TÜBİTAK

Turkish Journal of Mathematics

http://journals.tubitak.gov.tr/math/

Research Article

Turk J Math
(2017) 41: 733 – 748
© TÜBİTAK
doi:10.3906/mat-1503-6

Choosing the relaxation parameter in sequential block-iterative methods for linear systems

Touraj NIKAZAD*, **Shaghayegh HEIDARZADE**
School of Mathematics, Iran University of Science and Technology, Narmak, Tehran, Iran

Received: 02.03.2015 • **Accepted/Published Online:** 20.07.2016 • **Final Version:** 22.05.2017

Abstract: In this paper we introduce two strategies for picking relaxation parameters to control the semiconvergence behavior of a sequential block-iterative method. A convergence analysis is presented. We also demonstrate the performance of our strategies by examples taken from tomographic imaging.

Key words: Sequential block-iterative methods, Cimmino and CAV iteration, semiconvergence, relaxation parameters, tomographic imaging

1. Introduction

Ill-posed and large-scale problems, such as computed tomography, take place in many fields of mathematics and physical sciences. Usually these problems are handled by iterative methods instead of direct methods. There is an interest in regularizing iterative methods where the iteration vector can be considered as a regularized solution. Using incorrect and noisy input data, which are due to measurements or rounding errors, we obtain a more difficult problem to solve. The iteration index of an iterative method may be considered as a regularization parameter. Initially the iteration vectors approach a regularized solution. Nevertheless, continuing the iteration process often produces iteration vectors that are corrupted by noise; see [16, p. A2002] and [19, p. 1]. This phenomenon was called semiconvergence by Natterer [30, page 157]; for analysis of the phenomenon, see, e.g., [4, 20, 21, 23, 33, 35]. The typical overall error behavior is shown in Figure 1. If there is a reliable stopping rule then we may get a proper approximation of the sought solution, i.e. x^*. Otherwise, due to the semiconvergence phenomenon, the stopping criterion may stop the iteration process early or far from a proper iteration index. For this reason, we will study finding relaxation parameters for postponing the semiconvergence phenomenon.

Computational tomography, like many other large-scale ill-posed problems, leads to large linear systems of equations (often inconsistent) with noisy data, of the form

$$Ax \approx b,\ b = \bar{b} + \delta b. \tag{1}$$

Here $\bar{b}$ denotes the exact data and δb is the perturbation consisting of additive noise. We are interested in iterative algorithms for solving the linear system (1). The algorithms can be generally classified as being either sequential or simultaneous or block-iterative, see, e.g., [11], and the review paper [2] for a variety of specific algorithms of these kinds.

*Correspondence: tnikazad@iust.ac.ir
2010 *AMS Mathematics Subject Classification:* 65F10, 65R32.

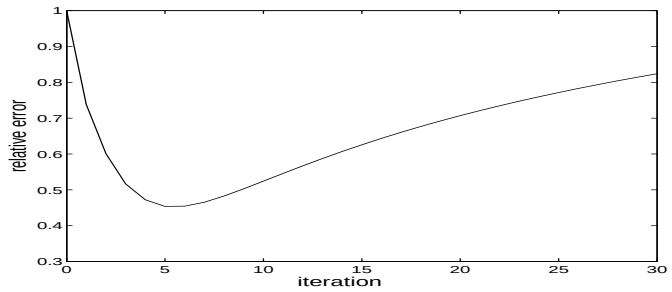

Figure 1. Semiconvergence phenomenon.

The algebraic reconstruction technique (ART) is a fully sequential method and has a long history and rich literature. Originally it was proposed by Kaczmarz [26], and independently, for use in image reconstruction, by Herman [23]. The vector of unknowns is updated at each equation of the system, after which the next equation is addressed. We define one cycle as one pass through all the data. The prototype of simultaneous algorithms is the well-known Cimmino method [12]. In this method the current iterate is first projected on all sets to obtain intermediate points. The next iterate is made by an averaging process, as a convex combination, of intermediate points. We now explain the block-iterative method. The basic idea of a block-iterative algorithm is to partition the data A and b of the system (1) into blocks of equations (rows) and treat each block according to the rule used in the simultaneous algorithm for the whole system, passing, e.g., cyclically over all the blocks. Figure 2 illustrates how these three methods work. Here H_i denotes the ith hyperplane, i.e. $H_i = \{x \in \mathbb{R}^n,\ \langle x, a^i\rangle = b_i\}$ where a^i and b_i indicate the ith row of A and b, respectively.

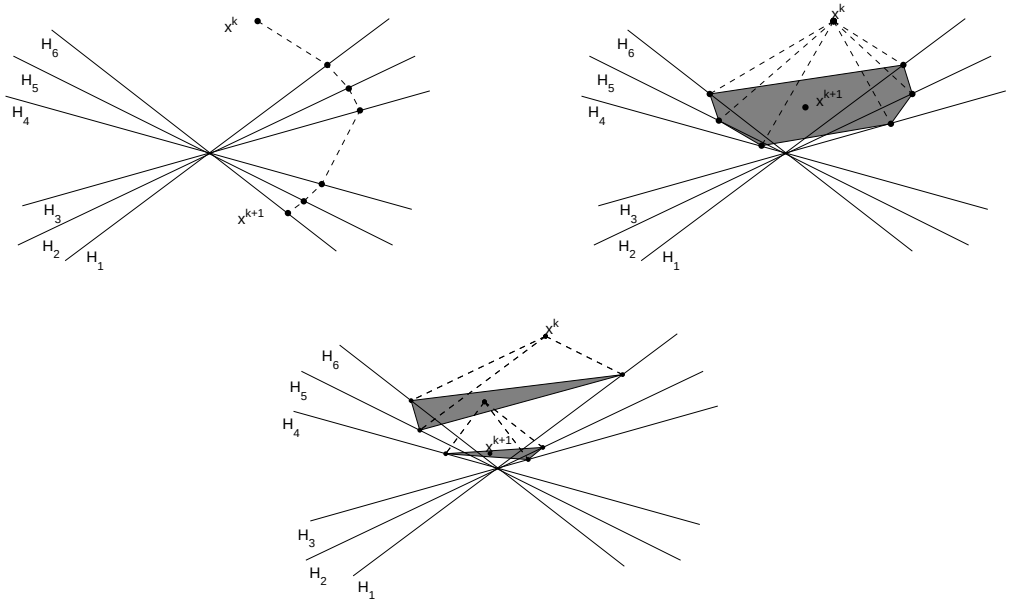

Figure 2. Full sequential method [top-left (a)], full simultaneous method [top-right (b)] and sequential block-iterative method [bottom (c)].

