View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by NASA Technical Reports Server

RIACS

Research Institute for Advanced Computer Science
NASA Ames Research Center

A Parallel Algorithm for the Eigenvalues and Eigenvectors of a General Complex Matrix

IN-61
DATE OVERRIDE
43040
P-35

Gautam Shroff

June 1989

Research Institute for Advanced Computer Science
NASA Ames Research Center

RIACS Technical Report 89.35

NASA Cooperative Agreement Number NCC2-387

(NASA-CR-188854) A PARALLEL ALGORITHM FOR THE EIGENVALUES AND EIGENVECTORS FOR A GENERAL COMPLEX MATRIX (Research Inst. for Advanced Computer Science) 35 p CSCL 09B

N92-11653

Unclas
G3/61 0043040

A Parallel Algorithm for the Eigenvalues and Eigenvectors of a General Complex Matrix

Gautam Shroff

June 1989

Research Institute for Advanced Computer Science
NASA Ames Research Center

RIACS Technical Report 89.35

NASA Cooperative Agreement Number NCC2-387

A Parallel Algorithm for the Eigenvalues and Eigenvectors for a General Complex Matrix

Gautam Shroff†

Research Institute for Advanced Computer Science
NASA Ames Research Center

RIACS Technical Report 89.35
June 1989

A new parallel Jacobi-like algorithm is developed for computing the eigenvalues of a general complex matrix. Most parallel methods for this parallel typically display only linear convergence. Sequential 'norm-reducing' algorithms also exit and they display quadratic convergence in most cases. The new algorithm is a parallel form of the 'norm-reducing' algorithm due to Eberlein. It is proven that the asymptotic convergence rate of this algorithm is quadratic. Numerical experiments are presented which demonstrate the quadratic convergence of the algorithm and certain situations where the convergence is slow are also identified. The algorithm promises to be very competitive on a variety of parallel architectures. In particular, the algorithm can be implemented using $n^2/4$ processors, taking $O(n \log^2 n)$ time for random matrices.

---

†Ph.D. student at the Computer Science Department of Rensselaer Polytechnic Institute, Troy, N.Y. Work reported herein supported by the Office of Naval Research under Contract N00014-86-k-0610, Cooperative Agreement NCC2-387 between the National Aeronautics and Space Administration (NASA) and the Universities Space Research Association (USRA). A portion of this research was carried out while the author was a Visiting Research Associate at the Research Institute for Advanced Computer Science (RIACS), NASA Ames Research Center, Moffett Field, CA 94035.

1 Introduction:

The QR algorithm [11] is the standard method for computing the eigenvalues of a general dense matrix on a traditional sequential computer. With the advent of parallel computers, a variety of parallel eigenvalue algorithms have been proposed. For hermitian matrices, there have been two different approaches. In the first approach, the matrix is reduced to tridiagonal form as in the QR algorithm. The eigenvalues of the tridiagonal matrix are found using either the divide and conquer method [4], or multisection [14]. The other approach has been to recognize the inherent parallelism in the Jacobi method [13,10], which was the standard algorithm for the problem before the discovery of the QR algorithm. Many parallel Jacobi methods for hermitian matrices have been proposed and implemented [2,1,23] and convergence of these methods has been proved [8,16,24].

For general, non-hermitian matrices, extensions of the divide and conquer or multisection algorithms are not known. Many attempts have been made to extend the Jacobi method to the general case [22,6,5,25,3] and some of these are suited for parallel implementation [22,6,25]. However, these parallel algorithms do not possess the quadratic convergence property typical of the Jacobi method for hermitian matrices. There have also been attempts at parallelizing the QR algorithm itself [26] and work along these lines may prove to be fruitful.

In this paper we develop a parallel Jacobi-like algorithm for general complex matrices based on a method first introduced by Eberlein [5] and prove that the algorithm converges quadratically.

The rest of the paper is organized as follows. In §2 we discuss Jacobi-like methods in general. In §3 the new parallel algorithm is described in detail. In §4 a proof of ultimate quadratic convergence for the algorithm is presented. Finally in §5 experimental results are presented and analyzed.

2 Jacobi-Like Methods:

A Jacobi-like method for reducing a matrix to condensed form performs a sequence of similarity transformations

$$A_{k+1} = M_k^{-1} A_k M_k \qquad k = 0, 1, 2 \ldots . \tag{1}$$

where $A_0 = A$ is the given $n \times n$ matrix and each of the $M_k, k = 0, 1, \ldots$ is identical to the unit matrix except in the positions (p_k, q_k), (q_k, p_k), (p_k, p_k) and (q_k, q_k).

The potential for parallel implementation arises as follows. If we have a pair transformations

$$M_{k_2}^{-1} M_{k_1}^{-1} A_k M_{k_1} M_{k_2} \tag{2}$$

and the indices $p_{k_1}, q_{k_1}, p_{k_2}, q_{k_2}$ are distinct, the matrices M_{k_1} and M_{k_2} commute. Therefore the transformations can be applied in any order with the same effect. In particular, they can be applied in parallel, first to the columns (from the right), and then to the rows (from the left). For certain choices of the sequence of pairs (p_k, q_k) as many as $n/2$ transformations can be applied in parallel [2,16].

At each iteration, the transformation M_k is chosen to annihilate certain elements of the matrix A_k. The annihilations proceed in 'sweeps', where one sweep consists of a sequence of pairs (p_k, q_k) in which every off-diagonal pair $\{(p, q), 1 \leq p < q \leq n\}$ occurs exactly once.

When A is hermitian, it is possible to choose the transformation M_k to be a unitary plane rotation that annihilates the (p_k, q_k) and (q_k, p_k) elements of A_k [8]. The sequence A_k always converges to a diagonal matrix (in practice), and under certain conditions convergence can be proved [8,16,24].

If the matrix A is non-hermitian (and in general non-normal), two classes of Jacobi-like methods have been proposed. In the first type of algorithm, M_k is restricted to be a unitary plane rotation and is chosen to annihilate the (q_k, p_k) element of A_k [6,25,3]. For most matrices A_0, the sequence A_k is observed to converge to an upper triangular matrix (Schur form). Such methods will be referred to as Schur type methods. Some of these are amenable to parallel implementation [6,25].

The other class of algorithms diagonalize a diagonalizable matrix using both unitary and non-unitary transformations [5,21]. M_k is chosen to minimize the magnitude of the (p_k, q_k) and (q_k, p_k) elements of A as well as reduce $||A||$, the Euclidean (or Frobenius) norm of A. ($||A||^2 = \sum_{ij} |a_{ij}|^2$.) To explain why the norm of A is reduced, we recall the following result.

Lemma 2.1 For any square matrix A,

$$\inf_M ||M^{-1}AM||^2 = \sum_{i=1}^{n} |\lambda_i^2| \tag{3}$$

where M is non-singular and λ_i are the eigenvalues of A.

Proof: The proof is due to Mirsky [17]. Let $Q^TAQ = \Lambda + T$ be the Schur decomposition of A. (Here Λ is a diagonal matrix containing the eigenvalues of A on the diagonal, and T is a strictly upper triangular matrix.) Let $D = diag(1, \epsilon, \epsilon^2, \ldots, \epsilon^{n-1})$ where $0 < \epsilon < 1$

$$\begin{aligned} ||D^{-1}Q^TAQD||^2 &= ||\Lambda||^2 + \sum_{i,j} |t_{ij}|^2 \epsilon^{2(j-i)} \\ &\leq ||\Lambda||^2 + \epsilon^2 ||T||^2. \end{aligned}$$

Since ϵ can be an arbitrarily small positive number, it follows that the right hand side of (3) is less than the left hand side. The opposite inequality also follows by the Schur decomposition of $M^{-1}AM$, so (3) is proved. □

This result can be used to show that reducing norm of A brings it closer to a normal matrix, which can be diagonalized by unitary transformations alone. This is also equivalent to reducing the size of elements of the matrix $C = AA^* - A^*A$, the departure from normality [19]. This type of algorithm will be referred to as a norm-reducing method.

These two classes of methods differ in two important respects. First, the convergence behavior is markedly different. The norm-reducing methods display ultimate quadratic convergence for most matrices, and a proof of quadratic convergence can be obtained assuming the matrix is diagonalizable [20]. The Schur methods exhibit only

linear convergence in general [6], and no proofs of global convergence or asymptotic convergence rates are known for general matrices.

On the other hand, computing the transformation M_k in the Schur methods requires knowledge of only the (p_k, q_k), (q_k, p_k), (p_k, p_k) and (q_k, q_k) elements of A, whereas for the norm-reducing methods the entire p_k'th and q_k'th row and column of A are required. Besides being more expensive, this has a more serious consequence in that it destroys parallelism. Although the transformations M_k can still be applied in parallel as described earlier, the effect is different from applying them sequentially, since the application of one transformation affects elements used to *compute* another. As a result, it is no longer possible to show that $||A||$ is always reduced by the transformation [5], and the quadratic convergence proof [20] no longer applies.

Sameh [22] developed a parallel version of the norm reducing algorithm for real matrices [7]. Sameh showed how to compute $n/2$ transformations that could be applied in parallel to a real matrix A and would always result in a reduction of $||A||$. Sameh also proved that the algorithm converged to a normal matrix. So for real matrices, we could use Sameh's algorithm to reduce the matrix to a normal matrix and then diagonalize the resulting matrix by the Jacobi method for normal matrices [9] (which uses complex arithmetic, but converges quadratically). However, we do not know of any investigation of the rate of convergence of Sameh's algorithm to a normal matrix.

In this paper we develop a parallel algorithm for general complex matrices but take a different approach from that of Sameh, combining the norm-reducing and Schur methods. We compute the transformations M_k as in the original norm-reducing methods [5,21], except for the following differences:

1. Instead of minimizing the magnitude of the (p_k, q_k) and (q_k, p_k) elements of A, we merely annihilate the (q_k, p_k) element as in the Schur methods.

2. We stop when we have reached triangular form, as opposed to diagonal form.

3. We compute and perform transformations in parallel.

We will show that parallelism can be achieved without sacrificing the property of quadratic convergence. Further, because we only seek triangular form, the method does not try to diagonalize some obviously defective matrices unlike the original norm-reducing methods (e.g. a matrix in Jordan canonical form).

3 Description of the Algorithm

For each similarity transformation M_k in (1), there is a pair (p_k, q_k), $p_k < q_k$, that identifies the submatrix where M_k differs from the identity.

The corresponding elements of A_k will be called pivot elements and the pair (p_k, q_k) will be called the pivot of the transformation. In what follows we will omit the subscript k when it is clear that only one pivot $(p_k, q_k) = (p, q)$ is involved.

