Hindawi Publishing Corporation
Journal of Electrical and Computer Engineering
Volume 2010, Article ID 972794, 33 pages
doi:10.1155/2010/972794

Review Article

Milestones in the Development of Iterative Solution Methods

Owe Axelsson

Institute of Geonics AS CR, 70800 Ostrava, Czech Republic

Correspondence should be addressed to Owe Axelsson, owe.axelsson@it.uu.se

Received 10 June 2010; Accepted 27 August 2010

Academic Editor: Christian Schlegel

Copyright © 2010 Owe Axelsson. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Iterative solution methods to solve linear systems of equations were originally formulated as basic iteration methods of defect-correction type, commonly referred to as Richardson's iteration method. These methods developed further into various versions of splitting methods, including the successive overrelaxation (SOR) method. Later, immensely important developments included convergence acceleration methods, such as the Chebyshev and conjugate gradient iteration methods and preconditioning methods of various forms. A major strive has been to find methods with a total computational complexity of optimal order, that is, proportional to the degrees of freedom involved in the equation. Methods that have turned out to have been particularly important for the further developments of linear equation solvers are surveyed. Some of them are presented in greater detail.

1. Introduction

In many applications of quite different types appearing in various sciences, engineering, and finance, large-scale linear algebraic systems of equations arise. A particular type of problems appear in signal processing. This also includes nonlinear systems of equation, which are normally solved by linearization at each outer nonlinear iteration step, but they will not be further discussed in this paper.

Due to their high demand of computer memory and computer time, which can grow rapidly with increasing problem size, direct solution methods, such as Gaussian elimination, are in general not feasible unless the size of the problem is relatively small. In the early computer age, when available size of computer central memories was very small and the speed of arithmetic operations slow, this was found to be the case even for quite modest-sized problems.

Even for modern computers with exceedingly large memories and very fast arithmetics it is still an issue because nowadays one wants to solve much more involved problems of much larger sizes, for instance to enable a sufficient resolution of partial differential equation problems with highly varying (material) coefficients, such as is found in heterogeneous media. Presently problems with up to billions of degrees of freedom (d.o.f.) are solved. For instance, if an elliptic equation of elasticity type is discretized and solved on a 3D mesh with 512 meshpoints in each coordinate direction, then an equation of that size arises.

A basic iteration method to solve a linear system

$$A\mathbf{x} = \mathbf{b}, \tag{1}$$

where A is nonsingular, has the following form.

Given an initial approximation $\mathbf{x}^0$, for $k = 0, 1, \ldots$ until convergence, let $\mathbf{r}^k = A\mathbf{x}^k - \mathbf{b}$,$\mathbf{e}^k = -\tau\mathbf{r}^k$, $\mathbf{x}^{k+1} = \mathbf{x}^k + \mathbf{e}^k$. Here, $\tau > 0$ is a parameter to be chosen.

This method can be described either as a defect ($\mathbf{r}^k$)—correction ($\mathbf{e}^k$) method or, alternatively, as a method to compute the stationary solution of the evolution equation

$$\frac{d\mathbf{x}(t)}{dt} = A\mathbf{x}(t) - \mathbf{b}, \quad t > 0, \ \mathbf{x}(0) = \mathbf{x}^0, \tag{2}$$

by timestepping with time-step τ, that is,

$$\mathbf{x}(t+\tau) = \mathbf{x}(t) - \tau(A\mathbf{x}(t) - \mathbf{b}), \quad t = 0, \tau, \ldots. \tag{3}$$

Such methods are commonly referred to as Richardson iteration methods (e.g., see [1–4]). However, already in 1823 Gauss [5] wrote, "Fast jeden Abend mache ich eine neue Auflage des Tableau, wo immer leicht nachzuhelfen ist. Bei der Einförmigkeit des Messungsgeschäfts gibt dies immer eine angenehme Unterhaltung; man sieht daran auch immer gleich, ob etwas Zweifelhaftes eingeschlichen ist, was noch wünschenswert bleibt usw. Ich empfehle Ihnen diesen Modus zur Nachahmung. Schwerlich werden Sie je wieder direct eliminieren, wenigstens nicht, wenn Sie mehr als zwei Unbekannte haben. Das indirecte Verfahren läßt sich halb im Schlafe ausführen oder man kann während desselben an andere Dingen denken." (Freely translated, "I recommend this modus operandi. You will hardly eliminate directly anymore, at least not when you have more than two unknowns. The indirect method can be pursued while half asleep or while thinking about other things.")

It holds that

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \tau\left(A\mathbf{x}^k - \mathbf{b}\right), \tag{4}$$

or

$$\mathbf{e}^{k+1} = (I - \tau A)\mathbf{e}^k, \tag{5}$$

where $\mathbf{e}^k = \mathbf{x} - \mathbf{x}^k$ is the iteration error and $\mathbf{x}$ is the solution of (1).

Hence,

$$\mathbf{e}^k = (I - \tau A)^k \mathbf{e}^0, \quad k = 0, 1, \ldots. \tag{6}$$

For convergence of the method, that is $\mathbf{e}^k \to 0$, the parameter τ must in general be chosen such that $\rho := \|I - \tau A\| < 1$, where $\|\cdot\|$ is a matrix norm, subordinate to the chosen vector norm. (We remark here that this is not possible if A is indefinite.)

Let $\rho(\cdot)$ denote the spectral radius of a matrix, that is, the maximal absolute value of the eigenvalues of the matrix.

If A is self–adjoint, then it can be shown that $\rho(A) = \|A\|_2 = \sqrt{\rho(A^*A)}$, where $\|\cdot\|_2$ denotes the matrix norm subordinate to the Euclidian vector norm. For general, nonsymmetric matrices it has been shown (e.g., see [6, page 162]) that there exist matrix norms that are arbitrarily close to the spectral radius. These can, however, correspond to an unnatural scaling of the matrix.

The rate of convergence is determined by the convergence factor ρ. For symmetric positive definite matrices, the optimal value of τ to minimize ρ, is $\tau = 2/(\lambda_1 + \lambda_n)$, where λ_1, λ_n are the extreme eigenvalues of A. Normally, however, the eigenvalues are not available.

As an example, for second order elliptic diffusion type of problems in $\Omega^d (d = 2, 3)$ using a standard central difference or a finite element method, the spectral condition number $\lambda_n/\lambda_1 = O(h^{-2})$, where h is the (constant) meshsize parameter. Hence, the number of iterations to reach a relative accuracy ε is of order $O(h^{-2})|\log \varepsilon|)$, $h \longrightarrow 0$.

Since each iteration uses $O(h^{-d})$ elementary arithmetic operations, this shows that the total number of operations needed to reduce the error to a given tolerance is of order $O(h^{-d-2})$. This is in general smaller than for a direct solution method when $d \geq 2$, but still far from the optimal order, $O(h^{-d})$, that we aim at.

To improve on this, often a splitting of the matrix A is used. It is readily shown that for any initial vector, the number of iterations required to get a relative residual, $\|\mathbf{r}^k\|/\|\mathbf{r}^0\| < \varepsilon$, for some $\varepsilon, 0 < \varepsilon < 1$, is at most $k_{it} = \lceil \ln(1/\varepsilon)/\ln(1/\rho) + 1 \rceil$, where $\lceil\,\rceil$ denotes the integer part. Frequently, $\rho = 1 - c\delta^r$, where c is a constant, r is a positive integer, often $r = 2$ and δ is a small number, typically $\delta = 1/n$, which decreases with increasing problems size n. This implies, that the number of iterations is propotional to $(1/\delta)^r$, which number increases rapidly when $\delta \longrightarrow 0$.

For $\tau = 1$, the splitting $A = C - R$ of A in two terms is used, where C is nonsingular. The iterative method (4) then takes the form

$$C\mathbf{x}^{k+1} = R\mathbf{x}^k + \mathbf{b}, \quad k = 0, 1, \ldots. \tag{7}$$

Method (7) is convergent if $\rho(C^{-1}R) < 1$. Splitting methods will be discussed in Section 2.

Let $B = C^{-1}R$. If $\|B\|$ is known and $\|B\| < 1$, we can use the following theorem to get a test when the iteration error is small enough, that is, when to stop the iterations.

Theorem 1. *Let $\|B\| < 1$, $B = C^{-1}R$, and $\mathbf{x}^m$ be defined by* (7). *Then,*

$$\|\mathbf{x} - \mathbf{x}^m\| \leq \frac{\|B\|}{1 - \|B\|}\left\|\mathbf{x}^m - \mathbf{x}^{m-1}\right\|, \quad m = 1, 2, \ldots. \tag{8}$$

Proof. From (7) follows $\mathbf{x}^{m+1} - \mathbf{x}^m = B(\mathbf{x}^m - \mathbf{x}^{m-1})$ and, by recursion,

$$\mathbf{x}^{m+k+1} - \mathbf{x}^{m+k} = B^{k+1}\left(\mathbf{x}^m - \mathbf{x}^{m-1}\right), \quad k = 0, 1, \ldots. \tag{9}$$

Note now that $\mathbf{x}^{m+p} - \mathbf{x}^m = \sum_{k=0}^{p-1}(\mathbf{x}^{m+k+1} - \mathbf{x}^{m+k})$. Hence, by the triangle inequality and (9)

$$\begin{aligned} \left\|\mathbf{x}^{m+p} - \mathbf{x}^m\right\| &\leq \sum_{k=0}^{p-1}\left\|B^{k+1}\right\|\left\|\mathbf{x}^m - \mathbf{x}^{m-1}\right\| \\ &\leq \frac{\|B\| - \|B\|^{p+1}}{1 - \|B\|}\left\|\mathbf{x}^m - \mathbf{x}^{m-1}\right\|. \end{aligned} \tag{10}$$

Letting $p \to \infty$ and noting that $\mathbf{x}^{m+p} \to \hat{\mathbf{x}}$, (8) follows. □

The basic iteration method (4) or the splitting methods in Section 2, can be improved in various ways. This will be the major topic of this paper.

Note first that application of the splitting in (7) requires in general that the matrix R is given in explicit form, which can make the method less viable.

The most natural way to improve (4) is to introduce an approximation C of A, to be used when the correction $\mathbf{e}^k$ in (4) is computed. The relation $\mathbf{e}^k = -\tau\mathbf{r}^k$ is then replaced by $C\mathbf{e}^k = -\tau\mathbf{r}^k$.

Such a matrix is mostly called preconditioner since, by a proper choice, it can significantly improve the condition number $\mathcal{K}$ of A, that is,

$$\mathcal{K}\left(C^{-1}A\right) \ll \mathcal{K}(A), \tag{11}$$

where $\mathcal{K}(B) = \|B\|\,\|B^{-1}\|$.

Clearly, in practice, the matrix C must be chosen such that the linear systems with C can be solved with relatively little expense compared to a solution method for A. In particular, the expense for C is much smaller than that for A using a direct solution method.

For badly scaled matrices A a simple, but often practically useful, choice of C is the (block) diagonal part D of A. Much more efficient choices will be discussed later in the paper.

Early suggestions to use such a matrix C can be found in papers by D'Yakonov [7] and Gunn [8]. For an automatic scaling procedure, see [9] and references therein.

In the present paper, we will survey various choices of C which have proven to be useful in practice. The paper attempts to give a more personal account of the development of iterative solution methods. It is also not our ambition to present the present state- of- the- art but rather to describe the unfolding of the field.

In the remainder of the paper, we discuss, in order, methods based on splitting of the given matrix, the accelerated iterative methods of the Chebyshev and (generalized) conjugate gradient types, pointwise and block incomplete factorization preconditioning methods, symmetrized preconditioners of SSOR and ADI type, approximate inverses, and augmented subspace preconditioning methods. If space had allowed it, it would have been followed by presentation of geometric and algebraic multigrid methods, two-level and multilevel methods, elementwise preconditioning methods, and domain decomposition methods. Also, iteration error estimates and influence of rounding errors, and preconditioners for matrices of saddle point type would have been included. The paper ends with some concluding remarks.

The following relations will be used; if $A = [a_{ij}]$, then $A^T = [a_{ji}]$ denotes the transpose of A, while $A^* = [\overline{a}_{ji}]$, denotes the Hermitian transpose.

2. Splitting Methods

A comprehensive, early presentation of splitting methods, and much more on iterative solution methods, is found in Varga [10]. Often there is a natural splitting of the given matrix as

$$A = C - R, \tag{12}$$

where C is nonsingular. This can be used to formulate an iterative solution method in the form

$$C\mathbf{x}^{k+1} = R\mathbf{x}^k + \mathbf{b}, \quad k = 0, 1, \dots. \tag{13}$$

This method converges if $\rho(C^{-1}R) < 1$.

Definition 1. (a) A matrix C is said to be *monotone* if C is nonsingular and $C^{-1} \geq 0$ (componentwise).

(b) $A = C - R$ is called a *regular splitting* [10], if C is monotone and $R \geq 0$.

(c) a *weak regular splitting* [11], if C is monotone and $C^{-1}R \geq 0$.

(d) a *nonnegative splitting* [12], if C is nonsingular and $C^{-1}R \geq 0$.

The following holds, see, for example, [6].

Theorem 2. *Let $A = C - R$ be a nonnegative splitting of A. Then, the following properties are equaivalent:*

(a) $\rho(B) < 1$, *that is, $A = C - R$ is a convergent splitting,*

(b) *$I - B$ is monotone,*

(c) *A is nonsingular and $G = A^{-1}R \geq 0$.*

(d) *A is nonsingular and $\rho(B) = \rho(G)/[1 + \rho(G)]$, where $G = A^{-1}R$.*

Corollary 1. *If $A = C - R$ is a weak regular splitting, then the splitting is convergent if and only if A is monotone.*

Proof. (see, e.g., [6]). □

A splitting method that became popular in the fifties is the SOR method. Here, $A = D - L - U$, where D is the (block) diagonal and L, U are the (block) lower and upper triangular parts of A, respectively. The successive relaxation method takes the form

$$\left(\frac{1}{\omega}D - L\right)\mathbf{x}^{k+1} = \left[\left(\frac{1}{\omega} - 1\right)D + U\right]\mathbf{x}^k + \mathbf{b}, \quad k = 0, 1, \dots, \tag{14}$$

where $\omega \neq 0$ is a parameter, called the relaxation parameter. For $\omega = 1$ one gets the familiar Gauss-Seidel method (Gauss 1823 [5] and Seidel 1814 [13]) and for $\omega > 1$ the successive overrelaxation (SOR) method (Frankel 1950 [14] and Young 1950 [15]).

For the iteration matrix in (14),

$$\mathcal{L}_\omega = \left(\frac{1}{\omega}D - L\right)^{-1}\left(\left(\frac{1}{\omega} - 1\right)D + U\right), \tag{15}$$

it holds that $\rho(\mathcal{L}_\omega) \leq |\omega - 1|$, where the upper bound is sharp. Therefore, the relaxation method is divergent for $\omega \leq 0$ and $\omega \geq 2$ (see, e.g., [6, 16]).

An optimal value of ω can be determined as follows. Assume then that A has property (A^π), that is, there exists a permutation matrix P such that PAP^T is a block tridiagonal matrix. The following Lemma holds.

Lemma 1 (see [15]). *Assume that A has property (A^π) and let $\omega \neq 0$. Let $B := D^{-1}(L + U)$. Then,*

(a) *if $\lambda \neq 0$ is an eigenvalue of $\mathcal{L}_\omega$ and μ satisfies*

$$\mu^2 = \frac{(\lambda + \omega - 1)^2}{(\omega^2\lambda)}, \tag{16}$$

then, μ is an eigenvalue of B,

(b) *if μ is an eigenvalue of B and λ satisfies*

$$\lambda + \omega - 1 = \omega\mu\lambda^{1/2}, \tag{17}$$

then, λ is an eigenvalue of $\mathcal{L}_\omega$.

Proof. For a short proof, see [6]. □

Theorem 3. *Assume that*

(a) *A has property (A^{π}), and*

(b) *the block matrix $B = I - D^{-1}A$ has only real eigenvalues.*

Then, the SOR method converges for any initial vector if and only if $\rho(B) < 1$ and $0 < \omega < 2$. Further, we have

$$\omega_{opt} = \frac{2}{1+\sqrt{1-\rho(B)^2}}, \tag{18}$$

for which the asymptotic convergence factor is given as

$$\min_{\omega}\rho(\mathcal{L}_{\omega}) = \rho\left(\mathcal{L}_{\omega_{opt}}\right) = \omega_{opt} - 1 = \frac{\left(1-\sqrt{1-\rho(B)^2}\right)}{\left(1+\sqrt{1-\rho(B)^2}\right)}. \tag{19}$$

Proof. Fort a short proof, see [6]. □

The eigenvalues of $C^{-1}A$ are in general complex, and for $\omega = \omega_{\text{opt}}$ it can be shown that they are distributed around a circle in the complex plane. This implies that the method can not be polynomially accelerated. (See Section 3 for a presentation of polynomial acceleration methods.) Further, the efficiency of the SOR method turns out to be critically dependent on the choice of ω.

A similar result as in Theorem 3 has been shown in [6], see also [17], that holds even if A does not have property (A^{π}), but is Hermitian.

Theorem 4. *Let A be Hermitian and positive definite and let*

$$\widetilde{\mathcal{L}}_{\omega} = D^{1/2}L_{\omega}D^{-1/2} = \left(\frac{1}{\omega}I - \widetilde{L}\right)^{-1}\left(\left(\frac{1}{\omega}-1\right)I - \widetilde{L}^{*}\right), \tag{20}$$

where $\widetilde{L} = D^{-1/2}LD^{1/2}$, and let $0 < \omega < 2$. Then,

$$\rho(L_{\omega})^2 = \rho\left(\widetilde{L}_{\omega}\right)^2 \leq 1 - \frac{2/\omega - 1}{(1/\omega - 1/2)^2\delta^{-1} + \gamma + 1/\omega}, \tag{21}$$

where

$$\gamma = \sup_{x\neq 0}\left\{\frac{\left[\left(\left|\mathbf{x},\widetilde{L}\mathbf{x}\right|^2/(\mathbf{x},\mathbf{x})\right) - 1/4(\mathbf{x},\mathbf{x})\right]}{\left(\widetilde{A}\mathbf{x},\mathbf{x}\right)}\right\}, \tag{22}$$

$$\delta = \lambda_{\min}\left(\widetilde{A}\right) = \frac{\min_{x\neq 0}\left(\widetilde{A}\mathbf{x},\mathbf{x}\right)}{(\mathbf{x},\mathbf{x})}.$$

Further, if $|(\mathbf{x},\widetilde{L}\mathbf{x})| \leq 1/2(\mathbf{x},\mathbf{x})$, then

$$\omega^{*} = \frac{2}{1+\sqrt{2\delta}}, \tag{23}$$

minimizes the upper bound of $\rho(L_{\omega})$ and we have

$$\rho(L_{\omega^{*}})^2 = \frac{1-\sqrt{\delta/2}}{1+\sqrt{\delta/2}}. \tag{24}$$

For a proof, see [6].

In Section 4, we present a symmetric version of the SOR method where acceleration is possible.

3. Accelerated Iterative Methods

In this section, the important Chebyshev and conjugate gradient iteration methods are presented.

Consider first the iterative method (4) with variable timesteps τ_k,

$$\mathbf{x}^{k+1} = \mathbf{x}^{k} - \tau_k C^{-1}\mathbf{r}^{k}, \qquad \mathbf{r}^{k} = A\mathbf{x}^{k} - \mathbf{b}, \quad k = 0,1,\ldots. \tag{25}$$

Here, $\{\tau_k\}$ is a sequence of iteration (acceleration) parameters. If $\tau_k = \tau$, $k = 0,1,\ldots$, we talk about a stationary iterative method, otherwise about a nonstationary or semiiterative method.

Let $\mathbf{e}^{k} = \mathbf{x} - \mathbf{x}^{k}$, the iteration error. Then, it follows from (25) that $\mathbf{e}^{k+1} = (I - \tau_k C^{-1}A)\mathbf{e}^{k}$, $k = 0,1,\ldots$, so $\mathbf{e}^{m} = P_m(C^{-1}A)\mathbf{e}^{0}$ (and $\mathbf{r}^{m} = AP_m(C^{-1}A)A^{-1}\mathbf{r}^{0} = P_m(AC^{-1})\mathbf{r}^{0}$). Here, $P_m(\lambda) = \Pi_{k=0}^{m}(1 - \tau_k\lambda)$ a polynomial of degree m having zeros at $1/\tau_k$ and satisfying $P_m(0) = 1$.

We want to choose the parameters $\{\tau_k\}$ such that $\|\mathbf{e}^{m}\|$ is minimized. However, this would mean that in general the parameters would depend on $\mathbf{e}^{0}$, which is not known. Also the eigenvalues of $C^{-1}A$ are not known. We then take the approach of minimizing $\|\mathbf{e}^{m}\|/\|\mathbf{e}^{0}\|$ for all $\mathbf{e}^{0}$; that is, we want to minimize $\|P_m(C^{-1}A)\mathbf{r}^{0}\|$.

3.1. The Chebyshev Iterative Method. In case the eigenvalues of $C^{-1}A$ are real and positive and if a positive lower (a) and (b) an upper bound are known of the spectrum, then we see that $\{\tau_k\}$ should be chosen such that $\max_{a\leq\lambda\leq b}|P_m(\lambda)|$ is minimized over all $P_m \in \Pi_m^0$, that is, over the set of polynomials of degree m satisfying $P_m(0) = 1$.

The solution to this min-max problem is well known,

$$P_m(\lambda) = \frac{\mathrm{T}_m((b+a-2\lambda)/(b-a))}{T_m((b+a)/(b-a))}, \tag{26}$$

where $T_m(z) = (1/2)[(z+(z^2-1)^{1/2})^m + (z-(z^2-1)^{1/2})^m] = \cos(m \arccos z)$ are the Chebyshev polynomials of the first kind. The corresponding values of τ_k satisfy

$$\frac{1}{\tau_k} = \frac{b-a}{2}\cos\Theta_k + \frac{b+a}{2}, \quad \Theta_k = \frac{2k-1}{2m}\pi, \; k = 1,2,\ldots,m, \tag{27}$$

which are the zeros of the polynomial. The corresponding method is referred to as the Chebyshev (one-step) acceleration method, see, for example, [10, 18]. It is an easy matter to show that

$$\frac{1}{T_m}\left(\frac{b+a}{b-a}\right) \leq 2\varrho^{m}, \quad \text{where } \varrho = \frac{(b^{1/2} - a^{1/2})}{(b^{1/2} + a^{1/2})}. \tag{28}$$

This implies that if the number of iterations satisfies $m \geq \ln\varrho^{-1}\ln(2/\varepsilon)$, that is, in particular if

$$m \geq \frac{1}{2}\left(\frac{b}{a}\right)^{1/2}\ln(2/\varepsilon), \quad \varepsilon > 0, \tag{29}$$

then $\|\mathbf{e}^{m}\|/\|\mathbf{e}^{0}\| \leq \varepsilon$.

The disadvantage with this method is that to make it effective one needs accurate estimates of a and b, and we need to determine m beforehand (which, however, can be done by (29)). The method cannot utilize any special distribution of the eigenvalues in the spectrum (as opposed to the conjugate gradient method, see below). More important, however, is that this method is actually numerically unstable (similarly to an explicit time stepping method for initial value problems when the time steps are too large). This is due to the fact that $\|I - \tau_k C^{-1}A\|$ is much larger than unity for several of the values τ_k. However, one may prove that with some particular permutation of the parameters, their instability effect can be avoided.

There is an alternative to the choice (25). Namely, there exists a three term form of the Chebyshev acceleration method

$$\mathbf{x}^{k+1} = \alpha_k \mathbf{x}^k + (1-\alpha_k)\mathbf{x}^{k-1} - \beta_k C^{-1}\mathbf{r}^k, \quad k = 1,2,\ldots, \tag{30}$$

where $\mathbf{x}^1 = \mathbf{x}^0 - (1/2)\beta_0 C^{-1}\mathbf{r}^0$.

Here, the parameters are chosen as $\beta_0 = 4/(a+b)$,

$$\alpha_k = \frac{a+b}{2}\beta_k, \qquad \beta_k^{-1} = \frac{a+b}{2} - \left(\frac{b-a}{4}\right)^2 \beta_{k-1} \qquad k = 1,2,\ldots. \tag{31}$$

Hence, we do not have to determine the number of steps beforehand. More importantly, it has been shown in [18] that this method is numerically stable. (For some related remarks, see [6]). A similar form of the method was proposed a long time ago, see Golub and Varga [19] and the references cited therein.

It is interesting to note that the parameters approach stationary values. If $C^{-1}A = I - B$ and B has eigenvalues in $[-\varrho, \varrho]$, $\varrho = \varrho(B) < 1$ (the spectral radius of B), then

$$a = 1 - \varrho, \qquad b = 1 + \varrho,$$
$$\alpha_k = \frac{a+b}{2}\beta_k \longrightarrow \frac{2}{\left[1 + \left(1 - \varrho^2\right)^{1/2}\right]}, \tag{32}$$

which is recognized as the parameter ω_{opt} of the optimal SOR method (see Section 2). Young [20] has proven that the asymptotic rate of convergence is retained even if one uses the stationary values throughout the iterations.

