2627

NRL Report 6959

UNCLASSIFIED

A Fortran Computer Program for Calculating the Oblate Spheroidal Radial Functions of the First and Second Kind and Their First Derivatives

A. L. VAN BUREN, R. V. BAIER, AND S. HANISH

Transducer Branch
Acoustics Division

January 20, 1970

NAVAL RESEARCH LABORATORY
Washington, D.C.

CONTENTS

ABSTRACT

The Helmholtz or scalar wave equation $(\nabla^2 + k^2)\,\Psi = 0$ is separable in oblate spheroidal coordinates η, ξ, φ with solutions $\Psi = S\,(ih, \eta)\; R\,(ih, -i\xi)\; \Phi\,(\varphi)$. The subject of this report is a Fortran computer program called OBRAD which numerically evaluates the radial solutions $R\,(ih, -i\xi)$. The printed output from OBRAD consists of radial functions of the first and second kind, $R^{(1),(2)}_{m\ell}$ $(ih, -i\xi)$, their first derivatives $\partial\, R^{(1),(2)}_{m\ell}(ih, -i\xi)/\partial\xi$, the separation constants or eigenvalues $A_{m\ell}\,(ih)$, and an accuracy check. This report first describes the input data cards and the output format. The theory of the oblate spheroidal wave function is then discussed. A description of the principal internal features of OBRAD is then given. Finally a computer listing of OBRAD is attached as an appendix.

PROBLEM STATUS

This is an interim report on a continuing NRL Problem.

AUTHORIZATION

NRL Problem S01-28
Project RR 102-08-41-5225

Manuscript submitted July 9, 1969.

A FORTRAN COMPUTER PROGRAM FOR CALCULATING THE OBLATE SPHEROIDAL RADIAL FUNCTIONS OF THE FIRST AND SECOND KIND AND THEIR FIRST DERIVATIVES

INTRODUCTION

The Helmholtz or scalar wave equation $(\nabla^2 + k^2)\ \Psi = 0$ is separable in oblate spheroidal coordinates, with solution $\Psi = S(\eta)\ R(\xi)\ \Phi(\varphi)$. The subject of this report is a Fortran computer program called OBRAD (OBlate RADial) which evaluates the solutions $R(\xi)$ in the radial spheroidal coordinate ξ. Although other methods of computing the radial functions $R_{ml}(\xi)^{(1),(2)}$ of the first and second kind and their first derivatives are available, it will be the procedure of this report to obtain them from expansions in terms of Bessel and Legendre functions.

Oblate spheroidal wave functions of the radial type constitute an essential element in numerical calculations involving diffraction, radiation, and scattering of acoustic waves, electromagnetic fields of circular disks and apertures, energy levels of certain nuclear models, and the resonant behavior of certain spheroidal cavities. An extensive list of references on calculations and applications of spheroidal wave functions is given in Ref. 1.

The two independent solutions $R_{ml}(\xi)^{(1),(2)}$ of the radial equation are characterized by four parameters: ξ (called X), M, H, and L. M is the integer separation constant relating to the solution for the rotational angle φ. H is equal to $kd/2$, where d is the interfocal distance, and k is the propagation constant or wave vector magnitude $2\pi/\lambda$. For each choice of M, H, and X there will be a set of solutions to the radial equation, each solution characterized by a separation constant or eigenvalue A. It is convenient to order these eigenvalues in an ascending sequence and label them with integers L, beginning with $L = M$ for the smallest eigenvalue, $L = M + 1$ for the next one, etc. This choice is made so that the solutions reduce to that for a spherical coordinate system as H approaches zero. In the spherical case the eigenvalues are simply given by $L\ (L + 1)$. For each choice of M, H, X, and L there will then be two independent radial functions.

Operationally the program OBRAD is divided into several parts. In the first part M and H are set and the eigenvalues are calculated for the desired range of L. In the second part, X is chosen, and the expansion functions, Bessel or Legendre, are obtained. Finally for each choice of L the expansion constants are obtained and combined with the expansion functions to give the radial functions and their first derivatives.

INPUT

The input consists of five data cards:

Data Card 1: Format 8I4. This card contains the integer value *M*1 of the first *M* desired; *IDM*, the increment in *M* used to generate higher values of *M*; *NM*, the number of values of *M* that are desired; *LI*, the initial integer value of *L*; *IDL*, the increment in *L*; *NL*, the number of values of *L* that are desired; *NH* the number of values of *H* that are desired; and *NX*, the number of values of *X* that are desired.

