MONITORING THE STAGE OF DIAGONALIZATION IN JACOBI–TYPE METHODS

Jürgen Götze

Institute of Network Theory and Circuit Design
Technical University of Munich
Arcisstr. 21, 80333 Munich, Germany

ABSTRACT

Since the stage of diagonalization of Jacobi–type methods is difficult to monitor in a parallel environment, it is usually proposed to execute a predetermined number of sweeps (iterations) on a parallel processor array. A possibility for monitoring the stage of diagonalization is essential in order to avoid the execution of a significant number of unnecessary sweeps. Based on a Lemma used for a generalized proof of the quadratic convergence of the Jacobi EVD and SVD methods a new criteria for monitoring the stage of diagonalization is derived. Using this criteria it can easily be monitored when the stage of quadratic convergence is reached (only one bit yields this information). Therefore, only the (small) number of quadratically convergent sweeps must be predetermined. A further similiar criteria particularly useful for Jacobi–type methods using CORDIC–based approximate rotations is also given.

1. INTRODUCTION

Jacobi's algorithm (JA) and Kogbetliantz's algorithm (KA) are the methods of choice for the fast parallel solution of eigenvalue and singular value problems, since they offer a significantly higher degree of parallelism than the respective QR–methods. Therefore, parallel implementations of these algorithms have been presented by several authors [1, 3, 16, 4]. For the KA it is advantageous to execute a QR–decomposition as a preparatory step and applying the KA to the resulting upper triangular matrix (called TKA in the sequel) [15, 14].

The complexity of the parallel implementations is mainly determined by the complexity of the rotation evaluations. Therefore, different strategies for modifying the rotation evaluations have been presented:

M1 approximate rotations [16, 3]

M2 factorized rotations [7, 18, 12]

M3 CORDIC processors [4, 5]

M4 Combination of the modifications M1–M3:

- **M1+M2** factorized approximate rotations [10, 9]
- **M1+M3** CORDIC–based approximate rotations [11]

Which of these modified schemes yields the most efficient parallel implementation depends on the particular parallel processor.

The global and the ultimate quadratic convergence of the JA [6, 19] and the TKA [14, 13] have been proved. In this paper we give the theorems and the outline of new proofs for the global and the ultimate quadratic convergence which hold for the JA *as well as* for the TKA and include the case of using approximate rotations (the proofs of the exact methods are obtained as special cases). Once an efficient modified rotatition scheme (M1–M4) that is well adapted to the particular parallel computer has been choosen, the given theorems make it easy to check for the global and the quadratic convergence of the designed modified JA/TKA.

For the parallel implementation of the JA/TKA it is usually proposed to execute a predetermined number of sweeps [1, 2, 15, 3] (e.g 10 sweeps for $n \leq 100$), since it is very difficult to monitor the stage of diagonalization of the matrix by the usually applied off–diagonal norm. Executing a predetermined number of sweeps means, however, that in general a significant number of unnecessary sweeps is executed. Based on a Lemma (Lemma 3.4) used for the proof of the quadratic convergence an alternative stopping criteria amenable to parallel implementations is derived in this paper. By this criteria it can be monitored when the stage of quadratic convergence is reached, such that only the number of quadratically convergent sweeps must be predetermined. The number of quadratically convergent sweeps is small (usually $1, \ldots, 3$ sweeps are sufficient) compared to the overall required number of sweeps. Therefore, the number of unnecessary sweeps can significantly be reduced by using the given criteria. Furthermore, a similiar criteria is derived, which can be used for the adaption of the accuracy of the CORDIC–based approximate rotations [11].

Section 2 gives a very brief review of the JA/TKA. In section 3 we restrict ourself to explain the main idea for getting proofs that hold for the JA *as well as* for the TKA. All Theorems and Lemmas are formulated. However, only the proofs for the Lemmas which are essential for the derivations in section 4 are given (for detailed proofs of the theorems see [10]). Section 4 deals with the determination of the required number of sweeps of the JA/TKA executed in a parallel environment.

