SIAM J. SCI. COMPUT.
Vol. 22, No. 4, pp. 1444–1460

FLEXIBLE CONJUGATE GRADIENTS*

YVAN NOTAY†

Abstract. We analyze the conjugate gradient (CG) method with preconditioning slightly variable from one iteration to the next. To maintain the optimal convergence properties, we consider a variant proposed by Axelsson that performs an explicit orthogonalization of the search directions vectors. For this method, which we refer to as *flexible* CG, we develop a theoretical analysis that shows that the convergence rate is essentially independent of the variations in the preconditioner as long as the latter are kept sufficiently small. We further discuss the real convergence rate on the basis of some heuristic arguments supported by numerical experiments. Depending on the eigenvalue distribution corresponding to the fixed reference preconditioner, several situations have to be distinguished. In some cases, the convergence is as fast with truncated versions of the algorithm or even with the standard CG method, whereas quite large variations are allowed without too much penalty. In other cases, the flexible variant effectively outperforms the standard method, while the need for truncation limits the size of the variations that can be reasonably allowed.

Key words. iterative methods for linear systems, acceleration of convergence, preconditioning

AMS subject classifications. 65F10, 65B99, 65N20

PII. S1064827599362314

1. Introduction.

As is well known, the conjugate gradient (CG) method combined with a suitable preconditioning is a choice method to solve large sparse $n \times n$ linear systems

$$A\,\mathbf{u} \;=\; \mathbf{b} \tag{1.1}$$

whose coefficient matrix A is symmetric and positive definite (e.g., [4, 9, 19, 21]). Ideally, a preconditioner is a symmetric and positive definite matrix B such that solving a system

$$B\,\mathbf{x} \;=\; \mathbf{y} \tag{1.2}$$

is relatively cheap, whereas the eigenvalues of $B^{-1}A$ are nicely clustered, favoring a rapid convergence.

Now, as preconditioning techniques become more sophisticated and apply to a wider class of problems, it is not unusual that the solution of (1.2) itself requires solving one or more subproblems for which, again, the preconditioned CG method is best suited. Although, from a theoretical point of view, it is more comfortable to assume that the system (1.2) is nevertheless solved accurately, in practice, the overall efficiency of the scheme may then critically depend on the ability to allow loose stopping criteria for the inner iterative solutions. In this case, the preconditioning step seen by the outer process is no longer $\mathbf{x} = B^{-1}\,\mathbf{y}$, but is

$$\mathbf{x} \;=\; \mathcal{B}(\mathbf{y}), \tag{1.3}$$

where $\mathcal{B}$ is a mapping from $\mathcal{R}^n$ to $\mathcal{R}^n$, in general nonlinear.

Early observation has indeed been made by Golub and Overton [17, 18] that the convergence rate of the outer CG process could be maintained even with very

*Received by the editors October 14, 1999; accepted for publication (in revised form) May 10, 2000; published electronically November 2, 2000. This research was supported by the "Fonds National de la Recherche Scientifique," Maître de recherches.

http://www.siam.org/journals/sisc/22-4/36231.html

†Service de Métrologie Nucléaire, Université Libre de Bruxelles (C.P. 165), 50, Av. F.D. Roosevelt, B-1050 Brussels, Belgium (ynotay@ulb.ac.be).

loose accuracy for the inner iterations. Unfortunately, much less is known from a theoretical point of view. An interesting analysis has been recently developed by Golub and Ye [20], but, as will be seen in section 3, the bound derived there tends to strongly overestimate nonlinearities effects. On the other hand, some insight can be gained by reexamining the studies on CG in the presence of errors [22, 23, 31, 26], letting the parameter governing the size of the errors be much larger than the roundoff unit considered in these works. However, only heuristic conclusions can be derived on this basis, and the latter display that the method may lose part of its efficiency when the extremal eigenvalues of $B^{-1}A$ are well separated.

In [4, p. 549], Axelsson proposed a variant of the CG algorithm that is specifically designed to accommodate variable preconditioners (see also Algorithm 5.3 in [8]). This algorithm automatically truncates to standard CG when using a fixed symmetric and positive definite preconditioner, whereas, in the general case, optimal convergence property in A-norm is preserved at the price of performing a full orthogonalization of the search direction vectors (with respect to the $(\cdot, A\,\cdot)$ inner product). Although this method is a particular case of the generalized CG (GCG) algorithm developed in [1, 2, 3, 7], we refer to it as *flexible* CG (FCG).[1]

Similar to any method not based on a short recurrence, FCG has to be combined in practice with a truncation or a restart strategy. In this respect, it is interesting to note that with maximal truncation one recovers the steepest descent algorithm, whereas adding one orthogonality constraint delivers an implementation of the usual CG method, in fact the one observed in [20] to be the most stable with respect to variations in the preconditioner.

Concerning the theoretical analysis, the results in the latter paper prove fast convergence only when the perturbations are below some given threshold, and the analysis fails otherwise. One may then resort to the general results on GCG with variable preconditioning [4, 7], which prove the well definiteness of the method and its convergence under rather weak assumptions on $\mathcal{B}$. However, the proved speed of convergence is clearly much too pessimistic in the case of FCG; see section 3 for details. On the other hand, from a practical point of view, no discussion seems available on which restart or truncation strategy is advisable, and on which benefit can be hoped by using this flexible variant instead of the standard CG algorithm.

In the present paper, we aim at filling these gaps. We first propose in section 2 a mixed truncation–restart strategy that combines the advantages of both pure truncation and pure restart. We next consider in section 3 individual steps of FCG and prove a bound on the decrease of the error between two successive steps that tends to the optimal bound as $\mathcal{B} \to B^{-1}$, while allowing quite large variations in the preconditioner. We further consider in section 4 several successive steps of the untruncated algorithm and show that there exists a matrix $\widehat{B}$ such that $\mathcal{B}(\mathbf{y}) = \widehat{B}\,\mathbf{y}$ for all vectors $\mathbf{y}$ to which $\mathcal{B}$ is effectively applied during these steps. Moreover, when $\mathcal{B}$ is close to some symmetric and positive definite matrix B^{-1}, the spectrum of $\widehat{B}\,A$ is only a slight perturbation of that of $B^{-1}A$. Finally, we discuss in section 5 the effects of truncation and compare FCG with the standard CG algorithm in view of some illustrative numerical examples. Our conclusions are summarized in section 6.

Concerning the use of variable preconditioning in combination with other methods for the outer iterations, we refer the reader to the literature: Chebyshev and Richardson iterations are considered in [16, 17, 18], the Uzawa algorithm is analyzed

[1] *Generalized* CG mostly refers to methods designed to solve unsymmetric systems, whereas the specific variant considered here solely permits the use of variable preconditioners in CG.

in [15], and a flexible variant of GMRES is discussed in [28, 34] (see also [29]).

Notation. Throughout this paper, A and B are $n \times n$ symmetric and positive definite matrices, and $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$ are the eigenvalues of $B^{-1}A$ in increasing order, whereas

$$\kappa \;=\; \frac{\lambda_n}{\lambda_1} \tag{1.4}$$

is the spectral condition number.

For any symmetric and positive definite matrix matrix C, $\|\cdot\|_C$ is the C-norm, that is, the norm associated with the $(\cdot, C\,\cdot)$ inner product: $\|\mathbf{v}\|_C = \sqrt{(\mathbf{v},\, C\,\mathbf{v})}$ for all $\mathbf{v}$. When G is a matrix, $\|G\|_C$ is the matrix norm induced by this vector norm: $\|G\|_C = \max_{\mathbf{z}\neq 0} (\|G\,\mathbf{z}\|_C/\|\mathbf{z}\|_C)$.

2. Algorithm and basic properties. We recall below (Algorithm 2.1) the method referred to as FCG in the introduction, that is, Algorithm 5(a) from [4, p. 549] or Algorithm 5.3 from [8]. As already stated, the main difference with standard CG lies in the explicit orthogonalization of the search direction vectors $\mathbf{d}_i$; more precisely, $\mathbf{d}_i$ is orthogonalized with respect to $(\cdot, A\,\cdot)$ against the m_i previous vectors, where $\{m_i\}_{i=0,1,\ldots}$ is a sequence of truncation parameters. The untruncated version corresponds to $m_i = i$ for all i, whereas the natural choice for m_i would be $m_i = \min(i,\, m_{\max})$ (pure truncation) or sequences like $m_i = 0, 1, \ldots, m_{\max}, 0, 1 \ldots, m_{\max}$, which would mean restarting the (untruncated) algorithm at each $m_{\max}+1$ iteration. However, as we discuss below, one should avoid using m_i less than 1 for any $i > 0$. This observation leads to the mixed truncation–restart strategy advised in Algorithm 2.1, which we generally observed to be more cost efficient than pure truncation (see section 5.3). In Algorithm 2.1 this specific truncation–restart strategy will be referred to as FCG($m_{\max}$).

