MATHEMATICS OF COMPUTATION
VOLUME 64, NUMBER 212
OCTOBER 1995, PAGES 1577–1588

# LOW-RANK MODIFICATION OF THE UNSYMMETRIC LANCZOS ALGORITHM

THOMAS HUCKLE

ABSTRACT. The unsymmetric Lanczos algorithm is an important method for eigenvalue estimation and for solving linear equations. Unfortunately, the algorithm may break down without providing useful information; this is referred to as a serious breakdown in the literature. Here, we introduce a low-rank modification of the original matrix $A$ in the case of a serious breakdown. This modification can be used to cure a serious breakdown as long as we have orthogonality of the already computed Lanczos vectors. We can switch to a new rank-1 modified matrix $\tilde{A} = A + ab^T$ such that

- the Lanczos algorithm has no serious breakdown in this step when applied on $\tilde{A}$,
- the already computed variables in the Lanczos algorithm for $A$ and $\tilde{A}$ coincide,
- using a Lanczos-based iterative solver, e.g. BCG or QMR, with start vectors $x_0 = 0$ and $v_1 = f$, we have $A^{-1}f = \tilde{A}^{-1}f$, and thus by continuing the Lanczos algorithm with $\tilde{A}$ we automatically get the desired solution $A^{-1}f$.

Also, if the Lanczos vectors have lost their orthogonality, we show theoretically and by numerical examples that the modified Lanczos method has the same convergence behavior as the Lanczos method without breakdown. Thus, in the case of a serious breakdown we only have to compute the new rank-1 modified matrix $\tilde{A}$ and step further in the original algorithm now using $\tilde{A}$.

## 1. A MODIFIED LANCZOS ALGORITHM

Let $A$ be an unsymmetric matrix, $v_1$ and $w_1$ two vectors of length 1 that are not orthogonal, and $\beta_1 = \gamma_1 = 0$, $d_1 = w_1^T v_1 \neq 0$. Then the unsymmetric Lanczos algorithm is defined for $j = 1, 2, \ldots$ by the recursion

$$\begin{gathered}
\alpha_j = w_j^T A v_j / d_j, \\
\hat{v}_{j+1} = Av_j - \alpha_j v_j - \beta_j v_{j-1}, \qquad \hat{w}_{j=1} = A^T w_j - \alpha_j w_j - \gamma_j w_{j-1}, \\
v_{j+1} = \frac{\hat{v}_{j+1}}{\|\hat{v}_{j+1}\|}, \qquad w_{j+1} = \frac{\hat{w}_{j+1}}{\|\hat{w}_{j+1}\|}, \\
d_{j+1} = w_{j+1}^T v_{j+1}, \quad \gamma_{j+1} = \frac{d_{j+1}\|\hat{v}_{j+1}\|}{d_j}, \quad \beta_{j+1} = \frac{d_{j+1}\|\hat{w}_{j+1}\|}{d_j}.
\end{gathered} \tag{1}$$

Received by the editor February 10, 1994 and, in revised form, October 17, 1994.

1991 *Mathematics Subject Classification*. Primary 65F10.

This work was supported by a research grant from the Deutsche Forschungsgemeinschaft.

A serious breakdown occurs if and only if $|d_j| < \varepsilon$ with $\|\hat{v}_j\| > \varepsilon$ and $\|\hat{w}_j\| > \varepsilon$ for a small positive $\varepsilon \ll 1$. The recursion (1) generates a tridiagonal matrix

$$T_j = \text{tridiag}\left(\frac{\gamma_k d_{k-1}}{d_k}, \alpha_k, \beta_{k+1}\right)_{k=1}^{j},$$

a diagonal matrix $D_j = \text{diag}(d_1, \dots, d_j)$, and $V_j = (v_1, \dots, v_j)$, $W_j = (w_1, \dots, w_j)$. This yields the matrix equations

$$AV_j = V_jT_j + \hat{v}_{j+1}e_j^T, \qquad W_j^TV_j = D_j, \tag{2}$$

where $e_j = (0, \dots, 0, 1, 0, \dots, 0)^T$ with 1 at the $j$th position.

These formulae can be applied to find eigenvalue estimates for $A$ [8, 1, 3] and for the iterative solution of linear equations in $A$, e.g. with QMR or BCG [9, 5, 13, 10]. In the case of a serious breakdown with $d_j \approx 0$, look-ahead strategies can be introduced to replace the tridiagonal matrix $T_j$ by a block tridiagonal matrix [11, 14, 6, 3, 4, 2, 7] or a band matrix [15].

In this note we will analyze another possible way to handle breakdown. To this end, we replace the original matrix $A$ by a rank-1 modification $A + ab^T$, where $a$ and $b$ are chosen in such a way that the Lanczos method applied on $A+ab^T$ has no breakdown in this step, and that the already computed numbers and vectors remain unchanged. Let us assume that the algorithm has computed all variables until the $(j-1)$st step, where we get $|d_j| = |w_j^Tv_j| \le \varepsilon$ for a given small positive $\varepsilon$ and $\|\hat{v}_j\| > \varepsilon$ and $\|\hat{w}_j\| > \varepsilon$. In the following we will show how $a$ and $b$ can be chosen to cure this serious breakdown. We cannot orthogonalize $Av_j$ against $w_j$, and $A^Tw_j$ against $v_j$, in order to define $v_{j+1}$ and $w_{j+1}$ by (1). But we can still orthogonalize $A^Tw_j$ against $v_1, \dots, v_{j-1}$. This gives