2. JACOBI'S AND KOGBETLIANTZ'S ALGORITHM

In the sequel the matrix $\mathbf{A}$ always describes a $n \times n$ symmetric matrix in the case of the JA. In the case of the TKA the matrix $\mathbf{A}$ always describes a $n \times n$ upper triangular matrix. The cyclic–by–row JA/TKA works by applying a sequence of orthogonal rotations to the left and right of $\mathbf{A}$ ($\mathbf{A}^{(1)} = \mathbf{A}$):

$$\text{for } k = 1, 2, \ldots$$

$$\text{JA: } \mathbf{A}^{(k+1)} = \mathbf{J}^T(p, q, \theta_k)\mathbf{A}^{(k)}\mathbf{J}(p, q, \theta_k) \quad (1)$$

$$\text{TKA: } \mathbf{A}^{(k+1)} = \mathbf{J}^T(p, q, \phi_k)\mathbf{A}^{(k)}\mathbf{J}(p, q, \psi_k) \quad (2)$$

where $\mathbf{J}(p, q, \varphi)$ is a plane rotation by angle φ in the (p, q) plane and the index pairs are choosen in a cyclic manner, e.g. cyclic–by–row

$$(p, q) = (1, 2)(1, 3) \ldots (1, n)(2, 3) \ldots (2, n) \ldots (n - 1, n) \quad (3)$$

The execution of all $N = n(n-1)/2$ index pairs (p,q) according to (3) is called a sweep.

For the JA the rotation $\mathbf{J}(p,q,\theta_k)$ is computed such that $a_{pq}^{(k+1)} (= a_{qp}^{(k+1)}) = 0$, i.e. with $t_\theta^{(k)} = \tan\theta_k$, $c_\theta^{(k)} = \cos\theta_k$ and $s_\theta^{(k)} = \sin\theta_k$ [8]:

$$\sigma^{(k)} = \frac{2a_{pq}^{(k)}}{a_{qq}^{(k)} - a_{pp}^{(k)}} \tag{4}$$

$$t_\theta^{(k)} = \frac{\sigma^{(k)}}{1 + \sqrt{1 + (\sigma^{(k)})^2}} \tag{5}$$

$$c_\theta^{(k)} = \frac{1}{\sqrt{1 + t_\theta^{(k)}}} \quad ; \quad s_\theta^{(k)} = t_\theta^{(k)} c_\theta^{(k)} . \tag{6}$$

The embedding of $(c_\theta^{(k)}, s_\theta^{(k)}, -s_\theta^{(k)}, c_\theta^{(k)})$ in the (pp, pq, qp, qq) positions of a $n \times n$ identity matrix yields $\mathbf{J}(p,q,\theta_k)$.

For the TKA the rotations $\mathbf{J}(p,q,\phi_k)$ and $\mathbf{J}(p,q,\psi_k)$ are computed such that $a_{qp}^{(k)} = 0$ is preserved ($a_{qp}^{(k+1)} = 0$) and $a_{pq}^{(k+1)} = 0$ is obtained [14]:
Case 1 ($a_{pp}^{(k)} \geq a_{qq}^{(k)}$):

$$\sigma_1^{(k)} = \frac{a_{qq}^{(k)} a_{pq}^{(k)}}{\left(a_{pp}^{(k)}\right)^2 - \left(a_{qq}^{(k)}\right)^2 + \left(a_{pq}^{(k)}\right)^2} \tag{7}$$

$$t_\phi^{(k)} = \frac{-\sigma_1^{(k)}}{1 + \sqrt{1 + (\sigma_1^{(k)})^2}} \tag{8}$$

$$t_\psi^{(k)} = \frac{a_{qq}^{(k)} t_\phi^{(k)} - a_{pq}^{(k)}}{a_{pp}^{(k)}} \tag{9}$$

Case 2 ($a_{qq}^{(k)} > a_{pp}^{(k)}$):

$$\sigma_2^{(k)} = \frac{a_{pp}^{(k)} a_{pq}^{(k)}}{\left(a_{qq}^{(k)}\right)^2 - \left(a_{pp}^{(k)}\right)^2 + \left(a_{pq}^{(k)}\right)^2} \tag{10}$$

$$t_\psi^{(k)} = \frac{\sigma_2^{(k)}}{1 + \sqrt{1 + (\sigma_2^{(k)})^2}} \tag{11}$$

$$t_\phi^{(k)} = \frac{a_{pp}^{(k)} t_\psi^{(k)} + a_{pq}^{(k)}}{a_{qq}^{(k)}} \tag{12}$$

The cosine–sine pairs for $t_\phi^{(k)}$ and $t_\psi^{(k)}$ are defined as in (6).

