SIAM J. MATRIX ANAL. APPL.
Vol. 13, No. 3, pp. 926–943, July 1992

014

LANCZOS METHODS FOR THE SOLUTION OF NONSYMMETRIC SYSTEMS OF LINEAR EQUATIONS*

WAYNE JOUBERT†

Abstract. The Lanczos or biconjugate gradient method is often an effective means for solving nonsymmetric systems of linear equations. However, the method sometimes experiences breakdown, a near division by zero which may hinder or preclude convergence. In this paper we present some theoretical results on the nature and likelihood of the phenomenon of breakdown. We also define several new algorithms that substantially mitigate the problem of breakdown. Numerical comparisons of the new algorithms and the standard algorithms are given.

Key words. linear systems, iterative methods, nonsymmetric, Lanczos

AMS(MOS) subject classifications. 65F10, 65F15

1. Introduction.

In this paper we consider methods for solving the linear system of equations

$$Au = b, \tag{1}$$

where $A \in \mathbb{C}^{N \times N}$ is a given nonsingular matrix.

When A is large and sparse, iterative methods in many cases are effective means for solving (1). In particular, when A is Hermitian and positive definite (HPD), the conjugate gradient (CG) method [21] is an effective solution technique for (1).

However, the case when A is nonsymmetric is substantially more difficult to solve efficiently by means of iterative methods. For example, the important CG method cannot be generalized to the nonsymmetric case without a serious loss of some of its more useful properties (see [9], [10], and [27]). This difficulty has led to the development of a wide variety of generalized CG methods having varying degrees of success (for an overview, see, for example, [42], [2], or [26]; see also [25]).

The biconjugate gradient or Lanczos method [29], [11], [41] is an important example of a generalized CG method. In many cases, Lanczos algorithms give some of the fastest solution times among all generalized CG methods (see, e.g., [30] and [1]).

However, the Lanczos method is known to break down in some cases. In practice, the occurrence of a breakdown or near-breakdown of the method can cause failure to converge to the solution of (1). Furthermore, the size of the iterates generated by the Lanczos method may become arbitrarily large during the iteration process, which can introduce numerical error into the approximate solution.

Comparatively little is known about the theoretical properties of the Lanczos method (see, e.g., [7]). The fact that Lanczos algorithms perform very well in some cases but fail in others heightens the need for further insight into the theoretical properties of the Lanczos method.

In this paper we present theoretical results on the Lanczos method as well as new algorithms that are better able to deal with the problem of breakdown of the Lanczos

*Received by the editors October 8, 1990; accepted for publication March 7, 1991. This work was supported in part by National Science Foundation grant DCR-8518722, by Department of Energy grant DE-FG05-87ER25048, and by Cray Research Inc. grant LTRDTD 1/18/90, with the University of Texas at Austin.

†The University of Texas at Austin, Austin, Texas 78713 (joubert@emx.utexas.edu).

method. This will be organized as follows. In §2 we define the Lanczos method and its algorithms, and in §3 we examine the conditions that lead to breakdown of the algorithms. Then in §§4 and 5 we give results on the likelihood of breakdown of the Lanczos algorithms, and in §6 we analyze the important categories of curable and incurable breakdown. After this, in §7 we define modified Lanczos algorithms, and the results of numerical experiments with these algorithms are presented in §8.

2. The Lanczos method. The Lanczos method is a particular instance of an iterative method, which is defined as a procedure that for a given initial guess $u^{(0)}$ may be used to compute subsequent iterates $\{u^{(i)}\}_{i\geq 1}$, which approximate the true solution $u = A^{-1}b$. We denote the corresponding residuals by $r^{(n)} = b - Au^{(n)}$, and the error vector is denoted by $e^{(n)} = u^{(n)} - u = -A^{-1}r^{(n)}$.

Specifically, the Lanczos method is defined by the following two properties:

$$u^{(n)} - u^{(0)} \in \mathbf{K}_n(r^{(0)}, A), \qquad r^{(n)} \perp \mathbf{K}_n(\tilde{r}^{(0)}, A^*). \tag{2}$$

Here, the Krylov space is defined as $\mathbf{K}_n(v, A) = \operatorname{span}\{A^i v\}_{i=0}^{n-1}$. The vector $\tilde{r}^{(0)}$ is an auxiliary vector supplied to the algorithm, typically defined by $\tilde{r}^{(0)} = \tilde{Z}^* r^{(0)}$ for some matrix $\tilde{Z}$ which is commonly set to $\tilde{Z} = I$. Here we use the notation X^* to denote the complex conjugate of X when the quantity X is a scalar and the conjugate transpose when X is a vector or matrix. We also define the standard inner product $(u, v) = u^* v$.

The first condition of (2) indicates that the method is a *polynomial method* in the matrix A. The second condition of (2), which is the orthogonality or *Petrov–Galerkin* condition, categorizes the method as an example of a *projection method* (see [26]). Unfortunately, in general there is no guarantee that the two conditions of (2) necessarily define a unique iterate $u^{(n)}$.

For certain choices of A and $\tilde{Z}$, the method (2) reduces to a standard conjugate gradient method (see [23] and [26]). For example, if A is HPD and $\tilde{Z} = I$, then (2) reduces to the conjugate gradient method of [21]; if A is HPD and $\tilde{Z} = A$, then (2) gives the standard conjugate residual method (see, e.g., [11] and [2]).

The abstract method (2) may be implemented by various algorithms. Three examples of such Lanczos algorithms are the Lanczos/Orthodir, Lanczos/Orthomin, and Lanczos/Orthores algorithms [23]. Of these, the Lanczos/Orthomin version is most commonly used and is also referred to as the biconjugate gradient (BCG) algorithm.

LANCZOS/ORTHODIR ALGORITHM

$$q^{(0)} = r^{(0)}; \qquad q^{(n)} = Aq^{(n-1)} - a_n q^{(n-1)} - b_n q^{(n-2)}, \quad n > 0,$$

$$\tilde{q}^{(0)} = \tilde{r}^{(0)}; \qquad \tilde{q}^{(n)} = A^*\tilde{q}^{(n-1)} - a_n^*\tilde{q}^{(n-1)} - b_n^*\tilde{q}^{(n-2)}, \quad n > 0,$$

$$a_n = \frac{(A^*\tilde{q}^{(n-1)}, Aq^{(n-1)})}{(\tilde{q}^{(n-1)}, Aq^{(n-1)})}, \qquad b_n = \frac{(\tilde{q}^{(n-1)}, Aq^{(n-1)})}{(\tilde{q}^{(n-2)}, Aq^{(n-2)})} \quad (b_1 = 0),$$

$$u^{(n+1)} = u^{(n)} + \hat{\lambda}_n q^{(n)}, \qquad \hat{\lambda}_n = \frac{(\tilde{q}^{(n)}, r^{(n)})}{(\tilde{q}^{(n)}, Aq^{(n)})},$$

$$r^{(n+1)} = r^{(n)} - \hat{\lambda}_n Aq^{(n)}, \qquad \tilde{r}^{(n+1)} = \tilde{r}^{(n)} - \hat{\lambda}_n^* A^*\tilde{q}^{(n)}.$$

LANCZOS/ORTHOMIN ALGORITHM

$$p^{(0)} = r^{(0)}; \qquad p^{(n)} = r^{(n)} + \alpha_n p^{(n-1)}, \quad n > 0,$$

$$\tilde{p}^{(0)} = \tilde{r}^{(0)}; \qquad \tilde{p}^{(n)} = \tilde{r}^{(n)} + \alpha_n^* \tilde{p}^{(n-1)}, \quad n > 0,$$

$$\alpha_n = \frac{(\tilde{r}^{(n)}, r^{(n)})}{(\tilde{r}^{(n-1)}, r^{(n-1)})},$$

$$u^{(n+1)} = u^{(n)} + \lambda_n p^{(n)}, \qquad \lambda_n = \frac{(\tilde{r}^{(n)}, r^{(n)})}{(\tilde{p}^{(n)}, Ap^{(n)})},$$

$$r^{(n+1)} = r^{(n)} - \lambda_n A p^{(n)}, \qquad \tilde{r}^{(n+1)} = \tilde{r}^{(n)} - \lambda_n^* A^* \tilde{p}^{(n)}.$$

LANCZOS/ORTHORES ALGORITHM

$$u^{(n+1)} = \lambda_n (r^{(n)} + \sigma_{n+1,n} u^{(n)} + \sigma_{n+1,n-1} u^{(n-1)}),$$

$$r^{(n+1)} = -\lambda_n (Ar^{(n)} - \sigma_{n+1,n} r^{(n)} - \sigma_{n+1,n-1} r^{(n-1)}),$$

$$\tilde{r}^{(n+1)} = -\lambda_n^* (A^* \tilde{r}^{(n)} - \sigma_{n+1,n}^* \tilde{r}^{(n)} - \sigma_{n+1,n-1}^* \tilde{r}^{(n-1)}),$$

$$\lambda_n = [\sigma_{n+1,n} + \sigma_{n+1,n-1}]^{-1},$$

$$\sigma_{n+1,n} = \frac{(\tilde{r}^{(n)}, Ar^{(n)})}{(\tilde{r}^{(n)}, r^{(n)})}, \qquad \sigma_{n+1,n-1} = -\frac{1}{\lambda_{n-1}} \frac{(\tilde{r}^{(n)}, r^{(n)})}{(\tilde{r}^{(n-1)}, r^{(n-1)})}.$$

We will say that an algorithm *breaks down* at a step n if $u^{(n-1)} \neq u$ has been successfully computed by the algorithm but $u^{(n)}$ cannot be computed by the algorithm due to some condition such as division by zero.