Each transformation M is one of the following three types of transformations.

Unitary Transformation:

The unitary transformation $U = \{u_{ij}\}$ with pivot (p, q) has the following structure in the positions where it differs from the identity.

$$\begin{pmatrix} u_{pp} & u_{pq} \\ u_{qp} & u_{qq} \end{pmatrix} = \begin{pmatrix} \cos x & -e^{i\theta} \sin x \\ e^{-i\theta} \sin x & \cos x \end{pmatrix},$$

where x and θ are real.

The following choice of x and θ ensures that the (q,p) element of $A' = U^T A U$ is zero [6]. Let

$$d_{pq} = (a_{qq} - a_{pp}),$$

$$d_{max} = d_{pq} \pm \sqrt{d_{pq}^2 + 4 a_{pq} a_{qp}}, \tag{4}$$

where the sign is chosen to achieve the largest absolute value. Then the parameters x and θ are given by

$$\tan x = \frac{2 e^{i\theta} a_{qp}}{d_{max}}, \tag{5}$$

where θ is chosen to make the value of $\tan x$ real.

In practice we need to bound the angle x to avoid migration of large elements from the upper triangle of the matrix to the lower part. We impose the bound

$$|\tan x| \leq 1.$$

(i.e., if $|\tan x| > 1$, we set it to 1.) Not using such a bound causes slowing of convergence in the earlier sweeps. This is similar to the bound used by Eberlein [6].

Shear Transformation:

The shear transformation $S = \{s_{ij}\}$ with pivot (p, q) has the following structure in the positions where it differs from the identity.

$$\begin{pmatrix} s_{pp} & s_{pq} \\ s_{qp} & s_{qq} \end{pmatrix} = \begin{pmatrix} \cosh y & -ie^{i\alpha} \sinh y \\ ie^{-i\alpha} \sinh y & \cosh y \end{pmatrix}, \tag{6}$$

where y and α are real.

Let $A' = S^{-1} A S$. The parameters y and α are chosen to zero the first order terms in $\partial ||A'||^2 / \partial y$ and $\partial ||A'||^2 / \partial \alpha$. It can be shown that with this choice, $||A'|| \leq ||A||$ [5].

Let

$$G_{pq} = \sum_{j \neq p,q} \{|a_{pj}|^2 + |a_{qj}|^2 + |a_{jp}|^2 + |a_{jq}|^2\}, \tag{7}$$

$$d_{pq} = (a_{qq} - a_{pp}), \tag{8}$$

$$\xi_{pq} = e^{i\alpha} a_{qp} + e^{-i\alpha} a_{pq}, \tag{9}$$

$$c_{pq} = \sum_{j=1}^{n} (a_{pj} a_{qj}^* - a_{jp}^* a_{jq}), \tag{10}$$

Note that c_{pq} is just the (p,q) element of $C = AA^* - A^*A$. Then the parameters α and y are given by

$$\alpha = \arg(c_{pq}) - \pi/2, \tag{11}$$

$$\tanh y = \frac{-|c_{pq}|}{2(|d_{pq}|^2 + |\xi_{pq}|^2) + G_{pq}}. \tag{12}$$

The effect of this transformation is also to reduce the size of the (p,q) element of C, the departure from normality [5]. Clearly, if c_{pq} is zero, the above transformation reduces to the identity. However, $||A||$ may still be non-normal with C having large diagonal elements. The following diagonal transformation reduces the size of the diagonal elements of C [21,18].

Diagonal Transformation:

The diagonal transformation D with pivot j is the identity matrix except for the jth diagonal element, which is t_j. Let

$$g_j = \left(\sum_{l \neq j} |a_{lj}|^2 \right)^{1/2},$$

$$h_j = \left(\sum_{l \neq j} |a_{jl}|^2 \right)^{1/2},$$

Then choosing

$$t_j = \sqrt{\frac{h_j}{g_j}},$$

minimizes the value of

$$\frac{g_j^2}{t_j^2} + h_j^2 t_j^2.$$

It can be shown [18] that

$$||D_j^{-1} A D_j||^2 = ||A||^2 - (g_j - h_j)^2. \tag{13}$$

Note that this transformation does not affect the diagonal elements of A, and therefore reduces the norm of the off-diagonal part of A, in addition to reducing $||A||$.

In practice we do not wish the matrices D to become too ill-conditioned, so we impose the restrictions

$$1/\tau < t_j < \tau \tag{14}$$

where τ is a constant.

Rotation: A rotation transformation R with pivot (p,q) is the composition of a shear and a unitary transformation, SU, each with the same pivot (p,q). The unitary transformation U is computed using the elements of the matrix $\tilde{A} = S^{-1}AS$.

Commuting Sequence: A sequence of pairs $(p_k, q_k) \in \{(p,q),\ 1 \leq p < q \leq n\}$ such that $p_1, q_1, p_2, q_2 \ldots p_m, q_m$ are distinct integers is called a commuting sequence of pairs. The methods described in [16,24] allow us to construct many commuting sequences where m is either $n/2$ or $(n-1)/2$.

Rotation-set: A rotation set is the composite transformation T

$$T = R_1 R_2 R_3 \ldots R_m, \tag{15}$$

where R_k is a rotation with pivot (p_k, q_k), and the sequence $\{(p_k, q_k)\}$ is a commuting sequence.

Parallel Rotation Set: In order to apply a rotation set T by computing $T^{-1}AT$ correctly according to the formulas (4) - (12), we would have to compute and apply each of the transformations R_k in sequence, since the application of one transformation affects the elements needed to compute the next. Instead, we will compute each of the transformations R_k using the original matrix A. Since the pairs (p_k, q_k) consist of distinct integers, the resulting transformations $\tilde{R}_k$ can all be computed and applied in parallel. We will call the resulting composite transformation $\tilde{T}$ a parallel rotation set with pivot sequence (p_k, q_k).

$$\tilde{T} = \tilde{R}_1 \tilde{R}_2 \ldots \tilde{R}_m. \tag{16}$$

Parallel Ordering: Let O be a sequence (p_k, q_k) of $N = n(n-1)/2$ pairs such that O is a concatenation of s commuting sequences. Further let each pair (p,q), $1 \leq p < q \leq n$ occur exactly once in the sequence O. Then O is called a parallel ordering.

Two examples of parallel orderings are shown in Fig.1. The matrices in the figure represent parallel orderings by indicating the commuting sequence to which each pair (p,q) belongs. The commuting sequences are numbered in the order they appear in the parallel ordering. These two orderings were introduced by Brent and Luk [2] and Luk and Park [16] respectively. Many other parallel orderings have been introduced as well [24,15]. The modulus ordering will figure in the quadratic convergence proof, so we define it now.

Modulus Ordering: A parallel ordering is called a modulus orderings if the pair (p,q) is in the $I(p,q)$th rotation set of the ordering, where

$$I(p,q) = 1 + (p + q - 3) \bmod n. \tag{17}$$

Sweep: One sweep of a parallel ordering O is a transformation composed of s parallel rotation sets and n diagonal transformations. Each parallel rotation set has as

$$\begin{pmatrix} x & 1 & 5 & 4 & 2 & 3 \\ & x & 3 & 2 & 5 & 4 \\ & & x & 1 & 4 & 2 \\ & & & x & 3 & 5 \\ & & & & x & 1 \\ & & & & & x \end{pmatrix} \qquad \begin{pmatrix} x & 1 & 2 & 3 & 4 & 5 \\ & x & 3 & 4 & 5 & 6 \\ & & x & 5 & 6 & 1 \\ & & & x & 1 & 2 \\ & & & & x & 3 \\ & & & & & x \end{pmatrix}$$

Brent Luk Ordering Modulus Ordering

Figure 1: Parallel orderings

its pivot sequence one of the commuting sequences that comprise the parallel ordering O. The value of s will be either n or $n-1$ depending on the parallel ordering used. For example, in the Brent-Luk ordering $s = n-1$, whereas in the Modulus ordering $s = n$. A sweep SW is defined as

$$\text{SW} = \tilde{T}_1 D_1 \tilde{T}_2 D_2 \ldots \tilde{T}_{n-1} D_{n-1} D_n \tag{18}$$

for the case $s = n-1$ and

$$\text{SW} = \tilde{T}_1 D_1 \tilde{T}_2 D_2 \ldots \tilde{T}_n D_n$$

if $s = n$. Each diagonal transformation D_p above has the pivot j_p, where $\{j_p\}$ is a permutation of $\{1 \ldots n\}$.

So at the end of a sweep, every pair (i,j) has been covered by a unitary and shear transformation, and every row and column has been covered by a diagonal transformation.

Parallel Norm Reducing Jacobi Method: We continue to perform sweeps until the following convergence criterion is met. Let

$$A_k = L_k + H_k + R_k$$

where L_k is strictly lower triangular, H_k is diagonal and R_k is strictly upper triangular. We stop when

$$||L_k|| < (n^2/2)\epsilon. \tag{19}$$

(ϵ is the value of machine precision, and the norm used is the Euclidean norm.) This is only one of many possible convergence tests that can be used. The ultimate quadratic convergence of the algorithm allows for considerable freedom in this choice.

Parallel Implementation: The parallel norm reducing Jacobi algorithm can be implemented on a square grid of $n^2/4$ processors as follows. Each processor holds four matrix elements. The $n/2$ diagonal processors are responsible for computing the rotation parameters (elements of R_k). The computation of the c_{pq}'s and G_{pq}'s is done

with each off-diagonal processor computing the required terms and sending them to the appropriate diagonal processor. The rotations are broadcast from the diagonal processors to the appropriate off-diagonal processors after which all the processors apply the rotations to the elements of the matrix. This constitutes one rotation set with the elements $a_{p_k q_k}$, $k = 1 \ldots n/2$ being the off-diagonal elements residing in the diagonal processors. Next the processor in the (1,1) position performs the appropriate diagonal transformations. Finally all the processors exchange data using the scheme given by Brent and Luk [2], and the above process is repeated $n - 1$ times. It can easily be verified that this implements a sweep as defined in (18). (Other parallel orderings [16,24] may be implemented by using an appropriate data exchange step.)

At the end of each sweep, the convergence test is carried out. The computation of $||L_k||$ can also be done in parallel. It is clear that one sweep of the algorithm takes $O(n \log n)$ time when implemented as described above (since accumulating the row and column information for each rotation set can be done in $O(\log n)$ time). In §5 it will be noted that $O(\log n)$ sweeps are required for random matrices, giving a complexity of $O(n \log^2 n)$ for the parallel algorithm.