In [19], the semiconvergence properties of the unconstrained simultaneous iterative reconstruction technique (SIRT) using a constant relaxation parameter (called the stationary method) were studied, whereas [16]

gave a detailed study of the phenomenon for projected nonstationary SIRT. Both papers proposed two new relaxation strategies (updated after each iteration), which provide a delayed semiconvergence phenomenon. Recently, in [17], the semiconvergence phenomenon was analyzed for stationary sequential block-iterations without any suggestion for picking the relaxation parameter.

In the present work, we consider the nonstationary sequential block-iterative algorithm allowing a certain family of relaxation parameters to control the semiconvergence phenomenon. We use two kinds of relaxation parameters: one is updated in each iteration and the other one is only updated after each cycle. In both cases, we give their convergence proofs based on [25].

Our paper is organized as follows. In Section 2 we recall the sequential block-iterative method and give some well-known instances. We introduce two strategies for picking relaxation parameters and give their convergence analysis in Section 3. In Subsection 3.1 we give two special cases of the strategies that satisfy conditions of Propositions 4 and 7. Furthermore, we consider a projected case of the sequential block-iterative method in Subsection 3.2 but without any convergence analysis. In Section 4 we give some numerical tests explaining the efficiency of our relaxation parameters.

2. Preliminaries and algorithms

Projection algorithms are successful iterative methods in the area of computational tomography; see, e.g., [6]. It is shown that projection methods often have a computational advantage, which makes them successful in many real-world applications. They commonly have the ability to handle huge-sized problems of dimensions beyond which more sophisticated methods cease to be efficient or even applicable due to memory requirements. In a simple way, one may classify projection methods as either sequential or simultaneous. The block-iterative methods, which lie between the sequential and simultaneous cases, have been studied in several works with different applications; see, e.g., [1, 5, 7–9, 14, 15, 18, 25, 32, 34].

Are the block-iterative methods still of interest? Probably the most well-known iterative methods are Krylov-type methods. However, steepest descent can be an excellent alternative when solving noisy linear systems arising from the discretization of ill-posed problems. The reason is again semiconvergence behavior and the blowing up of noise. Specifically, it was shown in [29] that, for an ill-posed problem arising in image restoration, steepest descent can have a more stable convergence behavior than the Krylov method CGLS.

The performance of different block-iterative methods, such as sequential and simultaneous block-iterative methods, with a fixed relaxation parameter was studied in [34]. The computational results showed that for multicore computers the sequential approach is preferable and the most useful for large problems. Therefore, in this paper we only consider a class of sequential block-iterative methods.

In this paper we consider the linear system of equations (1) such that $A \in \mathbb{R}^{m\times n}$ and $b \in \mathbb{R}^m$. Let $B = \{1, 2, \cdots, m\}$. The index set B is partitioned into q subsets B_s such that

$$B = \cup_{s=1}^{q} B_s. \tag{2}$$

Indeed, A and b are partitioned into q (not necessarily disjoint) row blocks $\{A_s\}$ and $\{b^s\}$, in accordance with $\{B_s\}$. Let $\langle x, y\rangle$ be the Euclidean inner product and $\|x\|$ the corresponding norm. For a matrix A, we define $\|A\| = \max_{x\neq 0} \|Ax\|/\|x\|$. Further, for a symmetric positive definite (SPD) matrix W, $\|x\|_W = \sqrt{\langle Wx, x\rangle}$ denotes a weighted Euclidean norm and $W^{1/2}$ is the square root of W. P_Ω is the orthogonal projection onto a closed convex set Ω and $N(A)$ denotes the null space of A. By $diag(z_i)$, where z_i can be either scalars or

matrices, we mean the (block) diagonal matrix

$$\begin{pmatrix} z_1 & & \\ & z_2 & \\ & & \ddots \end{pmatrix}.$$

We use the standard term sequential block-iterative (SeqBI), see [11, 25], such that an iterative step sequentially moves from one block to the next. In general we consider the following SeqBI algorithm:

Algorithm 1 SeqBI

Initialization: $x^0 \in \mathbb{R}^n$ *is arbitrary.*

Iterative Step: *For given* x^k, *compute*

$$\begin{aligned} x^{k,0} &= x^k, \\ x^{k,s} &= T_{k,s}(x^{k,s-1}),\ s = 1, \dots, q, \\ x^{k+1} &= x^{k,q}, \end{aligned}$$

where $T_{k,s}(x) = x + \lambda_{k,s} A_s^T M_s (b^s - A_s x)$. $\{M_s\}_{s=1}^q$ and $\lambda_{k,s}$ are SPD matrices and relaxation parameters, respectively. When $q = 1$ there is just one block so $M_s = M \in \mathbb{R}^{m\times m}$, and the method is called fully simultaneous iteration. On the other hand, when each block consists of a single row, then $q = m$ and $M_s \in \mathbb{R}$ for $s = 1, \cdots, m$ and we get a fully sequential iteration. In Algorithm 1, an iterative step moves from x^k to x^{k+1}, which is called a *cycle*, and it consists of subiterative steps (referred to as *atomic steps or iterations*) that move from $x^{k,s-1}$ to $x^{k,s}$.

Numerous well-known methods can be written in the form of Algorithm 1 for appropriate choices of matrix M_s. When M_s is equal to the identity matrix we get the classical Landweber method [28]. Cimmino's method [12] is obtained with $M_s = \frac{1}{m_s}\text{diag}(1/\|a^i\|^2)$ where a^i denotes the ith row of A_s and m_s stands for the number of rows in A_s. The CAV method [10] uses $M_s = \text{diag}(1/\sum_{j=1}^n N_j a_{ij}^2)$ where N_j is the number of nonzeros in the jth column of A_s.

3. Convergence results

Our main theoretical result is given in this section. As we mentioned before, we consider two strategies for picking relaxation parameters. They are updated after each iteration and cycle.

Let

$$\begin{aligned} \sigma_s &= \|M_s^{1/2} A_s\|,\ \sigma_{max} = \max\{\sigma_s\}_{s=1}^q, \\ \sigma_{min} &= \min\{\sigma_s\}_{s=1}^q,\ \theta = \sigma_{min}/\sigma_{max} \end{aligned}$$

for $s = 1, \cdots, q$. Assume an arbitrary sequence $\{\gamma_k\}$ such that

$$0 \le \gamma_k \le 1,\ \sum_{k=0}^{\infty} \gamma_k = \infty,\ \lim_{k\to\infty} \gamma_k = 0. \tag{3}$$

Let

$$S = \{x \in \mathbb{R}^n | A^T M A x = A^T M \bar{b}\} \tag{4}$$

where $M = diag(M_s)$ is a block diagonal matrix with diagonal entries $\{M_s\}_{s=1}^{q}$. Furthermore, assume that $\bar{x}$ has the minimum Euclidian norm among all vectors of S.

