NASA Technical Memorandum 89082

A PARALLEL SOLUTION FOR THE SYMMETRIC EIGENPROBLEM

GAYLEN A. THURSTON

(NASA-TM-89082) A PARALLEL SOLUTION FOR THE SYMMETRIC EIGENPROBLEM (NASA) 14 p CSCL 20K N87-17086

Unclas
G3/39 43746

JANUARY 1987

National Aeronautics and Space Administration

Langley Research Center
Hampton, Virginia 23665

INTRODUCTION

This memorandum outlines a completely parallel algorithm for the symmetric eigenproblem $AX = \lambda BX$. The algorithm is parallel in the sense that the numerical operations do not occur in a fixed sequence. Therefore, a large number of operations can be programmed to be performed concurrently on a computer with multiple central processing units.

The standard symmetric eigenvalue problem $AX = \lambda X$ has the property that the n eigenvalues of the principal submatrix of A of order n are separated by the $(n - 1)$ eigenvalues of the principal submatrix of order $(n - 1)$. The separation property delineates n intervals containing one eigenvalue. Each eigenvalue and corresponding eigenvector can be computed independently. The n eigenproblem calculations can be divided among multiple processing units.

A sufficient condition for the separation property to apply to the general symmetric problem $AX = \lambda BX$ is presented in this memorandum. The sufficient condition is slightly less restrictive than a condition which ensures that all the eigenvalues are real, namely, that B be positive definite.

A parallel algorithm is readily derived for problems that obey the separation property. Once the eigenproblem of order n is solved the algorithm solves the eigenproblem of order $(n + 1)$ and continues until $n = N$, the order of the square matrices A and B. Only one pass through the principal submatrices is required to solve for all the eigenvalues and eigenvectors of $AX = \lambda BX$.

Intermediate numerical results for eigenvectors must be stored. These results can be shared by the central processing units, or the results can be partitioned into separate blocks of data accessed by a single processor.

The algorithm is numerically stable, avoids factoring matrix B, and automatically arranges the eigenvalues in nondecreasing order. Multiple eigenvalues and closely spaced eigenvalues do not present any numerical difficulties.

The separation property is regarded in the literature as a theoretical property rather than the basis for a practical algorithm. The reason is the large number of matrix-vector multiplications required by the algorithm. The total mumber of scalar multiplications is on the order of $N^4/4$ compared to the comparable number of $5N^3/3$ for the Householder method for the standard problem. However, the input-output operations are straightforward for the parallel method so that the total computer cost is not necessarily as prohibitive as indicated by the count of scalar multiplications.

The body of the memorandum outlines the derivation of the separation property for the general symmetric problem $AX = \lambda BX$. The steps in the algorithm are next written explicitly. Finally, some of the numerical analysis for avoiding round-off errors in enforcing the separation property in a code for parallel processing is discussed. Special cases of uncoupled modes and multiple eigenvalues are included in the discussion.

SEPARATION PROPERTY

The separation property follows from examining the characteristic equation for the bordered eigenvalue problem of order n in the form

$$\begin{bmatrix} (\mu_1-\lambda) & 0. & \cdots & 0. & (c_1-\lambda d_1) \\ 0. & (\mu_2-\lambda) & \cdots & 0. & (c_2-\lambda d_2) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0. & 0. \;\cdots & (\mu_i-\lambda) \;\cdots & 0. & (c_i-\lambda d_i) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0. & 0. & \cdots & (\mu_{n-1}-\lambda) & (c_{n-1}-\lambda d_{n-1}) \\ (c_1-\lambda d_1) & (c_2-\lambda d_2) & (c_i-\lambda d_i) & (c_{n-1}-\lambda d_{n-1}) & (a_{nn}-\lambda b_{nn}) \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \cdots \\ y_i \\ \cdots \\ y_{n-1} \\ x_n \end{bmatrix} = 0 \qquad (1)$$

The μ_i are eigenvalues of the problem $A^{(n-1)} X = \lambda B^{(n-1)} X$ where the superscripts in parentheses denote the order of a principal submatrix. The bordered form in equation (1) follows from a similarity transformation using $U^{(n-1)}$, the matrix of orthonormal eigenvectors corresponding to the eigenvalues μ_i.