$$\begin{aligned}
\hat{w}_{j+1} &:= A^Tw_j - \frac{v_{j-1}^TA^Tw_j}{d_{j-1}}w_{j-1} \\
&= A^Tw_j - \frac{(\hat{v}_j + \alpha_{j-1}v_{j-1} + \beta_{j-1}v_{j-2})^Tw_j}{d_{j-1}}w_{j-1} \\
&= A^Tw_j - \frac{d_j\|\hat{v}_j\|}{d_{j-1}}w_{j-1}(\approx A^Tw_j),
\end{aligned}$$

$$w_{j+1} := \frac{\hat{w}_{j+1}}{\|\hat{w}_{j+1}\|}.$$

In the same way, we can derive $w_{j+2}$ by orthogonalizing $A^Tw_{j+1}$ against $v_{j-1}$, or more generally,

$$\begin{aligned}
\hat{w}_{j+k-1} &:= A^Tw_{j+k-2} - \frac{v_{j-1}^TA^Tw_{j+k-2}}{d_{j-1}}w_{j-1} = A^Tw_{j+k-2} - \frac{\hat{v}_j^Tw_{j+k-2}}{d_{j-1}}w_{j-1} \\
&= A^Tw_{j+k-2} - \frac{\|\hat{v}_j\|(v_j^TA^Tw_{j+k-3})}{d_{j-1}}w_{j-1},
\end{aligned} \tag{3}$$

$$w_{j+k-1} := \frac{\hat{w}_{j+k-1}}{\|\hat{w}_{j+k-1}\|}.$$

Now, let $k$ be the first index with $|w_{j+k-1}^TAv_j| \ge n(A)\varepsilon$, where $n(A)$ is some norm of $A$, for example the 1-norm. If there exists no such $k$, then we have the case of an incurable breakdown, and we have to restart (see [14, 11]). Otherwise, without further orthogonalization we can define

$$w_{j+k} := A^T w_{j+k-1}. \tag{4}$$

By this definition we relax the orthogonality conditions on $w_{j+k}$ and in return get the very useful equation

$$w_{j+k}^T A^{-1} v_{j-1} = 0. \tag{5}$$

Furthermore, $w_j, \dots, w_{j+k-1}$ are orthogonal on $v_1, \dots, v_{j-1}$, and $w_{j+k}$ is orthogonal on $v_1, \dots, v_{j-2}$. We want to find a new matrix $\tilde{A} := A + ab^T$ with vectors $a$ and $b$ such that nearly all computed variables remain unchanged and the Lanczos algorithm applied to $\tilde{A}$ has no breakdown in the $(j-1)$st step. We can satisfy these conditions by choosing $a \perp w_1, \dots, w_{j-2}$ and $b \in \operatorname{span}(w_{j-1}, \dots, w_{j+k})$. Let us define

$$a := v_{j-1} \quad \text{and} \quad b := \lambda_j w_{j+k}, \qquad \tilde{A} := A + \lambda_j v_{j-1} w_{j+k}^T. \tag{6}$$

We denote the new variables of the Lanczos method applied to $\tilde{A}$ by an additional tilde. Then, for the initial values $\tilde{v}_1 := v_1$ and $\tilde{w}_1 = w_1$ we get

$$\tilde{\alpha}_{r-1} = \alpha_{r-1},\ \tilde{v}_r = v_r,\ \tilde{w}_r = w_r,\ \tilde{d}_r = d_r,$$
$$\tilde{\beta}_r = \beta_r,\ \tilde{\gamma}_r = \gamma_r \quad \text{for } r = 2, \dots, j-1,$$

$$\begin{aligned}\tilde{\alpha}_{j-1} &= \frac{w_{j-1}^T (A + \lambda_j v_{j-1} w_{j+k}^T) v_{j-1}}{d_{j-1}} \\ &= \alpha_{j-1} + \lambda_j (w_{j+k}^T v_{j-1}) = \alpha_{j-1} + \lambda_j \|\hat{v}_j\| (w_{j+k-1}^T v_j).\end{aligned}$$

Here, we use (4) and the original recursion in $Av_{j-1} = \hat{v}_j + \alpha_{j-1} v_{j-1} + \beta_{j-1} v_{j-2}$. Furthermore, we get

$$\tilde{\hat{v}}_j = (A + ab^T) v_{j-1} - \tilde{\alpha}_{j-1} v_{j-1} - \beta_{j-1} v_{j-2} = \hat{v}_j,$$

and

$$\begin{aligned}\tilde{\hat{w}}_j &= (A + ab^T)^T w_{j-1} - \tilde{\alpha}_{j-1} w_{j-1} - \beta_{j-1} w_{j-2} \\ &= \hat{w}_j + \lambda_j (d_{j-1} w_{j+k} - (w_{j+k}^T v_{j-1}) w_{j-1}),\end{aligned} \tag{7}$$

which yields

$$\tilde{\hat{w}}_j^T \tilde{\hat{v}}_j = \hat{w}_j^T \hat{v}_j + \lambda_j d_{j-1} w_{j+k}^T \hat{v}_j = \hat{w}_j^T \hat{v}_j + \lambda_j d_{j-1} \|\hat{v}_j\| (w_{j+k-1}^T A v_j). \tag{8}$$

