such problems, the matrix A is often very large, sparse, and ill-conditioned. One important application of the LFP is in medical imaging, which we briefly describe here. When the size or the hardness of a tumor inside a person's body needs to be measured, a cross-section of the body containing the tumor (or a region suspected of containing the tumor) is considered and divided into small squares called pixels. These pixels can be numbered in different ways. For example, the pixel in the top-left corner is numbered as the first pixel, and the pixel in the bottom-right corner is numbered as the n-th pixel. Then, a set of m X-ray beams, each at specific angles, is projected onto this cross-section, such that the length each beam travels inside each pixel is known. Each time, a detector measures the total attenuation of the i-th X-ray beam. Let the attenuation of the i-th beam be denoted by b_i. The attenuation of the beam can be considered as a line integral of the beam's passage through the cross-section.

However, if we assume that the attenuation of the i-th beam inside the j-th pixel is a constant proportional to the length the i-th beam travels through the j-th pixel, and this constant is denoted by x_j, representing the thickness or density of the j-th pixel, with the length travelled by the i-th beam through the j-th pixel denoted by a_j^i, we arrive at a set of equations as follows

$$\sum_{j=1}^{n} a_j^i x_j = b_i, \; i = 1, \ldots, m, \quad (2.2)$$

which is a problem of the LFP type [17, 23]. Figure (1) clearly illustrates this. It should be noted that the measured data, due to errors arising from measurements and calculations, are often perturbed. Given that each time an X-ray is projected onto the object, the beam passes through only a few pixels of the object, the resulting system is sparse and often has small singular values. All these factors lead to the system being ill-conditioned in practice. In the next section, we will discuss some of the existing iterative methods for solving this system, along with their advantages and disadvantages.

3. Orthogonal projections and their applications in CFP

One of the main methods for solving the CFP problem is using iterative methods based on orthogonal projections.

Definition 3.1. Let $C \subseteq \mathbb{R}^n$ and $x \in \mathbb{R}^n$ be an arbitrary point. If there exists a point $y \in C$ such that

$$\|y - x\| \leqslant \|z - x\|, \forall z \in C, \quad (3.1)$$

then the point y is called the orthogonal projection of x onto C, denoted as $P_C(x)$. See Figure 2.

Theorem 3.2. *If C is a closed and convex set, then for every $x \in \mathbb{R}^n$, the orthogonal projection of x onto C exists and is unique. [4].*

Remark 3.3. The orthogonal projection of any $z \in \mathbb{R}^n$ onto the hyperplane $H := \{x \in \mathbb{R}^n | \langle a, x \rangle = b\}$ where $a \in \mathbb{R}^n \setminus \{0\}$ and $b \in \mathbb{R}$ is given by:

$$p_H(z) = z + \frac{b - \langle a, z \rangle}{\|a\|^2} a. \quad (3.2)$$

To ensure that H is a hyperplane, we must consider $a \neq 0$. Therefore, we always assume that the matrix A in (1.1) has no zero rows.

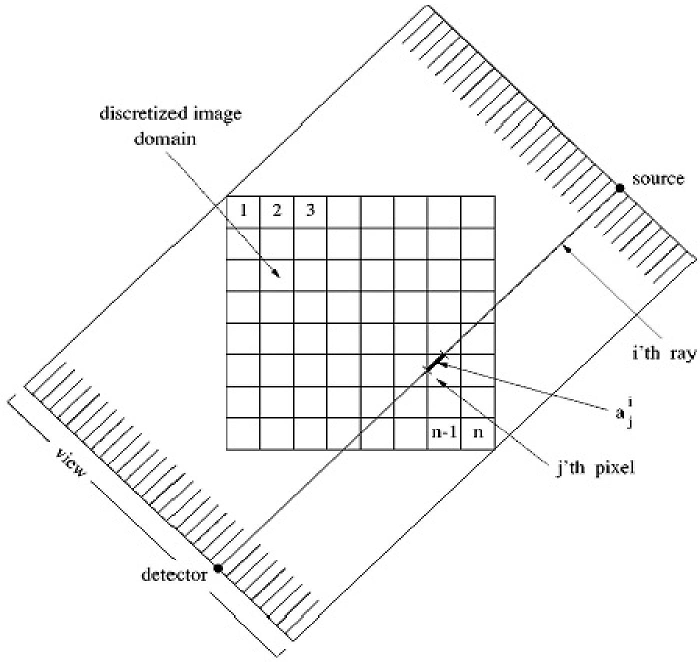

Figure 1: The fully discretized model of the image reconstruction problem.