Remark 2 *In many applications of computational tomography the linear system of equations (1) is typically underdetermined. This happens in limited angle applications (e.g., breast X-ray tomography) and few-projection measurements (where the X-ray dose should be limited). When the system of linear equations (1) is underdetermined, it is common practice to seek a solution that minimizes some objective function. The problem then is to select, from among all the feasible solutions, a particular solution that has a good chance of being near the correct image. One approach, which we are interested in, is to take the feasible solution of (1) having the minimum Euclidean norm $\|x\|_2$. There are other approaches such as minimizing one-norm, $\|x\|_1$, and minimizing total variation; see [13] and [32], respectively.*

We now consider the following two strategies. The first strategy is constant during a cycle, i.e.

$$\lambda_k = \begin{cases} \theta^4 \sigma_{min}^{-2}, & k = 0, 1 \\ \theta^4 \sigma_{min}^{-2} \gamma_k, & k \geq 2. \end{cases} \tag{5}$$

Therefore, the strategy (5) is updated after each cycle. The second strategy varies after each iteration, i.e.

$$\lambda_{k,s} = \begin{cases} \theta^4 \sigma_s^{-2}, & k = 0, 1 \\ \theta^4 \sigma_s^{-2} \gamma_k, & k \geq 2. \end{cases} \tag{6}$$

Note that $\lambda_k = \lambda_{k,s} \left(\sigma_s^2 / \sigma_{min}^2\right) \geq \lambda_{k,s}$.

Next, using [25], we show that Algorithm 1 with both strategies (5) and (6) converges. First we recall the following theorem:

Theorem 3 *[25, Theorem II.1, p.572] Let there exist $\rho > 0$ such that $\|A_s\|_{M_s} \leq \rho$ for $s = 1, 2, \cdots, q$ and $0 \leq \rho^2 \lambda_k \leq 2$ for $k \geq 0$. Also assume that the relaxation parameters λ_k are constant within each cycle.*

(a) If (1) is consistent and

$$\sum_{k=0}^{\infty} \min\{\rho^2 \lambda_k, 2 - \rho^2 \lambda_k\} = +\infty \tag{7}$$

then the sequence generated by Algorithm 1 converges to $\bar{x} + P_{N(A)}(x^0)$.

(b) If the partition (2) is disjoint and

$$\lim_{k \to \infty} \lambda_k = 0 \text{ and } \sum_{k=0}^{\infty} \lambda_k = +\infty \tag{8}$$

then the sequence generated by Algorithm 1 converges to $\bar{x} + P_{N(A)}(x^0)$ even if (1) is inconsistent.

We will show that the relaxation parameters (5) satisfy all conditions of Theorem 3. Therefore, we can use Theorem 3 to illustrate that the sequence generated by Algorithm 1 with relaxation parameter (5) converges.

Proposition 4 *The relaxation parameters (5) hold all conditions of Theorem 3.*

Proof Since $\|A_s\|_{M_s} = \|M_s^{1/2}A_s\| = \sigma_s \leq \sigma_{max}$, we put $\rho = \sigma_{max} > 0$ in Theorem 3. For the case $k = 0, 1$ we have

$$0 \leq \rho^2\lambda_k = \sigma_{max}^2 \frac{\theta^4}{\sigma_{min}^2} = \theta^2 \leq 1 \tag{9}$$

and for $k \geq 2$, using $0 \leq \gamma_k \leq 1$, we obtain that

$$0 \leq \rho^2\lambda_k = \theta^2\gamma_k \leq 1. \tag{10}$$

Using (10) and (3) we get

$$\begin{aligned}\sum_{k=0}^{\infty} \min\{\rho^2\lambda_k, 2-\rho^2\lambda_k\} &= \rho^2 \sum_{k=0}^{\infty} \lambda_k \\ &\geq \rho^2 \frac{\theta^4}{\sigma_{min}^2} \sum_{k=2}^{\infty} \gamma_k = \infty.\end{aligned}$$

Since conditions (8) are the same conditions as (3), we get that the strategy (5) satisfies all conditions of Theorem 3. □

Next we concentrate on the second strategy, i.e. (6). Combining [25, Theorem $A.3$, Remark $A.4$, and Theorem $A.10$], one gets the following convergence result:

Theorem 5 *Let there exist $\rho > 0$ such that $\|A_s\|_{M_s} \leq \rho$ for $s = 1, 2, \cdots, q$ and $0 \leq \rho^2\lambda_{k,s} \leq 2$ for $k \geq 0$ and $s = 1, 2, \cdots, q$. The relaxation parameters λ_k are allowed to vary in each iteration. If (1) is consistent and*

$$\sum_{k=0}^{\infty} t_k = +\infty \tag{11}$$

where $t_k = \min\{\tau_{k,s}\}_{s=1}^{q}$ and $\tau_{k,s} = \min\{\rho^2\lambda_{k,s}, 2-\rho^2\lambda_{k,s}\}$, then the sequence generated by Algorithm 1 converges to $\bar{x} + P_{N(A)}(x^0)$.

Remark 6 *As explained in Theorems 3 and 5, the sequence generated by Algorithm 1 converges to $\bar{x}+P_{N(A)}(x^0)$ irrespective of the disjointness of blocks where the linear system of equations (1) is consistent. Therefore, the solution set (4) is maintained as it is and we do not need to redefine a new solution set where $\{M_s\}_{s=1}^{q}$ are not disjoint. On the other hand, for the case of an inconsistent linear system, if the partition (2) is not disjoint but*

$$\sum_{s=1}^{q} A_s^T M_s b^s = A^T M b \tag{12}$$

holds, then the results of Theorem 3 are still true; see [25, Remark A.11, p. 576]. However, later, we only consider the disjoint partitioning case.

