

A Comparison of Methods for Computing the Eigenvalues and Eigenvectors of a Real Symmetric Matrix

By Paul A. White and Robert R. Brown

PART I. THE EIGENVALUES

I. Purpose. To compare those methods for computing the eigenvalues of a real symmetric matrix for which programs are readily available. The comparison was made on the basis of computing time and accuracy.

II. Methods Tested. Three codes were selected from SHARE which represented three different methods.

A. *Jacobi*. SHARE distribution 705 by MIT computing lab (FORTRAN). This is the original version of the Jacobi method [1] in which plane rotations are used to produce zeros in all off-diagonal positions using the maximum off-diagonal element as a pivot at each step. This is probably slower but more accurate than the "serial" version which pivots on the off-diagonal elements in sequence whether or not they are large.

B. *Givens*. SHARE distribution 664 (AN F202) by the AEC Computing and Applied Mathematics Center, NYU (FORTRAN). This is the method devised by W. Givens in 1954 at the Oak Ridge National Laboratory [2] in which plane rotations are used to produce a tri-diagonal matrix with the same eigenvalues as the original matrix. The roots of the matrix are computed by the use of Sturm's sequence derived from the tri-diagonal matrix.

C. *Householder*. SHARE distribution #1385 (AN 202) by the AEC Computing and Applied Mathematics Center, NYU (FORTRAN). This is a variation of the Givens method in which the tri-diagonal matrix is produced by an orthogonal transformation that does not depend on plane rotations. It is described in detail in [3] by J. H. Wilkinson. The Sturm sequences are used as before to obtain the roots from the tri-diagonal matrix.

Series overflow and/or underflow problems occurred in the earlier experiments because a scaling device was not incorporated into the codes of methods B and C. Both methods were modified to incorporate this feature in the decks used in the experiments described below. Modified listings are available from the authors.

III. Description of the Tests. Nine test matrices were used, eight 32×32 and one 8×8 . The eigenvalues of each were computed by all three methods and the accuracy and time of computation were compared. The eight 32×32 matrices were constructed by using a tensor product of five 2×2 matrices. This had the advantage of being able to generate the test matrices in the machine by inserting only five data cards. It was also possible to calculate in a very easy way the correct eigenvalues of these matrices. The 8×8 matrix was the well-known Rosser matrix [4].

Received July 23, 1963. Revised November 4, 1963.

For the eight 32×32 tensor product matrices, the eigenvalues were also computed by Jacobi procedures using double precision arithmetic applied to the single precision matrices obtained from the tensor product calculations.

These differed from the exact eigenvalue by less than one in the eighth place.

This would indicate that no significant rounding takes place in the tensor product or normalizing calculations.

IV. Summary of Results. The table below gives the following data for each of the eight tensor product matrices: the Euclidean norm, $\sqrt{\sum a_{ij}^2}$; the true maximum and minimum eigenvalues; the maximum and average error in the computed eigenvalues for each method; the computing time in minutes for each method. In the case of the Rosser matrix, all of the true eigenvalues and computed errors in eigenvalues are given together with the computation times.

The data indicate that the Householder method is consistently faster and more accurate than the Givens method. This is the same conclusion reached by Ortega [5] based on more extensive tests. In the Jacobi method the times and accuracies are definitely correlated as is to be expected, i.e., the faster the computation, the more accurate the result since the fewer number of arithmetic operations produces less round off errors. Whenever the computation time for the Jacobi method is less than that of the Householder method the results are more accurate.

The Jacobi times and accuracies vary considerably from one matrix to another while the results for the Householder method remain rather uniform. On the average the Jacobi times are a little longer and the accuracies a little less than those of the Householder method. The results for the Rosser matrix did not differ appreciably from one method to another.

The Householder method is definitely better than the Givens and being somewhat faster and more accurate on the average as well as being more consistent than the Jacobi method, it is to be preferred.

V. Documentation. The following data, codes, etc. were used in the development of the results in these experiments.

A. *The Eight Tensor Product Matrices.* For each of the eight test matrices, each of the five 2×2 matrices used to form the 32×32 tensor product is given together with its eigenvalues.

B. *The Correct Eigenvalues of the Eight Tensor Product Matrices.* In each of the eight cases, the correct eigenvalues were computed by forming the tensor product of the eigenvalues of the five 2×2 matrices.

A, B, C, D, and E designate the five pairs of eigenvalues for the five 2×2 matrices in each case.

C. *Computation of the Eigenvalues of the Eight Tensor Product Matrices by Givens' and Jacobi's Methods.* The eigenvalues of the eight tensor product matrices are computed by Givens' and Jacobi's methods. The input consists of the five 2×2 matrices in each case; each 2×2 matrix is on one card consisting of two vectors, one for each row.

D. *Computation of the Eigenvalues of the Eight Tensor Product Matrices by Householder's Method.* This is the same as C, with Householder's method instead of Givens' and Jacobi's.