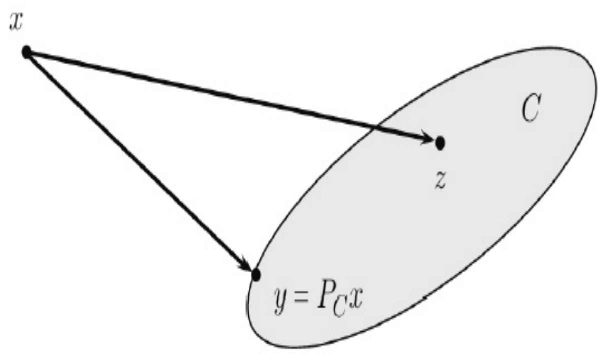

Figure 2: Orthogonal projection of points onto convex sets, showing perpendicular distances to the closest points.

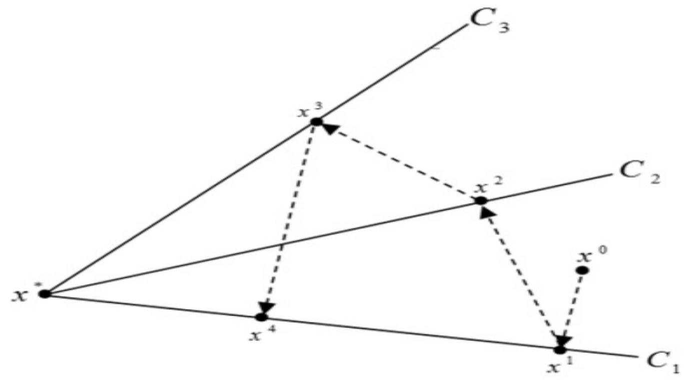

Figure 3: Illustration of the Algebraic Reconstruction Technique (ART), where iterative projections onto hyperplanes corresponding to linear equations gradually refine the solution.

Orthogonal projection operators are a type of operator widely used in solving convex feasibility problems. The use of orthogonal projection operators to solve convex feasibility problems involves considering an operator P_{C_i} for each set C_i in the definition of CFP, and combining these operators in various ways to create an algorithm for solving the convex feasibility problem. In general, the algorithms used to solve the CFP problem are iterative methods, which are classified into three categories: sequential, simultaneous, and block-iterative [11].

3.1. Sequential methods

In sequential algorithms, at each step, we only use one of the convex sets, and then move on to the next convex set. For example, the ART algorithm is a type of sequential algorithm. In this algorithm, we obtain the orthogonal projection of the initial point x^0 onto the set C_1 and call it x^1. Then, we obtain the orthogonal projection of x^1 onto the set C_2 and call it x^2, and so on. The orthogonal projection of the point x^{m-1} onto the set C_m is obtained, and we call it x^m. At this point, one iteration of the algorithm ends, and we again obtain the orthogonal projection of the point x^m onto the set C_1 and call it x^{m+1}, continuing in this manner. Sequential algorithms are also known as algebraic reconstruction techniques (ART). Figure 3 illustrates the operation of the ART algorithm for a convex feasibility problem with three sets [16, 22]. The ART algorithm for a system of linear equations is defined as algorithm 1 [26]. The ART

Algorithm 1 ART method for linear system of equations.

Given x^k, compute
$x^{k,0} = x^k$
for $i = 1, \cdots, m$ **do**
$\quad x^{k,j} = x^{k,j-1} + \lambda_k \frac{b_j - \langle a_j, x^{k,j-1} \rangle}{\|a_j\|^2} a_j$
end for
$x^{k+1} = x^{k,m}$.

(Here a_j and b_j denote j-th row A and b, respectively. The sequence $\{\lambda_k\}$ are relaxation parameters in $(0, 2)$.)

algorithm fall under the category of the Kaczmarz and Randomized Kaczmarz methods. In the following, we introduce these two methods.

3.1.1. Kaczmarz

In this subsection and the following one, we introduce and use two notations, x^* and $x^\dagger$, which are defined as follows. When system (1.1) is consistent, let x^* denote an exact solution. Since this may not always be the case, we instead consider the generalized solution $x^\dagger$, also known as the minimum-norm least-squares solution. The Kaczmarz method is one of the most well-known techniques for solving linear systems of the form $Ax = b$. This method was introduced in 1937 by Kaczmarz and was later applied in 1970 by Gordon, Bender, and Herman to problems in computerized tomography. The Kaczmarz algorithm is described as follows

$$x_k = x_{k-1} + \frac{b_i - \langle a_i, x_{k-1}\rangle}{\|a_i\|^2} a_i, \ k = 1, 2, \cdots., \tag{3.3}$$

where $i = (k \bmod m) + 1$, and the symbol $\|.\|$ represents the Euclidean norm $\mathbb{R}^n$.