ALGORITHM 2.1 (FCG).
Flexible Conjugate Gradient

- *Initialization*

$$\begin{aligned} \mathbf{u}_0 &\quad \text{arbitrary} \\ \mathbf{r}_0 &= \mathbf{b} \;-\; A\,\mathbf{u}_0 \end{aligned}$$

- *Iteration* $(i = 0, 1, \ldots)$

$$\begin{aligned} \mathbf{w}_i &= \mathcal{B}(\mathbf{r}_i) \\ \mathbf{d}_i &= \mathbf{w}_i \;-\; \sum_{k=i-m_i}^{i-1} \tfrac{(\mathbf{w}_i,\, A\,\mathbf{d}_k)}{(\mathbf{d}_k,\, A\,\mathbf{d}_k)}\,\mathbf{d}_k \\ \mathbf{u}_{i+1} &= \mathbf{u}_i \;+\; \tfrac{(\mathbf{d}_i,\, \mathbf{r}_i)}{(\mathbf{d}_i,\, A\,\mathbf{d}_i)}\,\mathbf{d}_i \\ \mathbf{r}_{i+1} &= \mathbf{r}_i \;-\; \tfrac{(\mathbf{d}_i,\, \mathbf{r}_i)}{(\mathbf{d}_i,\, A\,\mathbf{d}_i)}\,A\,\mathbf{d}_i \end{aligned}$$

Advised truncation strategy (FCG($m_{\max}$)):
$m_0 = 0;\; m_i = \max\big(1,\, \mathrm{mod}(i,\, m_{\max}{+}1)\big) \quad (i > 0)$
$m_{\max}$: *nonnegative integer.*

Assuming A symmetric and positive definite and $0 \leq m_{i+1} \leq m_i + 1$ for all i, it is easy to prove that (see [4, 7])

$$(\mathbf{d}_k,\, A\,\mathbf{d}_j) = 0 \quad \text{for all } j,\, k \text{ such that } i - m_i \leq j < k \leq i, \tag{2.1}$$

$$(\mathbf{r}_k,\, \mathbf{d}_j) = 0 \quad \text{for all } j,\, k \text{ such that } i - m_i \leq j < k \leq i + 1 \tag{2.2}$$

and that, letting $\mathbf{u} = A^{-1}\,\mathbf{b}$,

$$\|\mathbf{u} - \mathbf{u}_{i+1}\|_A \;=\; \min_{\mathbf{d} \in \mathrm{span}\{\mathbf{d}_{i-m_i},\, \ldots,\, \mathbf{d}_i\}} \|\mathbf{u} - \mathbf{u}_{i-m_i} - \mathbf{d}\|_A. \tag{2.3}$$

If $m_i = i$ (no truncation), these are the optimality properties satisfied by the standard CG method. Moreover, when $\mathcal{B} \equiv B^{-1}$ is a symmetric and positive definite matrix, it can be proved that the recursion defining $\mathbf{d}_i$ automatically truncates because $(\mathbf{w}_i,\, A\,\mathbf{d}_k) = 0$ for all $k < i - 1$. Algorithm 2.1 simplifies then to a particular implementation of the standard CG method, in fact, one of the variants discussed in the early works on CG [24, 27]. This variant is also referred to as inexact preconditioned CG (IPCG) in [20], because there it is observed to be the most stable with respect to variations in the preconditioner. Note, however, that it requires computing one more inner product per iteration than the standard algorithm described in most textbooks (e.g., [4, 9, 19, 21]). In the remainder of this paper, we use "standard CG" to refer to the latter algorithm, whereas the alternative implementation offered by Algorithm 2.1 with $m_{\max} = 1$ will be referred to as FCG(1).

Note that our motivation to propose a strategy in which m_i is never less than 1 originates in this close relation between FCG and CG. Indeed, we then have the guarantee that one will recover the usual CG convergence when $\mathcal{B}$ is a fixed preconditioner, whereas something would irremediably be lost with the pure restart that results when setting $m_i = 0$ for some $i > 0$.

3. Convergence analysis. Equation (2.3) proves that the error measured in the A-norm cannot increase, but it says nothing about the convergence rate. In [4, 7], it is proved that

$$\frac{\|\mathbf{u} - \mathbf{u}_{i+1}\|_A}{\|\mathbf{u} - \mathbf{u}_i\|_A} \;\leq\; \sqrt{1 - \left(\frac{\delta_2}{\delta_1}\right)^2}, \tag{3.1}$$

where

$$\delta_1 = \max_{\mathbf{v} \neq 0} \frac{(\mathbf{v},\, \mathcal{B}(\mathbf{v}))}{(\mathbf{v},\, A^{-1}\,\mathbf{v})}, \qquad \delta_2 = \min_{\mathbf{v} \neq 0} \sqrt{\frac{(\mathcal{B}(\mathbf{v}),\, A\,\mathcal{B}(\mathbf{v}))}{(\mathbf{v},\, A^{-1}\,\mathbf{v})}}.$$

In this analysis, the requirements on $\mathcal{B}$ are thus quite weak, but, unfortunately, the proved speed of convergence is likely to be much too pessimistic. This is easily seen by inspecting the case $\mathcal{B} \equiv B^{-1}$. Then, $\delta_1 = \lambda_n$, $\delta_2 = \lambda_1$, and (3.1) writes

$$\frac{\|\mathbf{u} - \mathbf{u}_{i+1}\|_A}{\|\mathbf{u} - \mathbf{u}_i\|_A} \;\leq\; \sqrt{1 - \kappa^{-2}}, \tag{3.2}$$

whereas the best available bound on the "local" decrease of the error is

$$\frac{\|\mathbf{u} - \mathbf{u}_{i+1}\|_A}{\|\mathbf{u} - \mathbf{u}_i\|_A} \;\leq\; \frac{\kappa - 1}{\kappa + 1}. \tag{3.3}$$

By way of comparison, for $\kappa = 9$, the latter results prove $\|\mathbf{u}-\mathbf{u}_{i+1}\|_A \leq 0.8\,\|\mathbf{u}-\mathbf{u}_i\|_A$, whereas (3.2) yields only $\|\mathbf{u}-\mathbf{u}_{i+1}\|_A \leq 0.994\,\|\mathbf{u}-\mathbf{u}_i\|_A$.

Except for very well conditioned problems, the bound (3.1) is therefore of little help to understand the real effects of the nonlinearities in $\mathcal{B}$. In the following theorem, we also bound the "local" decrease of the error, but with an expression that tends to (3.3) as $\mathcal{B} \to B^{-1}$.

THEOREM 3.1. *Let A, B be $n\times n$ symmetric and positive definite matrices and $\mathcal{B}$ a mapping from $\mathcal{R}^n$ to $\mathcal{R}^n$. Let $\mathbf{b}$, $\mathbf{u}_0$ be vectors of $\mathcal{R}^n$, and let $\{\mathbf{r}_i\}_{i=0,1,\ldots}$, $\{\mathbf{d}_i\}_{i=0,1,\ldots}$, $\{\mathbf{u}_i\}_{i=1,2,\ldots}$ be the sequences of vectors generated by applying Algorithm 2.1 to A, $\mathcal{B}$, $\mathbf{b}$, and $\mathbf{u}_0$ with some given sequence of nonnegative integer parameters $\{m_i\}_{i=0,1,\ldots}$.*

If, for any i,

$$\frac{\|\mathcal{B}(\mathbf{r}_i) - B^{-1}\,\mathbf{r}_i\|_B}{\|B^{-1}\,\mathbf{r}_i\|_B} \;\leq\; \varepsilon_i \;<\; 1, \tag{3.4}$$

then

$$\frac{\|\mathbf{u}-\mathbf{u}_{i+1}\|_A}{\|\mathbf{u}-\mathbf{u}_i\|_A} \;\leq\; \sqrt{1-\frac{4\,\kappa\,(1-\varepsilon_i)^2}{(\kappa+\varepsilon_i^2(\kappa-1)+(1-\varepsilon_i)^2)^2}} \;\leq\; \frac{\kappa\,\frac{1+\varepsilon_i}{1-\varepsilon_i}\,\frac{\left(1+\varepsilon_i^2\right)^2}{1-\varepsilon_i^2}-1}{\kappa\,\frac{1+\varepsilon_i}{1-\varepsilon_i}\,\frac{(1+\varepsilon_i^2)^2}{1-\varepsilon_i^2}+1}. \tag{3.5}$$