Eigenvectors: The transformations M_k are accumulated in a matrix P. In §5 it will be noted that the final matrix A_k we obtain on convergence is always diagonal. This allows us to read off the eigenvectors from the columns of the matrix P. Accumulating the transformations can also be done in parallel along with the iteration.

4 Theorems on Quadratic Convergence:

4.1 Quadratic Convergence for Diagonalizable Matrices:

We now show that the parallel norm reducing Jacobi method converges quadratically in the later stages of the iteration. Note that this does not say anything about global convergence, but only describes the rate of convergence if the algorithm does converge. First we introduce the notation. Let A be the diagonalizable matrix we are working with and Λ a diagonal matrix containing the eigenvalues of A on the diagonal. If Z is the transformation that diagonalizes A,

$$A = Z^{-1} \Lambda Z. \tag{20}$$

We divide A into a diagonal part D and an off-diagonal part E.

$$A = D + E.$$

Let λ_i be the eigenvalues of A, let δ be the minimum separation between distinct eigenvalues of A,

$$\delta = \min_{\lambda_i \neq \lambda_j} |\lambda_i - \lambda_j|. \tag{21}$$

By the Bauer-Fike theorem [11], for each λ_j, there is an element a_{kk} of D, such that

$$|\lambda_j - a_{kk}| < ||E||. \tag{22}$$

If A is close to a diagonal matrix so that

$$||E|| < \frac{\delta}{8},$$

then it follows from (22) that each diagonal element a_{qq} of A is close to exactly one eigenvalue $\lambda(a_{qq})$ of A,

$$|a_{qq} - \lambda(a_{qq})| < ||E||.$$

We will say that a_{qq} is affiliated to the eigenvalue $\lambda(a_{qq})$.

Let

$$\gamma = \frac{||\Lambda||}{\delta}. \tag{23}$$

Theorem 1: Let the diagonalizable matrix $A = A_1$ have distinct eigenvalues. Let (20) hold. Let A_N be the matrix obtained after applying one sweep of the parallel norm reducing Jacobi algorithm to A_1, using any parallel ordering. Let E_1 and E_N be the off-diagonal parts of A_1 and A_N respectively.

Suppose A_1 has already been sufficiently diagonalized, so that

$$||E_1|| \leq \frac{3^{-2n}\delta}{36\gamma^2 n^2}, \tag{24}$$

Then,

$$||E_N|| < K_1(\Lambda, \delta, n, \tau)||E_1||^2.$$

where K_1 is indepéndent of $||E_1||$ and τ is the bound (14) on the size of the diagonal transformations.

Theorem 2: Let $A = A_1$ be a diagonalizable matrix. Let (20) hold. Let A_N be the matrix obtained after applying two sweeps of rotation sets to A_1 using a modulus ordering (17). Let E_1 and E_N be defined as in Theorem 1. Suppose A_1 has already been sufficiently diagonalized, so that

$$||E_1|| \leq \left(\frac{5}{23}\right)^{2n} \frac{\delta}{8n}. \tag{25}$$

Also let A_1 be permuted so that diagonal elements affiliated to equal eigenvalues occupy adjacent positions. Then

$$||E_N|| < K_2(||\Lambda||, \delta, n, \tau)||E_1||^2.$$

where K_2 is independent of $||E_1||$, and τ is the bound (14) on the size of the diagonal transformations.

Notice that in Theorem 2, when we allow multiple eigenvalues, we assume a particular parallel ordering while Theorem 1 holds for any parallel ordering. Further, in Theorem 1 we can bound the error after one sweep whereas in Theorem 2 we bound the error after two sweeps. The assumption that diagonal elements affiliated to equal eigenvalues are in adjacent positions can be ensured by an appropriate permutation of the matrix during the iteration.

Outline of proof: The structure of the proof is similar to proofs of quadratic convergence of other Jacobi methods [20,9]. the argument is more involved here because we have to consider the effect of performing transformations in parallel. The outline of the proof is as follows.

1. It is shown that some of the angles x and y in (5), (12) are small if

$$||E|| = \epsilon < \delta/8. \tag{26}$$

2. It is shown that (26) can be maintained throughout the sweep if $||E_0||$ is small enough.

3. It is shown that after each unitary transformation, the a_{pq} element is almost annihilated in addition to the a_{qp} element which is exactly annihilated.

4. Bounds for the entire sweep are computed using the above results.

Notation: Throughout the proof, E will refer to the off diagonal part of the matrix A, E_r to the off-diagonal part of A_r, etc..

4.2 Preliminary Lemmas:

We begin by stating some known results on the structure of an almost diagonal matrix and the effect of a similarity transformation on the norm of a matrix.

If A has been permuted so that diagonal elements affiliated to equal eigenvalues occupy adjacent positions on the diagonal, we can partition A as follows.

$$A = \begin{pmatrix} A_{11} & A_{12} & \dots & A_{1m} \\ A_{21} & A_{22} & \dots & A_{2m} \\ \dots & \dots & \dots & \dots \\ A_{m1} & A_{m2} & \dots & A_{mm} \end{pmatrix}. \tag{27}$$

Each A_{jj} has its diagonal element affiliated to equal eigenvalues.

Lemma 4.2.1: If A is partitioned as in (27) and

$$||E|| \leq \frac{\delta}{8},$$

then for $j = 1 \dots m$,

$$||E_{jj}|| \leq ||A_{jj} - \lambda_j I|| \leq \frac{1}{2}\frac{||E||^2}{\delta},$$

Proof: Refer to Wilkinson [30]. □

Lemma 4.2.2:

$$||A||^2 - ||\Lambda||^2 \leq 4\gamma||E||^2.$$

Proof: Refer to Ruhe [20]. □

Lemma 4.2.3 Let $A' = S^{-1}AS$ where S is a shear transformation defined by (6), (11) and (12). Let the quantities G_{pq}, d_{pq}, ξ_{pq} and c_{pq} be defined by (7)-(10). Let K_{pq} be defined by

$$K_{pq} = c_{pq} - (d_{pq}a^*_{qp} - d^*_{pq}a_{pq}). \tag{28}$$

Then

$$\begin{aligned} \frac{\partial||A'||^2}{\partial y} &= 2\sinh 2y G_{pq} + 4\cosh 2y Im(K_{pq}e^{-i\alpha}) \\ &+ 2\sinh 4y(|d_{pq}|^2 + |\xi_{pq}|^2) - 4\cosh 4y Im(d^*_{pq}\xi_{pq}), \end{aligned} \tag{29}$$

$$\begin{aligned} \frac{1}{\sinh 2y}\frac{\partial||A'||^2}{\partial \alpha} &= -2Re(Ke^{-i\alpha}) + 4\sinh 2y Im(a^*_{qp}a_{pq}e^{-2i\alpha}) \\ &+ 2\cosh 2y Re(d^*_{pq}a_{pq}e^{-i\alpha} - d^*_{pq}a_{qp}e^{i\alpha}). \end{aligned} \tag{30}$$

and

$$c'_{pq} = \left\{ \frac{i}{2}\frac{\partial||A'||^2}{\partial y} - \frac{1}{\sinh 2y}\frac{\partial||A'||^2}{\partial \alpha} \right\}. \tag{31}$$

Proof: Refer to Eberlein [5] for (31) and Ruhe [20] for (29) and (30). □

4.3 Bounds on angles x and y:

By the definition (12) it can easily be shown that

$$|\tanh y| \leq 1/2. \tag{32}$$

The bounds $\cosh^2 y \leq 4/3$ and $\sinh^2 y \leq 1/3$ follow. Refer to [5] for details.

The following Lemma may easily be shown using the method in [8].

Lemma 4.3.0: If $||E|| \leq \frac{\delta}{8}$ then no rotation set can cause a diagonal element to change the eigenvalue it is affiliated to in partition (27).

Proof: Refer to Forsythe and Henrici [8] and Ruhe [20]. □

Therefore we can classify the pivot elements into two sets, depending on whether or not the corresponding diagonal elements are affiliated to equal eigenvalues. Refer to the partition (27) of A. We define the subset J of the parallel ordering O as follows.

$$J = \{(p_k, q_k) | \lambda(a_{p_k p_k}) \neq \lambda(a_{q_k q_k})\}. \tag{33}$$

Lemma 4.3.1: Suppose $||E|| \leq \delta/8$. Let the pivot $(p,q) \in J$. Then the angles x and y computed by (5) and (12) satisfy,

$$|\sinh y| \leq 1.01\frac{||E||}{\delta}, \quad \cosh y \leq \left(1 + \frac{32}{63}\frac{||E||^2}{\delta^2}\right) \leq 1.01,$$

and

$$|\sin x| \leq \frac{8}{3}\frac{||E||}{\delta}.$$

Proof: Consider d_{pq} defined by (8). Since $r \in J$,

$$|d_{pq}| \geq |\lambda_{pp} - \lambda_{qq}| - |a_{pp} - \lambda_{pp}| - |a_{qq} - \lambda_{qq}| \geq \delta - 2||E|| \geq \frac{3}{4}\delta.$$

From the definition of x, (5)

$$|\tan x| = \left|\frac{2a_{qp}}{d_{max}}\right|.$$

Since d_{max}, given by (4) satisfies $|d_{max}| \geq |d_{pq}|$, and $|a_{pq}| \leq ||E||$,

$$|\sin x| \leq |\tan x| \leq \frac{8}{3}\frac{||E||}{\delta}.$$

The bounds on y are proved by Ruhe [20] as follows. From the definition (12) of y,

$$|\tanh y| = \frac{|c_{pq}|}{2(|d_{pq}|^2 + |\xi_{pq}|^2) + G_{pq}} \leq \frac{|\xi_{pq}d_{pq}| + |K_{pq}|}{2|d_{pq}|^2}, \tag{34}$$

where we have used the following relation from [5],

$$-|c_{pq}| = Im(d_{pq}^*\xi_{pq}) - Im(K_{pq}e^{-i\alpha}). \tag{35}$$

It can be easily shown using (9) that $|\xi_{pq}| \leq \sqrt{2}||E||$. By (28) and (10) we get

$$\begin{aligned} |K_{pq}| &= |\sum_{j\neq p,q} a_{pj}a_{qj}^* - a_{jp}^*a_{jq}| \\ &\leq \frac{1}{2}\sum_{j\neq p,q} |a_{pj}|^2 + |a_{qj}|^2 + |a_{jp}|^2 + |a_{jq}|^2. \end{aligned} \tag{36}$$

From this we get $|K_{pq}| \leq ||E||^2/2$.