For the case of complex eigenvalues of $C^{-1}A$ with positive real parts and contained in an ellipse one may choose parameters similarly. See [6, 21, 22] for details. For comments on the optimaly of the method, see [23]. For application of the method for nonsymmetric problems, see [6, 24].

Perhaps the main thrust during the 1970 has been in using the conjugate gradient method as an acceleration method. Already much has been written on the subject; we refer to [25–27] for a historical account, to [18, 28–33] for an exposition of the preconditioned conjugate gradient PCG method and to [18, 34] for a survey of generalized and truncated gradient methods for nonsymmetric and indefinite matrix problems.

```
Given    $\mathbf{x}^{(0)}$, $\varepsilon$                 initial guess and absolute or
                                 relative stopping tolerance
Set      $\mathbf{x}^{(0)}$, $\mathbf{g} = A\mathbf{x} - \mathbf{b}$,
         $\delta_0 = \mathbf{g}^T\mathbf{g}$
         $\mathbf{d} = -\mathbf{g}$                   initial search direction
Repeat   until convergence
         $\mathbf{h} = A\mathbf{d}$
         $\tau = \delta_0/(\mathbf{d}^T\mathbf{h})$
         $\mathbf{x} = \mathbf{x} + \tau\mathbf{d}$               new approximation
         $\mathbf{g} = \mathbf{g} + \tau\mathbf{h}$               new (iterative) residual
         $\delta_1 = \mathbf{g}^T\mathbf{g}$
         if $\delta_1 \le \varepsilon$ then stop,       otherwise
         $\beta = \delta_1/\delta_0$, $\delta_0 = \delta_1$
         $\mathbf{d} = -\mathbf{g} + \beta\mathbf{d}$            new search direction
```

Algorithm 1: Standard conjugate gradient algorithm.

The advantage with conjugate gradient methods is that they are self adaptive; the optimal parameters are calculated by the algorithm so that the error in energy norm $\|e^l\|_{A^{1/2}} = \{(e^l)^T A e^l\}^{1/2}$ is minimized. This applies to a problem where C and A are symmetric and positive definite (SPD) or, more generally, if $C^{-1}A$ is similarly equivalent to an SPD matrix. Hence, there is no need to know any bounds for the spectrum. Since the method converges at least as fast as the Chebyshev method it follows that $\|x - x^m\|_{A^{1/2}} \le \varepsilon\|x - x^0\|_{A^{1/2}}$, if

$$m = \text{int}\left\{\frac{1}{2}\mathcal{K}^{1/2}\ln\left(\frac{2}{\varepsilon}\right) + 1\right\}. \tag{33}$$

We describe now the conjugate gradient method. Thereby we follow the presentations in [29, 35].

3.2. *The Preconditioned Conjugate Gradient Method.*

During the past 40 years or so, the preconditioned conjugate gradient method has become the major iterative solution method for linear systems of algebraic equations, in particular those arising in science and engineering. The author of these notes became interested in the method by the beginning of 1970 (cf. [18]).

The conjugate gradient algorithm to solve a system of linear equations the $A\mathbf{x} = \mathbf{b}$, where $A(n \times n)$ is symmetric and positive definite, was originally introduced by Hestenes and Stiefel [25] in 1950. Before we discuss the properties of the CG method, we describe its implementation. Namely, the algorithm consists of the steps in Algorithm 1.

What one sees from a first glance is that the CG algorithm is quite simple. Each iteration consists of one matrix-vector multiplication, two vector updates and two scalar products. Apart from the initial guess $\mathbf{x}^{(0)}$ (which can be taken to be the zero vector) and stopping tolerance, there are no other method parameters to be determined or tuned by the user. Thus, the method is easily programmable, cheap in terms of arithmetic operations and performs as a black box.

For some problems the standard (unpreconditioned) CG method performs impressively well and this can be explained by some particular properties of this powerful algorithm.

The CG method is best described as a method to minimize the quadratic functional

$$f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T A\mathbf{x} - \mathbf{b}^T\mathbf{x} + \mathbf{c}, \tag{34}$$

over a set of vectors. If A is nonsingular, then we can rewrite f in the form

$$f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T(A\mathbf{x} - \mathbf{b})^T A^{-1}(A\mathbf{x} - \mathbf{b}) - \frac{1}{2}\mathbf{b}^T A^{-1}\mathbf{b} + \mathbf{c}, \tag{35}$$

so, minimizing the quadratic functional is equivalent to solving the system $A\mathbf{x} = \mathbf{b}$. If A is singular and A^{-1} in (35) is replaced by a generalized inverse of A, then the above equivalence still holds if the minimization takes place on a subspace in the orthogonal complement to the null-space of A.

Given an initial approximation $\mathbf{x}^{(0)}$ and the corresponding residual $\mathbf{r}^{(0)} = A\mathbf{x}^{(0)} - \mathbf{b}$, the minimization in the conjugate gradient method takes place successively on a subspace

$$\mathcal{K}_k = \left\{\mathbf{r}^{(0)},\ A\mathbf{r}^{(0)},\ A^2\mathbf{r}^{(0)}, \dots, A^{k-1}\mathbf{r}^{(0)}\right\}, \tag{36}$$

of growing dimension. This subspace is referred to as the *Krylov set*.

In the derivation of the algorithm, the next approximate solution is constructed as

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \tau_k\mathbf{d}^{(k)}, \tag{37}$$

where τ_k is chosen

$$\tau_k = \frac{-\mathbf{d}^{(k)^T}\mathbf{g}^{(k)}}{\mathbf{d}^{(k)^T}A\mathbf{d}^{(k)}} = \frac{-\mathbf{d}^{(k)^T}\left(A\mathbf{x}^{(k)} - \mathbf{b}\right)}{\mathbf{d}^{(k)^T}A\mathbf{d}^{(k)}}, \tag{38}$$

which minimizes the function $f(\mathbf{x}^{(k)} + \tau\mathbf{d}^{(k)})$, $-\infty < \tau < \infty$. Also, the gradient of f at $\mathbf{x}^{(k+1)}$ is made orthogonal to the search direction $\mathbf{d}^{(k)}$. This is seen from the following relations:

$$\begin{aligned}\mathbf{x}^{(k+1)} &= \mathbf{x}^{(k)} + \tau_k\mathbf{d}^{(k)} \Longrightarrow A\mathbf{x}^{(k+1)} - \mathbf{b}\\ &= A\mathbf{x}^{(k)} - \mathbf{b} + \tau_k A\mathbf{d}^{(k)} \Longrightarrow \mathbf{g}^{(k+1)} = \mathbf{g}^{(k)} + \tau_k A\mathbf{d}^{(k)}\\ &\Longrightarrow \mathbf{d}^{(k)^T}\mathbf{g}^{(k+1)} = \mathbf{d}^{(k)^T}\mathbf{g}^{(k)} + \tau_k\mathbf{d}^{(k)^T}A\mathbf{d}^{(k)} = 0.\end{aligned} \tag{39}$$

As in Fourier type minimization methods, it turns out to be efficient to work with orthogonal (A-orthogonal) search directions $\mathbf{d}^{(k)}$ which, since A is symmetric, can be determined from a three-term recursion

$$\mathbf{d}^{(0)} = \mathbf{r}^{(0)}, \qquad \mathbf{d}^{(k+1)} = -A\mathbf{d}^{(k)} + \widetilde{\beta}_k\mathbf{d}^{(k)}, \quad k = 1, 2, \dots, \tag{40}$$

or equivalently, from

$$\mathbf{d}^{(k+1)} = -\mathbf{r}^{(k+1)} + \beta_k\mathbf{d}^{(k)}. \tag{41}$$

This recursive choice of search directions is done so that at each step the solution has smallest error in the A-norm, $\|\mathbf{x} - \mathbf{x}^{(k)}\|_A = \{\mathbf{e}^{(k)^T}A\mathbf{e}^{(k)}\}^{1/2}$, where $\mathbf{e}^{(k)} = \mathbf{x} - \mathbf{x}^{(k)}$ is the iteration error. As mentioned, the minimization takes place over the set of (Krylov) vectors $\mathcal{K}_k$ and, as is readily seen

$$\begin{aligned}\mathcal{K}_k &= \left\{\mathbf{x}^{(1)} - \mathbf{x}^{(0)},\ \mathbf{x}^{(2)} - \mathbf{x}^{(0)}, \dots, \mathbf{x}^{(k)} - \mathbf{x}^{(0)}\right\}\\ &= \left\{\mathbf{g}^{(0)},\ \mathbf{g}^{(1)}, \dots, \mathbf{g}^{(k-1)}\right\}\\ &= \left\{\mathbf{d}^{(0)},\ \mathbf{d}^{(1)}, \dots, \mathbf{d}^{(k-1)}\right\}.\end{aligned} \tag{42}$$

To summarize, the CG method possesses the following remarkable properties.

Theorem 5. *Let the CG Algorithm 1 be applied to a symmetric positive definite matrix A. Then, in exact arithmetic the following properties hold:*

(1) *the iteratively constructed residuals* $\mathbf{g}$ *are mutually orthogonal, that is,* $\mathbf{g}^{(k)^T}\mathbf{g}^{(j)} = 0$, $j < k$;

(2) *the search directions* $\mathbf{d}$ *are A-orthogonal (or conjugate), that is,* $\mathbf{d}^{(k)^T}A\mathbf{d}^{(j)} = 0$, $j < k$:

(3) *as long as the method has not converged, that is,* $\mathbf{g}^{(k)} \neq 0$, *the algorithm proceeds with no breakdown and* (42) *holds;*

(4) *as long as the method has not converged, the newly constructed approximation* $\mathbf{x}^{(k)}$ *is the unique point in* $\mathbf{x}^{(0)} \oplus \mathcal{K}_k$ *that minimizes* $\|\mathbf{e}^{(k)}\|_A = \|\mathbf{x} - \mathbf{x}^{(k)}\|_A$,

(5) *the convergence is monotone in A-norm, that is,* $\|\mathbf{e}^{(k)}\|_A < \|\mathbf{e}^{(k-1)}\|_A$ *and* $\mathbf{e}^{(m)} = 0$ *will be achieved for some* $m \leq n$.

For a proof of the above theorem consult, for instance, [29] or [6].

Since the method is optimal, that is it gives the smallest error on a subspace of growing dimension, it *terminates* with the exact solution (ignoring round-off errors) in at most n steps (the dimension of the whole vector space $\mathbf{x} \in R^n$). In fact, it can be readily seen that the CG algorithm terminates after m steps, where m is the degree of the minimal polynomial Q_m to A with respect to the initial residual vector, in other words, Q_m has the smallest degree of all polynomials Q for which $Q(A)\mathbf{r}^{(0)} = 0$. Therefore, the CG method can be viewed also as a direct solution method. However, in practice we want convergence to occur to an acceptable accuracy in much fewer steps than n or m. Thus, we use CG as an iterative method.

For further discussions of the CG methods, see [6, 34, 35].

When one experiments with CG to solve systems with various matrices one observes some phenomena which need special attention. This can be illustrated by a simple example.

Consider the solution of $A\mathbf{x} = \mathbf{b}$ by the standard conjugate gradient, where

$$A = \begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & & \\ & & -1 & 2 & -1 \\ & & & -1 & 1 \end{bmatrix}, \qquad b = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \tag{43}$$

The exact solution is $\hat{\mathbf{x}} = [1, 1, \ldots, 1]^T$. Starting with $\mathbf{x}^{(0)} = [0, 0, \ldots, 0]^T$ one finds that after k iterations

$$\mathbf{x}^{(k)} = \left[\frac{k}{k+1}, \frac{k-1}{k+1}, \ldots, \frac{1}{k+1}, 0, \ldots, 0\right]^T, \tag{44}$$

for $1 \leq k \leq n-1$ and $\mathbf{x}^{(n)} = \hat{\mathbf{x}}$. Hence, the information travels one step at a time from left to right and it takes n steps before the last component has changed at all. The algorithm converges exactly in n steps and terminates due to the final recurrence property of the method.

Another detail one observes is that the norm of the error, $\|\mathbf{x} - \mathbf{x}^{(k)}\|$, can be much larger than the norm of the iteratively computed residual.

These examples illustrate the fact that although the method has an optimal order of convergence rate in the *energy norm*, its actual convergence rate in spectral norm can be different and depends both on the distribution of the eigenvalues of the (preconditioned) system matrix and on the initial approximation (or residual). (For comparison, we note that the rate of convergence for steepest descent depends only on the ratio of the extremal eigenvalues of A.) Faster convergence for the CG method is expected when the eigenvalues are clustered.

One way to get a better eigenvalue distribution is to precondition A by a proper preconditioner B. Hence, in order to achieve a better eigenvalue distribution it is crucial in practice to use some form of preconditioning, that is, a matrix B which approximates A in some sense, which is relatively cheap to solve systems with and for which the spectrum of $B^{-1} A$ (equivalently $B^{-1/2}AB^{-1/2}$ if B is s.p.d.) is more favorable for the convergence of the CG method. As it turns out, if B is symmetric and positive definite, the corresponding preconditioned version, the PCG method, is best derived by replacing the inner product with $(\mathbf{u}, \mathbf{v}) = \mathbf{u}^T B\mathbf{v}$. It takes the following form, see Algorithm 2.

Here, $[B]^{-1}$ denotes the action of B^{-1}, that is, one does not multiply with the inverse matrix B^{-1}, but normally solves a linear system with matrix B.

In order to understand what is wanted of a *good* preconditioning matrix, we discuss first some issues of major importance related to the rate of convergence of the CG method. Thereby it becomes clear that the standard spectral condition number is often too simple to explain the detailed convergence behaviour. In particular we discuss the sub- and superlinear convergence phases frequently observed in the convergence history of the conjugate gradient method.

```
Given    $\mathbf{x}^{(0)},\ \varepsilon$                        initial guess and
                                             stopping tolerance
Set      $\mathbf{x}^{(0)},\ \mathbf{g} = A\mathbf{x} - \mathbf{b}$,
             $\mathbf{h} = [B]^{-1}\mathbf{g}$
         $\delta_0 = \mathbf{g}^T\mathbf{h}$
         $\mathbf{d} = -\mathbf{h}$                          initial search direction
Repeat   until convergence
         $\mathbf{h} = A\mathbf{d}$
         $\tau = \delta_0/(\mathbf{d}^T\mathbf{h})$
         $\mathbf{x} = \mathbf{x} + \tau\mathbf{d}$                    new approximation
         $\mathbf{g} = \mathbf{g} + \tau\mathbf{h}$                    new (iterative) residual
         $\delta_1 = \mathbf{g}^T\mathbf{g}$
         $\mathbf{h} = [B]^{-1}\mathbf{g}$                  new pseudoresidual
         $\delta_1 = \mathbf{g}^T\mathbf{h}$
         if $\delta_1 \leq \varepsilon$ then stop
         $\beta = \delta_1/\delta_0,\ \delta_0 = \delta_1$
         $\mathbf{d} = -\mathbf{h} + \beta\mathbf{d}$                 new search direction
```

Algorithm 2: Preconditioned conjugate gradient algorithm.

A preconditioner can be applied in two different manners, namely, as $B^{-1}A$ or BA. The first form implies the necessity to solve a system with B at each iteration step while the second form implies a matrix-vector multiplication with B (a *multiplicative preconditioner*). In the latter, case B can be seen as an approximate inverse of A. One can also use a hybrid form $\alpha B_1^{-1} + \beta B_2$.

The presentation here is limited to symmetric positive semidefinite matrices. It is based mainly on the articles [29, 31].

3.3. *On the Rate of Convergence Estimates of the Conjugate Gradient Method.*

Let A be symmetric, positive semidefinite and consider the solution of $A\mathbf{x} = \mathbf{b}$ by a preconditioned conjugate gradient method. In order to understand how an efficient preconditioner to A should be chosen we must first understand some general properties of the rate of convergence of conjugate gradient methods.

3.3.1. *Rate of Convergence Estimates Based on Minimax Approximation.*

As is well known (see e.g., [6, 30]), the conjugate gradient method is a norm minimizing method. For the preconditioned standard CG method, we have

$$\left\|\mathbf{e}^k\right\|_A = \min_{P_k \in \pi_k} \|P_k(B)\mathbf{e}^0\|_A, \tag{45}$$

where $\|\mathbf{u}\|_A = \{\mathbf{u}^T A\mathbf{u}\}^{1/2}$, $\mathbf{e}^k = \mathbf{x} - \mathbf{x}^k$ is the iteration error and π_k denotes the set of polynomials of degree k which are normalized at the origin, that is, $P_k(0) = 1$. This is a norm on the subspace orthogonal to the mullspace of A, that is, on the whole space, if A is nonsingular.

Consider the C-innerproduct $(\mathbf{u}, \mathbf{v}) = \mathbf{u}^T C\mathbf{v}$, and note that $B = C^{-1}A$ is symmetric with respect to this innerproduct, let $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n$ be orthonormal eigenvectors and let λ_i, $i = 1, \ldots, n$ be the corresponding eigenvalues of B. Let

$$\mathbf{e}^0 = \sum_{j=1}^{n} \alpha_j \mathbf{v}_j, \tag{46}$$

be the eigenvector expansion of the initial vector where $\alpha_{\mathrm{j}} = (\mathbf{e}^0, \mathbf{v}_i)$, $i = 1, \ldots, n$. Note further that the eigenvectors are both A- and C-orthogonal. Then, by the construction of the CG method, it follows

$$\mathbf{e}^k = \sum_{j=1}^{n} \alpha_j P_k\left(\lambda_j\right) \mathbf{v}_j, \tag{47}$$

and, using the nonnegativity of the eigenvalues, we find

$$\begin{aligned} \left\|\mathbf{e}^k\right\|_A &= \left\|\sum_{j=1}^{n} \alpha_j P_k(\lambda_j) \mathbf{v}_j\right\|_A = \left\{\sum_{j=1}^{n} \alpha_j^2 \lambda_j P_k^2\left(\lambda_j\right)\right\}^{1/2} \\ &\leq \left\{\sum_{1,\lambda_j>0}^{n} \alpha_j^2 \lambda_j\right\}^{1/2} \max_{\substack{1 \leq i \leq n \\ \lambda_i > 0}} |P_k(\lambda_i)| \\ &= \max_{\substack{1 \leq i \leq n \\ \lambda_i > 0}} |P_k(\lambda_i)| \|\mathbf{e}^0\|_A. \end{aligned} \tag{48}$$

Here we have used the A-orthogonality of the eigenvectors. Similarly, using the C-orthogonality, we find

$$\left\|\mathbf{e}^k\right\|_C = \left\{\sum_{j=1}^{n} \alpha_j^2 P_k^2\left(\lambda_j\right)\right\}^{1/2} \leq \max_{1 \leq i \leq n} \left| P_k\left(\lambda_j\right) \right| \|\mathbf{e}^0\|_C. \tag{49}$$

Due to the minimization property (45) there follows from (48) the familiar bound

$$\left\|\mathbf{e}^k\right\|_A \leq \min_{P_k \in \pi_k^1} \max_{\substack{1 \leq i \leq n \\ \lambda_i > 0}} |P_k(\lambda_i)| \|\mathbf{e}^0\|_A. \tag{50}$$

Estimate (50) is sharp in the respect that for every k there exists an initial vector for which equality is attained. In fact, for such a vector we necessarily have that $\alpha_j \neq 0$ if and only if α_j belongs to a set of $k+1$ points (the so-called Haar condition) where $\max_i |P_k(\lambda_i)|$ is taken. For such an initial vector (49) shows that, if the eigenvalues are positive, we have also

$$\left\|\mathbf{e}^k\right\|_C = \min_{P_k \in \pi_k^1} \max_{1 \leq i \leq n} |P_k(\lambda_i)| \|\mathbf{e}^0\|_C. \tag{51}$$

The rate of convergence of the iteration error $\|\mathbf{e}^k\|_A$ is measured by the average convergence factor

$$\left\{\frac{\left\|\mathbf{e}^k\right\|_A}{\|\mathbf{e}^0\|_A}\right\}^{1/k}. \tag{52}$$

Inequality (50) shows that this can be majorized with an estimate of the rate of convergence of a best polynomial approximation problem (namely the best approximation of the function $\equiv 0$, of polynomials in π_k^1) in maximum norm on the discrete set formed by the spectrum of B. Clearly, multiple eigenvalues are treated as single so the actual approximation problem is

$$\min_{P_k \in \pi_k^1} \max_{1 \leq i \leq m} \left| P_k\left(\widetilde{\lambda}_i\right) \right|, \tag{53}$$

where the disjoint positive eigenvalues $\widetilde{\lambda}_j$ have been ordered in increasing value, $0 < \widetilde{\lambda}_i < \cdots < \widetilde{\lambda}_m$, and m is the number of such eigenvalues. However, the solution of this problem requires knowledge of the spectrum, which is not available in general. Even if it is known, the estimate (53) can be troublesome in practice, since it involves approximation on a general discrete set of points.

Besides being costly to apply, such estimates do not give any qualitative insight in the behaviour of the conjugate gradient method for various typical eigenvalue distributions.

That is why we make some further assumptions on the spectrum in order to simplify the approximation problem and at the same time, present estimates which can be used both to estimate the number of iterations and to give some insight in the qualitative behaviour of the iteration method.

3.3.2. Standard Condition Number: Linear Convergence. If the eigenvalues are (densely) located in an interval $[a, b]$ where $a > 0$, we can majorize the best approximation problem on the discrete set with the best approximation problem on the interval and frequently still get a good estimate. We have

$$\min_{P_k \in \pi_k^1} \max_{1 \leq i \leq m} \left| P_k\left(\widetilde{\lambda}_i\right) \right| \leq \min_{P_k \in \pi_k^1} \max_{a \leq x \leq b} |P_k(x)|. \tag{54}$$

The solution to this min max problem is well known and uses Chebyshev polynomials. One finds that

$$\min_{P_k \in \pi_k^1} \max_{a \leq x \leq b} |P_k(x)| = \frac{1}{T_k((b+a)/(b-a))} = \frac{2\sigma^k}{1+\sigma^{2k}}, \tag{55}$$

where $\sigma = (1 - \sqrt{a/b})/(1 + \sqrt{a/b})$, and $T_k(x) = (1/2)[(x + \sqrt{x^2-1}) + (x - \sqrt{x^2-1})^k]$, $a = \widetilde{\lambda}_1$, $b = \widetilde{\lambda}_m$. Hence, the average rate of convergence of the upper bound approaches σ as $k \to \infty$. Also, it is readily found (see [35]) that the relative iteration error $\|e^k\|_A / \|e^0\|_A \leq \varepsilon$ if

$$k = k^*(a, b, \varepsilon) = \left\lceil \frac{\ln\left((1/\varepsilon) + \sqrt{(1/\varepsilon^2) - 1}\right)}{\ln \sigma^{-1}} \right\rceil. \tag{56}$$

Here $\lceil \xi \rceil$ denotes the smallest integer not less than ξ.

It turns out that the above holds more generally if A is nonsymmetric but the eigenvalues are contained in an ellipse with foci a, b, where $b \geq a > 0$, if one replaces σ with $\widehat{\sigma} = \sigma\sqrt{(1+\delta)/(1-\delta)}$, where δ is the eccentricity of the ellipse, (i.e., the ratio of the semiaxes) and $\delta < 2\sqrt{a/b}/(1 + a/b)$.

Also, in a similar way, the case of eigenvalues contained in two separate intervals or ellipses can be analysed, see, for example, [35] for further details.

When $b/a \to \infty$, $\delta = 0$, and $\varepsilon \to 0$ the following upper bound becomes an increasingly accurate replacement of (56),

$$k^* \leq \left\lceil \frac{1}{2}\sqrt{\frac{b}{a}} \ln \frac{2}{\varepsilon} \right\rceil. \tag{57}$$

The above estimate of the rate of convergence and of the number of iterations show that they depend only on the condition number b/a and on the eccentricity of the ellipse, containing the eigenvalues. Therefore, except in special cases, this estimate is not very accurate. When we use a more detailed information of the spectrum and the initial error vector, sometimes substantially better estimates can be derived. This holds for instance when there are well separated small and/or large eigenvalues. Before we consider this important case, we mention briefly another similar minimax result which holds when we use *different norms* for the iteration error vector and for the initial vector.

By (48), we have

$$\begin{aligned} \left\|\mathbf{e}^k\right\|_A &= \left\{\sum \alpha_j^2 \lambda_j P_k^2(\lambda_j)\right\}^{1/2} \\ &= \left\{\sum \alpha_j^2 \lambda_j^{1-2s} \lambda_j^{2s} P_k^2(\lambda_j)\right\}^{1/2} \\ &\leq \min_{P_k \in \pi_k^1} \max_{1 \leq \lambda_j \leq m} \left|\lambda_j^s P_k(\lambda_j)\right| \left\{\sum \alpha_j^2 \lambda_j^{(1-2s)}\right\}^{1/2} \\ &= \min_{P_k \in \pi_k^1} \max_{1 \leq \lambda_j \leq m} \left|\lambda_j^s P_k(\lambda_j)\right| \|\mathbf{e}^0\|_{A^{1-2s}}. \end{aligned} \tag{58}$$

If the initial vector is such that Fourier coefficients for the highest eigenvalue modes are dominating, then $\|\mathbf{e}^0\|_{A^{1-2s}}$ may exist and take not too large values even for some $s \geq 1/2$. We consider the interesting case where $s \geq 1/2$, for which the following theorem holds (see [6, 36]).