A complete proof of the separation property for the standard eigenvalue problem is derived in ref. (1). For the standard problem, the coefficients d_i vanish and b_{nn} is unity. The effect of these terms on the proof of the separation property is shown here.

The characteristic equation for the matrix of coefficients in equation (1) can be written as

$$f(\lambda) = (a_{nn} - \lambda b_{nn}) - \sum_{i=1}^{n-1} (c_i - \lambda d_i)^2/(\mu_i - \lambda) \qquad (2)$$

The separation property is that the roots of equation (2) satisfy the conditions

$$\lambda_1 \le \mu_1 \le \lambda_2 \quad \cdots \quad \mu_{i-1} \le \lambda_i \le \mu_i \quad \cdots \quad \le \mu_{n-1} \le \lambda_n \qquad (3)$$

where the μ_i are arranged in nondecreasing order.

A sufficient condition for the separation property to hold is derived by showing that $f(\lambda)$ has only one real root between consecutive values of the μ_i as indicated schematically in Fig. 1. Since $f(\lambda)$ goes from positive infinity to negative infinity, $f(\lambda)$ has only one root in each interval if the slope of $f(\lambda)$ never changes sign and is always negative.

$$\frac{df}{d\lambda} = -b_{nn} - \sum_{i=1}^{n-1} \left[\frac{-2d_i(c_i - \lambda d_i)}{(\mu_i - \lambda)} + \frac{(c_i - \lambda d_i)^2}{(\mu_i - \lambda)^2} \right] \tag{4}$$

or, rearranging terms,

$$\frac{df}{d\lambda} = -\{ (b_{nn} - \sum_{i=1}^{n-1} d_i^2) + [\sum_{i=1}^{n-1} \frac{(c_i - \mu_i d_i)^2}{(\mu_i - \lambda)^2}]\} \tag{5}$$

Therefore, a sufficient condition for the separation property to hold is that the right-hand side of equation (5) is negative. By dropping the positive quadratic terms inside the square brackets, a more restrictive condition is determined that holds independent of the value of λ.

$$\Delta_n = b_{nn} - \sum_{i=1}^{n-1} d_i^2 > 0 \tag{6}$$

However, the term Δ_n is a factor in a product equal to the determinant of matrix $B^{(n)}$ and condition (6) holds when matrix B is positive definite, a sufficient condition for all of the eigenvalues of $AX = \lambda BX$ to be real.

The result in equation (5) is not a complete proof of the separation property, conditions (3). The special cases of multiplicity of equal eigenvalues μ_i and of uncoupled modes where $c_i = d_i = 0$ must be considered, but these cases are included in the discussion in ref. (1) for the standard symmetric eigenvalue problem and are omitted here. The special cases are included in the algorithm presented in the next section.

The significant result for parallel processing is that the separation property allows concurrent computation of the roots of the characteristic equation, equation (2), and of the corresponding eigenvectors from equations (1). The concurrent computation will be examined further after outlining the complete parallel algorithm for the problem $AX = \lambda BX$.

COMPLETE PARALLEL ALGORITHM

The concurrent solution of the characteristic equation $f(\lambda) = 0$, equation (2), is one step in the parallel algorithm. The steps in solving the nth order problem $A^{(n)}X = \lambda B^{(n)}X$ are the following:

1. Given μ_i, the eigenvalues of $A^{(n-1)}X = \lambda B^{(n-1)}X$, and the corresponding matrix of eigenvectors $U^{(n-1)}$, compute the bordered form, equations (1), of order n.

$$c_i = \sum_{k=1}^{n-1} u_{ki}^{(n-1)} a_{kn} \qquad i = 1,2,3,\ldots,(n-1) \tag{7a}$$

$$d_i = \sum_{k=1}^{n-1} u_{ki}^{(n-1)} b_{kn} \qquad i = 1,2,3,\ldots,(n-1) \tag{7b}$$