In the absence of breakdown of the given algorithm, the above three algorithms are guaranteed to yield the iterates defined by (2), and furthermore, if the algorithm does not break down we have exact convergence $u^{(n)} = u$ if and only if $n = d(r^{(0)}, A)$. Here we define the *degree of a vector* v by

$$d(v, A) = \max\{d : \{A^i v\}_{i=0}^{d-1} \text{ linearly independent}\}.$$

When A is diagonalizable, the quantity $d(v, A)$ is the number of eigenvectors of A represented in v.

As noted above, for certain choices of A and $\tilde{Z}$ the method (2) reduces to a standard CG method. In such cases the above three algorithms above reduce to standard CG algorithms. In these cases, breakdown is known to be impossible (see [2]). However, in more general situations, the above three algorithms may indeed break down; for examples of this, see [24].

3. Breakdown of Lanczos algorithms. In this section we give conditions that characterize the situations in which each of the above three Lanczos algorithms experience breakdown.

For theoretical reasons, it is desirable to find characterizations of the conditions of breakdown of the algorithms that are based on the key *spaces* $\mathbf{K}_n(r^{(0)}, A)$ and $\mathbf{K}_n(\tilde{r}^{(0)}, A^*)$ rather than the *formulas* for the algorithms. In particular, we will characterize breakdown of the three Lanczos algorithms in terms of the *moment matrices*

$K_n(\tilde{r}^{(0)}, A^*)^* K_n(r^{(0)}, A)$ and $K_n(\tilde{r}^{(0)}, A^*)^* A K_n(r^{(0)}, A)$. Here we define the matrix $K_n(v, A) = [v \quad Av \quad \cdots \quad A^{n-1}v]$, a matrix whose columns span the Krylov space $\mathbf{K}_n(v, A)$.

The following three theorems give exact conditions for breakdown of the above algorithms. Detailed proofs may be found in [24]. A result similar to Theorem 2 is found in [11]; see also [41].

THEOREM 1 (Lanczos/Orthodir Breakdown). *Suppose Lanczos/Orthodir has successfully generated $u^{(n-1)} \neq u$. Then the following are equivalent:*

- *The algorithm does not break down at step n.*
- *The matrix $K_n(\tilde{r}^{(0)}, A^*)^* A K_n(r^{(0)}, A)$ is nonsingular.*
- *There exists a unique iterate $u^{(n)}$ satisfying (2).*

THEOREM 2 (Lanczos/Orthomin Breakdown). *Suppose Lanczos/Orthomin has successfully generated $u^{(n-1)} \neq u$. Then the following are equivalent:*

- *The algorithm breaks down at step n.*
- *Either $K_{n-1}(\tilde{r}^{(0)}, A^*)^* K_{n-1}(r^{(0)}, A)$ or $K_n(\tilde{r}^{(0)}, A^*)^* A K_n(r^{(0)}, A)$ is singular.*

THEOREM 3 (Lanczos/Orthores Breakdown). *Suppose Lanczos/Orthores has successfully generated $u^{(n-1)} \neq u$. Then the following are equivalent:*

- *The algorithm breaks down at step n.*
- *Either $K_n(\tilde{r}^{(0)}, A^*)^* K_n(r^{(0)}, A)$ or $K_n(\tilde{r}^{(0)}, A^*)^* A K_n(r^{(0)}, A)$ is singular.*

Thus we see that the Orthodir variant of Lanczos breaks down at a step precisely when (2) fails to define a unique iterate, or equivalently when the moment matrix $K_n(\tilde{r}^{(0)}, A^*)^* A K_n(r^{(0)}, A)$ is singular for these choices of n. On the other hand, the Orthomin and Orthores variants break down under further circumstances involving the moment matrix $K_n(\tilde{r}^{(0)}, A^*)^* K_n(r^{(0)}, A)$. Thus the Orthomin variant breaks down if and only if the Orthores variant breaks down, and if neither algorithm breaks down then the Orthodir variant does not break down.

These observations lead us to make the following definitions. We say that a (*hard*) *breakdown* of the Lanczos method (2) occurs at a step $n \leq d(r^{(0)}, A)$ if the moment matrix $K_n(\tilde{r}^{(0)}, A^*)^* A K_n(r^{(0)}, A)$ is singular. In this case, $u^{(n)}$ defined by (2) does not exist uniquely, so that *no* algorithm can be used to compute this iterate. On the other hand, we say that *soft breakdown* of the Lanczos method occurs at step n if the moment matrix $K_n(\tilde{r}^{(0)}, A^*)^* K_n(r^{(0)}, A)$ is singular. This condition causes failure of the Orthomin and Orthores variants but not the Orthodir variant.

Importantly, the conditions of hard and soft breakdown are conditions associated with the method (2), irrespective of the particular algorithms used to implement the method. Hard breakdown is a serious problem, a failure of the method defined by (2). On the other hand, soft breakdown is a condition that poses a problem only for certain algorithms but is not an intrinsic problem for the method (2), since some algorithms (e.g., Lanczos/Orthodir) may still be used to compute the iterates.

The Orthomin variant of Lanczos is generally preferable to the other variants, due to its economy and relative numerical stability. On the other hand, its vulnerability to the problem of soft breakdown may be remedied in theory by a temporary switch to the Orthodir variant in the event of a soft breakdown. This will be discussed further in §7.

In what follows we will consider some of the theoretical aspects of hard and soft breakdown of the Lanczos method.

4. Basic results on the likelihood of breakdown. One fundamental question to ask about the Lanczos algorithms is: how likely is an occurrence of breakdown for a given choice of A and $\tilde{Z}$? We recall that the matrix $\tilde{Z}$ defines the relationship

$\tilde{r}^{(0)} = \tilde{Z}^* r^{(0)}$.

To answer this question, we will use results from measure theory (see, e.g., [39] and [38]). Specifically, we define a field $\mathbb{K}$ to be either the reals $\mathbb{R}$ or the complex numbers $\mathbb{C}$, and for $A, \tilde{Z} \in \mathbb{K}^{N \times N}$ we ask: what is the measure of the set of vectors $r^{(0)} \in \mathbb{K}^N$ that cause hard or soft breakdown?

The following sequence of results begins to provide an answer to the question of the likelihood of breakdown. We show that in many cases, the set of initial residuals $r^{(0)}$ causing breakdown is only of zero measure, while in a few cases of A and $\tilde{Z}$ breakdown occurs for almost every vector $r^{(0)}$. It is desirable that the set of $r^{(0)}$ causing breakdown be measure zero, since this indicates that an initial guess vector $u^{(0)}$ chosen randomly has zero probability of causing breakdown in exact arithmetic.

To prove these measure-zero results, we begin with the following result on the measure of the set of zeros of a polynomial in several variables.

PROPOSITION 4 (Zero Sets of Polynomials). *Let $\mathbb{K} = \mathbb{R}$ or $\mathbb{C}$. If P is a complex nonzero polynomial in the variables $x_1, x_2, \cdots, x_N \in \mathbb{K}$, then $P(x) \neq 0$ for almost every $x = (x_1, x_2, \cdots, x_N) \in \mathbb{K}^N$.*

Proof. If $\mathbb{K} = \mathbb{R}$ and P is nonzero, then either $\operatorname{Re} P(z)$ or $\operatorname{Im} P(z)$ is a nonzero (real) polynomial; if $\mathbb{K} = \mathbb{C}$, we may decompose each x_i into real and imaginary parts, giving $2N$ variables, and consider the real polynomial $P(x)^* P(x)$. In any case, we may assume without loss of generality that P is a nonzero real polynomial of real variables.