Theorem 6. *Let π_k^1 denote the set of polynomials of degree k such that $P_k(0) = 1$. Then for $k = 1, 2\ldots$ and for any $s \geq 1/2$ such that $2s$ is an integer, it holds*

$$\frac{\left\|\mathbf{e}^k\right\|_A}{\|\mathbf{e}^0\|_A^{1-2s}} \leq \min_{P_k \in \pi_k^1} \max_{0 \leq x \leq 1} |x^s P_k(x)| \leq \left(\frac{s}{k+s}\right)^{2s}. \tag{59}$$

Remark 1. For $s = 1/2$ it holds

$$\max_{0 \leq x \leq 1} \left|x^{1/2} P_k(x)\right| = \frac{1}{2k+1}, \tag{60}$$

for $P_k(x) = U_{2k}(\sqrt{1-x})$ and for $s = 1$, it holds

$$\max_{0 \leq x \leq 1} |x P_k(x)| = \frac{1}{k+1} \tan \frac{\pi}{4k+4} < \frac{1}{(k+1)^2}, \tag{61}$$

for $P_k(x) = (x^{-1}(-1)^k)/(k+1)\tan(\pi/(4k+4))\ T_{k+1}((1+\cos(\pi/(2k+2)))x - \cos(\pi/(2k+2)))$ where $T_k(x)$ and $U_k(x)$ are the Chebyshev polynomials of kth degree of the first and second kind, respectively.

For other values (59) is an upper bound only, that is, not sharp. At any rate, it shows that the error $\|\mathbf{e}^k\|_A$ converges (initially) at least as fast as $(s/(k+s))^{2s}$, that is, as $1/(2k+1)$ for $s = 1/2$ and as $(1/(k+1))^2$ for $s = 1$.

Note that this convergence rate does not depend on the eigenvalues, in particular not on the spectral condition number.

Conclusion 1. By computing the initial approximation vector from a coarse mesh, the components for $\mathbf{e}^0$ for the first Fourier modes will be small and $\|\mathbf{e}^0\|_{A^{1-2s}}$ may take on values that are not very large even when $s = 1$ or bigger. Therefore, there is an initial decay of the residual as $O(k^{2s})$, independent of the condition number. Note, however, that the actual errors may not have decayed sufficiently even if the residual has.

We consider now upper bound estimates which show how the convergence history may enter a superlinear rate.

An Estimate to Show a Superlinear Convergence Rate Based on the K-Condition Number. A somewhat rough, but simple and illustrative superlinear convergence estimate can be obtained in terms of the so called K-condition number, (see [37, 38])

$$K = K(B) = \frac{((1/n)\operatorname{tr}(B))^n}{\det(B)} = \left(\frac{1}{n}\sum_{i=1}^{n} \lambda_i\right)^n \left(\Pi_{i=1}^n \lambda_i\right)^{-1}, \tag{62}$$

where we assume that B is s.p.d.

Note that $K^{1/n}$ equals the quotient between the arithmetic and geometric averages of the eigenvalues. This quantity is similar to the spectral condition number $\kappa(B)$ in that it is never smaller than 1, and is equal to 1 if and only if $B = \alpha I$, $\alpha > 0$ (recall that B is symmetrizable).

Based on the K-condition number, a superlinear convergence result can be obtained as follows.

Theorem 7. *Let $k < n$ be even and $k \geq 3 \ln K$. Then,*

$$\frac{\|\mathbf{e}^k\|_A}{\|\mathbf{e}^0\|_A} \leq \left(\frac{3\ln K}{k}\right)^{k/2}. \tag{63}$$

Proof. Let $k = 2m$ and the polynomial P_k be of a simplest possible form, that is, let it vanish at the m smallest and m largest eigenvalues of B. As follows from (48), we have then

$$\begin{aligned} \frac{\left\|\mathbf{e}^k\right\|_A}{\|\mathbf{e}^0\|_A} &\leq \max_{\lambda_1 \leq \lambda \leq \lambda_n} \left|\Pi_{i=1}^m \left(1 - \frac{\lambda}{\lambda_i}\right)\left(1 - \frac{\lambda}{\lambda_{n+1-i}}\right)\right| \\ &= \Pi_{i=1}^m \max_{\lambda_i \leq \lambda \leq \lambda_{n+1-i}} \left(\frac{\lambda}{\lambda_i} - 1\right)\left(1 - \frac{\lambda}{\lambda_{n+1-i}}\right) \\ &= \Pi_{i=1}^m \left(\frac{(\lambda_i + \lambda_{n+1-i})^2}{4\lambda_i \lambda_{n+1-i}} - 1\right) \\ &\leq \left(\left(\Pi_{i=1}^m \frac{(\lambda_i + \lambda_{n+1-i})^2}{4\lambda_i \lambda_{n+1-i}}\right)^{1/m} - 1\right)^m. \end{aligned} \tag{64}$$

The latter follows from $(\Pi_{i=1}^{m}(1-\Theta_i))^{1/m}+(\Pi_{i=1}^{m}\Theta_i)^{1/m}\leq 1$ with $\Theta_i = 4\lambda_i\lambda_{n+1-i}(\lambda_i+\lambda_{n+1-i})^2$. Using now twice the inequality between the arithmetic and geometric mean values, one has

$$\begin{aligned}\Pi_{i=1}^{m}\frac{(\lambda_i+\lambda_{n+1-i})^2}{4\lambda_i\lambda_{n+1-i}} &\leq \frac{\left((1/n)-2m\sum_{i=m+1}^{n-m}\lambda_i\right)^{n-2m}}{\Pi_{i=m+1}^{n-m}\lambda_i}\\ &\quad\times\frac{\Pi_{i=1}^{m}(\lambda_i+\lambda_{n+1-i}/2)^2}{\Pi_{i=1}^{m}\lambda_i\lambda_{n+1-i}}\\ &\leq \frac{\left(1/n\left(\sum_{i=m+1}^{n-m}\lambda_i+\sum_{i=1}^{m}(\lambda_i+\lambda_{n+1-i})\right)\right)^{n}}{\Pi_{i=m+1}^{n-m}\lambda_i\;\Pi_{i=1}^{m}\lambda_i\lambda_{n+1-i}}\\ &= K.\end{aligned}\tag{65}$$

Using $\exp(2x)-1\leq 2x/(1-x)$, $x<1$, we get the required estimate,

$$\begin{aligned}\frac{\|\mathbf{e}^k\|_A}{\|\mathbf{e}^0\|_A} &\leq \left(K^{2/k}-1\right)^{k/2}\leq\left(\frac{2\ln K}{k-\ln K}\right)^{k/2}\\ &\leq\left(\frac{2\ln K}{2k/3(k/3-\ln K)}\right)^{k/2}\leq\left(\frac{3\ln K}{k}\right)^{k/2}.\end{aligned}\tag{66}$$

□

A somewhat better result of the same type exists. The estimate is of similar type, that is,

$$\frac{\|\mathbf{r}^k\|_{C^{-1}}}{\|\mathbf{r}^0\|_{C^{-1}}}\leq\left(K^{1/k}-1\right)^{k/2},\tag{67}$$

where $\mathbf{r}^k = A\mathbf{e}^k$ was obtained using more complicated techniques, see [6, 37], and the references quoted therein. Note that here as $k/\ln K\rightarrow\infty$, we have

$$\begin{aligned}\frac{\|\mathbf{e}^k\|_{C^{-1}}}{\|\mathbf{e}^0\|_{C^{-1}}} &\leq\left(\mathbf{e}^{1/k\ln K}-1\right)^{k/2}\approx\left(1+\frac{\ln K}{2k}\right)^{\mathrm{k}/2}\left(\frac{\ln K}{k}\right)^{k/2}\\ &\approx K^{1/4}\left(\frac{\ln K}{k}\right)^{k/2},\end{aligned}\tag{68}$$

that is, the simpler upper bound in (63) is asymptotically worse than this bound (albeit in a different norm) by the factor $3^{k/2}/\ln K^{1/4}$.

The upper bounds in the above estimates involve a convergence factor which decreases with increasing iteration number and show hence a superlinear rate of convergence. Note, however, that $K^{1/n}\leq\kappa(B)\lesssim 4K$ (see [6]) where $\kappa(B)=\lambda_n/\lambda_1$ is the spectral condition number, so the K-condition number may take very large values when $\kappa(B)$ is large.

The estimates based on the K-condition number involve only "integral" characteristics of the preconditioned matrix (the trace and the determinant). Sometimes, it is possible to obtain a practical estimate of $K(B)$ which can be useful for the a priori construction of good preconditioners and for the a posteriori assessment of their quality, see Section 5 for further details.

The estimate

$$\frac{\left\|\mathbf{r}^k\right\|_{C^{-1}}}{\|\mathbf{r}^0\|_{C^{-1}}}\leq\left(K^{1/k}-1\right)^{k/2}\leq\varepsilon\tag{69}$$

shows that

$$K^{1/2}\left(1-K^{-1/k}\right)^{k/2}\leq\varepsilon\tag{70}$$

or

$$\frac{k}{2}\log_2\frac{1}{1-K^{-1/k}}\geq\log_2\frac{K^{1/2}}{\varepsilon},\tag{71}$$

which holds if

$$k>\log_2 K+2\log_2\frac{1}{\varepsilon}.\tag{72}$$

Hence, when $K\gg\varepsilon^{-2}$ the estimated number of iterations depends essentially only on $\log_2 K$, that is, depends little on the relative accuracy ε, which indicates a fast superlinear convergence, when $k>\log_2 K$.

When actually estimating the number of iterations, Theorem 6 shows a useful result only when $k>O(\ln K)=n(\ln K^{1/n})$, that is, the quotient between the arithmetic and geometric averages of the eigenvalues, which equals $K^{1/n}$, must be close to unity and the eigenvalues must be very well clustured so $K^{1/n}=1+O(n^{-\varepsilon})$ for some $\varepsilon>0$; otherwise the estimated number of iterations will be $O(n)$, which is normally a useless result. The next example illustrates this further.

Example 1. Consider a geometric distribution of eigenvalues of A, $\lambda_j = j^s$, $j=1,2,\ldots,n$ for some positive s. Here, asymptotically

$$\operatorname{tr}(A)=\sum_{1}^{n}j^s\sim\frac{1}{s+1}\,n^{s+1},\quad n\longrightarrow\infty.\tag{73}$$

Using Stirling's formula, we find

$$\det(A)=\prod_{1}^{n}\lambda_j=\left(\prod_{1}^{n}j\right)^{s}\sim(2\pi n)^{s/2}\left(\frac{n}{e}\right)^{ns},\quad n\longrightarrow\infty,\tag{74}$$

so,

$$\kappa(A)^{1/n}\sim\frac{e^s}{s+1},\quad n\longrightarrow\infty.\tag{75}$$

Hence, s must be sufficiently small for the estimate in Theorem 6. to be useful. On the other hand, the spectral condition number (i) $\kappa(A)=n^s$, and the simple estimate based on $\kappa(A)$ leads to $k\sim O(n^{s/2})$ and gives hence, asymptotically, a smaller upper bound when $s<2$. For further discussions on superlinear rate of convergence, see [39].

3.4. Generalized Conjugate Gradient Methods. The rate of convergence estimates as given above, holds for a restricted class of matrices, symmetric or, more generally, for normal matrices.

To handle more general classes of problems for which such optimal rate of convergence results as in (45) holds, one needs more involved methods. Much work has been devoted to this problem. This includes methods like generalized minimum residual (GMRES), see Saad and Schultz [40], generalized conjugate residual (GCR), and generalized conjugate gradient (GCG), see [6] and for further details [41]. As opposed to the standard conjugate and gradient method, they require a long version of updates for the search directions, as the newest search direction at each stage is not in general, automatically (in exact precision) orthogonal to the previous search directions, but must be orthogonalized at each step. This makes the computational expense per step grow linearly and the total expense grows quadratically with the iteration index. In addition, due to finite precision, there is a tendency of loss of orthogonality, even for symmetric problems when many iterations are required. One remedy which has been suggested is to use the method only for a few steps, say 10, and restart the method with the current approximation as initial approximation.

Clearly, however, in this way, the optimal convergence property of the whole Krylov set of vector is lost. For this and other possibilities, see, for example, [42].

Another important version of the generalized conjugate gradient methods occurs when one uses variable preconditioners. Variable preconditioners, that is, a preconditioner changed from one iteration to the next iteration step, are used in many contexts.

For instance, one can use variable drop tolerance, computed adaptively, in an incomplete factorization method (see Section 4). When the given matrix is partitioned in two by two blocks, it can be efficient to use inner iterations when solving arising systems for one, or both, of the diagonal block matrices, see, for example, [43], and the flexible conjugate gradient method in Saad, [44, 45].

Due to space limitations, the above topics will not be further discussed in this paper.

4. Incomplete Factorization Methods

There exist two classes of preconditioning methods that are closely related to direct solution methods. In this paper, we make a survey only of their main ingredients, but delete many of the particular aspects.

The first method is based on incomplete factorization were some entries arising during a triangular factorization are neglected to save in memory. The deletion can be based on some drop tolerance criterion or on a normally a priori, chosen sparsity pattern. The factorization based on a drop tolerance takes the following form. During the elimination (or equivalently, triangular factorization), the off-diagonal entries are accepted only if they are not too small. For instance,

$$a_{ij} := \begin{cases} a_{ij} - a_{ir}a_{rr}^{-1}a_{rj} & \text{if } \left|a_{ij}\right| \geq \varepsilon \sqrt{a_{ii}a_{jj}}, \\ 0, & \text{otherwise.} \end{cases} \tag{76}$$

Here, ε, $0 < \varepsilon \ll 1$ is the drop-tolerance parameter. Such methods may lead to too much fill-in (i.e., $a_{ij} \neq 0$ in positions where the original entry was occupied by a zero), because to be robust, they may require near machine-precision drop tolerances. Furthermore, as direct solution methods, they are difficult to parallelize efficiently.

The incomplete factorization method can readily be extended to matrices partitioned in block form. Often, instead of a drop tolerance, one prescribes the sparsity pattern of the triangular factors in the computed preconditioner, that is, entries arising outside the chosen pattern are ignored. An early presentation of such incomplete factorization methods was given by Meijerink and van der Vorst [46]. One can make a diagonal compensation of the neglected entries, that is add them to the diagonal entries in the same row, possibly first multiplied by some scalar Θ,$0 < \Theta \leq 1$. For discussions of such approaches, see [29, 30, 47, 48]. This frequently moves small eigenvalues, corresponding to the smoother harmonics, to cluster near the origin, in this way sometimes improving the spectral condition number by an order of magnitude (see [6, 47]).

The other class of methods are based on approximate inverses G, for instance such that minimizes a Frobenius norm of the error matrix $I - GA$, see Section 5 for further details. To be sufficiently accurate these methods lead frequently to nearly full matrices. This can be understood as the matrices we want to approximate are often sparse discretizations of diffusion problems. The inverse of such an operator is a discrete Green's function which, as wellknown, often has a significantly sized support on nearly the whole domain of definition.

However, we can use an additive approximation of the inverse involving two, or more, terms which is approximate on different vector subspaces. By defining in this way the preconditioner recursively on a sequence of lower dimensional subspaces, it may preserve the accurate approximation property of the full, inverse method while still needing only actions of sparse operators.

Frequently, the given matrices are partitioned in a natural way in a two by two block form. For such matrices, it can be seen that the two approaches are similar. Consider namely

$$A = \begin{bmatrix} A_1 & A_{12} \\ A_{21} & A_2 \end{bmatrix}, \tag{77}$$

where we assume that A_1 and the Schur complement matrix $S = A_2 - A_{21}A_1^{-1}A_{12}$ are nonsingular. (This holds, in particular, if A is symmetric and positive definite.) We can construct either a block approximate factorization of A or approximate the inverse of A on additive form. As the following shows, the approaches are related. First, a block matrix factorization of A is

$$A = \begin{bmatrix} A_1 & 0 \\ A_{21} & S \end{bmatrix} \begin{bmatrix} I_1 & A_1^{-1}A_{12} \\ 0 & I_2 \end{bmatrix}, \tag{78}$$

where I_1, I_2 denote the unit matrices of proper order. For its inverse, it holds

$$\begin{aligned} A^{-1} &= \begin{bmatrix} I_1 & -A_1^{-1}A_{12} \\ 0 & I_2 \end{bmatrix} \begin{bmatrix} A_1^{-1} & 0 \\ -S^{-1}A_{21}A_1^{-1} & S^{-1} \end{bmatrix} \\ &= \begin{bmatrix} A_1^{-1} + A_1^{-1}A_{12}S^{-1}A_{21}A_1^{-1} & -A_1^{-1}A_{12}S^{-1} \\ -S^{-1}A_{21}A_1^{-1} & S^{-1} \end{bmatrix}, \end{aligned} \tag{79}$$

or

$$A^{-1} = \begin{bmatrix} A_1^{-1} & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} -A_1^{-1}A_{12} \\ I_2 \end{bmatrix} S^{-1}\left[-A_{21}A_1^{-1},\ I_2\right]. \tag{80}$$

A straightforward computation reveals that $A_{\widetilde{V}} \equiv \widetilde{V}^T A \widetilde{V} = S$ and, hence,

$$\begin{aligned} A^{-1} &= \begin{bmatrix} A_1^{-1} & 0 \\ 0 & 0 \end{bmatrix} + \widetilde{V}\left(\widetilde{V}^T A \widetilde{V}\right)^{-1}\widetilde{V}^T \\ &= \begin{bmatrix} A_1^{-1} & 0 \\ 0 & 0 \end{bmatrix} + \widetilde{V}A_{\widetilde{V}}^{-1}\widetilde{V}^T, \end{aligned} \tag{81}$$

where

$$\widetilde{V} = \begin{bmatrix} -A_1^{-1}A_{12} \\ I_2 \end{bmatrix}. \tag{82}$$

Let $M_1 \simeq A_1$ be an approximation of A_1 (for which linear systems are simpler to solve than for A_1) and let $G_1 \simeq A_1^{-1}$ be a sparse approximate inverse. Possibly $G_1 = M_1^{-1}$. Then,

$$\begin{aligned} M &= \begin{bmatrix} M_1 & 0 \\ A_{21} & B_2 \end{bmatrix} \begin{bmatrix} I_1 & M_1^{-1}A_{12} \\ 0 & I_2 \end{bmatrix} \\ &= \begin{bmatrix} M_1 & A_{12} \\ A_{21} & A_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & B_2 + A_{21}M_1^{-1}A_{12} - A_2 \end{bmatrix} \end{aligned} \tag{83}$$

is a preconditioner to A and

$$B = \begin{bmatrix} G_1 & 0 \\ 0 & 0 \end{bmatrix} + VB_2^{-1}V^T \tag{84}$$

is an approximate inverse, where $V = \begin{bmatrix} -G_1A_{12} \\ I_2 \end{bmatrix}$ and B_2 is an approximation of S. If $B_2 = V^T\widetilde{A}V$, where $\widetilde{A} = \begin{bmatrix} G_1^{-1} & A_{12} \\ A_{21} & A_2 \end{bmatrix}$, then

$$\begin{aligned} B &= \begin{bmatrix} G_1 & 0 \\ 0 & 0 \end{bmatrix} + V\left(V^T\widetilde{A}V\right)^{-1}V^T \\ &= \begin{bmatrix} G_1 & 0 \\ 0 & 0 \end{bmatrix} + VS\left(\widetilde{A}\right)V^T, \end{aligned} \tag{85}$$

where $S(\widetilde{A}) = A_2 - A_{21}G_1A_{12}$. If $M_1 = G_1^{-1}$, then in this case

$$B = M^{-1}. \tag{86}$$

Hence, a convergence estimate for one method can be directly applied for the other method as well. For further discussions of block matrix preconditioners, see, for example, [49–52]. As can be seen from the above, Schur complement matrices play a major role in both matrix factorizations. For sparse approximations of Schur complement matrix, in particular element, e.g., element type approximations, see, for example [53–55].

We consider now multilevel extensions of the additive approximate inverse subspace correction method. It is illustrative to consider first the exact inverse and its relation to Gaussian (block matrix) elimination.

4.1. The Exact Inverse on Additive Form. Let then $A^{(0)} = A$ and consider a matrix

$$A^{(k)} = \begin{bmatrix} A_1^{(k)} & A_{12}^{(k)} \\ A_{21}^{(k)} & A_2^{(k)} \end{bmatrix}, \tag{87}$$

in a sequence defined by

$$A^{(k+1)} \equiv S_2^{(k)} = A_2^{(k)} - A_{21}^{(k)}A_1^{(k)^{-1}}A_{21}^{(k)}, \qquad k = 0,\ 1, \ldots,\ k_0, \tag{88}$$

where each $A_1^{(k)}$ in nonsingular, being a block diagonal of a symmetric positive definite matrix. Hence, the following recursion holds

$$\begin{aligned} A^{(k)^{-1}} = &\begin{bmatrix} A_1^{(k)^{-1}} & 0 \\ 0 & 0 \end{bmatrix} \\ &+ \begin{bmatrix} -A_1^{(k)^{-1}}A_{12}^{(k)} \\ I_2^{(k+1)} \end{bmatrix} A^{(k+1)^{-1}}\left[A_{21}^{(k)}A_1^{(k)^{-1}},\ I_2^{(k+1)}\right], \end{aligned} \tag{89}$$

$k = 0, 1, \ldots, k_0$. Here, $I_2^{(k+1)}$ is the identity matrix on level $k + 1$. Note that in this example the dimensions decrease with increasing level number and the final matrix (i.e., Schur complement) in the sequence is $A^{(k_0)} = S_2^{(k_0+1)}$. The above recursion can be rewritten in compact form

$$A^{-1} = \begin{bmatrix} A_1^{(0)^{-1}} & 0 \\ 0 & 0 \end{bmatrix} + UD^{-1}L, \tag{90}$$

where the kth column of the block upper triangular matrix U equals $\begin{bmatrix} -A_1^{(k)^{-1}}A_{12}^{(k)} \\ I_2^{(k+1)} \end{bmatrix}$ and $L = U^T$. Further,

$$D = \begin{bmatrix} A_1^{(k)} & & & 0 \\ & A^{(2)} & & \\ & & \ddots & \\ & 0 & & A^{(k_0)} \end{bmatrix}. \tag{91}$$

Hence, this is the (block matrix) Gaussian elimination method applied directly to form the inverse matrix. In this way, there is no need to first form the factorization $A = \widetilde{L}D\widetilde{U}$ and then $A^{-1} = \widetilde{U}^{-1}D^{-1}\widetilde{L}^{-1}$. As is wellknown and readily seen, the columns of $\widetilde{U}^{-1}$ and $\widetilde{L}^{-1}$ are formed directly with no additional computation, from those of $\widetilde{U}$ and $\widetilde{L}$, respectively. Note that $\widetilde{U}^{-1}$ is upper (block) triangular and $\widetilde{L}^{-1}$ is lower (block) triangular).

The matrix D contains the (block) pivot matrices which arise during the factorization. Permutations to increase the stability by finding dominating pivots can be done by replacing $A^{(k+1)}$ with $P^{(k)^T}\widetilde{A}^{(k+1)}P^{(k)}$ where $\widetilde{A}^{(k+1)} = P^{(k)}A^{(k+1)}P^{(k)^T}$ is the permuted matrix on which the next elimination step takes place.

An incomplete factorization method for approximate inverses can be defined by approximating each arising Schur complement matrix with some sparse matrix $\widetilde{B}^{(k+1)}$ and possibly also approximating $A_1^{(k)^{-1}}$ with some matrix $B_1^{(k)}$, to yield the approximate inverse

$$B^{(k)} = \begin{bmatrix} B_1^{(k)} & 0 \\ 0 & 0 \end{bmatrix} + V^{(k)}\widetilde{B}^{(k+1)}V^{(k)^T}, \tag{92}$$

where $V^{(k)} = \begin{bmatrix} -B_1^{(k)}A_{12}^{(k)} \\ I_2^{(k+1)} \end{bmatrix}$.

In forming the approximate Schur complement one can use a simpler matrix $D_1^{(k)}$than $B_1^{(k)}$, often a diagonal matrix suffices. The intermediate Schur complement matrix $\widetilde{S}_2^{(k)} = A_2^{(\mathrm{k})} - A_{21}^{(k)}D_1^{(k)}A_{12}^{(k)}$ can be possibly further approximated by deleting certain off-diagonal entries to preserve sparsity. These entries can be compensated for by modifying the diagonal of $\widetilde{S}_2^{(k)}$ to form the final approximation $\widetilde{B}^{(k+1)}$. Thereby, it can be important to make the approximate Schur complement exact on some particular vector or vector space. (We do not go into these aspects further here, see [43, 56] for details.)