2. Solve the characteristic equation for the bordered form, equation (1), for the eigenvalues λ_j.

$$f(\lambda_j) = (a_{nn} - \lambda_j\, b_{nn}) - \sum_{i=1}^{n-1} \frac{(c_i - \lambda_j\, d_i)^2}{(\mu_i - \lambda_j)} = 0 \qquad j = 1,2,3,\ldots,n \tag{8}$$

3. Compute the elements of the matrix of orthonormal eigenvectors for the bordered problem, $V^{(n)}$.

$$v_{ij} = s_{ij} v_{nj} \tag{9a}$$

where

$$s_{ij} = - (c_i - \lambda_j\, d_i)/(\mu_i - \lambda_j) \qquad \begin{array}{l} i = 1,2,3,\ldots,n-1 \\ j = 1,2,3,\ldots,n \end{array}$$

The nth element of each eigenvector, v_{nj}, is determined by normalizing the orthogonality relation containing B as a weighting matrix,

$$v_{nj} = [\, b_{nn} + \sum_{i=1}^{n-1} s_{ij}(s_{ij} + 2\, d_i)]^{-1/2} \qquad j = 1,2,3,\ldots,n \tag{9b}$$

4. Compute the matrix of orthonormal eigenvectors, $U^{(n)}$, for the problem $A^{(n)}X = \lambda\, B^{(n)}X$.

$$u_{ij}^{(n)} = \sum_{k=1}^{n-1} u_{ik}^{(n-1)} v_{kj} \qquad \begin{array}{l} i = 1,2,3,\ldots,(n-1) \\ j = 1,2,3,\ldots,n \end{array} \tag{10a}$$

$$u_{nj}^{(n)} = v_{nj} \qquad j = 1,2,3,\ldots,n \tag{10b}$$

The complete algorithm updates the order n to $(n + 1)$, $\mu_j^{(n+1)} = \lambda_j^{(n)}$, and repeats the four steps above until $n = N$, the order of the original problem $AX = \lambda BX$. This completes the algorithm. The problem is solved for all the eigenvalues and corresponding eigenvectors with the eigenvalues automatically arranged in nondecreasing order. The eigenvectors are orthonormal with B as a weighting matrix,

$$(U^{(N)})^T B U^{(N)} = I \tag{11}$$

All the computations in the four steps with subscript j are independent and can be programmed for computation in parallel. The algorithm is almost doubly parallel since the row operations with subscript i are also independent for each value of i. Any parallel processing with i subscripts depends on the data storage of the matrices $U^{(n-1)}$ and $U^{(n)}$ and on their accessability by the different central processing units. The possibility of parallel processing with the row operations on subscript i is merely noted here; the rest of this memorandum examines the parallel operations with the j subscript.

NUMERICAL STABILITY

Separation Property

The algorithm summarized by the four steps in the preceding section is parallel in theory. In practice, the linear independence of the computed intermediate eigenvectors must be preserved without too much round-off error. The linear independence is achieved by enforcing the separation property, computing accurate roots of the characteristic equation, equations (2), and avoiding zero or small divisors in the eigenvector computation, equations (9).

In programming the parallel algorithm, the separation property reduces to programming each central processing unit to solve equation (8) for a distinct λ_j such that

$$\mu_{j-1} \leq \lambda_j \leq \mu_j \tag{12}$$

The characteristic equation, $f(\lambda_j) = 0$, is nonlinear and must be solved iteratively. An iteration sequence that singles out the root that satisfies condition (12) is based on finding roots of another function, $G(\lambda)$,

$$G(\lambda) = (\mu_{j-1} - \lambda)(\mu_j - \lambda) f(\lambda) = 0 \tag{13}$$

The function $G(\lambda)$ can be written as the difference of two functions,

$$G(\lambda) = g_1(\lambda) - g_2(\lambda) \tag{14}$$

The first function is a cubic in λ,

$$g_1(\lambda) = (\mu_{j-1} - \lambda)(\mu_j - \lambda)(a_{nn} - \lambda\, b_{nn}) - (\mu_{j-1} - \lambda)(c_j - \lambda d_j)^2 - (\mu_j - \lambda)(c_{j-1} - \lambda\, d_{j-1})^2 \tag{15}$$

