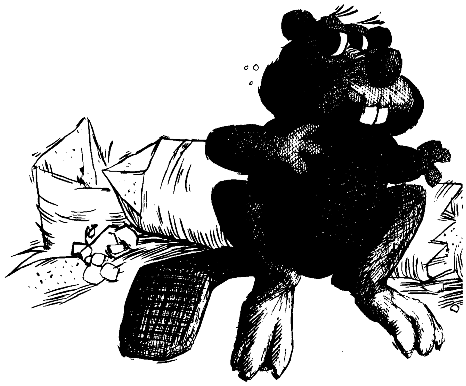
UNIVERSITY OF WATERLOO COMPUTER SCIENCE DEPARTMENT
UNIVERSITY OF WATERLOO COMPUTER SCIENCE DEPARTMENT
UNIVERSITY OF WATERLOO COMPUTER SCIENCE DEPARTMENT
UNIVERSITY OF WATERLOO COMPUTER SCIENCE DEPARTMENT



A Stabilized Algorithm for Tridiagonalization of an Unsymmetric Matrix
Research Report

Wei Pai Tang

CS-88-14

April, 1988

A stabilized algorithm for tridiagonalization of an unsymmetric matrix *

Wei Pai Tang†

April 8, 1988

Abstract

It is well known that all the algorithms for tridiagonalization of an unsymmetric matrix suffer from serious breakdown problems. Some methods, such as the unsymmetric Lanczos method, also have a stability problem. In this paper we present a new algorithm for tridiagonalization of a general matrix. The breakdown problems can be avoided before $\lfloor n/2 \rfloor$ steps. If the breakdown occurs after $\lfloor n/2 \rfloor$ steps, we can still continue this process. However, instead of reducing the matrix to a tri-diagonal form, it is reduced to a *comrade* matrix. In particular, the comrade form is invariant under deflation process[6], which is very useful in computation of several eigenvalues. Some improvements for a stable implementation are also described here. The test shows that this algorithm is stable. The eigenvalues of the reduced matrix are very good approximations of the original matrix.

Key Words. unsymmetric matrix, Hessenberg matrix, comrade matrix, eigenvalue, tridiagonalization, breakdown

1 Introduction

Algorithms for the tridiagonalization of an unsymmetric matrix are well known to have serious breakdowns. Therefore numerical analysts have ignored these algorithms for a long time. The QR method has been very popular for solving the eigenvalue problem of the unsymmetric matrix, but as the order n of the matrix increases above 100, the work for every iteration involved in QR method is $O(n^2)$. Recently the Lanczos method has been reconsidered, especially if only a few of the eigenvalues are needed.

*This research was supported by the Natural Sciences and Engineering Research Council of Canada.

†Department of Computer Science, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1.

But Lanczos method is essentially to tridiagonalize the original matrix. The problems of the instability and breakdown make Lanczos method very difficult to use in practice. Some pivoting techniques were used for reducing the chances of breakdown [4,6]. Some incomplete orthogonalization methods have been tried [5], but the breakdown problem has not been solved successfully. The key to the solution is to find a way of choosing a proper initial vector pair p and q to guarantee that the breakdown situation will not occur.

In this paper a new algorithm for tridiagonalization of a general matrix is presented. When a breakdown or a small pivot occurs before $\lfloor n/2 \rfloor$ steps, the algorithm can automatically adjust the initial vector pair to recover it. Unlike the restarting strategy this algorithm does not waste the all prior computation, only a few adjustments need to be made. If the breakdown occurs after $\lfloor n/2 \rfloor$ steps we can still continue the reduction. However, the matrix will be reduced to a *comrade* form i.e. tri-diagonal plus some nonzeroes in the last column [1]. An important fact of this comrade matrix is the invariance of the form under deflation process. Thus, we can take full advantage of the sparsity in this form during the computations of multiple eigenvalues. Several stabilization strategies are shown in this paper. The test results show that this algorithm is very stable.

The next section is the main part of the paper in which the new algorithms for the tridiagonalization of an unsymmetric matrix will be presented. In section 3 some stability considerations will be discussed.

2 The algorithm

There are several ways of viewing the tridiagonalization process. It could be thought as a biorthogonalization process of a Krylov space[5], but the algorithm presented here is based on Wilkinson's discussion of tri-diagonal reduction, with modifications to avoid breakdown.

The algorithm for reducing a general matrix to a tri-diagonal form is well known. Orthogonal transformations are used to reduce the matrix to Hessenberg form. Then, elementary Gauss transformations are applied to get the tri-diagonal structure. But this reduction process breaks down if some pivoting elements are zero. Simple interchange strategies do not work, since it will cause fill in below the sub-diagonal of the matrix. Attempts to solve this problem have proved unsuccessful.