The eigenvectors for the smallest eigenvalues of A provide efficient column vectors for the matrix V to reduce significantly the condition number of $B\,A$ as compared to that of A. However, in general the eigenvectors are not known, and even if they are known it would be costly to apply the corresponding preconditioner as V would be a full matrix. A more viable choice is to let V be defined by the basis functions $\{\varphi_i^{(H)}\}$ of a coarse mesh (or coarsened matrix graph) so that

$$\operatorname{Im} V = \operatorname{span}\left\{\varphi_i^{(H)}\right\}. \tag{93}$$

V, V^T acts then, respectively, as prolongation and restriction operators and

$$V = \begin{bmatrix} J_{12} \\ I_2 \end{bmatrix}, \tag{94}$$

where J_{12} is the interpolation matrix from the coarse mesh (Ω_H) to the fine mesh (Ω_h), and we assume $\Omega_H \subset \Omega_h$. Further, letting the matrices be variationally defined, as in a finite element method, we have

$$A_H = V^TAV, \tag{95}$$

where A is the finite element matrix on the fine mesh.

Now, the eigenvectors for the smaller eigenvalues of A_H are normally accurate approximations of the corresponding eigenvectors for A. Furthermore, the eigenvectors of A_H are members of Im V. Therefore, the matrix V in (94) acts nearly as well as the eigenvector matrix but, in addition, is sparse. Hence the approximate inverse takes the form

$$B = G + \sigma VA_H^{-1}V^T, \tag{96}$$

where $\sigma = \lambda_{\max}(GA)$, $A_H = V^TAV$.

Here, the projection matrix

$$P = VA_H^{-1}V^TA, \tag{97}$$

projects vectors on the subspace Im V, containing normally good approximations of the eigenvectors for the smallest eigenvalues of A, that is, those who may cause severe ill-conditioning. Clearly, the approximation is more accurate as closer Ω_H is to Ω_h. However, the cost of the action of P (mainly the coarse mesh solver for the action of A_H^{-1}) increases when Ω_H expands. One can balance Ω_H to Ω_h in order to let the action of P involve the same order of computational complexity as an action of A, that is, $O(h^{-2})$ for a sparse matrix A. Assuming that the cost of an action of A_H^{-1} is $O(H^{-2.5})$ in a 2D diffusion problem (e.g., using a modified incomplete factorization method as preconditioner for the conjugate gradient method), we find $H = h^{4/5}$. The number of outer iterations with preconditioner B depends also on the choice of G. We refer the discussion of how G can be chosen properly to [56].

As an example, for a model diffusion problem with constant coefficients on a regular mesh, say for the Laplacian operator on unit square, the eigenvectors for the Laplacian $(-\Delta)$ on Ω_h with Dirichlet boundary conditions are

$$v_{k,l}^{(h)} = \sin k\pi \sin l\pi y, \quad x, y \in \Omega_h, \tag{98}$$

where $k, l = 1, 2, \ldots, h^{-1} - 1$, for the eigenvalues

$$\lambda_{k,l}^{(h)} = \left(2\sin\frac{k\pi h}{2}\right)^2 + \left(2\sin\frac{l\pi h}{2}\right)^2. \tag{99}$$

For the coarse mesh, it holds

$$v_{k,l}^{(H)} = \sin k\pi x \sin l\pi y, \quad x, y \in \Omega_H, \tag{100}$$

where k, $l,=$ $1, 2, \ldots, H^{-1}$, and here $Vv_{k,l}^{(H)}$ are good approximations (interpolants) of the eigenvectors $v_{k,l}^{(h)}$ on Ω_h for the smallest eigenvalues.

An alternative choice of matrix V is to take eigenvectors from a nearby problem, normally defined by taking limit values of some problem parameter, see [56].

Multigrid, algebraic multilevel and algebraic multigrid methods have been presented thoroughly in, for example [29, 43, 57, 58]. Because of space limitations, they can not be presented here.

4.2. Symmetrization of Preconditioners; the SSOR and ADI Methods. As we have seen, the incomplete factorization methods require first a factorization step. There exists simpler preconditioning methods that require no factorization but have a form similar to the incomplete factorization methods. We will present two methods of this type. As an introduction, consider first an iterative method of the form

$$M\left(\mathbf{x}^{l+1} - \mathbf{x}^{l}\right) = \mathbf{b} - A\mathbf{x}^{l}, \quad l = 0,\ 1, \ldots, \tag{101}$$

to solve $A\mathbf{x} = \mathbf{b}$, where A and M are nonsingular. As we saw in Section 2, the asymptotic rate of convergence is determined by the spectral radius of the iteration matrix

$$B = I - M^{-1}A. \tag{102}$$

For a method such as the SOR method (which also requires no factorization), with optimal overrelaxation parameter ω (assuming that A has property A^{π} or A is s.p.d., see Section 2), the eigenvalues of the corresponding iteration matrix B are situated on a circle. No further acceleration is then possible.

There is, however, a simple remedy to this, based on taking a step in the forward direction of the chosen ordering, followed by a backward step, that is, a step in the opposite order to the vector components. This method is said to have its origin in the early days of computers when programs were stored on tapes that had to be rewound before a new forward SOR step could begin. It was found that this otherwise useless computer time for the rewinding could be better used for a backward SOR sweep!

As we will see, for symmetric and positive definite matrices the combined forward and backward sweeps correspond to a s.p.d. matrix which, contrary to the SOR method, has the advantage that it can be used as a preconditioning matrix in an iterative acceleration method. This method, called the SSOR method, will be defined later.

For an early discussion of the SSOR method, used as a preconditioner, see [59]. For discussions about symmetrization of preconditioners, see [6, 60, 61]. More generally, if A is s.p.d, we consider the symmetrization of an iterative method in the form

$$\mathbf{x}^{l+1} = \mathbf{x}^{l} + M^{-1}\left(\mathbf{b} - A\mathbf{x}^{l}\right). \tag{103}$$

For the analysis only, we consider the transformed form of (103),

$$\mathbf{y}^{l+1} = \left(I - A^{1/2}\, M^{-1}A^{1/2}\right)\mathbf{y}^{l} + \widetilde{\mathbf{b}}, \tag{104}$$

where

$$\mathbf{y}^{l} = A^{1/2}\mathbf{x}^{l}, \qquad \widetilde{\mathbf{b}} = A^{1/2}M^{-1}\widetilde{\mathbf{b}}. \tag{105}$$

If M is unsymmetric, the iteration matrix $I - A^{1/2}M^{-1}A^{1/2}$ is also unsymmetric. We will now consider a method using M and another preconditioner chosen so that the iteration matrix for the combined method becomes symmetric. We call this the *symmetrization* of the method.

Let M_1, M_2 be two such preconditioning matrices. Let

$$B_i = I - \widetilde{M}_i^{-1}, \qquad \widetilde{M}_i = A^{-1/2}M_iA^{-1/2}, \tag{106}$$

and consider the combined iteration matrix B_2B_1. As we will now see, it arises as an iteration matrix for the combined method

$$\begin{aligned} M_1\left(\mathbf{x}^{l+1/2} - \mathbf{x}^{l}\right) &= \mathbf{b} - Ax^{l}, \\ M_2\left(\mathbf{x}^{l+1} - \mathbf{x}^{l+1/2}\right) &= \mathbf{b} - A\mathbf{x}^{l+1/2}, \quad l = 0,\ 1,\ \ldots. \end{aligned} \tag{107}$$

For the analysis only, we transform this to the form

$$\begin{aligned} \mathbf{y}^{l+1/2} - \mathbf{y}^{l} &= \widetilde{\mathbf{b}}^{(1)} - \widetilde{M}_1^{-1}\mathbf{y}^{l}, \\ \mathbf{y}^{l+1} - \mathbf{y}^{l+1/2} &= \widetilde{\mathbf{b}}^{(2)} - \widetilde{M}_2^{-1}\mathbf{y}^{l+1/2}, \end{aligned} \tag{108}$$

where

$$\widetilde{\mathbf{b}}^{(i)} = A^{(1/2)}M_i^{-1}\mathbf{b}. \tag{109}$$

This iteration takes the form

$$\mathbf{y}^{l+1} = \widetilde{\mathbf{b}}^{(2)} + \left(I - \widetilde{M}_2^{-1}\right)\widetilde{\mathbf{b}}^{(1)} + \left(I - \widetilde{M}_1^{-1}\mathbf{y}^{l}\right), \tag{110}$$

that is,

$$\mathbf{y}^{l+1} = \widetilde{\mathbf{b}}^{(2)} + \left(I - \widetilde{M}_2^{-1}\right)\widetilde{\mathbf{b}}^{(1)} + \left(I - \widetilde{M}_2^{-1}\right)\left(I - \widetilde{M}_1^{-1}\right)\mathbf{y}^{l}, \tag{111}$$

or

$$\mathbf{y}^{l+1} = \widehat{\mathbf{b}} + B_2B_1\mathbf{y}^{l}, \quad l = 0,\ 1, \ldots, \tag{112}$$

where

$$\widehat{\mathbf{b}} = \widetilde{\mathbf{b}}^{(2)} + \left(I - \widetilde{M}_2^{-1}\right)\mathbf{b}^{(1)}. \tag{113}$$

For the following we need a lemma.

Lemma 2. *If A, B, and C are Hermitian positive definite and each pair of them commute, then ABC is Hermitian positive definite.*

Proof. We have $(ABC)^{*} = CBA$ and use commutativity to find

$$CBA = BCA + BAC = ABC. \tag{114}$$

Hence, ABC is Hermitian. Next, we show that the product of two s.p.d matrices that commute is positive definite. We have

$$A^{-1/2}ABA^{1/2}ABA^{1/2} = A^{1/2}BA^{1/2}, \tag{115}$$

which is Hermitian positive definite. Hence, by similarity, the eigenvalues of AB are positive and, since

$$(AB)^{*} = AB, \tag{116}$$

AB is Hermitian positive definite. In the same way, $(AB)C$ is Hermitian positive definite. □

Lemma 3. *Let A be s.p.d. and assume either of the following additional conditions:*

(a) $M_2^* = M_1$.

(b) *M_1, M_2 are s.p.d. $\rho(A^{1/2}M_i^{-1}A^{1/2}) < 1$, $i = 1, 2$, and the pair of matrices M_1, M_2, commutes.*

Then, the combined iteration method (107) *converges if and only if $M_1 + M_2 - A$ is s.p.d.*

Proof. It is readily seen that $\mathbf{y} = \hat{\mathbf{b}} + B_2B_1\mathbf{y}$ (i.e., the iteration method is consistent with $A\mathbf{x} = \mathbf{b}$), where $\mathbf{y} = A^{1/2}\mathbf{x}$ and $\mathbf{x} = A^{-1}\mathbf{b}$. Hence,

$$\mathbf{y} - \mathbf{y}^{l+1} = B_2B_1\left(\mathbf{y} - \mathbf{y}^l\right), \tag{117}$$

and the iteration method (112), and hence (107) converges for any initial vector if and only if $\rho(B_2B_1) < 1$, where $\rho(\cdot)$ denotes the spectral radius. But

$$\begin{aligned} B_2B_1 &= I - \widetilde{M}_1^{-1} - \widetilde{M}_2^{-1} + \widetilde{M}_1^{-1}\widetilde{M}_2^{-1} \\ &= I - A^{1/2}M_1^{-1}(M_1 + M_2 - A)M_2^{-1}A^{1/2}. \end{aligned} \tag{118}$$

It is readily seen that under either of the given conditions (a) or (b),

$$M_1^{-1}(M_1 + M_2 - A)M_2^{-1} = M_1^{-1} + M_2^{-1} - M_1^{-1}AM_2^{-1} \tag{119}$$

is symmetric. Further, it is positive definite if and only if $M_1 + M_2 - A$ is positive definite. Hence, $I - B_2B_1$ is s.p.d. Further, $B_2B_1 = (I - \widetilde{M}_2^{-1})(I - \widetilde{M}_1^{-1})$ is symmetric, and a similarity transformation shows that B_2B_1 is similar to $(I - \widetilde{M}_2^{-1})^{1/2}\ (I - \widetilde{M}_1^{-1})(I - \widetilde{M}_2^{-1})^{1/2}$, which is a congruence transformation of $I - \widetilde{M}_1^{-1}$, whose eigenvalues are positive. Hence, B_2B_1 has positive eigenvalues, so the eigenvalues of B_2B_1 are contained in the interval $(0, 1)$ and, in particular, $\rho(B_2B_1) < 1$. □

The proof of Lemma 3 shows that B_2B_1 is symmetric, so the combined iteration method is a *symmetrized version* of either of the simple methods.

Let us now consider a special class of symmetrized methods. We let A be split as $A = D + L + U$, where we assume that D is s.p.d., and let

$$V = \left(1 - \frac{1}{\omega}\right)D + L, \qquad H = \left(1 - \frac{1}{\omega}\right)D + U, \tag{120}$$

$\widehat{D} = (2/\omega - 1)D$, where ω is a parameter, $0 < \omega < 2$. (Here, L and U are not necessarily the lower and upper triangular parts of A.) Note that

$$\widehat{D} + V + H = A, \tag{121}$$

so this is also a splitting of A. As an example of a combined, or symmetrized, iteration method, we consider the preconditioning matrix

$$C = \left(\widehat{D} + V\right)\widehat{D}^{-1}\left(\widehat{D} + H\right), \tag{122}$$

and show that this leads to a convergent iteration method

$$C\left(\mathbf{x}^{l+1} - \mathbf{x}^l\right) = \mathbf{b} - A\mathbf{x}^l, \qquad l = 0,\ 1, \ldots. \tag{123}$$

This corresponds to choosing $M_1 = \widehat{D}^{-1/2}(\widehat{D} + H)$ and $M_2 = (\widehat{D} + V)\widehat{D}^{-1/2}$, and it can be seen that the conditions of Lemma 3 hold if the conditions in the next theorem hold.

Theorem 8. *Let $A = D + L + U$, where D is s.p.d. Let $V, H, \widehat{D}$ be defined by* (120), *and assume that either (a) or (b) holds, where*

(a) $U = L^*$

(b) *L, U are s.p.d. and each pair of matrices L, U, D commute. Then the eigenvalues λ of the matrix $C^{-1}A$, where C is defined in* (122), *are contained in the interval* $0 < \lambda \le 1$.

Proof. This can be shown either by verifying the conditions in Lemma 3 or more directly as follows. As in the proof of Lemma 3, it follows that C is s.p.d. Hence, the eigenvalues of $C^{-1}A$ are positive. Further,

$$C = \widehat{D} + V + H + V\widehat{D}^{-1}H, \tag{124}$$

so, by (121)

$$C = A + V\widehat{D}^{-1}H. \tag{125}$$

Under either condition (a) or (b), $C = V\widehat{D}^{-1}H$ is symmetric and positive semidefinite.

This shows that $\mathbf{x}^*C\mathbf{x} \ge \mathbf{x}^*A\mathbf{x}$ for all $\mathbf{x}$, so the eigenvalues of $C^{-1}A$ are bounded above by 1. □

We will now show that the matrix C can also efficiently be used as a preconditioning matrix, which for a proper value of the parameter ω, and under an additional condition, can even reduce the order of magnitude of the condition number. In this respect, note that when C is used as a preconditioning matrix for the Chebyshev iterative method, it is not necessary to have C scaled so that $\lambda(C^{-1}A) \le 1$, because it is suffices then that $0 < m \le \lambda(C^{-1}A) \le M$, for some numbers m, M. Hence, the factor $2/\omega - 1$ in $\widehat{D}^{-1}$ can be neglected.

Theorem 9. *Let $A = D + L + U$ be a splitting of A, where A and D are s.p.d. and either (a) $U = L^*$ or (b) L, U are s.p.d. and each pair of D, L, U commute. Then, the eigenvalues of matrix $C^{-1}A$, where*

$$C = \left(\frac{1}{\omega}D + L\right)\widehat{D}^{-1}\left(\frac{1}{\omega}D + U\right) \tag{126}$$

and $0 < \omega < 2$, $\widehat{D} = (2/\omega - 1)D$, are contained in the interval

$$\left[\frac{(2-\omega)}{\left\{1 + \omega(1/\omega - 1/2)^2\delta^{-1} + \omega\gamma\right\}},\ 1\right], \tag{127}$$

where

$$\begin{aligned} \delta &= \min_{\mathbf{x}\neq 0}\frac{\mathbf{x}^TA\mathbf{x}}{\mathbf{x}^TD\mathbf{x}} \\ \gamma &= \max_{\mathbf{x}\neq 0}\frac{\mathbf{x}^T(LD^{-1}U - 1/4D)\mathbf{x}}{\mathbf{x}^TA\mathbf{x}}. \end{aligned} \tag{128}$$

Further, if there exists a vector for which $\mathbf{x}^T(L+U)\mathbf{x}^T(L+U)\mathbf{x} \le 0$, then $\gamma \ge -1/4$, and if

$$\rho\left(\widetilde{L}\widetilde{U}\right) \le \frac{1}{4}, \tag{129}$$

then $\gamma \le 0$, and if

$$\rho\left(\widetilde{L}\widetilde{U}\right) \le \frac{1}{4} + O(\delta), \quad \text{then } \gamma \le O(1),\ \delta \longrightarrow 0. \tag{130}$$

Here, $\widetilde{L} = D^{-1/2}LD^{-1/2}$.

Proof. It is readily seen that

$$\begin{aligned} C &= \frac{1}{2-\omega}\left(\frac{1}{\omega}D + L\right)\left(\frac{1}{\omega}D\right)^{-1}\left(\frac{1}{\omega}D + U\right) \\ &= \frac{1}{2-\omega}\left[A + \left(\frac{1}{\omega} - 1\right)D + \omega LD^{-1}U\right] \\ &= \frac{1}{2-\omega}\left[A + \omega\left(\frac{1}{\omega} - \frac{1}{2}\right)^2 D + \omega\left(LD^{-1}U - \frac{1}{4}D\right)\right]. \end{aligned} \tag{131}$$

This shows the lower bound in (127); the upper bound follows by Theorem 8. By choosing a vector for which $\mathbf{x}^T(L+U)\mathbf{x} \le 0$, it follows that

$$\frac{\mathbf{x}^T(LD^{-1}U - (1/4)D)\mathbf{x}}{\mathbf{x}^T A\mathbf{x}} \ge \frac{\mathbf{x}^T(LD^{-1}\mathbf{x} - (1/4)D\mathbf{x})}{\mathbf{x}^T D\mathbf{x}} \ge -\frac{1}{4}, \tag{132}$$

which shows $\gamma \ge -1/4$. The remainder of the theorem is immediate. □

4.2.1. The Condition Number. Theorem 9 shows that the optimal value of ω to minimize the upper bound of the condition number of $C^{-1}A$ is the value that minimizes the real-valued function

$$f(\omega) = \frac{1 + \omega(1/\omega - 1/2)^2\delta^{-1} + \omega\gamma}{2-\omega}. \tag{133}$$

It is readily seen (see Axelsson and Barker, 1984 [30]), that $f(\omega)$ is minimized for

$$\begin{aligned} \omega^* &= \frac{2}{1 + 2\sqrt{(1/2+\gamma)\delta}}, \\ \min_{\omega} f(\omega) &= f(\omega^*) = \sqrt{\left(\frac{1}{2} + \gamma\right)\delta^{-1}} + \frac{1}{2}. \end{aligned} \tag{134}$$

In general, δ is not known, but we may know that $\delta = O(h^2)$, for some problem parameter, $h \to 0$ (such as for the step length in second-order elliptic problems). Then, if $\gamma = O(1)$, $h \to 0$, we let $\omega = 2/(1+\xi h)$ for some $\xi > 0$, in which case

$$f(\omega) = O(h^{-1}) = O\left(\sqrt{\delta^{-1}}\right), \quad h \longrightarrow 0. \tag{135}$$

This means that $C^{-1}A$ has an order of magnitude smaller condition number than A itself, which latter is $O(\delta^{-1})$.

We consider now two applications of Theorem 9.

4.2.2. The SSOR Method. In the first case, L is the lower triangular part of A or the lower block triangular part, if A is partitioned in block matrix form and $U = L^*$. Then,

$$C = \frac{1}{2-\omega}\left(\frac{1}{\omega}D + L\right)\left(\frac{1}{\omega}D\right)^{-1}\left(\frac{1}{\omega}D + L^*\right), \tag{136}$$

is a symmetrized version of the SOR method and is called the SSOR (*symmetric successive overrelaxation*) method.

As an example, for an elliptic differential equation of second order it can be seen that the condition $\rho(\widetilde{L}\widetilde{L}^T) \le 1/4$ holds for problems with Dirichlet boundary conditions and constant coefficients. For extensions of this, see Axelsson and Barker [30]. For the model difference equation on a square domain with side π, we have

$$\delta = 2\left(\sin\frac{h}{2}\right)^2, \quad \gamma \le 0, \tag{137}$$

and we find

$$\begin{aligned} \omega^* &= \frac{2}{1 + 2\sin h/2} \sim \frac{2}{1+h}, \\ f(\omega^*) &= \sqrt{\frac{1}{2\delta}} + \frac{1}{2} \sim h^{-1} + \frac{1}{2}, \quad h \longrightarrow 0. \end{aligned} \tag{138}$$

4.3. The ADI Method. In the second case of methods of (101), we let L denote the off-diagonal part of the difference operator working in the x-direction and U off-diagonal part of the difference operator in the y-direction. D is its diagonal part. Then, the matrix

$$\widehat{C} = \left(\frac{1}{\omega}D + L\right)\left(\frac{1}{\omega}D\right)^{-1}\left(\frac{1}{\omega}D + U\right), \tag{139}$$

is called an *alternating direction preconditioning matrix* and the corresponding iteration method is called the ADI (alternating direction iteration) method. In this method, we solve alternately one-dimensional difference equations in x- and y-directions. Much has been written on the ADI-method which was originally presented in Peaceman and Rachford [62]; see Varga [10], for an early influential presentation and Birkhoff et al. [63] and Wachspress [64], for instance.

As we will see, for the model difference equations we get the same optimal value of ω as in (138). The condition $\gamma = O(1)$ may be less restrictive, for the ADI-method, but the condition of commutativity is much more restrictive, as the following lemma shows.