The second function is

$$g_2(\lambda) = (\mu_{j-1} - \lambda)(\mu_j - \lambda)\left[\sum_{i=1}^{j-2} \frac{(c_i - \lambda\, d_i)^2}{(\mu_i - \lambda)} - \sum_{i=j+1}^{n-1} \frac{(c_i - \lambda\, d_i)^2}{(\mu_i - \lambda)} \right] \quad (16)$$

If the inequality (6) is satisfied, the cubic $g_1(\lambda)$ is guaranteed to have three real roots. At least one of the three roots, $\lambda = \bar{\lambda}$, satisfies condition (12), as shown schematically in Fig. 2. The figure also shows the function $g_2(\lambda)$ and λ_j, the exact root of $G(\lambda)$ that satisfies conditions (12). The zeroth approximation $\bar{\lambda}$ for λ_j can be improved by solving the cubic equation in $\Delta\lambda$ generated by replacing λ with

$$\lambda = \bar{\lambda} + \Delta\lambda \quad (17)$$

in $G(\lambda)$ and retaining up through linear terms in $\Delta\lambda$ in the Taylor's series for the terms in brackets in $g_2(\lambda)$. The iteration can be continued by updating $\bar{\lambda}$ with λ from equation (17) until $\Delta\lambda$ approaches zero.

Checking The Separation Property

At some point before the end of step 4, the results from the different central processing units for the eigenvalues λ_j must be checked for special cases. There are two types of special cases.

The first special case is uncoupled modes where

$$c_I = d_I = 0$$

For this case, the complete ordering of the λ_j in conditions (3) must be checked so that $\lambda_I = \mu_I$ or $\lambda_{I+1} = \mu_I$, but not $\lambda_{I+1} = \lambda_I = \mu_I$.

The second special case for the numerical analysis is multiple values of the μ_i. When $\mu_{j-1} = \mu_j = \mu$, then $\lambda_j = \mu$. The iterative solution for $G(\lambda)$ must recognize the correct order of the roots, and also avoid the apparent division by zero in equation (16) when the number of multiple values of μ_i exceeds two. Both special cases must be flagged at some point to prevent division by zero in computing the eigenvectors of the bordered problem, equations (9).

Further Check on the Separation Property

Before examining the eigenvector calculation for the special cases, two overall checks on the solutions of the characteristic equations are listed here. The sum and the product of the roots can be monitored at each value of n. Given the sum and the product of the eigenvalues for the problem of order (n-1),

$$S(n-1) = \sum_{i=1}^{n-1} \mu_i \tag{18a}$$

$$P(n-1) = \prod_{i=1}^{n-1} \mu_i \tag{18b}$$

an independent check is available for the sum and product of the next order

$$S(n) = \sum_{j=1}^{n} \lambda_j \tag{19a}$$

$$P(n) = \prod_{j=1}^{n} \lambda_j \tag{19b}$$

The checks on equations (19) follow from clearing the characteristic equation, $f(\lambda) = 0$, of fractions and writing it as an nth degree polynomial. The independent checks are then

$$S(n) = S(n-1) + [a_{nn} + \sum_{i=1}^{n-1} d_i(\mu_i d_i - 2\ c_i)]/\Delta_n \tag{20a}$$

$$P(n) = -P(n-1)\ [a_{nn} - \sum_{i=1}^{n-1} c_i/\mu_i]/\Delta_n \tag{20b}$$

where Δ_n is defined in equation (6).