Therefore (34) becomes

$$|\tanh y| \leq \left(\frac{2\sqrt{2}}{3} + \frac{1}{18}\right)\frac{||E||}{\delta} \leq \frac{||E||}{\delta}.$$

The remaining inequalities follow from $\cosh y = (1-\tanh^2 y)^{-\frac{1}{2}}$ and $|\sinh y| = |\cosh y \tanh y|$. □

4.4 Effect of one parallel rotation set:

Consider the parallel rotation set defined in §2 (16).

$$T = R_1 R_2 \dots R_m = S_1 U_1 S_2 U_2 \dots S_m U_m.$$

(We have dropped the $\tilde{}$ notation.) Since the pivots (p_r, q_r) for the rotation set form a commuting sequence,

$$T = S_1 S_2 \dots S_m U_1 U_2 \dots U_m. \tag{37}$$

Notation: $A = A_0$ is the matrix to which the rotation set is being applied. For $r = 1 \dots m$,

$$A_r = S_r^{-1} \dots S_1^{-1} A_0 S_1 \dots S_r = \{a_{kj}^{(r)}\} \tag{38}$$

$$\varepsilon_r = ||E_r||.$$

Further,

$$\hat{A} = U_m^{-1} \dots U_1^{-1} S_m^{-1} \dots S_1^{-1} A_0 S_1 \dots S_m U_1 \dots U_m = \{\hat{a}_{kj}^{(m)}\}' \tag{39}$$

We will use the definitions (7)-(10) and (28) for the quantities c_{pq}, G_{pq}, etc. The use of the superscript (r) or $\hat{}$ will denote that the quantity is computed using the elements of A_r or $\hat{A}$ respectively. We will need the norm

$$||A||_\Sigma = \sum_{k,j=1}^{n} |a_{kj}|.$$

Note that for any matrix A, $||A|| \le ||A||_\Sigma \le n||A||$.

Note that $d_{p_r q_r}^{(r)} = d_{p_r q_r}^{(0)}$ since the pivots (p_r, q_r) form a commuting sequence. Similarly $\xi_{p_r q_r}^{(r)} = \xi_{p_r q_r}^{(0)}$. We will drop the superscripts for these quantities, so $d_{p_r q_r} = d_{p_r q_r}^{(0)}$ etc. for all r.

In previous Lemmas we have assumed that $\varepsilon_r \le \delta/8$. However, ε_r may increase during a sweep. We need to bound the growth of ε_r during the parallel rotation set so that we can then find conditions at beginning of the sweep that will ensure that $||E|| \le \delta/8$ holds throughout the sweep. To do this we will use the $||\ ||_\Sigma$ norm defined above and bound the growth of $||E_r||_\Sigma$.

Lemma 4.4.1: If $||E_0||_\Sigma \le \delta/8$ then

$$||E_m||_\Sigma \le 4.6||E_0||. \tag{40}$$

If A has distinct eigenvalues we get the slightly sharper bound

$$||E_m||_\Sigma \le 2.8||E_0||. \tag{41}$$

Proof: We first estimate the elements of E_m that are not pivot elements in the rotation set. We define Ψ_r for all $r = 0 \ldots m$,

$$\Psi_r = \sum_{kj} |a_{kj}^{(r)}| \quad k \neq j \text{ and } (k,j) \neq (p_r, q_r) \text{ for any } r.$$

Let

$$A' = S_m^{-1} \ldots S_1^{-1} A_0. \tag{42}$$

So,

$$A_m = A' S_1 S_2 \ldots S_m.$$

Consider the element $a_{kj}^{(0)}$. It is affected by only one of the shears S_r in (42). The effect takes the form

$$a'_{kj} = \cosh y_r a_{kj}^{(0)} \pm \tilde{a}_{kj}^{(0)} e^{\pm i\alpha_r} \sinh y_r.$$

where $\tilde{a}_{kj}^{(0)}$ is some other non-pivot element of A_0. If

$$|\cosh y_r| \leq c \quad \text{and} \quad |\sinh y_r| \leq s \quad \text{for all } r,$$

$$|a'_{kj}| \leq c|a_{kj}^{(0)}| + s|\tilde{a}_{kj}^{(0)}|.$$

If Ψ' is defined analogous to Ψ_r,

$$\Psi' = \sum_{k,j} |a'_{kj}| \leq \sum_{k,j} (c|a_{kj}^{(0)}| + s|\tilde{a}_{kj}^{(0)}|),$$

where the sum taken over all non pivot pairs. But $\sum |a_{kj}| = \sum |\tilde{a}_{kj}^{(0)}|$, therefore

$$\Psi' \leq (c+s)\Psi_0.$$

By a similar analysis, $\Psi_m \leq (c+s)\Psi'$. So,

$$\Psi_m \leq (c+s)^2 \Psi_0. \tag{43}$$

We now have to consider the pivot elements $a_{p_r q_r}^{(m)}$ and $a_{q_r p_r}^{(m)}$. Consider one pivot element with $(p,q) = (p_r, q_r)$. It is affected by only one shear, $S = S_r$. By direct calculation

$$a_{pq}^{(m)} = a_{pq}^{(0)} + e^{i\alpha}\{-\frac{i}{2}\sinh 2y d_{pq} + \sinh^2 y \xi_{pq}\}.$$

and

$$a_{qp}^{(m)} = a_{qp}^{(0)} + e^{-i\alpha}\{-\frac{i}{2}\sinh 2y d_{pq} + \sinh^2 y \xi_{pq}\}.$$

Therefore

$$|a_{pq}^{(m)}| + |a_{qp}^{(m)}| \leq |a_{pq}^{(0)}| + |a_{qp}^{(0)}| + |\sinh 2y||d_{pq}| + 2|\sinh^2 y||\xi_{pq}|.$$

We substitute the value

$$\tanh y = \frac{-|c_{pq}^{(0)}|}{2(|d_{pq}|^2 + |\xi_{pq}|^2) + G_{pq}^{(0)}}.$$

for $\tanh y$ from (12). (The angle y is computed using the elements of A_0, hence the use of the (0) superscript here.) Also, we write $c_{pq}^{(0)}$ in terms of $K_{pq}^{(0)}$ and use (34). So,

$$|a_{pq}^{(m)}| + |a_{qp}^{(m)}| \leq |a_{pq}^{(0)}| + |a_{qp}^{(0)}| + \cosh^2 y \left\{ |\xi_{pq}| + \frac{|K_{pq}^{(0)}|}{|d_{pq}|} \right\} + 2\sinh^2 y |\xi_{pq}|.$$

Now we use the bounds c and s for $|\cosh y|$ and $|\sinh y|$. Also we use $|\xi_{pq}| \leq |a_{pq}^{(0)}| + |a_{qp}^{(0)}|$ which follows from (9), and $|d_{pq}| \geq 3\delta/4$.

$$|a_{pq}^{(m)}| + |a_{qp}^{(m)}| \leq (1 + c^2 + 2s^2)(|a_{pq}^{(0)}| + |a_{qp}^{(0)}|) + \frac{4c^2|K_{pq}^{(0)}|}{3\delta}. \tag{44}$$

Define Φ_r as the sum of the magnitudes of the off-diagonal pivot elements,

$$\Phi_r = \sum_r |a_{p_r q_r}^{(m)}| + |a_{q_r p_r}^{(m)}|.$$

By summing all the equations (44) for $(p, q) = (p_r, q_r)$, $r = 1 \ldots m$, we get

$$\Phi_m \leq (1 + c^2 + 2s^2)\Phi_0 + \frac{4c^2}{3\delta} \sum_r |K_{p_r q_r}^{(0)}|. \tag{45}$$

Using (36) we can show that

$$\sum_r |K_{p_r q_r}^{(0)}| \leq ||E_0||^2 \leq ||E_0||_\Sigma^2.$$

Using this, the assumption $||E_0||_\Sigma \leq \delta/8$ and adding (43) and (45) we get

$$||E_m||_\Sigma \leq (1 + c^2 + 2s^2 + c^2/6 + 2cs)||E_0||_\Sigma.$$

Inserting the bounds from (32) we get (40). If A has distinct eigenvalues we can use the bounds from Lemma 4.3.1 to get (41). □

Let ε_0 denote the value of $||E||$ at the beginning of the sweep. Let ε be the maximum value of $||E||$ throughout a sweep of N rotation sets.

Lemma 4.4.2 If $\varepsilon_0 \leq \delta/8$ then,

$$\varepsilon \leq n(4.5)^N \varepsilon_0.$$

If A has distinct eigenvalues we get the smaller bound

$$\varepsilon \leq n(2.8)^N \varepsilon_0$$

Proof: The diagonal similarity transformation cannot increase the value of $||E||$ by (13). Further, the unitary transformations cannot increase $||E||$ either (this can easily be shown using (5) and the fact that the transformations U_ν are unitary). The results then follow from Lemma 4.4.1.. □.

Now we have bounded the growth of ε_r. We will use this in §4.5 and §4.6 to show that $\varepsilon \leq \delta/8$ holds throughout the sweep. For the moment we will continue to assume it does.

The next step in the proof is the crucial step in establishing quadratic convergence. The unitary transformations are chosen so as to annihilate the pivot elements $\hat{a}_{q_r p_r}$. We need to show that in addition to this, the pivot elements $\hat{a}_{p_r q_r}$ are *almost* annihilated.

Lemma 4.4.3: Let $\varepsilon \leq \delta/8$. If the pivot pair $(p_r, q_r) \in J$, where J is defined by (33),

$$|\hat{a}_{p_r q_r}| \leq 36\gamma^2 \frac{\varepsilon^2}{\delta}$$

where γ and Λ are defined by (23) and (20).

Proof: Consider one pivot pair $(p_r, q_r) = (p, q)$. From (28) we can write

$$\hat{a}_{pq} = \frac{(\hat{c}_{pq} - \hat{K}_{pq})}{\hat{d}^*_{pq}}. \tag{46}$$

because the unitary transformation U_m is chosen to ensure that $\hat{a}_{qp} = 0$. We already have the estimates $|\hat{K}_{pq}| \leq \varepsilon^2/2$ and $\hat{d}_{pq} \geq 3\delta/4$ (since $\varepsilon \leq \delta/8$). We need to estimate $\hat{c}_{pq}$.