We present here an algorithm which can postpone the breakdown after $n/2$ steps, but if such a breakdown does occur then, we also show some algorithms which can keep the process going. However, we can not reduce the matrix to a pure tri-diagonal form: it can be reduced to a *comrade*-form. In this section, we shall first present a version of the method which assume there is no small element at positions $(i+1, i)$ $i = 1, \cdots, n-1$. Then, how to remove a small pivot element $h_{i+1,i}$ will be discussed in the next section.

First, we give some notation and list some basic facts for later reference. Let

$$A = (\mathbf{a_1}, \mathbf{a_2}, \ldots, \mathbf{a_n}) = \begin{bmatrix} \tilde{\mathbf{a}}_1 \\ \tilde{\mathbf{a}}_2 \\ \vdots \\ \tilde{\mathbf{a}}_n \end{bmatrix}$$

a matrix $\in R^{n\times n}$, where $\mathbf{a}_i$ $i = 1, \cdots, n$ are the column vectors of the matrix A, while $\tilde{\mathbf{a}}_i$ $i = 1, \cdots, n$ are the row vectors. We will use $\mathbf{e}_i$ to denote the vector which all elements are zero, except the ith element,which is one, and $\mathbf{e}_i^t$ for its transpose. The following elementary matrices are used intensively in our discussion.

1. The inverse of the matrix

$$G_{i;\Pi} = I + \mathbf{e}_i \sum_{\substack{k \in \Pi \\ i < k \leq n}} \alpha_k \mathbf{e}_k^t$$

is

$$G_{i;\Pi}^{-1} = I - \mathbf{e}_i \sum_{\substack{k \in \Pi \\ i < k \leq n}} \alpha_k \mathbf{e}_k^t,$$

where Π is a subset of integers $\in [i, i+1, \cdots, n]$. Two most often used cases are

$$G_{i;j} = I + \alpha \mathbf{e}_i \mathbf{e}_j^t,$$

and

$$G_{i;(j,k)} = I + \mathbf{e}_i(\alpha_i \mathbf{e}_j^t + \alpha_k \mathbf{e}_k^t).$$

Their inverse are

$$G_{i;j}^{-1} = I - \alpha \mathbf{e}_i \mathbf{e}_j^t$$

and

$$G_{i;(j,k)}^{-1} = I - \mathbf{e}_i(\alpha_j \mathbf{e}_j^t + \alpha_k \mathbf{e}_k^t),$$

respectively.

2. $\tilde{A} = G_{i;\Pi}^{-1} A G_{i;\Pi}$ is similar to A . The effect of this similar transformation is that of adding $\alpha_i \mathbf{a}_i$ to $\mathbf{a}_k$, $k \in \Pi$, and $i \leq k \leq n$ then subtracting $\sum \alpha_k \tilde{\mathbf{a}}_k$ from $\tilde{\mathbf{a}}_i$. In particular, if the left zero profile of $\tilde{\mathbf{a}}_i$ is included in the left zero profiles of $\tilde{\mathbf{a}}_k$, $k \in \Pi$, then this transformation does not change the left zero profile of $\tilde{\mathbf{a}}_i$. Therefore, any Hessenberg matrix H will preserve the Hessenberg form under this similar transformation.

Theorem 2.1 *For any non-deflated Hessenberg matrix H, the breakdown of the tridiagonalization process can be avoided before $\lfloor n/2 \rfloor$ steps. If the breakdown or small pivot does occur after that, H can be reduced to a comrade matrix.*

Proof: Suppose

$$H = \begin{bmatrix} h_{11} & h_{12} & h_{13} & \cdots & h_{1,n-1} & h_{1n} \\ h_{21} & h_{22} & h_{23} & \cdots & h_{2,n-1} & h_{2n} \\ 0 & h_{32} & h_{33} & \cdots & h_{3,n-1} & h_{3n} \\ 0 & 0 & h_{43} & \cdots & h_{4,n-1} & h_{4n} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & & h_{n,n-1} & h_{nn} \end{bmatrix}$$

Because H is non-deflated, we know $h_{i+1,i} \neq 0 \quad i = 1, \cdots, n-1$. We may also assume that $h_{i+1,i} >> \epsilon, \quad i = 1, \cdots, n-1$. The non-zero elements $h_{i,j}, \quad j > i+1$ of H are partitioned into $i-2$ zones as the picture in the following page shows. The tridiagonalization process then is also divided into $i-2$ steps. Each step of the reduction will annihilate the non-zeros in the corresponding zone. The eliminations at each step i are divided into i stages. We have labeled the first five steps in the following matrix. The notation i^j in this matrix indicates that this location will be annihilated at j-th elimination stage of the step i. To simplify the notation, we will always use h_{ij} to denote the element of matrix $H_l^{(k)}$ where $H_l^{(k)}$ is the l-th intermediate stage at k-th step of the tridiagonalization procedure.