Lemma 4. *Let A, B be two Hermitian matrices of order n. Then $AB = BA$ if and only if A and B have a common set of orthonormal eigenvectors.*

Proof. If such a common set of eigenvectors $\{\mathbf{v}_i\}$ exists, then $A\mathbf{v}_i = \sigma_i\mathbf{v}_i$, $B\mathbf{v}_i = \tau_i\mathbf{v}_i$ and

$$AB\mathbf{v}_i = \sigma_i\tau_i\mathbf{v}_i \qquad i = 1,\ 2, \ldots, n. \tag{140}$$

Since the eigenvector space of an Hermitian matrix is complete, we therefore have

$$AB\mathbf{x} = BA\mathbf{x}, \quad \forall \mathbf{x} \in \mathcal{C}^n, \tag{141}$$

which shows that $AB = BA$. Conversely, suppose that $AB = BA$. As A is Hermitian, take U to be a unitary matrix that diagonalizes A, that is

$$\widetilde{A} = UAU^* = \begin{bmatrix} \gamma^1 I_1 & & & 0 \\ & \gamma^2 I_2 & & \\ & & \ddots & \\ 0 & & & \gamma_r I_r \end{bmatrix}, \tag{142}$$

where $\gamma^1 < \gamma^2 < \cdots < \gamma_r$ are the distinct eigenvalues of A and I_j is the identity matrix of order n_j, the multiplicity of γ_j. (Here A is posssibly permuted accordingly.) Let $\widetilde{B} = U\ BU^*$ and partition $\widetilde{B}$ corresponding to the partitioning of A, that is,

$$\widetilde{B} = \begin{bmatrix} B_{11}B_{12} & \cdots B_{1r} \\ \vdots & \\ B_{r1}B_{r2} & \cdots B_{rr} \end{bmatrix}. \tag{143}$$

Since $AB = BA$, we have

$$\widetilde{A}\widetilde{B} = U\ ABU^* = U\ BAU^* = \widetilde{B}\widetilde{A}. \tag{144}$$

Carying out the block multiplication $\widetilde{A}\widetilde{B} = \widetilde{B}\widetilde{A}$, we find that this, in turn, implies $B_{ij} = 0$, $i \neq j$, since $\gamma_i \neq \gamma_j$, $i \neq j$. Simply stated, a (block) matrix commutes with a (block) diagonal matrix if and only if it is itself (block) diagonal. Hence, $\widetilde{B}$ is block diagonal and each Hermitian submatrix $B_{i,i}$ has n_i orthonormal eigenvectors that are also eigenvectors of the submatrix $\gamma_i I_i$ of $\widetilde{A}$. Since $\sum_{i=1}^{r} n_i = n$ and all eigenvectors are orthonormal, A and B must have the same set of eigenvectors. □

For the second-order elliptic difference equation in two space dimensions, it turns out that the commutativity of L and U essentially corresponds to the property that the original problem is separable, that is, that solutions of $\mathcal{L}u = f$ can be written in the form $u = \varphi(x)\psi(y)$. This means that the coefficients $a(x, y)$ and $b(x, y)$ in the differential operator $\partial/\partial x[a(x, y)\partial u/\partial x] + \partial/\partial y[b(x, y)\partial u/\partial x] + c(x, y)u$ must satisfy $a(x, y) = a(x)$,$b(x, y) = b(y)$, and $c(x, y) = c$, a constant. Hence, if $a(x, y) = b(x, y)$, then $a(x, y) = b(x, y) = a$, a constant. Furthermore, the convex closure of the meshpoints must be a rectangle with sides parallel to the coordinate axes (Varga, [10] 1962). If $A = A_1 + A_2$, the ADI-method can be written in the form

$$\begin{aligned} (I + \tau_1 A_1)\mathbf{x}^{l+1/2} &= (I - \tau_1 A_2)\mathbf{x}^{l} + \tau_1 \mathbf{b}, \\ (I + \tau_2 A_2)\mathbf{x}^{l+1} &= (I - \tau_2 A_1)\mathbf{x}^{l+1/2} + \tau_2 \mathbf{b}, \qquad l = 0,\ 1, \ldots. \end{aligned} \tag{145}$$

This is the *Peaceman-Rachford* [62] *iteration method.* The iteration matrix M is similar to

$$(I - \tau_2 A_1)(I + \tau_1 A_1)^{-1}(I - \tau_1 A_2)(I + \tau_2 A_2)^{-1}. \tag{146}$$

When A_1, A_2 are Hermitian positive definite, their eigenvalues $\lambda_i^{(1)}$, $\lambda_i^{(2)}$ are positive, and

$$\begin{aligned} \left\| (I - \tau_2 A_1)(I + \tau_1 A_1)^{-1} \right\|_2 &= \rho\left((I - \tau_2 A_1)(I + \tau_1 A_1)^{-1} \right) \\ &= \max_i \left| \frac{1 - \tau_2 \lambda_i^{(1)}}{1 + \tau_1 \lambda_i^{(1)}} \right|. \end{aligned} \tag{147}$$

Thus,

$$\begin{aligned} \rho(M) &= \rho\left((I - \tau_2 A_1)(I + \tau_1 A_1)^{-1}(I - \tau_1 A_2)^{-1}(I + \tau_2 A_2)^{-1} \right) \\ &\leq \left\| (I - \tau_2 A_1)(I + \tau_1 A_1)^{-1}(I - \tau_1 A_2)(I + \tau_2 A_2)^{-1} \right\|_2 \\ &\leq \left\| (I - \tau_2 A_1)(I + \tau_1 A_1)^{-1} \right\|_2 \\ &\quad \times \left\| (I - \tau_1 A_2)(I + \tau_2 A_2)^{-1} \right\|_2 \\ &= \mu(\tau_1, \tau_2), \end{aligned}$$

$$\rho(M) \leq \mu(\tau_1, \tau_2) = \max_i \left| \frac{1 - \tau_2 \lambda_i^{(1)}}{1 + \tau_1 \lambda_i^{(1)}} \right| \max_i \left| \frac{1 - \tau_1 \lambda_i^{(2)}}{1 + \tau_2 \lambda_i^{(2)}} \right|. \tag{148}$$

Note that for $\tau_1 = \tau_2 = \tau > 0$, $\mu(\tau_1, \tau_2) = \mu(\tau, \tau) < 1$, so we have $\rho(M) < 1$, that is convergence for any $\tau > 0$. This holds even if A_1, A_2 do not commute. Note also that when A_1 and A_2 commute, we have

$$\rho(M) = \mu(\tau_1,\ \tau_2). \tag{149}$$

Let us continue the analyses for the general case where A_1, A_2 do not necessarily commute. We want to compute the optimal values of τ_1 and τ_2 such that $\mu(\tau_1, \tau_2)$ is minimized. For simplicity, we assume that α, β are the same lower and upper bounds of the eigenvalues of A_1 and A_2, that is, $0 < \alpha \leq \lambda_i^{(j)} \leq \beta$, $j = 1, 2$. We have

$$\begin{aligned} \mu(\tau_1, \tau_2) \leq \max&\left\{ \left| \frac{1 - \tau_2 \alpha}{1 + \tau_1 \alpha} \right|, \left| \frac{1 - \tau_2 \beta}{1 + \tau_1 \beta} \right| \right\} \\ \times \max&\left\{ \left| \frac{1 - \tau_1 \alpha}{1 + \tau_2 \alpha} \right|, \left| \frac{1 - \tau_1 \beta}{1 + \tau_2 \beta} \right| \right\}. \end{aligned} \tag{150}$$

We want to choose $\tau_1, \tau_2 > 0$ such that this bound is as small as possible. Note, then, that for such values of τ_1, τ_2 we must have $1 - \tau_i \alpha > 0$ and $1 - \tau_i \beta < 0$. Next note that each factor in the bound (150) is minimized when

$$\frac{1 - \tau_2 \alpha}{1 + \tau_1 \alpha} = \frac{\tau_2 \beta - 1}{\tau_1 \beta + 1}, \qquad \frac{1 - \tau_1 \alpha}{1 + \tau_2 \alpha} = \frac{\tau_1 \beta - 1}{\tau_2 \beta + 1}, \tag{151}$$

respectively, that is, when

$$\begin{aligned}\tau_1\tau_2 - \frac{\alpha+\beta}{2\alpha\beta}(\tau_1-\tau_2) - \frac{1}{\alpha\beta} &= 0,\\ \tau_1\tau_2 + \frac{\alpha+\beta}{2\alpha\beta}(\tau_1-\tau_2) - \frac{1}{\alpha\beta} &= 0,\end{aligned} \tag{152}$$

respectively. Thus, both factors are simultaneously minimized when $\tau_1 = \tau_2$, and then $\tau_1 = \tau_2 = 1/\sqrt{\alpha\beta}$.

Theorem 10. *Let $A = A_1 + A_2$, where A_1, A_2, are s.p.d., and consider the Peaceman-Rachford ADI method* (145) *to solve $A\mathbf{x} = \mathbf{b}$ with $\tau_1 = \tau_2 = 1/\sqrt{\alpha\beta}$. The spectral radius of the corresponding iteration matrix M satisfies*

$$\rho(M) \le \min_{\tau_1,\tau_2} \mu(\tau_1,\tau_2) \le \left(\frac{1-\sqrt{\alpha/\beta}}{1+\sqrt{\alpha/\beta}}\right)^2 \sim 1 - 4\sqrt{\frac{\alpha}{\beta}}, \tag{153}$$

if $\alpha/\beta \to 0$.

Proof. For $\tau_1 = \tau_2 = 1/\sqrt{\alpha\beta}$, we have

$$\mu(\tau_1,\tau_2) = \left(\frac{1-\sqrt{\alpha/\beta}}{1+\sqrt{\alpha/\beta}}\right)^2. \tag{154}$$

□

Remark 2. For a model difference equation for a second-order elliptic differential equation problem on a square with side π, we have with stepsize h,

$$\begin{aligned}\alpha &= \left(\frac{\sin(h/2)}{h/2}\right)^2 \sim 1,\\ \beta &= \left(\frac{\cos(h/2)}{h/2}\right)^2 \sim \frac{4}{h^2}, \quad h \longrightarrow 0.\end{aligned} \tag{155}$$

Then,

$$\begin{aligned}\mu(\tau_1,\tau_2) &= \left(\frac{1-\tan(h/2)}{1+\tan(h/2)}\right)^2\\ &= \frac{1-\sin(h)}{1+\sin(h)} \sim 1 - 2h, \quad h \longrightarrow 0.\end{aligned} \tag{156}$$

Note that this is just the convergence factor we get for the SOR method with an optimal overrelaxation parameter.

Since $\rho(M) \le \mu(\tau_1,\tau_2)$, this means that the ADI method with parameters (chosen as above) converges at least as fast as the SOR method. Note, however, that in the ADI-method we must solve two systems of equations with tridiagonal coefficient matrices $(I - \tau A_i)$ on each step, while the pointwise SOR method requires no solution of such systems.

4.4. The Commutative Case. Assume now that A_1, A_2 commute. Then, as we have seen, M is symmetric and has real eigenvalues, and we can apply the Chebyshev acceleration method. The eigenvalues of the corresponding preconditioned matrix $\widetilde{C}$ are related to the eigenvalues of M by

$$\lambda\left(\widetilde{C}\right) = 1 - \lambda(M). \tag{157}$$

Since $-\rho(M) \le \lambda(M) \le \rho(M)$, where

$$\rho(M) = \left(\frac{1-\sqrt{\alpha/\beta}}{1+\sqrt{\alpha/\beta}}\right)^2 \sim 1 - 4\sqrt{\frac{\alpha}{\beta}}, \tag{158}$$

we have

$$\begin{aligned}4\sqrt{\frac{\alpha}{\beta}} &\sim 1 - \rho(M) \le \lambda\left(\widetilde{C}\right)\\ &= 1 + \rho(M) \sim 2 - 4\sqrt{\frac{\alpha}{\beta}} \sim 2, \quad \frac{\alpha}{\beta} \longrightarrow 0.\end{aligned} \tag{159}$$

The asymptotic rate of convergence of the Chebyshev accelerated method therefore is

$$2\sqrt{\frac{1-\rho(M)}{1+\rho(M)}} \sim 2\sqrt{2}\left(\frac{\alpha}{\beta}\right)^{1/4}, \quad \alpha/\beta \longrightarrow 0. \tag{160}$$

For the model difference equation, we have the asymptotic rate of convergence

$$\sim 2h^{1/2}, \quad h \longrightarrow 0. \tag{161}$$

4.5. The Cyclically Repeated ADI Method. The real power of the ADI method is brought forth when we use a sequence of parametrs τ_l. Assume that A_1, A_2 commute, then we choose the parameters τ_l cyclically. With a cycle of q parameters and with the assumption

$$0 < \alpha \le \lambda_i^{(j)} \le \beta, \quad j = 1,\ 2, \tag{162}$$

we get the iteration matrix

$$M^{(q)} = \prod_{p=1}^{q} \left(I + \tau_p A_2\right)^{-1}\left(I - \tau_p A_1\right)\left(I + \tau_p A_1\right)^{-1}\left(I - \tau_p A_2\right). \tag{163}$$

The eigenvalues of $M^{(q)}$ are

$$\prod_{p=1}^{q} \frac{1-\tau_p\lambda_i^{(1)}}{1+\tau_p\lambda_i^{(1)}} \cdot \frac{1-\tau_p\lambda_i^{(2)}}{1+\tau_p\lambda_i^{(2)}}. \tag{164}$$

In the same way as above, $\rho(M^{(q)})$ is minimized when

$$d(\alpha,\beta,q) = \max_{\alpha\le x\le\beta} \prod_{p=1}^{q} \left|\frac{1-\tau_p x}{1+\tau_p x}\right|. \tag{165}$$

4.6. A Preconditioning Method for Complex Valued Matrices. Complex valued systems of equations arise in many applications. A commonly occuring case is the solution of a matrix polynomial equation

$$Q_m(A)x = b, \tag{166}$$

where A is a real square matrix and Q_m is a polynomial of degree m that has no zeroes at the eigenvalues of A. Here Q_m can be factored in the product of second degree, and possibly some factors of first degree polynomials with real coefficients.

The second degree polynomials can be factored in products of first degree polynomials with complex coefficients.

Consider then a linear system

$$Az = b. \tag{167}$$

in the form

$$(\mathcal{R} + iS)(x + iy) = c + id, \tag{168}$$

where $\mathcal{R}, S$ are real matrices of order n and $x, y, c, d \in \mathcal{R}^n$.

The system can be solved in complex arithmetic. However, complex arithmetic leads to heavier computational complexity and it is in general difficult to precondition complex valued matrices, as the eigenvalues of the given matrix or the preconditioned matrix can be spread in the whole complex plane and the iterative solution method may then converge too slowly.

One can alternatively apply a preconditioned conjugate gradient method to the Hermitian positive definite normal matrix system $A^H Au = A^H b$ for which the eigenvalues are real. At any rate, this involves complex arithmetic that costs typically three to four times as much as corresponding real arithmetic.

Complex arithmetic can be avoided by rewriting (168) in real valued form, such as

$$A^{(1)}\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} R & -S \\ S & R \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} c \\ d \end{bmatrix}, \tag{169}$$

or

$$A^{(2)}\begin{bmatrix} x \\ -y \end{bmatrix} = \begin{bmatrix} R & S \\ S & -R \end{bmatrix}\begin{bmatrix} x \\ -y \end{bmatrix} = \begin{bmatrix} c \\ d \end{bmatrix}. \tag{170}$$

The block matrices are here real but, in general, nonsymmetric and/or indefinite. For the solution, one can use a generalized conjugate gradient method such as GMRES [40] or GCG [65].

For $A^{(1)}$ it holds that any eigenvalue λ appears in complex conjugate pairs $\lambda, \overline{\lambda}$. For $A^{(2)}$, which is real symmetric, for any eigenvalue $\lambda \neq 0, -\lambda$ is also an eigenvalue. Thus, the spectrum $\sigma(A^{(1)})$ is symmetric with respect to the real axes and the spectrum $\sigma(A^{(2)})$ is symmetric with respect to the imaginary axes, that is, in both cases the spectrum embraces the origin. From best polynomial approximation properties it is known that such point distributions leads to polynomials of essentially a square degree as for the same approximation accuracy compared to the case with a one-sided spectrum.

In [21], one finds further explanations why Krylov subspace methods can be inefficient for solving complex valued systems, represented in the above real forms. Several iterative solution methods, such as the QMR [22] have been developped and proven to be efficient for these types of problems. However, it is difficult to precondition complex valued matrices and unpreconditioned methods converge in general very slowly.

Following [66], we consider here instead an approach based on rewriting the equation in the form (170).

Instead of solving the full block matrix system we apply a Schur complement approach by the elimination of one component, which results in the following reduced system

$$Cx = f, \tag{171}$$

where

$$\begin{aligned} C &= R + SR^{-1}S, \\ f &= c + SR^{-1}d. \end{aligned} \tag{172}$$

As an introduction, assume first that R is symmetric and positive definite and S is symmetric and positive semidefinite. As a preconditioner to the matrix C in (171) we take $R + S$.

For the generalized eigen value problem,

$$\mu(R + S)z = (R + SR^{-1}S)z, \tag{173}$$

it holds then

$$\mu(I + H)y = (I + H^2)y, \tag{174}$$

where $\mu = R^{1/2}z$ and $H = R^{-1/2}SR^{-1/2}$.

If $\lambda Rz = Sz$, $z \neq 0$, or, equivalently, $Hy = \lambda y$, $y \neq 0$, it follows from (174) that

$$\mu = \frac{1 + \lambda^2}{(1 + \lambda)^2}, \tag{175}$$

that is,

$$\mu = \frac{1}{1 + (2\lambda/(1 + \lambda^2))}. \tag{176}$$

Since, by assumption $\lambda \geq 0$, it follows

$$\frac{1}{2} \leq \mu \leq 1, \tag{177}$$

and the condition number satisfies the bound $\mathcal{K}((R + S)^{-1}C) \leq 2$.

The correspondingly preconditioned conjugate gradient method to solve (174) converges therefore rapidly

There exists an even more efficient form of the iteration matrix that also shows that we can weaken the assumptions made on R and S. Hence, consider

$$\begin{aligned} Rx - Sy &= c, \\ Sx + Ry &- d \end{aligned} \tag{178}$$

and assume that α is a real parameter such that $R + \alpha S$ is nonsingular. Such a parameter exists if ker $(R) \cap$ ker $(S) = \emptyset$.

Multiplying the first equation by $-\alpha(R+\alpha S)^{-1}$, the second by $(R+\alpha S)^{-1}$ and adding yields

$$(R+\alpha S)^{-1}(S-\alpha R)x+y=(R+\alpha S)^{-1}(d-\alpha c). \quad (179)$$

Now multiplying this equation by S, using $Sy=Rx-c$, and rewriting the equation properly, we find

$$r\equiv Rx-c+S(R+\alpha S)^{-1}((S-\alpha R)x+\alpha c-d)=0. \quad (180)$$

When solving the system by iteration, such as by Chebyshev iterations, r will be the residual and we observe that r can be written in the form (see (179))

$$\begin{gathered} y=(R+\alpha S)^{-1}((S-\alpha R)x+\alpha c-d),\\ r=Rx-Sy-c. \end{gathered} \quad (181)$$

In this form there is no need to compute the right hand side vector f initially as if (168) is used and the vector y is found during the iteration process. This saves two solutions with the matrix $R+\alpha S$. Since we need few iterations, such a saving can be important to decrease the total expense of the method.

To solve (179) we use $R+\alpha S$ as a preconditioner. The resulting preconditioned matrix takes the form

$$\begin{aligned} M_\alpha &= (R+\alpha S)^{-1}\left[R+S(R+\alpha S)^{-1}(S-\alpha R)\right]\\ &= (R+\alpha S)^{-1}[(R+\alpha S)-\alpha S](R+\alpha S)^{-1}R\\ &\quad + (R+\alpha S)^{-1}S(R+\alpha S)^{-1}S\\ &= \left((R+\alpha S)^{-1}R\right)^2+\left((R+\alpha S)^{-1}S\right)^2. \end{aligned} \quad (182)$$

This form can also be used to derive eigenvalue estimates in more general cases than was done above. If R is nonsingular, we find

$$M_\alpha=(I+\alpha R^{-1}S)^{-2}(I+R^{-1}S)^2. \quad (183)$$

Therefore, the preconditioned matrix is a rational function in the matrix $R^{-1}S$. It follows that the eigenvalues μ of E_α satisfy

$$\mu=\frac{1+\lambda^2}{(1+\alpha\lambda)^2}, \quad (184)$$

were λ is an eigenvalue of $R^{-1}S$.

We want to choose α to minimize the spectral condition number

$$\mathcal{K}(M_\alpha)=\frac{\mu_{\max}}{\mu_{\min}}. \quad (185)$$

Theorem 11. *Assume that R is s.p.d and S is s.p.s-d. Then, the extreme eigenvalues of the preconditioned matrix M_α, defined in* (182) *satisfy*

$$\begin{gathered} \mu_{\min}=\begin{cases}\dfrac{1}{1+\alpha^2} & \textit{if } 0\le\alpha\le\hat{\lambda},\\ \dfrac{1+\hat{\lambda}^2}{\left(1+\alpha\hat{\lambda}\right)^2} & \textit{if } \alpha\ge\hat{\lambda},\end{cases}\\ \mu_{\max}=\begin{cases}1 & \textit{if } \hat{\alpha}\le\alpha,\\ \dfrac{1+\hat{\lambda}^2}{\left(1+\alpha\hat{\lambda}\right)^2} & \textit{if } 0\le\alpha\le\hat{\alpha},\end{cases} \end{gathered} \quad (186)$$

where $\hat{\lambda}$ is the maximal eigenvalue of $R^{-1}S$,

$$\begin{gathered} R^{-1}S\le\hat{\lambda}I,\\ \hat{\alpha}=\frac{\hat{\lambda}}{1+\sqrt{1+\hat{\lambda}^2}}. \end{gathered} \quad (187)$$

The spectral condition number is minimized when $\alpha=\hat{\alpha}$, in which case

$$\begin{gathered} \mu_{\min}=\frac{1}{1+\hat{\alpha}^2},\quad \mu_{\max}=1,\\ \mathcal{K}\left(M_\alpha\right)=1+\hat{\alpha}^2=2\frac{\sqrt{1+\hat{\lambda}^2}}{1+\sqrt{1+\hat{\lambda}^2}}. \end{gathered} \quad (188)$$

Proof. The bounds of the extreme eigenvalues follow by elementary computations of $\mu=(1+\lambda^2)/(1+\alpha\lambda)^2$, $0\le\lambda\le\hat{\lambda}$. Similarly, it is readily seen that $\mu_{\max}/\mu_{\max}$ is minimized for some α in the interval $\hat{\alpha}\le\alpha\le\hat{\lambda}$, where $\mu_{\max}=1$. Hence, it is minimized for $\alpha=\arg\max_{\hat{\alpha}\le\alpha}(1+\alpha^2)^{-1}$, that is, for $\alpha=\hat{\alpha}$. □

For applications, see [66]. An important application arises when one uses Padé type approximations, and related implicit Runge-Kutta methods (see [67]), to solve initial value problems.

4.7. Historical Remarks.

Because incomplete factorization methods has had a strong influence on the development of preconditioning methods we give here some historical remarks.

The idea of an incomplete factorization method goes back to early papers by Buleev [68], Varga [10], Oliphant [69], Dupont et al. [70], Dupont [71], and Woźnički [72], where it was presented for matrices of a type arising from difference approximations of elliptic problems. The first more general form (unmodified methods for pointwise matrices) was studied for M-matrices by Meijerink and van der Vorst [46]. For a review and general formalism for describing such methods, see Axelsson [18], Birkhoff et al. [63], Beauwens [12], and Il'in [60]. For a similar but more involved type of methods for difference matrices, which allowed for variable parameters from one iteration to the next, see Stone [73].

A modified form of the method, where a certain row sum criterion was imposed, was studied by Gustafsson [47]. Actually, as is readily seen, the method of Dupont et al. [70] and as further discussed in Axelsson [59], using a perturbation technique, can be seen as a modified version of the general incomplete factorization method when applied to the five-point elliptic difference matrices, assuming that no fill-in is accepted outside the sparsity structure of A itself and assuming a natural ordering of the grid points. The advantage of modified versions is that they can give condition numbers of the iteration matrices that are of an order of magnitude smaller than for the original matrix.

The incomplete factorization method can be readily generalized to matrices partitioned in block matrix form. This was done first for matrices partitioned in block tridiagonal form in Axelsson et al. [49] and Concus et al. [50], the latter being based on earlier work by Underwood [74]. A general form was presented in Axelsson [67] and Beauwens and Ben Bouzid [52], where existence of the method was proven for M-matrices.

The existence, that is, the existence of nonzero pivot entries of pointwise incomplete factorization methods for M-matrices was first shown by Meijerink and van der Vorst [46] and, for pointwise H-matrices, by Varga et al. [75]. The existence of incomplete factorization methods for M-matrices in block form was shown in Axelsson et al. [49] and Concus et al. [50] for block tridiagonal matrices; in Axelsson [76] and Beauwens and Ben Bouzid [52], for general block matrices; and in Axelsson and Polman [51] for relaxted versions of such methods.

Kolotilina [77] shows the existence of convergent splittings for block H-matrices, and Axelsson [78] shows the existence of general incomplete factorizations for block H-matrices.

5. Approximate Inverses Methods

In many applications, it is of interest to compute approximations of the inverse (A^{-1}) of a given matrix A, such that these approximations can be readily used in various iterative methods.

Let G denote an approximation of A^{-1}.

Following [6], first we present an example of an explicit and an implicit method, which is followed by a general framework for computing approximate inverses. At the end, we present an efficient way to construct symmetric and positive definite approximate inverses.

An approximate inverse to a given operator may be constructed in several ways. The simplest way is to use a Neumann expansion, that is let $D^{-1}A = I - E$, where D is the diagonal of A, for instance.

Assuming that $\|E\| < 1$, then the expansion

$$A^{-1} = (I - E)^{-1}D^{-1} = (I + E + E^2 + \cdots)D^{-1} \tag{189}$$

is convergent and any truncated part of this series provides an approximate inverse. However, this will normally give poore approximations. As we will see, more accurate approximate inverses can be constructed as best, possibly weighted, Frobenius norm approximations.

In many applications the matrix A is sparse, but the exact inverse will be just a full matrix. A natural condition on G then arises: we can impose that G has some a priori chosen sparsity pattern (the same as A or different) which will make the calculations with G easy and cheap, and also will provide a sufficient accuracy.

Let A have order n and $\underline{S} = \{(i, j), 1 \le i \le n;\ 1 \le i \le j \le n\}$. Any proper subset S of $\underline{S}$ will be referred to as a sparsity pattern $S \subset \underline{S}.S_L$ denotes the corresponding sparsity pattern for the lower triangular matrix and $S_{\widetilde{L}}$ denotes the corresponding sparsity pattern for the strictly lower triangular matrix.

For simplicity, we use the same notation S for matrices having sparsity pattern S. Thus, $A \in S$ if $a_{ij} \neq 0 \Leftrightarrow (i, j) \in S$.

5.1. Explicit Methods . In these methods, an approximation of the inverse A^{-1} of a given nonsingular matrix A is computed explicitly, that is, without solving a linear globally coupled system of equations.

Let S be a sparsity pattern. We want to compute $G \in S$, such that

$$(GA)_{ij} = \delta_{ij}, \quad (i, j) \in S, \tag{190}$$

that is

$$\sum_{k:(i,k)\in S} g_{ik}a_{kj} = \delta_{ij}, \quad (i, j) \in S. \tag{191}$$

Some observations can be made from (191):

(i) the elements in each row of G can be computed independently;

(ii) even if A is symmetric, G in not necessarily symmetric, because $g_{i,j}$, $j \neq i$, and $g_{j,i}$ are, in general, not equal.

5.2. Implicit Methods . These methods require that A is factored first. In practice, they are used mainly for band or "envelope" matrices. The algorithm was presented in [79]. It is based on an idea in [80]; see also [81].

Suppose that $A = LD^{-1}U$ is a triangular matrix factorization of A. If A is a band matrix then L and U are also band matrices.

Let

$$L = I - \widetilde{L}, \quad U = I - \widetilde{U}, \tag{192}$$

where $\widetilde{L}$ and $\widetilde{U}$ are strictly lower and upper triangular matrices correspondingly.

The following lemma holds.

Lemma 5. *Using the above notations it holds that*

(i) $A^{-1} = DL^{-1} + \widetilde{U}A^{-1}$,

(ii) $A^{-1} = U^{-1}D + A^{-1}\ \widetilde{L}$.