First consider $c^{(m)}_{pq}$. We can estimate this from Lemma 4.2.3. Using (29) and the relation $A_{r+1} = S_r^{-1} A_r S_r$,

$$\begin{aligned}\frac{\partial ||A_{r+1}||^2}{\partial y_r} &= 2\sinh 2y_r G^{(r)}_{pq} + 4\cosh 2y_r Im(K^{(r)}_{pq} e^{-i\alpha_r}) \\ &+ 2\sinh 4y_r(|d_{pq}|^2 + |\xi_{pq}|^2) - 4\cosh 4y_r Im(d^*_{pq}\xi_{pq}).\end{aligned}$$

Rewrite this as

$$\begin{aligned}\frac{\partial ||A_{r+1}||^2}{\partial y_r} &= 4(\sinh y_r(2(|d_{pq}|^2 + |\xi_{pq}|^2) + G^{(0)}_{pq}) - \cosh y_r Im(d^*_{pq}\xi_{pq})) \\ &+ 2(\sinh 4y_r - 4\sinh y_r)(|d_{pq}|^2 + |\xi_{pq}|^2) + 4(\cosh 4y_r - \cosh y_r) Im(d^*_{pq}\xi_{pq}) \\ &+ 2\sinh 2y_r G^{(r)}_{pq} + 4\cosh 2y_r Im(K^{(r)}_{pq} e^{-i\alpha_r}) - 4\sinh y_r G^{(0)}_{pq}\end{aligned}$$

The first term is of first order in y_r. We estimate it as follows.

$$\begin{aligned}4(\sinh y_r(2(|d_{pq}|^2 + |\xi_{pq}|^2) + G^{(0)}_{pq}) - \cosh y_r Im(d^*_{pq}\xi_{pq})) &= 4\cosh y_r(-|c^{(0)}_{pq}| - Im(d^*_{pq}\xi_{pq})) \\ &= 4\cosh y_r Im(-K^{(0)}_{pq} e^{-i\alpha_0})\end{aligned}$$

where we have used the value of $\tanh y_r$ from (12) and (35), noting that y_r is computed using the elements of A_0. The point to note is that this choice of y_r makes the above term $O(\epsilon^2)$. We can now use the bounds $|K_{pq}^{(r)}| \leq \epsilon^2/2$, $G_{pq}^{(r)} \leq \epsilon^2$ for all r, as well as the angle bounds from Lemma 4.3.1. We also use the relation $\delta^2 \leq 2||A_0||^2$. Finally we arrive at the estimate

$$|\frac{\partial ||A_{r+1}||^2}{\partial y_r}| \leq 27||A_0||^2 \frac{\epsilon^2}{\delta^2}. \tag{47}$$

Using (30),

$$\begin{aligned} \frac{1}{\sinh 2y_r}\frac{\partial ||A_{r+1}||^2}{\partial \alpha_r} &= -2Re(K_{pq}^{(r)} e^{-i\alpha_r}) + 4\sinh 2y\, Im(a_{qp}^* a_{pq} e^{-2i\alpha}) \\ &+ 2\cosh 2y\, Re(d_{pq}^* a_{pq} e^{-i\alpha_r} - d_{pq}^* a_{qp} e^{i\alpha_r}). \end{aligned}$$

Here all the terms are $O(\epsilon^2)$ except the last one. But recall how angle α_r is chosen (11). With this choice $Re(c_{pq}^{(0)} e^{-i\alpha_r})$ is zero. Using (28),

$$Re(c_{pq}^{(0)} e^{-i\alpha_r}) = -Re(d_{pq}^* a_{pq} e^{-i\alpha_r} - d_{pq}^* a_{qp} e^{i\alpha_r}) + Re(K_{pq}^{(0)} e^{-i\alpha_r}) = 0.$$

This gives us a second order estimate for the last term as well. Using the bounds on the angles etc. we arrive at

$$|\frac{1}{\sinh 2y_r}\frac{\partial ||A_{r+1}||^2}{\partial \alpha_r}| \leq 8||A_0||^2 \frac{\epsilon^2}{\delta^2}. \tag{48}$$

From Lemma 4.2.3,

$$|c_{pq}^{(r)}| \leq |\frac{1}{2}\frac{\partial ||A_{r+1}||^2}{\partial y_r}| + |\frac{1}{\sinh 2y}\frac{\partial ||A_{r+1}||^2}{\partial \alpha_r}|.$$

Using (47) and (48),

$$|c_{pq}^{(r)}| \leq 22||A_0||^2 \frac{\epsilon^2}{\delta^2}.$$

Recall that the pivots (p_r, q_r) form a commuting sequence. Therefore we can move any shear S_r to the mth position to get a bound on $c_{p_r q_r}^{(m)}$. Such a permutation does not affect the matrix A_m. Therefore

$$|c_{p_r q_r}^{(m)}| \leq 22|A_0||^2 \frac{\epsilon^2}{\delta^2} \text{ for all } r = 1 \ldots m.$$

Now we have to consider the effect of the unitary transformations. The only one we need to consider is U_m. To see why, consider $C_r = A_r A_r^* - A_r^* A_r$, and $\hat{C} = \hat{A}\hat{A}^* - \hat{A}^*\hat{A}$.

$$\hat{C} = U_m^* \ldots U_2^* U_1^* C_m U_1 U_2 \ldots U_m,$$

since $U_r^* U_r = I$. Let $(p,q) = (p_m, q_m)$. Since the pivots (p_r, q_r) form a commuting sequence, the only unitary transformation transformation affecting $c_{pq}^{(m)}$ is U_m. Now note that U_m does not affect the norm of the block

$$\begin{pmatrix} c_{pp} & c_{pq} \\ c_{qp} & c_{qq} \end{pmatrix}.$$

By definition (10) we have the bound $|c_{jj}| \leq \varepsilon^2$ for all j. So we can obtain the following estimate for $\hat{c}_{pq}$,

$$\hat{c}_{pq} \leq 23||A_0||^2 \frac{\varepsilon^2}{\delta^2}$$

Again, since we can have any of the pivots (p_r, q_r) in position m by a permutation that leaves $\hat{A}$ invariant, the above bound is true for all pivots (p_r, q_r).

Using (46),

$$\begin{aligned} |\hat{a}_{p_r q_r}| &\leq \frac{|\hat{c}_{pq}| + |\hat{K}_{pq}|}{|\hat{d}_{pq}^*|} \\ &\leq \frac{4}{3\delta}(23||A_0||^2 + \frac{1}{2}||A_0||^2)\frac{\varepsilon^2}{\delta^2} \\ &\leq 32||A_0||^2 \frac{\varepsilon^2}{\delta^3}. \end{aligned}$$

Now we use Lemma 4.2.2,

$$\begin{aligned} |\hat{a}_{p_r q_r}| &\leq \frac{32}{\delta^3}(||\Lambda||^2 + 4\gamma\varepsilon^2)\varepsilon^2 \\ &\leq 36\gamma^2 \frac{\varepsilon^2}{\delta} \end{aligned} \tag{49}$$

where we have used $\gamma \geq \frac{1}{2}$ and $\varepsilon \leq \delta/8$. □

4.5 Proof of Theorem 1:

Consider a sweep as defined in (18)

$$\mathrm{SW} = T_1 D_1 T_2 D_2 \ldots T_n D_n. \tag{50}$$

Here we have assumed that the parallel ordering consists of n commuting sequences. The discussion is similar if only $n-1$ commuting sequences are involved in a sweep, and the bounds derived here apply for that case as well.

From condition (24) and Lemma 4.4.2 the maximum value ε that $||E||$ can take during the sweep is

$$\varepsilon \leq n(2.8)^n \frac{3^{-2n}\delta}{36\gamma^2 n^2} \leq \frac{\delta}{8}.$$

Now suppose $(k,j) \notin P_l$ for any $l < \nu + 1$. The pair $(\hat{k},\hat{j})$ cannot be a pivot pair for the current rotation set (ν), so it is either in some P_l for $l < \nu$ or is not. If $(\hat{k},\hat{j}) \notin P_l$ for any $l < \nu$, we use (61) to get the first term of the maximum in (57). Alternatively, if $(\hat{k},\hat{j}) \in P_l$ for some $l < \nu$ we get the second term of the maximum in (57). The factor τ in (57) must be included because of the diagonal transformation, which can increase the size of an element by this factor. A similar argument leads to (59) if $(k,j) \in P_l$ for some $l < \nu + 1$.

Similarly, for iteration (52), we get (60). For (58) the first two terms in the maximum are obtained as above. We need to consider the additional possibility that $(k,j) \in P_{\nu+1}$. But the pivot elements after the rotation set are bounded by (49), from Lemma 4.4.3, which leads to the third term in (58). □

The following Lemma is similar to one proved by Hansen [12], while discussing the quadratic convergence of the Jacobi method for symmetric matrices.

Lemma 4.5.2: A solution (Ω_ν, Θ_ν) to the recurrence

$$\Omega_{\nu+1} = c_0 \max\{(c_1 + c_2)\Omega_\nu, c_1\Omega_\nu + c_2\Theta_\nu\},$$

$$\Theta_{\nu+1} = c_0 \max\{(c_1 + c_2)\Theta_\nu, c_1\Theta_\nu + c_2\Omega_\nu\},$$

$$\Theta_1 = 0,$$

where $c_i > 0$ for all i, is given by

$$\Omega_\nu = c_0^{\nu-1}(c_1 + c_2)^{\nu-1}\Omega_1 \quad \text{for } \nu = 1,2,\ldots \tag{62}$$

$$\Theta_\nu = c_0^{\nu-1}\{(c_1 + c_2)^{\nu-1} - c_1^{\nu-1}\}\Omega_1 \quad \text{for } \nu = 1,2,\ldots \tag{63}$$

Proof: The proof is by induction. The case $\nu = 1$ is easily verified. Assuming the propositions (62) and (63) are true for $\nu - 1$,

$$\begin{aligned}
\Omega_{\nu+1} &= \Omega_1 c_0^\nu \max\{(c_1 + c_2)^\nu, \\
&\qquad c_1(c_1 + c_2)^{\nu-1} + c_2((c_1 + c_2)^{\nu-1} - c_1^{\nu-1})\Big\} \\
&= \Omega_1 c_0^\nu \max\Big\{(c_1 + c_2)^\nu, (c_1 + c_2)^\nu - c_1^{\nu-1}c_2\Big\} \\
&= \Omega_1 c_0^\nu (c_1 + c_2)^\nu.
\end{aligned}$$

and

$$\begin{aligned}
\Theta_{\nu+1} &= \Omega_1 c_0^\nu \max\Big\{(c_1 + c_2)\Big\{(c_1 + c_2)^{\nu-1} - c_1^{\nu-1}\Big\}, \\
&\qquad c_1\Big\{(c_1 + c_2)^{\nu-1} - c_1^{\nu-1}\Big\} + c_2(c_1 + c_2)^{\nu-1}\Big\}, \\
&= \Omega_1 c_0^\nu \max\Big\{(c_1 + c_2)^\nu - c_1^{\nu-1}(c_1 + c_2), (c_1 + c_2)^\nu - c_1^\nu\Big\} \\
&= \Omega_1 c_0^\nu \{(c_1 + c_2)^\nu - c_1^\nu\}.
\end{aligned}$$

□

MISSING
PRECEDING PAGE [redacted] NOT FILMED

Now consider the recurrences (57)-(60). We could reduce them to two recurrences if we didn't have the third term in (58). Then if $\Omega_1 = ||E_1||$, the quantities $\Omega_{2\nu-1}$ and $\Theta_{2\nu-1}$ are upper bounds for ω_ν and θ_ν with $c_0 = \tau$, $c_1 = 1.06$ and $c_2 = 4\varepsilon/\delta$. To take care of the third term in (58) all we have to ensure is that

$$c_0^{\nu-1}(c_1 + c_2)^{\nu-1}\Omega_1 \geq 36\tau\gamma^2\frac{\varepsilon^2}{\delta} \quad \text{for all } \nu > 1.$$

Since Ω_ν is monotonically increasing with these values of c_i ($\tau \geq 1$), we only need to ensure this for $\nu = 2$. We can use $\varepsilon = n2.8^n$ by Lemma 4.4.2. We get the condition

$$||E_1|| \leq \frac{3^{-2n}\delta}{36\gamma^2 n^2}$$

which is guaranteed by (24).