Linear Independence of Eigenvectors

Two special cases were identified above that must be flagged to prevent division by zero in computing eigenvectors of the bordered problem. To avoid round-off errors in the eigenvector computation, a second subscript $I(j)$ can be defined that is related to every eigenvalue λ_j by the condition that the absolute value of $(\mu_I - \lambda_j)$ is a minimum for the set of values $(\mu_i - \lambda_j)$. The quantity $(\mu_I - \lambda_j)$ appears in the iterative solution of $G(\lambda_j) = 0$ and can be computed at that time rather than performing the indicated subtraction when λ_j is nearly equal to μ_I. Then, if $(\mu_I - \lambda_j)$ is less than $(c_I - \lambda_j d_I)$ in absolute value, an alternate form of equations (9) is preferable for the eignvector computation,

$$v_{nj} = -v_{Ij}\ (\mu_I - \lambda_j)/(c_I - \lambda_j) = v_{Ij}\ /s_{Ij} \tag{21a}$$

$$v_{ij} = s_{ij}\ v_{Ij}\ /s_{Ij} \qquad i = I\ i = n \tag{21b}$$

$$v_{Ij} = \frac{(c_I - \lambda_j d_I)}{\{b_{nn}(\mu_I - \lambda_j)^2 + \sum_{i=1}^{n-1} [(\mu_I - \lambda_j)^2 s_{ij}^2 + 2 d_i s_{ij}(\mu_I - \lambda_j)]\}^{1/2}} \tag{21c}$$

The special case of an uncoupled mode with $c_I = d_I = 0$ is a limiting case of equations (21) with $V_{Ij} = 1$ and the remaining $v_{ij} = 0$. The second special case of multiple eigenvalues $\mu_{j-1} = \mu_j = \mu_I$ is also a limiting case of equations (21) where equation (21c) takes the form

$$v_{Ij} = \frac{(c_I - \lambda_j d_I)}{[(c_{j-1} - \lambda_j d_{j-1})^2 + (c_j - \lambda_j d_j)^2]^{1/2}} \tag{22}$$

Therefore, if the I(j) subscripts are set properly and the quantities $(\mu_I - \lambda_j)$ are computed precisely, the eigenvectors will be linearly independent. The matrix of eigenvectors $V^{(n)}$ will be orthonormal without the need for checking linear independence. The accurate computation of the quantities $(\mu_I - \lambda_j)$ is discussed further in the appendix.

CONCLUDING REMARKS

A completely parallel algorithm has been presented for the symmetric eigenproblem $AX = \lambda BX$. The algorithm is straightforward and can serve as a benchmark problem for testing the efficiency of concurrent central processing units.

The operation count for the algorithm is relatively high, but the storage reqirements and input-output operations are well-ordered. The total computer cost for the algorithm may prove to be competitive with more conventional algorithms.

This memorandum outlines the basic algorithm for dense matrices. The algorithm can be adapted to banded matrices. It can also be modified for constrained problems where the B matrix is singular. An example of the latter problem arises when Lagrangian multipliers are introduced to constrain vibration modes. The formulation with multipliers retains symmetry, but results in some zero elements on the diagonal of the B matrix. The zero elements act to alter the separation property, but the algorithm remains parallel.

APPENDIX

Computing Small Roots of Cubic Equations

The body of this memorandum outlines the solution of $G(\lambda)$, equation (14), by an iterative sequence of cubic equations. As the iteration converges, one root of the cubic equations approaches zero. To prevent round-off errors affecting the final result of the iteration, the standard algorithm for real roots of cubic equations should be slightly modified. The standard algorithm has a change of variables that can be the source of round-off error.

Setting up the iterative sequence also involves a change of variables indicated by $\lambda = \bar{\lambda} + \Delta\lambda$ in equation (17). For later use in computing eigenvectors, the change in variables should also explicitly include two other expressions

$$(\mu_{j-1} - \lambda) = (\mu_{j-1} - \bar{\lambda}) + \Delta\lambda \tag{A1a}$$

$$(\mu_j - \lambda) = (\mu_j - \lambda) + \Delta\lambda \tag{A1b}$$

As $\bar{\lambda}$ is updated by $\Delta\lambda$ to approach λ_j at each iteration step, the quantities in equations (A1) are updated in a similar fashion.

