

SUPPLEMENTARY DIRECTIONAL RELAXATIONS FOR THE ACCELERATION OF KACZMARZ'S PROJECTION METHOD

CONSTANTIN POPA*

Abstract. Starting from an extension of Kaczmarz's method, obtained by us in a previous paper, we introduce new directions for projections. We prove that by this, we don't modify the set of limit points of the original extended Kaczmarz algorithm. For the class of boundary value problems or integral equations of the first kind, we describe a method for constructing these new directions. It is based on considering coarser level of discretization for the initial problem. Some numerical experiments are also presented.

MSC 2000. 65F10, 65F20.

Keywords. Least-squares problems, Kaczmarz's projection algorithm, supplementary projections.

1. EXTENDED KACZMARZ PROJECTION ALGORITHM

Many important “real-world” problems give rise, after a suitable discretization to linear least-squares formulations as: find $x^* \in \mathbb{R}^n$ such that

$$(1) \quad \|Ax^* - b\| = \inf\{\|Ax - b\|, x \in \mathbb{R}^n\},$$

where A is an $m \times n$ real matrix and $b \in \mathbb{R}^m$ a given vector (by $\|\cdot\|$ we denoted the Euclidean norm and $\langle \cdot, \cdot \rangle$ will denote the corresponding scalar product on some space \mathbb{R}^q). In many cases, the matrix A is large, sparse (sometimes without a regular positioning of its nonzero entries) and ill-conditioned. Thus, trying to solve (1) by “direct” solvers, based on Singular Value or QR decompositions (see e.g. [4]) is not indicated, more appropriate being the “iterative” algorithms. In this paper we are concerned with a special class of iterative solvers for (1) – **successive projection algorithms**. S. Kaczmarz described for the first time such a method in [6], thus all the further developments and generalizations were usually called “Kaczmarz-like” algorithms (see [1], [2] for an overview in this sense). In [8] (see also [10]) we proposed an extension of the classical Kaczmarz's algorithm, that covers the general inconsistent case for (1). It will be called **Kaczmarz Extended projection method (KE**, for short) and will be briefly described in what follows. For this, we shall denote by B^t , $R(B)$, $N(B)$ the transpose, range and null space of a matrix B , respectively, by P_S the orthogonal projection onto a closed convex set $S \subset \mathbb{R}^q$ and

*“OVIDIUS” University, Faculty of Mathematics and Computer Science, Blvd. Mamaia 124, 900527 Constanța, Romania, e-mail: cpopa@univ-ovidius.ro.

by $a_i \in \mathbb{R}^n$, $\alpha_j \in \mathbb{R}^m$ the i -th row and j -th column of A , for which we shall suppose that

$$(2) \quad a_i \neq 0, \alpha_j \neq 0, \forall i = 1, \dots, m, j = 1, \dots, n.$$

Moreover, all the vectors appearing in the paper will be considered as column vectors. Let $f_i(b; \cdot), F(b; \cdot) : \mathbb{R}^n \rightarrow \mathbb{R}^n$, $\varphi_j, \Phi : \mathbb{R}^m \rightarrow \mathbb{R}^m$ be the applications defined by

$$(3) \quad f_i(b; x) = x - \frac{\langle x, a_i \rangle - b_i}{\|a_i\|^2} a_i, \quad F(b; x) = (f_1 \circ \dots \circ f_m)(b; x),$$

$$(4) \quad \varphi_j(y) = y - \frac{\langle y, \alpha_j \rangle}{\|\alpha_j\|^2} \alpha_j, \quad \Phi(y) = (\varphi_1 \circ \dots \circ \varphi_n)(y).$$

ALGORITHM 1 (**KE**). Let $y^0 = b$ and $x^0 \in \mathbb{R}^n$; for $k = 0, 1, 2, \dots$ do

$$(5) \quad y^{k+1} = \Phi(y^k), \quad b^{k+1} = b - y^{k+1}, \quad x^{k+1} = F(b^{k+1}; x^k).$$

The following convergence result was proved in [10].