Data Card No. 2: Format D32.25. This card contains *AH*, which is the initial decimal value of *H*.

Data Card No. 3: Format D32.25. This card contains *DH*, which is the increment in *H*.

Data Card No. 4: Format D32.25. This card contains *XI*, which is the initial decimal value of *X*.

Data Card No. 5: Format D32.25. This card contains *DX* which is the increment in *X*.

OUTPUT

The output consists of numerical tables, one page for each set of selected values *H*, *M*, and *X*. Each table gives the radial functions of the first and second kind, *R*1 and *R*2, their first derivatives, *R1D* and *R2D*, and the eigenvalue for all choices of *L* that were requested. Only 18 significant figures are printed in the table, although 26 significant figures are calculated and more than 18 of these may be accurate.

An accuracy check is included for the radial functions and their first derivatives. This is obtained by comparing the theoretical value of the Wronskian $W\,[R^{(1)}, R^{(2)}]$ of the radial solutions to the value actually calculated from the radial functions and their first derivatives. It gives either the number of digits that agree in the theoretical and calculated Wronskians (or one less, because of truncation error). When $X = 0$, either $R^{(1)}$ or its first derivative is equal to zero. The Wronskian is then insensitive to inaccuracies in either $R^{(2)}$ or its first derivative. In this case the accuracy is determined by subtracting from 25 the number of significant figures that are inaccurate due to subtraction errors (Ref. 11).

Experience has demonstrated that this program will deliver correct results if the eigenvalues $A_{m\ell}$ are correct and the Wronskians check as noted above. Examination of the eigenvalues for continuity is a helpful check on their correctness.

A sample page of the output from OBRAD is presented in Appendix C.

PARAMETER RANGES

To use this program effectively, it is necessary to understand the limitations on the four parameters M, H, L, and X. The ranges that the program has been tested for are as follows:

$$M = 0 \text{ through } 10$$
$$H = 0.01 \text{ through } 75$$
$$L = M \text{ through } M + 49$$
$$X = 0;\ 0.02 \text{ through } 100$$

Limitations on parameters are as follows.

M: In general, accuracy is not much affected with increasing M. For this reason one could reasonably expect good results for values of M greater than 10.

H: The range on H may easily be extended in both directions. Good results should be obtained for values of H as small as 0.001. However, the accuracy may fall off for values of H greater than 75, especially when X is small and L is large. Since the matrix determination of starting values for the eigenvalues as programmed in OBRAD is inadequate for values of H greater than 75, a formula given by Meixner (2) is used in this case.

L: The upper limit on L may be extended beyond $M + 49$. For $H \leq 20$, L can probably be extended to $M + 79$. As H is increased from 20 to 75, the upper limit on L must be reduced from $M + 79$ to the present limit of $M + 49$. As was mentioned, a larger matrix for computing eigenvalues would be required to extend the upper limit on L beyond this. When the range is extended, the eigenvalues should be examined carefully for continuity. Since the difference between successive eigenvalues becomes nearly constant for large L, this is the best check on their validity.

X: The range for X was determined by the physical problem. $X = 0$ represents the surface of a disk and is useful for this reason. The flattest near-disk that one might consider would probably correspond to a value of X no less than 0.02. The upper limit on X was chosen arbitrarily and could probably be extended with little difficulty to well over 100.

ACCURACY CURVES

Several graphs of the calculated accuracy as a function of H and for a fixed value of X are given below in Figs. 1 through 5. The arrow indicates the range of accuracy for $L = M$ to $L = M + 49$ and for $M = 0$ to $M = 10$. The lower accuracy usually corresponds to higher L. For the parameter ranges listed above OBRAD will produce values for $R^{(1)}$ and its first derivative that are accurate to at least 20 significant figures. When the Wronskian check is less than 20 significant figures, it indicates lower accuracy only in $R^{(2)}$ and its first derivative.