The final accuracy of the iteration depends on computing small roots of a sequence of cubic equations of the form

$$e'_3 x^3 + e'_2 x^2 + e'_1 x + e_o = 0 \tag{A2}$$

where $x = \Delta\lambda$. At each iteration step, an accurate solution of equation (A2) is required for the smallest root in absolute value or else for the root that gives a current value of $\lambda_j - \bar{\lambda}$ that satisfies conditon (12).

The standard algorithm, ref. 2, divides equation (A2) by the coefficient e'_3 to reduce it to the form

$$x^3 + e_2 x^2 + e_1 x + e_o = 0 \tag{A3}$$

Next, a change of variables is introduced,

$$y = x + (e_2/3) \tag{A4}$$

The change of variables transforms the cubic in x to the canonical form

$$y^3 + 3py + 2q = 0 \tag{A5}$$

where

$$p = -(e_2/3)^2 + (e_1/3)$$

$$q = (e_2/3)^3 - (e_1/2)(e_2/3) + (e_o/2)$$

The standard algorithm for solving equation (A5) computes a discriminant D whose sign determines the number of real roots,

$$D = p^3 + q^2 \tag{A6}$$

However, when e_o is small compared to e_2, less round-off error results from computing D from an equivalent expression

$$D = (e_2/3)^2[e_o(e_2/3) - (e_1^2/12)] + (e_o/2)[(e_o/2) - e_1(e_2/3)] \tag{A7}$$

When the discriminant D is negative, the cubic equation has real roots. The three real roots of the cubic in x are

$$x_1 = -(e_2/3) - 2r \cos \alpha \tag{A8a}$$

$$x_2 = [-(e_2/3) + r \cos \alpha] + (3)^{1/2} r \sin \alpha \tag{A8b}$$

$$x_3 = [-(e_2/3) + r \cos \alpha] - (3)^{1/2} r \sin \alpha \tag{A8c}$$

where

$$r = (-p)^{1/2} \; [q/(q^2)^{1/2}]$$

$$\alpha = \frac{1}{3} \sin [(D/p^3 \;)^{1/2}]^{-1}$$

and the sign of each square root is positive.

When the discriminent D is positive, the cubic equation has two complex roots and one real root,

$$x_1 = -(e_2/3) + U + V \tag{A9a}$$

$$x_2 = -(e_2/3) - (U+V)/2 + i \; (3^{1/2}/2)(U-V) \tag{A9b}$$

$$x_3 = -(e_2/3) - (U+V)/2 - i \; (3^{1/2}/2)(U-V) \tag{A9c}$$

where

$U = [-q + (D)^{1/2}]^{1/3}$, $V = [-q - (D)^{1/2}]^{1/3}$, and the signs on the cube roots are chosen so that $UV = -p$.

Theoretically, the iteration sequence for computing the λ_j looks for real roots, $x = \Delta\lambda$, from the cubic equations. However, round-off error can return a small positive value for D when D is near zero. When D is exactly zero, the cubic has at least two equal roots. The iteration sequence searching for real roots can be continued for random small positive D by simply dropping the small imaginary parts of the complex roots.

The iteration sequence converges when $e_o = 0$ in equation (A3). If the assumption is made that the computation for e_o contains less round-off error than the round-off error introduced by the change in variables in equation (A4), then the smallest root of the cubic in x can also be computed from

$$x_1 = \Delta\lambda = -e_o/(x_2 x_3) \tag{A10}$$

where the x_i from equations (A8) have been reorderd in nondecreasing absolute value.