$$H = \left[\begin{array}{cccccccccccccccccc}
x & x & 1^1 & 2^1 & 2^1 & 3^1 & 3^1 & 4^1 & 4^1 & 5^1 & 5^1 & \cdot\cdot & \cdot\cdot & :: & :: & !! & !! & * \\
x & x & x & 2^2 & 3^2 & 3^2 & 4^2 & 4^2 & 5^2 & 5^2 & \cdot\cdot & \cdot\cdot & :: & :: & !! & !! & * & * \\
 & x & x & x & 3^3 & 4^3 & 4^3 & 5^3 & 5^3 & \cdot\cdot & \cdot\cdot & :: & :: & !! & !! & * & * & \cdot\cdot \\
 & & x & x & x & 4^4 & 5^4 & 5^4 & \cdot\cdot & \cdot\cdot & :: & :: & !! & !! & * & * & \cdot\cdot & \cdot\cdot \\
 & & & x & x & x & 5^5 & \cdot\cdot & \cdot\cdot & :: & :: & !! & !! & * & * & \cdot\cdot & \cdot\cdot & :: \\
 & & & & x & x & x & \cdot\cdot & :: & :: & !! & !! & * & * & \cdot\cdot & \cdot\cdot & :: & :: \\
 & & & & & x & x & x & :: & !! & !! & * & * & \cdot\cdot & \cdot\cdot & :: & :: & \\
 & & & & & & x & x & x & !! & * & * & \cdot\cdot & \cdot\cdot & :: & :: & & \\
 & & & & & & & x & x & x & * & \cdot\cdot & \cdot\cdot & :: & :: & & & \\
 & & & & & & & & x & x & x & \cdot\cdot & :: & :: & & & & \\
 & & & & & & & & & x & x & x & :: & & & & & \\
 & & & & & & & & & & x & x & x & & & & & \\
 & & & & & & & & & & & x & x & x & & & & \\
 & & & & & & & & & & & & x & x & x & & & \\
 & & & & & & & & & & & & & x & x & x & & \\
 & & & & & & & & & & & & & & x & x & x & \\
 & & & & & & & & & & & & & & & x & x & x \\
 & & & & & & & & & & & & & & & & x & x
\end{array}\right]$$

Our proof is constructive, namely, a sequence of similar transformations, which will annihilate the non-zero elements $h_{ij}, j > i+1$, is constructed. The procedure is similar to the one in Wilkinson's book [6] except the remedy for the breakdown and stability. The key to the success of this procedure is to use both $h_{i,i+1}$ and $h_{i+1,i}$ as pivot elements. If any of $h_{i,i+1}$ becomes zero or very small at step i during the tridiagonalization process, the element $h_{i+2,i+1}$ will be used to bring a non-zero pivot element (or normal pivot element) at position $(i, i+1)$ so that the procedure can be continued. Meanwhile, the Hessenberg form is still well preserved. The penalty of fixing a breakdown or small pivot is to bring about a few non-zero elements in the positions which have been annihilated previously. Thus, we need to eliminate these refill-ins. The cost of the reelimination is just double the cost of annihilation process at step i. Our experiences show that real breakdown has rarely happened, but the small pivots will appear now and then.

Step 1. There is is only one element needed to be annihilated. If the pivot element h_{12} is zero at this step, there are two cases: first, if all $h_{1j} = 0, \quad j = 3, \cdots, n$, then this is a deflated case. The eigenvalue problem is decomposed as two smaller problems. We simply leave this row and go to the next. If there is some $h_{1j} \neq 0$ then let $G_{1;3} = I - \mathbf{e}_1\mathbf{e}_3^t$ such that

$$H_1^{(1)} = G_{1;3}^{-1} H G_{1;3}$$

have a non-zero h_{12} element. The effect of $G_{1;3}^{-1} H G_{1;3}$ is that of adding the third row to the first row and then subtracting the first column from the third column. Furthermore, the Hessenberg form of H is well preserved. It is easy to see the new $h_{12} \neq 0$ because of $h_{32} \neq 0$. Now we are able to assume $h_{12} \neq 0$. Let

$$\alpha_{23} = -h_{13}/h_{12}$$

and

$$G_{2:3} = I + \alpha_{23}\mathbf{e}_2\mathbf{e}_3.$$

$H_1^{(1)} = G_{2;3}^{-1} H G_{2;3}$ will have a zero element at $(1, 3)$ and the step 1 is complete[1].

