A NEW ALGORITHM FOR THE EIGENVALUE PROBLEM OF MATRICES*

CAI DA-YONG HONG JING
(*Department of Applied Mathematics, Tsinghua University, Beijing, China*)

Abstract

In this paper, a new algorithm for the eigenvalue problem of matrices is given. Namerical examples show that it could be a remarkable approach for practical purposes. Some open problems are listed.

§1. Introduction

The prevalent technique for obtaining the eigenvalue of matrices are based on either (a) a factorisation of the matrix A into special factors (say LU, QR) leading to a matrix sequence $\{A_k\}$, which is isospectral with the matrix A and, in a sense, tends to some limit while k goes to infinity, or (b) Jacobi-like methods, power method and others.

To solve a system of linear equations, a new factorization and splitting procedure (QIF) is proposed in [1], which is more convenient for parallel computation. Recently, a more detailed analysis for this factorization is given in [2].

Based on QIF, an algorithm for the eigenvalue problem of matrices is given in this paper, which is essentially a block LU facforization. In §2, the QIF procedure is briefly introduced. §3 is devoted to the algorithm description. The proof of convergence is given in §4. Several numerical examples are presented in §5. They show that the algorithm here could be an attractive method. Finally, some open problems are listed in §6.

§2. QIF Factorization

Let A be an $n \times n$ matrix. Now we consider a factorization of the form:

$$A = WZ \tag{2.1}$$

* Received December 19, 1987.

where W and Z have the matrix form:

$$W = \begin{bmatrix} 1 & & \mathbf{0} & 0 \\ W_{2,1} & 1 & & W_{2,n} \\ \vdots & W_{3,2} & \mathbf{1} & \vdots \\ W_{n-1,1} & & \mathbf{0} & W_{n-1,n} \\ 0 & & & 1 \end{bmatrix}, Z = \begin{bmatrix} z_{1,1} & z_{1,2} & \cdots & & z_{1,n} \\ & z_{2,2} & & z_{2n-1} & \\ \mathbf{0} & & z_{i,i} & & \mathbf{0} \\ & & & & \\ z_{n,1} & & & & z_{n,n} \end{bmatrix} \tag{2.2}$$

where the elements of W and Z are given by

$$w_{i,j} = \begin{cases} 1, & i = j, \\ w_{i,j}, & (i,j) \in D, \\ 0, & \text{otherwise} \end{cases} \tag{2.3}$$

and

$$z_{i,j} = \begin{cases} 0, & (i,j) \in D, \\ z_{i,j}, & \text{otherwise}, \end{cases} \tag{2.4}$$

$$D = \{(i,j) | (i = 2, \cdots, [(n+1)/2] \text{ and } j = i-1) \text{ or}$$
$$(i = [(n+1)/2], \cdots, (n-1) \text{ and } j = i+1)\}. \tag{2.5}$$

By substituting (2.2)–(2.3) into (2.1) and comparing corresponding terms of the matrices A and WZ, we have

(i) The elements of the first and last rows of Z are given immediately by

$$z_{1,i} = a_{1,i} \quad \text{and} \quad z_{n,i} = a_{n,i}, \quad i = 1, 2, \cdots, n.$$

(2) Then the sets of $n \times n$ linear systems given by:

$$\begin{aligned} z_{1,1} w_{i,1} + z_{n,1} w_{i,n} &= a_{i,1}, \\ z_{1,n} w_{i,1} + z_{n,n} w_{i,n} &= a_{i,n} \end{aligned} \tag{2.6}$$

are solved to get the values of $w_{i,1}$ and $w_{i,n}$ for $i = 1, 2, \cdots, n-1$. This completes the first stage and calculation of the outermost ring of matrices W and Z. The remaining elements of W and Z are computed in a similar way. Totally $(n-1)/2$ such steps are needed to compute matrices W and Z.

In [2], a necessary and sufficient condition for this procedure without permutations is presented. A pivot strategy is discussed there.