REFERENCES

1. Wilkinson, J. H.: The Algebraic Eigenvalue Problem. Clarendon Press, (Oxford), 1965.

2. Applicable Mathematics. Rektorys, K., Editor, M.I.T Press (Cambridge, Mass.) 1969.

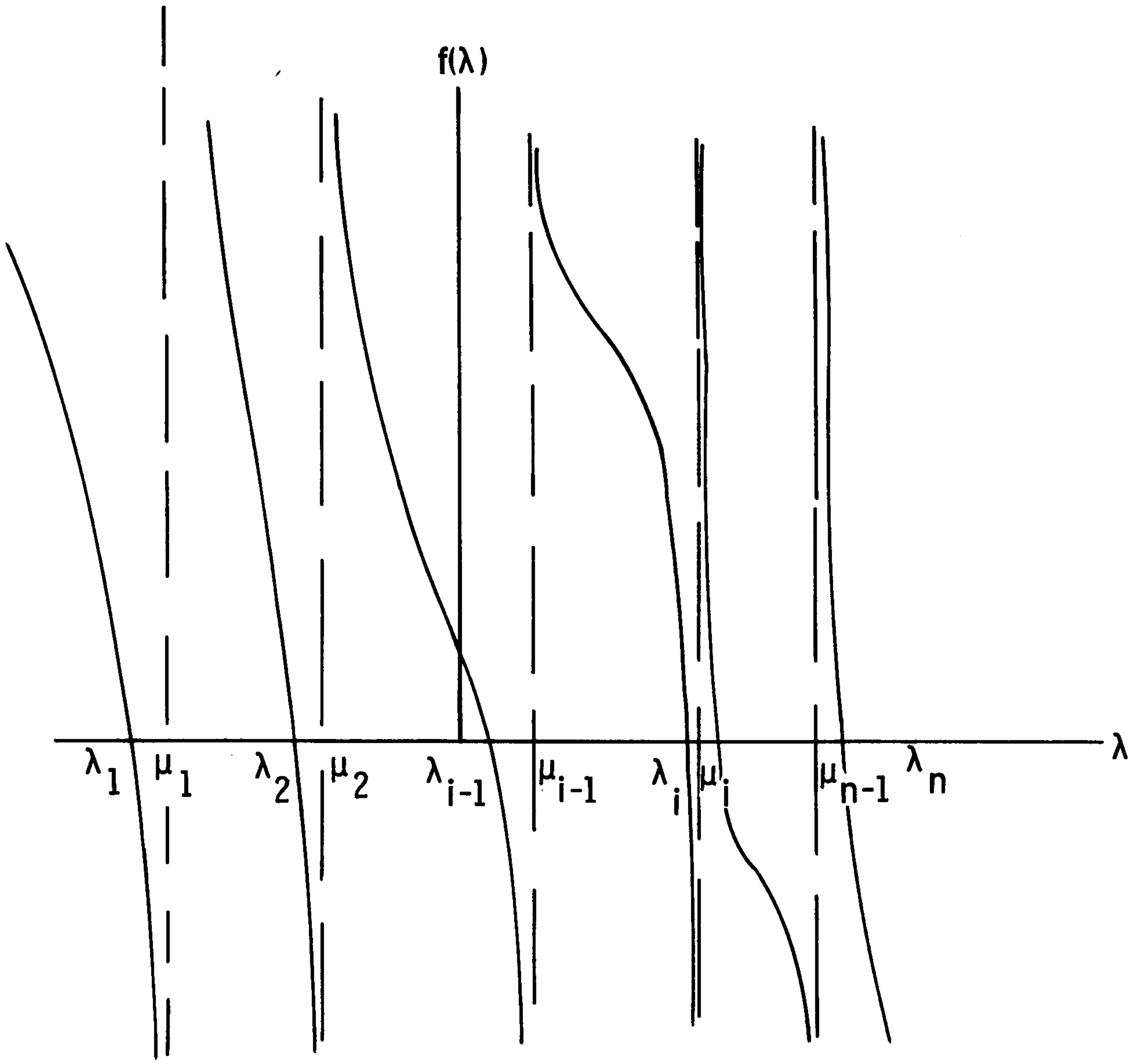

Figure 1. Schematic diagram illustrating separation property for roots of $f(\lambda)$, equation (2).