Proof. Clearly, by (2.3), for any real α,

$$\|\mathbf{u}-\mathbf{u}_{i+1}\|_A \;\leq\; \|\mathbf{u}-\mathbf{u}_i-\alpha\,\mathcal{B}(\mathbf{r}_i)\|_A \;=\; \|\mathbf{r}_i-\alpha\,A\,\mathcal{B}(\mathbf{r}_i)\|_{A^{-1}}.$$

Then let $\mathbf{t}_i = B^{-1}\,\mathbf{r}_i - \mathcal{B}(\mathbf{r}_i)$. One has

$$(\mathbf{t}_i,\,A\,\mathbf{t}_i) \;=\; \frac{(\mathbf{t}_i,\,A\,\mathbf{t}_i)}{(\mathbf{t}_i,\,B\,\mathbf{t}_i)}\,(\mathbf{t}_i,\,B\,\mathbf{t}_i) \;\leq\; \lambda_n\,\varepsilon_i^2\,(\mathbf{r}_i,\,B^{-1}\,\mathbf{r}_i),$$

whereas

$$|\left((I-\alpha\,A\,B^{-1})\mathbf{r}_i,\,\mathbf{t}_i\right)| \;\leq \|\mathbf{t}_i\|_B\,\|B^{-1}\,(I-\alpha\,A\,B^{-1})\mathbf{r}_i\|_B \;\leq\; \varepsilon_i\,\beta\,(\mathbf{r}_i,\,B^{-1}\,\mathbf{r}_i),$$

where $\beta = \|I-\alpha\,A\,B^{-1}\|_{B^{-1}} = \max(|1-\alpha\,\lambda_1|,\,|1-\alpha\,\lambda_n|)$. Hence, for $\alpha \geq 0$,

$$\begin{aligned}\|\mathbf{r}_i-\alpha\,A\,\mathcal{B}(\mathbf{r}_i)\|_{A^{-1}}^2 &= \|(I-\alpha\,A\,B^{-1})\mathbf{r}_i\|_{A^{-1}}^2 + 2\,\alpha\,\left((I-\alpha\,A\,B^{-1})\mathbf{r}_i,\,\mathbf{t}_i\right) + \alpha^2\,\|\mathbf{t}_i\|_A \\ &\leq \left(\mathbf{r}_i,\,(I-\alpha\,B^{-1}\,A)A^{-1}(I-\alpha\,A\,B^{-1})\,\mathbf{r}_i\right) \\ &\quad + \left(2\,\alpha\,\beta\,\varepsilon_i+\alpha^2\,\lambda_n\,\varepsilon_i^2\right)\;\left(\mathbf{r}_i,\,B^{-1}\,\mathbf{r}_i\right).\end{aligned}$$

Clearly, the right-hand side of the latter inequality cannot be larger than $(1-\eta_i)\|\mathbf{u}-\mathbf{u}_i\|_A^2 = (1-\eta_i)(\mathbf{r}_i,\,A^{-1}\,\mathbf{r}_i)$ if the matrix

$$(1-\eta_i)\,A^{-1} \;-\; (I-\alpha\,B^{-1}\,A)A^{-1}(I-\alpha\,A\,B^{-1}) \;-\; \left(2\,\alpha\,\beta\,\varepsilon_i+\alpha^2\,\lambda_n\,\varepsilon_i^2\right)\,B^{-1}$$

is nonnegative definite. This holds if and only if

$$-\alpha^2\,\lambda^2 \;+\; \left(2\,\alpha-2\,\alpha\,\beta\,\varepsilon_i-\alpha^2\,\lambda_n\,\varepsilon_i^2\right)\,\lambda \;-\; \eta_i \;\geq 0 \quad \text{for all } \lambda \in \sigma\left(B^{-1}A\right). \tag{3.6}$$

Consider now

$$\alpha = 2\,\left(\tfrac{\lambda_n+\varepsilon_i^2(\lambda_n-\lambda_1)}{1-\varepsilon_i}+(1-\varepsilon_i)\,\lambda_1\right)^{-1}.$$

One has $0 < \alpha \leq 2(\lambda_n + \lambda_1)^{-1}$ and therefore $\beta = 1 - \alpha\,\lambda_1$, yielding

$$2\,\alpha - 2\,\alpha\,\beta\,\varepsilon_i - \alpha^2\,\lambda_n\,\varepsilon_i^2 \;=\; \alpha\,(2\,(1-\varepsilon_i) + \alpha\,\varepsilon_i\,(2\lambda_1 - \varepsilon_i\,\lambda_n)) \;=\; \alpha^2\,(\lambda_n + \lambda_1).$$

Hence, the sum of the roots of the polynomial in the left-hand side of (3.6) is just $(\lambda_n + \lambda_1)$; (3.6) therefore holds for $\eta_i = \lambda_1\,\lambda_n\,\alpha^2$, since these two roots are then precisely equal to λ_1 and λ_n, respectively; the left inequality (3.5) readily follows. To prove the right one, note that

$$\frac{\kappa\,(1-\varepsilon_i)^2}{\left(\kappa + \varepsilon_i^2(\kappa-1) + (1-\varepsilon_i)^2\right)^2} \;=\; \frac{\kappa\,\frac{1+\varepsilon_i}{1-\varepsilon_i}\,\frac{\left(1+\varepsilon_i^2\right)^2}{1-\varepsilon_i^2}}{\left(\kappa\,\frac{1+\varepsilon_i}{1-\varepsilon_i}\,\frac{(1+\varepsilon_i^2)^2}{1-\varepsilon_i^2} \;+\; \frac{(1+\varepsilon_i^2)(1-2\,\varepsilon_i)}{(1-\varepsilon_i)^2}\right)^2}$$

whereas $\frac{(1+\varepsilon_i^2)(1-2\,\varepsilon_i)}{(1-\varepsilon_i)^2} \leq 1$ holds since $(1-2\,\varepsilon_i) + \varepsilon_i^2\,(1-2\,\varepsilon_i) \leq 1 - 2\,\varepsilon_i + \varepsilon_i^2$. ∎

We now comment on the assumption (3.4), which will also be used in the next section. It will require that the relative error for the approximate solution of (1.2) is less than ε_i in the B-norm. This latter norm is the most natural if a fixed number of inner CG iterations is performed with B as system matrix, since standard analysis will then result in a bound on this measure of the error. Even when the stopping criterion for the inner iterations is based on the relative residual error, it remains that the error in B-norm is guaranteed to decrease monotonically. Further, in practice, it often happens that all relative measures of the error decrease more or less similarly. Hence, *on average*, one may expect ε_i to be close to the user prescribed tolerance. In this respect, note that we do not require ε_i to be less than 1 at *each* iteration. Thus, even from a purely theoretical point of view, it is not necessary to use a tolerance on the relative residual error that is small enough to guarantee some uniform bound on ε_i.

Now, one should remember that, in general, inner iterations will not be performed directly to solve $B\,\mathbf{x} = \mathbf{y}$, but rather will be involved in some subproblem(s) whose solution is needed for the computation of $\mathbf{x}$. Then an analysis is required to check that a small error for this (these) subproblem(s) cannot be magnified and result in a large global error on $\mathbf{x}$. Of course, such an analysis is application dependent and lies therefore outside the scope of the present paper.

Finally, it is interesting to compare our bound (3.5) with the one recently obtained by Golub and Ye [20, Theorem 3.6]. This comparison is possible because they measure the errors with a criterion similar to (3.4), which is even identical when $\mathcal{B}$ is defined by inner CG iterations with B as the system matrix. However, they consider *two* successive steps of FCG(1) (referred to as IPCG), and their bound, for $\varepsilon \to 0$, tends to the standard bound for two CG iterations. The latter is of course better than the bound for the two steepest descent iterations, and our result therefore cannot compete for small ε. Nevertheless, as seen in Figure 1, Golub–Ye analysis strongly overestimates perturbations effects, so that our bound becomes better for ε larger than approximately 0.05.