Step 2. At stage 1 of the step 2, let

$$G_{1;4,5} = I - \mathbf{e}_1(\alpha_4^{(2)}\mathbf{e}_4^t + \alpha_5^{(2)}\mathbf{e}_5^t),$$

where

$$\alpha_4^{(2)} = -h_{1,4}/h_{1,2}; \quad \alpha_5^{(2)} = -h_{1,5}/h_{1,2}.$$

and

$$H_1^{(2)} = G_{1;4,5}^{-1} H_1^{(1)} G_{1;4,5}$$

[1]Actually, we can eliminate all $h_{1j}, j = 3, \cdots, n$, here. But if breakdown does occur before $\lfloor n/2 \rfloor$ steps, there are possibilities of refill-ins at these positions. In order to avoid to eliminate them multiple times, we will leave some of h_{1j} for now and only annihilate these which will not be refill-ined late.

has zero elements at position $(1,4)$ and $(1,5)$. Then the stage 1 is completed. At stage 2, if h_{23} is not zero, the step 2 can be completed by letting

$$H_2^{(2)} = G_{2;4}^{-1} H_1^{(2)} G_{2;4}$$

where

$$G_{2;4} = I - \frac{h_{24}}{h_{23}} \mathbf{e}_2 \mathbf{e}_4^t.$$

But, we may also find that the element $h_{2,3}$ of the resulted $H_1^{(2)}$ is zero or very small. That means a breakdown, or risk of stability of the tridiagonalization process has taken place at this stage. Here is a way to rescue this tridiagonalization process: let

$$G_{2;3} = I + \beta \mathbf{e}_2 \mathbf{e}_4^t,$$

where $\beta \neq 0$ can be set to improve the stability, then

$$H_2^{(2)} = G_{2;3}^{-1} H_1^{(1)} G_{2;3}$$

will have a non-zero element at position $(2,3)$. But this process will also bring a non-zero element at position $(1,4)$, which was just annihilated in the previous stage. This new fill-in at position $(1,4)$ can not be eliminated by $h_{1,2}$ at this time, since that will bring back the zero at $(2,3)$. This is why the breakdown has not been resolved before. Fortunately, there is one more non-zero element at $(5,4)$ which also can be used to eliminate the non-zero at position $(1,4)$ without destroying the Hessenberg form. Let

$$G_{1;5} = I + \alpha_{1;5} \mathbf{e}_1 \mathbf{e}_5^t$$

where

$$\alpha_{1,5} = -h_{1,4}/h_{5,4}.$$

Then the element $h_{1,4}$ of

$$H_3^{(2)} = G_{1;5}^{-1} H_2^{(2)} G_{1;5}$$

is zero. It is not difficult to see that there will be a new fill-in at position $(1,5)$ and it can be annihilated by the pivot element at $(1,2)$.

At this point, the second step of reduction can be completed by letting

$$H_4^{(2)} = G_{2;4}^{-1} H_3^{(2)} G_{2;4}.$$

Step I. By induction, we can assume that this process has been successfully proceeded to stage $i-1$ of the step i since pivots $h_{j,j+1} \neq 0$,

$j = 1, \cdots, i-1$. The resulted matrix is as follows:

$$H = \begin{bmatrix}
x & x & 0 & 0 & & . & . & . & . & & & 0 & 0 & :: & :: & !! & !! & * \\
x & x & x & 0 & & . & . & . & . & & 0 & 0 & :: & :: & !! & !! & * & * \\
 & x & x & x & 0 & & . & . & & 0 & 0 & :: & :: & !! & !! & * & * & .. \\
 & & x & x & x & 0 & & & 0 & 0 & :: & :: & !! & !! & * & * & .. & .. \\
 & & & x & x & x & 0 & 0 & 0 & :: & :: & !! & !! & * & * & .. & .. & :: \\
 & & & & x & x & X & i^i & :: & :: & !! & !! & * & * & .. & .. & :: & :: \\
 & & & & & x & x & x & :: & !! & !! & * & * & .. & .. & :: & :: & * \\
 & & & & & & x & x & x & !! & * & * & .. & .. & :: & :: & * & * \\
 & & & & & & & x & x & x & * & .. & .. & :: & :: & * & * & . \\
 & & & & & & & & x & x & x & .. & :: & :: & * & * & . & . \\
 & & & & & & & & & x & x & x & :: & * & * & . & . & ! \\
 & & & & & & & & & & x & x & x & * & . & . & ! & ! \\
 & & & & & & & & & & & x & x & x & . & ! & ! & :: \\
 & & & & & & & & & & & & x & x & x & ! & :: & :: \\
 & & & & & & & & & & & & & x & x & x & :: & \\
 & & & & & & & & & & & & & & x & x & x & \\
 & & & & & & & & & & & & & & & x & x & x \\
 & & & & & & & & & & & & & & & & x & x
\end{bmatrix}$$