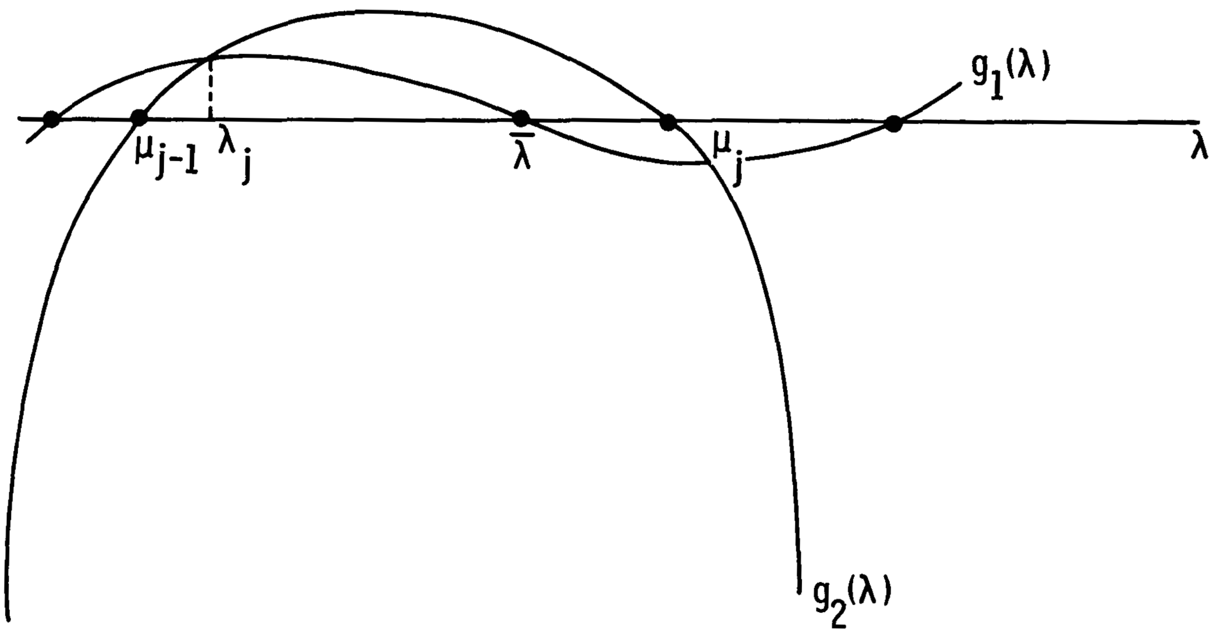

Figure 2. Schematic diagram showing roots of the cubic function of $g_1(\lambda)$, zeros of $g_2(\lambda)$, and λ_j where $g_1(\lambda) = g_2(\lambda)$.

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA TM-89082		

4. Title and Subtitle	5. Report Date
A Parallel Solution for the Symmetric Eigenproblem	January 1987
	6. Performing Organization Code 505-63-11-03
7. Author(s) Gaylen A. Thurston	8. Performing Organization Report No.
	10. Work Unit No.
9. Performing Organization Name and Address NASA Langley Research Center Hampton, VA 23665-5225	11. Contract or Grant No.
	13. Type of Report and Period Covered Technical Memorandum
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, DC 20546-0001	14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

This memorandum outlines a completely parallel algorithm for the symmetric eigenproblem $AX = \lambda BX$. The algorithm is parallel in the sense that the numerical operations do not occur in a fixed sequence. Therefore, a large number of operations can be programmed to be performed concurrently on a computer with multiple central processing units. The standard symmetric eigenvalue problem $AX = \lambda X$ has the property that the n eigenvalues of the principal submatrix of A of order n are separated by the $(n-1)$ eigenvalues of the principal submatrix of order $(n - 1)$. The separation property delineates n intervals containing one eigenvalue. Each eigenvalue and corresponding eigenvector can be computed independently. The n eigenproblem calculations can be divided among multiple processing units.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
parallel solution symmetric eigenproblem	Unclassified - Unlimited Subject Category - 39

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	14	A02