Now, for small $\lambda_j$ with

$$|\lambda_j| > \left| \frac{\varepsilon \|\hat{w}_j\|}{d_{j-1}(w_{j+k-1}^T A v_j)} \right|$$

the Lanczos algorithm applied on $\tilde{A}$ can continue without breakdown. Furthermore, with $A$ also $\tilde{A}$ is regular. This is obvious by considering (5) and the Sherman-Morrison-Woodbury formula (SMW)

$$\tilde{A}^{-1} = (A + ab^T)^{-1} = A^{-1} - A^{-1} ab^T A^{-1} / (1 + b^T A^{-1} a)$$

with

$$\widetilde{A}^{-1}v_{j-1} = A^{-1}v_{j-1} \quad \text{and} \quad 1 \pm w_{j+k}^T A^{-1} v_{j-1} = 1 \pm w_{j+k-1}^T v_{j-1} = 1 \neq 0.$$

For $\widetilde{A}$, equation (2) gives

$$\widetilde{A}\widetilde{V}_m = \widetilde{V}_m \widetilde{T}_m + \tilde{v}_{m+1} e_m^T. \tag{9}$$

For $A$, equation (9) yields

$$\begin{aligned} A\widetilde{V}_m &= \widetilde{V}_m \widetilde{T}_m + \tilde{v}_{m+1} e_m^T - \lambda_j v_{j-1} w_{j+k}^T \widetilde{V}_m \\ &= \widetilde{V}_m (\widetilde{T}_m - \lambda_j e_{j-1} w_{j+k}^T \widetilde{V}_m) + \tilde{v}_{m+1} e_m^T. \end{aligned}$$

Thus, the tridiagonal matrix in (2) has to be replaced by a tridiagonal matrix plus a rank-1 matrix. In general, there may occur more breakdowns. Hence, by treating all these, say $p$, breakdowns in the same way, we arrive at a representation

$$\begin{aligned} A\widetilde{V}_m &= \widetilde{V}_m \widetilde{T}_m + \tilde{v}_{m+1} e_m^T - \sum_{s=1}^{p} \lambda_s v_{j_s-1} w_{j_s+k_s}^T \widetilde{V}_m \\ &= \widetilde{V}_m \left( \widetilde{T}_m - \sum_{s=1}^{p} \lambda_s e_{j_s-1} w_{j_s+k_s}^T \widetilde{V}_m \right) + \tilde{v}_{m+1} e_m^T \\ &= \widetilde{V}_m (\widetilde{T}_m - \widehat{E}_m X_m^T) + \tilde{v}_{m+1} e_m^T \end{aligned} \tag{10}$$

with

$$\widehat{E}_m = (e_{j_1-1}, \dots) \quad \text{and} \quad X_m^T = \Lambda W_{\#}^T \widetilde{V}_m,$$

where $\Lambda = \operatorname{diag}(\lambda_1, \dots, \lambda_p)$ and $W_{\#} = (w_{j_1+k_1} \cdots w_{j_p+k_p})$. The matrix $X_m$ can be computed recursively by

$$X_{m+1}^T = \begin{pmatrix} & \lambda_1 w_{j_1+k_1}^T \tilde{v}_{m+1} \\ X_m^T & \vdots \\ & \vdots \end{pmatrix}.$$

Note, that $X_m^T$ has $m$ columns and the number of rows is equal to $p$, the number of occurred breakdowns. In the case of a further breakdown we have to add a new row to $X_m^T$ with the first elements in this new row all zero.

**Modified Lanczos algorithm:**

```
start with vectors $v_1$ and $w_1$ having 2-norm 1, $d_1 = w_1^T v_1 \neq 0$, $\beta_1 = \gamma_1 = 0$,
for $j = 2, 3, \dots$
    compute $\alpha_{j-1}$, $\hat{v}_j$, $\hat{w}_j$, $v_j$, $w_j$ as in (1), so long as $\|\hat{v}_j\| > \varepsilon$ and $\|\hat{w}_j\| > \varepsilon$;
    if $|d_j| = |w_j^T v_j| < \varepsilon$ : replace $A$ by $A + \lambda_j v_{j-1} w_{j+k}^T$ with (3), (4), (6);
                                compute the new $\alpha_{j-1}$, $w_j$, $d_j$ by (7) and (8);
    compute $\beta_j$, $\gamma_j$ by (1)
```

The additional costs of this modified algorithm are one inner product $w_{j+k}^T \tilde{v}_m$ and one SAXPY for each occurred breakdown in order to compute the new Lanczos vector $\tilde{v}_{m+1}$. For $\tilde{w}_{m+1}$ we need no additional computations in view of $v_{j-1}^T \tilde{w}_m = 0$ in exact arithmetic. It is also possible to compute $\widetilde{A}$ explicitly in $O(n^2)$ operations, but in most applications, e.g. for sparse matrices, this will be too expensive.

Note that we could also replace (3) and (4) by corresponding equations for $v_{j+k}$ instead of $w_{j+k}$. This would lead to a new matrix $\widetilde{A}$ of the form $\widetilde{A} = A + \lambda_j v_{j+k} w_{j-1}^T$.