SUMMARY

No.	MATRIX			GIVENS			JACOBI			HOUSEHOLDER		
	Eucl. Norm.	True Max. Eigen.	True Min. Eigen.	Max. Err.	Av. Err.	Time Min.	Max. Err.	Av. Err.	Time Min.	Max. Err.	Av. Err.	Time Min.
1	7484990	3256673.6	299321.67	9.0	4.68	.19	15.0	5.13	.44	6.0	3.40	.16
2	5632541	3139134.7	315537.41	9.0	4.07	.20	14.0	3.53	.34	6.0	2.40	.16
3	6439353	2872126.2	365685.91	7.0	3.94	.20	11.0	3.44	.33	5.0	2.24	.15
4	7757154	3114796.4	331550.41	4.0	1.68	.15	2.0	.69	.02	3.0	1.26	.15
5	5494709	2912495.3	365327.66	9.0	4.24	.19	5.0	1.88	.14	7.0	3.04	.16
6	7111379	2754619.0	394587.33	7.0	4.18	.19	3.0	.80	.03	5.0	2.42	.15
7	8505404	4682912.5	157228.33	9.0	3.50	.17	32.0	4.86	.44	9.0	2.72	.14
8	8516891	3238700.1	319422.37	10.0	4.32	.17	3.0	.84	.06	8.0	2.77	.13
av. of above 8 Rosser					3.83	.18		2.65	.22		2.53	.15
		1020.04901843			-.0001			-.0001			-.0001	
		1020			-.0002			-.0002			-.0002	
		1019.90195136			-.0001			-.0002			-.0001	
		1000			-.00009	.01		-.00007			-.00009	.01
		1000			-.00013			-.00012			-.00017	
		.098048641			-.0000033			+.0000029			+.000041	
		0.0			+.0000073			-.0000031			+.0000078	
		1020.04901843			+.0001			+.0001			+.0001	

E. *Computation of the Eigenvalues of the Rosser Matrix by All Three Methods.* The eigenvalues of the Rosser matrix are computed by the three methods. The matrix as given on the input cards has eight row vectors.

F. *Program: 32 Eigenvalues from Five 2×2 Matrices.* This is the listing for the program that produced the data in B.

G. *Multimethod Eigenvalue Program.* This is a listing for the program that produced the data in C. The five 2×2 matrices for each case are the data. The program computes in a subroutine the 32×32 tensor product matrix which is used as data for each of the eigenvalue computations by Givens' and Jacobi's methods which are subroutines in the main program.

H. *Tensor Product.* This routine forms the tensor product of $N, 2 \times 2$ matrices and is a subroutine in the main program *G*.

I. *Givens Method.* This is the Givens method for computing the eigenvalues of a real symmetric matrix. It is written as a subroutine of the main program *G*.

J. *Jacobi Method.* This is the Jacobi method written as a subroutine of the main program *G*.

K. *Householder Method.* This is the Householder method written as a subroutine for the main program *G*. The listing of *G* as given here does not include this method. A revised version was used in which it was added as "method 3." It has also been modified from the SHARE version to prevent over- and underflow.

PART II. THE EIGENVECTORS

I. Purpose. The purpose was the same as in Part I, except this time both eigenvalues and eigenvectors were computed and times and accuracies compared.

II. Methods Tested. The same three programs were used as before. Each program provides a method for the computation of the eigenvectors. The procedure in the Jacobi program is the obvious one of computing the product of the rotations.

In the programs that use the Givens and Householder procedures to obtain the tri-diagonal matrix, the same method is used. Wilkinson's method [6] is applied to find the eigenvectors of the tri-diagonal matrix, and then the reflections are applied in reverse order to obtain the original eigenvectors.

We shall call the Givens procedure followed by the above Method A and Householder's procedure followed by the above Method B.

III. Description of Tests. The same eight tensor product matrices were used together with a 21×21 matrix which was used by Wilkinson [6] to illustrate how catastrophic errors may arise in computing the vectors of a tri-diagonal matrix.

IV. Summary of Results. Each matrix has 32 vectors corresponding to the 32 eigenvalues and each vector 32 components for a total of 1024 components per matrix. However, the vectors of a tensor product matrix are simply tensor products of the vectors of its five 2×2 matrices; hence a relatively smaller number of different magnitudes is really involved. In fact each of the 32 vectors uses exactly the same set of numbers for its components and this set varies from 12 to 18 different numbers. Thus there are at most 18 different computed numbers involved in the 1024 components of any matrix. The next table shows the number of places of