We know that for any point x, the polynomial P is the zero polynomial if and only if the polynomial and all its derivatives are zero at x. Let V_0 denote the set of zeros of P in $\mathbb{R}^N$. Suppose the set V_0 has nonzero measure. We know from integration theory (see, for example, [44, pp. 128f]) that almost every point of V_0 is a point of density in each of the N coordinate directions. We recall that $x \in \mathbb{R}$ is a point of density of a measurable subset $S \subseteq \mathbb{R}$ if for any sequence of intervals I_n such that $x \in I_n$ with measure $m(I_n) \to 0$ we have $m(S \cap I_n)/m(I_n) \to 1$.

It is easily seen that at such points in V_0, the first partial derivatives of P must necessarily be zero. Let V_1 be the points of V_0 where all first derivatives are also zero. We have just shown that V_0 and V_1 both have the same nonzero measure. The argument may be repeated for V_1 to show all second partial derivatives of f are zero at almost every point of V_0, and so forth, resulting in the fact that P and all its derivatives are zero on a set which has nonzero measure. The proof is completed by selecting any one of these points. □

This result may be immediately applied to Lanczos moment matrices, by using the fact that the determinant of a matrix is a polynomial in the elements of the matrix.

COROLLARY 5 (Lanczos Moment Matrices). *Let $\mathbb{K} = \mathbb{R}$ or $\mathbb{C}$ and $\tilde{Z}$, $A \in \mathbb{C}^{N \times N}$. For some integer m let $f(r) = \det[K_n(\tilde{Z}^* r, A^*)^* A^m K_n(r, A)]$. Then either $f(r)$ is zero for all $r \in \mathbb{K}^N$ or it is zero only on a measure-zero set of vectors r in $\mathbb{K}^N$.*

Proof. If $\mathbb{K} = \mathbb{R}$, then $f(r)$ is a polynomial in the N variables $r_i \equiv e_i^* r$, where e_i represents the standard unit basis vector. If $\mathbb{K} = \mathbb{C}$, then $f(r)$ is a polynomial in the $2N$ variables $\operatorname{Re} r_i$ and $\operatorname{Im} r_i$. □

From this we conclude that the relevant Lanczos moment matrices are singular either for every $r^{(0)}$ or only for a measure-zero set of vectors $r^{(0)}$.

In order to proceed, we must show some results concerning the degrees of vectors with respect to the matrix A. We define the *degree of a matrix* $d(A) = \min\{\deg(P) : P \text{ monic}, P(A) = 0\}$. We note that for every v we have $0 \leq d(v, A) \leq d(A) \leq N$. We show here that almost every vector v satisfies $d(v, A) = d(A)$.

PROPOSITION 6 (Degrees of Vectors). *Let* $\mathbb{K} = \mathbb{R}$ *or* $\mathbb{C}$ *and* $A \in \mathbb{K}^{N\times N}$. *Then almost every* $v \in \mathbb{K}^N$ *satisfies* $d(v, A) = d(A)$.

Proof. The result follows from the fact that almost every vector necessarily contains a nonzero component of each of the generalized eigenvectors of A. The details are found in [24]. □

We noted earlier that breakdown is equivalent to the singularity of an appropriate moment matrix *only* when the iteration number n satisfies $n \leq d(r^{(0)}, A)$. Due to this technical point, in order to prove the desired results we define notation for the set of vectors $r^{(0)}$ for which this condition is satisfied. We define the set $\mathcal{T}_n(A) = \{v \in \mathbb{C}^N : n \leq d(v, A)\}$. The set $\mathcal{T}_n(A)$ is the set of initial residuals $r^{(0)}$ for which the singularity/nonsingularity of the moment matrices is relevant to the question of breakdown at step n.

We established in Proposition 6 that $\mathcal{T}_{d(A)}(A) \cap \mathbb{K}^N$ contains almost every vector in $\mathbb{K}^N$. It will be noted that $\mathcal{T}_n(A)$ is monotone, in the sense that $m \leq n$ implies $\mathcal{T}_m(A) \supseteq \mathcal{T}_n(A)$. Thus for any $n \leq d(A)$, $\mathcal{T}_n(A) \cap \mathbb{K}^N$ necessarily contains almost every vector in $\mathbb{K}^N$.

We now present the major theorem, which gives the three basic possibilities for breakdown of Lanczos algorithms. The upshot of this result is that for a given iteration number $n \leq d(A)$, for the set of vectors $r^{(0)}$ for which $d(r^{(0)}, A) \geq n$ (which amounts to almost every vector), either breakdown is impossible, breakdown always occurs, or breakdown occurs only for a nonempty measure-zero set of vectors.

THEOREM 7 (Lanczos Breakdown, Iterate n). *Let* $\mathbb{K} = \mathbb{R}$ *or* $\mathbb{C}$, $A, \tilde{Z} \in \mathbb{K}^{N\times N}$, *and* $n \leq d(A)$. *Then exactly one of the following three conditions holds for the Lanczos method with* $\tilde{r}^{(0)} = \tilde{Z}^* r^{(0)}$.

(i) *Hard breakdown at step* n *occurs for every vector* $r^{(0)} \in \mathcal{T}_n(A) \cap \mathbb{K}^N$ (*and thus at least for almost every* $r^{(0)} \in \mathbb{K}^N$).

(ii) *Hard breakdown at step* n *occurs for a nonempty measure-zero set of vectors* $r^{(0)} \in \mathcal{T}_n(A) \cap \mathbb{K}^N$ (*and thus a nonempty measure-zero set of vectors in* $\mathbb{K}^N$).

(iii) *Hard breakdown at step* n *occurs for no vectors* $r^{(0)} \in \mathcal{T}_n(A) \cap \mathbb{K}^N$ (*and thus for at most a measure-zero set of vectors in* $\mathbb{K}^N$).

Furthermore, the same result holds if "hard breakdown" is replaced by "soft breakdown" in the statement of this theorem.

Proof. For vectors $r^{(0)} \in \mathcal{T}_n(A) \cap \mathbb{K}^N$, breakdown is equivalent to singularity of an appropriate moment matrix. The set $\mathcal{T}_n(A) \cap \mathbb{K}^N$ amounts to almost every vector in $\mathbb{K}^N$. Now, by Corollary 5, the set S_n of vectors in $\mathbb{K}^N$ for which the moment matrix of dimension n is singular is either the set of all vectors or a subset of measure zero. If the moment matrix is singular for every vector (i.e., $S_n = \mathbb{K}^N$), then it is singular for every vector in $\mathcal{T}_n(A) \cap \mathbb{K}^N$, giving case (i) above. Otherwise the set S_n is measure zero in $\mathbb{K}^N$. Thus $\mathcal{B}_n \equiv S_n \cap (\mathcal{T}_n(A) \cap \mathbb{K}^N)$ is of measure zero and is either empty or nonempty. □

5. Measure-zero results. When the Lanczos method may be reduced to a standard conjugate gradient method, it is known that breakdown at any step is impossible. In the general case, we would like to show at least that breakdown occurs for no more than a measure-zero set of vectors. We will show below that this is true in many but not all cases.

We begin by considering the simple case of $n = N = d(A)$. We may then show a more general result by projecting the problem to this simpler case. Similar results are shown in [40] and [50].

THEOREM 8 (Measure-Zero Breakdown, Case $n = N = d(A)$). *Let* $\mathbb{K} = \mathbb{R}$ *or*

$\mathbb{C}$, *and let* A, $\tilde{Z} \in \mathbb{K}^{N\times N}$ *with* A *and* $\tilde{Z}$ *nonsingular and* $d(A) = N$, *and let* m *be some integer. Then for* $n = N$ *and almost every* $r \in \mathbb{K}^N$, $K_n(\tilde{Z}^*r, A^*)^*A^mK_n(r, A)$ *is nonsingular.*

Proof. Let $S = \{r \in \mathbb{K}^N : d(r, A) = N\}$ and $S' = \{\tilde{r} \in \mathbb{K}^N : d(\tilde{r}, A^*) = N\}$. By Proposition 6, S and S' each contain almost every vector in $\mathbb{K}^N$. Furthermore, for every $r \in S$, $K_n(r, A)$ is a *square* full-rank matrix, and thus nonsingular. Also, for every $\tilde{Z}^*r \in S'$ (i.e., $r \in \tilde{Z}^{-*}S'$), $K_n(\tilde{Z}^*r, A^*)$ is a square nonsingular matrix. The proof is completed by noting that $S \cap \tilde{Z}^{-*}S'$ contains almost every vector in $\mathbb{K}^N$ and that the product of two square nonsingular matrices is nonsingular. □

We now prove the result for a more general situation including nearly all iteration numbers $n \leq d(A)$. The following result applies to a wide range of choices of $\tilde{Z}$, including, for example, the common choice $\tilde{Z} = I$. The idea of the proof is to reduce the general problem back to the case of $n = N = d(A)$ by projecting onto an appropriate A-invariant Krylov subspace. The range of possible values of n covered by this theorem will be discussed below.