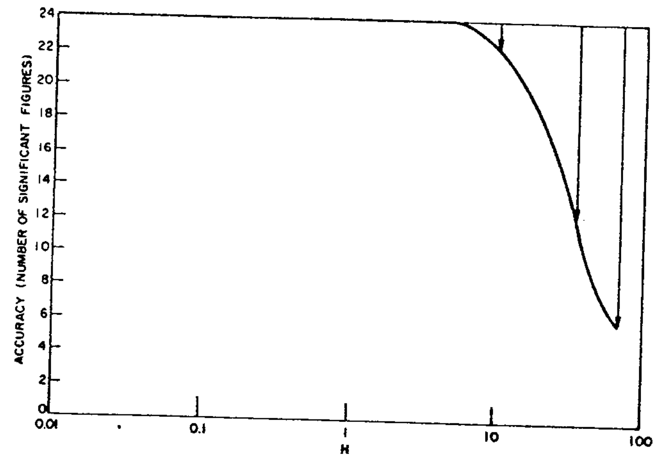

Fig. 1 Calculated accuracy as a function of H for $X = 0.0$

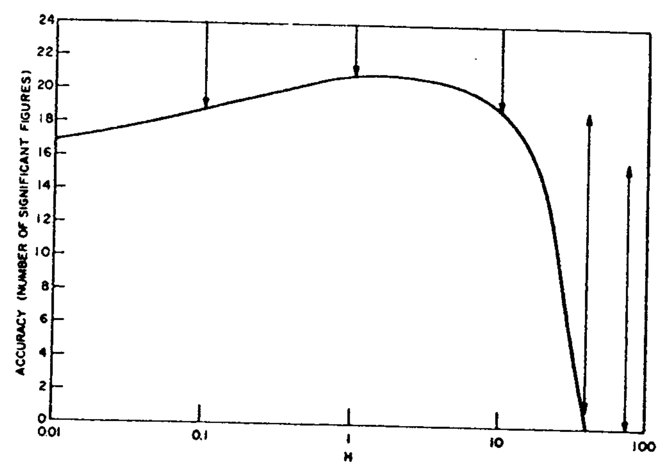

Fig. 2 Calculated accuracy as a function of H for $X = 0.02$

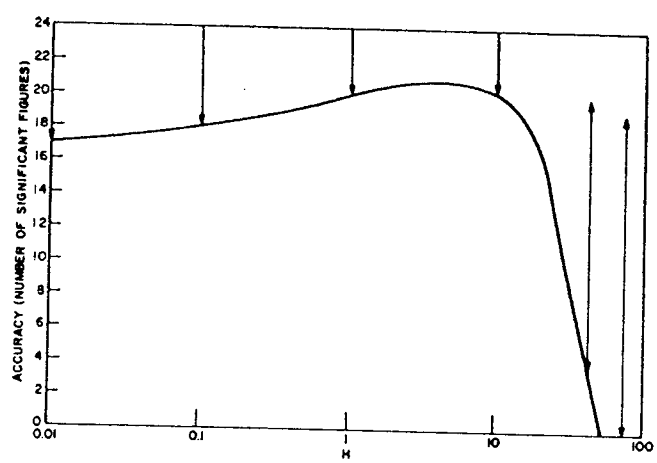

Fig. 3 Calculated accuracy as a function of H for $X = 0.10$

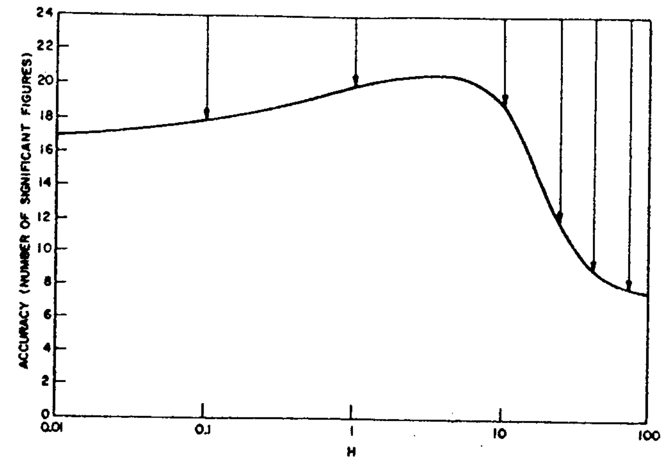

Fig. 4 Calculated accuracy as a function of H for $X = 1.00$

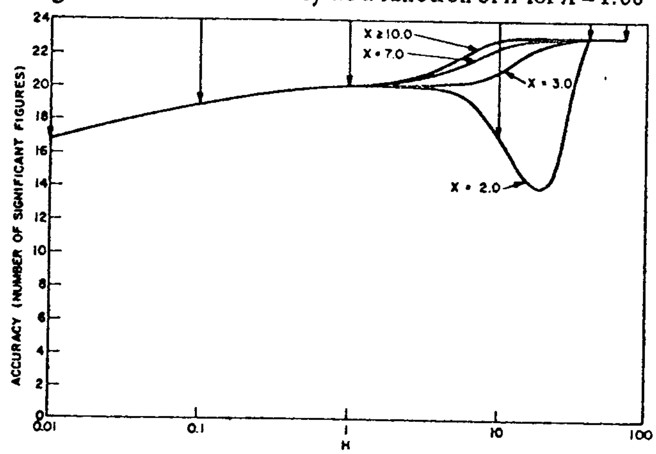

Fig. 5 Calculated accuracy as a function of H for $X \geq 2.00$

COMPUTATION TIME

Using the CDC 3800 the compilation time of OBRAD is about 48 seconds. The execution time varies, but if all 50 values of L are requested, the average time will be about 0.4 second for each set of $R1$, $R1D$, $R2$, $R2D$, and eigenvalue.

SOLUTION OF THE HELMHOLTZ EQUATION IN OBLATE SPHEROIDAL COORDINATES

Details of the oblate spheroidal coordinate system, which is an orthogonal coordinate system, are given in Ref. 1. Briefly, the three oblate spheroidal coordinates are ξ, η, and φ, where $0 \leq \xi \leq \infty$, $-1 \leq \eta \leq 1$, and $0 \leq \varphi \leq 2\pi$. The surfaces of constant ξ, the radial coordinate, are represented in the xyz Cartesian system by the locus

$$\frac{x^2+y^2}{\left(\frac{d}{2}\right)^2\left(\xi^2+1\right)} = 1 - \frac{z^2}{\left(\frac{d}{2}\right)^2\xi^2}. \tag{1}$$

This describes an oblate spheroid whose interfocal distance is d. The surfaces of constant η, the angle coordinate, are represented by the locus

$$\frac{x^2+y^2}{\left(\frac{d}{2}\right)^2\left(1-\eta^2\right)} = 1 + \frac{z^2}{\left(\frac{d}{2}\right)^2\eta^2}. \tag{2}$$