NUMBER OF ACCURATE DIGITS IN COMPONENTS OF EACH OF 32 EIGENVECTORS

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	Total Compu- tation Time (min.)
Matrix #1	2	2	3	2	2	2	2	3	3	3	3	3	3	3	4	5	5	5	5	5	5	5	5	5	5	5	5	4	3	2	2	.29	
Method A	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	.57	
Jacobi	2	2	3	2	2	2	2	3	3	3	3	3	3	3	4	5	5	5	5	5	5	5	5	5	5	5	5	4	2	2	.27		
Method B	2	3	2	3	2	2	2	2	3	4	4	4	5	5	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	.28		
Matrix #2	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	.46		
Method A	2	4	2	4	2	2	2	2	2	4	3	4	5	5	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	.23		
Jacobi	3	2	3	4	3	4	4	4	3	3	3	4	4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	4	3	2	.28		
Method B	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	.42		
Matrix #3	3	2	3	2	3	4	4	4	2	3	3	4	4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	4	4	4	.23		
Method A	2	2	2	2	2	2	2	4	5	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	.21		
Jacobi	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	.03		
Method B	2	2	2	2	3	2	4	5	4	5	6	5	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	.21		
Matrix #4	2	3	4	3	2	4	3	2	3	2	3	5	5	5	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	.28		
Method A	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	.18		
Jacobi	2	3	4	2	2	3	2	3	3	3	3	5	5	5	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	.24		
Method B	2	3	3	3	3	4	2	2	4	4	4	4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	4	2	2	.27		
Matrix #5	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	.03		
Method A	2	3	4	3	2	4	3	2	3	2	3	5	5	5	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	.23		
Jacobi	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	.27		
Method B	2	2	2	2	3	2	4	5	4	5	6	5	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	.23		
Matrix #6	2	3	4	3	2	4	3	2	3	2	3	5	5	5	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	.27		
Method A	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	.03		
Jacobi	2	2	2	2	3	2	4	5	4	5	6	5	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	.23		
Method B	2	2	2	2	3	2	4	5	4	5	6	5	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	.27		
Matrix #7	2	2	2	0	3	3	0	0	2	3	0	0	5	0	5	0	6	0	5	0	0	0	5	5	5	5	5	5	0	3	3	.27	
Method A	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	.58		
Jacobi	2	4	0	0	2	0	0	4	5	0	0	5	0	0	5	4	4	4	0	0	3	0	0	0	2	2	0	0	0	0	.21		
Method B	2	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	.25		
Matrix #8	2	4	0	0	2	0	0	4	5	0	0	5	0	0	5	4	4	4	0	0	3	0	0	0	2	2	0	0	0	0	.07		
Method A	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	.20		
Jacobi																																	
Method B																																	

Approximately same as Method A

Approximately same as Method A

Approximately same as Method A

WILKINSON MATRIX
VECTORS CORRESPONDING TO (\pm) MAX EIGENVALUE

Component. . .	1	2	3	4	5	6	7
True	1.00000000	.74619419	.30299994	.08590250	.01880748	.00336146	.00050815
Method A & Method B	+1	.74619445	.30300065	.08590426	.01881339	.00338739	0
	-1	.74619422	.30299995	.08590250	.01880748	.00336147	.00050815
Jacobi	+1	.74619402	.30299984	.08590244	.01880746	.00336146	.00050815
	-1	.74619411	.30299992	.08590248	.01880748	.00336146	.00050815
Component. . .	8	9	10	11	12	13-21	
True	.00006659	.00000771	.00000080	.00000007	.00000001	0	
Method A & Method B	0	0	0	0	0	0	
	.00006659	.00000771	.00000080	.00000007	.00000001	0	
Jacobi	.00006659	.00000771	.00000080	.00000007	.00000001	0	
	same	same	same	same	same	0	

V. **Documentation.** All of the programs are the same as before in which indicators are set to produce vectors as well as values.

L. Eigenvectors of all different 2×2 matrices involved in any of the eight tensor product matrices were computed here. Actually they were redone by hand to obtain eight correct digits.

M. The eigenvectors of the eight tensor product matrices were computed by using H. The input consists of five 2×2 matrices for each case in which each matrix consists of two eigenvectors of the corresponding 2×2 matrix for its rows. The rows of the tensor product of five such matrices are the vectors of the corresponding tensor product matrix.

N. The eigenvalues and vectors are computed by the three methods using C set to compute vectors as well as values with the additional subroutine D included.

O. The eigenvalues and vectors of the Wilkinson matrix given as 21 row vectors are computed using the same routine as that used in N.

Aerospace Corporation
University of Southern California
Los Angeles, California

1. C. G. J. JACOBI, "Ein leichtes Verfahren, die in der Theorie der Säkularstörungen vorkommenden Gleichungen numerisch aufzulösen," *J. Reine Angew. Math.*, v. 30, 1846, p. 51-95.
2. WALLACE GIVENS, *Numerical Computation of the Characteristic Values of a Real Symmetric Matrix*, Oak Ridge National Laboratory Report ORNL-1574.
3. J. H. WILKINSON, "Householder's method for the solutions of the algebraic eigen problem," *The Computer Journal*, v. 3, p. 23-27.
4. J. B. ROSSER, C. LANCZOS, M. R. HESTENES & W. KARUSH, "Separation of close eigenvalues of a real symmetric matrix," *J. Res. Nat. Bur. Standards*, v. 47, 1951, p. 291-297.
5. JAMES M. ORTEGA, *An Error Analysis of Householder's Method for the Symmetric Eigenvalue Problem*, Stanford Univ. Appl. Math. and Statistics Lab., Tech. Report No. 18.
6. J. H. WILKINSON, "The calculation of the eigenvectors of codiagonal matrices," *The Computer Journal*, 1958, p. 90-96.