We define here a *definite* matrix to be a matrix B satisfying $v^*Bv \neq 0$ for every nonzero v. A real matrix B is definite if and only if either its Hermitian part $H(B) \equiv (B + B^*)/2$ or its negative $-H(B)$ is HPD (see [26]).

THEOREM 9 (Measure-Zero Breakdown, Iteration n). *For* $\mathbb{K} = \mathbb{R}$ *or* $\mathbb{C}$, *let* A, $\tilde{Z} \in \mathbb{K}^{N\times N}$, *and let* $\tilde{Z}$ *satisfy the condition that for some polynomials* F *and* G *over* $\mathbb{K}$ *and for some definite matrix* $H \in \mathbb{K}^{N\times N}$, $\tilde{Z} = F(A)^*HG(A)$ *is nonsingular. Let* m *be arbitrary and let* $n \leq d(A)$ *be such that there exists* $v \in \mathbb{K}^N$ *satisfying* $d(v, A) = n$. *Then there exist* $r \in \mathbb{K}^N$ *and* $\tilde{r} = \tilde{Z}^*r$ *such that* $K_n(\tilde{r}, A^*)^*A^mK_n(r, A)$ *is nonsingular.*

Proof. We have $\tilde{Z} = [F(A)]^*HG(A)$ nonsingular with $H \in \mathbb{K}^{N\times N}$ definite and polynomials F and G over $\mathbb{K}$. We seek to find $r \in \mathbb{K}^N$ such that

$$K_n([G(A)]^*H^*F(A)r, A^*)^*A^mK_n(r, A)$$

is nonsingular.

By hypothesis, there exists v such that $d(v, A) = n$. Let Q be the ℓ^2-orthogonal projector onto the A-invariant subspace $\mathbf{K}_n(v, A)$ of $\mathbb{K}^N$. We note that $Q \in \mathbb{K}^{N\times N}$ is Hermitian and $AQ = QAQ$. Then it is sufficient to find $r \in \operatorname{Range} Q$ such that

$$K_n([G(\check{A})]^*\check{H}^*F(\check{A})r, \check{A}^*)^*\check{A}^mK_n(r, \check{A})$$

is nonsingular, where $\check{A} = QAQ$ and $\check{H} = QHQ$.

Without loss of generality we may assume $r \in \mathbb{K}^n$ and $\check{A}$, $\check{H} \in \mathbb{K}^{n\times n}$, by restriction of the operators to $\operatorname{Range} Q$. Importantly, since H is definite, the restricted matrix $\check{H}$ is definite, thus nonsingular. Furthermore, since the spectrum of $\check{A}$ is contained in the spectrum of A, we have that $F(\check{A})$ and $G(\check{A})$ are nonsingular. Therefore, by applying Theorem 8 to this restricted problem, we have the desired result. □

It remains to be determined which iteration numbers n the above result may be applied to. In the case of complex spaces, an A-invariant (complex) Krylov subspace of *any* size $n \leq d(A)$ exists. On the other hand, if A is real and subspaces over the reals are sought, then some values of n may be excluded. In particular, if A has no real eigenvalues, then for every real r, the value of $d(r, A)$ must be even. This is true because such r must necessarily contain matched pairs of complex conjugate generalized eigenvectors in order to be a real vector.

These observations are made precise by the following proposition.

PROPOSITION 10 (Vectors of a Given Degree). *Let* $\mathbb{K} = \mathbb{R}$ *or* $\mathbb{C}$, $A \in \mathbb{K}^{N\times N}$, *and* $0 \le n \le d(A)$. *Then there does not exist a vector* $r \in \mathbb{K}^N$ *satisfying* $d(r, A) = n$, *if and only if all of the following conditions hold:* $\mathbb{K} = \mathbb{R}$, n *is odd, and* A *has no real eigenvalues.*

Proof. For a detailed proof, see [24]. □

To summarize, we see that for the common choice of $\tilde{Z} = I$, breakdown cannot occur for any of the three Lanczos algorithms except for a set of vectors $r^{(0)}$ which has zero measure in $\mathbb{C}^N$. On the other hand, when A is real, in order to guarantee the same result for vectors in $\mathbb{R}^N$, it is sufficient that A have at least one real eigenvalue.

Though this condition may not be necessary, nonetheless some restriction is necessary in order to limit breakdown to a measure-zero set of real vectors, as the following example shows. Let

$$A = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 3 & -1 \\ 0 & 0 & 1 & 3 \end{bmatrix}$$

and let $\tilde{r} = r$. We note that A is a normal matrix and $d(A) = 4$. After some algebra it may be shown that for $n = 3$ and for *every* real r, $K_n(\tilde{r}, A^*)^* K_n(r, A)$ is singular. As a practical consequence, the standard BCG algorithm (Lanczos/Orthomin, $\tilde{Z} = I$) necessarily breaks down at step 4 for every real $r^{(0)}$ satisfying $d(r^{(0)}, A) \ge 3$, that is, for almost every real vector $r^{(0)}$.

This result contradicts a result of [50, p. 278]. It also sheds light on the comments of [22, p. 21], [49, p. 389], and [4, p. 328] on selecting an alternate pair of initial vectors r, $\tilde{r}$ in the case of breakdown. In particular, this counterexample shows that for some cases of A and $\tilde{Z}$, there is no practical way to apply the standard Lanczos algorithms in real arithmetic if the initial vectors r and $\tilde{r}$ are forced to satisfy the relationship $\tilde{r} = \tilde{Z}^* r$. Thus, in some cases it is necessary to choose r and $\tilde{r}$ as arbitrary *independent* vectors, or else they must be chosen complex. This is also in agreement with the comments of [7], where it is suggested that complex vectors be used, since in general the invariant subspaces for A and A^* are complex.

We conclude that for a certain small class of matrix problems, the standard BCG algorithm cannot be used successfully. On the other hand, since the problem of hard or soft breakdown for almost every real $r^{(0)}$ can only occur for *odd* values of n, a look-ahead procedure could be used to skip over these steps. This will be described further in §7.

6. Curable and incurable breakdown. We now analyze hard and soft breakdown of the Lanczos method from an alternate viewpoint. In the previous sections we considered the behavior of the Lanczos method at a given step n as the initial residual $r^{(0)}$ was allowed to range over all possible values in $\mathbb{C}^N$ or $\mathbb{R}^N$. In this section we will instead consider the behavior of Lanczos for a *fixed* value of $r^{(0)}$, for various values of n.

If hard (soft) breakdown of the Lanczos method occurs at step n, then we say that it is an *incurable hard* (*soft*) *breakdown* if for every step m satisfying $n < m \le d(r^{(0)}, A)$, hard (soft) breakdown also occurs at step m. A hard (soft) breakdown that is not incurable is said to be *curable.* These concepts are an adaptation of the definitions given in [46] and [37] regarding the nonsymmetric Lanczos method for solving the eigenvalue problem.

As mentioned earlier, an occurrence of soft breakdown, whether curable or incurable, could be remedied by a temporary switch to the Orthodir variant of Lanczos. On

the other hand, an instance of curable hard breakdown could in theory be remedied by an algorithm that skipped over the steps for which hard breakdown occurred. This idea of a look-ahead Lanczos algorithm is developed for the eigenvalue problem in [37]. Look-ahead techniques for Lanczos and other methods constitute a current area of research; see, for example, [18]–[20], [3], [35], [13]–[15], and [28].

Unfortunately, a curable hard breakdown may occur for as many as $N-1$ steps, where N is the size of A. For example, we let

$$A = e_N e_1^* + \sum_{i=1}^{N-1} e_i e_{i+1}^*,$$

a permutation matrix, and we let $r^{(0)} = \tilde{r}^{(0)} = e_N$. Here, e_i represents the standard unit basis vector. Then

$$K_N(\tilde{r}^{(0)}, A^*)^* A K_N(r^{(0)}, A) = \sum_{i=1}^{N} e_i e_{N-i+1}^*.$$

Thus the Lanczos method with $\tilde{Z} = I$ experiences hard breakdown at every step $n < N$ but not at step $n = N$. In this case, a look-ahead algorithm would have to perform a look-ahead over a prohibitively large number of steps.

We now consider *incurable* hard and soft breakdown. The phenomena of incurable hard and soft breakdown have straightforward characterizations. The following result is similar to the Mismatch Theorem of [46].