If we are interested in eigenvalue estimates for $A$, we need $\alpha_j$, $\beta_j$, $\gamma_j$, $d_j$, and $X_m^T$, which already occur in the modified Lanczos algorithm. Then, we can compute the eigenvalues of $\widetilde{T}_m - \widehat{E}_m X_m^T$ as approximations to the eigenvalues of $A$, e.g. using inverse iteration in connection with Sherman-Morrison-Woodbury (SMW) (or secular equation methods).

In [10] Parlett has introduced the idea of a low-rank modification in connection with the symmetric Lanczos algorithm for solving $Ax = b$ with several right-hand sides. If the linear equation is solved using the Lanczos algorithm, one can continue the algorithm with a new right-hand side which leads to a low-rank perturbation of the tridiagonal Lanczos matrix. The idea presented here perturbs the original matrix $A$ and is therefore quite different from the one in [10].

## 2. Solving linear equations using the modified Lanczos algorithm

If we want to solve a linear equation of the form $Ax = b$ $(r_0 = b - Ax_0 = v_1)$, we can also apply the modified Lanczos method. First, let us assume that the starting vector $x_0 = 0$. Then we get the same starting residual $r_0 = b$ for both $A$ and $\widetilde{A}$. This is always possible by replacing $x$ by $x - x_0$ and $b$ by $b - Ax_0$. Now, we can compute a solution $\tilde{x}$ for $\widetilde{A}x = b$ with the modified Lanczos algorithm, e.g. with QMR or any other Lanczos-based method, starting with the matrix $A$ and switching to $\widetilde{A}$ if a serious breakdown occurs. Then, we can use SMW to compute

$$
\begin{aligned}
x = A^{-1}b &= \left(\widetilde{A} - \sum_{s=1}^{p} \lambda_s v_{j_s-1} w_{j_s+k_s}^T\right)^{-1} b = (\widetilde{A} - V_\# Y_\#^T)^{-1} b \\
&= \tilde{x} + \widetilde{A}^{-1} V_\# (I - Y_\#^T \widetilde{A}^{-1} V_\#)^{-1} Y_\#^T \tilde{x} = \tilde{x}
\end{aligned}
\tag{11}
$$

with

$$V_\# = (v_{j_1-1}, \dots, v_{j_p-1}) \quad \text{and} \quad Y_\#^T = \Lambda W_\#^T.$$

Here, (11) follows from the equation

$$
\begin{aligned}
w_{j+k}^T \tilde{x} &= w_{j+k-1}^T A (A + V_\# Y_\#^T)^{-1} v_1 \\
&= w_{j+k-1}^T v_1 - w_{j+k-1}^T V_\# B^{-1} \begin{pmatrix} \lambda_1 w_{j_1+k_1-1}^T \\ \vdots \end{pmatrix} v_1 = 0
\end{aligned}
$$

by using (4) and the related orthogonality relations. Thus, the solution $\widetilde{A}^{-1}b$ computed with the modified Lanczos algorithm gives also the desired vector $x = A^{-1}b$ without additional costs.

Until now we have ignored the numerical aspects of the method. Here, the main questions are:

(i) how to define a breakdown (how to choose $\varepsilon$),
(ii) how large should be the rank-1 perturbation (how to choose $\lambda$),

(iii) how can we take into account the numerical loss of the biorthogonality of the Lanczos vectors.

First, if $\varepsilon$ is too large (e.g. $> 10^{-3}$), then we can get many unnecessary rank-1 perturbations. On the other hand, if $\varepsilon$ is too small (e.g. $< 10^{-10}$), then we miss a possible breakdown which leads to a loss of accuracy. This loss of accuracy may cause nonconvergence or a series of subsequent breakdowns. Many numerical examples in MATLAB with machine precision $10^{-16}$ showed that $\varepsilon = 10^{-6}$ or $10^{-5}$ give a good criterion for deciding whether a breakdown occurs or not. For a similar discussion, see [3] and [12]. In [12], Parlett suggests $\varepsilon = \text{eps}^{1/3}$ or $\text{eps}^{1/4}$ for machine precision eps, and our numerical examples support this choice.

Similar problems arise with $\lambda$. If $\lambda$ is too small, then the breakdown is not cured. If $\lambda$ is too large, then $ab^T$ predominates $A$ in $\tilde{A}$, and the convergence behavior of the iterative method may change totally. Specifically, if we think of preconditioned iterative methods, then by introducing a predominant rank-1 perturbation the spectrum of the new matrix may get worse, and the effect of the preconditioner may be disturbed. Hence, let us define

$$\lambda_j = \Theta \frac{\varepsilon \|\hat{w}_j\|}{d_{j-1}(w_{j+k-1}^T A v_j)} \tag{12}$$

with $|\Theta| > 1$ to cure the breakdown. Furthermore, $\lambda_j$ should be less than $n(A)$ for some norm of $A$. Then, the convergence of the iterative method will remain nearly unchanged.

Unfortunately, the Lanczos vectors will lose their biorthogonality in the course of the algorithm. Therefore, the new matrix $\tilde{A}$ and the related tridiagonal matrix $\tilde{T}$ computed by the modified Lanczos algorithm will satisfy (9) not exactly. How good (9) and (10) are fulfilled depends on the magnitude of the numbers $\lambda_j(w_s^T v_{j-1})$ and $\lambda_j(w_{j+k}^T v_s)$ for $s = 1, \dots, j-2$. In exact arithmetic these numbers would be zero independent of $\lambda_j$, but in practice, $w_s^T v_{j-1}$ may be near 1. Hence, reorthogonalization seems to be necessary. But we will show in the following that the loss of orthogonality only perturbs the convergence for solving $\tilde{A}^{-1}b$ but has no influence on the convergence to the really wanted $A^{-1}b$ as long as the modified algorithm converges.