THEOREM 2. For any matrix A satisfying (2) and any vector $b \in \mathbb{R}^m$ the following are true:

(i) the sequence $(x^k)_{k \geq 0}$ generated by the algorithm **KE** converges and

$$(6) \quad \lim_{k \rightarrow \infty} x^k = P_{N(A)}(x^0) + GP_{R(A)}(b),$$

where the $n \times m$ matrix G is a generalized inverse of A ;

(ii) we have the equality

$$(7) \quad LSS(A; b) = \{P_{N(A)}(x^0) + GP_{R(A)}(b), x^0 \in \mathbb{R}^n\},$$

where $LSS(A; b)$ is the set of all solutions of (1).

REMARK 1. Despite of the fact that some relaxation parameters can be introduced in (5) (see e.g. [9]), the above algorithm **KE** is not enough fast for becoming an efficient iterative solver. For this, in the next section of the paper we shall emphasize some possible ways to accelerate it, by focussing our attention on a method based on the addition of some supplementary directions for projection in the first and third steps of (5). \square

2. EXTENDED KACZMARZ WITH SUPPLEMENTARY PROJECTIONS

A widely used method to accelerate convergence for square nonsingular systems of the form $Ax^* = b$ is to (formally) transform it by means of some invertible matrices Q and P as follows

$$(8) \quad Ax^* = b \Leftrightarrow (QAP)(P^{-1}x^*) = Qb.$$

In the paper [11] we extended this technique to problems of the form (1), by considering the transformed least-squares formulation: find $z^* \in \mathbb{R}^n$ such that

$$(9) \quad \|(QAP)z^* - Qb\| = \inf\{\|(QAP)z - Qb\|, z \in \mathbb{R}^n\}.$$

But, although some possible choices exist for Q and P such that (9) and (1) are equivalent in some sense, these matrices are forced to satisfy some orthogonality assumptions which restrict the applicability of the method.

Another possibility for improving the behavior of algorithm (5) is to use, instead of the classical orthogonal projections f_i and φ_j from (3) and (4) oblique or generalized oblique ones, of the form

$$(10) \quad f_i^G(b; x) = x - \frac{\langle x, a_i \rangle - b_i}{\|a_i\|_{G^{-1}}^2} G^{-1} a_i, \quad \varphi_j^S(y) = y - \frac{\langle y, \alpha_j \rangle}{\|\alpha_j\|_{S^{-1}}^2} S^{-1} \alpha_j,$$

where G and S are symmetric and positive semidefinite matrices, G^{-1} and S^{-1} some generalized inverses and $\|\cdot\|_{G^{-1}}$ $\|\cdot\|_{S^{-1}}$ the corresponding ‘‘energy’’ seminorms. These techniques are essentially used in the field of image processing and they usually have a ‘‘regularization’’ property rather than a convergence speed-up (see in this sense [2] and references therein).

In the present paper, we shall consider another possibility to improve convergence properties of the above algorithm **KE**. It is based on the introduction, in steps 1 and 3 of (5) of supplementary directions for projection. If these directions are properly chosen we can obtain an improvement of the (generalized) condition number of A , which will generate a better behavior of **KE**. But, there is a very important problem that must be clarified before starting to analyze this technique. Indeed, adding in (5) new directions for projection is (formally) equivalent with adding new rows and/or columns to the matrix A and such kind of transformations will modify the initial problem (1). The surprising result which we shall prove in the rest of this section is that this doesn't happen. More clear, we shall prove that the new algorithm will generate sequences of approximations converging to the solutions of the initial problem (1). For simplifying the notations and the proofs, we shall consider the case of only one direction introduced in steps 1 and 3 from (5), but the extension to an arbitrary number is straightforward. Thus, let $a_0 \in \mathbb{R}^n$, $\alpha_0 \in \mathbb{R}^m$ and $b_0 \in \mathbb{R}$ be a new row, column and component of the right hand side b , defined as linear combinations of the old ones, i.e.

$$(11) \quad a_0 = \sum_{i=1}^m \beta_i a_i \neq 0, \quad \alpha_0 = \sum_{j=1}^n \gamma_j \alpha_j \neq 0, \quad b_0 = \sum_{i=1}^m \beta_i b_i, \quad \beta_i, \gamma_j \in \mathbb{R}.$$