We next give an example and show that (12) holds for a partition that is not disjoint. Similar to [16], consider the following slightly modified problem in the form of a regularized problem with regularization parameter α

and balancing parameter $\mu = \|M\|$:

$$\min_{x\in\mathbb{R}^n} \frac{1}{2}\left(\|Ax-b\|_M^2 + \alpha^2\mu\|x\|^2\right) = \min_{x\in\mathbb{R}^n} \frac{1}{2}\left\|\widehat{A}x-\widehat{b}\right\|_{\widehat{M}}^2,$$

where

$$\widehat{M} = \operatorname{diag}(M, \mu I),\ M = diag(M_1, \cdots, M_q)$$
$$\widehat{A} = \begin{pmatrix} A \\ \alpha I \end{pmatrix},\ \widehat{b} = \begin{pmatrix} b \\ 0 \end{pmatrix}.$$

We consider the following partitioning of $\widehat{A}$, $\widehat{b}$ and the corresponding weight matrix as follows:

$$\widetilde{A} = \begin{pmatrix} A_1 \\ \alpha I \\ \vdots \\ A_q \\ \alpha I \end{pmatrix},\ \widetilde{b} = \begin{pmatrix} b_1 \\ 0 \\ \vdots \\ b_q \\ 0 \end{pmatrix} \tag{13}$$

$$\widetilde{M} = \begin{pmatrix} \widetilde{M_1} & & \\ & \ddots & \\ & & \widetilde{M_q} \end{pmatrix} \tag{14}$$

where $\widetilde{M_s} = diag(M_s, \mu I)$ for $s = 1, \cdots, q$. It is easy to check that the partitioning (13–14) is not disjoint but it satisfies (12) if $\{A_s\}_{s=1}^q$ are disjoint.

Proposition 7 *The relaxation parameters (6) hold for all conditions of Theorem 5.*

Proof As in Proposition 4, we put $\rho = \sigma_{max} > 0$, and for $k = 0, 1$ we obtain

$$0 \le \rho^2\lambda_{k,s} = \sigma_{max}^2\frac{\theta^4}{\sigma_s^2} = \theta^2\left(\frac{\sigma_{min}}{\sigma_s}\right)^2 \le 1. \tag{15}$$

Using $0 \le \gamma_k \le 1$ we get

$$0 \le \rho^2\lambda_{k,s} = \theta^2\left(\frac{\sigma_{min}}{\sigma_s}\right)^2\gamma_k \le 1 \tag{16}$$

where $k \ge 2$. Using (16) we obtain $\tau_{k,s} = \rho^2\lambda_{k,s}$. Therefore,

$$t_k = \min\{\rho^2\lambda_{k,s}\}_{s=1}^q \ge \theta^4\gamma_k,$$

which verifies (11). □

3.1. Choosing γ_k

In this section we consider two special cases for γ_k that are inspired by Elfving et al. [19]. Following [19, (2.13)] we consider the equation

$$g_{k-1}(y) = (2k-1)y^{k-1} - (y^{k-2} + \cdots + y + 1) = 0, \tag{17}$$

which has a unique real root $\zeta_k \in (0,1)$. The roots satisfy $0 < \zeta_k < \zeta_{k+1} < 1$ and $\lim_{k\to\infty} \zeta_k = 1$ (see [19, Propositions 2.3, 2.4]), and they can easily be precalculated; see the Table.

Table. The unique root $\zeta_k \in (0,1)$ of $g_{k-1}(y) = 0$.

k	ζ_k	k	ζ_k	k	ζ_k	k	ζ_k	k	ζ_k	k	ζ_k
2	0.3333	7	0.8156	12	0.8936	17	0.9252	22	0.9424	27	0.9531
3	0.5583	8	0.8392	13	0.9019	18	0.9294	23	0.9449	28	0.9548
4	0.6719	9	0.8574	14	0.9090	19	0.9332	24	0.9472	29	0.9564
5	0.7394	10	0.8719	15	0.9151	20	0.9366	25	0.9493	30	0.9578
6	0.7840	11	0.8837	16	0.9205	21	0.9396	26	0.9513	31	0.9592

Now we consider the two following choices for γ_k:

$$\gamma_k^I = 1 - \zeta_k, \quad \gamma_k^{II} = \frac{1-\zeta_k}{(1-\zeta_k^k)^2}. \tag{18}$$

Proposition 8 *Both parameters γ_k^I and γ_k^{II}, defined by (18), hold for (3).*

Proof Since $0 < \zeta_k < 1$ and $\lim_{k\to\infty} \zeta_k = 1$, see [19, Propositions 2.3, 2.4], one gets $0 < \gamma_k^I < 1$ and $\lim_{k\to\infty} \gamma_k^I = 0$. Using [19, Proposition 2.4, Inequality (2.17)] we have $\zeta_k < \frac{2k}{2k+1}$, which leads to $1 - \zeta_k > 1/(2k+1)$. Thus, γ_k^I holds for all three conditions in (3).

Based on [19, Proposition 3.3] we obtain $0 < \gamma_k^{II} < 1$. Using the following inequality, see [19, (3.10)],

$$\zeta_k \le \left(\frac{k-1}{2k-1}\right)^{1/k}, \tag{19}$$

we get

$$0 < \gamma_k^{II} = \frac{1-\zeta_k}{(1-\zeta_k^k)^2} \le \frac{1-\zeta_k}{\left(\frac{k}{2k-1}\right)^2}. \tag{20}$$

Therefore, using (20), we conclude that $\lim_{k\to\infty} \gamma_k^{II} = 0$. Combining $\gamma_k^{II} > 1 - \zeta_k$ and $\zeta_k < \frac{2k}{2k+1}$, one easily gets

$$\gamma_k^{II} > 1 - \zeta_k > \frac{1}{2k+1},$$

which shows that $\sum_k^\infty \gamma_k^{II} = +\infty$. Therefore, all three conditions in (3) are satisfied by γ_k^{II}. □

3.2. Using additional information

The use of a priori information (like nonnegativity) when solving an inverse problem is a well-known technique to improve the quality of the reconstruction. Let Ω be a closed convex set in $\mathbb{R}^n$ such that $\Omega \cap S \neq \emptyset$. We consider the following algorithm, which is the projected version of Algorithm 1:

Algorithm 9 Projected SeqBI (PSeqBI)
Initialization: $x^0 \in \mathbb{R}^n$ *is arbitrary.*
Iterative Step: *For given* x^k, *compute*

$$\begin{aligned} x^{k,0} &= x^k, \\ x^{k,s} &= T_{k,s}(x^{k,s-1}), \ \ s = 1, \dots, q, \\ x^{k+1} &= P_\Omega(x^{k,q}), \end{aligned}$$

where P_Ω denotes orthogonal projection onto Ω.

For the case $q = 1$, i.e. the full simultaneous method, the convergence analysis of Algorithm 9 is shown in [3] where the relaxation parameters are constant during iterations. However, there is no convergence analysis where the relaxation parameters are updated during the iterations or cycles. Consequently, convergence analysis for Algorithm 9 is not available and it remains an open problem.