where X is the pivot element for row i, namely, element $h_{i,i+1}$ and i^i denotes the last element needed to be annihilated at stage i of the step i. If this element $h_{i,i+1}$ is a normal pivot, then the last stage of this step can be completed by letting

$$H_i^{(i)} = G_{i;i+2}^{-1} H_{i-1}^{(i)} G_{i;i+2}.$$

and

$$G_{i,i+2} = I - \frac{h_{i,i+2}}{h_{i,i+1}} \mathrm{e}_i \mathrm{e}_{i+2}^t.$$

The last non-zero element in the i-th zone will be annihilated after that.

If $h_{i,i+1}$ is zero or very small at this stage, we will use the same strategy in step 2 to bring a normal pivot element at $(i, i+1)$ from position $(i+2, i+1)$ to recover the breakdown. Let us first discuss the breakdown happens before $\lfloor n/2 \rfloor$. After a normal pivot is brought into the position $(i, i+1)$, there will be a refill-in at position $(i-1, i+2)$. For the same reason in step 2, we have to use $h_{i+3,i+2}$ as pivot element to annihilate this refill-in. Unfortunately, after we eliminate the refill-in at position $(i-1, i+2)$, there will be another new refill-in at position $(i-2, i+3)$. Repeatly, we are refilling in the zone i until we finally reach the row 1. After the refill-in at position $(1, 2i+1)$ is annihilated, another half of the zone i have been refilled as the following picture shows:

$$
H = \begin{bmatrix}
x & x & 0 & 0 & & . & . & . & . & & & 0 & r & :: & :: & !! & !! & * \\
x & x & x & 0 & & . & . & . & . & & 0 & r & :: & :: & !! & !! & * & * \\
& x & x & x & 0 & & . & . & & 0 & r & :: & :: & !! & !! & * & * & .. \\
& & x & x & x & 0 & & & 0 & r & :: & :: & !! & !! & * & * & .. & .. \\
& & & x & x & x & 0 & 0 & r & :: & :: & !! & !! & * & * & .. & .. & :: \\
& & & & x & x & X & i^{i} & :: & :: & !! & !! & * & * & .. & .. & :: & :: \\
& & & & & x & x & x & :: & !! & !! & * & * & .. & .. & :: & :: & * \\
& & & & & & x & x & x & !! & * & * & .. & .. & :: & :: & * & * \\
& & & & & & & x & x & x & * & .. & .. & :: & :: & * & * & . \\
& & & & & & & & x & x & x & .. & :: & :: & * & * & . & . \\
& & & & & & & & & x & x & x & :: & * & * & . & . & ! \\
& & & & & & & & & & x & x & x & * & . & . & ! & ! \\
& & & & & & & & & & & x & x & x & . & ! & ! & :: \\
& & & & & & & & & & & & x & x & x & ! & :: & :: \\
& & & & & & & & & & & & & x & x & x & :: & * \\
& & & & & & & & & & & & & & x & x & x & * \\
& & & & & & & & & & & & & & & x & x & x \\
& & & & & & & & & & & & & & & & x & x
\end{bmatrix},
$$

where r indicates the refill-in at zone i. Now, we can use the regular pivots $h_{j,j+1}, \quad j = 1, \cdots, i$ to annihilate the remaining non-zero elements in zone i.

If $i < \lfloor n/2 \rfloor$, all the refill-ins can be annihilated using both pivots $h_{i,i+1}$ and $h_{i+1,i}$. Unfortunately, when $i \geq \lfloor n/2 \rfloor$ the last "bad" refill-in does not have an extra pivot to be annihilated (see following picture).

$$
H = \begin{bmatrix}
x & x & 0 & 0 & & & . & . & . & . & . & . & . & . & & & & \\
x & x & x & 0 & & & . & . & . & . & . & . & . & . & & & & \\
& x & x & x & 0 & & & . & . & . & . & . & . & & & & & b \\
& & x & x & x & 0 & & & . & . & . & . & & & & & 0 & g \\
& & & x & x & x & 0 & & & . & . & & & & & 0 & g & :: \\
& & & & x & x & x & 0 & & & & & & & 0 & g & :: & :: \\
& & & & & x & x & x & 0 & & & & & 0 & g & :: & :: & * \\
& & & & & & x & x & x & 0 & & & 0 & g & :: & :: & * & * \\
& & & & & & & x & x & x & 0 & 0 & g & :: & :: & * & * & . \\
& & & & & & & & x & x & x & g & :: & :: & * & * & . & . \\
& & & & & & & & & x & x & x & :: & * & * & . & . & ! \\
& & & & & & & & & & x & x & x & * & . & . & ! & ! \\
& & & & & & & & & & & x & x & x & . & ! & ! & :: \\
& & & & & & & & & & & & x & x & x & ! & :: & :: \\
& & & & & & & & & & & & & x & x & x & :: & * \\
& & & & & & & & & & & & & & x & x & x & * \\
& & & & & & & & & & & & & & & x & x & x \\
& & & & & & & & & & & & & & & & x & x
\end{bmatrix}
$$