Furthermore, in view of the loss of orthogonality we need in every step one additional SAXPY and one additional inner product to compute the new vector $\hat{w}_{j+1}$. In the following we will analyze the modified algorithm, taking into account the loss of orthogonality.

In exact arithmetic the modified Lanczos algorithm, e.g. with QMR, would be equivalent to solving $\tilde{A}x = b$ with this method, but in view of the loss of orthogonality the already computed numbers are no longer connected with $\tilde{A}$. Let us consider the influence of the loss of orthogonality. For simplicity, we assume that there is only one serious breakdown in the $(j-1)$st step $(d_j \approx 0)$. Then we can write the recursion in the matrix form

$$AV^1 = V^1 T^1 + \hat{v}_{j-1} e_{j-2}^T, \tag{13}$$

$$AV^2 = V^2 T^2 + \beta_{j-1} v_{j-2} e_{j-1}^T - \lambda_j v_{j-1} w_{j+k}^T V^2 + \hat{v}_{m+1} e_m^T \tag{14}$$

with

$$V^1 = (v_1, \dots, v_{j-2}) \quad \text{and} \quad V^2 = (v_{j-1}, \dots, v_m).$$

Combining (13) and (14), we get

$$AV = VT - \lambda_j v_{j-1} w_{j+k}^T (0 \ V^2) + \hat{v}_{m+1} e_m^T. \tag{15}$$

Here, $V = (V^1, V^2)$, and $T^1$, $T^2$, $T$ are tridiagonal matrices with coefficients as in (1). If the orthogonality would still hold, we could replace in (15) the 0 by $V^1$, which would give us again relations between the approximate solutions of $\tilde{A}^{-1}b$ and $A^{-1}b$.

First, let us consider the Galerkin method. Here, we define an approximate solution $x_m = Vz_m$ in such a way that

$$\begin{aligned} 0 &= W^T r_m = W^T(AVz_m - Ve_1) \\ &= W^T V((T - \lambda_j e_{j-1} w_{j+k}^T (0 \ V^2))z_m - e_1) + W^T \hat{v}_{m+1} e_m^T. \end{aligned} \tag{16}$$

If we ignore the last term, which would be 0 in exact arithmetic, the Galerkin condition (16) leads to

$$Hz_{m,H} = (T - \lambda_j e_{j-1} w_{j+k}^T (0 \ V^2))z_{m,H} = e_1, \qquad x_{m,H} = Vz_{m,H}. \tag{17}$$

If we try to solve $\tilde{A}x = b$ in spite of the loss of orthogonality, we would use

$$Tz_{m,T} = e_1, \qquad x_{m,T} = Vz_{m,T}, \tag{18}$$

which would provide us also with the solution for $Ax = b$ in exact arithmetic. Hence, we have two different ways to define estimates for $A^{-1}b$. Let us assume that both matrices $T$ and $H$ are regular. With (14) we get

$$\begin{aligned} w_{j+k}^T (0 \ V^2) T^{-1} e_1 &= w_{j+k-1}^T (0 \ V^2 T + \hat{v}_{m+1} e_{m-j+2}^T) T^{-1} e_1 \\ &= w_{j+k-1}^T (0 \ V^2) e_1 + w_{j+k-1}^T (0 \ \hat{v}_{m+1} e_{m-j+2}^T) T^{-1} e_1 \\ &= (e_m^T T^{-1} e_1)(w_{j+k-1}^T \hat{v}_{m+1}). \end{aligned}$$

Together with (15), this shows

$$\begin{aligned} AVT^{-1}e_1 &= v_1 + \hat{v}_{m+1}(e_m T^{-1} e_1) - \lambda_j v_{j-1} w_{j+k}^T (0 \ V^2) T^{-1} e_1 \\ &= v_1 + (e_m T^{-1} e_1)(\hat{v}_{m+1} - \lambda_j (w_{j+k-1}^T \hat{v}_{m+1}) v_{j-1}). \end{aligned}$$

In view of (18), we get for the desired solution

$$A^{-1}b = A^{-1}v_1 = x_{m,T} + (e_m^T T^{-1} e_1)(-A^{-1}\hat{v}_{m+1} + \lambda_j (w_{j+k-1}^T \hat{v}_{m+1}) A^{-1} v_{j-1}).$$

Similarly,

$$w_{j+k}^T (0 \ V^2) T^{-1} e_{j-1} = (e_m^T T^{-1} e_{j-1})(w_{j+k-1}^T \hat{v}_{m+1})$$

and

$$Ax_{m,H} = AVH^{-1}e_1 = v_1 + \hat{v}_{m+1}(e_m^T H^{-1} e_1)$$

with

$$e_m^T H^{-1} e_1 = (e_m^T T^{-1} e_1) \frac{1}{1 - \lambda_j (e_m^T T^{-1} e_{j-1})(w_{j+k-1}^T \hat{v}_{m+1})}.$$