Defining the off–diagonal quantity $S^{(k)}$ (a slightly different definition is used for the JA ($\varrho = 1/2$) and the TKA ($\varrho = 1$) with respect to a generalized description) by

$$S^{(k)} = \sqrt{\varrho \left[\|\mathbf{A}^{(k)}\|_F^2 - \sum_{i=1}^{n} \left(a_{ii}^{(k)}\right)^2 \right]} \tag{13}$$

the execution of (1) or (2) yields:

$$\left[S^{(k+1)}\right]^2 = \left[S^{(k)}\right]^2 - \left[\left(a_{pq}^{(k)}\right)^2 - \left(a_{pq}^{(k+1)}\right)^2\right] \tag{14}$$

Obviously, the maximal reduction of $S^{(k)}$ is obtained if $a_{pq}^{(k+1)} = 0$, what is obtained by the above computations of the rotation parameters.

For a reduction of the off–diagonal quantity $S^{(k)}$, however, it is not necessary to meet $a_{pq}^{(k+1)} = 0$ but it is sufficient that

$$\left|a_{pq}^{(k+1)}\right| = |d| \cdot \left|a_{pq}^{(k)}\right| \text{ with } 0 \leq |d| < 1 \tag{15}$$

where

$$|d(t,\sigma)| = \left| \frac{1 - 2t/\sigma - t^2}{1 + t^2} \right| \tag{16}$$

Note, that in the case of the JA $a_{qp}^{(k+1)} = a_{qp}^{(k+1)}$ always holds while in the case of the TKA $a_{qp}^{(k+1)} = 0$ must always be met in order to preserve the upper triangular structure.

This is the basis for using approximate rotations [16, 3, 10]. It is convenient to derive the approximate rotations from the tangent of the rotation angle instead of the the cosine–sine pair. One of many possible approximations that meets $|d| < 1$ for the JA is using $\tilde{t}_\theta^{(k)} = \sigma^{(k)}$ instead of (5). It is always possible to use the same approximations as for the JA for the TKA [10]. Here, one must distinguish between using case 1 if $a_{pp}^{(k)} \geq a_{qq}^{(k)}$ and case 2 if $a_{qq}^{(k)} > a_{pp}^{(k)}$ in order to guarantee $|d| < 1$.

3. GLOBAL AND ULTIMATE QUADRATIC CONVERGENCE

In this section the theorems for the global and the ultimate quadratic convergence of the JA/TKA using approximate rotations are given. The known results for the exact schemes are obtained as special cases ($d = 0$). Furthermore, the generalized proofs of these theorems [10] hold for the TKA as well as for the JA. This is obtained by assuming, that during the TKA and the JA the matrices remain 'essential triangular' [14]. Although for the JA $\mathbf{A}^{(k)^T} = \mathbf{A}^{(k)}$ actually holds, we only use one essential triangular part, such that the JA also proceeds as shown in figure 2.1.1 of [14] for the TKA. This agrees with our definitions of $S^{(k)}$ in (13) (JA: $\varrho = 1/2$; TKA: $\varrho = 1$).

Theorem 3.1 (Global Convergence) If $0 \leq |d| < 1$ holds throughout the TKA *and* the JA, then the column (and row) cyclic schemes of these algorithms are always convergent.

Proof: This proof is modeled after the proof of the global convergence of the TKA using exact rotations [14]. For details of the proof see [10]. □

Theorem 3.2 (Ultimate Quadratic Convergence) Let

$$|\sigma_i - \sigma_j| \geq 2\delta \qquad (i \neq j) \tag{17}$$

and suppose we have reached the stage r, where

$$S^{(r)} < \frac{\delta}{4} \tag{18}$$

then for some ($k > r$) and with $d(t,\sigma) \to 0$ for $\sigma \to 0$

$$S^{(k+N)} \leq \frac{\gamma}{\sqrt{1 - d_{max}^2}} \cdot \frac{\left[S^{(k)}\right]^2}{\delta} + O\left(\left[S^{(k)}\right]^3\right) \tag{19}$$

holds, where $\gamma = 7/6$ for the TKA and $\gamma = 1$ for the JA.

For the proof of Theorem 3.2 the following Lemmas are needed.