4. Experimental issues

This section consists of two subsections. We first discus the 'fast semiconvergence' phenomenon and implementation of Algorithm 1 in Subsection 4.1. We report on some numerical tests (six different tests) with examples taken from tomographic imaging. We use the AIR Tools software package [22] and the SNARK09 software package [27].

In Subsection 4.2 we consider the following notations and concepts. For the choices of M_s we always use Cimmino's method. The noise-free right-hand side of the linear system of equations (1) was taken as the product of the matrix and the reshaped vector. We use additive independent Gaussian noise of mean 0 and relative noise levels ($\|\delta b\|/\|\bar{b}\|$) 5% and 10% where $b = \bar{b} + \delta b$. In all numerical tests, noisy data are used. We define the relative error of x^k as $\|x^k - x^*\|/\|x^*\|$ where x^* is the original image, i.e. $Ax^* = \bar{b}$. Apart from the unconstrained case we also consider the constrained version where Ω is the nonnegative orthant. Note that the positiveness of the sought solution is a natural constraint in computerized tomography applications. Since our theoretical results show that Algorithm 1 (using strategies (5) and (6)) converges to $\bar{x} + P_{N(A)}(x^0)$ and using Remark 2, one should select the starting point x^0 such that $P_{N(A)}(x^0) = 0$. We take $x^0 = 0$ in all numerical tests. We also compare different strategies, (5), (6), and $\lambda = 1$ (without relaxation parameter), where Algorithm 1 is used. Furthermore, the results of the CGLS method and error minimizing relaxation strategy (EMR) [31, (3.16), case s = 1] are compared with Algorithm 1. When CGLS is used, the linear system (1) is scaled by $W^{1/2}$, i.e. $W^{1/2}A$ and $W^{1/2}b$ are used instead of A and b, respectively. Here, W denotes the weight SPD matrix defined by the fully simultaneous Cimmino method.

We next give some notes on the fast semiconvergence phenomenon and implementation of Algorithm 1.

4.1. Fast semiconvergence and implementation issues

To accelerate the algorithm, where noise-free data are used, one may choose a proper starting point x^0 or "efficient" order in blocks and rows of A or both of them. Using a proper starting point and efficient ordering may have a strong effect on the practical performance of algebraic reconstruction techniques; see [24, 34]. However, using noisy data may give the opposite results because of the semiconvergence phenomenon. Figure 3 shows the relative errors of iterates using Algorithm 1, relaxation parameter $\lambda = 30$, and starting point $x^0 = 0.5x^*$ (i.e. a proper starting point). Here we use our first test data with two noise levels. As seen in the figure, increasing the noise level (10%) leads to "faster semiconvergence" compared with the 5% noise. Our intention by fast semiconvergence is to get diverging behavior of relative error within early iterations. As seen in Figure 3, the iteration vector approaches a regularized solution at the first iteration (because of the proper starting point). However, continuing the iteration leads to iteration vectors that are deteriorated by noise. Therefore, choosing a proper starting point and reordering the rows and blocks may make fast semiconvergence. Since our purpose is to postpone the semiconvergence phenomenon, we do not discuss how to choose the starting point or block reordering in this paper. As we explained before, in our numerical tests, we only consider the starting point $x^0 = 0$. We also propose natural ordering of the rows, which arises from treating rays according to the position of the pixels in the projections. It should be mentioned that, depending on the different applications, various numbers of blocks can be used. In medical imaging, e.g., breast X-ray tomography, the X-ray dose should be limited. Therefore, few-projection measurements, i.e. few numbers of blocks, have been considered. On the other hand, there is no limitation on X-ray dose and many projection measurements can be used in industrial imaging.

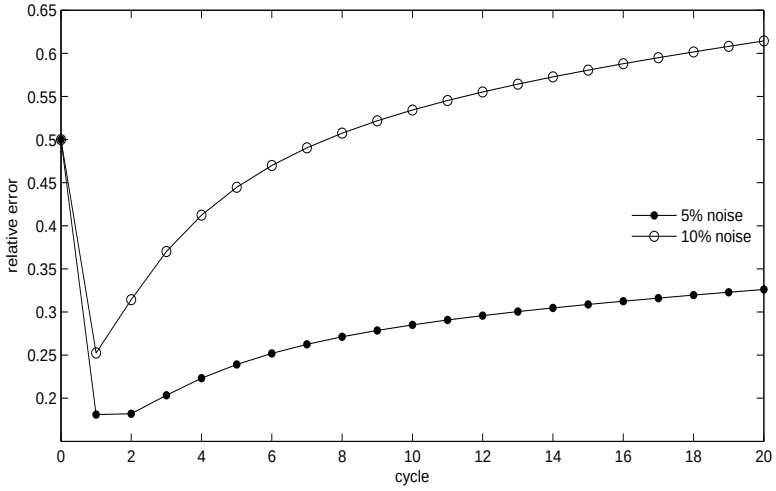

Figure 3. Fast semiconvergence of SeqBI iterations with constant relaxation parameter $\lambda = 30$ and starting point $x^0 = 0.5x^*$, for two different noise levels (5% and 10%).

To implement Algorithm 1, we assume that $\{M_s\}_{s=1}^q$ are diagonal matrices as in the Cimmino and CAV methods. Therefore, we only need two matrix-vector multiplications and one component-wise multiplication in each iteration, i.e. computing $y = b^s - A_s x$ as one matrix-vector multiplication, computing $z = M_s y$ as component-wise multiplication, and finally another matrix-vector multiplication $A_s^T z$. The next part of the

algorithm deals with computing relaxation parameters (5) and (6). As we mentioned before, the sequences (18) can be cheaply precalculated. Computing the largest singular value of $M_s^{1/2}A_s$ seems to be the most expensive part of our relaxation strategies. However, in general, computing the related singular values has to be done to obtain the acceptable range of relaxation parameters; see Theorems 3 and 5. Note that computing the square root of a matrix is expensive in general. However, since M_s is a diagonal matrix, taking the square root of its diagonal components gives $M_s^{1/2}$. Furthermore, the multiplication $M_s^{1/2}A_s$ is not so costly. To compute the largest singular values $\{\sigma_s\}_{s=1}^{q}$, one may use the power method. We have to mention that computation of the sequences (18) and all singular values $\{\sigma_s\}_{s=1}^{q}$ can be precalculated before the first iteration and we can use them for the rest of the iterations.

4.2. Numerical results

In our first test, using the AIR Tools software package, the original head phantom is discretized into 365×365 pixels. We take 88 projections (evenly distributed between 0 and 179 degrees) with 516 rays per projection. The resulting projection matrix A has dimension 40892×133225, so the system of equations is underdetermined.