Hence, the approximate solutions that derive from (18) and (17) depend both on the size of $(e_m^T T^{-1} e_1)\hat{v}_{m+1}$, and this last term describes the convergence of the computed approximations for solving $\tilde{A}^{-1}b$. If therefore $x_{m,H}$ or $x_{m,T}$ gives a good approximation for $A^{-1}b$, then the other will also be a good approximation. Thus, the vector $x_{m,T} = VT^{-1}e_1$, the solution for the Galerkin condition of the modified Lanczos algorithm using only the tridiagonal matrix without rank-1 modification (as if one would like to solve $\widetilde{A}^{-1}b$), converges in the same way like $x_{m,H}$, the estimate (17) which also takes into account the rank-1 term. But, in view of the loss of orthogonality, $x_{m,T}$ and $x_{m,H}$ will in general be no longer approximate solutions to $\widetilde{A}x = b$.

In the same way, instead of the Galerkin condition, we can use the QMR condition [4],

$$\min \left\| \Omega_m \left( \left( T_{m+1,m} - \sum_{s=1}^{p} \lambda_s e_{j_s-1} w_{j_s+k_s}^T V_m \right) z_m - e_1 \right) \right\| \tag{19}$$

or

$$\min \| \Omega_m (T_{m+1,m} z_m - e_1) \| \tag{20}$$

with $T_{m+1,m}$ a tridiagonal matrix with $m+1$ rows and $m$ columns and $\Omega_m$ a diagonal scaling matrix, e.g. $\Omega_m$ the $m \times m$ identity matrix. To get solutions for $Ax = b$, we can solve (17), (18), (19), (20) directly, or use recursive updates of QR-decompositions of the occurring tridiagonal matrices as described in [3].

Hence, we get the following general method for solving linear equations based on the Lanczos method:

**Iterative solution of linear equations with the modified Lanczos algorithm:**

- transform $Ax = b$ such that the starting value is $x_0 = 0$;
- choose a Lanczos-based iterative algorithm e.g. as described in [13] or [3];
- if a serious breakdown occurs, use the computed last Lanczos vectors to find a rank-1 modification of $A$ as described in the modified Lanczos algorithm; continue in the basic iterative method, now using the new modified matrix without any further changes.

There are some variations in getting the rank-1 term in the case of a serious breakdown. For example,

- one can compute a fixed number of new vectors $w_{j+p-1}$ of the form (3), and then choose the rank-1 term and the index $k$ in such a way that $|w_{j+k-1}^T A v_j|$ is maximal [3],
- use $v_{j+k-1}$ and $w_j$ defined as in (3-4) to get a rank-1 perturbation,
- replace the recursion (3) for the vectors $w_{j+k-1}^T$ by another formula that leads to the same orthogonality relations.

The advantages of the modified Lanczos solvers are that

- they are very easy to program, because the original algorithm remains unchanged,
- the additional costs of two inner products and two SAXPY's per iteration are negligible for most matrices,
- the convergence behavior is unchanged.

## 3. Numerical examples

The numerical examples have to demonstrate two assertions:

- the constructed rank-1 perturbation gives the desired solution independent of the loss of orthogonality,
- the modified algorithms with properly chosen $\varepsilon$ and $\lambda$ work in practice.

First, we give an example in which with full orthogonality the modified Lanczos algorithm provides us with the solution for $Ax = b$ and $\tilde{A}\tilde{x} = b$, where $x = \tilde{x}$. We choose $\varepsilon = 10^{-6}$, $b = v_1 = e_1$, $w_1 = (1, 1, 1, u_4, \dots)$ with random numbers $u_j$ between 0 and 1, and the matrix

$$A = \begin{pmatrix} 0 & \dots & \dots & 0 & 1 \\ 1 & 0 & \dots & \dots & 0 \\ 0 & \ddots & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & 1 & 0 \end{pmatrix}.$$

Here, a serious breakdown occurs only in the first step $j = 2$. Examples with $A$ appear often in connection with breakdowns in the Lanczos algorithm because one can easily produce a serious breakdown in one of the first iteration steps. We run the modified Lanczos algorithm with matrix size $n = 150$ for $j = 2, 3, \dots, 170 = m$ and different values of $\Theta$ in (12). In the QMR condition we always choose $\Omega_m$ to be the identity matrix. For other random initial vectors $v_1$ and $w_1$ with no breakdown we get in the 170th step relative residuals $\|r_j\|/\|r_0\|$ between $10^{-6}$ and $10^{-10}$.

TABLE 1. Norm of the relative residual with exact breakdown at $j = 2$, $m = 170$

| $\Theta$ | $r(\tilde{A})$ | $r(A)$ (17) | $r(A)$ (18) | $r(A)$ (19) | $r(A)$ (20) |
|---|---|---|---|---|---|
| 10 | $5.8E-06$ | $5.8E-06$ | $5.8E-06$ | $4.5E-09$ | $4.5E-09$ |
| 100 | $5.4E-10$ | $5.4E-10$ | $5.4E-10$ | $4.2E-10$ | $4.2E-10$ |
| $10^3$ | $1.6E-09$ | $1.6E-09$ | $1.6E-09$ | $2.7E-10$ | $2.6E-10$ |
| $10^4$ | $9.7E-10$ | $9.7E-10$ | $9.5E-10$ | $2.1E-10$ | $2.0E-10$ |
| $10^5$ | $4.4E-08$ | $4.4E-08$ | $4.4E-08$ | $2.5E-08$ | $2.2E-08$ |

In the second example, we replaced $A$ by a perturbed matrix $A + \delta E$, where $E$ was a random matrix with entries between $-1$ and 1. Using $\delta$, one can design near breakdowns to analyze the stability of the modified Lanczos algorithm. For $|\delta|$ larger than $10^{-4}$ no breakdown occurred in the modified algorithm. For example, with $\delta = 10^{-3}$ the solution was computed without rank-1 modification with relative residual norm $6.7E-06$ for $m = 150$ and $5.0E-8$ in the 152nd step.