In 1971, Kunio Tanabe established the convergence theory of the Kaczmarz method for consistent linear systems and related the solution of $Ax = b$ to the iterations of the Kaczmarz method. Kunio Tanabe stated the convergence theorem of the Kaczmarz method as follows:

Theorem 3.4. *[33] For an $m \times n$ matrix A with nonzero rows, and for any m-dimensional column vector b, suppose $\{x_k\}$ is a sequence generated by (3.3), and $\{y_j\}$ is a subsequence of $\{x_k\}$ such that $k - j * m = 0 (k = 0, 1, \cdots)$, then $\{y_j\}$ converges and*

$$\lim_{i\to\infty} y_j = P_{N(A)} y_0 + A^\dagger b,$$

where $y_0 = x_0 \in \mathbb{C}^n$ is an arbitrary point.

In 2021, Kang and Zhou introduced a new error expression for the Kaczmarz method [24] and analyzed the behavior of this new expression. They also derived the convergence rate for Kaczmarz-type methods. In particular, using the new error expression, they provided a simpler proof of the convergence of the Kaczmarz method.

Theorem 3.5. *[24] Suppose $Ax = b$ is consistent, and the sequence of vectors $\{x_k\}_{k=1}^m$ is generated by the Kaczmarz iteration (3.3). Then, for any initial vector $x_0 \in \mathbb{R}^n$, the following relation holds*

$$\|x_{k+1} - P_{N(A)}x_0 - x^\dagger\|^2 \leqslant \left(1 - \frac{1}{\|a_{k+1}\|^2 \|(a_{k+1}^T P_{k+1})^\dagger\|^2}\right) \|x_k - P_{N(A)}x_0 - x^\dagger\|^2,$$

where $\|(a_{k+1}^T P_{k+1})^\dagger\| \geqslant \|A^\dagger\|_2$.

Corollary 3.6. *[24] Suppose $Ax = b$ is consistent, and the sequence of vectors $\{x_k\}_{k=1}^\infty$ is generated by the classical Kaczmarz iteration (3.3). Then, for any initial vector $x_0 \in \mathbb{R}^n$, the following relation holds*

$$\|x_{k+1} - P_{N(A)}x_0 - x^\dagger\|^2 \leqslant \left(1 - \frac{1}{max_{i=1,2,\ldots,m} \|a_i\|^2 \|(a_i^T P_i)^\dagger\|^2}\right)^{k+1} \|x_0 - P_{N(A)}x_0 - x^\dagger\|^2.$$

The result in 3.6 shows that the sequence and subsequence of vectors $\{x_k\}$ converge to $P_{N(A)}x_0 + x^\dagger$, which is consistent with the results in [33].

3.1.2. Randomized Kaczmarz

The randomized Kaczmarz iteration can be described as follows

$$x_k = x_{k-1} + \frac{b_i - (a_i, x_{k-1})}{\|a_i\|^2} a_i, \quad k = 1, 2, \ldots, \tag{3.4}$$

where i is chosen from the set $\{1, 2, \cdots, m\}$ with probability $p_r = \frac{\|a_i\|^2}{\|A\|_F^2}$. The probability p_r used in the randomized Kaczmarz method is chosen to accelerate convergence of the iteration by prioritizing more informative rows of the matrix A. Each row a_i of the matrix corresponds to a hyperplane in the solution space. Rows with larger norm (i.e., $\|a_i\|$) define hyperplanes that are steeper or more influential in shaping the geometry of the feasible region. By choosing them more often, the algorithm makes faster progress toward the solution. The following two convergence theorems provide upper bounds for the error in both the consistent and inconsistent cases; see [24] for further details.

Theorem 3.7. *Suppose $Ax = b$ is consistent, and $\{x_k\}$ is the sequence of vectors generated by the randomized Kaczmarz iteration (3.4). Then, the following expressions hold:*

- *a) For exact data:*

$$E\left[\|x_{k+1} - x^*\|^2\right] \leqslant (1 - \kappa(A)^{-2})^{k+1}\|x_0 - x^*\|^2$$

- *b) For noisy data:*

$$E\left[\|x_{k+1} - x^*\|^2\right] \leqslant (1 - \kappa(A)^{-2})^{k+1}\|x_0 - x^*\|^2 + \frac{\delta^2}{\sigma^2_{min}(A)}$$

(Here, σ_{min} is the smallest singular value of A.)