We shall define the new projections $f_0(b; \cdot)$ and φ_0 (see (3) and (4)) by

$$(12) \quad f_0(b; x) = x - \frac{\langle x, a_0 \rangle - b_0}{\|a_0\|^2} a_0, \quad \varphi_0(y) = y - \frac{\langle y, \alpha_0 \rangle}{\|\alpha_0\|^2} \alpha_0,$$

and the new applications $F_0(b; \cdot)$ and Φ_0 by

$$(13) \quad F_0(b; x) = (f_0 \circ f_1 \circ \cdots \circ f_m)(b; x), \quad \Phi_0(y) = (\varphi_0 \circ \varphi_1 \circ \cdots \circ \varphi_n)(y).$$

REMARK 2. The fact that f_0 and φ_0 were introduced after the projections f_1 and φ_1 , respectively is not restrictive because the set of solutions for (1) and

the set of limit points of the algorithm **KE** are both invariant under (left-right) orthogonal transformations of A . \square

According to the above definitions and notations, the new algorithm of the form (5), with supplementary directions a_0 and α_0 for projection is the following.

ALGORITHM 3 (KE_0). Let $y^0 = b$ and $x^0 \in \mathbb{R}^n$; for $k = 0, 1, 2, \dots$ do

$$(14) \quad y^{k+1} = \Phi_0(y^k), \quad b^{k+1} = b - y^{k+1}, \quad x^{k+1} = F_0(b^{k+1}; x^k).$$

Let now T be the $(m+1) \times m$ matrix with the rows (in this order) $\beta^t, \varepsilon_1^t, \dots, \varepsilon_m^t$, where $\beta = (\beta_1, \dots, \beta_m)^t \in \mathbb{R}^m$ (see (11)) and $\{\varepsilon_1, \dots, \varepsilon_m\}$ is the canonical basis in \mathbb{R}^m . Similarly, let S be the $(n+1) \times n$ matrix with the rows (in this order) $\gamma^t, \tau_1^t, \dots, \tau_n^t$, where $\gamma = (\gamma_1, \dots, \gamma_n)^t \in \mathbb{R}^n$ (see again (11)) and $\{\tau_1, \dots, \tau_n\}$ is the canonical basis in \mathbb{R}^n . Then, we define the matrices \bar{A}, \hat{A} and the vector $\bar{b} \in \mathbb{R}^{m+1}$ by

$$(15) \quad \bar{A} = TA, \quad \bar{b} = Tb, \quad \hat{A} = AS^t.$$

REMARK 3. The matrices \bar{A}, \hat{A} and the vector \bar{b} are of the form

$$(16) \quad \bar{A} = \begin{bmatrix} a_0^t \\ a_1^t \\ \vdots \\ a_m^t \end{bmatrix}, \quad \hat{A} = \text{col}\{\alpha_0, \alpha_1, \dots, \alpha_n\}, \quad \bar{b} = (b_0, b_1, \dots, b_m). \quad \square$$

Let $\bar{F}(\bar{b}; \cdot), \hat{\Phi}$ be the applications defined as in (3) and (4), but with respect to the matrices \bar{A} and \hat{A} , instead of A . From (11)–(13) we then have

$$(17) \quad \bar{F}(\bar{b}; x) = F_0(b; x), \quad \forall x \in \mathbb{R}^n, \quad \hat{\Phi}(y) = \Phi_0(y), \quad \forall y \in \mathbb{R}^m.$$

Moreover, the algorithm KE_0 can be written in the following form similar to (14)

$$(18) \quad y^{k+1} = \hat{\Phi}(y^k), \quad \bar{b}^{k+1} = T(b - y^{k+1}), \quad x^{k+1} = \bar{F}(\bar{b}^{k+1}; x^k).$$

Let also $\bar{Q}, \tilde{Q}, \bar{R}, \bar{G}$ and $\tilde{\Phi}$ be the matrices appearing in the convergence analysis of the algorithm **KE** from [10], but constructed with respect to \bar{A} and \hat{A} , instead of A , respectively.