where b is the "bad" refill-in and g is the "good" refill-in which can be annihilated by the normal means. Thus, after the elimination of all "good" refill-ins, we have an extra non-zero remains unannihilated in the last column. We may leave this bad fill-in there and continue the reduction process

The resulted matrix after step i will be:

$$H = \left[\begin{array}{cccccccccccccccccc}
x & x & 0 & & & & & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & & & & \\
x & x & x & 0 & & & & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & & & & \\
 & x & x & x & 0 & & & & \cdot & \cdot & \cdot & \cdot & & & & & & b \\
 & & x & x & x & 0 & & & & \cdot & \cdot & & & & & & & 0 \\
 & & & x & x & x & 0 & & & & & & & & & & 0 & :: \\
 & & & & x & x & x & 0 & & & & & & & & 0 & :: & :: \\
 & & & & & x & x & x & 0 & & & & & & 0 & :: & :: & * \\
 & & & & & & x & x & x & 0 & & & & 0 & :: & :: & * & * \\
 & & & & & & & x & x & x & 0 & & 0 & :: & :: & * & * & \cdot \\
 & & & & & & & & x & x & x & 0 & :: & :: & * & * & \cdot & \cdot \\
 & & & & & & & & & x & x & x & :: & * & * & \cdot & \cdot & ! \\
 & & & & & & & & & & x & x & x & * & \cdot & \cdot & ! & ! \\
 & & & & & & & & & & & x & x & x & \cdot & ! & ! & :: \\
 & & & & & & & & & & & & x & x & x & ! & :: & :: \\
 & & & & & & & & & & & & & x & x & x & :: & * \\
 & & & & & & & & & & & & & & x & x & x & * \\
 & & & & & & & & & & & & & & & x & x & x \\
 & & & & & & & & & & & & & & & & x & x
\end{array}\right]$$

Then the reduction can be continued. If any breakdown or small pivot occurs, we can use the exact same procedure to recover it and after fixing the breakdown or bringing in a large pivot there will be a "bad" refill-in in the last column. The following picture lists all the possible positions for these bad fill-ins.

$$H = \left[\begin{array}{cccccccccccccccccccc}
x & x & & & & & & & & & & & & & & & & & & b \\
x & x & x & & & & & & & & & & & & & & & & & \\
 & x & x & x & & & & & & & & & & & & & & & & b \\
 & & x & x & x & & & & & & & & & & & & & & & \\
 & & & x & x & x & & & & & & & & & & & & & & b \\
 & & & & x & x & x & & & & & & & & & & & & & \\
 & & & & & x & x & x & & & & & & & & & & & & b \\
 & & & & & & x & x & x & & & & & & & & & & & \\
 & & & & & & & x & x & x & & & & & & & & & & b \\
 & & & & & & & & x & x & x & & & & & & & & & \\
 & & & & & & & & & x & x & x & & & & & & & & b \\
 & & & & & & & & & & x & x & x & & & & & & & \\
 & & & & & & & & & & & x & x & x & & & & & & b \\
 & & & & & & & & & & & & x & x & x & & & & & \\
 & & & & & & & & & & & & & x & x & x & & & & b \\
 & & & & & & & & & & & & & & x & x & x & & & \\
 & & & & & & & & & & & & & & & x & x & x & & b \\
 & & & & & & & & & & & & & & & & x & x & x & \\
 & & & & & & & & & & & & & & & & & x & x & x \\
 & & & & & & & & & & & & & & & & & & x & x
\end{array}\right],$$

It is not difficult to see that the bad fill in can only occur in the last column at every other row, when the breakdown occurs after $\lfloor n/2 \rfloor$ steps.

As mentioned above, this process also can be viewed as the readjusting of the initial vector for the biorthogonalization of the Krylov spaces. It is well known that an improper initial vector will cause breakdown[4]. It was hitherto unknown how to choose the initial vector to ensure the success of the procedure. In our process, we choose $\mathbf{e}_1$ as initial vector p and q, which may cause the breakdown. However the process of bring in the nonzero pivot and the elimination of these refill-ins is actually equivalent to readjusting the initial vector (when $2i < n$). The "good" fill-in at row 1 is the new initial vector. The elimination of the rest of the "good" fill-in is the readjusting process. But when the breakdown occurs after $\lfloor n/2 \rfloor$ steps, this kind of adjusting does not work any more. The "bad" fill-ins will remain at column n. We could not get any readjusting information from this process.