Theorem 3.8. *Suppose $\{x_k\}$ is the sequence of vectors generated by the randomized Kaczmarz iteration (3.4). For any initial vector $x_0 \in R^n$, the following relation holds:*

$$E\left[\|x_{k+1} - P_{N(A)}x_0 - x^\dagger\|^2\right] \leqslant (1 - \frac{1}{\kappa^2(A)})^{k+1}\|x_0 - P_{N(A)}x_0 - x^\dagger\|^2 + (1 - (1 - \frac{1}{\kappa^2(A)})^{k+1})\|A^\dagger\|^2\|(I - P_{R(A)})b\|^2. \tag{3.5}$$

In (3.5), if $b \in R(A)$, then $\|(I - P_{R(A)})b\| \equiv 0$, and the following relation holds:

$$E\left[\|x_{k+1} - P_{N(A)}x_0 - x^\dagger\|^2\right] \leqslant (1 - \frac{1}{\kappa^2(A)})^{k+1}\|x_0 - P_{N(A)}x_0 - x^\dagger\|^2. \tag{3.6}$$

As $k \to \infty$, $(1 - \frac{1}{\kappa^2(A)})^{k+1}$ approaches zero. Therefore, $\|x_{k+1} - P_{N(A)}x_0 - x^\dagger\|^2 \to 0$, in other words,

$$x_{k+1} \to P_{N(A)}x_0 + x^\dagger. \tag{3.7}$$

This shows that the sequence of vectors $\{x_k\}_{k=1}^\infty$ generated by the randomized Kaczmarz method converges to $P_{N(A)}x_0 + x^\dagger$ when $Ax = b$ is consistent. Furthermore, from Theorem 3.8, it is easily concluded that when $b \notin R(A)$ (inconsistent case), the sequence $\{x_k\}$ is a bounded sequence. In fact, Constantin Popa [30] pointed out that the Kaczmarz algorithm (classic version) with $x_0 = 0$ produces a sequence $\{x_k\}$ that converges to $x^\dagger$ if and only if $Ax = b$ is consistent. The convergence behavior of the randomized Kaczmarz method is essentially the same as that of the classical Kaczmarz method. Theorem 3.8 shows that the Kaczmarz and randomized Kaczmarz iterations do not converge to $x^\dagger$ for solving an inconsistent system, even when $x_0 = 0$.

3.2. Simultaneous methods

In these types of algorithms, at each step, all the sets C_i are used simultaneously. When moving from the current solution x^k to the next solution x^{k+1}, the point x^k is projected onto each of the sets C_i, and then the convex combination of these projections is calculated, which is called x^{k+1}. One example of such simultaneous algorithms is the Cimmino algorithm [12], which is used for solving linear systems. Instead of orthogonal projections, oblique projections can also be used. The component averaging (CAV) algorithm is one of the most well-known iterative methods where oblique projections are used [9]. Figure 4 shows how iterations are performed in simultaneous algorithms. In this figure, there are three convex sets, and $x^{k,n}$ denotes $P_{C_n}(x^k)$.

Now, consider system of linear equations (1.1) that results from discretizing an ill-posed problem. Suppose $\{\lambda_k\}_{k\geqslant 0}$ are the relaxation parameters and M is a given symmetric positive definite matrix. The

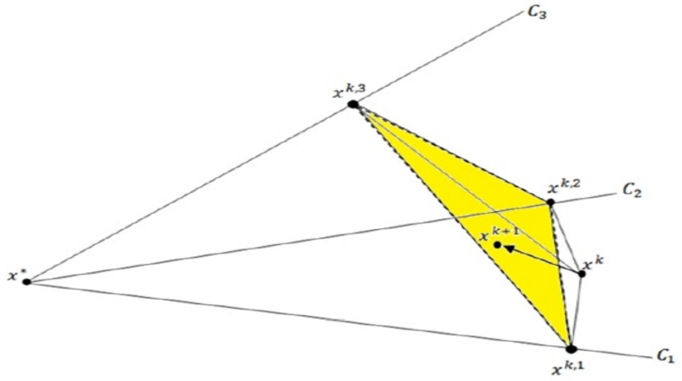

Figure 4: Visualization of Cimminos method, a fully simultaneous projection approach where updates are based on weighted averages of projections onto all hyperplanes.

Algorithm 2 Fully simultaneous

Given x^k, compute
$x^{k+1} = x^k + \lambda_k A^T M(b - Ax^k),\ k = 0, 1, \cdots .$

iterative simultaneous algorithm for a system of linear equations is defined as in Algorithm 2 [27]. Suppose $\bar{b} = M^{1/2}b$ and $\bar{A} = M^{1/2}A$, and consider the following weighted least squares problem

$$\min f(x),\ \text{where } f(x) = \frac{1}{2}\|b - Ax\|_M^2 = \frac{1}{2}\|\bar{b} - \bar{A}x\|^2. \tag{3.8}$$

Let x^* be the solution of 3.8. The following convergence result holds for this problem [21, 7].

Theorem 3.9. *Suppose $\rho = \rho(A^T MA)$ and $.0 \leqslant \epsilon \leqslant \lambda_k \leqslant (2 - \epsilon)/\rho$. If $\epsilon > 0$ or $\epsilon = 0$ and*

$$\sum_{k=0}^{\infty} \min(\rho\lambda_k, 2 - \rho\lambda_k) = +\infty,$$

then the iterative algorithm 2 converges to a solution of problem 3.8, such as x^*. Additionally, if $x^0 \in R(A^T)$, then x^* is the solution with the smallest Euclidean norm among all solutions (and, of course, this solution is unique).