THEOREM 11 (Mismatch Theorem). *Suppose that for a particular choice of initial vectors r and $\tilde{r}$ and for a given m and for a given value of n satisfying $n < d(r, A)$, we have that either $n = 0$ or $K_n(\tilde{r}, A^*)^* A^m K_n(r, A)$ is nonsingular. Then the matrix $K_{n+k}(\tilde{r}, A^*)^* A^m K_{n+k}(r, A)$ is singular for every k satisfying $n < n + k \leq d(r, A)$ if and only if there exist vectors p and $\tilde{p}$ and integers $d > 0$ and $d' \geq 0$ such that $d(r, A) = n + d$, $d(\tilde{r}, A^*) = n + d'$, and*

$$\mathbf{K}_{n+d}(r, A) = \text{Range}[K_n(r, A) \quad K_d(p, A)],$$

$$\mathbf{K}_{n+d'}(\tilde{r}, A^*) = \text{Range}[K_n(\tilde{r}, A^*) \quad K_{d'}(\tilde{p}, A^*)],$$

$$K_{d'}(\tilde{p}, A^*)^* A^m K_{n+d}(r, A) = 0, \qquad K_{n+d'}(\tilde{r}, A^*)^* A^m K_d(p, A) = 0,$$

where $[K_n(r, A) \quad K_d(p, A)]$ and $[K_n(\tilde{r}, A^*) \quad K_{d'}(\tilde{p}, A^*)]$ represent block matrices.

Proof. See [24]. □

From this result we have the following consequences.

COROLLARY 12 (Breakdown Equivalence). *For a particular run of Lanczos, incurable hard breakdown occurs at step n if and only if incurable soft breakdown occurs at step n.*

Proof. First, we suppose incurable hard breakdown occurs at step n. Let $m + 1 \leq n$ indicate the first step at which the incurable hard breakdown has occurred. Using the characterization from Theorem 11, since $\mathbf{K}_{m+d}(r, A) = A\mathbf{K}_{m+d}(r, A)$ and $\mathbf{K}_{m+d'}(\tilde{r}, A^*) = A^*\mathbf{K}_{m+d'}(\tilde{r}, A^*)$, we see that incurable soft breakdown also occurs at step $m + 1$, and thus at step n. The opposite implication is proved analogously. □

COROLLARY 13 (Restarting of Lanczos). *Suppose that for a run of the Lanczos method breakdown does not occur at step $n \geq 0$ but incurable breakdown occurs at step $n + 1$. Then $d(r^{(n)}, A) = d(r^{(0)}, A) - n$. Thus the Lanczos method applied to initial*

residual $r^{(0)\prime} = r^{(n)}$ *with any auxiliary residual vector* $\tilde{r}^{(0)\prime}$ *is guaranteed to converge (if breakdown does not recur) within* $d(r^{(0)}, A) - n$ *steps.*

Proof. We first show that $d(r^{(n)}, A) \leq d(r^{(0)}, A) - n$. Using (2) and the characterization of Theorem 11, we have $r^{(n)} - r^{(0)} \in A\mathbf{K}_n(r^{(0)}, A)$ so that $r^{(n)} \in \mathbf{K}_{n+d}(r^{(0)}, A)$, i.e., $r^{(n)} = [K_n(r^{(0)}, A)\ K_d(p, A)]\beta$ for some β. Furthermore, $r^{(n)} \perp \mathbf{K}_n(\tilde{r}^{(0)}, A^*)$, so that $r^{(n)} \in \mathbf{K}_d(p, A)$, since $K_n(\tilde{r}^{(0)}, A^*)^* K_n(r^{(0)}, A)$ is nonsingular and $K_n(\tilde{r}^{(0)}, A^*)^* K_d(p, A)$ is zero. We also note that $\mathbf{K}_d(p, A)$ is A-invariant: $v \in \mathbf{K}_d(p, A)$ implies $Av \in \mathbf{K}_{n+d}(r^{(0)}, A)$. But $(w, Av) = (A^*w, v) = 0$ for all $w \in \mathbf{K}_{n+d'}(\tilde{r}^{(0)}, A^*)$. Thus $Av \perp \mathbf{K}_{n+d'}(\tilde{r}^{(0)}, A^*)$, implying $Av \in \mathbf{K}_d(r^{(0)}, A)$. Thus it is necessarily true that $d(r^{(n)}, A) \leq d$.

We now show that $d(r^{(n)}, A) \geq d(r^{(0)}, A) - n$. We have that $r^{(n)} = P(A)r^{(0)}$ where $P(0) = 1$ and $\deg P \leq n$. Now, let $\tilde{P}$ be the minimal polynomial for $r^{(n)}$. Then $0 = \tilde{P}(A)r^{(n)} = \tilde{P}(A)P(A)r^{(0)}$, from which we obtain $\deg \tilde{P} + n \geq \deg \tilde{P} + \deg P = \deg(\tilde{P} \cdot P) \geq d(r^{(0)}, A)$, so that $\deg P = n$ and $d(r^{(n)}, A) = d(r^{(0)}, A) - n$. □

This result suggests that the restarting of the Lanczos method is a possible remedy for the problem of incurable breakdown. It is not clear, however, how restarting affects the number of iterations required to satisfy some convergence criterion such as $||r^{(n)}||/||r^{(0)}|| < \zeta$. Restarting of the Lanczos method is a technique that was investigated experimentally in [31] and also mentioned in [47]. This technique will be considered in more detail below.

7. Modified Lanczos algorithms. The theoretical results given above for the Lanczos method may be applied to the development of modified Lanczos algorithms that are better able to deal with the problem of breakdown or near-breakdown. In [24] the algorithm BCGNB is defined. It embodies three particular strategies for remedying the breakdown problem of the standard BCG algorithm, which uses $\tilde{Z} = I$:

1. In the case of a near-soft-breakdown, a switch to the Orthodir variant is made for the step for which soft breakdown is a problem. A similar idea was used in [5] to develop a hybrid Orthodir/Orthomin conjugate residual algorithm for the case of A symmetric indefinite. Such a hybrid algorithm may be easily developed for the Lanczos method, based on the observation that the vector $p^{(n)}$ (alternatively, $\tilde{p}^{(n)}$) of the Orthomin variant of Lanczos is a scalar multiple of the vector $q^{(n)}$ (alternatively, $\tilde{q}^{(n)}$) of the Orthodir variant, whenever these quantities are well defined.

2. In the case of near-hard-breakdown for up to $s - 1$ steps, where s is a parameter supplied to the algorithm, a look-ahead procedure similar to that of [37] is used to skip over those steps.

3. For near-hard-breakdown for more than $s - 1$ steps, a restart of the algorithm is performed, based on the last iterate for which near-hard-breakdown was not indicated. This is done in the hope that this is a case of an incurable breakdown.

Derivations of the formulas for the BCGNB algorithm are given in [24]. The implementation of this algorithm requires the development of adequate tests for near-hard-breakdown and near-soft-breakdown that account for the difficulties of finite precision arithmetic. Tests are defined and experimental results for the choices of tolerances for the tests are given in [24].

Experimental evidence indicates that the most effective of the three remedies listed above is the technique of restarting. Due to its simplicity, we will state here the criterion used to determine whether to restart. In particular, using the notation for the Lanczos/Orthomin algorithm given in §2, we say that if the criterion

$$\frac{|(\tilde{p}^{(n)}, Ap^{(n)})|}{||\tilde{p}^{(n)}|| \cdot ||Ap^{(n)}||} < \epsilon_3$$

is satisfied, then a restart of the algorithm will be performed, using the new initial residual $r^{(0)\prime} = r^{(n)}$ and the new auxiliary initial residual $\tilde{r}^{(0)\prime} = \tilde{Z}^* r^{(0)\prime}$ with $\tilde{Z} = I$. A simple restarted BCG algorithm may be implemented based solely on this technique. The particular choice of $\epsilon_3 = \epsilon_M^{1/2}$ has proven to be useful for a number of problems where ϵ_M is unit roundoff error, the smallest floating point number such that $1+\epsilon_M > 1$. This particular criterion for testing for near-hard-breakdown has the advantage of limiting the size of the growth of $||r^{(n)}||$ over the course of the run.