Remark 10 *In image reconstruction from projections, some rays may not cross the reconstruction region (phantom). Therefore, the multiplication of the number of projections and rays may differ with the number of equations in (1).*

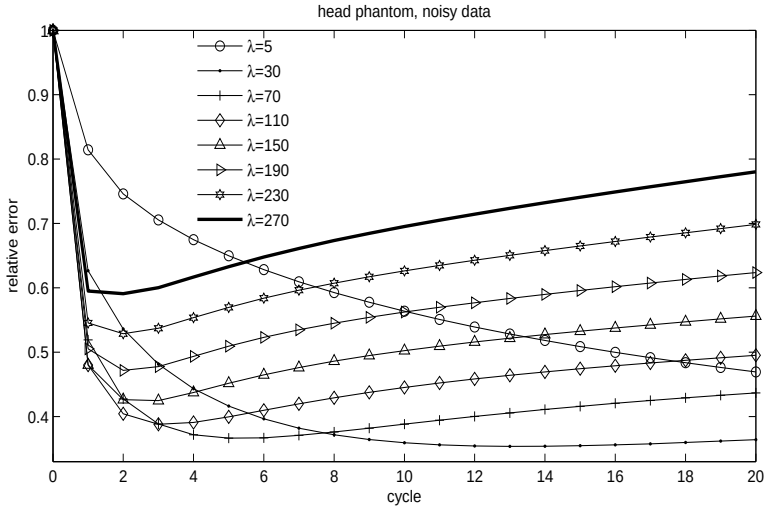

Figure 4. Relative error histories of SeqBI iterations with different relaxation parameters for 5% noise level.

Figure 4 shows the relative error behavior of Algorithm 1 for various relaxation parameters for a noise level of 5%. Based on Theorems 3 and 5, relaxation parameters have to lie in $0 < \lambda_k < 622.11$. In this figure we avoid considering relaxation parameters of less than 5 or larger than 270, which cause slow convergence and diverging behavior, respectively. As is seen, the proper relaxation parameter is $\lambda = 30$ and the semiconvergence phenomenon occurs at the 13th cycle. Our intention with the "proper relaxation parameter" is to get the smallest relative error within a few iterations and stable behavior for the rest of iterations.

Figure 5 (part a) shows the relative errors of different strategies with 5% noise level. The strategy (6) with $\gamma_k = \gamma_k^I$ gives an acceptable result compared with other strategies. Using the results shown in Figure 5 (part b), adding more noise to the right-hand side of (1) gives fast semiconvergence for constant relaxation parameter $\lambda = 30$ and the CGLS method, whereas our strategies show stable behavior. As is seen, using a nonnegativity constraint reduces the relative error and the strategy (6) with $\gamma_k = \gamma_k^I$ gives a stable behavior. In Figure 5 (parts a and b), using the EMR strategy [31, (3.16)] and constant relaxation parameter $\lambda = 1$ (without relaxation) gives very fast semiconvergence and slow convergence, respectively. Algorithm 1 with our strategies and $\lambda = 30$ seems more robust against noise than CGLS.

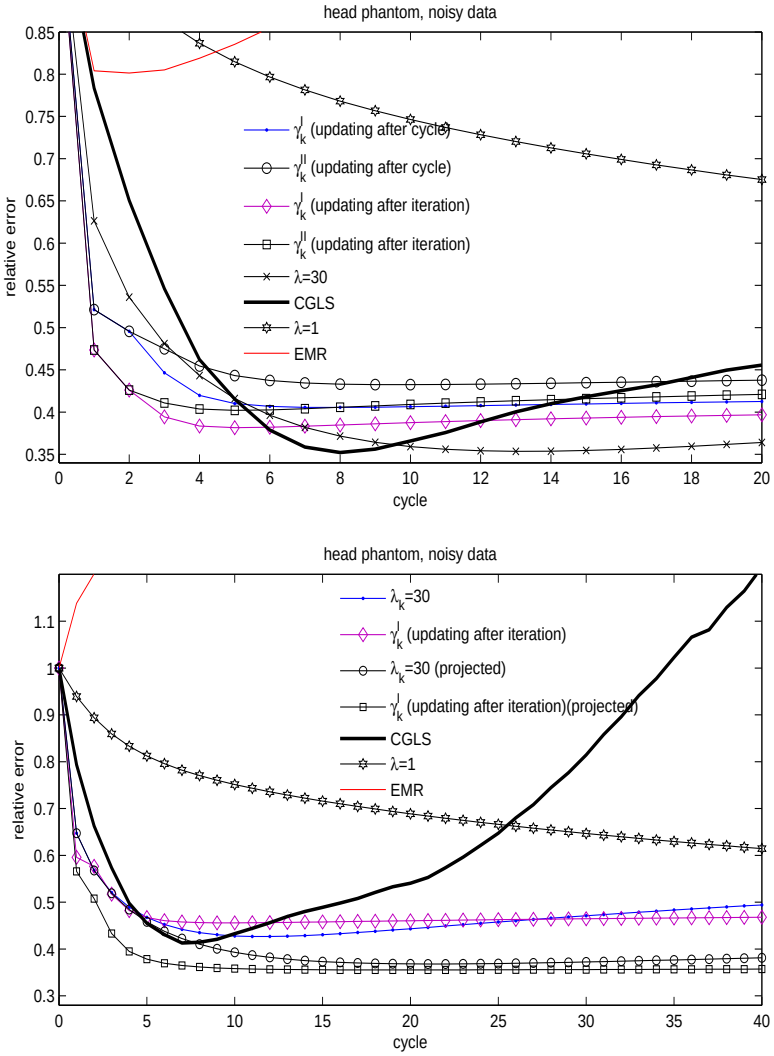

Figure 5. Relative error histories of SeqBI iterations with different relaxation strategies and the CGLS method, for two different noise levels of 5% [top (a)] and 10% [bottom (b)].

In the second test, we use the SNARK09 software package. We work with the standard head phantom from [23]. The phantom is discretized into 63×63 pixels, and 16 projections (evenly distributed between 0 and 174 degrees) with 99 rays per projection are used. The resulting matrix A has dimensions of 1376×3969,

so the system of equations is highly underdetermined. In this test we consider 5% noise for the right-hand side vector. Figure 6 shows the relative histories of Algorithm 1 with our different relaxation strategies. The strategies (5) and (6) with two choices for γ_k give similar results. As explained for the first test, Algorithm 1 with our strategies is more robust against noise than CGLS. Furthermore, using the EMR strategy [31, (3.16)] and $\lambda = 1$ (without relaxation) gives fast semiconvergence and slow convergence, respectively. Here, after the 12th cycles, the proper relaxation parameter $\lambda = 4.91$ gives results similar to those of our strategies. However, the strategies cause faster convergence behavior than the proper relaxation parameter within the first 10 cycles.