In the next subsection, we explain three choices of the matrix M.

3.2.1. Some Fully Simultaneous Methods

For suitable choices of the matrix M, some well-known fully simultaneous methods can be written in the form of Algorithm 2, which we briefly explain in the next subsection. These methods are commonly used to solve systems of linear equations or other optimization problems, where the iterative approach involves simultaneous updates across multiple sets or variables.

If we consider the identity matrix M in Algorithm 2, the classical Landweber method is obtained. If we set the matrix $M = \frac{1}{m}\text{diag}(1/\|a_i\|^2)$ in Algorithm 2, we obtain the Cimmino method, where m is the number of rows of the matrix A, and a_i is the i-th row of A. The CAV method was introduced by Censor and Gordon. In this method, the matrix M is given by

$$M = \text{diag}\left(\frac{1}{\sum_{j=1}^{n} N_j a_{ij}^2}\right),$$

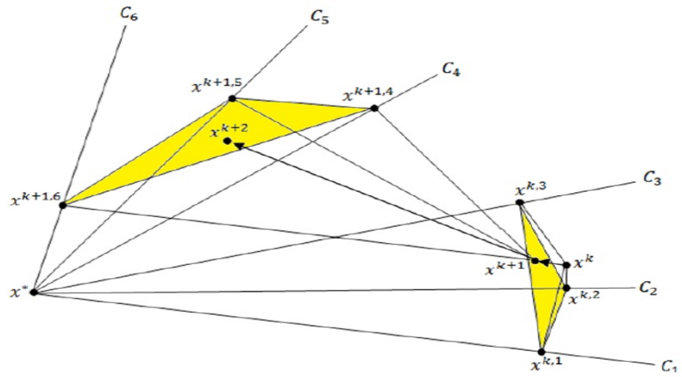

Figure 5: Illustration of a sequential projection method, where projections are applied one after another onto individual constraint sets in a fixed or cyclic order.

where N_j is the number of non-zero elements in the j-th column of the matrix A. The matrix form of diagonally relaxed orthogonal projections (DROP) method is given as follows

$$\begin{aligned} x^{k+1} &= x^k + \lambda_k S A^T M(b - Ax^k), \ k = 0, 1, 2, \cdots \\ S &= diag\left(\frac{m}{N_j}\right), \\ M &= \frac{1}{m} diag\left(\frac{1}{\|a_i\|^2}\right). \end{aligned}$$

3.3. Block methods

Since sequential algorithms have higher speed, and despite the lower speed of simultaneous algorithms, they have the advantage of parallelization, block methods were introduced to combine both advantages. In block methods, the problem is initially divided into several blocks, each containing a set of convex sets C_i from the convex feasible set of the problem. This division breaks the problem down into smaller sub problems, which makes it particularly useful for problems where there are many convex sets. Block algorithms are categorized into two general types: sequential and simultaneous block, depending on the algorithm used to solve the problem within each block (sequential or simultaneous), and how the solutions from each block are utilized to construct subsequent iterations.

3.3.1. Sequential block methods

In sequential block algorithms, the blocks are selected in order, and within each block, the next solution is generated using the current solution and the sets within the block by applying a simultaneous algorithm. This process continues until all blocks have been selected. After reaching the last block, a cycle is completed, and the first block is used again for the next cycle. This process continues iteratively. Figure 5 shows how iterations are carried out in sequential block algorithms for a convex feasible problem with 6 sets, where the problem is divided into two blocks: $B_1 = \{C_1, C_2, C_3\}$ and $B_2 = \{C_4, C_5, C_6\}$.

3.3.2. Simultaneous block methods

In simultaneous block algorithms, the current solution x^k is used simultaneously for all blocks, and within each block, one of the types of simultaneous algorithms is applied. Then, based on the solutions obtained from each block, the next solution x^{k+1} is generated using various methods, often involving convex combinations. Simultaneous block algorithms are generally slower than sequential block algorithms but are preferred due to their parallel implementation capability.