Another algorithm that suffers from the same breakdown problems as the Lanczos algorithm and can be remedied by similar techniques is the conjugate gradient squared (CGS) algorithm of [45]. The CGS algorithm is defined as follows:

CGS Algorithm

$$p_2^{(0)} = f^{(0)} = r^{(0)},$$

$$f^{(n)} = r^{(n)} + \alpha_n h^{(n)}, \qquad p_2^{(n)} = f^{(n)} + \alpha_n (h^{(n)} + \alpha_n p_2^{(n-1)}),$$

$$h^{(n+1)} = f^{(n)} - \lambda_n A p_2^{(n)},$$

$$u^{(n+1)} = u^{(n)} + \lambda_n (f^{(n)} + h^{(n+1)}), \qquad r^{(n+1)} = r^{(n)} - \lambda_n A (f^{(n)} + h^{(n+1)}),$$

$$\lambda_n = \frac{(\tilde{r}^{(0)}, r^{(n)})}{(\tilde{r}^{(0)}, A p_2^{(n)})}, \qquad \alpha_{n+1} = \frac{(\tilde{r}^{(0)}, r^{(n+1)})}{(\tilde{r}^{(0)}, r^{(n)})}.$$

This algorithm experiences breakdown in precisely the same instances as the BCG algorithm, in exact arithmetic. The CGS algorithm commonly requires roughly the same computational work per iteration as BCG, and half the number of iterations; however, the numerical effects of near-hard-breakdown are frequently more severe (see, e.g., [48]).

A restarted CGS algorithm, CGSNB, is defined in [24]. The criterion used to test for near-breakdown is

$$\frac{|(\tilde{r}^{(0)}, A p_2^{(n)})|}{||\tilde{r}^{(0)}|| \cdot ||A p_2^{(n)}||} \leq \epsilon_h.$$

In the runs presented below, we use the setting $\epsilon_h = 10\epsilon_M^{1/2}$.

We now describe one further technique for mitigating the problem of breakdown. The Mismatch Theorem [46], [24] indicates that incurable breakdown is caused by irregular left- and right-eigenvector distributions in $r^{(0)}$ and $\tilde{r}^{(0)}$. To remedy this problem, a randomized vector may be used for $r^{(0)}$. This may easily be done by setting the initial guess $u^{(0)}$ to be a vector of random entries of an appropriate size. To do this, we let v be a vector whose elements are random numbers uniformly distributed on $[-1, 1]$, and we set $u^{(0)}$ to be a multiple of v scaled so that $||Au^{(0)}|| = ||b||$. Experiments using this technique will be given below. Similarly, if a choice of $u^{(0)}$ is already known that is near the true solution, then the given vector may be perturbed by a small random vector in order to give a new randomized choice of $u^{(0)}$. Experimental results from using random vectors with Lanczos algorithms are also reported in [12].

8. Numerical results. In this section we present numerical results with the algorithms described in this paper. We are primarily concerned with the algorithms BCG, BCGNB, CGS, and CGSNB. For comparison, we will also consider the full

TABLE 1
Model problem, $h^{-1} = 128$, ITMAX=3000. *Number of iterations.*

Method \ Dh:	0	2^{-3}	2^{-2}	2^{-1}	2^{0}	2^{1}	2^{2}	2^{3}	2^{4}	2^{5}
GMRES(∞)	290	269	245	220	200	189	186	189	207	249
BCG	308	341	299	1518	–	–	–	–	533	–
BCG, random $u^{(0)}$	309	354	300	310	313	301	299	302	290	293
BCGNB	308	353	284	338	253	240	243	240	302	962
CGS	272	254	222	–	–	–	–	–	–	–
CGS, random $u^{(0)}$	193	189	200	192	193	175	225	212	216	197
CGSNB	272	284	212	196	151	162	158	173	156	256

GMRES algorithm GMRES(∞) [43], the restarted algorithm GMRES(k), and the normal equations algorithm LSQR [33].

The GMRES(∞) algorithm is of particular interest, since it gives iteration counts that are minimal among all polynomial methods, in the sense that $||r^{(n)}||$ is minimized with respect to all possible polynomial methods in A. However, the method is generally too expensive to be practical.

Unless otherwise stated, we make the following assumptions. For the test runs we utilize the initial guess vector of $u^{(0)} = 0$. For the sake of simplicity, we use the simple stopping test

$$\frac{||r^{(n)}||}{||b||} < \zeta = 10^{-6}.$$

Also, we use the basic choice $\tilde{r}^{(0)} = r^{(0)}$, i.e., $\tilde{Z} = I$, for the Lanczos-type algorithms.

The University of Texas System Cray X-MP/24 vector computer was used to perform the runs presented here. Single precision real arithmetic was used, with unit roundoff given by $\epsilon_M = (7.1 \times 10^{-15})$.

The first set of test matrix problems to be considered arises from the finite difference discretization of the boundary value problem

$$-u_{xx}(x,y) - u_{yy}(x,y) + Du_x(x,y) = G(x,y) \quad \text{on } \Omega = [0,1]^2,$$
$$u(x,y) = 1 + xy \quad \text{on } \partial\Omega.$$

We utilize central differencing to discretize this problem, with uniform mesh spacing h in either direction. This yields a matrix of size $N = (n_h - 1)^2$ (where $h = 1/n_h$), after boundary points have been eliminated. The right-hand-side function $G(x,y)$ is defined so that the true solution is $u(x,y) = 1 + xy$ on Ω. By varying the constant D, the amount of nonsymmetry of the matrix may be varied. Specifically, for a given h there exist a symmetric matrix A_S and a skew-symmetric matrix A_N, independent of D, such that $A = A_S + D \cdot A_N$.

In Tables 1–3 we consider the unpreconditioned problem and also the (left) ILU- and MILU-preconditioned problem (see [17] and [32]). Runs for which convergence was not possible in ITMAX iterations are labeled by (–).

We make the following observations about these runs.

- For the unpreconditioned problem, the standard BCG and CGS algorithms break down in a number of cases, but the use of random $u^{(0)}$ or the use of BCGNB or CGSNB resulted in convergence. Furthermore, the iteration counts for the algorithms BCG and BCGNB are in general comparatively close to those of the "best" method, GMRES(∞), while these algorithms have short economical recurrences, unlike GMRES(∞). This underscores the importance of the Lanczos algorithms as economical solution techniques.

TABLE 2
Model problem, $h^{-1} = 128$, ILU-preconditioning, ITMAX=500. *Number of iterations.*

Method \ Dh:	0	2^{-3}	2^{-2}	2^{-1}	2^{0}	2^{1}	2^{2}	2^{3}	2^{4}	2^{5}
GMRES(∞)	92	83	74	64	52	41	32	26	19	14
BCG	94	102	90	82	178	53	35	28	22	17
BCG, random $u^{(0)}$	94	102	92	87	55	45	34	28	22	17
BCGNB	94	102	88	77	128	67	35	27	21	17
CGS	74	68	64	90	–	97	26	18	12	9
CGS, random $u^{(0)}$	63	59	58	55	48	33	25	18	13	9
CGSNB	74	68	61	73	48	39	26	18	12	9

TABLE 3
Model problem, $h^{-1} = 128$, MILU-preconditioning, ITMAX=500. *Number of iterations.*

Method \ Dh:	0	2^{-3}	2^{-2}	2^{-1}	2^{0}	2^{1}	2^{2}	2^{3}	2^{4}	2^{5}
GMRES(∞)	27	25	24	26	28	28	25	19	14	10
GMRES(∞), random $u^{(0)}$	33	29	28	29	31	31	29	24	19	14
BCG	31	27	29	33	30	37	30	23	15	10
BCG, random $u^{(0)}$	38	34	33	37	44	40	38	29	23	18
BCGNB	28	27	29	30	34	35	30	23	15	10
CGS	21	18	17	20	22	22	19	15	9	6
CGS, random $u^{(0)}$	24	18	20	22	22	23	21	16	12	9
CGSNB	21	18	17	20	22	27	20	15	9	6

- For the ILU-preconditioned problems, in most cases all methods worked well. For the case of $Dh = 1$, BCG gave an excessive number of iterations, but this was remedied significantly by BCGNB and much more so by the use of random $u^{(0)}$. Similarly, CGS could not converge, but CGSNB and CGS with random $u^{(0)}$ both converged.
- For all of the MILU-preconditioned problems, all of the Lanczos-type algorithms performed quite well. In particular, the BCG algorithm gave approximately the same number of iterations as GMRES(∞).

Figures 1 and 2 give representative plots of the convergence behavior of the algorithms for the case of $h^{-1} = 128$, $Dh = 4$, and no preconditioning. These results show that the new algorithms keep the residual size better behaved than the standard BCG and CGS algorithms over the course of the run.