§3. QIF Algorithm Description

Let $A_1 = A$. The algorithm is similar to the QR method except that QR transformation is replaced by the QIF factorisation method in §2.

Denote

$$A_k = W_k Z_k \tag{3.1}$$

where W_k and Z_k have the form as in (2.2);

$$A_{k+1} = Z_k W_k. \tag{3.2}$$

Assume A_k is a nonsingular matrix, and so are W_k and Z_k.

Now, our algorithm can be written as follows:

(1) $A_1 = A$.

(2) For $k = 1, 2, \cdots$, do,

$A_k = W_k Z_k$,

$A_{k+1} = Z_k W_k = (a_{i,j}^{(k+1)})$.

(3) If $\max_{(i,j)\in D} |a_{i,j}^{(k+1)}| > \varepsilon$, then goto (2) else goto (4)

where ε is a given accuracy tolerance, say 10^{-5}, and D is shown in (2.5).

(4) Let $L = [n/2], G = \begin{bmatrix} z_{i,i} & z_{i,n-i+1} \\ z_{n-i+1} & z_{n-i+1,n-i+1} \end{bmatrix}$ for $i = 1, 2, \cdots, L$

to solve $\det(G_i - \lambda I) = 0$ to obtain the eigenvalue pairs. If $\mathrm{mod}(n, 2)$ is not 0, then $z_{[\frac{n+1}{2}],[\frac{n+1}{2}]}$ is the last eigenvalue.

§4. Convergence of QIF Algorithm

In this section, we always assume that QIF factorization of a given matrix can be done successfully.

4.1. Definition and preliminaries.

Definition 1. *A matrix is a W-matrix or Z-matrix if it has the form as in (2.2) respectively.*

Lemma 1. *The product and inverse of W-matrix and Z-matrix is still a W-matrix or Z-matrix.*

Lemma 2. *The QIF factorization of matrix A is unique if A is nonsingular.*

The proof of these lemmas is straightforward and hence is omitted here.

4.2. Convergenve theorem and its proof.

Theorem 1. *Let A be a nonsingular matrix of order n. If $A_1 = A$ and $\{A_k\}(k = 1, 2, \cdots)$ is the matrix sequence defined by the QIF algorithm in §3, then A_k is isospectral with A.*

Proof. On the basis of our algorithm, it follows that

$$A_k = W_k Z_k$$

and

$$A_{k+1} = Z_k W_k = W_x^{-1} A_k W_k. \tag{4.1}$$

By induction, the proof is completed.

Theorem 2. *Let A be a nonsingular matrix of order n. If its eigenvalues $\lambda_1, \lambda_2, \cdots, \lambda_n$ satisfy*

$$|\lambda_1| > |\lambda_2| > \cdots > |\lambda_n| > 0 \tag{4.2}$$

and the corresponding right-hand eigenvectors are

$$x_1, x_2, \cdots, x_n.$$

then, with the assumption that $x = [x_1, x_2, \cdots, x_n]$ has QIF factorization

$$X = W_x Z_x, \tag{4.3}$$

$\{A_k\}$ converges to a Z-matrix while $k \to \infty$.

Proof. From (4.1), it follows that

$$A_k = W_{k-1}^{-1} A_{k-1} W_{k-1} = W_{k-1}^{-1} W_{k-2}^{-1} A_{k-2} W_{k-2} W_{k-1} = \cdots = \tilde{W}_k^{-1} A \tilde{W}_k \tag{4.4}$$

where

$$\tilde{W}_k = W_1 W_2 \cdots W_{k-1}.$$

Denoting $\tilde{Z}_k = Z_{k-1} Z_{k-2} \cdots Z_1$, it is easy to get

$$A^k = \tilde{W}_k \tilde{Z}_k. \tag{4.5}$$

Therefore, $\tilde{W}_k$ is the W-factor of A^k's factorization. Using (4.3) and $A = X \wedge X^{-1}$ with $\Lambda = \mathrm{diag}(\lambda_1, \lambda_2, \cdots \lambda_n)$, we obtain