Now consider θ_n, the maximum non pivot element at the end of the sweep.

$$\begin{aligned}\theta_n \leq \Theta_{2n-1} &= \tau^{2n-1}\left[\left(1.06 + \frac{4\varepsilon}{\delta}\right)^{2n-1} - 1.06^{2n-1}\right]\Omega_1 \\ &\leq \tau^{2n}\frac{12n^2 3^n}{\delta}\Omega_1\varepsilon\end{aligned}$$

where we have used $\varepsilon \leq \delta/8$. Substituting the value $\Omega_1 = ||E_1||$ and using $\varepsilon \leq \delta/8$ as well as $\varepsilon = n(2.8)^n||E_1||$ from from Lemma 4.4.2 we obtain

$$\theta_{2n-1} \leq (3\tau)^{2n}\frac{12n^3}{\delta}||E_1||^2.$$

So the non-pivot elements at the end of the sweep are bounded by the above. The remaining n off-diagonal pivot elements from the last rotation set are bounded by (49) so adding the bounds on these elements and the bound above, we get

$$||E_N|| \leq K_1(\Lambda, \gamma, \delta, \tau)||E_1||^2,$$

where

$$K_1 = \frac{48\gamma^2 n^3(3\tau)^{2n}}{\delta}||E_1||^2.$$

□

4.6 Proof of Theorem 2:

If A has multiple eigenvalues we can no longer always use the angle bounds from Lemma 4.3.1, since the pivot pairs need not belong to the set J (33). In Theorem 2 we assume that the parallel ordering used is the modulus ordering (17). We will assume that n, the order of the matrix, is even. The proof for odd n is similar.

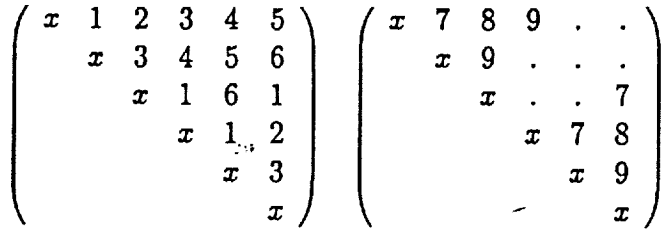

Figure 2:

Consider $2n-3$ rotation sets using a modulus ordering

$$T_1 D_1 T_2 D_2 \ldots T_{n-1} D_{n-1} D_n \ldots T_{2n-3} D_{2n-3}. \tag{64}$$

Note that we consider $2n-3$ rotation sets even though one sweep of the modulus ordering involves only n rotation sets. We are including some of the rotation sets of the next sweep. Fig.2 illustrates the rotation sets we consider for the case $n = 6$.

From (25) and Lemma 4.4.2, ϵ, the maximum value of $||E||$ during the application of two sweeps of rotation sets to A_1 ($2n$ rotation sets) is

$$\epsilon \leq n(4.6)^{2n} \left(\frac{5}{23}\right)^{(2n)} \frac{\delta}{8n} = \frac{\delta}{8}.$$

Notation: A_1 is the matrix before applying the two sweeps of rotation sets. For $\nu = 1, 2, \ldots,$

$$\tilde{A}_\nu = D_\nu^{-1} T_\nu^{-1} A_\nu, \tag{65}$$

$$\bar{A}_\nu = \tilde{A}_\nu T_\nu D_\nu, \tag{66}$$

$$A_{\nu+1} = \bar{A}_\nu.$$

We define an 'antidiagonal' χ_ν as

$$\chi_\nu = \{(k, j) | k + j = \nu\}.$$

Let P_ν be the pivot sequence used by the νth rotation set. ordering. Then it is easily shown that (refer to Fig.2)

$$P_\nu = \chi_{\nu+2} \cup \chi_{n+\nu+2} \text{ if } \nu \leq n,$$

$$P_\nu = \chi_{\nu+2} \cup \chi_{\nu-n+2} \text{ if } \nu > n.$$

We partition E_ν, the off diagonal part of A_ν as follows.

$$E_\nu = B_\nu + R_\nu + F_\nu,$$

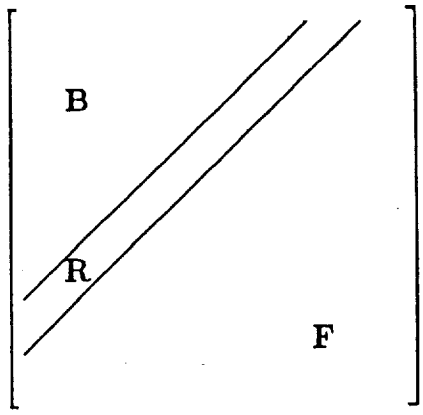

Figure 3: Case 1

$$B_\nu = \{b_{kj}^{(\nu)}\},\ R_\nu = \{r_{kj}^{(\nu)}\},\ F_\nu = \{f_{kj}^{(\nu)}\}.$$

where

$$b_{kj}^{(\nu)} = \begin{cases} a_{kj}^{(\nu)} & k+j < \nu+2 \\ 0 & \text{otherwise} \end{cases}$$

$$r_{kj}^{(\nu)} = \begin{cases} a_{kj}^{(\nu)} & k+j = \nu+2 \\ 0 & \text{otherwise} \end{cases}$$

$$f_{kj}^{(\nu)} = \begin{cases} a_{kj}^{(\nu)} & k+j > \nu+2 \\ 0 & \text{otherwise} \end{cases}$$

$\tilde{E}_\nu$ and $\bar{E}_\nu$ are similarly partitioned.

We will prove by induction on ν that there exists M_ν independent of $||E_1||$ such that

$$||B_\nu||_\Sigma \leq M_\nu ||E_1||^2. \tag{67}$$

For $\nu = 1$, $||B_1||_\Sigma = 0$ so the proposition is true. We assume it is true for ν.

Case 1: $\nu \leq n$. The partition of E_ν into B_ν, R_ν and F_ν is shown in Fig.3. First consider the pivots in $\chi_{n+\nu+2}$. It is easy to see that these pivots cannot affect the elements of B_ν, since they couple elements of F_ν among themselves. Now consider the pivots in χ_ν.

Consider the transformation (65). Consider an element b_{kj} of B_ν. It is coupled with another element $a_{\bar{k}\bar{j}}$ by a shear and a unitary transformation. Let the pivot pair of the transformation be (p, q). Let x and y be the angles involved in the unitary and shear transformations respectively.

1. $a_{\bar{k}\bar{j}} = b_{\bar{k}\bar{j}} \in B_\nu$.

Let b'_{kj} be the value of the element after the shear and unitary transformations,

$$\begin{aligned} |b'_{kj}| \ \leq\ & (|\cos x||\cosh y| + |\sin x||\sinh y|)|b_{kj}| + \\ & (|\sin x||\cosh y| + |\cos x||\sinh y|)|b_{\bar{k}\bar{j}}|. \end{aligned} \tag{68}$$

Using the angle bounds from (32) we get

$$|b'_{kj}| \leq \sqrt{3}(|b_{kj}| + |b_{\hat{k}\hat{j}}|).$$

After the diagonal transformation we get

$$|\tilde{b}_{kj}| \leq \tau\sqrt{3}(|b_{kj}| + |b_{\hat{k}\hat{j}}|). \tag{69}$$

2. $a_{\hat{k}\hat{j}} = f_{\hat{k}\hat{j}} \in F_\nu$. and $(p,q) \notin J$.
It is evident that in this case $k, \hat{k} \geq p$ and $j, \hat{j} \leq q$. Since the matrix has been permuted so that diagonal elements affiliated to equal eigenvalues occupy adjacent positions on the diagonal, it has the structure (27). Therefore b_{kj} lies in one of the diagonal blocks E_{ll} in the partition (27), and by Lemma 4.2.1

$$|\tilde{b}_{kj}| \leq \frac{1}{2}\epsilon^2. \tag{70}$$

3. $a_{\hat{k}\hat{j}} = f_{\hat{k}\hat{j}} \in F_\nu$. and $(p,q) \in J$.
We can use the angle bounds from Lemma 4.3.1 and (68) to show that

$$|b'_{kj}| \leq 1.06|b_{kj}| + \frac{4\epsilon}{\delta}|f_{\hat{k}\hat{j}}|.$$

Further, since $|f_{\hat{k}\hat{j}}| \leq \epsilon$,

$$|b'_{kj}| \leq 1.06|b_{kj}| + \frac{4\epsilon^2}{\delta}.$$

After the diagonal transformation

$$|\tilde{b}_{kj}| \leq \tau\{1.06|b_{kj}| + \frac{4\epsilon^2}{\delta}\}. \tag{71}$$

Adding (69), (70) and (71) for all $\tilde{b}_{kj}$ we get a bound for $||\tilde{B}_\nu||_\Sigma$,

$$||\tilde{B}_\nu||_\Sigma \leq \tau\left\{4.6||B_\nu||_\Sigma + n^2\epsilon^2\frac{(4+\delta)}{2\delta}\right\}.$$