TABLE 2. Norm of the relative residual with breakdown at $j = 2$, $\varepsilon = 10^{-6}$, $\delta = 10^{-5}$, QMR(20)

| $m/\Theta$ | 10 | 100 | $10^3$ | $10^4$ | $10^5$ |
|---|---|---|---|---|---|
| 150 | $1.4E-06$ | $5.2E-08$ | $3.6E-08$ | $8.9E-10$ | 0.66 |
| 151 | $1.1E-06$ | $5.0E-08$ | $1.7E-08$ | $7.5E-10$ | 0.45 |
| 152 | $1.0E-06$ | $4.5E-08$ | $1.7E-08$ | $7.2E-10$ | $1.5E-08$ |
| 160 | $1.0E-06$ | $3.7E-08$ | $1.5E-08$ | $6.9E-10$ | $1.4E-09$ |
| 170 | $1.0E-06$ | $3.2E-08$ | $1.5E-08$ | $6.9E-10$ | $7.2E-10$ |

Both examples show how the convergence depends on the choice of $\Theta$. If $\Theta$ is too small, then the breakdown is not cured and the accuracy is not good. For $\Theta$ too large, the spectra of $\tilde{A}$ and $A$ are different, and therefore the algorithm needs more iterations.

A serious breakdown is a very rare event. But the rank-1 modification is independent of a breakdown. Hence, to show the numerical behavior, we can also introduce a rank-1 modification without breakdown; this will be called an artificial breakdown.

In the following we introduce an artificial breakdown in the 39th step. In spite of $|d_{40}| > \varepsilon$, we use a rank-1 modification of the Lanczos algorithm as if a breakdown had occurred. Here, we take as third example a matrix $A$ with 1 as antidiagonal elements and 0.5 on the first subdiagonal, $n = 150$, and $\Theta = 1000$. For $n = 5$, $A$ looks like

$$A = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 \\ .5 & 0 & 0 & 1 & 0 \\ 0 & 0.5 & 1 & 0 & 0 \\ 0 & 1 & 0.5 & 0 & 0 \\ 1 & 0 & 0 & 0.5 & 0 \end{pmatrix}.$$

The right-hand side $b$ was chosen as a random vector of norm 1. The last column in Table 3 shows the residual with no breakdown, the other columns are the results of using an artificial breakdown for $j = 40$. So we can compare the behavior of the original and the modified Lanczos algorithm using the Galerkin method. Tables 3 and 4 show that in view of the loss of orthogonality the residual error $\tilde{A}x - b$ is large while the true residual $Ax - b$ gets small.

TABLE 3. Example 3 with artificial breakdown at $j = 40$ or no breakdown (last column), Galerkin method, $\Theta = 1000$

| $m$ | $r(\tilde{A})$ | $r(A)$ (17) | $r(A)$ (18) | $r(A)$ Gal |
|---|---|---|---|---|
| 35 | $1.12E-04$ | $1.12E-04$ | $1.12E-04$ | $1.12E-04$ |
| 40 | $1.36E-06$ | $1.25E-06$ | $1.25E-06$ | $1.82E-06$ |
| 45 | $4.70E-07$ | $5.24E-07$ | $5.14E-07$ | $1.45E-07$ |
| 50 | $1.28E-08$ | $1.72E-08$ | $1.72E-08$ | $1.91E-08$ |
| 60 | $1.30E-07$ | $4.77E-10$ | $4.77E-10$ | $4.10E-10$ |
| 70 | $1.30E-07$ | $1.28E-11$ | $1.28E-11$ | $1.70E-11$ |
| 80 | $1.30E-07$ | $7.90E-12$ | $7.89E-12$ | $3.25E-12$ |

The last example differs from the preceding example by the size $n = 2000$ and $A(n, n) = 30$. This large diagonal element is introduced to have one large eigenvalue which leads to fast convergence for the corresponding eigenvector and to an early loss of the orthogonality of the Lanczos vectors. Furthermore, we introduce two artificial breakdowns at $j = 20$ and $j = 40$. The last two columns in Table 4 are for the case of no artificial breakdowns.