Proof. Consider the following

$$A = LD^{-1}U \Longrightarrow A^{-1} = U^{-1}DL^{-1} \Longrightarrow \left(I - \widetilde{U}\right)A^{-1} \\ = DL^{-1} \Longrightarrow A^{-1} = DL^{-1} + \widetilde{U}A^{-1}. \tag{193}$$

Also,

$$A^{-1}\left(I - \widetilde{L}\right) = U^{-1}D \Longrightarrow A^{-1} = U^{-1}D + A^{-1}\widetilde{L}. \tag{194}$$

□

Since DL^{-1} is lower triangular and $\widetilde{U}$ is upper triangular, using (i) we can compute entries in the upper triangular part of A^{-1} with no need to use entries of L^{-1}. Similarly, using (ii) we can compute entries of the lower triangular part A^{-1} without computing U^{-1}.

Suppose now that A is a block banded matrix with a semibandwidth p, and we want to form A^{-1} also as block banded with a semibandwidth $q : q \geq p$. The identities (i) and (ii) can be used then for the computation of the upper and lower parts of A^{-1}.

Remark 3. (i) The algorithm involves only matrix × matrix operations.

(ii) There is no need to compute any entries outside the bands.

(iii) If A is symmetric then it suffices executing only (i) or (ii).

(iv) It can be seen that $(A^{-1})_{nn} = D_{nn}^{-1}$.

There are two drawbacks with the above algorithm. It requires first the factorization $A = LD^{-1}U$ and even if A is s.p.d, the band matrix part of A^{-1}, which is computed, need not be s.p.d. The next example illustrates this.

Example 2. Consider an s.p.d. matrix

$$G = \begin{bmatrix} 1 & -2 & 1 \\ -2 & 5 & -3 \\ 1 & -3 & 4 \end{bmatrix}, \\ G_{\text{band}} = \begin{bmatrix} 1 & -2 & 0 \\ -2 & 5 & -3 \\ 0 & -3 & 4 \end{bmatrix}, \tag{195}$$

is indefinite.

5.3. A General Framework for Computing Approximate Inverses. It turns out that both the explicit and implicit method can be characterized as methods to compute best approximations of A^{-1} of all matrices having a given sparsity pattern, in some norm. The basic idea is due to Kolotilina and Yeremin [38, 79], see [6]. Recall that the trace function is defined by $\operatorname{tr}(A) = \sum_{i=1}^{n} a_{ii}$, which also equals $\sum_{i=1}^{n} \lambda_i(A)$. Let a sparsity pattern S be given. Consider the functional

$$F_W(G) \equiv \|I - GA\|_W^2 = \operatorname{tr}\left((I - GA)W(I - GA)^T\right), \tag{196}$$

where the weight matrix W is s.p.d. If $W \equiv I$ then $\|I - GA\|_I$ is the Frobenius norm of $I - GA$.

Clearly $F_W(G) \geq 0$. If $G = A^{-1}$ then $F_W(G) = 0$. We want to compute the entries of G in order to minimize $F_W(G)$, that is, to find $\widehat{G} \in S$, such that

$$\left\|I - \widehat{G}A\right\|_W \leq \|I - GA\|_W, \quad \forall G \in S. \tag{197}$$

The following properties of the trace function will be used

$$\operatorname{tr} A = \operatorname{tr} A^T, \ \operatorname{tr}(A + B) = \operatorname{tr} A + \operatorname{tr} B. \tag{198}$$

Then,

$$\begin{aligned} F_W(G) &= \operatorname{tr}(I - GA)W(I - GA)^T \\ &= \operatorname{tr}\left(W - GAW - W(GA)^T + GAW(GA)^T\right) \\ &= \operatorname{tr} W - \operatorname{tr} GAW - \operatorname{tr}(GAW)^T + \operatorname{tr} GAWA^TG^T. \end{aligned} \tag{199}$$

Further, as we are interested in minimizing F_W with respect to $G \in S$, we consider the entries $g_{i,j}$ as variables. The necessary condition for a minimizing point are then

$$\frac{\partial F_W(G)}{\partial g_{ij}} = 0, \quad (i, j) \in S. \tag{200}$$

From (199) and (200), we get

$$-2\left(WA^T\right)_{ij} + 2\left(GAWA^T\right)_{ij} = 0, \tag{201}$$

or

$$\left(GAWA^T\right)_{ij} = \left(WA^T\right)_{ij}, \quad (i, j) \in S. \tag{202}$$

Depending on the particular matrix A and the choice of S and W, (202) may or may not have a solution. We give some examples where a solution exists.

Example 3. Let A be s.p.d. Choose $W = A^{-1}$ which is also s.p.d. Then, (202) implies

$$(GA)_{ij} = \delta_{ij}, \quad (i, j) \in S, \tag{203}$$

which is the formula for the previously presented explicit method which, hence, is a special case of the more general framework for computing approximate inverses using weighted Frobenius norm.

Example 4. Let $W = (A^TA)^{-1}$. Then (202) implies

$$(G)_{ij} = (A^{-1})_{ij}, \quad (i, j) \in S, \tag{204}$$

which is the relation for the previously presented implicit method. In this case the entries of G are the corresponding entries of the exact inverse.

Example 5. Let $W = I$. Then,

$$\begin{gathered} F_W(G) = n - \operatorname{tr}(GA), \\ \left(GAA^T\right)_{ij} = \left(A^T\right)_{ij}, \quad (i,j) \in S. \end{gathered} \tag{205}$$

This method is also explicit.

We can expect that such methods will be accurate only if all elements of A which are not used in the computations are zero or are relatively small. In some cases the quality of the computed approximation G to A^{-1} can be significantly improved using diagonal compensation of the entries of A which are outside S. The best approximation G to A^{-1} in a (weighted) Frobenius norm is in general not symmetric and, as we have seen, not always positive definite. For this reason, the next, alternate method, is considered.

5.4. Constructing a Symmetric and Positive Definite Approximate Inverse.

For some methods (as in the preconditioned Chebyshev and the conjugate gradient iteration methods) it is of importance to use s.p.d. preconditioners. As we have seen, the methods described till now do not guarantee that G will be such a matrix.

In order to compute an s.p.d. approximate inverse of an s.p.d. matrix, we can proceed as follows. It will be shown that this approximation gives a best approximation to minimize the K-condition number of the correspondingly preconditioned matrix.

A Symmetric and Positive Definite Approximate Factorized Inverse. Seek an approximate inverse in the form $G = LL^T$, where $L \in S_L$,

$$S_L = \{(i,j) \in S,\ i \geq j\}, \tag{206}$$

and let

$$S_{\widetilde{L}} = \{(i,j) \in S_L,\ i > j\}, \tag{207}$$

that is, denote by S_L and $S_{\widetilde{L}}$ the lower and strictly lower triangular part of the sparsity set S.

Theorem 12. *Let A be s.p.d. and consider matrices L with sparsity pattern S_l. Let the matrix $\widehat{L}$ be computed by the following steps.*

(i) *Compute first $\widetilde{L}$ such that*

$$\left(\widetilde{L}A\right)_{ij} = A_{ij}, \quad (i,j) \in S_{\widetilde{L}}, \tag{208}$$

and $\widetilde{L}_{ij} = 0$, $(i,j) \in S^c_{\widetilde{L}}$ (the complement set).

(ii) *Let $\widehat{L} = D(I - \widetilde{L})$, where*

$$\begin{gathered} D = \operatorname{diag}(d_1, d_2, \ldots, d_n), \\ d_i = \frac{1}{\left[\left(I - \widetilde{L}\right)A\left(I - \widetilde{L}^T\right)\right]_{i,i}^{1/2}}. \end{gathered} \tag{209}$$

Then, $\widehat{L} \in S_L$ and minimizes the K-condition number of $\widehat{L}A\widehat{L}^T$, that is,

$$\frac{\left((1/n)\operatorname{tr}\left(\widehat{L}A\widehat{L}^T\right)\right)^n}{\det\left(\widehat{L}A\widehat{L}^T\right)} = \inf_{L \in S_L} \frac{((1/n\ )\operatorname{tr}(LAL^T))^n}{\det(LAL^T)}. \tag{210}$$

Proof. Let $D =$ diag $(d_1, \ldots, d_n)$ denote the diagonal part of a matrix $X \in S_L$ and let $\widetilde{X} = I - D^{-1}X$, that is, $\widetilde{X} \in S_{\widetilde{L}}$. Then,

$$\begin{aligned} &\frac{((1/n)\operatorname{tr}(XAX^T))^n}{\det(XAX^T)} \\ &\quad = \frac{((1/n)\sum_i (XAX^T)_{ii})^n}{(\det(X))^2 \det(A)} \\ &\quad = \frac{\left((1/n)\sum_i \left[D\left(I - \widetilde{X}\right)A\left(I - \widetilde{X}^T\right)D\right]_{i,i}\right)^n}{(\det(X))^2 \det(A)} \\ &\quad = \frac{\left((1/n)\sum_i d_i^2\left[\left(I - \widetilde{X}\right)A\left(I - \widetilde{X}^T\right)\right]_{i,i}\right)^n}{(\Pi_i d_i^2)\det(A)} \\ &\quad = \frac{((1/n)\ \sum_i \alpha_i^2)^n}{\Pi_i \alpha_i^2} \cdot \frac{\Pi_i\left[\left(I - \widetilde{X}\right)A\left(I - \widetilde{X}^T\right)\right]_{i,i}}{\det(A)}, \end{aligned} \tag{211}$$

where $\alpha_i^2 = d_i^2[(I - \widetilde{X})A(I - \widetilde{X}^T)]_{i,i}$.

Note now that

$$\Pi_i\left[\left(I - \widetilde{X}\right)A\left(I - \widetilde{X}^T\right)\right]_{i,i}, \tag{212}$$

does not depend on d_i, so we can minimize this factor independently of d_i.

Consider then the general weighted Frobenius norm minimization problem

$$\min_{G \in S} \operatorname{tr}(I - GB)W(I - GB)^T. \tag{213}$$

As we have seen, its solution G satisfies the relation

$$\left(GBWB^T\right)_{i,j} = \left(WB^T\right)_{i,j}, \quad \forall (i,j) \in S. \tag{214}$$

Let now $G = \widetilde{X}$, $W = A$, $B = I$, $S = S_{\widetilde{L}}$. Then,

$$\left(GBWB^T\right)_{ij} = \left(WB^T\right)_{ij} \tag{215}$$

takes the form

$$\left(\widetilde{X}A\right)_{ij} = A_{ij} \quad \forall i,j \in S_{\widetilde{L}}. \tag{216}$$

This is an explicit method and since the minimization is done rowwise it follows from (213), with the chosen matrices G, B and W, that each of

$$\left[\left(I - \widetilde{X}\right)A\left(I - \widetilde{X}\right)^T\right]_{i,i} \quad i = 1, \ldots, n \tag{217}$$

is minimized separately. By construction $\widetilde{L}$ satisfies (216), so the minimization problem is has the solution $\widetilde{X} = \widetilde{L}$. Hence,

$$\min_{\widetilde{\mathrm{X}}} \Pi_i \left[\left(I - \widetilde{X}\right) A \left(I - \widetilde{X}\right)^T \right]_{i,i} = \Pi_i \left[\left(I - \widetilde{L}\right) A \left(I - \widetilde{L}\right)^T \right]_{i,i}. \tag{218}$$

Consider next the first factor in (211). Here,

$$\frac{\left((1/n) \sum_j \alpha_j^2\right)^n}{\Pi_j \alpha_j^2} \geq 1, \tag{219}$$

since a geometric average is less or equal to an arithmetic average. Equality is taken if and only if all α_j are equal and with no limitation we can take $\alpha_j = 1$, $j = 1, \ldots, n$. Hence,

$$d_i^2 = \frac{1}{\left[\left(I - \widetilde{L}\right) A \left(I - \widetilde{L}\right)^T \right]_{i,i}} \tag{220}$$

which completes the proof. □

The method above provides a simple and cheap method to compute approximate inverses on factorized form. The proof of the theorem shows that the K-condition number is reduced in a way as follows from the next corollary.

Corollary 2. *Let $\widehat{\mathrm{L}}$, D be defined as in Theorem 12. Then,*

$$\begin{aligned} K\left(\widehat{L} A \widehat{L}^T\right) &\equiv \frac{\left((1/n) \operatorname{tr}\left(\widehat{L} A \widehat{L}^T\right)\right)^n}{\det\left(\widehat{L} A \widehat{L}^T\right)} \\ &= \frac{\Pi_1^n \left[\left(I - \widetilde{L}\right) A \left(I - \widetilde{L}^T\right) \right]_{ii}}{\det(A)}, \end{aligned} \tag{221}$$

where $\widetilde{\mathrm{L}} = I - D^{-1} L$.

Hence, the trace is replaced by a product, that is the n'th power of the arithmetic average is replaced the n'th power of a geometric average. This is illustrated in the next example.

Example 6. Let $S_L = \{(1,1), (2,2), \ldots, (n,n)\}$, that is, let L be a diagonal matrix. Then, we find $d_i^2 = a_{ii}$ and Corollary 2 shows that

$$K\left(LAL^T\right) = \min_{L \in S_L} \frac{((1/n) \operatorname{tr}(LAL^T))^n}{\det(LAL^T)} = \frac{\Pi_1^n a_{ii}}{\det(A)}, \tag{222}$$

which is to be compared with

$$K(A) = \frac{((1/n) \operatorname{tr}(A))^n}{\det(A)} = \frac{\left((1/n) \sum_1^n a_{ii}\right)^n}{\det(A)}, \tag{223}$$

that is, we have

$$K\left(LAL^T\right) = \left(\frac{\underline{g}}{\underline{a}}\right)^n K(A). \tag{224}$$

Hence, the K-condition number $K(LAL^T)$ of the diagonally scaled matrix LAL^T is substantially smaller than $K(A)$ if the geometric average $\underline{g}$ of the diagonal entries a_{ii} of A are much smaller than their arithmetic average $\overline{a}$. This holds when the entries a_{ii} vary significantly. Note that it always holds that $\underline{g} \leq \overline{a}$.

We conclude this section by mentioning that the K-condition number can be take large values even for seemingly harmless eigenvalue distributions.

Example 7 (Arithmetic distribution). Let $0 < \lambda_1 < \lambda_2 < \cdots < \lambda_n$, the eigenvalues of B be distributed uniformly as an arithmetic sequence in the interval $[a, b]$, $a = \lambda_1$, $b = \lambda_n$. For simplicity, assume that $n/2$ is even. Then,

$$K(B) = \frac{(b + a)/2)^n}{\prod_1^n \lambda_i}. \tag{225}$$

On the other hand, (57) shows $k \leq (1/2)\sqrt{b/a} \ln 2/\varepsilon$, which is asymptotically smaller than n if $(b/a)n^{-2} = o(1)$. In particular, if b/a does not depend on n then we have $k \leq O(\ln 1/\varepsilon)$. Therefore, the estimate in Theorem 6 inferior even to the simple estimate in (56). For other distributions, however, Theorem 6 can give a smaller upper bound.

6. Augmented Subspace Preconditioning Method

6.1. Introduction; Preconditioners for Very Ill-Conditioned Problems. In this section, we consider the solution of systems $A\mathbf{x} = \mathbf{b}$, where A is an $n \times n$ matrix which is symmetric and positive definite (s.p.d) and can have a very large condition number, that is, be *ill-conditioned.* Such systems arise typically for near-limit values of some problem parameter. (Ratio of material coefficients, aspect ratio of the domain, nearly incompressible materials in elasticity theory, etc.) The condition number can be additionally very large due to the size of the matrix A (a small value of the discretization parameter) and also due to an irregular mesh and/or large aspect ratios of the mesh in partial differential equation (PDE) problems.

If the size of the system is not too large one can use direct solution methods, possibly coupled with an iterative refinement method.

Let $B = LDL^T$ (or $B = \widetilde{L}\widetilde{L}^T$) be a triangular matrix or the Cholesky factorization of A. Due to finite precision computations (say, in single precision) in general B is only an approximation of A. The iterative refinement method takes the following form.

Algorithm 1 (Iterative refinement method). Given $\mathbf{x}^{(0)} = 0$

```
    for $k = 0, 1, 2, \ldots$, until convergence
        (i) compute $\mathbf{r}^{(k)} = \mathbf{b} - A\mathbf{x}^{(k)}$,
        (ii) solve $B\mathbf{d}^{(k)} = \mathbf{r}^k$,
        (iii) let $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \mathbf{d}^{(k)}$, and repeat (i)–(iii).
```

Frequently, it suffices with one iterative refinement step. The ability of iterative refinement to produce a more accurate solution vector depends crucially on how the computation of the residual vector $\mathbf{r}^{(k)}$ in (i) is implemented. A safe way is to use double precision for this computation but possibly single precision in (ii) and (iii). However, as described in [30], if one rewrites the computation of $A\mathbf{x}^{(k)}$ as a sum of differences, in some cases it suffices to use single precision in (i) also.

The computational labor is normally dominated by the initial factorization of A. For large systems this cost can become too big as it grows in general fast with problem size. (For an elliptic difference problem on a 3D $N \times N \times N$ mesh it grows as $O(N^7)$ for certain band-matrix orderings. Furthermore, the demand of memory to store the factor L grows as $O(N^5)$. For certain nested dissection and other orderings, the complexity is somewhat reduced, however.)

Therefore, iterative solution methods become the ultimate methods of choice. As we have seen in Section 1, the basic idea behind the iterative solution technique is to use a cheaper (incomplete) factorization or other approximation B of A and to compensate for this approximation by repeating the steps in the iterative refinement method until the residual is sufficiently small. In addition, to speed up the convergence of the method, one or more acceleration parameters are introduced, for instance, (iii) becomes $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)}$ for certain parameters α_k.

By a proper choice of B the number of required iterations may not be too big while the expense in solving systems with matrix B may not be larger than the order of some "work units", for instance, can correspond to a few actions (matrix-vector multiplications) of A on a vector. In this way, one can gain significantly in computational labor and less demand of memory resources as compared with a direct solver. Actually, the direct solver can be viewed as an approximate factorization with the full amount of fill-in allowed, while as we have seen, in a incomplete factorization method one controls the amount of fill-in either by using a predetermined sparsity pattern in L or by allowing a variable pattern, which depends on some relative drop tolerance. (Such a drop tolerance is to delete a fill-in entry a_{ij} if there holds $|a_{ij}| \leq \varepsilon\sqrt{a_{ii}a_{jj}}$, $j \neq i$ for some ε, $0 < \varepsilon < 1$. More details can be found in [82]).

A problem with iterative solution methods for ill-conditioned systems is that they may stagnate, that is, there is no further improvement as the method proceeds. This occurs typically for minimum residual or minimum A-norm methods. For other type of methods even divergence may be observed. Another problematic issue is the fact that if the residual norm has taken a small value, this does not necessarily mean that the error norm is sufficiently small, since

$$\begin{aligned}\left\|\mathbf{x} - \mathbf{x}^{(k)}\right\|_2 &= \left\|A^{-1}A\left(\mathbf{x} - \mathbf{x}^{(k)}\right)\right\|_2 \leq \|A^{-1}\|_2\left\|\mathbf{r}^{(k)}\right\|_2 \\ &= \frac{1}{\lambda_{\min}(A)}\left\|\mathbf{r}^{(k)}\right\|_2,\end{aligned} \tag{226}$$

and here $\lambda_{\min}(A)$ takes very small values for ill-conditioned systems. Hence, even if $\|\mathbf{r}^{(k)}\|$ is small, $\|\mathbf{x} - \mathbf{x}^{(k)}\|$ may still be large. For ill-conditioned systems one sees then typically a reduction of the residual to some limit value while the errors hardly decay at all. This was illustrated in Section 3. For studies on the influence of inexact arithmetics, see for example [83–85].

This situation can be significantly improved by using a proper preconditioner. Then,

$$\begin{aligned}\left\|\mathbf{x} - \mathbf{x}^{(k)}\right\|_2 &= \left\|\left(B^{-1}A\right)^{-1}B^{-1}A\left(\mathbf{x} - \mathbf{x}^{(k)}\right)\right\|_2 \\ &\leq \frac{1}{\lambda_{\min}(B^{-1}A)}\left\|\widetilde{\mathbf{r}}^{(k)}\right\|_2,\end{aligned} \tag{227}$$

where $\widetilde{\mathbf{r}}^{(k)} = B^{-1}A(\mathbf{x} - \mathbf{x}^{(k)}) = B^{-1}\mathbf{r}^{(k)}$ is the so called *preconditioned* or *pseudo-residual*. Here $\lambda_{\min}(B^{-1}A) \gg \lambda_{\min}(A)$ with a proper preconditioner. Therefore, the importance of choosing a proper preconditioner is twofold:

(1) to increase the rate of convergence while keeping the expense in solving systems with B low, and

(2) to enable a small error norm when the pseudoresidual is small.

Preconditioning methods, such as the modified incomplete factoriztion method, multigrid and multilevel methods, aim at reducing arror components correponding both to the large eigenvalues with rapidly oscillating components and the smaller eigenvalues for smoother eigen functions. In the modified method, this is partly achieved by letting the preconditioner by exact for a particular smooth component of the solution, such as for the constant component vector. It has been shown, see [6, 47] when applied for elliptic difference problems, that under certain conditions the spectral condition number is reduced from $O(h^{-2})$ to $O(h^{-1})$. In multigrid methods, one works on two or more levels of meshes where the finer grid component should smooth out the fast, oscillating components in the iteration error, while the coarser mesh should handle the smooth components. Under certain conditions, such methods way reduce the above condition number to optimal order, $O(1)$, as $h \to 0$.

The multigrid method was first introduced for finite difference methods in the 1960s by Fedorenko [86], and Bakhvalov [87], and further developed and advocated by Brandt in the 1970s, see, for example, Brandt [88]. For finite elements it has been pursued by, for example, Braess [89], Hackbusch [90], Bramble et al. [91], Mandel et al. [92], McCormick [57], Bramble et al. [93] and Bank et al. [94], among others.

As it turns out, such standard preconditioning methods, namely (modified) incomplete factorization ((M)ILU), [46, 47], Multigrid (MG) [90], or Algebraic Multilevel Iteration (AMLI), [95–97], methods may not be efficient in both and in particular, in the second of the above mentioned requirements. This might be due to the fact that the smallest eigenvalue (in the preconditioned system) is caused by some problem parameter which these methods leave unaffected. Therefore there is a demand for new types of preconditioners (or new combinations of already known preconditioners). To satisfy the above need, two types of preconditioners have been constructed:

(a) deflation methods,

(b) augmented matrix methods,

which we now describe.

6.2. Deflation Methods. The deflation technique is based on a projection matrix. Assume that A has a number of (very) small eigenvalues, say $\widetilde{m}$, $0 < \lambda_1 \leq \lambda_2 \ldots \leq \lambda_{\widetilde{m}}$, and let $\mathcal{W} = \{\mathbf{w}^{(i)}\}$, $i = 1, \ldots, \widetilde{m}$ be their corresponding eigenvectors ($A\mathbf{w}_i = \lambda_i \mathbf{w}_i$). Let V be a rectangular matrix of order $n \times m$, where $m < n$ (in practice $m \ll n$) of full rank, where the m columns of V span a subspace γ, such that $\operatorname{Im} \gamma$ contains the eigenvectors corresponding to the "bad" subspace $\mathcal{W}$. Hence, $m \geq \widetilde{m}$.

Lemma 6. *Let $P = AVA_V^{-1}V^T$, where $A_V = V^TAV$. Then, the following holds:*

(a) $P^2 = P$, *that is a projector;*

(b) $P(AV) = AV$;

(c) $(I - P)\mathbf{b} = 0$ *if* $\mathbf{b} \in \mathcal{I}m(AV)$;

(d) $P^TV = V$;

(e) $(I - P)A$ *is symmetric and positive definite and has a nullspace of dimension m.*

Proof. Note first that A_V is nonsingular since V has a full rank ($= m$). The statements follow now by straightforward computations. □

Lemma 6 shows that P is projection matrix which maps any vector onto AV. Similarly, P^T is a projection matrix which maps Vonto itself. We will use the matrix P in three slightly different ways to solve ill-conditioned systems

We split first the right-side vector $\mathbf{b}$ in two components:

$$\mathbf{b} = P\mathbf{b} + (I - P)\mathbf{b}. \tag{228}$$

(These components are A^{-1} orthogonal, i.e., $(Pb)^TA^{-1}(I - P)\mathbf{b} = 0$.) The first splits the computation of thesolution vector corresponding.

Method 1 (Splitting of the solution vector).

Let

$$\mathbf{x}^{(0)} = VA_V^{-1}V^T\mathbf{b}. \tag{229}$$

Then,

$$A\mathbf{x}^{(0)} = P\mathbf{b}. \tag{230}$$

Solve

$$A\mathbf{z} = (I - B)\mathbf{b}. \tag{231}$$

The solution $\mathbf{x}$ of $A\mathbf{x} = \mathbf{b}$ is then

$$\mathbf{x} = \mathbf{x}^{(0)} + \mathbf{z}. \tag{232}$$

Here, $\mathbf{x}^{(0)}$ and $\mathbf{z}$ are A-orthogonal.