Lemma 3.3 If (17) and (18) hold, then for some ordering of the σ_i

$$||a_{ii}^{(r)}| - \sigma_i| < \frac{\delta}{2} \tag{20}$$

and

$$||a_{ii}^{(r)}| - |a_{jj}^{(r)}|| > \delta \tag{21}$$

hold. Furthermore, if $a_{ii}^{(r)} \geq a_{jj}^{(r)}$ holds, then

$$|a_{ii}^{(r)}| > \frac{3}{2}\delta \tag{22}$$

Proof: For the proofs of (20) and (21) see Paige and van Dooren [17] (singular values) and Wilkinson [19] (eigenvalues). The proof of (22) is similiar to the proof of (21). With (17) and (20) we have

$$|a_{ii}^{(r)}| \geq |\sigma_i - \sigma_j| - ||a_{ii}^{(r)}| - \sigma_i| > 2\delta - \frac{\delta}{2} = \frac{3}{2}\delta \quad \square$$

Lemma 3.4 For $k > r$ we obtain the following estimates

$$|\sigma_1^{(k)}| < \frac{|a_{pq}^{(k)}|}{2\delta} \;\; ; \;\; |\sigma_2^{(k)}| < \frac{|a_{pq}^{(k)}|}{2\delta} \;\; ; \;\; |\sigma^{(k)}| < \frac{|a_{pq}^{(k)}|}{\delta}$$

and thereby

$$|\sigma_1^{(k)}| < 1/4 \; ; \qquad |\sigma_2^{(k)}| < 1/4 \; ; \qquad |\sigma^{(k)}| < 1/2$$

Proof: For the TKA case 1 we have with (21):

$$\begin{aligned} |\sigma_1^{(k)}| &= \frac{|a_{pq}^{(k)}|}{\left(a_{pp}^{(k)} - a_{qq}^{(k)}\right)\frac{a_{pp}^{(k)} + a_{qq}^{(k)}}{a_{qq}^{(k)}} + \frac{(a_{pq}^{(k)})^2}{a_{qq}^{(k)}}} \\ &< \frac{|a_{pq}^{(k)}|}{\left(a_{pp}^{(k)} - a_{qq}^{(k)}\right)\cdot 2} < \frac{|a_{pq}^{(k)}|}{2\delta} \end{aligned}$$

With (18) we have $|a_{pq}^{(k)}| \leq S^{(k)} \leq S^{(r)} < \delta/4$ from which $|\sigma_1^{(k)}| < 1/4$ follows. For the TKA case 2 and the JA we obtain the upper bounds in a similiar way. $\square$

Proof of Theorem 3.2: Following Paige and van Dooren [17] or Wilkinson [19] ($x_i := a_{pq}^{(i)}$) with $|x_{i+1}| = |d_i x_i|$ (instead of $x_{i+1} = 0$) and using the above Lemmas yields the proof (see [10]). $\square$

We do not distinguish between the different σ ((4), (7), (10)) in the following, since the derivations hold for the JA as well as for the TKA. Lemma 3.4 shows that for $|\sigma|$ less than the given upper bounds the JA/TKA works in the region of quadratic convergence. Therefore, for $|\sigma|$ less than these values an approximation must be used that meets $d(t,\sigma) \rightarrow 0$ for $\sigma \rightarrow 0$ in order to preserve the ultimate quadratic convergence. For $|\sigma|$ greater than these values it is sufficient that the approximation fullfills $0 \leq |d(t,\sigma)| < 1$. This can be used to extend the amount of possible approximations [10] and also for monitoring the stage of diagonalization as shown in the following section.

	$n = 10$		$n = 20$	
ℓ	$S^{(\ell)}$	$\|\sigma^{(\ell)}\|_{max}$	$S^{(\ell)}$	$\|\sigma^{(\ell)}\|_{max}$
0	9.76e+00	–	2.05e+01	–
1	2.58e+00	1.79e+01	5.16e+00	6.97e+01
2	4.19e–01	1.92e+00	1.91e+00	9.16e+01
3	2.72e–02	1.15e+00	4.54e–01	2.62e+01
4	4.71e–05	2.88e–02*	1.42e–02	3.40e–01*
5	2.20e–10	9.27e–05	2.08e–05	1.16e–02
6	9.02e–16	1.00e–10	3.12e–11	8.32e–06
7			2.54e–15	4.77e–11

Table 1: $S^{(\ell)}$ and $|\sigma^{(\ell)}|_{max}$ after sweep ℓ

4. MONITORING THE STAGE OF DIAGONALIZATION

Usually the stopping criteria for the JA/TKA is a small off–diagonal quantity $S^{(k)}$. It is difficult, however, to monitor this quantity in a parallel environment, since the computation of $S^{(k)}$ requires the collection of data distributed over the entire matrix/processor array. Therefore, it is usually proposed to execute a predetermined number of sweeps on a parallel machine [1, 2, 3, 15, 10].