LEMMA 4. The following are true

$$(19) \quad N(\hat{A}^t) = N(A^t), \quad Ty^{k+1} = T(P_{N(A^t)}(b) + \tilde{\Phi}^{k+1}b).$$

Proof. The first equality holds from (15) and the definition of the matrix S . For the second one, we firstly note that (see [13]) $\hat{\Phi} = \tilde{\Phi} \oplus P_{N(\hat{A}^t)} = \tilde{\Phi} \oplus P_{N(A^t)}$. Then, using the construction of the sequence $(y^k)_{k \geq 0}$ in (18) we get

$$Ty^{k+1} = T(\hat{\Phi}^{k+1}b) = T\left((P_{N(A^t)} + \tilde{\Phi})^{k+1}(b)\right) = T\left(P_{N(A^t)}(b) + \tilde{\Phi}^{k+1}b\right),$$

i.e. the second equality in (19). \square

Now we are able to prove a convergence result similar with Theorem 1 for the new algorithm KE_0 (see (14) or (18)).

THEOREM 5. *For any matrix A satisfying (2), any vector $b \in \mathbb{R}^m$ and any new elements a_0, α_0, b_0 as in (11) the following are true:*

(i) *the sequence $(x^k)_{k \geq 0}$ generated by the algorithm KE_0 converges and*

$$(20) \quad \lim_{k \rightarrow \infty} x^k = P_{N(A)}(x^0) + \bar{G}TP_{R(A)}(b),$$

(ii) *we have the equality*

$$(21) \quad LSS(A; b) = \{P_{N(A)}(x^0) + \bar{G}TP_{R(A)}(b), x^0 \in \mathbb{R}^n\}.$$

Proof. (i) For simplifying the presentation, we shall use the notation b_A for the vector $P_{R(A)}(b)$. Then, as in [10] we get

$$(22) \quad \begin{aligned} x^{k+1} - (P_{N(\bar{A})}(x^0) + \bar{G}Tb_A) &= \\ &= \tilde{Q}[x^k - (P_{N(\bar{A})}(x^0) + \bar{G}Tb_A)] + \bar{R}Tb^{k+1} - \bar{R}Tb_A. \end{aligned}$$

Then, by using (19) we obtain

$$\bar{R}Tb^{k+1} - \bar{R}Tb_A = \bar{R}T\bar{b}_A - \bar{R}Ty^{k+1} = -\bar{R}T\tilde{\Phi}^{k+1}b$$

which together with (22) gives

$$x^{k+1} - (P_{N(\beta A)}(x^0) + \bar{G}Tb_A) = \tilde{Q}[x^k - (P_{N(\bar{A})}(x^0) + \bar{G}Tb_A)] - \bar{R}T\tilde{\Phi}^{k+1}b.$$

Then, by using similar arguments as in the proof of Theorem 3.1 from [10] and observing that $N(\bar{A}) = N(A)$ we get (20).

(ii) For a consistent problem of the form (1) we shall denote by $S(A; b)$ the set of its (classical) solutions. Then (see e.g. [4])

$$(23) \quad LSS(A; b) = S(A; b_A).$$

But, using again the equality $N(\bar{A}) = N(A)$ and the construction of \bar{A} , we get $S(A; b_A) = S(\bar{A}; Tb_A)$. Thus, from (23) we obtain

$$(24) \quad LSS(A; b) = S(\bar{A}; Tb_A).$$

According to the characterization (7) (see also [10]) we know that $x_{LS} = Gb_A$ (where x_{LS} is the unique minimal norm solution of (1)), thus

$$(25) \quad x_{LS} = \bar{G}Tb_A.$$

Let $LSS_0(A; b)$ be the set of all limit points of the algorithm KE_0 (see (20)), i.e.

$$(26) \quad LSS_0(A; b) = \{P_{N(A)}(x^0) + \bar{G}Tb_A\}.$$

Then, from (25)–(26) we obtain

$$LSS_0(A; b) = S(\bar{A}; Tb_A),$$

which together with (24) gives (21) and the proof is complete. \square

Let us now suppose that the problem (1) is consistent, A is with nonzero rows and a_0, b_0 are as in (11). We then consider the classical Kaczmarz algorithm with supplementary projections: let $x^0 \in \mathbb{R}^n$; for $k = 0, 1, 2, \dots$ do

$$(27) \quad x^{k+1} = \bar{F}(\bar{b}; x^k),$$

with \bar{b} and $\bar{F}(\bar{b}; \cdot)$ from (16) and (17). Then, the following consequence of the above Theorem 3 holds.