This defines a hyperboloid of one sheet whose focus is located a distance $d/2$ from the origin of the Cartesian system.

The surfaces of constant φ, the rotational coordinate, are half planes whose edge is the z axis.

Any point in three-space can then be represented by the triad (ξ, η, φ).

The scalar wave equation $(\nabla^2 + k^2)\,\Psi = 0$ is separable in oblate spheroidal coordinates. Adopting with slight modifications the notation of Morse and Feshbach (3), we write

$$\Psi = S(ih, \eta)\, R(ih, -i\xi) \begin{matrix}\cos m\varphi \\ \sin m\varphi\end{matrix}, \tag{3}$$

where

$$h = \frac{kd}{2}. \tag{4}$$

The angle function S and the radial function R satisfy the ordinary differential equations

$$\frac{d}{d\eta}\left[(1-\eta^2)\frac{dS}{d\eta}\right] + \left(A + h^2\eta^2 - \frac{m^2}{1-\eta^2}\right)S = 0, \tag{5}$$

$$\frac{d}{d\xi}\left[(1+\xi^2)\frac{dR}{d\xi}\right] - \left(A - h^2\xi^2 - \frac{m^2}{1+\xi^2}\right)R = 0. \tag{6}$$

Here A represents a separation constant dependent on m, ℓ, and h. There are two solutions to both Eqs. (5) and (6). Consider only the first solution $S^{(1)}$ to Eq. (5). When $h \to 0$, Eq. (5) reduces to the standard equation for the associated Lengendre function $P_\ell^m(\eta)$ of the first kind, where the separation constants are $A = \ell(\ell+1)$; $\ell = m, m+1, \ldots$. Thus, for each pair of integers m and ℓ, both Eq. (5) and Eq. (6) have a solution only for special values of $A = A_{m\ell}(ih)$. For $h \neq 0$ we can write

$$S_{ml}^{(1)}(ih, \eta) = {\sum_n^{\infty}}' d_n(ih|m\ell)\, P_{m+n}^m(\eta). \tag{7}$$