Here, we propose a different criteria for monitoring the stage of diagonalization. The respective value is the maximum of the values $|\sigma^{(k)}|$ of sweep ℓ of the JA/TKA:

$$|\sigma^{(\ell)}|_{max} = \max_k |\sigma^{(k)}| \text{ where } k = (\ell-1)N + 1 : \ell N \tag{23}$$

According to Lemma 3.4 the stage of quadratic convergence is reached if

$$\text{JA: } |\sigma^{(\ell)}|_{max} < 1/2 \quad \text{TKA: } |\sigma^{(\ell)}|_{max} < 1/4 \tag{24}$$

Executing $1, \ldots, 3$ (depending on the required accuracy) additional quadratically convergent sweeps after (24) is met is sufficient to reach the stopping condition (a sufficiently small off–diagonal quantity).

While the value of the off–diagonal quantity $S^{(\ell)}$ after sweep ℓ is hard to compute in a parallel environment (distributed data), it is easy to check if (24) is met. The values $\sigma^{(k)}$ are computed in the diagonal processors in order to evaluate the rotations. Therefore, each diagonal cell i must only set a binary value $flag_i = 1$ if (23) is not met for any of the rotations evaluated in the respective sweep. With $flag = \cup_i flag_i$ the information if the JA/TKA has reached the stage of quadratic convergence ($flag = 0$) is obtained.

In table 1 the values of $S^{(\ell)}$ and $|\sigma^{(\ell)}|_{max}$ obtained after each sweep of the JA are shown for random symmetric matrices of dimension $n = 10, 20$. The sweep in which (23) is first met is indicated by $*$. Obviously, $2, 3$ additional sweeps are sufficient after quadratic convergence is reached.

Monitoring the stage of diagonalization by the given criteria does not allow to determine the required number of sweeps exactly. The required number of sweeps is determined by the number of sweeps until reaching quadratic convergence and a predetermined (small) number of additional quadratically convergent sweeps. Therefore, the number of executed unnecessary sweeps is significantly reduced compared to working with a predetermined number of sweeps for the complete Jacobi method.

Monitoring the stage of diagonalization is particularly necessary for the Jacobi–type method using CORDIC–based approximate rotations [11] in order to increase the accuracy during the

ℓ	$n = 10$		$n = 20$	
	$S^{(\ell)}$	$\|\sigma^{(\ell)}\|_{mean}$	$S^{(\ell)}$	$\|\sigma^{(\ell)}\|_{mean}$
0	1.00e+01	–	2.09e+01	–
1	1.84e+00	4.84e+00	6.08e+00	4.92e+00
2	1.69e–01	2.53e–01	1.68e+00	3.37e+00
3	3.75e–03	1.48e–02*	4.06e–01	1.35e–01
4	8.70e–07	1.46e–04	6.69e–03	6.34e–02
5	8.39e–15	1.32e–08	1.15e–05	2.84e–04*
6			9.46e–12	1.56e–07

Table 2: $S^{(\ell)}$ and $|\sigma^{(\ell)}|_{mean}$ after sweep ℓ.

course of the algorithm. A further value for monitoring the stage of diagonalization is derived with respect to this method. It is the average value of $|\sigma^{(k)}|$ of sweep ℓ of the JA/TKA:

$$|\sigma^{(\ell)}|_{mean} = \frac{1}{N} \sum_{k} \sigma^{(k)} \text{ where } k = (\ell - 1)N + 1 : \ell N \quad (25)$$

The $|\sigma^{(k)}|$ are not decreasing in a monotone manner as $S^{(k)}$. Therefore, in contrary to $S^{(k)}$ the values $|\sigma^{(k)}|$ do not reflect the stage of diagonalization. If a complete sweep is considered, however, $|\sigma^{(\ell)}|_{mean}$ shows a similiar behaviour as $S^{(\ell)}$ (at least for increasing ℓ). Therefore, it can be used as a measure for the stage of diagonalization together with (23).

In table 2 the values of $S^{(\ell)}$ and $|\sigma^{(\ell)}|_{mean}$ obtained after each sweep of the JA are shown for random symmetric matrices of dimension $n = 10, 20$. $|\sigma^{(\ell)}|_{mean}$ shows a similiar behaviour as $S^{(\ell)}$. The sweep in which $|\sigma^{(\ell)}|_{max}$ meets (24) for the first time is again indicated by $*$.

In order to preserve the quadratic convergence of the Jacobi–type methods using CORDIC–based approximate rotations it is necessary to increase the accuracy of the approximation (number of CORDIC angles) during the course of the algorithm [11]. $|\sigma^{(\ell)}|_{max}$ and $|\sigma^{(\ell)}|_{mean}$ (reflected by the maximal and the average of the shift values) can be used to control this adaptation of the number of CORDIC angles. Using an adaptive version of the CORDIC–based approximate rotations results in a significant reduction of the complexity of the respective Jacobi methods.