3 Stability consideration.

In the last section the breakdown is successfully postponed. Here we will discuss how to remove the small pivot elements $h_{i+1,i} \quad i = 1, \cdots, n-1$.

Let

$$H = \begin{bmatrix} x & x & x & x & x & x & x & x & x & x & x & x \\ x & x & x & x & x & x & x & x & x & x & x & x \\ & x & x & x & x & x & x & x & x & x & x & x \\ & & x & x & x & x & x & x & x & x & x & x \\ & & & x & x & x & x & x & x & x & x & x \\ & & & & \epsilon & x & x & x & x & x & x & x \\ & & & & & x & x & x & x & x & x & x \\ & & & & & & x & x & x & x & x & x \\ & & & & & & & x & x & x & x & x \\ & & & & & & & & x & x & x & x \\ & & & & & & & & & x & x & x \\ & & & & & & & & & & x & x \end{bmatrix}$$

be the resulted Hessenberg matrix after the orthogonal similar transformation. There is one small pivot at position $(i^* + 1, i^*)$. Then let

$$U = \begin{bmatrix} 1 & & & & & & & & & & & \\ & 1 & & & & & & & & & & \\ & & 1 & & & & & & & & & \\ & & & 1 & & & & & & & & \\ & & & & 1 & & & & & & & \\ & & & & & \epsilon & & & & & & \\ & & & & & & \epsilon & & & & & \\ & & & & & & & \epsilon & & & & \\ & & & & & & & & \epsilon & & & \\ & & & & & & & & & \epsilon & & \\ & & & & & & & & & & \epsilon & \\ & & & & & & & & & & & \epsilon \end{bmatrix},$$

where $u_{ii} = 1, \quad i = 1, .., i^*; u_{ii} = \epsilon, \quad i = i^* + 1, .., n; u_{ij} = 0, \quad i \neq j$, then

$$U^{-1}HU = \begin{bmatrix} x & x & x & x & x & \epsilon & \epsilon & \epsilon & \epsilon & \epsilon & \epsilon & \epsilon \\ x & x & x & x & x & \epsilon & \epsilon & \epsilon & \epsilon & \epsilon & \epsilon & \epsilon \\ & x & x & x & x & \epsilon & \epsilon & \epsilon & \epsilon & \epsilon & \epsilon & \epsilon \\ & & x & x & x & \epsilon & \epsilon & \epsilon & \epsilon & \epsilon & \epsilon & \epsilon \\ & & & x & x & \epsilon & \epsilon & \epsilon & \epsilon & \epsilon & \epsilon & \epsilon \\ & & & & x & x & x & x & x & x & x & x \\ & & & & & x & x & x & x & x & x & x \\ & & & & & & x & x & x & x & x & x \\ & & & & & & & x & x & x & x & x \\ & & & & & & & & x & x & x & x \\ & & & & & & & & & x & x & x \\ & & & & & & & & & & x & x \end{bmatrix}.$$

Thus, the small pivot is transferred to a small up-right submatrix which is safe for our reduction process.

If there are more than one small pivots in the subdiagonal such as

$$H = \begin{bmatrix} x & x & x & x & x & x & x & x & x & x & x & x \\ x & x & x & x & x & x & x & x & x & x & x & x \\ & x & x & x & x & x & x & x & x & x & x & x \\ & & x & x & x & x & x & x & x & x & x & x \\ & & & x & x & x & x & x & x & x & x & x \\ & & & & \epsilon & x & x & x & x & x & x & x \\ & & & & & x & x & x & x & x & x & x \\ & & & & & & x & x & x & x & x & x \\ & & & & & & & \epsilon & x & x & x & x \\ & & & & & & & & x & x & x & x \\ & & & & & & & & & x & x & x \\ & & & & & & & & & & x & x \end{bmatrix},$$

where two small pivots are at position $(i_1 + 1, i_1)$ and $(i_2 + 2, i_2)$. Let

$$U = \begin{bmatrix} 1 & & & & & & & & & & & \\ & 1 & & & & & & & & & & \\ & & 1 & & & & & & & & & \\ & & & 1 & & & & & & & & \\ & & & & 1 & & & & 0 & & & \\ & & & & & \epsilon & & & & & & \\ & & & & & & \epsilon & & & & & \\ & & & 0 & & & & \epsilon & & & & \\ & & & & & & & & \epsilon^2 & & & \\ & & & & & & & & & \epsilon^2 & & \\ & & & & & & & & & & \epsilon^2 & \\ & & & & & & & & & & & \epsilon^2 \end{bmatrix},$$

where $u_{ii} = 1, \quad i = 1, .., i_1; \; u_{ii} = \epsilon, \quad i = i_1, .., i_2; \; u_{ii} = \epsilon^2, \quad i = i_2{+}1, .., n;$