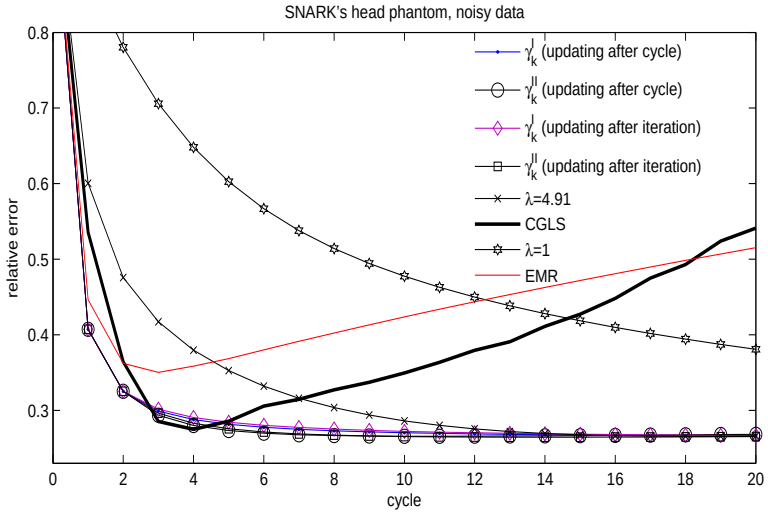

Figure 6. Relative error histories of SeqBI iterations with different relaxation strategies and CGLS method for 5% noise level.

We next consider four phantoms, i.e. 'threephases', 'fourphases', 'grains', and 'ppower', generated by the AIR Tools software package. All four phantoms are discretized into 225×225 pixels, and 36 projections (evenly distributed between 0 and 179 degrees) with 318 rays per projection are used. The resulting matrix A has dimensions of 10264×50625. Here, we again use 5% noise for the right-hand side vector. Figure 7 shows the relative error histories in SeqBI iterations using noisy data (5%) with different phantoms and relaxation strategies. In all results (top and bottom), using constant relaxation parameter $\lambda = 1$ (without relaxation) leads to slow convergence. The results of our strategies are almost equivalent for all four phantoms. In Figure 7 (parts a and b), the EMR strategy [31, (3.16)] shows fast convergence behavior within the first cycles and the semiconvergence phenomenon after the fourth cycle. Figure 7 (parts c and d) shows that the EMR strategy leads to fast semiconvergence, although it gives slow convergence within the first cycles. However, Figure 7 shows that the EMR strategy is unstable like the CGLS method.

Remark 11 *As explained in Theorems 3 and 5, the sequence of cycles converges to* $\bar{x}+P_{N(A)}(x^0)$ *where* $\bar{x} \in S$; *see (4). Since our numerical tests are done with noisy data, both theorems are still valid where the set* S *is defined with noisy data, i.e.* $S = \{x \in \mathbb{R}^n | A^T M A x = A^T M b\}$. *For this reason, our numerical results only show semiconvergence behavior.*

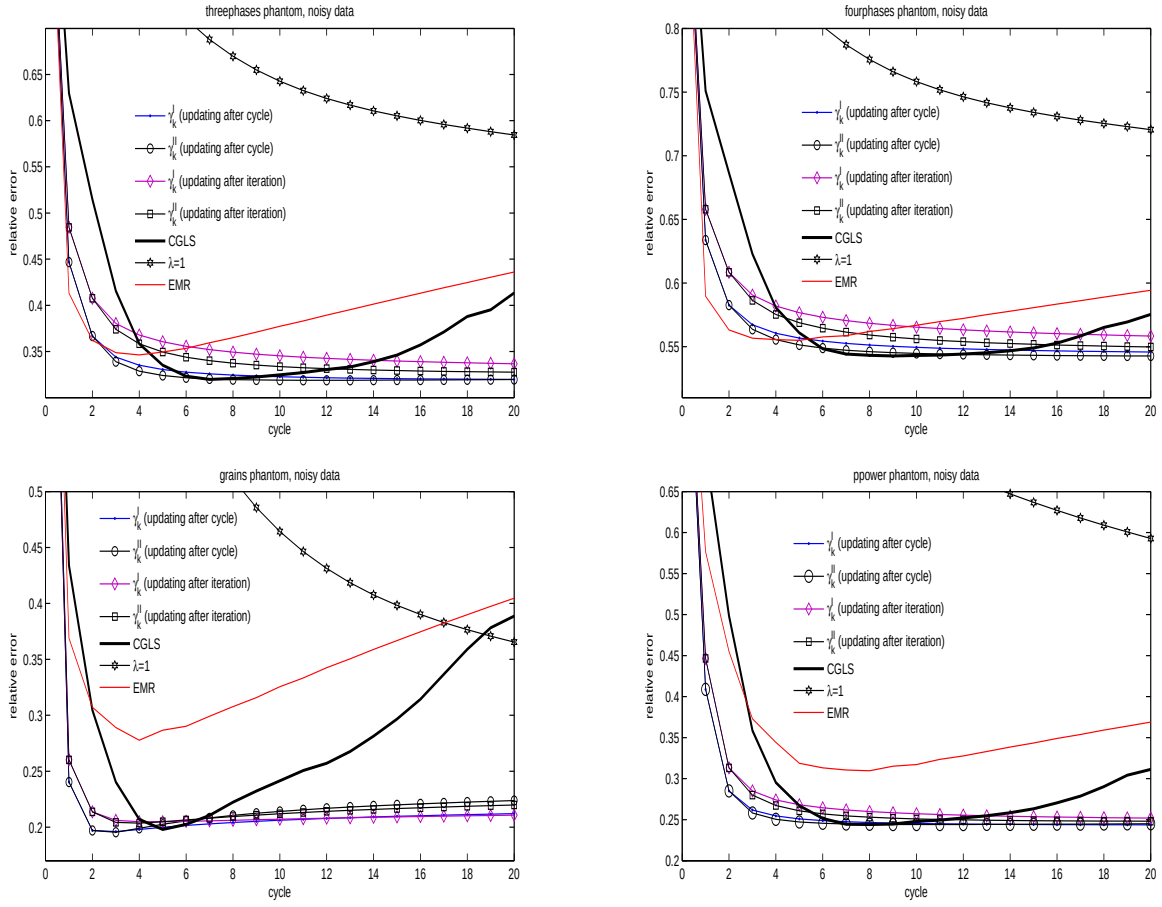

Figure 7. Relative error histories of SeqBI iterations with different relaxation strategies and CGLS method for 5% noise level. Top: threephases [left (a)] and fourphases [right (b)] phantoms; bottom: grains [left (c)] and ppower [right (d)] phantoms.