The prime sign means that $n = 0, 2, 4, \ldots$ if $l - m$ is even and $n = 1, 3, 5, \ldots$ if $l - m$ is odd. Substituting Eq. (7) into Eq. (5) and using known recursion formulas for $P_\ell^m(\eta)$, one obtains the following three-term formula for the expansion coefficients:

$$- \frac{(2m+n+2)\,(2m+n+1)\,h^2}{(2m+2n+3)(2m+2n+5)} d_{n+2}$$

$$+ \left[(m+n)\,(m+n+1) - A_{m\ell} - \frac{2\,(m+n)\,(m+n+1) - 2m^2 - 1}{(2m+2n-1)\,(2m+2n+3)} h^2 \right] d_n \tag{8}$$

$$- \frac{n\,(n-1)\,h^2}{(2m+2n-3)\,(2m+2n-1)}\; d_{n-2} = 0,$$

with the asymptotic relation

$$\frac{d_{n+2}}{d_n} \xrightarrow[n \to \infty]{} 0 \tag{9}$$

and the normalization

$$\sum_n{}' \; \frac{(n+2m)!}{n!} d_n\,(ih|m\ell) = \frac{(\ell+m)!}{(\ell-m)!}\,. \tag{10}$$

A knowledge of $A_{m\ell}$ would then allow d_n to be calculated by an iterative process.

This program, however, is concerned with only the radial functions $R^{(1)}_{ml}\,(ih, -i\xi)$ and $R^{(2)}_{ml}\,(ih, -i\xi)$. Using the general principle that any solution of the scalar wave equation (say the angle function S) is a suitable kernel for the integral representation of a second solution (say the radial function $R^{(1)}$) the following expression for the radial function of the first type $R^{(1)}$ is obtained by integration over S:

$$R^{(1)}_{ml}\,(ih, -i\xi) = \frac{(\ell-m)!}{(\ell+m)!} \left(\frac{\xi^2+1}{\xi^2} \right)^{m/2} \sum_n{}' \, i^{\,n+m-\ell} \, \frac{(n+2m)!}{n!} \, d_n\,(ih|m\ell)\; j_{n+m}\,(h\,\xi), \tag{11}$$

where j_n is the spherical Bessel function of the first type.

Using a known recursion formula for the spherical Bessel functions, the derivative of $R^{(1)}$ is obtained:

$$\frac{dR^{(1)}_{ml}}{d\xi}\;(ih, -i\xi) = \frac{(\ell-m)!}{(\ell+m)!} \left(\frac{\xi^2+1}{\xi^2} \right)^{m/2} \sum_n{}' \, i^{\,n+m-\ell} \, \frac{(n+2m)!}{n!} \, d_n\,(ih|m\ell)$$

$$\times \left[\frac{h\,(n+m)}{(2n+2m+1)} j_{n+m-1}\,(h\xi) - \frac{h(n+m+1)}{2n+2m+1} j_{n+m+1}\,(h\xi) - \frac{m}{(\xi^3+\xi)} j_{n+m}\,(h\xi) \right]. \tag{12}$$

Using an asymptotic form for the spherical Bessel function, we can find asymptotic forms for the radial function of the first kind:

$$R_{ml}^{(1)} \quad (ih, -i0) = \frac{(\ell - m)!}{(\ell + m)!} \frac{i^{m-\ell} (2m)!\, d_0\,(ih|m\ell)\, h^m}{(2m+1)!!}, \quad \ell - m = \text{even},$$

$$= 0, \qquad \ell - m = \text{odd}, \tag{13}$$

$$\frac{dR_{ml}^{(1)}}{d\xi} (ih, -i0) = \frac{(\ell - m)!}{(\ell + m)!} \frac{i^{m-\ell+1} (2m+1)!\, d_1\,(ih|m\ell)\, h^{m+1}}{(2m+3)!!}, \quad \ell - m = \text{odd},$$

$$= 0, \qquad \ell - m = \text{even}. \tag{14}$$

Here $(2m + 1)!! = (2m + 1)(2m - 1)(2m - 3) \ldots (3)(1)$. Similarly the radial function of the second kind $R_{ml}^{(2)}(ih, -i\xi)$ can be expanded in terms of $y_n(h\xi)$, the Neumann function or spherical Bessel function of the second kind:

$$R_{ml}^{(2)} \quad (ih, -i\xi) = \frac{(\ell - m)!}{(\ell + m)!} \left(\frac{\xi^2 + 1}{\xi^2}\right)^{m/2} \sum_n{}' i^{n+m-\ell} \frac{(n+2m)!}{n!} d_n\,(ih|m\ell)\, y_{n+m}(h\xi),$$

$$\frac{dR_{ml}^{(2)}}{d\xi} (ih, -i\xi) = \frac{(\ell - m)!}{(\ell + m)!} \left(\frac{\xi^2 + 1}{\xi^2}\right)^{m/2} \sum_n{}' i^{n+m-\ell} \frac{(n+2m)!}{n!} d_n\,(ih|m\ell) \tag{15}$$

$$\times \left[\frac{h(n+m)}{2n+2m+1} y_{n+m-1}(h\xi) - \frac{h(n+m+1)}{2n+2m+1} y_{n+m+1}(h\xi) - \frac{m}{(\xi^3 + \xi)} y_{n+m}(h\xi)\right]. \tag{16}$$

Since this expansion contains the Neumann function, its usefulness is limited to large values of $h\xi$. A second method is necessary to obtain the radial function of the second kind for small values of $h\xi$.

Consider the special expansion of the oblate angle function first discovered by Baber and Hasse (4):

$$S_{m\ell}(ih, \eta) = e^{h\eta} \sum_{n=0}^{\infty} \mathfrak{A}_n^{m\ell} \; P_{m+n}^m \; (\eta). \tag{17}$$

When this is substituted into Eq. (5), and the recursion formulas of $P_\ell^m(\eta)$ used, it is found that the expansion coefficients $\mathfrak{A}_n^{m\ell}$ satisfy the three-term recursion formula

$$\begin{aligned} &\frac{2h\,(n+m+1)\,(n+2m+1)}{(2n+2m+3)} \mathfrak{A}_{n+1}^{m\ell} \\ &- \left[(n+m)\,(n+m+1) - A_{m\ell} - h^2\right] \mathfrak{A}_n^{m\ell} \\ &- \frac{2hn\,(n+m)}{2n+2m-1} \mathfrak{A}_{n-1}^{m\ell} = 0, \end{aligned} \tag{18}$$

with the asymptotic condition

$$\frac{\mathfrak{A}^{m\ell}_{n+1}}{\mathfrak{A}^{m\ell}_{n}} \xrightarrow[n \to \infty]{} \frac{h}{n}. \tag{19}$$

By substituting $i\xi$ for η and Q^m_{m+n} for P^m_{m+n}, one can obtain a series expansion of an oblate wave function in the radial coordinate ξ. Noting that the asymptotic value of $Q(z)$, the associated Legendre function of the second kind, is

$$Q^s_t\ (z) \propto \frac{1}{z^{t+1}}, \tag{20}$$

we can use Eq. (17) in its modified form to obtain an expansion of the radial function of the third kind ($\equiv R^{(3)}$) that will have a radial dependence $e^{ih\xi}/h\xi$ as $\xi \to \infty$. Allowing for appropriate constants, this procedure leads to the formula

$$R^{(3)}_{m\ell}(ih, -i\xi) = e^{i\left[h\,\xi - (\ell+1)(\pi/2)\right]} \sum_{n=-m}^{\infty} \frac{i}{m!\,h}^{2m+1} \frac{\mathfrak{A}^{m\ell}_{n}}{\mathfrak{A}^{m\ell}_{-m}}\ Q^m_{m+n}\ (i\xi)\ . \tag{21}$$

Now the radial function of the third kind is related to the radial functions of the first and second kind by the formula

$$R^{(3)} = R^{(1)} + i\,R^{(2)}\,. \tag{22}$$

Equation (21) can be separated into real and imaginary parts:

$$R^{(3)} = (\alpha + i\beta)\,(\gamma + i\delta)\,, \tag{23}$$

where
$$\alpha + i\beta = e^{i\left[h\,\xi - (\ell+1)\,(\pi/2)\right]} \tag{24}$$

and
$$\gamma + i\delta = \sum_{n=-m}^{\infty} \frac{i}{m!\,h}^{2m+1} \frac{\mathfrak{A}^{m\ell}_{n}}{\mathfrak{A}^{m\ell}_{-m}} Q^m_{m+n}\ (i\xi)\ . \tag{25}$$

We can now identify
$$R^{(1)} = \alpha\gamma - \beta\,\delta \tag{26}$$

and
$$R^{(2)} = \alpha\,\delta + \beta\,\gamma \tag{27}$$

or
$$R^{(2)} = \alpha\,\delta + \beta\left[\frac{R^{(1)} + \beta\,\delta}{\alpha}\right]. \tag{28}$$

The form given by Eq. (28) is used for $R^{(2)}$ to avoid computational difficulties associated with γ. $R^{(2)}$ can now be calculated using Eqs. (24) and (25) and the previously calculated value for $R^{(1)}$.