We now consider a more difficult class of finite difference problems, namely, central finite differencing applied to the Dirichlet problem

$$-u_{xx}(x,y) - u_{yy}(x,y) + D[(y - \tfrac{1}{2})u_x(x,y) + (x - \tfrac{1}{3})(x - \tfrac{2}{3})u_y(x,y)],$$

$$-43\pi^2 u(x,y) = G(x,y) \quad \text{on } \Omega = [0,1]^2,$$

$$u(x,y) = 1 + xy \quad \text{on } \partial\Omega,$$

with $G(x,y)$ chosen as before so that the true solution is $u(x,y) = 1 + xy$. Again, we let h denote the mesh size in each direction. For $D = 0$ and h small, the matrix generated by this problem is a symmetric indefinite matrix with 16 distinct negative eigenvalues and the rest of the spectrum positive.

The standard conjugate residual algorithm applied to this problem with $h^{-1} = 128$ and $D = 0$ requires 766 iterations to converge to $||r^{(n)}||/||b|| < \zeta = 10^{-6}$. In any case, this is a difficult problem to solve.

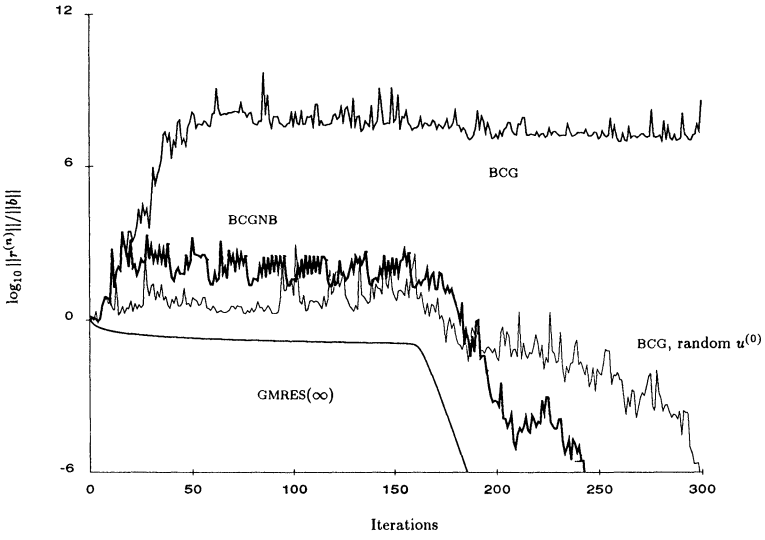

FIG. 1. *Residual behavior:* $h^{-1} = 128$, $Dh = 4$.

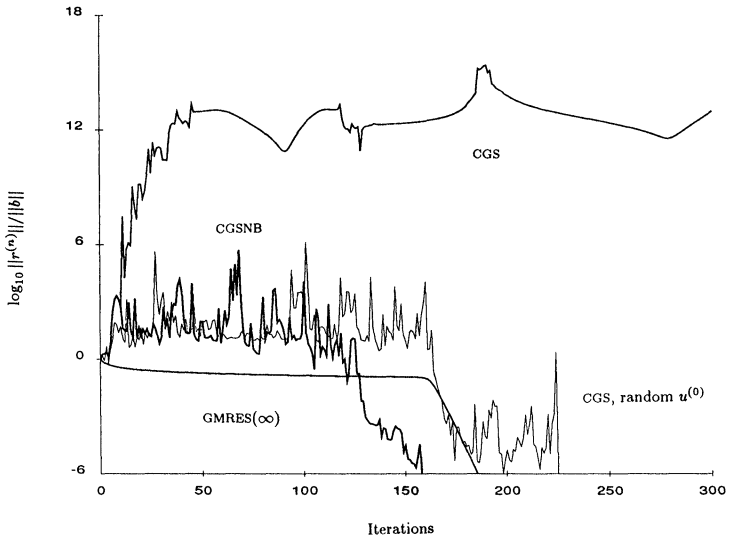

FIG. 2. *Residual behavior:* $h^{-1} = 128$, $Dh = 4$.

In Table 4 we give numerical results for various algorithms applied to this problem. The BCG algorithm applied to the $Dh = .5$ case with a second random vector $u^{(0)}$ failed to converge. Also, for the CGS algorithm applied to $Dh = .5$ with random $u^{(0)}$, convergence was indicated, but the final *true* value of $||r^{(n)}||/||b||$ had a (degraded) value of $.12 \times 10^{-3}$.

We now comment on these runs.

- The BCG algorithm applied to this problem gave good results. Similar results

TABLE 4
Indefinite problem, $h^{-1} = 128$, ITMAX=8000. *Iterations.*

Method \ Dh:	0	2^{-3}	2^{-2}	2^{-1}
BCG	820	1803	2209	3384
BCG, random $u^{(0)}$	839	1767	2707	4079
BCGNB	835	1814	2530	–
BCGNB, $\epsilon_3 = \epsilon_M^{1/2}/100$	835	1814	2397	4794
CGS	939	–	–	–
CGS, random $u^{(0)}$	789	2307	3268	6329
CGSNB	–	–	–	–
CGSNB, $\epsilon_h = \epsilon_M^{1/2}/100$	939	–	–	–
GMRES(5)	–	–	–	–
GMRES(10)	–	–	–	–
GMRES(20)	–	–	–	–
GMRES(40)	–	–	–	–
LSQR	–	–	–	–

were given by using random $u^{(0)}$, but for some random choices of $u^{(0)}$ the algorithm did not converge.

- The BCGNB algorithm converged for all cases except $Dh = .5$. It was found that a smaller value of ϵ_3, causing a more stringent requirement for restarting, was able to give convergence in this case.
- The CGS algorithm was not able to converge for the nonsymmetric cases. This was effectively remedied by random $u^{(0)}$ except for the $Dh = .5$ case. The CGSNB algorithm was not able to converge for the nonsymmetric cases. This might suggest that the use of random $u^{(0)}$ is a safer strategy than restarting, in general.
- For comparison purposes, we give run results for the GMRES(k) algorithm, which is the GMRES(∞) algorithm restarted every k iterations. This is known to be an effective algorithm for solving linear systems for which A is a definite matrix. We see that for a wide range of choices of k, the restarted GMRES algorithm was not able to converge for any of the problems in the given number of iterations, even when k was rather large. Furthermore, the normal equations algorithm LSQR was not able to solve these problems either in the given number of iterations.

A major conclusion to be drawn from this example is that the class of Lanczos-type methods is an important alternative for solving difficult matrix problems such as this problem. In fact, for this problem the Lanczos-type methods were the only methods that converged among all the methods tried.

9. Conclusions. In this paper we have examined some theoretical aspects of the Lanczos method and have defined and tested modified Lanczos algorithms which are of significant use in remedying the convergence problems of the Lanczos method. The results given here indicate that the class of Lanczos-type methods is an important alternative for solving nonsymmetric systems of equations. However, further research is necessary to better understand the behavior of the Lanczos algorithms, particularly in finite precision arithmetic. A theoretical understanding of the phenomenon of loss of orthogonality of the Lanczos vectors for the nonsymmetric case is necessary (see [16] and [34]). Despite these difficulties, positive experimental results with the Lanczos method continue to make it a significant method for solving nonsymmetric systems of equations.

REFERENCES

[1] R. O. ABBASIAN, *Lanczos algorithms for the acceleration of nonsymmetrizable iterative methods*, Master's thesis, Report CNA-193, Center for Numerical Analysis, University of Texas at Austin, Austin, TX, May 1984.

[2] S. F. ASHBY, T. A. MANTEUFFEL, AND P. E. SAYLOR, *A taxonomy for conjugate gradient methods*, SIAM J. Numer. Anal., 27 (1990), pp. 1542–1568.

[3] D. BOLEY AND G. GOLUB, *The nonsymmetric Lanczos algorithm and controllability*, Numerical Analysis Project Manuscript NA-90-06, Computer Science Department, Stanford University, Stanford, CA, May 1990.

[4] R. L. CAUSEY AND R. T. GREGORY, *On Lanczos' algorithm for tridiagonalizing matrices*, SIAM Rev., 3 (1961), pp. 322–328.

[5] R. CHANDRA, S. C. EISENSTAT, AND M. H. SCHULTZ, *The modified conjugate residual method for partial differential equations*, in Advances in Computer Methods for Partial Differential Equations II, R. Vichnevetsky, ed., IMACS, Rutgers University, New Brunswick, NJ, 1977, pp. 13–19.

[6] J. CULLUM, W. KERNER, AND R. WILLOUGHBY, *A generalized nonsymmetric Lanczos procedure*, Comput. Phys. Comm., 53 (1989), pp. 19–48.

[7] J. CULLUM AND R. A. WILLOUGHBY, *A practical procedure for computing eigenvalues of large sparse nonsymmetric matrices*, in Large Scale Eigenvalue Problems: Proceedings of the IBM Europe Institute Workshop on Large Scale Eigenvalue Problems, Oberlech, Austria, July 8–12, 1985, J. Cullum and R. A. Willoughby, eds., North-Holland, Amsterdam, 1986, pp. 193–240.