An identical argument can be used to show a bound for iteration (66),

$$\begin{aligned} ||\tilde{B}_\nu||_\Sigma &\leq \tau\left\{4.6||\tilde{B}_\nu||_\Sigma + n^2\epsilon^2\frac{(4+\delta)}{2\delta}\right\}. \\ &\leq \tau^2\left\{22||B_\nu||_\Sigma + 3n^2\epsilon^2\frac{(4+\delta)}{\delta}\right\}. \end{aligned}$$

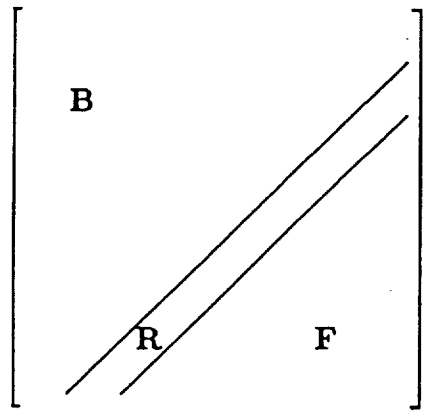

Figure 4: Case 2

Now we consider $B_{\nu+1}$. This includes elements of $\bar{R}_\nu$ as well as those of $\bar{B}_\nu$. But using Lemma 4.4.3 and accounting for the diagonal transformation,

$$\bar{r}_{kj} \leq 36\gamma^2\tau^2\frac{\epsilon^2}{\delta}$$

Adding the bound for $\bar{R}_\nu$ that we get from this to the bound on $\bar{B}_\nu$,

$$||B_{\nu+1}||_\Sigma \leq \tau^2\left\{22||B_\nu||_\Sigma + Q\epsilon^2\right\}, \tag{72}$$

where

$$Q = \left(3n^2\frac{(4+\delta)}{\delta} + 36n\frac{\gamma^2}{\delta}\right). \tag{73}$$

Case 2: $\nu > n$. Refer to Fig.4. First consider the pivots in $\chi_{\nu-n+2}$. These only can couple elements of B_ν with other elements of B_ν. Using the angle bounds from (32) it is easily seen that this can only increase the size of $||B_\nu||_\Sigma$ by a factor of 12.

The analysis for the pivots in $\chi_{\nu+2}$ is identical as that for the case $\nu \leq n$. Taking into account the factor of 12 that can arise due to the pivots in $\chi_{\nu-n+2}$ we get

$$||B_{\nu+1}||_\Sigma \leq 12\tau^2\left\{22||B_\nu||_\Sigma + Q\epsilon^2\right\}.$$

Solving this recurrence,

$$||B_\nu||_\Sigma \leq (264\tau^2)^\nu 12\tau^2 Q\epsilon^2.$$

After $2n-3$ rotation sets, $B_\nu = E_\nu$. We have considered $2n-3$ rotation sets in the sweep. We have to consider 3 more to complete two sweeps. In these 3 rotation sets, $||E||_\Sigma$ can grow by a factor of only $(4.6)^3$, by Lemma 4.4.1. Therefore, using the fact that $||E|| \leq ||E||_\Sigma$ and the value $\epsilon = n(4.6)^{2n}||E_1||$,

$$||E_N|| \leq K_2(\Lambda, \delta, n)||E_1||^2,$$

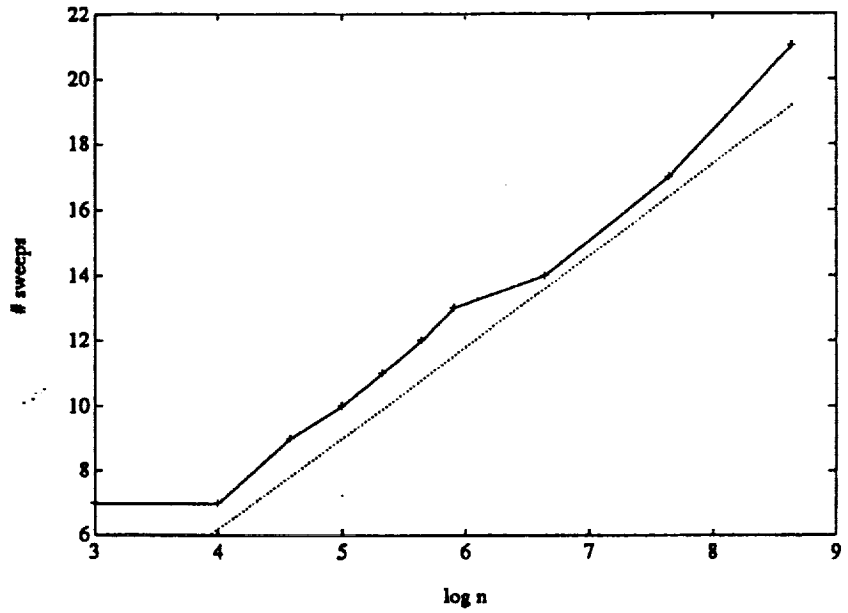

Figure 5: Results for Random matrices.

where

$$K_2 = (4.6)^3(3^7\tau^2)^{2n}12n^2\tau^2Q.$$

□.

5 Numerical Results:

We present results of experiments with the parallel norm reducing algorithm (PNRJ) for a variety of test matrices. For the convergence test (19) we use $\epsilon = 10^{-15}$, and the value $\tau = 10^8$ is used in (14) to bound the diagonal transformations. In Fig.5 results are presented for random real matrices. The number of sweeps required for convergence (according to the criterion (19)) is shown as a function of $\log_2 n$, where n is the order of the matrix. The results for random complex matrices were similar. The dotted line is a reference line of slope 2.8. So we can empirically state that algorithm PNRJ requires $2.8\log_2 n$ sweeps to converge, for random matrices.

In Fig.5 we compare the order of convergence of algorithm PNRJ with a Schur type Jacobi method, in which the non-unitary transformations are omitted (PSU). The latter is very similar to the method proposed by Eberlein [6]. We consider results

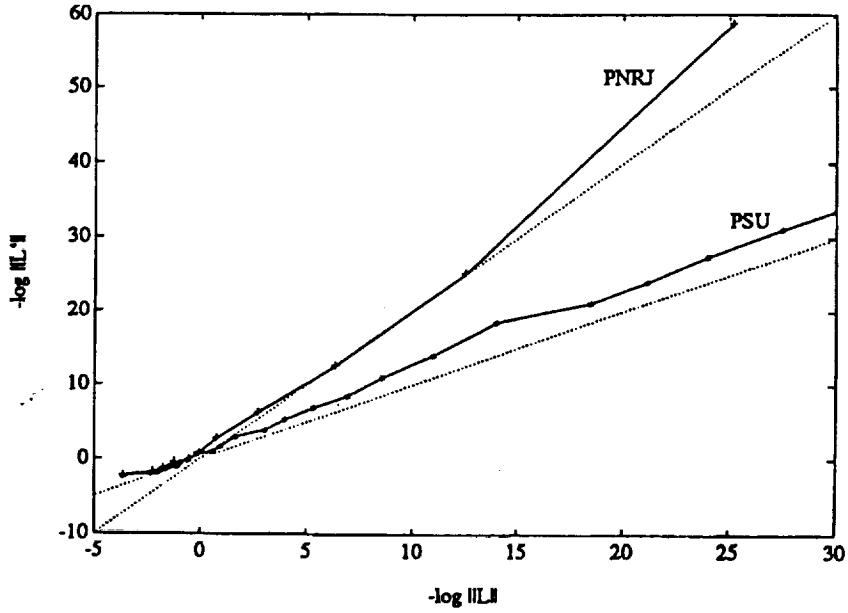

Figure 6: Order of Convergence (30 × 30 Random Matrix)

for a 30 × 30 random matrix. We plot the logarithm (base 2) of the norm of the lower triangle of the matrix before each sweep ($||L||$) against the value after each sweep ($||L'||$). The slope of the plot indicates the order of convergence. Two reference lines of slopes one and two are also shown (dotted lines). Notice that the order of convergence is two for the norm reducing method but only one for the purely unitary method.

The convergence behavior of Jacobi-like methods generally deteriorates as the matrix A becomes increasingly non-normal. We generated increasingly non-normal matrices using the method of Stewart [25]. These matrices have the form

$$A = U(D + \alpha F)U^T \tag{74}$$

where U is unitary, $D = \mathrm{diag}(1, 2, \ldots, n)$ and F is a random strictly upper triangular matrix. The parameter α controls the non-normality of the matrix A.

In Table 1 we present results for 24 × 24 Stewart matrices with different values of α. The degradation in the convergence rates is present for both PSU and PNRJ, but PNRJ degrades more slowly. In fact for $\alpha = 8$ PSU did not converge. We observed ultimate quadratic convergence for PNRJ in all cases, though the onset of this behavior is delayed as α increases.

	# sweeps	
α	PNRJ	PSU
1	8	17
2	9	28
4	12	36
8	17	∞

Table 1: Performance for 24×24 Stewart matrices.

	# sweeps	
n	PNRJ	PSU
8	12	20
12	23	47

Table 2: Performance for Frank matrices.

Frank matrices $B_n = \{b_{pq}\}$ are defined by

$$b_{pq} = \begin{cases} \min(i,j) & \text{if } j \geq i-1 \\ 0 & \text{otherwise} \end{cases} . \tag{75}$$

The smaller eigenvalues of these matrices are known to be very ill-conditioned. Results for Frank matrices are shown in Table 2.

In Table 3 we report the accuracy of the computed eigenvalues and eigenvectors. Results are shown for all of the matrices considered above. 'Max Error' is the maximum error in the computed eigenvalues (with the eigenvalues computed by EISPACK taken as the true values). P is the matrix of normalized eigenvectors computed by PNRJ, V is the matrix of normalized eigenvectors computed by EISPACK, and Λ is the diagonal matrix of eigenvalues computed by PNRJ.

We first note that the computed normalized eigenvector matrix P has condition comparable to the eigenvector matrix V obtained from EISPACK. Also, the residual $||AP - P\Lambda||_2$ is small, indicating that the computed eigenvalue-eigenvector pair is a good approximation. The error in the computed eigenvalues degrades with increasing non-normality of the matrices, as is to be expected. For the extremely ill-conditioned Frank matrix of order 12, the ill-conditioned eigenvalues differ from those computed by EISPACK by a large amount. However, it was noticed that the well-conditioned eigenvalues were computed with low error.