$$A = W_x Z_x \Lambda Z_x^{-1} W_x^{-1}. \tag{4.6}$$

In order to make the subsequence argument more transparent we shall assume for simplicity that $n = 3$ hereafter. Thus

$$X = \begin{bmatrix} x_{1,1} & x_{1,2} & x_{1,3} \\ x_{2,1} & x_{2,2} & x_{2,3} \\ x_{3,1} & x_{3,2} & x_{3,3} \end{bmatrix}, \quad X^{-1} = Y = \begin{bmatrix} y_{1,1} & y_{1,2} & y_{1,3} \\ y_{2,1} & y_{2,2} & y_{2,3} \\ y_{3,1} & y_{2,2} & y_{3,3} \end{bmatrix}.$$

By the algorithm in §2, we obtain

$$W_x = \begin{bmatrix} 1 & 0 & 0 \\ \xi_1 & 1 & \xi_2 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.7}$$

where W_x is a W-factor of X (see (4.3)) and

$$\xi_1 = \frac{\begin{vmatrix} x_{2,1} & x_{2,3} \\ x_{3,1} & x_{3,3} \end{vmatrix}}{\begin{vmatrix} x_{1,1} & x_{1,3} \\ x_{3,1} & x_{3,3} \end{vmatrix}}, \quad \xi_2 = \frac{\begin{vmatrix} x_{1,1} & x_{1,3} \\ x_{2,1} & x_{2,3} \end{vmatrix}}{\begin{vmatrix} x_{1,1} & x_{1,3} \\ x_{3,1} & x_{3,3} \end{vmatrix}} \tag{4.8}$$

We also have

$$A^k = X \begin{pmatrix} \lambda_1^k & & 0 \\ & \lambda_2^k & \\ 0 & & \lambda_3^k \end{pmatrix} Y = (f_{ij}) \tag{4.9}$$

where

$$f_{i,j} = \sum_{l=1}^{3} \lambda_l^k x_{i,l} y_{l,j}, \qquad i, j = 1, 2, 3. \tag{4.10}$$

After some algebraic operations , we have the expression of factor $\tilde{W}_k$ in (4.5) as

$$\tilde{W}_k = \begin{bmatrix} 1 & 0 & 0 \\ \eta_1(k) & 1 & \eta_2(k) \\ 0 & 0 & 1 \end{bmatrix} \tag{4.11}$$

where

$$\eta_1(k) = \frac{\begin{vmatrix} f_{2,1} & f_{2,3} \\ f_{3,1} & f_{3,3} \end{vmatrix}}{\begin{vmatrix} f_{1,1} & f_{1,3} \\ f_{3,1} & f_{3,3} \end{vmatrix}}, \quad \eta_2(k) = \frac{\begin{vmatrix} f_{1,1} & f_{1,3} \\ f_{2,1} & f_{2,3} \end{vmatrix}}{\begin{vmatrix} f_{1,1} & f_{1,3} \\ f_{3,1} & f_{3,3} \end{vmatrix}} . \tag{4.12}$$

Substituting (4.6) into (4.4), yields

$$A_k = (\tilde{W}_k^{-1} W_x)(Z_x \lambda Z_x^{-1})(W^{-1}\tilde{W}_k) . \tag{4.13}$$

Using (4.11) and (4.7), we obtain

$$W^{-1}\tilde{W}_k = \begin{bmatrix} 1 & 0 & 0 \\ \eta_1 - \xi_1 & 1 & \eta_2 - \xi_2 \\ 0 & 0 & 1 \end{bmatrix} . \tag{4.14}$$

To obtain the expected conclusion, what we should do is only to prove

$$\lim_{k \to \infty} (\eta_i(k) - \xi_i) = 0, \qquad i = 1, 2. \tag{4.15}$$

Due to $\left|\frac{\lambda_2}{\lambda_1}\right| < 1$ and $\left|\frac{\lambda_3}{\lambda_1}\right| < 1$, $\eta_i(k)(i = 1, 2)$ perform like 0/0 while k goes to infinity. Considering k as a continuous variable and using L'Hospital rule successively, we can prove (4.15). This completes the proof.