5. CONCLUSIONS

In this paper two criteria for monitoring the stage of diagonalization in Jacobi–type methods have been presented. In contrary to the usual criteria (the off–diagonal quantity $S^{(k)}$) these criteria are well suited for parallel implementations. Furthermore, they are particularly useful for designing adaptive CORDIC–based approximate rotations (adaption of the accuracy of the approximation) in order to preserve the ultimate quadratic convergence in the Jacobi–type methods based on these CORDIC–based approximate rotations.

6. REFERENCES

[1] R.P. Brent and F.T. Luk. The Solution of Singular Value and Symmetric Eigenvalue Problems on Multiprocessor Arrays. *SIAM J. Sci. Stat. Comput.*, 6:69–84, 1985.

[2] R.P. Brent, F.T. Luk, and C. van Loan. Computation of the Singular Value Decomposition Using Mesh Connected Processors. *J. VLSI Computer Systems*, 1:242–270, 1985.

[3] J.P. Charlier, M. Vanbegin, and P. van Dooren. On Efficient Implementations of Kogbetliantz's Algorithm for Computing the Singular Value Decomposition. *Numer. Math.*, 52:279–300, 1988.

[4] J.-M. Delosme. Bit-Level Systolic Algorithm for the Symmetric Eigenvalue Problem. In *Proc. Int. Conf. on Application Specific Array Processors*, pages 771–781, Princeton (USA), 1990.

[5] M.D. Ercegovac and T. Lang. Redundant and On–Line CORDIC: Application to Matrix Triangularization and SVD. *IEEE Trans. on Computers*, 39:725–740, 1990.

[6] G. Forsythe and P. Henrici. The Cyclic Jacobi Method for Computing the Pricipal Values of a Complex Matrix. *Trans. Amer. Math. Soc.*, 94:1–23, 1960.

[7] W.M. Gentleman. Least Squares Computations by Givens Rotations Without Square Roots. *J. Inst. Maths Applics*, 12:329–336, 1973.

[8] G.H. Golub and C.F. van Loan. *Matrix Computations*. The John Hopkins University Press, second edition, 1989.

[9] J. Götze. Parallel Methods for Iterative Matrix Computations. In *Proc. IEEE Int. Symp. on Circuits and Systems*, pages 233–236, Singapore, 1991.

[10] J. Götze. On the Parallel Implementation of Jacobi's and Kogbetliantz's Algorithm. Technical Report YALEU/DCS/RR-879, Dept. of Computer Science, Yale University, New Haven (USA), 1991 (*to appear SIAM J. Sci. & Stat. Comput.*).

[11] J. Götze, S. Paul, and M. Sauer. An Efficient Jacobi–Like Algorithm for Parallel Eigenvalue Computation. *to appear IEEE Trans. on Computers*, 1993.

[12] J. Götze and U. Schwiegelshohn. A Square Root and Division Free Givens Rotation for Solving Least Squares Problems on Systolic Arrays. *SIAM J. Sci Stat. Comput.*, 12:800–807, 1991.

[13] V. Hari. On the Quadratic Convergence of the Serial Singular Value Decomposition Jacobi Methods for Triangular Matrices. *SIAM J. Sci. Stat. Comput.*, 10:1076–1096, 1989.

[14] V. Hari and K. Veselic. On Jacobi Methods for Singular Value Decomposition. *SIAM J. Sci. Stat. Comput.*, 8:741–754, 1987.

[15] F.T. Luk. A Triangular Processor Array for Computing Singular Values. *Lin. Alg. and Its Applic.*, 77:259–273, 1986.

[16] J.J. Modi and J.D. Pryce. Efficient Implementation of Jacobi's Diagonalization Method on the DAP. *Numer. Math.*, 46:443–454, 1985.

[17] C.C. Paige and P. van Dooren. On the Quadratic Convergence of Kogbetliantz's Algorithm for Computing the Singular Value Decomposition. *Lin. Alg. and Its Applic.*, 77:301–313, 1986.

[18] W. Rath. Fast Givens Rotations for Orthogonal Similarity Transformations. *Numer. Math.*, 40:47–56, 1982.

[19] J.H. Wilkinson. Note on the Quadratic Convergence of the Cyclic Jacobi Process. *Numer. Math.*, 4:296–300, 1962.