COROLLARY 6. *In the above hypothesis, the sequence $(x^k)_{k \geq 0}$ generated by the algorithm (27) converges and*

$$(28) \quad \lim_{k \rightarrow \infty} x^k = P_{N(A)}(x^0) + \bar{G}Tb.$$

Moreover, we have the equality

$$(29) \quad S(A; b) = \{P_{N(A)}(x^0) + \bar{G}Tb, x^0 \in \mathbb{R}^n\}.$$

3. THE CONSTRUCTION OF SUPPLEMENTARY DIRECTIONS

Although it is possible to construct supplementary directions for projection in the general case of an inconsistent problem like (1) (see [12]), we shall refer in this section to the consistent case, in which the classical Kaczmarz algorithm (27) applies. We have chosen this case for presentation in the present paper because it refers to a very important class of problems – discretizations of boundary value problems or integral equations of the first kind. In both these cases we start from an initial linear operator equation

$$(30) \quad Lu = f,$$

formulated on a (real) Hilbert space H of functions defined on a given domain $\Omega \subset \mathbb{R}^q$ (usually $q = 1, 2, 3$). If (30) is a boundary value problem we shall refer to a classical finite element or finite differences discretization, whereas it is a first kind integral equation, we shall refer at some Galerkin-like discretization techniques as those described in [7] (see also [3]). The common idea for both cases is to consider an initial set of linearly independent functions $\{\phi_1, \dots, \phi_n\} \subset H$ and to “project” the (continuous) problem (30) onto the vector space generated by $\{\phi_1, \dots, \phi_n\}$, denoted here by G_n . Then, we get the “discrete” formulation of (30): find $u^n = \sum_{i=1}^n x_i \phi_i \in G_n$ such that

$$(31) \quad (Lu^n, \phi_i) = (f, \phi_i), \forall i = 1, \dots, n,$$

where (\cdot, \cdot) is the scalar product on H . From (31) we get the associated linear system

$$(32) \quad Ax = b,$$

with A an $n \times n$ invertible matrix, $b \in \mathbb{R}^n$ given by

$$(33) \quad (A)_{ij} = (L\phi_j, \phi_i), \quad b_i = (f, \phi_i)$$

and $x = (x_1, \dots, x_n)^t \in \mathbb{R}^n$ contains the (unknown) components of the approximate solution u^n from (31), with respect to the basis $\{\phi_1, \dots, \phi_n\}$. In most of the cases, the system (33) is big, sparse and ill-conditioned. This will determine a “bad” behavior for both direct and iterative solvers. Kaczmarz's iteration (27) being a “row-action method” (i.e. a method that uses only one row of A at a time, see e.g. [1]) is suitable to be used for big and sparse matrices A , but the ill-conditioning of this will “slow” its convergence. In order to improve it, we can consider new directions for projection which, in our (consistent) case (32) is equivalent to “adding” new rows to A and corresponding new components to b , according to (11). In this sense we can use the following idea, coming from the field of “multigrid algorithms” (see [5]) that is, we consider a new set of functions $\{\phi_{n+1}, \dots, \phi_m\}$ which are linear combinations of the initial ones, i.e.

$$(34) \quad \phi_k = \sum_{i=1}^n \alpha_{ki} \phi_i, \quad k = n+1, \dots, m.$$