Although the unitary transformations (5) were chosen to annihilate only the elements a_{qp} in the lower triangle of the matrix, the final matrix was in fact very close to

A	Max Error	$\|\|AP - P\Lambda\|\|_2$	cond(P)	cond(V)
30×30 Random Matrix	3.55e-14	4.18e-14	6.46e+00	6.86e+00
24×24 Stewart Matrices				
$\alpha = 1$	1.10e-13	1.09e-13	8.81e+00	1.00e+01
$\alpha = 2$	1.49e-13	2.00e-13	4.80e+01	5.64e+01
$\alpha = 4$	1.03e-12	1.72e-11	1.13e+03	1.20e+03
$\alpha = 8$	1.56e-10	2.13e-13	5.41e+05	5.28e+05
Frank Matrices				
$n = 8$	6.06e-11	1.07e-12	7.89e+03	8.41e+03
$n = 12$	1.64e-06	9.80e-14	1.48e+08	1.49e+08

Table 3: Accuracy of Computed Eigenvalues and Eigenvectors.

diagonal in all the cases we tried. This is to be expected for diagonalizable matrices since the norm-reducing transformations move the matrix toward a normal matrix. However, even for nearly defective matrices, the algorithm actually diagonalizes a nearby diagonalizable matrix, so the final matrix is still diagonal, except that the computed eigenvalues have larger errors. Therefore, in this algorithm the eigenvectors can be read off from the columns of the matrix obtained by accumulating the transformation matrices T_r. This is of course not so for the Schur type methods.

6 Discussion and Conclusions:

We are unable to prove global convergence of algorithm PNRJ. The difficulty comes from the fact that we cannot prove that the parallel algorithm always causes a decrease in $||A||$, as well as from the cyclic choice of the pivots. The global convergence proof for Eberlein's norm reducing method [5] as well as that for Sameh's parallel algorithm for real non-symmetric matrices relies on the provable norm reduction property as well as an optimal choice of pivots.

Optimal pivoting strategies (which require searching the elements of the commutator C at each iteration) are difficult to implement efficiently and may not be worth doing merely to obtain the global convergence proof, when cyclic strategies perform just as well in practice.

Regarding the reduction in $||A||$, although we are unable to prove it in general for algorithm PNRJ, we observed that in all the experiments described in the previous section, the norm $||A||$ never increased after a rotation set and invariably decreased (except during the last few sweeps, when the matrix was already normal). Therefore

ORIGINAL PAGE IS
OF POOR QUALITY

we conclude that the question of global convergence of algorithm PNRJ is still open.

Various implementation issues remain to be investigated in order to improve the algorithm further. In particular, the broadcasting of information is an undesirable feature because communication of this sort is costly in all parallel computers. However, if we consider the proof of quadratic convergence we notice that the global information does not play a significant role. Therefore we could consider not computing global information during the later iterations which would reduce the cost of computation as well as communication in the later stages of the iteration. Experiments along these lines are currently in progress. Another issue that needs to be considered is a block version of the algorithm similar to block Jacobi methods for the hermitian eigenvalue problem [1,24].

In spite of using non unitary transformations, PNRJ computes eigenvalues and eigenvectors with accuracy comparable to that of EISPACK. This feature is inherited from the norm reducing methods on which PNRJ is based [5,21]. To explain the stability of these methods, we can argue that the iterates A_k are moving closer to a normal matrix and therefore their eigenproblem is becoming successively better conditioned. Further, as long as only small norm similarity transformations are used, the final matrix will be exactly similar to a matrix that is close to the original matrix [29]. However, a formal error analysis proving the stability of norm-reducing methods does not appear to have been carried out, and is an area open for further investigation.

To compare algorithm PNRJ with the QR algorithm let us consider operation counts. The single shift hessenberg QR algorithm for general complex matrices [11] computes the Schur decomposition in approximately $26n^3$ complex floating point operations. Here we have used the empirical observation that usually about 3 iterations are required to decouple one eigenvalue. On the other hand, on a sequential machine, PNRJ requires about $9n^3$ operations for *one sweep*. So for random matrices, PNRJ is slower than the QR algorithm by a factor of about $\log_2 n$. However on a parallel computer the situation is not so clear since efficient parallel implementations of the QR algorithm (like $O(n \log^2 n)$ time using $O(n^2)$ processors) are not known. This is an area of ongoing research.

As described in the previous section, algorithm PNRJ slows down as we move to matrices that are increasingly non normal. Although the QR algorithm also displays this degradation, the effect of non normality is much less pronounced. Understanding the effect of non normality on algorithm PNRJ in more detail is also an important topic for research.

Finally, a version of the algorithm that uses only real arithmetic for real matrices is desirable. Real Jacobi-like methods that converge quadratically for certain classes of matrices have been investigated by Veselic [27,28]. Sameh's algorithm [22] is a parallel algorithm for real matrices. Work is in progress to develop a quadratically convergent parallel algorithm for real matrices.

To summarize, we have introduced algorithm PNRJ, a parallel Jacobi like algo-

rithm that diagonalizes a diagonalizable matrix. It has been proved that the algorithm converges quadratically for any parallel ordering of the pivots if the matrix has distinct eigenvalues. In the presence of multiple eigenvalues quadratic convergence can be shown if a particular parallel ordering, the modulus ordering, is used to choose the pivots.

Experimental results confirm the ultimate quadratic convergence of the algorithm, though the onset of quadratic convergence is delayed with increasing non-normality of the matrices. For nearly defective matrices the well conditioned eigenvalues are computed accurately, though the convergence is also delayed in this case. For random matrices the algorithm exhibits an improved rate of convergence compared to some other parallel Jacobi methods for the unsymmetric eigenvalue problem.

Algorithm PNRJ can be implemented using $n^2/4$ processors and performs one 'sweep' in $O(n \log n)$ time. For random matrices, it is empirically observed that $O(\log n)$ sweeps are necessary for convergence.

7 Acknowledgements:

I wish to thank Rob Schreiber for valuable advice and suggestions and for reading the manuscript with great care.

References

[1] Christian Bischof. *Computing the Singular Value Decomposition on a Distributed System of Vector Processors.* Technical Report 87-869, Department of Computer Science, Cornell University, 1987.

[2] R.P. Brent and F.T. Luk. The solution of singular-value and symmetric eigenvalue problems on multiprocessor arrays. *SIAM J. Sci. Stat. Comp.*, 6(1):69–84, 1985.

[3] R.L. Causey. Computing eigenvalues of non hermitian matrices by methods of Jacobi type. *J. SIAM*, 6:172–181, 1958.

[4] J.J Dongarra and D.C. Sorensen. A fully parallel algorithm for the symmetric eigenvalue problem. *SIAM J. Sci. Stat. Comp.*, 8(2):138–154, 1987.

[5] P.J. Eberlein. A Jacobi method for the automatic computation of eigenvalues and eigenvectors of an arbitrary matrix. *J. SIAM*, 10:74–88, 1962.

[6] P.J. Eberlein. On the Schur decomposition of a matrix for parallel computation. *IEEE Trans. Comp.*, 36:167–174, 1987.

[7] P.J. Eberlein and Boothroyd J. Solution to the eigenproblem by a norm-reducing Jacobi-type method. *Num. Math.*, 4:24–40, 1968.

[8] G.E. Forsythe and P. Henrici. The cyclic Jacobi method for computing the principal values of a complex matrix. *Trans. Amer. Math. Soc.*, 94:1–23, 1960.

[9] H.H. Goldstine and Horwitz L.P. A procedure for the diagonalization of normal matrices. *J. ACM*, 6:176–195, 1959.

[10] H.H. Goldstine, F.J. Murray, and J. von Neumann. The Jacobi method for real symmetric matrices. *J. ACM*, 6:59–96, 1959.

[11] G. Golub and C. Van Loan. *Matrix Computations.* Johns Hopkins University Press, 1983.

[12] E.R. Hansen. *On Jacobi methods and block Jacobi methods for computing matrix eigenvalues.* PhD thesis, Stanford University, 1960.

[13] C.G.J. Jacobi. Über ein verfarhen, die in theorie der säkularstörungen vorkommenden gleichungen numerisch aufzulösen. *J. Reine Angew. Math.*, 30:51–95, 1846.

[14] Sy-Shin Lo, Bernard Philippe, and Ahmed Sameh. A multiprocessor algorithm for the symmetric eigenvalue problem. *SIAM J. Sci. Stat. Comp.*, 8(2):155–165, 1987.

[15] F.T. Luk and H.T. Park. *On Parallel Jacobi Orderings.* Technical Report EE-CEG-86-5, School of Electrical Engineering, Cornell University, 1986.

[16] F.T. Luk and H.T. Park. *A Proof of Convergence for two Parallel Jacobi SVD Algorithms.* Technical Report EE-CEG-86-12, School of Electrical Engineering, Cornell University, 1986.

[17] L. Mirsky. On the minimization of matrix norms. *Amer. Math. Monthly*, 65:106–107, 1958.

[18] E.E. Osborne. On pre-conditioning of matrices. *J. ACM*, 7:338–345, 1960.

[19] A. Ruhe. The norm of a matrix after a similarity transformation. *BIT*, 9:53–58, 1969.

[20] A. Ruhe. On the quadratic convergence of a generalization of the Jacobi method to arbitrary matrices. *BIT*, 8:210–231, 1968.

[21] H. Rutishauser. Une methode pour le calcul des values propres des matrices non symetriques. *Comptes Rendus*, 259:2758, 1964.

[22] A.H. Sameh. On Jacobi and Jacobi-like algorithms for a parallel computer. *Math. Comp.*, 25(115):579–590, 1971.

[23] R. Schreiber. Solving eigenvalue and singular value problems on an undersized systolic array. *SIAM J. Sci. Stat. Comp.*, 7(2):441–451, 1986.

[24] G. Shroff and R. Schreiber. *On the Convergence of the Cyclic Jacobi Method for Parallel Block Orderings.* Technical Report RPI-CS-88-11, Computer Science Dept., Rensselaer Polytechnic Institute, 1988. (To appear in SIAM J. Matrix Anal. Appl.).

[25] G.W. Stewart. A Jacobi-like algorithm for computing the schur decomposition of a nonhermitian matrix. *SIAM J. Sci. Stat. Comp.*, 6:853–864, 1985.

[26] R. van de Geijn. *Implementing the QR-Algorithm on an array of processors.* PhD thesis, University of Maryland, College Park, 1987.

[27] K. Veselic. On a class of Jacobi-like procedures for diagonalizing arbitrary real matrices. *Numer. Math.*, 33:157–172, 1979.

[28] K. Veselic. A quadratically convergence Jacobi-like method for real matrices with complex conjugate eigenvalues. *Numer. Math.*, 33:425–435, 1979.

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

[30] J.H. Wilkinson. A note on the quadratic convergence of the cyclic Jacobi process. *Numer. Math.*, 4:296–300, 1962.

PAGE INTENTIONALLY BLANK