4. Equivalent linear operator. The analysis of inexact preconditioning is intrinsically difficult because $\mathcal{B}$ is in general nonlinear and therefore cannot be represented in standard matrix form. Indeed, it would be too restrictive to assume $\mathcal{B}(\mathbf{x}+\mathbf{y}) = \mathcal{B}(\mathbf{x}) + \mathcal{B}(\mathbf{y})$ for any $\mathbf{x}$, $\mathbf{y}$. However, we do not in fact need a matrix representation of $\mathcal{B}(\mathbf{x})$ valid for any $\mathbf{x}$: Our interest is restricted to the vectors $\mathbf{r}_0$, $\mathbf{r}_1$, ... to which $\mathcal{B}$ is effectively applied during the course of the iterations. Moreover, to

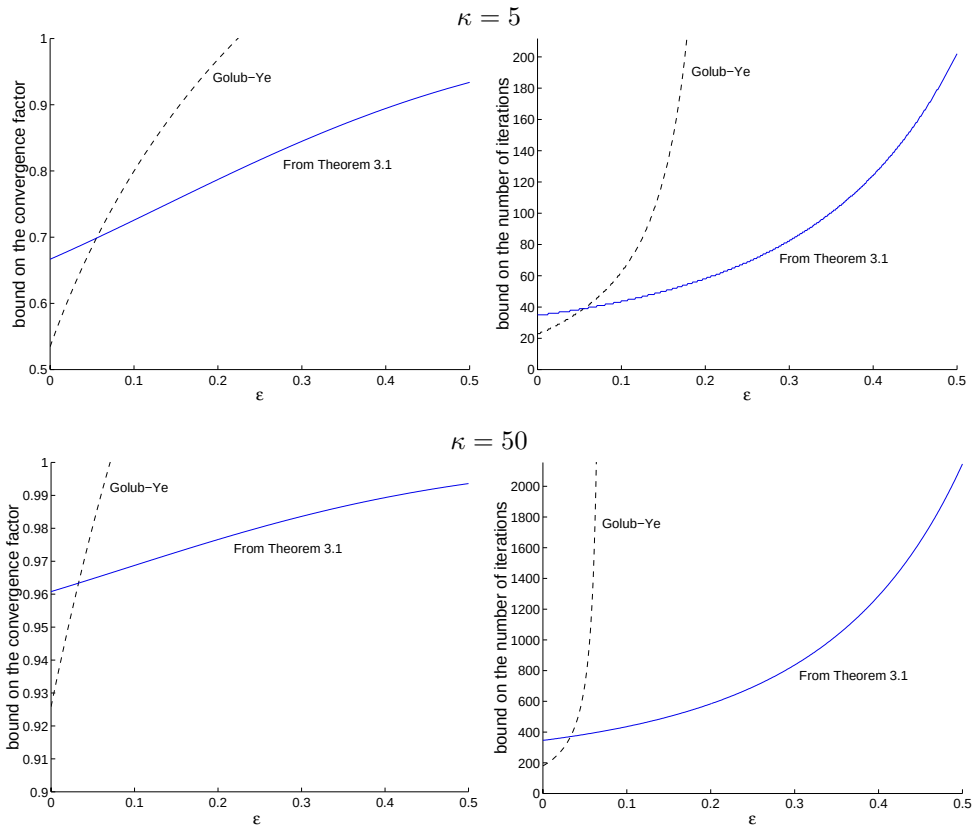

FIG. 1. *Left: convergence factor as estimated by* (3.5) *(solid line) and by Golub–Ye analysis (dashed line). Right: corresponding bound on the number of iterations to reduce the relative error in A-norm by* 10^{-6}.

analyze $\|\mathbf{u} - \mathbf{u}_{i+1}\|$ in view of (2.3), it is not useful to know something about $\mathcal{B}(\mathbf{r}_j)$ for all j. We have only to find a matrix $\widehat{B}_{i,\ell}$ such that

$$\mathcal{B}(\mathbf{r}_j) \;=\; \widehat{B}_{i,\ell}\,\mathbf{r}_j \qquad \text{for } j = i-\ell,\ldots,i \tag{4.1}$$

with $\ell \le m_i$.

It is rather obvious that such a matrix exists as long as the vectors $\mathbf{r}_{i-\ell}, \ldots, \mathbf{r}_i$ are linearly independent (there are even infinitely many possibilities if $\ell + 1 < n$). When using a fixed preconditioner, it is known that this linear independence holds because $(\mathbf{r}_j, B^{-1}\mathbf{r}_k) = 0$ for all $j \neq k$. In the general case, the linear independence is easily proved as soon as Theorem 3.1 (or (3.1)) applies to guarantee $\|\mathbf{r}_i\|_{A^{-1}} < \|\mathbf{r}_{i-1}\|_{A^{-1}} < \cdots < \|\mathbf{r}_{i-\ell}\|_{A^{-1}}$. Indeed, if $\sum_{k=j}^{i} \xi_k \mathbf{r}_k = 0$ with $\xi_j \neq 0$ for some $j \ge i-\ell$, one should have $(\mathbf{r}_j, \mathbf{d}_j) = 0$ (since by (2.2) $(\mathbf{r}_k, \mathbf{d}_j) = 0$ for $k = j+1, \ldots, i$). But this would imply $\mathbf{r}_{j+1} = \mathbf{r}_j$ (see Algorithm 2.1), and this stagnation contradicts the assumption that the error is strictly decreasing.

Now, to gain a further insight, one has to analyze how close the eigenvalues of $\widehat{B}_{i,\ell}A$ are to those of the "target" preconditioned system matrix $B^{-1}A$. This is the purpose of the next theorem. For technical reasons, we assume $m_i = i$ for all i; that is, we limit the analysis to the untruncated version of Algorithm 2.1. We believe that

this is not too severe a limitation because, as discussed below, one should expect from the theorem more qualitative than quantitative insight.

THEOREM 4.1. *Let the assumptions of Theorem* 3.1 *hold, and assume in addition that* $m_i = i$ *for all* i *in Algorithm* 2.1. *Let* i, ℓ *be some nonnegative integers such that* $\ell \leq i$. *For* $j = i - \ell, \ldots, i$, *let*

$$\mathbf{t}_j \;=\; B^{-1}\,\mathbf{r}_j - \mathcal{B}(\mathbf{r}_j), \tag{4.2}$$

and let ε, γ_t, *and* γ_r *be such that*

$$\frac{\|\mathbf{t}_j\|_B}{\|B^{-1}\,\mathbf{r}_j\|_B} \leq \varepsilon, \tag{4.3}$$

$$\frac{|(\mathbf{t}_j,\, B\,\mathbf{t}_k)|}{\|\mathbf{t}_j\|_B\,\|\mathbf{t}_k\|_B} \leq \gamma_t, \qquad k = j+1, \ldots, i, \tag{4.4}$$

$$\frac{|(\mathbf{t}_j,\, \mathbf{r}_k)|}{\|\mathbf{t}_j\|_B\,\|B^{-1}\,\mathbf{r}_k\|_B} \leq \gamma_r, \qquad k = j+1, \ldots, i. \tag{4.5}$$

If

$$\widehat{\varepsilon} \;=\; \varepsilon\,\sqrt{\frac{1+\gamma_t\,\ell}{1-\varepsilon\,\gamma_r\,\ell}} \;<\; 1, \tag{4.6}$$

then there exists a matrix $\widehat{B}_{i,\ell}$ *satisfying*

$$\widehat{B}_{i,\ell}\,\mathbf{r}_j \;=\; \mathcal{B}(\mathbf{r}_j), \qquad j = i-\ell, \ldots, i, \tag{4.7}$$

and such that, noting $W(C)$ *the field of value of the matrix* C, *one has*

$$W\left(A^{1/2}\widehat{B}_{i,\ell}A^{1/2}\right) \;\subset\; \left\{\overline{\lambda} \in \mathcal{C} \;:\; \left|\frac{\overline{\lambda}}{\lambda} - 1\right| \leq \widehat{\varepsilon} \text{ for } \lambda \in W\left(A^{1/2}B^{-1}A^{1/2}\right)\right\} \tag{4.8}$$

and

$$\sigma\left(\widehat{B}_{i,\ell}\,A\right) \;\subset\; \bigcup_{k=1}^{n} \mathcal{D}_k, \tag{4.9}$$

where

$$\mathcal{D}_k \;=\; \left\{\overline{\lambda} \in \mathcal{C} \;:\; \left|\overline{\lambda} - \lambda_k\right| \leq \widehat{\varepsilon}\,\lambda_n\right\} \tag{4.10}$$

is the disk of radius $\widehat{\varepsilon}\lambda_n$ *centered at the* k*th eigenvalue of* $B^{-1}A$. *Moreover, if* s *such disks form a connected domain isolated from the remainder, there are precisely* s *eigenvalues in this domain.*

Finally, letting $\mathbf{z}_k$ *be an eigenvector of* $A^{1/2}\,B^{-1}\,A^{1/2}$ *associated with eigenvalue* λ_k, *there holds*

$$\left|\frac{\left(\mathbf{z}_k,\, A^{1/2}\,\widehat{B}_{i,\ell}\,A^{1/2}\,\mathbf{z}_k\right)}{(\mathbf{z}_k,\, \mathbf{z}_k)} - \lambda_k\right| \;\leq\; \widehat{\varepsilon}\,\lambda_k. \tag{4.11}$$

Proof. With (4.3), one easily sees that Theorem 3.1 applies, and therefore that $\mathbf{r}_{i-\ell}, \ldots, \mathbf{r}_i$ are linearly independent as shown above. There exists then a unique $n \times n$ matrix $\widehat{B}_{i,\ell}$ satisfying (4.7) and

$$\widehat{B}_{i,\ell}\,\mathbf{v} \;=\; B^{-1}\,\mathbf{v} \qquad \text{for all } \mathbf{v}:\ (\mathbf{v},\, B^{-1}\mathbf{r}_j) = 0,\ j = i-\ell,\ldots,i. \tag{4.12}$$