§5. Numerical Examples

Example 1.

$$A=\begin{bmatrix}1&1&0&0&0\\1&2&1&0&0\\0&1&3&1&0\\0&0&1&4&1\\0&0&0&1&5\end{bmatrix},\qquad B=\begin{bmatrix}10&1&2&3&4\\1&9&-1&2&-3\\2&-1&7&3&-5\\3&2&3&12&-1\\4&-3&-5&-1&15\end{bmatrix}.$$

The two matrices given in this example are symmetric with well-conditioned eigenvalues. After 16 iterations, the resulting matrices are

$$A_{16}=\begin{bmatrix}4.066329&1.000000&.000000&.000000&.028869\\.000000&1.396981&1.000000&.253980&-.000001\\.000000&.000000&.253843&.000000&.000000\\.000001&-2.925703&1.000000&3.644405&-.000002\\5.389012&.000000&.000000&1.000000&5.638444\end{bmatrix},$$

$$B_{16}=\begin{bmatrix}15.526360&-.039843&2.000000&3.337678&.756950\\.000002&9.385128&-3.670916&-.182145&.000000\\.000000&.000000&1.655266&.000000&.000000\\.000000&.256873&-2.614020&6.975264&.000000\\-1.362161&-.400391&-5.000000&-1.844194&19.457980\end{bmatrix}$$

respectively. It is very easy to obtain their eigenvalues.

Example 2. In this example

$$A=\begin{bmatrix}5&1&-2&0&-2&5\\1&6&-3&2&0&6\\-2&-3&8&-5&-6&0\\0&2&-5&5&1&-2\\-2&0&-6&1&6&-3\\5&6&0&-2&-3&8\end{bmatrix},$$

A is still a symmetric matrix, of which the eigenvalues are

$$\lambda_{1,2}=-1.598735,\quad \lambda_{3,4}=4.55990,\quad \lambda_{5,6}=16.142745.$$

The iterative result is

$$A_{18}=\begin{bmatrix}16.142750&3.182151&-2.000000&.000000&-.661227&.000000\\.000000&4.455986&5.050735&-1.417750&.000001&.000000\\.000000&.000000&-1.598734&.000001&.000000&.000000\\.000000&.000000&-.000001&-1.598736&.000000&.000000\\.000000&-.000003&6.025439&-6.822922&4.455992&.000000\\.000000&2.810211&.000000&-2.000000&-2.355618&16.142740\end{bmatrix}.$$

Example 3.

$$A = \begin{bmatrix} 0 & 4 & -1 & 3 & 2 & 1 \\ -4 & 0 & 7 & 0 & 1 & 3 \\ 1 & -7 & 0 & 2 & 9 & 1 \\ -3 & 0 & -2 & 0 & -4 & 5 \\ -2 & -1 & -9 & 4 & 0 & 1 \\ -1 & -3 & -1 & -5 & -1 & 0 \end{bmatrix}.$$

It is a skew-symmetric matrix. After 50 iterations we have

$$A_{50} = \begin{bmatrix} 24.013300 & 3.450934 & -1.000000 & 3.000000 & -5.291406 & 78.573320 \\ .000000 & 24.573290 & 3.633369 & 13.171040 & 12.398500 & -.000001 \\ .000000 & .000000 & -18.936540 & -46.808870 & .000000 & .000000 \\ .000000 & .000000 & 7.762646 & 18.936590 & .000000 & .000000 \\ -.000001 & -53.913680 & -9.604937 & -32.648360 & -24.573280 & -0.000001 \\ -9.230482 & -5.684251 & -1.000000 & -5.000000 & -.654436 & -24.013350 \end{bmatrix}.$$

From A_{50}, the eigenvalue computation is simple.