[8] ———, *Computing eigenvalues of large matrices, some Lanczos algorithms and a shift and invert strategy*, Research Report RC 15835(#67507) 11/13/89, IBM T. J. Watson Research Center, Yorktown Heights, NY; in Proc. 5th Instituto de Investigaciones en Mathematicas Aplicadas y en Sistemas, Universidad Nacional Autonomada Mexico, 5th International Workshop on Numerical Analysis, Merida, Mexico, January 3–6, 1989, S. Gomez and J. P. Hennart, eds., Society for Industrial and Applied Mathematics, Philadelphia, PA, March 1991.

[9] V. FABER AND T. MANTEUFFEL, *Necessary and sufficient conditions for the existence of a conjugate gradient method*, SIAM J. Numer. Anal., 21 (1984), pp. 352–362.

[10] ———, *Orthogonal error methods*, SIAM J. Numer. Anal., 24 (1987), pp. 170–187.

[11] R. FLETCHER, *Conjugate gradient methods for indefinite systems*, in Numerical Analysis Dundee 1975, G. A. Watson, ed., Lecture Notes in Mathematics 506, Springer-Verlag, New York, 1976, pp. 73–89.

[12] R. FREUND, *Conjugate gradient type methods for linear systems with complex symmetric coefficient matrices*, SIAM J. Sci. Statist. Comput., 13 (1992), pp. 425–448.

[13] R. W. FREUND, M. H. GUTKNECHT, AND N. M. NACHTIGAL, *An implementation of the look-ahead Lanczos algorithm for non-Hermitian matrices, Part* I, RIACS Tech. Report 90.45, NASA Ames Research Center, Moffett Field, CA, November 1990.

[14] R. W. FREUND AND N. NACHTIGAL, *An implementation of the look-ahead Lanczos algorithm for non-Hermitian matrices, Part* II, RIACS Tech. Report 90.46, NASA Ames Research Center, Moffett Field, CA, November 1990.

[15] ———, QMR: *A quasi-minimal residual method for non-Hermitian linear systems*, Numer. Math., 60 (1991), pp. 315–339.

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

[17] I. GUSTAFSSON, *Stability and rate of convergence of modified incomplete Cholesky factorization methods*, Ph.D. thesis, Chalmers University of Technology and the University of Goteborg, Goteborg, Sweden, April 1979.

[18] M. H. GUTKNECHT, *The unsymmetric Lanczos algorithms and their relations to Padé approximation, continued fractions, and the QD algorithm*, in Proc. Copper Mountain Conference on Iterative Methods, April 1–5, 1990.

[19] ———, *A completed theory of the unsymmetric Lanczos process and related algorithms, Part* I, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 594–639.

[20] ———, *A completed theory of the unsymmetric Lanczos process and related numerical algorithms, Part* II, IPS Research Report 90-16, ETH-Zentrum, Zurich, Switzerland, Sept. 1990; SIAM J. Matrix Anal. Appl., submitted.

[21] M. R. HESTENES AND E. L. STIEFEL, *Methods of conjugate gradients for solving linear systems*, J. Res. Nat. Bur. Standards, 49 (1952), pp. 409–436.
[22] A. S. HOUSEHOLDER, *The Theory of Matrices in Numerical Analysis*, Dover, New York, 1964.
[23] K. C. JEA AND D. M. YOUNG, *On the simplification of generalized conjugate-gradient methods for nonsymmetrizable linear systems*, Linear Algebra Appl., 52/53 (1983), pp. 399–417.
[24] W. D. JOUBERT, *Generalized conjugate gradient and Lanczos methods for the solution of nonsymmetric systems of linear equations*, Ph.D. thesis and Report CNA-238, Center for Numerical Analysis, University of Texas, Austin, TX, January 1990.
[25] ———, *Iterative methods for the solution of nonsymmetric systems of linear equations*, Report CNA-242, Center for Numerical Analysis, University of Texas at Austin, Austin, TX, February 1990.
[26] W. D. JOUBERT AND T. A. MANTEUFFEL, *Iterative methods for nonsymmetric linear systems*, in Iterative Methods for Large Linear Systems, D. R. Kincaid and L. J. Hayes, eds., Academic Press, Boston, 1990, pp. 149–171.
[27] W. D. JOUBERT AND D. M. YOUNG, *Necessary and sufficient conditions for the simplification of generalized conjugate gradient algorithms*, Linear Algebra Appl., 88/89 (1987), pp. 449–485.
[28] S. K. KIM AND A. T. CHRONOPOULOS, *An efficient nonsymmetric Lanczos method on parallel vector computers*, Tech. Report TR 90-38, Computer Science Department, University of Minnesota, Minneapolis, MN, July 1990.
[29] C. LANCZOS, *An iteration method for the solution of the eigenvalue problem of linear differential and integral operators*, J. Res. Nat. Bur. Standards, 45 (1950), pp. 255–282.
[30] H. P. LANGTANGEN AND A. TVEITO, *A numerical comparison of conjugate gradient-like methods*, Comm. Appl. Numer. Methods, 4 (1988), pp. 793–798.
[31] R. T. MCLAY, *Finite element simulation of coupled fluid flow, heat transfer and magnetic fields with applications to welding*, Ph.D. thesis, Department of Aerospace Engineering and Engineering Mechanics, University of Texas at Austin, Austin, TX, August 1988.
[32] 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*, Math. Comp., 31 (1977), pp. 148–162.
[33] C. C. PAIGE AND M. A. SAUNDERS, LSQR: *an algorithm for sparse linear equations and sparse least squares*, ACM Trans. Math. Software, 8 (1982), pp. 43–71.
[34] B. N. PARLETT, *The Symmetric Eigenvalue Problem*, Prentice–Hall, Englewood Cliffs, NJ, 1980.
[35] ———, *Reduction to tridiagonal form and minimal realizations*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 567–593.
[36] B. N. PARLETT AND D. TAYLOR, *A look ahead Lanczos algorithm for unsymmetric matrices*, Report PAM-43, Center for Pure and Applied Mathematics, University of California, Berkeley, CA, 1981.
[37] B. N. PARLETT, D. R. TAYLOR, AND Z. A. LIU, *A look-ahead Lanczos algorithm for unsymmetric matrices*, Math. Comp., 44 (1985), pp. 105–124.
[38] H. L. ROYDEN, *Real Analysis*, Second Edition, MacMillan, New York, 1968.
[39] W. RUDIN, *Real and Complex Analysis*, McGraw-Hill, New York, 1966.
[40] H. RUTISHAUSER, *Beiträge zur Kenntnis des Biorthogonalisierungs-Algorithmus von Lanczos*, Z. Angew. Math. Phys., 4 (1953), pp. 35–56. (In German.)
[41] Y. SAAD, *The Lanczos biorthogonalization algorithm and other oblique projection methods for solving large unsymmetric systems*, SIAM J. Numer. Anal., 19 (1982), pp. 485–506.
[42] Y. SAAD AND M. H. SCHULTZ, *Conjugate gradient-like algorithms for solving nonsymmetric linear systems*, Math. Comp., 44 (1985), pp. 417–424.
[43] ———, GMRES: *a generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Statist. Comput., 7 (1986), pp. 856–869.
[44] S. SAKS, *The Theory of the Integral*, G. E. Stechert, New York, 1937.
[45] P. SONNEVELD, CGS, *a fast Lanczos-type solver for nonsymmetric linear systems*, SIAM J. Sci. Statist. Comput., 10 (1989), pp. 36–52.
[46] D. R. TAYLOR, *Analysis of the look ahead Lanczos algorithm*, Ph.D. thesis, Department of Mathematics, University of California, Berkeley, CA, November 1982; Report PAM-108, Center for Pure and Applied Mathematics, University of California, Berkeley, CA.
[47] H. A. VAN DER VORST, *Conjugate gradient methods*, in Proc. Modern Numerical Algorithms for Supercomputers: A Tutorial Seminar, October 9–13 1989, University of Texas System Center for High Performance Computing, Austin, TX, pp. 268–324.

[48] H. A. VAN DER VORST, Bi-CGSTAB: *a fast and smoothly converging variant of* Bi-CG *for the solution of nonsymmetric linear systems*, SIAM J. Sci. Statist. Comput., 13 (1992), pp. 631–644.

[49] J. H. WILKINSON, *The Algebraic Eigenvalue Problem*, Oxford University Press, London, 1965.

[50] T. YAMAMOTO, *On Lanczos' algorithm for tri-diagonalization*, J. Sci. Hiroshima Univ. Ser. A-I, 32 (1968), pp. 259–284.