Note that $A\mathbf{z} = \mathbf{b} - A\mathbf{x}^{(0)}$. The matrix A_V is normally of small order and the arising system in (229) can be solved with relatively little expense using a direct solution method. Furthermore, the system (231) is well-conditioned on the solution subspace, because, as follows from part (c) of Lemma 6, $(I - P)\mathbf{b}$, and hence $\mathbf{z}$ do not contain components of any of the first m "small" eigenvectors w_i, $i = 1, 2, \ldots, m$. Hence, (231) can be solved by the CG method with a rate of convergence determined by the *effective condition number* λ_n/λ_{m+1}, which is expected to be substantially smaller than λ_n/λ_1.

However, the method requires exact solution of systems with A_V and for some problems m is not that small. Also, it is assumed that the projection $P\mathbf{b}$ is computed exactly (or to a sufficient accuracy), which may be unfeasible in many applications.

Method 2 (Defect-correction with projectors). In the presence of round-off errors, $\mathbf{x}^{(0)}$ may not be sufficiently accurate and $\mathbf{b} - A\mathbf{x}^{(0)}$ may still contain components in the "bad" subspace. A defect-correction (iterative refinement) procedure may then help. Let $\mathbf{x}^{(0)} = 0$ for $k = 0, 1, 2, \ldots$, until convergence. Compute $\mathbf{r}^{(k)} = \mathbf{b}$-$A\mathbf{x}^{(k)}$. Solve $A\mathbf{d}^{(k)} = \mathbf{b}$–$A\mathbf{x}^{(k)}$ as follows:

(i) $\mathbf{d}^{(k,0)} = VA_V^{-1}V^T\mathbf{r}^{(k)}$,

(ii) $A\mathbf{z}^{(k)} = (I - P)\mathbf{r}^{(k)}$, or $A\mathbf{y}^{(k)} = \mathbf{r}^{(k)} - A\mathbf{d}^{(k,0)}$,

(iii) $\mathbf{d}^{(k)} = \mathbf{d}^{(k,0)} + \mathbf{y}^{(k)}$.

Let $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \mathbf{d}^{(k)}$.

In this method, it normally suffices with few defect-correction steps.

For some extremely ill-conditioned systems, the implementation of the defect-correction method as a preconditioning method may be necessary. Note then that as follows from Lemma 6, $(I - P)A$ is symmetric so the standard conjugate gradient method can be used.

Method 3 (Preconditioning by a projection matrix). Let $\mathbf{x}^{(0)} = VA_V^{-1}V^T\mathbf{b}$. Solve $(I - P)A\mathbf{z} = (I - P)\mathbf{b}$, or $(I - P)A\mathbf{z} = \mathbf{b} - A\mathbf{x}^{(0)}$ by CG iteration. Let $\mathbf{x} = \mathbf{z} + \mathbf{x}^{(0)}$.

Note that $\mathbf{x}^{(0)}$ is contained in the null-space of $(I - P)A$, since $(I - P)A\mathbf{x}^{(0)} = (I - P)P\mathbf{b} = (P^2 - P)\mathbf{b} = 0$. Here, the system $(I - P)A$ is well-conditioned on the orthogonal complement to the null-space and, in addition, the right-hand-side has no or only small components in the bad subspace.

Methods 1, 2, and 3 require accurate solution of systems with the matrix A_V. It is a viable step for small values of m. However, when the dimension of the "bad" subspace of A is relatively big, it may be too costly. Furthermore, the iteration Method 3 involves two multiplications with A (one involved in P and one required to compute $A\mathbf{z}^{(k)}$) at each iteration step when computing the search direction vectors and is therefore particularly expensive.

Another issue to comment on is that the above methods are assumed to move the components of the eigenvectors for the smallest eigenvalues of A to become exactly zero. However, this can be sensitive to perturbations and occurs only in exact arithmetic. As we have seen, it is a viable method for small dimensions of the subspace causing the ill-conditioning but it may be inefficient for larger dimensions.

Deflation methods have been used and analysed by [98–100], among others.

In the next section and Section 4, we present a method which move the small eigenvalues to the cluster of bigger eigenvalues which is much less dependent on having the right subspace spanned by the columns of V and which do not require exact solution of systems with A_V.

6.3. Augmented Matrix Preconditioning Methods, the Ideal Case. We now present an alternative method to handle ill-posed problems. In this method the small eigenvalues are moved to the cluster of bigger eigenvalues, instead of being deflated to zero, as in the deflation method. The method is an extension of the method presented in [101]. The presentation here is based in [25, 102]. First, we consider $B = I + VV^T$ as a (multiplicative) preconditioner to A.

In this case one must scale the column vectors appearing in V properly. A method involving an automatic scaling is based on a projection matrix. Let then

$$B = I + \sigma V A_V^{-1} V^T, \qquad A_V = V^T A V. \tag{233}$$

Let $(\lambda_i, \mathbf{v}_i)_{i=1}^m$ be the eigenpairs of A for the smallest eigenvalues, $0 < \lambda_1 \le \lambda_2 \le \cdots \lambda_m$. If $V = [\mathbf{v}_1, \ldots, \mathbf{v}_m]$, we get then

$$\widetilde{\lambda}_i = \lambda_i(BA) = \begin{cases} \lambda_i + \sigma, & 1 \le i \le m, \\ \lambda_i, & m+1 \le i \le n. \end{cases} \tag{234}$$

Hence, σ determines how much the smallest eigenvalues are moved. If $\lambda_{m+1} \le \lambda_1 + \sigma$ and $\lambda_m + \sigma \le \lambda_n$, then $\lambda_{m+1} \le \widetilde{\lambda}_i \le \lambda_n$, that is, the m smallest eigenvalues have been moved to the cluster $[\lambda_{m+1}, \lambda_n]$ of bigger eigenvalues and the spectral condition number of BA is $\kappa(BA) = \lambda_n/\lambda_{m+1}$, which normally means a significant reduction, compared to $\kappa(A) = \lambda_n/\lambda_1$.

The above illustrates what can be achieved in an ideal case. In practice, the exact eigenvectors $\{\mathbf{v}_1, \ldots, \mathbf{v}_m\}$ (or the subspace spanned by them) are not known. Even if the eigenvectors are known it is not efficient to use them to form the matrix V because they are in general not sparse and the matrix vector multiplications with V will be costly. Hence, in practice other vectors which are sparse but spans about the same subspace as the smoothest eigenvectors must be used; otherwise the expense of the preconditioner would be too high. We consider therefore more general subspaces spanned by the column vectors of V. The next lemma will be useful.

Lemma 7. *Let A be s.p.d. Then,*

$$P = A^{1/2} V \left(V^T A V\right)^{-1} V^T A^{1/2}, \tag{235}$$

is an orthogonal projection, that is, $P^2 = P$ and $P^* = P$. Therefore, the only eigenvalues of P are 0 and 1.

Proof. Consider the following

$$\begin{aligned} P^2 &= A^{1/2} V \left(V^T A V\right)^{-1} V^T A^{1/2} A^{1/2} V \left(V^T A V\right)^{-1} V^T A^{1/2} \\ &= A^{1/2} V \left(V^T A V\right)^{-1} V^T A^{1/2} = P. \end{aligned} \tag{236}$$

□

The next theorem shows (what can also be expected) that the clustering can never get worse for expanding subspaces spanned by the column-vectors of V, that is, there holds a monotonicity principle.

Theorem 13 (¡xref ref-type="bibr" rid="B97"/¿])). *Let A and $\widehat{A}$ be s.p.d. matrices of order $n \times n$ and let V_k be rectangular matrices of order $n \times m_k$, $k = 1, 2$ such that* $\operatorname{rank} V_k = m_k$, *$k = 1, 2$. If $\operatorname{Im} V_1 \subseteq \operatorname{Im} V_2$, then for all i, $1 \le i \le n$ the following inequality holds*

$$\begin{aligned} &\lambda_i\left(\left(I + V_2\left(V_2^T \widehat{A} V_2\right)^{-1} V_2^T\right) A\right) \\ &\quad \ge \lambda_i\left(\left(I + V_1\left(V_1^T \widehat{A} V_1\right)^{-1} V_1^T\right) A\right). \end{aligned} \tag{237}$$

Proof. It is readily seen that the proposition holds if $F = V_2(V_2^T \widehat{A} V_2)^{-1} V_2^T - V_1(V_1^T \widehat{A} V_1)^{-1} V_1^T$ is negative definite. But since $\operatorname{Im} V_1 \subseteq \operatorname{Im} V_2$, there exists some matrix Q of order $m_2 \times m_1$ such that $V_1 = V_2 Q$. Then, with $D_K = V_k^T \widehat{A} V_k$, we have

$$\begin{aligned} F &= V_2 \left(D_2^{-1} - Q D_1^{-1} Q^T\right) V_2^T \\ &= V_2 D_2^{-1/2} \left(I - D_2^{-1/2} Q D_1^{-1/2} Q^T D_2^{-1/2}\right) D_2^{-1/2} V_2^T, \end{aligned} \tag{238}$$

where

$$P \equiv D_2^{-1/2} Q D_1^{-1/2} Q^T D_2^{-1/2} = D_2^{-1/2} Q \left(Q^T D_2 Q\right)^{-1} Q^T D_2^{-1/2}, \tag{239}$$

is an orthogonal projector ($P^2 = P$), whose eigenvalues are 0 and 1. □

Corollary 3. *If $\operatorname{Im} V_1 = \operatorname{Im} V_2$ then $I + V_2 D_2^{-1} V_2^T = I + V_1 D_1^{-1} V_1^T$.*

Proof. In this case, Q in $V_1 = V_2 Q$ is invertible. Thus, $D_2^{-1/2} Q (Q^T D_2 Q)^{-1} Q^T D_2^{-1/2} = I$. □

Remark 4. The above corollary shows that the individual eigenvectors of A are not needed when constructing the matrix V; we are rather interested in the subspace spanned by them.

The most interesting case for us is when $\operatorname{Im} V \supset \operatorname{span}\{\mathbf{v}_1, \ldots, \mathbf{v}_m\}$, where v_i are the eigenvalues of A for the smallest eigenvalues $\lambda_1, \ldots, \lambda_m$. Then, the preconditioner $B = I + \sigma V A_V^{-1} V^T$ moves the smallest eigenvalues λ_i at least to $\lambda_i + \sigma$, where σ is a scaling parameter.

Theorem 14. *Let $B = I + \sigma V A_V^{-1} V^T$ and let $\lambda_1, \dots, \lambda_m$ be the smallest eigenvalues of A. Then, for the eigenvalues of BA there holds.*

$$\widetilde{\lambda}_i \geq \begin{cases} \lambda_i + \sigma, & 1 \leq i \leq m \\ \lambda_i, & m+1 \leq i \leq n. \end{cases} \tag{240}$$

Proof. Let $V_1 = [\mathbf{v}_1, \dots, \mathbf{v}_m]$, where $\{\mathbf{v}_i\}_1^m$ are the first m eigenvalues of A and let $V_2 = V$. Then, Lemma 7 and (234) show the result. □

It may happen that the eigenvalues are moved too far so that the maximum eigenvalue of BA is much larger that of A.

Theorem 15. *Let A be s.p.d. of order $n \times n$ and let the rectangular matrix V of order $n \times m$ be defined as $V = [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_m]$. Assume that* $\operatorname{rank} V = m$. *Further, define $\widetilde{A} = (I + \sigma V A_V^{-1} V^T)A$, where $A_V = V^T A V$. Then,*

$$\lambda_{\max}\left(\widetilde{A}\right) \leq \sigma + \lambda_{\max}(A). \tag{241}$$

Proof. The result follows from the following relations:

$$\begin{aligned} \lambda_{\max}\left(\widetilde{A}\right) &\leq \lambda_{\max}(A) + \sigma \sup \frac{\mathbf{x}^T V \left(V^T A_V V\right)^{-1} V^T \mathbf{x}}{\mathbf{x}^T A^{-1} \mathbf{x}} \\ &= \lambda_{\max}(A) + \sigma \sup \frac{\mathbf{x}^T A^{1/2} V \left(V^T A V\right)^{-1} V^T A^{1/2} \mathbf{x}}{\mathbf{x}^T \mathbf{x}} \\ &= \lambda_{\max}(A) + \sigma, \end{aligned} \tag{242}$$

where the last equality follows from Lemma 7 □

It follows from Theorems 6.3 and 16 that the optimal value of $\sigma = \lambda_{m+1}$, in which case $\kappa(BA) \leq (\lambda_n + \sigma)/\lambda_{m+1}$. In general, λ_{m+1} may not be known. With $\sigma = \lambda_n$, we obtain $\kappa(BA) \leq (2\lambda_n)/\lambda_{m+1}$.

The moral we can draw from the above is that the suggested technique can be a very useful means to reduce the condition number of a given matrix A if we have information about the eigenvectors corresponding to the smallest eigenvalues of A. Since in practice this is hardly ever the case, a natural step to undertake is to consider not the individual eigenvectors but the subspace spanned by some approximation of them.

In the next section, we present a generalized form of the augmented matrix preconditioner which allows for both approximate subspaces and the replacement of A_V by a simpler matrix B_V.

7. Preconditioners with an Approximate Subspace Correction Term

The preconditioner presented in the previous subsection will now be extended to include an approximate subspace correction term.

We replace first A_V with a possibly simpler matrix B_V. The resulting eigenvalue bounds are found in the next theorem.

Theorem 16. *Let A be s.p.d. Define the preconditioner B as $B = I + \sigma V B_V^{-1} V^T$,where,$\sigma > 0$,* $\operatorname{Im} V \supseteq \operatorname{span} \{v_1, \dots, v_m\}$, *where v_i are the eigenvectors of A for the smallest eigenvalues and B_V is an $m \times m$ s.p.d. approximation of A_V. Then, the eigenvalues $\lambda(BA)$ of BA are bounded as follows:*

(a)

$$\begin{aligned} &\min\{\sigma\lambda_{\min}(B_V^{-1} A_V) + \lambda_1, \lambda_{m+1}\} \\ &\quad \leq \lambda(BA) \leq \sigma\lambda_{\max}(B_V^{-1} A_V) + \lambda_{\max}(A). \end{aligned} \tag{243}$$

(b) *With $\sigma = \lambda_{\max}(A)/\lambda_{\max}(B_V^{-1} A_V)$, we have*

$$\min\{\lambda_{\max}(A)/\kappa(B_V^{-1} A_V) + \lambda_1, \lambda_{m+1}\} \leq \lambda(BA) \leq 2\lambda_{\max}(A). \tag{244}$$

Proof. The minimal eigenvalue of BA can be estimated as

$$\begin{aligned} \lambda_{\min}(BA) &= \inf_x \left\{ \frac{\mathbf{x}^T (I + \sigma V B_V^{-1} V^T) \mathbf{x}}{\mathbf{x}^T A^{-1} \mathbf{x}} \right\} \\ &= \inf_x \left\{ \frac{\mathbf{x}^T \mathbf{x}}{\mathbf{x}^T A^{-1} \mathbf{x}} + \sigma \frac{\mathbf{x}^T V B_V^{-1} V^T \mathbf{x}}{\mathbf{x}^T V A_V^{-1} V^T \mathbf{x}} \cdot \frac{\mathbf{x}^T V A_V^{-1} V^T \mathbf{x}}{\mathbf{x}^T A^{-1} \mathbf{x}} \right\} \\ &= \inf_x \left\{ \frac{\mathbf{x}^T \mathbf{x}}{\mathbf{x}^T A^{-1} \mathbf{x}} + \sigma\lambda_{\min}(B_V^{-1} A V) \frac{\mathbf{x}^T V A_V^{-1} V^T \mathbf{x}}{\mathbf{x}^T A^{-1} \mathbf{x}} \right\}. \end{aligned} \tag{245}$$

Here, $\inf_x \mathbf{x}^T \mathbf{x}/\mathbf{x}^T A^{-1} \mathbf{x} = \lambda_1$ and $\inf_{x; V^T x = 0} \mathbf{x}^T \mathbf{x}/\mathbf{x}^T A^{-1} \mathbf{x} = \lambda_{m+1}$.

The lower bound equals the minimal eigenvalue of the matrix $\widehat{B}A$, where

$$\begin{aligned} \widehat{B} &= \widehat{B}(V) \equiv I + \widehat{\sigma} V\ A_V^{-1} V^T \\ \widehat{\sigma} &= \sigma\lambda_{\min}(N_V^{-1} A_V). \end{aligned} \tag{246}$$

By Theorem 14, we have with $V_1 =$ span $\{v_1, \dots, v+m\}$ and V_2 a matrix satisfying $\operatorname{Im} V_2 \supseteq \operatorname{Im} V_1$, that

$$\lambda_i\left(\widehat{B}(V_2)A\right) \geq \lambda_i\left(\widehat{B}(V_1)A\right) \geq \min\{\lambda_1 + \widehat{\sigma}, \lambda_{m+1}\} \tag{247}$$

and, in particular, for $V_2 = V$

$$\begin{aligned} \lambda_{\min}(BA) &\geq \min(\lambda_1 + \widehat{\sigma}, \lambda_{m+1}) \\ &= \min\{\lambda_1 + \sigma\lambda_{\min}(B_V^{-1} A_V), \lambda_{m+1}\} \end{aligned} \tag{248}$$

which is the lower bound in part (a). The upper bound follows in a similar way. Since there is no upper bound assumed on rank V, and since $\sup_{x, V^T x = 0} \mathbf{x}^T \mathbf{x}/\mathbf{x}^T A^{-1} \mathbf{x} \leq \lambda_{\max}(A)$, we obtain

$$\lambda_{\max}(BA) \leq \sigma\lambda_{\max}(B_V^{-1} A V) + \lambda_{\max}(A). \tag{249}$$

If we let $\sigma = \lambda_{\max}(A)/\lambda_{\max}(B_V^{-1} A_V)$, we get the stated lower bound in (b) and $\lambda_{\max}(BA) \leq 2\lambda_{\max}(A)$. □

Since normally $\lambda_{\max}(A)$ and $\lambda_{\max}(B_V^{-1}A_V)$ are readily estimated, the given choice of σ is viable. It may increase the maximal eigenvalue with a factor 2, which is acceptable.

Corollary 4. *If $\kappa(B_V^{-1}A_V) \le \lambda_{\max}(A)\lambda_{m+1}$, then*

$$\kappa(BA) \le \frac{2\lambda_{\max}(A)}{\lambda_{m+1}}, \tag{250}$$

that is, the upper bound, coincides with the bound where $B_V = A_V$. In particular, if $\kappa(A_V) \le \lambda_{\max}(A)/\lambda_{m+1}$, then one can simply let $B_V = I$ or

$$B_V = \operatorname{diag}(A_V). \tag{251}$$

Hence, seen that the matrix A_V in the preconditioner can be replaced with a simpler matrix B_V where the action of B_V^{-1} is cheap, without deteriorating the eigenvalue bounds.

It still remains to weaken the assumption

$$\operatorname{Im} V \supseteq \operatorname{span}\{v_1, \dots v_m\}, \tag{252}$$

as this is not easy to satisfy in practice. Due to space limitations this will not be presented here. Instead, refer to [56].

8. Krylov Subspace Methods for Singular Systems

Singular systems, that is, with a nontrivial kernel, arise in various applications, such as in boundary value problems with pure Neumann type boundary conditions imposed on the whole boundary. Nullspaces of large dimension may arise in finite element methods using edge element methods, see, for example, [104] and in the analysis of Markov chains when stationary probability vectors of stochastic matrices are computed, see [105, 106], see also [107].

A singular system does not always have a solution and it is more appropriate to consider the least squares problem: find $\mathbf{x} \in R^n$ such that $\|\mathbf{b} - A\mathbf{x}\| \le \|\mathbf{b} - A\mathbf{x}\|$ for all $x \in R^n$. We recall that a basic iterative solution method to solve a linear system, either has the form

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \tau_k \mathbf{r}^k, \qquad \mathbf{r}^k = \mathbf{b} - A\mathbf{x}^k, \quad k = 0, 1, \dots, \tag{253}$$

or the preconditioned form

$$\text{solve } B\delta^k = \tau_k \mathbf{r}^k, \tag{254}$$

and update

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \delta^k, \quad k = 0, 1, \dots . \tag{255}$$

Here, B is a preconditioner to A. Similarly, as we have seen, more involved methods, such as (generalized) CG-methods (GCG, GMRES, GCR, etc.) are based on approximations taken from a Krylov subspace

$$K(A, \mathbf{r}^0, k) = \left\{\mathbf{r}^0, A\mathbf{r}^0, \dots, A^k\mathbf{r}^0\right\} \quad \text{or} \quad K(B^{-1}A, \mathbf{r}^0, \text{k}). \tag{256}$$

In general, they are based on a minimum residual approach where, at each iteration step, we compute an updated solution that satisfies the best approximation property

$$\min_{x \in K(A, \mathbf{r}^0, k)} \|\mathbf{b} - A\mathbf{x}\| \quad \text{or} \quad \min_{x \in K(B^{-1}A, \mathbf{r}^0, k)} \|\mathbf{b} - A\mathbf{x}\|. \tag{257}$$

We will show that the convergence of such iterative solution methods can stall, or suffer a breakdown, when applied to certain singular systems. For the analysis, we will use the following properties of relevant subspaces for matrix A. In this generality, they can be stated for rectangular matrices.

Definition 2. Let $A \in \mathcal{R}^{m\times n}$. Then $R(A)$ of dimension $\le n$, called the *range* of A, is the subspace spanned by the column vectors $\mathbf{a}_{.j}$ that is

$$\mathbf{y} \in \mathcal{R}(A) \Longleftrightarrow \mathbf{y} = \sum_{j=1}^{n} \alpha_j \mathbf{a}_{.j}. \tag{258}$$

Definition 3. $\mathcal{N}(A)$, of dimension $\le n$, is the *nullspace* of $\mathbf{A}$, that is, the subspace of vectors $\underline{v} \in R^n$ s.t. $A\underline{v} = \underline{0}$.

By a classical result (see e.g., [6]), it holds $R(A)^\perp = \mathcal{N}(A^T)$

Definition 4. A linear system $A\mathbf{x} = \mathbf{b}$ is called *consistent* if $b \in R(A)$ and *inconsistent* otherwise. If $A\mathbf{x} = \mathbf{b}$ is consistent, there exists a solution. Cle arly, any system with a nonsingular matrix A is consistent.

To provide a general method, applicable for all types of systems, we will use a least squares type of method, that is determine an approximate solution to $\mathbf{x}$ s.t. $\min_{\mathbf{x}}\|\mathbf{b} - A\mathbf{x}\|$ (or, similarly, a preconditioned form).

In practice, the approximations are mostly computed by a Krylov subspace method, where at each step a solution $\mathbf{x}^k$ is computed such that

$$\left\|\mathbf{b} - A\mathbf{x}^k\right\| \le \|\mathbf{b} - A\mathbf{x}\|, \quad \forall x \in \mathcal{K}(A, \mathbf{r}^0, k), \tag{259}$$

or a preconditioned form of the method. As the next Theorem shows even if the system is consistent, a breakdown can occur.

Theorem 17. *Any minimum residual Krylov subspace method may suffer a breakdown for some initial vector if and only if $R(A) \cap \mathcal{N}(A)$ contains a nontrivial vector.*

Proof. The sufficiency follows since if $R(A) \cap \mathcal{N}(A) \ne \{\underline{0}\}$, there exists a nonzero vector $\mathbf{x} \in \mathcal{N}(A)$ which is also in $R(A)$. Then, at some stage k, there exists a vector $\mathbf{r}^k = \mathbf{y}$, where $A\mathbf{y} = \mathbf{x}$, $\mathbf{x} \in \mathcal{N}(A)$. Hence, $A\mathbf{r}^k = A\mathbf{y} = \mathbf{x}$, which implies $A^2\mathbf{r}^k = A\mathbf{x} = 0$, so a zero vector arises in the Krylov subspace at some stage. This means that convergence stalls. On the other hand, $R(A) \cap \mathcal{N}(A) = \{\underline{0}\}$ implies the existence of nonzero vectors $A^k r_0$ of any order in the Krylov subspace, which implies that there is an improved approximate solution for each higher stage $k + 1$. □

Example 8. Let

$$A = \begin{bmatrix} 0 & 1 & -1 & 0 \\ -1 & 2 & -1 & 0 \\ 1/2 & 1/2 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \qquad (260)$$

Here, $A\underline{\mathbf{e}} = 0$, $\underline{\mathbf{e}}^T = (1, 1, , 1, 0)$, that is, $\underline{\mathbf{e}} \in \mathcal{N}(A)$.

Furthermore, there is a solution of $A\mathbf{y} = \underline{\mathbf{e}}$, for instance

$$\mathbf{y} = 1/2 \begin{bmatrix} 3 \\ 3 \\ 1 \\ 0 \end{bmatrix}. \qquad (261)$$

Hence, $\underline{\mathbf{e}} \in R(A) \cap \mathcal{N}(A)$, where $\underline{\mathbf{e}} \neq \underline{0}$. Since $A^2\mathbf{r}^0 = 0$, with $\mathbf{y}$ as initial vector, the Krylov subspace for the system $A\mathbf{x} = 0$ stalls at the second step.