Example 4.

$$A = \begin{bmatrix} -1.750 & -0.500 & 16.5 & -4.50 \\ -2.000 & -3.000 & 46.0 & -13.00 \\ -6.375 & 3.750 & -6.75 & 2.75 \\ -21.25 & 14.500 & -44.5 & 15.50 \end{bmatrix}.$$

This matrix has the eigenvalues $\lambda_{1,2,3,4} = 1$ and a nonlinear elementary factor $(\lambda - 1)^4$. After 120 iterations, it follows that

$$A_{120} = \begin{bmatrix} 1.32398200 & -5.00000000 & 16.50000000 & -.33515920 \\ .00000034 & 2.01772100 & -3.30319200 & -.00000020 \\ .00000010 & .33519230 & -.08556835 & -.00000006 \\ .25398720 & 14.50000000 & -44.50000000 & .74386950 \end{bmatrix}.$$

The convergence is acceptable. It is pretty slow.

Example 5.

$$A = \begin{bmatrix} -41.250 & 21.500 & -27.500 & 11.000 \\ -72.500 & 31.000 & 12.000 & 1.500 \\ -55.125 & 39.000 & -135.750 & 44.000 \\ -159.75 & 124.500 & -476.500 & 152.000 \end{bmatrix}.$$

It has eigenvalues $\lambda_{1,2} = 1 \pm i, \quad \lambda_{3,4} = 2 \pm 3i$.

$$A_{16} = \begin{bmatrix} -8.124342 & 21.500000 & -27.500000 & .541634 \\ 0.000000 & -1.999967 & 1.333379 & .000000 \\ -.000000 & -7.499605 & 3.999948 & .000000 \\ -205.862800 & 124.500000 & -476.500000 & 12.124370 \end{bmatrix}.$$

So we could compute its eigenvalues easily.

Up to 30 numerical examples have been computed. All, except two, are convergent. The failure cases are

Case 1.

$$A=\begin{bmatrix} -34.750 & 17.500 & -17.500 & 7.500 \\ -75.500 & 33.000 & 6.000 & 3.500 \\ -21.375 & 18.750 & -81.750 & 25.250 \\ -44.250 & 52.500 & -290.500 & 87.500 \end{bmatrix}.$$

Its eigenvalues are $\lambda_{1,2}=1\pm i,\quad \lambda_{3,4}=1\pm i$.

Case 2.

$$A=\begin{bmatrix} 3 & 2 & 1 & 2 & 1 & 4 & 1 & 2 \\ 2 & 1 & 3 & 1 & 2 & 2 & 1 & 4 \\ 0 & 3 & 1 & 2 & 1 & 2 & 1 & 3 \\ 0 & 0 & 1 & 1 & 2 & 1 & 3 & 1 \\ 0 & 0 & 0 & 10^{-1} & 3 & 1 & 4 & 1 \\ 0 & 0 & 0 & 0 & 10^{-6} & 2 & 1 & 4 \\ 0 & 0 & 0 & 0 & 0 & 1 & 2 & 3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 3 & 2 \end{bmatrix}.$$

Here, the QIF factorization without permutations breaks down.

§6. Conclusion and Open Problems

As shown in §5, this algorithm could be a remarkable approach for matrix eigenvalue problems. After one (or two) eigenvalue(s) is obtained, the matrix can be reduced into a matrix with smaller size. In this way, the algorithm would be more economical.

Some important problems are still unknown:

1. Necessary and sufficient condition of the convergence of the algorithm.
2. Accelerating technique for this algorithm.
3. Relationship between it and other algorithms.
4. In [3], the relationship between TODA flow and the QR method is deduced for some cases. We can ask whether there is some special flow which is a generalized form of this algorithm.

References

[1] D. J. Evans, A. Hadjidimos, D. Noutsos, The parallel solution of bended linear equations by the new quadrant interlocking factorization (Q.I.F) method, *Int Jour. Comp. Math.*, 9 (1981), 151–62.

[2] Xie Song-mao, WZ-factorization and its error analysis, submitted to *J.C.M.* (In Chinese).

[3] Moody. T. Chu, Asymptotic analysis of TODA lattice on diagonalizable matrices, *Nonlinear Analysis, Theory, Methods and Applications*, 9 (1985), 193–201.