Let $E = \widehat{B}_{i,\ell} - B^{-1}$. First, (4.9) and the subsequent assertion on the localization of the eigenvalues follow from a standard result of perturbation theory [35, p. 88] if $\|A^{1/2}\,E\,A^{1/2}\|_2 \le \widehat{\varepsilon}\,\lambda_n$. Now,

$$\begin{aligned}
\|A^{1/2}\,E\,A^{1/2}\|_2^2 &= \max_{\mathbf{z}\neq 0} \frac{(\mathbf{z},\, A^{1/2}\,E^T\,A\,E\,A^{1/2}\,\mathbf{z})}{(\mathbf{z},\,\mathbf{z})} \\
&= \max_{\mathbf{z}\neq 0} \frac{(E\,\mathbf{z},\, A\,E\,\mathbf{z})}{(\mathbf{z},\, A^{-1}\,\mathbf{z})} \\
&\le \lambda_n^2 \max_{\mathbf{z}\neq 0} \frac{(E\,\mathbf{z},\, B\,E\,\mathbf{z})}{(\mathbf{z},\, B^{-1}\,\mathbf{z})}
\end{aligned}$$

and (4.9) holds if

$$\max_{\mathbf{z}\neq 0} \sqrt{\frac{(E\,\mathbf{z},\, B\,E\,\mathbf{z})}{(\mathbf{z},\, B^{-1}\,\mathbf{z})}} \;=\; \|B\,E\|_{B^{-1}} \;\le\; \widehat{\varepsilon}. \tag{4.13}$$

Likewise, letting $\mathbf{v} = A^{1/2}\mathbf{z}$,

$$\frac{(\mathbf{z},\, A^{1/2}\,\widehat{B}_{i,\ell}\,A^{1/2}\,\mathbf{z})}{(\mathbf{z},\,\mathbf{z})} \;=\; \frac{(\mathbf{v},\, \widehat{B}_{i,\ell}\,\mathbf{v})}{(\mathbf{v},\, A^{-1}\,\mathbf{v})} \;=\; \frac{(\mathbf{v},\, B^{-1}\,\mathbf{v})}{(\mathbf{v},\, A^{-1}\,\mathbf{v})}\left(1 + \frac{(\mathbf{v},\, E\,\mathbf{v})}{(\mathbf{v},\, B^{-1}\,\mathbf{v})}\right),$$

and, noting that $\frac{(\mathbf{v},\, B^{-1}\,\mathbf{v})}{(\mathbf{v},\, A^{-1}\,\mathbf{v})} \in [\lambda_1,\, \lambda_n] = W\left(A^{1/2}B^{-1}A^{1/2}\right)$, we have that (4.8) follows if (4.13) holds since

$$\max_{\mathbf{v}\neq 0} \frac{|(\mathbf{v},\, E\,\mathbf{v})|}{(\mathbf{v},\, B^{-1}\,\mathbf{v})} \;\le\; \max_{\mathbf{v}\neq 0} \frac{\|\mathbf{v}\|_{B^{-1}}\,\|B\,E\,\mathbf{v}\|_{B^{-1}}}{\|\mathbf{v}\|_{B^{-1}}^2} \;=\; \|B\,E\|_{B^{-1}}. \tag{4.14}$$

Further, noting that $\mathbf{v}_k = A^{1/2}\,\mathbf{z}_k$ satisfies $A^{-1}\,\mathbf{v}_k = \lambda_k^{-1}\,B^{-1}\,\mathbf{v}_k$, one has

$$\begin{aligned}
\left|\frac{\left(\mathbf{z}_k,\, A^{1/2}\,\widehat{B}_{i,\ell}\,A^{1/2}\,\mathbf{z}_k\right)}{(\mathbf{z}_k,\,\mathbf{z}_k)} - \lambda_k\right| &= \frac{|\,(\mathbf{z}_k,\, A^{1/2}\,E\,A^{1/2}\,\mathbf{z}_k)\,|}{(\mathbf{z}_k,\,\mathbf{z}_k)} \\
&= \frac{|\,(\mathbf{v}_k,\, E\,\mathbf{v}_k)\,|}{(\mathbf{v}_k,\, A^{-1}\,\mathbf{v}_k)} \\
&= \lambda_k\,\frac{|\,(\mathbf{v}_k,\, E\,\mathbf{v}_k)\,|}{(\mathbf{v}_k,\, B^{-1}\,\mathbf{v}_k)},
\end{aligned}$$

showing with (4.14) that (4.11) also holds when (4.13) is satisfied.

Hence we are left with proving (4.13). In this view, note first that by (4.12) we may restrict the maximum in the left-hand side of (4.13) to vectors of the form $\mathbf{z} = \sum_{k=i-\ell}^{i} \xi_k \mathbf{r}_k$. Let then, for $j_0 = 1, \ldots, \ell+1$,

$$y_{j_0} \;=\; \|\mathbf{r}_{j_0+i-\ell-1}\|_{B^{-1}}\,\xi_{j_0+i-\ell-1}$$

and set $y = \begin{pmatrix} y_1 & y_2 & \dots & y_{\ell+1} \end{pmatrix}^T$. Note that, for $i - \ell \le k \le i$, $E\,\mathbf{r}_k = \widehat{B}_{i,\ell}\,\mathbf{r}_k - B^{-1}\,\mathbf{r}_k = -\mathbf{t}_k$, one has

$$\frac{(E\,\mathbf{z},\, B\,E\,\mathbf{z})}{(\mathbf{z},\, B^{-1}\,\mathbf{z})} \;=\; \frac{y^T\,G\,y}{y^T\,H\,y},$$

where G, H are the symmetric $(\ell+1) \times (\ell+1)$ matrices with components

$$G_{j_0k_0} = \frac{(\mathbf{t}_j,\, B\,\mathbf{t}_k)}{\|\mathbf{r}_j\|_{B^{-1}}\,\|\mathbf{r}_k\|_{B^{-1}}},$$
$$H_{j_0k_0} = \frac{(\mathbf{r}_j,\, B^{-1}\,\mathbf{r}_k)}{\|\mathbf{r}_j\|_{B^{-1}}\,\|\mathbf{r}_k\|_{B^{-1}}}$$

(letting $j = j_0 + i - \ell - 1$ and $k = k_0 + i - \ell - 1$).

Now, (4.3) and (4.4) straightforwardly yield $G_{j_0j_0} \le \varepsilon^2$ and $G_{j_0k_0} \le \varepsilon^2\,\gamma_t$ for $k_0 \ne j_0$. Hence, the largest eigenvalue of G is bounded by

$$\lambda_{\max}(G) \;\le\; \|G\|_\infty \;\le\; \varepsilon^2\,(1 + \gamma_t\,\ell).$$

Concerning H, we note that $H_{j_0j_0} = 1$ for all j_0, whereas, when $k > j$,

$$(\mathbf{r}_k,\, B^{-1}\,\mathbf{r}_j) \;=\; (\mathbf{r}_k,\, \mathcal{B}(\mathbf{r}_j)\,) + (\mathbf{r}_k,\, \mathbf{t}_j) \;=\; (\mathbf{r}_k,\, \mathbf{t}_j)$$

because $\mathcal{B}(\mathbf{r}_j)$ is a linear combination of $\mathbf{d}_0$, $\mathbf{d}_1$, $\dots$, $\mathbf{d}_j$, and (2.2) applies therefore to show $(\mathbf{r}_k,\, \mathcal{B}(\mathbf{r}_j)\,) = 0$.[2] One then readily obtains with (4.5) that $|H_{k_0j_0}| = |H_{j_0k_0}| \le \varepsilon\,\gamma_r$ for $k_0 > j_0$. The Gerschgorin theorem then yields

$$\lambda_{\min}(H) \;\ge\; 1 - \varepsilon\,\gamma_r\,\ell.$$

$\frac{y^T\,G\,y}{y^T\,H\,y} \le \widehat{\varepsilon}^2$ and therefore the required result readily follows. ∎

Note that, with (2.3), Theorem 4.1 yields

$$\frac{\|\mathbf{u} - \mathbf{u}_{i+1}\|_A}{\|\mathbf{u} - \mathbf{u}_{i-\ell}\|_A} \;\le\; \min_{\substack{\mathcal{P}_{\ell+1}\text{pol. of degree }(\ell+1)\\ \mathcal{P}_{\ell+1}(0)=1}} \left\|\mathcal{P}_{\ell+1}\left(A^{1/2}\,\widehat{B}_{i,\ell}\,A^{1/2}\right)\right\|. \tag{4.15}$$

A bound on the convergence rate can then be deduced by selecting shifted Chebyshev polynomials for which, thanks to (4.8), we may apply Eiermann analysis based on the field of value [14, Theorem 1]. Clearly, the resulting bound on the asymptotic convergence factor will tend to the usual bound $\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}$ as $W(A^{1/2}\widehat{B}_{i,\ell}A^{1/2})$ gets closer to $[\lambda_1,\, \lambda_n]$, that is, for ε "sufficiently small."