TABLE 4. Example 4 with artificial breakdown at $j = 20$, 40 or no breakdown, $\Theta = 1000$

| $m$ | $r(\tilde{A})$ | $r(A)$ (18) | $r(A)$ (20) | $r(A)$ Gal | $r(A)$ QMR |
|---|---|---|---|---|---|
| 15 | $1.1E-02$ | $1.1E-02$ | $1.1E-02$ | $1.1E-02$ | $1.1E-02$ |
| 25 | $8.8E-04$ | $8.8E-04$ | $6.9E-04$ | $8.8E-04$ | $6.9E-04$ |
| 35 | $4.7E-05$ | $4.7E-05$ | $2.9E-05$ | $4.3E-05$ | $2.7E-05$ |
| 45 | $4.2E-01$ | $2.9E-05$ | $6.0E-05$ | $1.1E-05$ | $8.7E-06$ |
| 55 | $4.2E-01$ | $1.2E-06$ | $3.1E-05$ | $1.1E-06$ | $3.0E-07$ |
| 65 | $2.0E-07$ | $2.0E-07$ | $2.9E-07$ | $6.2E-09$ | $6.4E-09$ |
| 75 | $4.2E-01$ | $4.5E-09$ | $3.1E-09$ | $6.2E-09$ | $9.2E-10$ |

These numerical examples show that the low-rank modification of the modified Lanczos method and also the loss of orthogonality does not change the convergence towards the desired solution $A^{-1}b$.

## 4. CONCLUSIONS

We have introduced a low-rank modification to the unsymmetric Lanczos algorithm. This approach can be used to cure a serious breakdown. For solving linear equations $Ax = b$, in the case of a serious breakdown in the $j$th step, one has only to replace the matrix $A$ by a rank-1 modification $\tilde{A}$ and continue the Lanczos algorithm in the $(j-1)$st step with this new matrix but using only the tridiagonal matrix without rank-1 modification to compute approximations for $A^{-1}b$. This generates directly approximate solutions for the linear equation. In exact arithmetic the algorithm would compute the solution $\tilde{A}^{-1}b = A^{-1}b$. Theoretically, and by numerical examples, we show that the loss of orthogonality of the Lanczos vectors has no influence on the convergence, and that the low-rank modified algorithm has the same convergence behavior as the original Lanczos-based method. In the case of a serious breakdown we can compute the new matrix $A$ explicitly, or we keep the rank-1 term in memory and then have to compute two additional inner products and SAXPY's in every iteration step. If we want to compute eigenvalue estimates based on the modified Lanczos algorithm, we have to compute the eigenvalues of a tridiagonal matrix plus a low-rank term.

## ACKNOWLEDGMENT

This paper was written while the author was visiting Stanford University. The hospitality of Professor Gene Golub is gratefully acknowledged. The author is also indebted to Roland Freund for many useful hints and discussions. All computations are done in MATLAB.

## BIBLIOGRAPHY

1. J. Cullum and R. A. Willoughby, *A practical procedure for computing eigenvalues of large sparse nonsymmetric matrices*, Large Scale Eigenvalue Problems (J. Cullum and R. A. Willoughby, eds.), North-Holland, Amsterdam, 1986, pp. 193–240.

2. R. W. Freund, G. H. Golub, and N. M. Nachtigal, *Iterative solution of linear systems*, Acta Numerica (1991), 1–44.

3. R. W. Freund, M. H. Gutknecht, and N. M. Nachtigal, *An implementation of the look-ahead Lanczos algorithm for non-Hermitian matrices*, SIAM J. Sci. Statist. Comput. **14** (1993), 137–158.

4. R. W. Freund and N. M. Nachtigal, *QMR: a quasi-minimal residual method for non-Hermitian linear systems*, Numer. Math. **60** (1991), 315–339.

5. R. Fletcher, *Conjugate gradient methods for indefinite systems*, Numerical Analysis Dundee 1975 (G. A. Watson, ed.), Lecture Notes in Math., vol. 506, Springer, Berlin, 1976, pp. 73–89.

6. M. H. Gutknecht, *The unsymmetric Lanczos algorithms and their relations to Padé approximation, continued fractions, and the QD algorithm*, Proc. Copper Mountain Conference on Iterative Methods, Computational Math Group, The University of Colorado at Denver, 1990.

7. W. D. Jaubert, *Generalized conjugate gradient and Lanczos methods for the solution of non-symmetric systems of linear equations*, Ph.D. Dissertation, University of Texas, Austin, TX, 1990.

8. C. Lanczos, *An iteration method for the solution of the eigenvalue problem of linear differential and integral equations*, J. Res. Nat. Bur. Standards **45** (1950), 255–282.

9. ——, *Solution of systems of linear equations by minimized iterations*, J. Res. Nat. Bur. Standards **49** (1952), 33–53.

10. B. N. Parlett, *A new look at the Lanczos algorithm for solving systems of linear equations*, Linear Algebra Appl. **29** (1980), 323–346.

11. B. N. Parlett, D. R. Taylor and Z. A. Liu, *A look-ahead Lanczos algorithm for unsymmetric matrices*, Math. Comp. **44** (1985), 105–124.

12. B. N. Parlett, *Reduction to tridiagonal form and minimal realizations*, SIAM J. Matrix Anal. Appl. **13** (1992), 567–593.

13. Y. Saad, *The Lanczos biorthogonalization algorithm and other oblique projection methods for solving large unsymmetric systems*, SIAM J. Numer. Anal. **19** (1982), 485–506.

14. D. R. Taylor, *Analysis of the look ahead Lanczos algorithm*, Ph.D. Dissertation, University of California, Berkeley, CA, 1982.

15. Q. Ye, *A breakdown-free variation of the nonsymmetric Lanczos algorithm*, Math. Comp. **62** (1994), 179–207.

Institut für Angewandte Mathematik und Statistik, Universität Würzburg, Am Hubland 12, D-97074 Würzburg, Germany