Figure 6 illustrates how iterations are carried out in simultaneous block algorithms for a convex feasible problem, where the problem is divided into two blocks. The main idea in row-block algorithms for

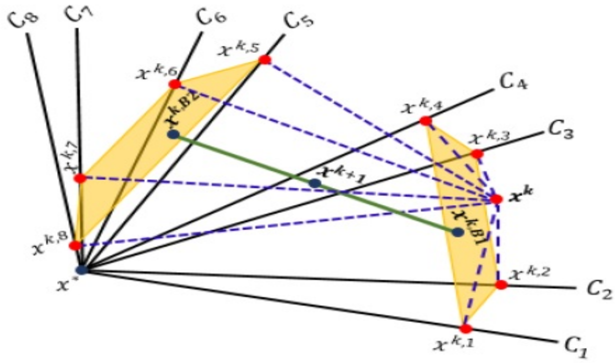

Figure 6: Illustration of a simultaneous block projection method, where projections are performed in parallel onto blocks of constraint sets, and the results are combined to update the solution.

the linear system 1.1 is that the matrix A and the vector b are decomposed into blocks of equations. Then, within each block, a simultaneous algorithm is employed. Block methods enable the use of subsets of the data rather than the entire dataset, making them well-suited for high-dimensional problems. Moreover, as mentioned earlier, block methods can be very useful for parallel implementation.

Assume that the system of linear equations 1.1 is decomposed into p blocks of equations, where each equation appears in at least one block, and some equations may appear in multiple blocks simultaneously. We use the symbols A_t and b^t to denote the t-th block of equations from A and b, respectively. For the case where the blocks do not share equations, we can write:

$$A = \begin{pmatrix} A_1 \\ A_2 \\ \vdots \\ A_p \end{pmatrix}, \quad b = \begin{pmatrix} b^1 \\ b^2 \\ \vdots \\ b^p \end{pmatrix}.$$

The simultaneous block algorithm for the linear system is defined as Algorithm 3 [27] In this algorithm

Algorithm 3 Simultaneous block

Given z^k, compute
$z^{k,0} = z^k$
$z^{k,t} = z^k + \lambda_t A_t^T M_t (b^t - A_t z^k), \quad t = 1, \cdots, p$
$z^{k+1} = \frac{1}{p} \sum_{t=1}^{p} z^{k,t}.$

$\{\lambda_t\}_{t=1}^p$ and $\{M_t\}_{t=1}^p$ are, respectively, a set of relaxation parameters and symmetric positive definite matrices.In the algorithm, a "cycle" refers to using all the data at least once, meaning transitioning from point z^k to point z^{k+1}. Each cycle of Algorithm 3 can easily be written as a non-block simultaneous algorithm. In fact, the sum $z^{k,t}$ over t leads us to the following result:

$$\sum_{t=1}^{p} z^{k,t} = pz^k + \sum_{t=1}^{p} A_t^T \lambda_t M_t (b^t - A_t z^k) = pz^k + A^T M (b - Az^k),$$

where $M = diag(\lambda_t M_t)$ is a block diagonal, symmetric positive definite matrix (assuming the relaxation parameters are positive). Thus, each cycle of Algorithm 3 can be written as:

$$z^{k+1} = z^k + \frac{1}{p} A^T M (b - Az^k). \tag{3.9}$$

Algorithm 4 Sequential block

Given y^k, compute
$y^{k,0} = y^k$
$y^{k,t} = y^{k,t-1} + \lambda_t A_t^T M_t (b^t - A_t y^{k,t-1}), \quad t = 1, 2, \cdots, p,$
$y^{k+1} = y^{k,p}$

The block-sequential iterative algorithm for the linear system is defined as Algorithm 4 [27]. In this algorithm, $\{\lambda_t\}_{t=1}^p$ and $\{M_t\}_{t=1}^p$ are, respectively, a set of relaxation parameters and symmetric positive definite matrices. The following convergence theorem holds for Algorithm 4 [14]

Theorem 3.10. *Given that*

$$0 < \epsilon \leqslant \lambda_t \leqslant \frac{2-\epsilon}{\rho(A_t^T M_t A_t)}, \quad t = 1, 2, \cdots, p,$$

then the sequence of cycles $\{y^k\}$ defined by Algorithm 4 converges to a point that satisfies a specific system of equations. Moreover, if $b \in R(A)$ and $y^0 \in R(A^T)$, then $\{y^k\}$ converges to a point that is the solution of the system $Ax = b$ with the minimum Euclidean norm.

Here, we introduce several well-known examples that are obtained by selecting specific weight matrices. Suppose that

$$a^{t,j}, \ t = 1, 2, \cdots, p, \ j = 1, 2, \ldots, m(t),$$

denotes the j-th row of the block $A_t \in \mathbb{R}^{m(t)\times n}$.

Example 3.11. Suppose

$$M_t = \operatorname{diag}\left(\frac{w_{tj}}{\|a^{t,j}\|^2}\right), \quad t = 1, 2, \ldots, p, \tag{3.10}$$

where $w_{tj} > 0$ and $\sum_{j=1}^{m(t)} w_{tj} = 1$ are predefined weights. By using these weights in Algorithm 4, we obtain the Cimmino method [7].