Corollary 5. *(a) If $\mathcal{N}(A) = \mathcal{N}(A^T)$, in particular if A is symmetric, then minimal residual type methods-based the Krylov subspace converges.*

(b) More generally, this holds if A is a $(H-)$ normal matrix.

Proof. (a) Since $R(A)^{\perp} = \mathcal{N}(A^T)$, it holds

$$\begin{gathered} R(A)^{\perp} = \mathcal{N}(A) \quad \text{and, hence} \\ R(A) \cap \mathcal{N}(A) = R(A) \cap R(A)^{\perp} = \{\underline{0}\}. \end{gathered} \qquad (262)$$

(b) For a normal matrix, there exists a unitary matrix U that diagonalizes A, that is

$$U^T A U = D. \qquad (263)$$

Hence,

$$U^T A^T U = D^T = D. \qquad (264)$$

Therefore, if $AU\mathbf{x} = D\mathbf{x} = 0$ for some $\mathbf{x} \neq 0$, then also $A^T U\mathbf{x} = D\mathbf{x} = 0$, so

$$U\mathbf{x} \in \mathcal{N}(A), \quad U\mathbf{x} \in \mathcal{N}\left(A^T\right), \qquad (265)$$

for any such vector $\mathbf{x}$. Since U is nonsingular, this implies $\mathcal{N}(A) = \mathcal{N}(A^T)$. □

Remark 5. Corollary 5 can be extended to H-normal matrices, that is matrices for which A commutes with its H-adjoint,

$$A' = H^{-1} A^* H, \qquad (266)$$

for some Hermitian matrix H see, for example, [6].

A remedy to avoid breakdowns for matrices A for which the vector space $\mathcal{V} = R(A) \cap \mathcal{N}(A)$ is nontrivial, is to work in a subspace orthogonal to $\mathcal{V}$. This can be achieved by use of the augmented subspace projection method in Section 6. This method works also to avoid situations, where $R(A) \cap \mathcal{N}(A)$ contains eigenvectors to A corresponding to nearly zero eigenvalues, causing a near breakdown or, in finite precision computations, an actual breakdown. For further comments on near breakdowns, see, for example, [83–85].

9. Concluding Remarks

Some milestones in the development iterative solutions methods have been presented. By the combination of improved methods and the developments of computer hardware one can presently solve problems with a degree of freeadoms nearly billionfold compared to that in the early ages of the computer age.

There remains still, however, very difficult problems such as in multiphysics and heterogeneous media problems and various forms of inverse problems, which need further improvement of solution methods.

Some problems, such as those arising in constrained optimization and mixed finite element methods, lead to matrices on saddle pointform. Due to space limitations, they have not been discussed in this paper, however, see, for example, [108]. In the last centuries, much work has been devoted to multigrid, algebraic multigrid and multilevel iteration methods which have shown an optimal order of performance for many types of problems, for example see [58]. Also, domain decomposition methods which go back to the Schwarz alternating decomposition method, have shown developments, see, for example, [109–112]. For an early survey of domain decomposition methods, see [113]. For the same reason, they could not be discussed in this paper. Much work has also been developed to parallelization aspects of solution methods. This topic deserves a separate survey article and has also not been discussed in this paper.

References

[1] L. F. Richardson, "The approximate arithmetical solution by finite differences of physical problems involving differential equations, with an application to the stresses in a masonry dam," *Philosophical Transactions of the Royal Society of London*, vol. 210, pp. 307–357, 1911.

[2] L. F. Richardson, "How to solve differential equations approximately by arithmetic," *Mathematical Gazette*, vol. 12, pp. 415–421, 1925.

[3] R. V. Southwell, *Relaxation Methods in Engineering Science*, Oxford University Press, Oxford, UK, 1940.

[4] D. M. Young, "On Richardson's method for solving linear systems with positive definite matrices," *Journal of Mathematical Physics*, vol. 32, pp. 243–255, 1954.

[5] C. F. Gauss, "Brief an Gerling vom 26 Dec.1823, Werke," vol. 9, pp. 278–281, A translation by G.E. Forsythe, in MTAC vol. 5 255–258, 1950.

[6] O. Axelsson, *Iterative Solution Methods*, Cambridge University Press, New York, NY, USA, 1994.

[7] E. G. D'Yakonov, "On the iterative method for the solution of finite difference equations," *Doklady Akademii Nauk SSSR*, vol. 138, pp. 522–525, 1961.

[8] J. E. Gunn, "The solution of elliptic difference equations by semi-explicit iterative techniques," *SIAM Journal on Numerical Analysis*, pp. 24–45, 1964.

[9] R. E. Bank, "An automatic scaling procedure for a D'Yakanov-Gunn iteration scheme," *Linear Algebra and Its Applications*, vol. 28, pp. 17–33, 1979.

[10] R. S. Varga, *Matrix Iterative Analysis*, Prentice Hall, Englewood Cliffs, NJ, USA, 1962.

[11] J. H. Ortega and W. C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, Orlando, Fla, USA, 1970.

[12] R. Beauwens, "Factorization iterative methods, M-operators and H-operators," *Numerische Mathematik*, vol. 31, no. 4, pp. 335–357, 1978.

[13] L. Seidel, "Ueber ein verfahren, die gleichungen, auf welche die methode der kleinsten quadrate führt, sowie lineäre gleichungen überhaupt, durch successive annäherung aufzulösen," *Abhandlungen der Bayerischen Akademie der Wissenschaften. Mathematisch-Naturwissenschaftliche Abteilung*, vol. 11, pp. 81–108, 1814.

[14] S. P. Frankel, "Convergence rates of iterative treatments of partial differential equations," *Mathematical Tables and Other Aids to Computation*, vol. 4, pp. 65–75, 1950.

[15] D. M. Young, *Iterative methods for solving partial difference equations of elliptic type*, Doctoral Thesis, Harward University, 1950, Cambridge, MA.

[16] D. M. Young, *Iterative Solution of Large Systems*, Academic Press, Orlando, Fla, USA, 1971.

[17] O. Axelsson, H. Lu, and B. Polman, "On the numerical radius of matrices and its application to iterative solution methods," *Linear and Multilinear Algebra*, vol. 37, pp. 225–238, 1994.

[18] O. Axelsson, "Solution of linear systems of equations: iterative methods," in *Sparse Matrix Techniques*, V. A. Barker, Ed., LNM no. 572, pp. 1–51, Springer, Berlin, Germany, 1977.

[19] G. H. Golub and R. S. Varga, "Chebyshev semi-iterative methods, successive overrelaxation iterative methods, and second-order Richardson Iterative Methods—part I and II," *Numerische Mathematik*, vol. 3, pp. 147–168, 1961.

[20] D. M. Young, "Second degree iterative methods for the solution of large linear systems," *Journal of Approximation Theory*, vol. 5, pp. 137–148, 1972.

[21] R. Freund, "On conjugate gradient type methods and polynomial preconditioners for a class of complex non-Hermitian matrices," *Numerische Mathematik*, vol. 57, pp. 285–312, 1990.

[22] R. Freund, "Conjugate gradient-type methods for linear systems with complex symmetric matrices," *SIAM Journal on Scientific and Statistical Computing*, vol. 13, pp. 425–448, 1992.

[23] B. Fischer and R. Freund, "Chebyshev polynomials are not always optimal," *Journal of Approximation Theory*, vol. 65, no. 3, pp. 261–272, 1991.

[24] T. A. Manteuffel, "The Tchebychev iteration for nonsymmetric linear systems," *Numerische Mathematik*, vol. 28, no. 3, pp. 307–327, 1977.

[25] M. R. Hestenes and E. Stiefel, "Methods of conjugate gradients for solving linear systems," *Journal of Research of the National Bureau of Standards. Section B*, vol. 49, pp. 409–436, 1952.

[26] G. H. Golub and D. P. O'Leary, "Some history of the conjugate gradient and Lanczos alghorithms: 1948–1976," *SIAM Review*, vol. 31, pp. 50–102, 1989.

[27] J. K. Reid, "The use of conjugate gradients for systems of linear equations posessing "Property A"," *SIAM Journal on Numerical Analysis*, vol. 9, pp. 325–332, 1972.

[28] P. Concus, G. H. Golub, and D. P. O'Leary, "A generalized conjugate gradient method for the numerical solution of elliptic partial differential equations," in *Sparse Matrix Computations*, J. R. Bunch and D. J. Rose, Eds., pp. 309–332, Academic Press, New York, NY, USA, 1976.

[29] O. Axelsson, "Optimal preconditioners based on rate of convergence estimates for the conjugate gradient method," in *Lecture Notes of IMAMM '99*, S. Mika and M. Brandner, Eds., pp. 5–56, University of West Bohemia in Pilsen, 1999.

[30] O. Axelsson and V. A. Barker, *Finite Element Solution of Boundary Value Problems. Theory and Computations*, Academic Press, Amsterdam, The Netherlands, 1984.

[31] O. Axelsson, "Condition numbers for the study of the rate of convergence of the conjugate gradient method," in *Iterative Methods in Linear Algebra II*, S. Margenov and P. S. Vassilevski, Eds., pp. 3–33, IMACS, NJ, USA, 1999.

[32] O. Nevanlinna, *Convergence of Iterations for Linear Equations*, ETH Zürich, Basel, Switzerland, 1993.

[33] L. Yu. Kolotilina, *Lecture Notes in Mathematics*, vol. 1457, Springer, Berlin, Germany, 1989.

[34] Y. Saad, *Iterative Methods for Sparse Linear Systems*, PWS, NJ, USA, 1996.

[35] O. Axelsson and V. A. Barker, "Finite element solutions of boundary value problems," in *Theory and Computation*, SIAM Classics, Applied Mathematics, Philadelphia, USA, 2001.

[36] O. Axelsson and I. Kaporin, "On the sublinear and superlinear rate of convergence of conjugate gradient methods," *Numerical Algorithms*, vol. 25, no. 1–4, pp. 1–22, 2000.

[37] I. E. Kaporin, "New convergence results and preconditioning strategies for the conjugate gradient method," *Numerical Linear Algebra with Applications*, vol. 1, pp. 179–210, 1994.

[38] L. Y. Kolotilina and A. Y. Yeremin, "Sparse approximate inverse preconditionings I. Theory," *SIAM Journal on Matrix Analysis and Applications*, vol. 14, pp. 45–58, 1993.

[39] O. Axelsson and J. Karátson, "Superlinearly convergent CG methods via equivalent preconditioning for nonsymmetric elliptic operators," *Numerische Mathematik*, vol. 99, no. 2, pp. 197–223, 2004.

[40] Y. Saad and M. H. Schultz, "GMRES: a generalized minimum residual algorithm for solving nonsymmetric linear systems," *SIAM Journal on Scientific and Statistical Computing*, pp. 856–869, 1986.

[41] O. Axelsson and M. Nikolova, "Conjugate gradient minimum residual method (GCG- MR) with variable preconditioners and a relation between residuals of the GCG-MR and GCG-OR methods," *Communications in Mathematical Analysis*, vol. 1, pp. 371–388, 1997.

[42] A. Greenbaum, "Comparison of splittings used with the conjugate gradient algorithm," *Numerische Mathematik*, vol. 33, no. 2, pp. 181–193, 1979.

[43] O. Axelsson, "Stabilization of algebraic multilevel iteration methods; additive methods," *Numerical Algorithms*, vol. 21, no. 1–4, pp. 23–47, 1999.

[44] Y. Saad, "A flexible inner-outer preconditioned GMRES algorithm," *SIAM Journal on Scientific Computing*, vol. 14, pp. 461–469, 1993.

[45] V. Simoncini and D. B. Szyld, "Flexible inner-outer Krylov subspace methods," *SIAM Journal on Numerical Analysis*, vol. 40, no. 6, pp. 2219–2239, 2002.

[46] J. A. Meijerink and H. A. van der Vorst, "An iterative solution method for linear systems of which the coefficient matrix is a symmetric M-matrix," *Mathematics of Computation*, vol. 31, pp. 148–162, 1979.

[47] I. Gustafsson, "A class of first order factorization methods," *BIT*, vol. 18, no. 2, pp. 142–156, 1978.

[48] O. Axelsson and G. Lindskog, "On the rate of convergence of the preconditioned conjugate gradient method," *Numerische Mathematik*, vol. 48, no. 5, pp. 499–523, 1986.

[49] O. Axelsson, S. Brinkkemper, and V. P. Il'in, "On some versions of incomplete block-matrix factorization iterative

methods," *Linear Algebra and Its Applications*, vol. 58, pp. 3–15, 1984.

[50] P. Concus, G. H. Golub, and G. Meurant, "Block preconditioning for the conjugate gradient method," *Statistics and Computing*, vol. 6, pp. 220–252, 1985.

[51] O. Axelsson and B. Polman, "On approximate factorization methods for block matrices suitable for vector and parallel processors," *Linear Algebra and Its Applications*, vol. 77, pp. 3–26, 1986.

[52] R. Beauwens and M. Ben Bouzid, "On sparse block factorization, iterative methods," *SIAM Journal on Numerical Analysis*, vol. 24, pp. 1066–1076, 1987.

[53] J. Kraus, "Algebraic multilevel preconditioning of finite element matrices using local Schur complements," *Numerical linear Algebra with Applications*, vol. 13, no. 1, pp. 49–70, 2006.

[54] O. Axelsson, R. Blaheta, and M. Neytcheva, "Preconditioning of boundary value problems using elementwise Schur complements," *SIAM Journal on Matrix Analysis and Applications*, vol. 31, no. 2, pp. 767–789, 2009.

[55] M. Neytcheva, "On element-by-element Schur complement approximations," *Linear Algebra and its Applications*, 2010, In press.

[56] O. Axelsson and A. Padiy, "On the additive version of the algebraic multilevel iteration method for anisotropic elliptic problems," *SIAM Journal of Scientific Computing*, vol. 20, no. 5, pp. 1807–1830, 1999.

[57] S. McCormick, "Multilevel adaptive methods for partial differential equations," in *Frontiers in Applied Mathematics*, vol. 6, SIAM, Philadelphia, Pa, USA, 1989.

[58] P. S. Vassilevski, "Multilevel block factorization preconditioners," in *Matrix-Based Analysis and Algorithms for Solving Finite Element Equations*, Springer, New York, NY, USA, 2008.

[59] O. Axelsson, "A generalized SSOR method," *BIT*, vol. 12, no. 4, pp. 443–467, 1972.

[60] V. P. Il'in, "Incomplete factorization methods," *Soviet Journal of Numerical Analysis and Mathematical Modelling*, vol. 3, pp. 179–198, 1988.

[61] O. Axelsson, "A survey of preconditioned iterative methods for linear systems of algebraic equations," *BIT*, vol. 25, no. 1, pp. 166–187, 1985.

[62] D. W. Peaceman and H. H. Rachford Jr., "The numerical solution of parabolic and elliptic differential equations," *Journal of the Society for Industrial and Applied Mathematics*, vol. 3, pp. 28–41, 1955.

[63] G. Birkhoff, R. S. Varga, and D. M. Young, "Alternating direction implicit methods," in *Advances in Computers*, F. Alt and M. Rubinoff, Eds., vol. 3, pp. 189–273, 1962.

[64] E. Wachspress, *Iterative Solution of Elliptic Systems and Applications to the Neutron Diffussion Equations of Reactor Physics*, Englewood Cliffs, NY, USA, Prentice Hall, 1966.

[65] O. Axelsson, "A generalized conjugate gradient, least square method," *Numerische Mathematik*, vol. 51, no. 2, pp. 209–227, 1987.

[66] O. Axelsson and A. Kucherov, "Real valued iterative methods for solving complex symmetric linear systems," *Numerical Linear Algebra with Applications*, vol. 7, no. 4, pp. 197–218, 2000.

[67] O. Axelsson, "On the efficiency of a class of A-stable methods," *BIT*, vol. 14, no. 3, pp. 279–287, 1974.

[68] N. I. Buleev, "A numerical method for the solution of two-dimensional and three-dimensional equations of diffussion," *Mathematics Sbornik*, vol. 51, pp. 227–238, 1960.

[69] T. A. Oliphant, "An extrapolation process for solving linear systems," *Quarterly of Applied Mathematics*, vol. 20, pp. 257–267, 1962.

[70] T. Dupont, R. P. Kendall, and H. H. Rachford Jr., "An approximate factorization procedure for solving self-adjoint elliptic difference equations," *SIAM Journal on Numerical Analysis*, vol. 5, pp. 554–573, 1968.

[71] T. Dupont, "A factorization procedure for the solving of elliptic difference equations," *SIAM Journal on Numerical Analysis*, pp. 753–782, 1968.

[72] Z. Woźnički, "AGA two-sweep iterative methods and their application in critical reactor calculations," *Nukleonika*, vol. 23, pp. 941–968, 1978.

[73] H. S. Stone, "Iterative solution of implicit approximations of multidimensional partial differential equations," *SIAM Journal on Numerical Analysis*, vol. 5, pp. 530–558, 1968.

[74] R. R. Underwood, "An approximate factorization procedure based on the block Cholesky decomposition and its use with the conjugate gradient method," Report NRDO-11386, Nuclear Energy Division, General Electric Co., San Jose, Calif, USA, 1976.

[75] R. S. Varga, E. B. Saff, and V. Mehrman, "Incomplete factorizations of matrices and connections with H-matrices," *SIAM Journal on Numerical Analysis*, vol. 17, pp. 787–793, 1980.

[76] O. Axelsson, "A general incomplete block-matrix factorization method," *Linear Algebra and Its Applications*, vol. 74, pp. 179–190, 1986.

[77] L. Yu. Kolotilina, "On approximate inverses of block H-matrices," in *Numerical Analysis and Mathematical Modelling*, Moscow, Russia, 1989.

[78] O. Axelsson, "Preconditioning methods for block H-matrices," in *Computer Algorithms for Solving Linear Systems*, E. Spedicato, Ed., vol. 77 of *NATO ASI Series*, pp. 169–184, Springer, Berlin, Germany, 1991.

[79] L. Yu. Kolotilina and A. Yu. Yeremin, "On a family of two-level preconditionings of the incomplete block factorization type," *Soviet Journal of Numerical Analysis and Mathematical Modelling*, vol. 1, pp. 292–320, 1986.

[80] K. Takahishi, J. Fagan, and M. S. Chen, "Formation of a sparse bus impedance matrix and its application to short circuit study," in *Proceedings of the 8th Power Industry Computer Application Conference (PICA)*, pp. 63–69, Minneapolis, Minn, USA, 1973.

[81] A. M. Erisman and W. F. Tinney, "On computing certain elements of the inverse of a sparse matrix," *Communications of the ACM*, vol. 18, pp. 177–179, 1975.

[82] Z. Zlatev, *Computational Methods for General Sparse Matrices*, Kluwer Academic Publishers Group, Boston, London, 1991.

[83] A. Greenbaum and Z. Strakos, "Predicting the behaviour of finite precision Lanczos and conjugate gradient computations," *SIAM Journal on Matrix Analysis and Applications*, vol. 13, pp. 121–137, 1992.

[84] H. A. van der Vorst, "The convergence behaviour of preconditioned CG and CG-S in the presence of rounding errors," in *Preconditioned Conjugate Gradient Methods*, vol. 1457 of *Lecture Notes in Mathematics*, pp. 126–136, Springer, Berlin, Germany, 1989.

[85] Y. Notay, "On the convergence rate of the conjugate gradients in presence of rounding errors," *Numerische Mathematik*, vol. 65, pp. 301–317, 1993.

[86] R. Fedorenko, "The spead of convergence of one iterative process," *USSR Computational Mathematics and Mathematical Physics*, vol. 4, pp. 227–235, 1964.

[87] N. S. Bakhvalov, "Numerical solution of a relaxation method with natural constraints on the elliptic operator," *USSR Computational Mathematics and Mathematical Physics*, vol. 6, pp. 101–135, 1966.

[88] A. Brandt, "Multi-level adaptive solution to boundary-value problems," *Mathematics of Computation*, vol. 31, pp. 333–390, 1977.

[89] D. Braess, *Finite Elements: Theory, Fast Solvers, and Applications in Solid Mechanics*, Cambridge University Press, Cambridge, UK, 2001, 2nd edition.

[90] W. Hackbusch, *Multigrid Methods and Applications*, Springer, Berlin, Germany, 1985.

[91] J. H. Bramble, J. E. Pasciak, and J. Xu, "Parallel multilevel preconditioners," *Mathematics of Computation*, vol. 55, pp. 1–22, 1990.

[92] J Mandel, S. McCormick, and J. Ruge, "An algebraic theory for multigrid methods for variational problems," *SIAM Journal on Numerical Analysis*, vol. 25, pp. 91–110, 1988.

[93] H. Bramble, "Multigrid Methods," vol. 294 of *Pitman Research Notes in Mathematics Series*, Longman Scientific and Technical, 1993.

[94] R. E. Bank, T. F. Dupont, and H. Yserentant, "The hierarchical basis multigrid method," *Numerische Mathematik*, vol. 52, no. 4, pp. 427–458, 1988.

[95] O. Axelsson and P. S. Vassilevski, "Algebraic multilevel preconditioning methods I," *Numerische Mathematik*, vol. 56, pp. 157–177, 1989.

[96] O. Axelsson and P. S. Vassilevski, "Algebraic multilevel preconditioning methods II," *Numerische Mathematik*, vol. 27, pp. 1569–1590, 1990.

[97] O. Axelsson and M. Neytcheva, "Algebraic multilevel iteration method for Stieltjes Matrices," *Numerical Linear Algebra with Applications*, pp. 213–236, 1994.

[98] L. Mansfield, "Damped Jacobi preconditioning and coarse grid deflation for conjugate gradient iteration on parallel computers," *SIAM Journal on Scientific Computing*, vol. 12, pp. 1314–1323, 1991.

[99] R. Nicolaides, "Deflation of conjugate gradients with application to boundary value problems," *SIAM Journal on Numerical Analysis*, vol. 24, pp. 355–365, 1987.

[100] Z. Dostal, "Conjugate gradient method with preconditioning by projector," *International Journal of Computer Mathematics*, vol. 23, pp. 315–324, 1988.

[101] O. Axelsson, M. Neytcheva, and B. Polman, "An application of the bordering method to solve nearly singular systems," *Vestnik Moskovskogo Universiteta, Seria 15, Vychisl. Math. Cybern*, vol. 1, pp. 3–25, 1996.

[102] A. Padiy, O. Axelsson, and B. Polman, "Generalized augmented matrix preconditioning approach and its application to iterative solution of ill-conditioned algebraic systems," *SIAM Journal on Matrix Analysis and Applications* , vol. 22, no. 3, pp. 793–819, 2000.

[103] Y. Notay and A. van de Velde, "Coarse grid acceleration of parallel incomplete preconditioners," in *Iterative Methods in Linear Algebra II*, S. Margenov and P. Vassilevski, Eds., vol. 3 of *Computational and Applied Mathematics*, pp. 106–130, IMACS, 1996.

[104] O. Biro, K. Preis, and K. R. Richter, "On the use of the magnetic vector potential in the nodal and edge finite element analysis of 3D magnetostatic problem," *IEEE Transactions on Magnetics*, vol. 32, pp. 651–654, 1996.

[105] K. Wu, N. Nunan, J. W. Crawford, I. M. Young, and K. Ritz, "An efficient Markov chain model for the simulation of heterogeneous soil structure," *Soil Science Society of America Journal*, vol. 68, pp. 346–351, 2004.

[106] K. Tanabe, "Characterization of linear stationary iterative processes for solving a singular system of linear equations," *Numerische Mathematik*, vol. 22, pp. 349–359, 1974.

[107] A. Dax, "The convergence of linear stationary iterative processes for solving singular unstructured systems of linear equations," *SIAM Review*, vol. 32, no. 4, pp. 611–635, 1990.

[108] O. Axelsson and R. Blaheta, "Preconditioning of matrices partitioned in 2 × 2 block form: eigenvalue estimates and Schwarz DD for mixed FEM," *Numerical Linear Algebra with Applications*, vol. 17, pp. 787–810, 2010.

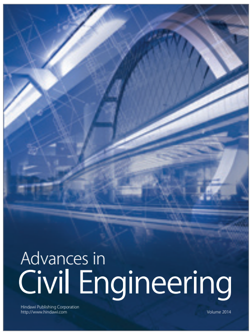
[109] H. A. Schwarz, "Über einen Grenzübergang durch alternierendes verfahren," *Vierteljahrsschrift der Naturforschenden Gesellschaft in Zürich*, vol. 15, pp. 272–286, 1870.

[110] H. A. Schwarz, "Über einige abbildungsaufgaben," *Journal für die Reine und Angewandte Mathematik*, vol. 70, pp. 105–120, 1869.

[111] A. Tosseli and O. B. Widlund, "Domain Decomposition Methods," in *Algorithms and Theory*, Springer, Berlin, Germany, 2005.

[112] R. Blaheta, "Space decomposition preconditioners and parallel solvers," in *Numerical Mathematics and Advanced Applications*, pp. 20–38, Springer, Berlin, Germany, 2004, Proceedings of ENUMATH '03.

[113] B. F. Smith, P. E. Bjorstad, and W. D. Gropp, *Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations*, Cambridge University Press, Cambridge, UK, 1996.



Hindawi

Submit your manuscripts at
http://www.hindawi.com