We did not pursue this direction, however, because the requirements on ε would be too stringent and no longer compatible with our general goal of allowing loose accuracy for the inner solutions. Indeed, in theory, we can only say about γ_t and γ_r that they do not exceed 1 by virtue of the Cauchy–Schwarz inequality (with respect to the $(\cdot,\, B\,\cdot)$ inner product). Hence, for realistic ε, the theoretical bound on $\widehat{\varepsilon}$ grows quickly with the number of iterations, even if in practice γ_t and γ_r are both expected to be small, because we act in a very large space and the error vectors $\mathbf{t}_j$ have at first sight no serious reason to be aligned in one preferred direction.

[2] This is the technical detail that motivates the restriction to the untruncated version of Algorithm 2.1.

Theorem 4.1 is also too pessimistic when it requires no truncation, whereas FCG(1) or even standard CG is observed to be quite robust in practice; see, e.g., [17, 18, 20]. This is not completely surprising if one takes into account the general analysis of perturbed CG sequences [22, 23, 31, 26], which indeed display that the *linear* convergence properties of the method are essentially preserved, at least when such perturbations are tiny. Extrapolating these conclusions to larger perturbations gives then a heuristic explanation of the observed robustness.

Now, fast convergence is often possible only on account of the so-called *superlinear* convergence of CG. This arises when the condition number is relatively large, but with a good separation of the extremal eigenvalues, so that interesting bounds on the convergence can nevertheless be proved by selecting polynomials that vanish exactly at these isolated eigenvalues [6, 32]. Since (4.9) and (4.11) show that the spectrum of $\widehat{B}_{i,\ell}A$ is only a slight perturbation of that of $B^{-1}A$,[3] we expect a similar superlinear convergence when applying untruncated FCG, even if, here again, trying to prove this rigorously would lead to too stringent requirements on ε.

Concerning truncated FCG, however, we cannot put forward in this context the heuristic argument above based on the general results about perturbed CG sequences. Indeed, detailed analysis reveals that these superlinear bounds are then essentially unreliable, especially when the perturbations are not that small [26]. Hence, we rather expect a big difference between truncated and untruncated versions when the extremal eigenvalues are well separated.

5. Numerical results.

Theorem 3.1 delivers a bound that, at first order in ε, corresponds to the usual bound for the steepest descent in which the condition number is magnified by a factor $\frac{1+\varepsilon}{1-\varepsilon}$. As this does not take into account global convergence effects, this is clearly much too pessimistic for (sufficiently) small ε when using FCG(m) with $m > 0$. On the other hand, attempts to improve the analysis as in Theorem 4.1 or in [20] lead to too stringent requirements on ε.

We therefore resorted to numerical tests to further assess the real convergence rate. In section 5.1, we conducted very simple experiments with artificial examples to investigate to what extent the number of iterations can be realistically estimated by making the substitution $\kappa \to \kappa\frac{1+\varepsilon}{1-\varepsilon}$ in the standard bounds for the CG method with a fixed preconditioner. On the other hand, in sections 5.2 and 5.3, we considered more realistic computations, aiming principally at comparing FCG with standard CG while assessing the real effects of truncation.

5.1. Some artificial experiments.

Here, we let $A = \mathrm{diag}(\lambda_i)$ with

Case 1: $\lambda_i = 1 + \kappa\,\frac{i-1}{n-1}$, $\kappa = 5$, $i = 1, \ldots, n$;
Case 2: $\lambda_i = 1 + \kappa\,\frac{i-1}{n-1}$, $\kappa = 50$, $i = 1, \ldots, n$;
Case 3: $\lambda_1 = 10^{-2}$, $\lambda_i = 1 + \kappa_2\,\frac{i-2}{n-2}$, $\kappa_2 = 10$, $i = 2, \ldots, n$.

We further set $B = I$ and consider two kinds of perturbations for the "solution" of $I\,\mathbf{w}_i = \mathbf{r}_i$:

(a) $\mathbf{w}_i = \mathbf{r}_i + \varepsilon\,\frac{\|\mathbf{r}_i\|}{\|\mathbf{f}\|}\,\mathbf{f}$, the components of $\mathbf{f}$ being pseudorandom numbers uniformly distributed in $[-1, 1]$;
(b) $\mathbf{w}_i$ computed from inner CG iterations with the zero vector as initial approximation, $B = I$ as system matrix, and ε as prescribed accuracy on the relative residual error (which is also equal to the relative error measured in

[3] The eigenvalues of $\widehat{B}_{i,\ell}A$ satisfy $\lambda_k(\varepsilon) = \frac{\left(\mathbf{z}_k,\, A^{1/2}\,\widehat{B}_{i,\ell}\,A^{1/2}\,\mathbf{z}_k\right)}{(\mathbf{z}_k,\,\mathbf{z}_k)} + \mathcal{O}\left(\varepsilon^2\right)$; see [35, p. 69].

TABLE 1

Number of iterations for the artificial examples; FCG(1) is used for Cases 1 and 2 and FCG(∞) is used for Case 3.

ε	0	10^{-2}	10^{-1}	$\frac{1}{7}$	$\frac{1}{4}$	$\frac{1}{3}$	$\frac{1}{2}$	1
Case 1:	$\lambda_i = 1 + \kappa\,(i-1)/(n-1)$, $\kappa = 5$							
Estimate (5.1)	17	17	18	19	21	23	29	
Random pert.	15	15	16	17	19	22	28	
Inner PCG		15	16	17	19	21	24	
(# inner it.)		(117)	(64)	(66)	(56)	(52)	(47)	(49)
Case 2:	$\lambda_i = 1 + \kappa\,(i-1)/(n-1)$, $\kappa = 50$							
Estimate (5.1)	52	52	57	60	67	73	89	
Random pert.	49	49	55	59	69	81	116	
Inner PCG		50	54	56	64	75	71	
(# inner it.)		(397)	(216)	(222)	(191)	(153)	(141)	(155)
Case 3:	$\lambda_1 = 10^{-2}$, $\lambda_i = 1 + \kappa_2\,(i-2)/(n-2)\,$, $\kappa_2 = 10$, $i > 1$							
Estimate (5.2)	35	35	38	39	45	48	59	
Random pert.	31	31	32	33	37	40	49	
Inner PCG		31	33	33	40	41	42	
(# inner it.)		(246)	(132)	(130)	(119)	(90)	(83)	(99)

the B-norm); since $B = I$, this inner CG process is nontrivial only if one uses a "preconditioner" C such that $C^{-1}B \neq I$, and we actually selected C defined by $C = \mathrm{diag}(c_i)$ with $c_i^{-1} = 1 + 10\,\frac{i-1}{n-1}$ (hence $\kappa(C^{-1}B) = 10$).

For Cases 1 and 2, the heuristic estimate of the number of iterations is

$$k_\delta \;=\; \mathrm{int}\left[\frac{1}{2}\,\sqrt{\kappa\tfrac{1+\varepsilon}{1-\varepsilon}}\,\ln\frac{2}{\delta}\right] + 1, \tag{5.1}$$

where δ is the prescribed accuracy on the relative error measured in the A-norm. For Case 3, we have to take into account that the first eigenvalue is isolated. A heuristic estimate is then obtained by multiplying κ_2 by $\frac{1+\varepsilon}{1-\varepsilon}$ in the appropriate bound, which gives [6, 26]

$$k_\delta \;=\; \mathrm{int}\left[\frac{1}{2}\sqrt{\kappa_2\tfrac{1+\varepsilon}{1-\varepsilon}}\left(\ln\frac{2}{\delta} + \ln\frac{\lambda_2}{\lambda_1}\right)\right] \;+\; \mathrm{int}\left[\sqrt{\kappa_2\tfrac{1+\varepsilon}{1-\varepsilon}} + 1\right] \;+\; 1. \tag{5.2}$$

We made tests with both truncated and untruncated FCG, using in each case $n = 10^4$, $\delta = 10^{-6}$, a pseudorandom right-hand side, and the zero vector as initial approximation. It turned out that, for both Cases 1 and 2, there was almost no difference between FCG(1) and the untruncated version, so we report in Table 1 the results for FCG(1) only. On the other hand, for Case 3, truncated versions converged much more slowly, so we report the results for the untruncated version only; in other cases the comparison with the heuristic estimate (5.2) is meaningless anyway (truncated and untruncated versions are compared further in section 5.3). Besides, we also report the total number of inner iterations for the choice (b), where $\mathbf{w}_i$ is generated by performing inner CG iterations; the number given for $\varepsilon = 1$ is then the number of CG iterations when A is directly preconditioned by C.