5. Conclusion

We introduce two strategies for picking relaxation parameters to make a delay in the semiconvergence phenomenon. Since noisy data corrupt the approximated solution after a few iterations (or cycles), we want to diminish the iteration vectors after a few iterations. We can cut off the iteration process by using our relaxation parameters, which converge to zero. Our tests show that the updating of relaxation parameters after each iteration has better performance than updating after each cycle.

Acknowledgments

We thank Tommy Elfving and Per Christian Hansen for their valuable comments. We thank the referees for constructive comments that helped to improve the presentation.

References

[1] Aharoni R, Censor Y. Block-iterative projection methods for parallel computation of solutions to convex feasibility problems. Linear Algebra Appl 1989; 120: 165-175.

[2] Bauschke HH, Borwein JM. On projection algorithms for solving convex feasibility problems. SIAM Rev 1996; 38: 367-426.

[3] Bertsekas DP. Nonlinear Programming. Belmont, MA, USA: Athena Scientific, 1995.

[4] Brianzi P, Di Benedetto F, Estatico C. Improvement of space-invariant image deblurring by preconditioned Landweber iterations. SIAM J Sci Comput 2008; 30: 1430-1458.

[5] Byrne C. Block-iterative interior point optimization methods for image reconstruction from limited data. Inverse Probl 2000; 16: 1405-1419.

[6] Censor Y, Chen W, Combettes PL, Davidi R, Herman GT. On the effectiveness of projection methods for convex feasibility problems with linear inequality constraints. Comput Optim Appl 2012; 51: 1065-1088.

[7] Censor Y, Elfving T. Block-iterative algorithms with diagonally scaled oblique projections for the linear feasibility problem. SIAM J Matrix Anal A 2002; 24: 40-58.

[8] Censor Y, Elfving T, Herman GT, Nikazad T. On diagonally-relaxed orthogonal projection methods. SIAM J Sci Comput 2008; 30: 473-504.

[9] Censor Y, Gordon D, Gordon R. BICAV: an inherently parallel algorithm for sparse systems with pixel-dependent weighting. IEEE T Med Imaging 2001; 20: 1050-1060.

[10] Censor Y, Gordon D, Gordon R. Component averaging: An efficient iterative parallel algorithm for large and sparse unstructured problems. Parallel Comput 2001; 27: 777-808.

[11] Censor Y, Zenios SA. Parallel Optimization: Theory, Algorithms, and Applications. New York, NY, USA: Oxford University Press, 1997.

[12] Cimmino G. Calcolo approssimato per le soluzioni dei sistemi di equazioni lineari. La Ricerca Scientifica 1938; 16: 326-333 (in Italian).

[13] Donoho D. Compressed sampling. IEEE T Inform Theory 2006; 52: 1289-1306.

[14] Eggermont PPB, Herman GT, Lent A. Iterative algorithms for large partitioned linear systems, with applications to image reconstruction. Linear Algebra Appl 1981; 40: 37-67.

[15] Elfving T. Block-iterative methods for consistent and inconsistent linear equations. Numer Math 1980; 35: 1-12.

[16] Elfving T, Hansen PC, Nikazad T. Semi-convergence and relaxation parameters for projected SIRT Algorithms. SIAM J Sci Comput 2012; 34: A2000-A2017.

[17] Elfving T, Hansen PC, Nikazad T. Semi-convergence properties of Kaczmarz's method. Inverse Probl 2014; 30: 1-16.

[18] Elfving T, Nikazad T. Properties of a class of block-iterative methods. Inverse Probl 2009; 25: 1-13.

[19] Elfving T, Nikazad T, Hansen PC. Semi-convergence and relaxation parameters for a class of SIRT algorithms. Electron T Numer Anal 2010; 37: 321-336.

[20] Engl HW, Hanke M, Neubauer A. Regularization of Inverse Problems. Dordrecht, the Netherlands: Kluwer Academic Publishers, 1996.

[21] Hansen PC. Rank-Deficient and Discrete Ill-Posed Problems: Numerical Aspects of Linear Inversion. Philadelphia, PA, USA: SIAM, 1998.

[22] Hansen PC, Saxild-Hansen M. AIR tools-A MATLAB package of algebraic iterative reconstruction methods. J Comput Appl Math 2012; 236: 2167-2178.

[23] Herman GT. Image Reconstruction from Projections: The Fundamentals of Computerized Tomography. New York, NY, USA: Academic Press, 1980.

[24] Herman GT, Meyer LB. Algebraic reconstruction techniques can be made computationally efficient. IEEE T Med Imaging 1993; 12: 600-609.

[25] Jiang M, Wang G. Convergence studies on iterative algorithms for image reconstruction. IEEE T Med Imaging 2003; 22: 569-579.

[26] Kaczmarz S. Angenäherte auflösung von systemen linearer gleichungen. Bulletin International de l'Académie Polonaise des Sciences et des Lettres 1937; 35: 355-357 (in German).

[27] Klukowska J, Davidi R, Herman GT. SNARK09: A software package for reconstruction of 2D images from 1D projections. Comput Meth Prog Bio 2013; 110: 424-440.

[28] Landweber L. An iterative formula for Fredholm integral equations of the first kind. Am J Math 1951; 73: 615-624.

[29] Nagy JG, Palmer KM. Steepest descent, CG, and iterative regularization of ill-posed problems. BIT 2003; 43: 1003-1017.

[30] Natterer F. The Mathematics of Computerized Tomography. Philadelphia, PA, USA: SIAM, 2001.

[31] Nikazad T, Abbasi M, Elfving T. Error minimizing relaxation strategies in Landweber and Kaczmarz type iterations. J Inverse Ill-Pose P 2017; 25: 35-56.

[32] Nikazad T, Davidi R, Herman GT. Accelerated perturbation-resilient block-iterative projection methods with application to image reconstruction. Inverse Probl 2012; 28: 1-19.

[33] Piana M, Bertero M. Projected Landweber method and preconditioning. Inverse Probl 1997; 13: 441-464.

[34] Sørensen HHB, Hansen PC. Multicore performance of block algebraic iterative reconstruction methods. SIAM J Sci Comput 2014; 36: C524-C546.

[35] Van der Sluis A, Van der Vorst HA. SIRT- and CG-type methods for the iterative solution of sparse linear least-squares problems. Linear Algebra Appl 1990; 130: 257-303.