Example 3.12. Suppose

$$M_t = (A_t A_t^T)^\dagger, \quad t = 1, 2, \ldots, p, \tag{3.11}$$

where $Q^\dagger$ represents the pseudoinverse of the matrix Q. Furthermore, if we set $p = m$ in Algorithm 4, we obtain the ART (Algebraic Reconstruction Technique) algorithm [20, 25].

3.4. Advantages and Disadvantages

Sequential algorithms are faster than simultaneous algorithms [8], but simultaneous algorithms have the essential feature that they can be implemented in parallel. For example, in the Cimmino algorithm, the operation of obtaining each orthogonal image can be done by a separate processor. In block-sequential algorithms, it is possible to implement the method in parallel within each block, while moving sequentially between blocks. This allows us to take advantage of both sequential and simultaneous methods simultaneously. In block-simultaneous algorithms, the operation related to each block can be performed by a separate processor, and then the final solution of each iteration can be obtained, leading to high parallelizability of these types of algorithms. Table 1 compares different algorithms for solving convex feasibility problems in terms of solution speed and parallelizability [32].

Table 1: Comparison of various algorithms for solving convex feasibility problems.

Algorithm	Speed	Parallelizability
sequential	very high speed	no parallelizability
simultaneous	very low speed, even with parallelization	very high parallelizability
sequential block	high speed	low parallelizability
simultaneous block	low speed	high parallelizability

4. Conclusion

In this paper, we have explored various sequential, simultaneous, and block-based algorithms for solving convex feasibility problems. We presented both theoretical foundations and practical implementations of these algorithms, highlighting their respective advantages and limitations.

Sequential algorithms, while offering high speed, do not support parallelization, which limits their scalability for large problems. On the other hand, simultaneous algorithms, despite their lower speed, allow for high parallelization, making them suitable for large-scale problems where parallel computing resources are available.

Block-based algorithms, both sequential and simultaneous, combine the benefits of both approaches. Block-sequential algorithms offer the advantage of parallel implementation within blocks while maintaining a sequential structure between blocks. This hybrid approach balances speed with parallelizability. Block-simultaneous algorithms, while slower, offer excellent parallelizability, making them ideal for distributed computing environments.

Ultimately, the choice of algorithm depends on the specific problem and computational resources. Sequential methods are preferable for smaller-scale problems, while block-based and simultaneous methods should be considered for larger problems that benefit from parallel processing. Further research and development can focus on optimizing these methods for even better performance in diverse computational environments.

Future work could explore a broader class of projection methods, such as string-averaging techniques, which generalize and unify sequential, simultaneous, and block-iterative schemes. A comparative study focusing on their theoretical properties, convergence behavior, and practical efficiency in large-scale applications would be particularly valuable. Additionally, examining hybrid approaches or integrating these methods with acceleration strategies may offer further improvements.

Acknowledgement(s)

We would like to thank the referees for their valuable comments and constructive suggestions, which helped improve the quality of the manuscript.

References

[1] Aharoni, R., and Censor, Y. 1989. Block-iterative projection methods for parallel computation of solutions to convex feasibility problems. Linear Algebra and Its Applications, 165175. 1

[2] Bauschke, H. H., and Borwein, J. M. 1996. On projection algorithms for solving convex feasibility problems. SIAM review 38, 367426. doi: 10.1137/S0036144593251710 1, 2

[3] Byrne, C. 2000. Block-iterative interior point optimization methods for image reconstruction from limited data. Inverse Problems 16, 1405. doi: 10.1088/0266-5611/16/5/316 1

[4] Cegielski, A. 2012. Iterative methods for fixed point problems in Hilbert spaces, volume 2057. Springer. doi: 10.1007/978-3-642-30901-4 3.2

[5] Censor, Y. 1981. Row-action methods for huge and sparse systems and their applications. SIAM review, 23(4):444–466. doi: 10.1137/1023097

[6] Censor, Y. 1984. Iterative methods for the convex feasibility problem. North-Holland Mathematics Studies, 8391. 1

[7] Censor, Y., and Elfving, T. 2002. Block-iterative algorithms with diagonally scaled oblique projections for the linear feasibility problem. SIAM Journal on Matrix Analysis and Applications 24, 4058. doi: 10.1137/S089547980138705X 2

[8] Censor, Y., Gordon, D., and Gordon, R. 2001. BICAV: A block-iterative parallel algorithm for sparse systems with pixel-related weighting. Medical Imaging, IEEE Transactions on 20, 10501060. doi: 10.1109/42.959302 1, 3.2, 3.11

[9] Censor, Y., Gordon, D., and Gordon, R. 2001. Component averaging: An efficient iterative parallel algorithm for large and sparse unstructured problems. Parallel computing 27, 777 808. doi: 10.1016/S0167-8191(00)00100-9 1, 3.4