One sees that the heuristic estimates predict relatively well the actual convergence, although they cannot pretend in any way to be *upper bounds* on the number of

iterations. In concrete applications, the optimal ε will clearly depend on the overhead involved by outer iterations, and it is therefore difficult to define a general rule. The sensitivity to variations in the preconditioner is somewhat higher when the condition number is relatively large, but such effects seem very slight, at least on these examples. We therefore do not necessarily confirm that ε should be kept proportional to $\kappa^{-1/2}$ as advised in [20].

5.2. A problem with no separated eigenvalues. We consider the linear systems resulting from the finite element discretization of the linear elasticity equations. The aim is to compute the displacement at each gridpoint. Thus, in three dimensions, there are three unknowns per gridpoint, one for each component. If these unknowns are ordered separately, the system matrix presents the 3×3 block structure

$$A = \begin{pmatrix} A_{xx} & A_{xy} & A_{xz} \\ A_{yx} & A_{yy} & A_{yz} \\ A_{zz} & A_{zy} & A_{zz} \end{pmatrix}.$$

A is then known to be spectrally equivalent to its block diagonal part [5, 10]; thus we select

$$B = \begin{pmatrix} A_{xx} & & \\ & A_{yy} & \\ & & A_{zz} \end{pmatrix}.$$

B is itself easily preconditioned, because each diagonal block is similar to a weakly anisotropic discrete Laplace operator. Using more particularly the finite element scheme described in [10], these blocks are even symmetric M-matrices, allowing an appropriate use of incomplete factorization methods (e.g., [4, 13]). This preconditioner of B can serve as is to precondition A [10, 30], but there are some (potential) advantages in using an inner–outer iterative scheme [5], because A is much denser than B; hence it is worth trying to keep the number of multiplications by A to a minimum.

For the numerical test, we consider more particularly the mining stability problem referred to as "Stope" in [11], which is discretized on a $40 \times 40 \times 40$ regular grid by means of the finite element scheme described in [10]. The numerically computed smallest eigenvalues of $B^{-1}A$ are $0.301, 0.326, 0.369, \ldots$, and the largest ones (in decreasing order) are $1.98, 1.91, 1.79, \ldots$. Thus, $\kappa = 6.58$ and there are no isolated eigenvalues.

We report in Table 2 the number of (outer) iterations necessary to reduce the relative residual error below 10^{-6} whenever using the zero vector as an initial approximation; ε is the tolerance used for the inner iterations, the stopping test being also based on the relative residual error and checked independently for each of the three systems $A_{xx}\,\mathbf{w}_x = \mathbf{r}_x$, $A_{yy}\,\mathbf{w}_y = \mathbf{r}_y$, $A_{zz}\,\mathbf{w}_z = \mathbf{r}_z$ (note for completeness that these diagonal blocks were preconditioned by the dynamic relaxed block ILU algorithm from [25]); we also indicate the values taken by the heuristic estimate (5.1).

One may observe a nice agreement between these numerical results and our previous conclusions: As long as the inner solutions remain reasonably accurate, the standard CG algorithm and FCG($m_{\max}$) behave similarly and deliver a convergence close to that of the method with exact preconditioning. Note, however, that for very inaccurate inner solutions, FCG tends to be more stable than CG, and it seems to still obey the rule that the condition number is just multiplied by $\frac{1+\varepsilon}{1-\varepsilon}$. As this stabilization is already obtained for FCG(1), we recommend its use for problems with

TABLE 2
Results for the problem with no separated eigenvalues.

ε	10^{-6}	10^{-3}	10^{-2}	10^{-1}	$\frac{1}{7}$	$\frac{1}{4}$	$\frac{1}{3}$	$\frac{1}{2}$
CG	17	17	17	18	19	29	34	68
FCG(1)	17	17	17	18	18	19	22	31
FCG(15)	17	17	17	18	18	20	21	31
Estimate (5.1)	19	19	19	21	22	24	26	33

this type of eigenvalue distribution. Practitioners may be satisfied with the standard CG algorithm, but for a fairly small overhead (one more inner product computation per iteration) FCG(1) allows the inner solutions to be still less accurate and therefore cheaper.

5.3. A problem with well-separated extremal eigenvalues. To generate a simple example in which $B^{-1}A$ has isolated eigenvalues, we apply a very basic additive Schwarz preconditioning scheme (e.g., see [12] and the references therein) to a problem from [33], namely, the linear system resulting from the five-point approximation of

$$-\partial_x\, a\, \partial_x\, u - \partial_y\, a\, \partial_y\, u = f$$

in the unit square, with homogeneous Dirichlet boundary conditions on the bottom boundary and homogeneous Neumann boundary conditions elsewhere, and with $a = f = 100$ in the box $\left(\frac{1}{4}, \frac{1}{4}\right) \times \left(\frac{1}{4}, \frac{1}{4}\right)$ and $a = 1$, $f = 0$ elsewhere; we use a uniform grid of mesh size $h = \frac{1}{160}$.

The partitioning is based on a division of the domain (that is, the unit square) in 4×2 subdomains with internal boundaries located at $x = \frac{1}{4}$, $x = \frac{1}{2}$, $x = \frac{3}{4}$, and $y = \frac{1}{2}$. Since we want to simulate numerical difficulties that may be unavoidable in other contexts, we deliberately omit the coarse grid correction and consider minimal overlap; that is, the unknowns for each subdomain are just those corresponding to the gridpoints effectively located in the subdomain or on its boundary.

Letting B be the preconditioner corresponding to exact subdomain solves, we have that the numerically computed smallest eigenvalues of $B^{-1}A$ are $0.207\,10^{-3}$, 0.0486, 0.698, ..., and the largest ones (in decreasing order) are 4.00, 2.08, 2.00, ... Hence, $\kappa = 1.9\,10^4$, which is much too large to obtain a satisfactory convergence without the superlinear convergence effects mentioned at the end of section 4.

We report in Table 3 the number of (outer) iterations necessary to reduce the relative residual error below 10^{-6} whenever we use the zero vector as an initial approximation; ε is again the tolerance for the inner subdomain solves, also verified with respect to the relative residual error; for these solves, we used the standard MILU preconditioning without fill-in [4, 13].

The results for both FCG(1) and the untruncated version (FCG(∞)) just confirm our expectation from the previous discussions. The good news is that satisfactory convergence for fairly moderate accuracy of the inner solutions is already restored with reasonable values of $m_{\max}$. We do not discuss which couple (ε, $m_{\max}$) is most effective because this heavily depends on the context, for instance on the quality of the preconditioner for the inner iterations. Note, however, that we programmed the orthogonalization of $\mathbf{d}_i$ in a classical Gram–Schmidt fashion, so that the communication cost per iteration is essentially independent of $m_{\max}$; the tradeoff between increasing $m_{\max}$ or decreasing ε to achieve a given convergence can thus be analyzed by

TABLE 3
Numbers of iteration for the problem with small isolated eigenvalues; Tr–FCG ($m_{\max}$) refers to Algorithm 2.1 with $m_i = \min(i, m_{\max})$.

ε	10^{-6}	10^{-3}	10^{-2}	10^{-1}	$\frac{1}{7}$	$\frac{1}{4}$	$\frac{1}{3}$	$\frac{1}{2}$
FCG(1)	76	86	115	250	400	771	>999	>999
FCG(5)	80	86	117	158	192	167	201	230
FCG(10)	61	86	88	92	125	156	131	142
FCG(20)	62	62	83	86	90	94	97	129
FCG(30)	59	60	63	67	68	72	94	105
FCG(45)	60	61	63	63	64	66	69	77
FCG(∞)	58	59	60	62	62	64	66	71
Tr–FCG(10)	60	84	85	114	125	124	160	144
Tr–FCG(20)	62	80	83	87	90	94	95	105

comparing the cost of the inner solves with that of local vector operations. Which convergence one should target is difficult to predict, but, as a general rule, it seems reasonable to admit only a moderate deterioration of the convergence from the "ideal" case with exact preconditioning. Indeed, inner–outer iterative processes are primarily motivated by the high cost of outer iterations, for instance, the communication cost in the present example. (Would it not be the case, one probably should skip inner iterations and use one application of the corresponding preconditioner instead.)