$u_{ij} = 0, \quad i \neq j.$

$$
U^{-1}HU = \begin{bmatrix}
x & x & x & x & x & \epsilon & \epsilon & \epsilon & \epsilon^2 & \epsilon^2 & \epsilon^2 & \epsilon^2 \\
x & x & x & x & x & \epsilon & \epsilon & \epsilon & \epsilon^2 & \epsilon^2 & \epsilon^2 & \epsilon^2 \\
 & x & x & x & x & \epsilon & \epsilon & \epsilon & \epsilon^2 & \epsilon^2 & \epsilon^2 & \epsilon^2 \\
 & & x & x & x & \epsilon & \epsilon & \epsilon & \epsilon^2 & \epsilon^2 & \epsilon^2 & \epsilon^2 \\
 & & & x & x & \epsilon & \epsilon & \epsilon & \epsilon^2 & \epsilon^2 & \epsilon^2 & \epsilon^2 \\
 & & & & x & x & x & x & \epsilon & \epsilon & \epsilon & \epsilon \\
 & & & & & x & x & x & \epsilon & \epsilon & \epsilon & \epsilon \\
 & & & & & & x & x & \epsilon & \epsilon & \epsilon & \epsilon \\
 & & & & & & & x & x & x & x & x \\
 & & & & & & & & x & x & x & x \\
 & & & & & & & & & x & x & x \\
 & & & & & & & & & & x & x
\end{bmatrix}
$$

For the case of having more than two small pivots a similar scaling can be used to remove all small pivots in the sub-diagonal. It is also worth to indicate that this kind of scaling can be done dynamically during the reduction process.

The question will naturally arise: is this algorithm stable? The answer is positive. There are many discussions about the instability of the unsymmetric Lanczos method. Let us examine the instability source carefully. Look over the algorithm given above. There are two possible sources of the instability. The easiest to find is the small pivot element. Another is as Busiger[5] pointed out : when only Gaussian elimination approach is used for reducing a matrix to a Hessenberg form , even when no small pivot elment appears, there is some possibility of exponential growth in the result. But we claim both the instable sources can be avoided in our algorithm.

The small pivot problem has been resolved in the construction of our algorithm and the previous discussion. The second source of the instability can also be dismissed. Businger gives a example which shows that if we use the Gaussian elimination technique to reduce the matrix to Hessenberg form, the element will grow exponentially. The reason for this is: When we use Gaussian approach exclusively, it causes the accumulation of the elements. This kind of danger can be avoided. It is easy to see, the Hessenberg form is not necessary before the tridiagonalization process starts. It is only needed for the convenience of discussion. In the real application we may mix the reduction of a matrix to Hessenberg form with the tridiagonalization process. By inserting the Householder transformations between the steps of our tridiagonalization process, the possibility of the exponential growth can be largely reduced. It is well known that the Householder approach will smooth the growth of the elements. Our experiences of this algorithm shows this observation is valid. The numerical tests using matlab show that this algorithm is stable. The eigenvalues of the reduced matrix are very good approximations of the original matrix.

Acknowledgment

The author wishes to thank J. Wilkinson and G. Golub for reading my first draft. Unfortunately, Jimes passed away too early to see this final work. Thanks are also due to B. N. Parlett for his encouragement of completing this report. Author also would like to express the deep appreciation to C. Moler. It is very hard to imagine how much research I can do without his matlab.

References

[1]. P.A. Businger(1969). "*Reducing a Matrix to Hessenberg form*", Math. Comp.. Vol 23 No.108 819-821.

[2]. G. H. Golub and C. F. Van Loan (1983) Matrix Computations. Johns Hopkins.

[3]. A. S. Householder (1964). "The theory of Matrix in Numerical Analysis", Ginn(Blaidell), Boston, Massachusetts.

[4]. B. N. Parlett and D. R. Taylor(1981), "*A Look Ahead Lanczos algorithm for unsymmetric matrix,*" PAM-43. University of California, Berkeley, Center for Pure and Applied Mathematics.

[5]. D. R. Taylor(1982). "*Analysis of the look ahead Lanczos Algorithm,*" PAM-108. Ph. D. Dissertation, University of California, Berkeley, Center for Pure and Applied Mathematics.

[6]. J. H. Wilkinson(1965), *The Algebraic Eigenvalue Problem* Oxford Univ. Press.