[10] Censor, Y., and Herman, G. T. 1987. On some optimization techniques in image reconstruction from projections. Applied Numerical Mathematics, 3(5):365391. doi: 10.1016/0168-9274(87)90028-6 1, 3.2

[11] Censor, Y., and Zenios, S. A. 1997. Parallel optimization: Theory, algorithms, and applications. Oxford University Press on Demand. 1

[12] Cimmino, G., and delle Ricerche, C. N. 1938. Calcolo approssimato per le soluzioni dei sistemi di equazioni lineari. Istituto per le applicazioni del calcolo. 1, 3

[13] Elfving, T. 1980. Block-iterative methods for consistent and inconsistent linear equations. Numerische Mathematik 35, 112. 1, 3.2

[14] Elfving, T., and Nikazad, T., 2009. Properties of a class of block-iterative methods. Inverse problems 25, 115011. 1

[15] Gordon, R. Bender, R. and Herman. G. T., 1970. Algebraic reconstruction techniques (art) for three-dimensional electron microscopy and x-ray photography. Journal of theoretical Biology, 29(3):471481. doi: 10.1016/0022-5193(70)90109-8 3.3.2

[16] Gubin, L., Polyak, B., and Raik, E. 1967. The method of projections for finding the common point of convex sets. USSR Computational Mathematics and Mathematical Physics 7, 124. doi: 10.1016/0041-5553(67)90113-9 1

[17] Gustafsson, B. 1996. Mathematics for computer tomography. Physica Scripta, 38. doi: 10.1088/0031-8949/1996/T61/006 3.1

[18] Harshbarger, T. B., and Twieg, D. B. 1999. Iterative reconstruction of single-shot spiral mri with off resonance. IEEE transactions on medical imaging 18, 196–205. 2

[19] Herman, G. T., 1980. Image reconstruction from projections. The fundamentals of computerized tomography, 316. 1

[20] Herman, G. T., 2009. Fundamentals of computerized tomography: image reconstruction from projections. Springer Science & Business Media. 1

[21] Jiang, M., and Wang, G., 2003. Convergence studies on iterative algorithms for image reconstruction. Medical Imaging, IEEE Transactions on 22, 569579. 3.12

[22] Kaczmarz, S. 1937. Angenäherte auflösung von systemen linearer gleichungen. Bulletin International de lAcademie Polonaise des Sciences et des Lettres, 355357. 3.2

[23] Kak, A. C., and Slaney, M. 1988. Principles of computerized tomographic imaging. IEEE press. 1, 3.1

[24] Kang, C.-g., Zhou, H., 2021. The extensions of convergence rates of Kaczmarz-type methods, Journal of Computational and Applied Mathematics 113099. 1, 2

[25] Natterer, F. 1986. The mathematics of computerized tomography. John Wiley, New York. 3.1.1, 3.5, 3.6, 3.1.2

[26] Nikazad, T., and Abbasi, M., 2013. An acceleration scheme for cyclic subgradient projections method. Computational Optimization and Applications 54, 7791. 1, 3.12

[27] Nikazad, T., Abbasi, M., and Elfving, T. Error minimizing relaxation strategies in landweber and kaczmarz type iterations. 3.1

[28] Nikazad, T., Davidi, R., and Herman, G. 2012 Accelerated perturbation-resilient block-iterative projection methods with application to image reconstruction. Inverse problems 28, 035005. doi: 10.1088/0266-5611/28/3/035005 3.2, 3.3.2, 3.3.2

[29] Penfold, S., Schulte, R. W., Censor, Y., Bashkirov, V., McAllister, S., Schubert, K., and Rosenfeld, A. B. 2010. Block-iterative and string-averaging projection algorithms in proton computed tomography image reconstruction. 1

[30] Popa, C., 1995. Least-squares solution of overdetermined inconsistent linear systems using Kaczmarzs relaxation, Int. J. Comput. Math. 7989. 1

[31] Rodriguez, A., 2004. Principles of magnetic resonance imaging. Revista mexicana de física 50, 272286. 3.1.2

[32] Sørensen, H. H. B., and Hansen, P. C., 2014. Multicore performance of block algebraic iterative reconstruction methods. SIAM Journal on Scientific Computing 36, C524C546. doi: 10.1137/130920642 1

[33] Tanabe, K., 1971. Projection method for solving a singular system of linear equations and its applications, Numer. Math. 203214. 3.4

[34] Wang, G., Vannier, M. W. and Cheng. P. C., 1999. Iterative x-ray cone-beam tomography for metal artifact reduction and local region reconstruction. Microscopy and microanalysis, 5(1):58-65. doi: 10.1017/S1431927699000057 3.4, 3.1.1

[35] Steffen, R., Hill, D. R., and DuPont, H. L. (2015). Travelers diarrhea: a clinical review. Jama, 313(1), 71–80. 1