For the sake of completeness, we also performed some runs using Algorithm 2.1 with a pure truncation strategy (Tr–FCG($m_{\max}$)). Clearly, the additional work with respect to FCG($m_{\max}$) does not pay off since the convergence is nearly the same in all cases, whereas FCG($2\,m_{\max}$) is much faster despite a similar cost per iteration. We also tested a pure restart approach, but we did not succeed in obtaining a useful convergence. This is not surprising since, even with a fixed preconditioner, pure restarting has a disastrous effect with this kind of eigenvalue distribution.

6. Conclusions. When one is willing to use inexact preconditioning, one should distinguish two types of problems.

For problems with a regular distribution of the eigenvalues, for which the fast convergence is due to a nice condition number, inexact preconditioning may be used quite safely, even with very loose stopping criteria for the inner iterations. It is not necessary to use a method much more costly than the standard CG algorithm, but FCG(1) brings still more stability at the price of a fairly small overhead.

The situation is more tricky when a relatively fast convergence depends on "superlinear" effects in connection with a good separation of the extremal eigenvalues. A convergence close to that of the ideal case with exact preconditioning can still be obtained with FCG($m_{\max}$), but at the price of increasing $m_{\max}$ while decreasing the accuracy requirements for the inner solutions. Hence, one will not be able to use stopping criteria as loosely as in the preceding case, and the FCG process itself will be more costly.

These considerations also tell us that whenever using inner–outer iterative schemes, improving the basic preconditioner always brings some additional benefit. For instance, removing small isolated eigenvalues not only accelerates the convergence with exact preconditioning, but also indirectly allows a decrease in the cost of inner

solutions.

Generally speaking, we feel that coupled inner–outer iterations can be efficient when at least either the global preconditioner or the one used for the inner solutions is a highly effective. Indeed, in the former case, the CG process will be very stable, allowing quite loose stopping criteria for the inner iterations. On the other hand, in the latter case, it is relatively cheap to prevent stability problems by requiring a high accuracy for the inner solutions. These considerations also show that when both preconditioners are of medium quality, then coupled inner–outer iterations are not necessarily advisable.

Acknowledgments. My interest in variable preconditioning originates from private discussions with O. Axelsson and his coworkers M. Neytcheva and B. Polman. I also thank R. Blaheta and R. Kohut who provided the code that generates the matrix for the first test example. Mark Embree provided me with some interesting remarks, pointing out the possible use of Eiermann results. Magolu Monga-Made informed me of reference [20] as soon as it appeared.

REFERENCES

[1] O. Axelsson, *Conjugate gradient type methods for unsymmetric and inconsistent systems of linear equations*, Linear Algebra Appl., 29 (1980), pp. 1–16.

[2] O. Axelsson, *A generalized conjugate gradient, least square method*, Numer. Math., 51 (1987), pp. 209–227.

[3] O. Axelsson, *A restarted version of a generalized preconditioned conjugate gradient method*, Comm. Appl. Numer. Methods, 4 (1988), pp. 521–530.

[4] O. Axelsson, *Iterative Solution Methods*, Cambridge University Press, Cambridge, UK, 1994.

[5] O. Axelsson, *On Iterative Solvers in Structural Mechanics; Separate Displacement Orderings*, Tech. report 9721, Department of Mathematics, Catholic University, Nijmegen, The Netherlands, 1997.

[6] O. Axelsson and G. Lindskog, *On the rate of convergence of the preconditioned conjugate gradient method*, Numer. Math., 48 (1986), pp. 499–523.

[7] O. Axelsson and P. S. Vassilevski, *A black box generalized conjugate gradient solver with inner iterations and variable-step preconditioning*, SIAM J. Matrix Anal. Appl., 12 (1991), pp. 625–644.

[8] O. Axelsson and P. S. Vassilevski, *Variable-step multilevel preconditioning methods.* I. *Self-adjoint and positive definite elliptic problems*, Numer. Linear Algebra Appl., 1 (1994), pp. 75–101.

[9] R. Barrett, M. Berry, T. F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. van der Vorst, *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*, SIAM, Philadelphia, 1994.

[10] R. Blaheta, *Displacement decomposition—incomplete factorization preconditioning techniques for linear elasticity problems*, Numer. Linear Algebra Appl., 1 (1994), pp. 107–126.

[11] R. Blaheta, O. Jakl, and J. Starý, *A parallel cg solver for FE analysis of* 3D *problems in geomechanics*, in Geomechanics 96, Z. Rakowski, ed., Balkema, Rotterdam, 1997, pp. 159–163.

[12] T. Chan and T. Mathew, *Domain decomposition algorithms*, Acta Numer., 3 (1994), pp. 61–143.

[13] T. Chan and H. van der Vorst, *Approximate and incomplete factorizations*, in Parallel Numerical Algorithms, D. E. Keyes, A. Samed, and V. Venkatakrishnan, eds., ICASE/LaRC Interdisciplinary Series in Science and Engineering, Vol. 4, Kluwer Academic, Dordecht, 1997, pp. 167–202.

[14] M. Eiermann, *Fields of values and iterative methods*, Linear Algebra Appl., 180 (1993), pp. 167–197.

[15] H. C. Elman and G. H. Golub, *Inexact and preconditioned Uzawa algorithms for saddle point problems*, SIAM J. Numer. Anal., 31 (1994), pp. 1645–1661.

[16] E. Giladi, G. H. Golub, and J. B. Keller, *Inner and outer iterations for the Chebyshev algorithm*, SIAM J. Numer. Anal., 35 (1998), pp. 300–319.

[17] G. H. Golub and M. Overton, *Convergence of two-stage Richardson iterative procedure for*

solving systems of linear equations, in Numerical Analysis, G. Watson, ed., Lectures Notes in Math. 912, Springer-Verlag, Berlin Heidelberg, New York, 1983, pp. 128–139.

[18] G. H. Golub and M. L. Overton, *The convergence of inexact Chebyshev and Richardson iterative methods for solving linear systems*, Numer. Math., 53 (1988), pp. 571–593.

[19] G. H. Golub and C. F. van Loan, *Matrix Computations*, 3rd ed., The John Hopkins University Press, Baltimore, MD, 1996.

[20] G. H. Golub and Q. Ye, *Inexact preconditioned conjugate gradient method with inner-outer iterations*, SIAM J. Sci. Comput., 21 (1999), pp. 1305–1320.

[21] A. Greenbaum, *Iterative Methods for Solving Linear Systems*, SIAM, Philadelphia, 1977.

[22] A. Greenbaum, *Behavior of slightly perturbed Lanczos and conjugate-gradient recurrences*, Linear Algebra Appl., 113 (1989), pp. 7–63.

[23] A. Greenbaum and Z. Strakos, *Predicting the behavior of finite precision Lanczos and conjugate gradient computations*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 121–137.

[24] M. R. Hestenes and E. Stiefel, *Methods of conjugate gradients for solving linear systems*, J. Res. Nat. Bur. Standards, 49 (1952), pp. 409–436.

[25] M. M. Monga-Made and Y. Notay, *Dynamically relaxed block incomplete factorizations for solving two- and three-dimensional problems*, SIAM J. Sci. Comput., 21 (2000), pp. 2008–2028.

[26] Y. Notay, *On the convergence rate of the conjugate gradients in presence of rounding errors*, Numer. Math., 65 (1993), pp. 301–317.

[27] J. K. Reid, *On the method of conjugate gradients for the solution of large sparse systems of linear equations*, in Large Sparse Sets of Linear Equations, J. Reid, ed., Academic Press, London, 1971, pp. 231–254.

[28] Y. Saad, *A flexible inner-outer preconditioned GMRES algorithm*, SIAM J. Sci. Comput., 14 (1993), pp. 461–469.

[29] Y. Saad, *Iterative Methods for Sparse Linear Systems*, PWS Publishing, New York, 1996.

[30] P. Saint-Georges, G. Warzee, R. Beauwens, and Y. Notay, *High performance PCG solvers for FEM structural analyses*, Internat. J. Numer. Methods Engrng., 39 (1996), pp. 1313–1340.

[31] Z. Strakos, *On the real convergence rate of the conjugate gradient method*, Linear Algebra Appl., 154/156 (1991), pp. 535–549.

[32] A. van der Sluis and H. A. van der Vorst, *The rate of convergence of conjugate gradients*, Numer. Math., 48 (1986), pp. 543–560.

[33] H. A. van der Vorst, *The convergence behaviour of preconditioned CG and CG-S*, in Preconditioned Conjugate Gradient Methods, O. Axelsson and L. Kolotilina, eds., Lectures Notes in Math. 1457, Springer-Verlag, New York, 1990, pp. 126–136.

[34] H. A. van der Vorst and C. Vuik, *GMRESR: A family of nested GMRES methods*, Numer. Linear Algebra Appl., 1 (1994), pp. 369–386.

[35] J. H. Wilkinson, *The Algebraic Eigenvalue Problem*, Clarendon Press, Oxford, 1965.