These functions are obtained by using “coarser” levels of discretization for Ω (see the next section). Then, we “extend” the $n \times n$ matrix A and $b \in \mathbb{R}^n$ from (33) to the $m \times n$ matrix \bar{A} and $\bar{b} \in \mathbb{R}^m$, defined by

$$(35) \quad (\bar{A})_{ij} = (L\phi_j, \phi_i), \quad \bar{b}_i = (f, \phi_i), \quad i = 1, \dots, m, \quad j = 1, \dots, n$$

and we consider the (full-rank “overdetermined”) system

$$(36) \quad \bar{A}x = \bar{b}.$$

Because of (34), (35) and the invertibility of A , it results that the systems (32) and (36) have the same unique solution. Unfortunately, we have not yet a systematic proof for the fact that the matrix \bar{A} is “better conditioned” than the initial one A , but we can send the reader (for at least some comments in this sense) to the paper [5]. We shall see in the next section of the paper that such a construction determines a very good improvement of Kaczmarz's algorithm convergence properties.

4. NUMERICAL EXPERIMENTS

We made our experiments on the following one dimensional convection-diffusion equation

$$(37) \quad \begin{cases} -u''(t) + \alpha u'(t) = f(t), & t \in \Omega = (0, 1), \\ u(0) = u(1) = 0 \end{cases}$$

where $\alpha \geq 0$ is the convection coefficient. For $n = 2^q, q \geq 1$ we considered a uniform discretization of $\Omega = (0, 1)$ with the mesh size $h = \frac{1}{n}$ and the discretization points

$$(38) \quad t_i = i \cdot h, \quad i = 0, \dots, n.$$

As discretization functions we used the one dimensional piecewise linear polynomial functions defined by

$$(39) \quad \phi_i(t) = \begin{cases} \frac{t-t_{i-1}}{h}, & t \in [t_{i-1}, t_i], \\ \frac{t_{i+1}-t}{h}, & t \in (t_i, t_{i+1}], \\ 0, & \text{else.} \end{cases} \quad i = 1, \dots, n-1.$$

Then, we considered the variational formulation of (37) on the Sobolev space $H = H_0^1(\Omega)$

$$(40) \quad (u', v')_2 + \alpha (u', v)_2 = (f, v)_2, \quad \forall v \in H_0^1(\Omega),$$

where by $(\cdot, \cdot)_2$ we denoted the scalar product in $L^2(\Omega)$. By “projecting” (40) onto the subspace $G_{n-1} \subset H_0^1(\Omega)$ generated by the basis functions ϕ_i from (39) we obtained the linear system (32), with A a square tridiagonal matrix of dimension $n-1$ and $b \in \mathbb{R}^{n-1}$ given by

$$(41) \quad (A)_{ij} = ((\phi_j)', (\phi_i)')_2 + \alpha ((\phi_j)', \phi_i)_2, \quad b_i = (f, \phi_i)_2.$$

We solved this system for $n = 64$ and $\alpha = 0$ (i.e. the one dimensional Dirichlet problem) with the classical Kaczmarz algorithm (without supplementary projections) and the stopping rule

$$(42) \quad \|Ax^k - b\| \leq 10^{-3}.$$

The number of iterations for fulfilling (40) was greater than 10 000. Moreover, the spectral condition number of A was $k_2(A) = 1\,659.4$. Then, we constructed the new directions for projection as follows – we considered a “coarser” discretization of $\Omega = (0, 1)$ with the mesh size $2h$, i.e.

$$(43) \quad \hat{t}_i = i \cdot 2h, \quad i = 0, \dots, \frac{n}{2}$$

and the functions $\hat{\phi}_1, \dots, \hat{\phi}_{\frac{n}{2}}$, constructed as in (39), but with respect to (41). It can be easily proved that

$$(44) \quad \hat{\phi}_j = 0.5\phi_{2j-1} + \phi_{2j} + 0.5\phi_{2j+1}, \quad \forall j = 1, \dots, \frac{n}{2} - 1,$$

i.e. each $\hat{\phi}_j$ is a linear combination of $\phi_i, i = 1, \dots, n-1$. We re-denoted the new constructed functions as

$$(45) \quad \phi_{(n-1)+j} = \hat{\phi}_j, \quad j = 1, \dots, \frac{n}{2} - 1$$

and we defined the new $\frac{n}{2} - 1$ rows in the extended matrix \bar{A} from (35) and the corresponding new components of \bar{b} by (see (41) and (43))

$$(46) \quad (\bar{A})_{ij} = ((\phi_j)', (\phi_i)')_2 + \alpha ((\phi_j)', \phi_i)_2, \quad \bar{b}_i = (f, \phi_i)_2,$$

$i = n, \dots, n + \frac{n}{2} - 1, j = 1, \dots, n-1$.

REMARK 4. The previous idea can be repeated as many time as needed, by considering coarser levels of discretization with meshsizes $2^2h, 2^3h$, etc. All the new corresponding discretization functions will be linear combinations of the

form (42) of the previous ones, i.e. at the end, they will be linear combinations of the initial ones $\phi_i, i = 1, \dots, n - 1$. \square

Coming back to our particular case $n = 64$ we considered three coarser levels ($2h, 4h$ and $8h$) and we added at A and b the corresponding new rows and components constructed as before. In this way we obtained a matrix \bar{A} of dimensions 116×64 and a corresponding right hand side \bar{b} . We solved the system (36) with the Kaczmarz algorithm (27). The number of iterations for fulfilling (40) was 317 which is a much better value than the one obtained before. Moreover, the “generalized” spectral condition number of \bar{A} (defined as the square root of the ratio between the biggest and smallest singular values of it, see e.g. [4]) was $k_2(\bar{A}) = 431.7$, which is also much better than the previous $k_2(A)$.

REMARK 5. Although the results are very “promising”, we have to observe that the improvement obtained before for Kaczmarz's algorithm is not mesh-independent, i.e. the number of iterations for solving the extended system (36) still depends on the dimension n of the initial system (32) (see also [5]). Moreover, it also depends on the number of “coarser” levels used for the construction of the new directions. Work is in progress for eliminating also these aspects. \square

NOTE. All the numerical experiments have been performed with the Numerical Linear Algebra software package OCTAVE (freely available on Internet).

REFERENCES

- [1] CENSOR, Y., *Row-action methods for huge and sparse systems and their applications*, SIAM Review, **23**, no. 4, pp. 444–466, 1981.
- [2] CENSOR, Y. and STAVROS A. Z., *Parallel optimization: theory, algorithms and applications*, “Numer. math. and Sci. Comp.” Series, Oxford Univ. Press, New York, 1997.
- [3] ENGL., H. W., *Regularization methods for the stable solution of inverse problems*, Surv. Math. Ind., **3**, pp. 71–143, 1993.
- [4] GOLUB, G. H. and VAN LOAN, C. F., *Matrix computations*, The John's Hopkins Univ. Press, Baltimore, 1983.
- [5] GRIEBEL, M., *Multilevel algorithms considered as iterative methods on semidefinite systems*, SIAM J. Sci. Comp., **15**, no. 3, pp. 547–565, 1994.
- [6] KACZMARZ, S., *Angenaherte Auflosung von Systemen linearer Gleichungen*, Bull. Acad. Polonaise Sci. et Lettres **A**, pp. 355–357, 1937.
- [7] KRESS, R., *Linear integral equations*, Springer-Verlag, Berlin, 1989.
- [8] POPA, C., *Least-squares solutions of overdetermined inconsistent linear systems using Kaczmarz's relaxation*, Inter. J. Comp. Math., **55**, pp. 79–89, 1995.
- [9] POPA, C., *Extensions of block-projections methods with relaxation parameters to inconsistent and rank-deficient least-squares problems*, BIT, **38**, no. 1, pp. 151–176, 1998.
- [10] POPA, C., *Characterization of the solutions set of inconsistent least-squares problems by an extended Kaczmarz algorithm*, Korean J. Comp. Appl. Math., **6**, no. 1, pp. 51–64, 1999.
- [11] POPA, C., *Preconditioned Kaczmarz-extended algorithm with relaxation parameters*, Korean J. Comp. Appl. Math., **8**, no. 1, pp. 9–26, 2000.
- [12] POPA, C., *Direct and iterative Kaczmarz-like solvers*, submitted for publication, 2002.

-
- [13] TANABE, K., *Projection method for solving a singular system of linear equations and its applications*, Numer. Math., **17**, pp. 203–214, 1971.

Received by the editors: November 3, 1998.