Similarly the first derivative of the second radial function $R^{(2)}$ is obtained:

$$\frac{dR^{(2)}}{d\xi} = \alpha\,\nu + \beta\left[\frac{\frac{dR^{(1)}}{d\xi} + \beta\,\nu}{\alpha}\right], \tag{29}$$

where

$$\mu + i\nu = \sum_{n=-m}^{\infty} \frac{\mathcal{A}_n^{m\ell}}{\mathcal{A}_{-m}^{m\ell}} \left\{ Q^m_{n+m}(i\xi) \left[\frac{\xi(m+n)}{\xi^2+1} + ih \right] + i\,\frac{(2m+n)}{\xi^2+1}\, Q^m_{m+n-1}(i\xi) \right\}. \tag{30}$$

When $X = 0$, asymptotic forms can be used for $R^{(2)}$ and $dR^{(2)}/d\xi$ which take advantage of the loss of independence of $R^{(1)}$and $R^{(2)}$. These special formulas are obtained by rewriting Eqs. (4.6.15) and (4.6.16) in Flammer (10) to include $d_n(ih\,|m\ell)$ satisfying Eq. (10).

DESCRIPTION OF THE COMPUTER PROGRAM OBRAD

The Fortran IV computer program used to calculate the oblate spheroidal radial functions of the first and second kinds, their first derivatives, and the eigenvalues is listed in Appendix B. Some details of this program are given below.

Expansion Functions

Several special functions are required: the factorial functions, the associated Legendre functions of the second kind, and the spherical Bessel functions of the first and second kinds.

1. The factorials are calculated in the main program in statement 96 + 2 lines through statement 53. It was necessary to scale FACT $(N + 1) = N!$ for $N = 170$ to 296 to prevent overflow, since the maximum exponent available on the CDC 3800 at NRL is 307.

2. The associated Legendre functions of the second kind, $Q_n^m(iX)$, where X is the radial coordinate, is calculated in the subroutine QLEG. QLEG is called after M and X have been set and when $H = AH$, the first choice for H, since $Q_n^m(iX)$ is independent of H. It returns values of $Q_n^m(iX)$ for $N = 0$ to $N = 126 + 2M$. $Q_n^m(iX)$ is either purely real or purely imaginary depending on whether N is even or odd respectively. Therefore when N is odd, the real answer returned by QLEG must be multiplied by i to obtain $Q_n^m(iX)$. For fixed M these values are stored for all choices of X in the matrix OUTPUT $(N + 2, IX)$. Here IX indicated the specific X, and $N + 2$ is chosen so that the first element stored in OUTPUT is $Q_{-1}^m(iX)$. When M changes, QLEG is again called for each X when $H = AH$.

$Q_{-1}^m(iX)$ is calculated in the main program in statements 4 through 5 using $Q_0^m(iX)$ and $Q_1^m(iX)$ in a backward recursion formula. QLEG uses limiting forms for $Q_n^m(iX)$ when $X = 0$. When $X > 0$, $Q_n^0(iX)$ is calculated from a hypergeometric series, and $Q_n^m(iX)$ is then obtained by a forward recursion formula. These expressions are given as Eqs. 30 through 32 in Ref. 1 and Eq. 9 on page XVI in Ref. 5.

The output of QLEG was carefully checked for the entire range of M and X necessary for OBRAD and found to have an accuracy of at least 20 significant figures.

3. The spherical Bessel function of the first kind is calculated in the subroutine SBESF. SBESF is called after H and X have been set and when L is equal to $L1$, the first choice for L. It returns values for $j_n\,(HX)$ for $N = 0$ to $N = 145$, unless HX is greater than or equal to 100 when it returns values for $N = 0$ to $N = 145 + M$. SBESF calculates $j_n\,(HX)$ by a series expansion when $HX < 0.4$, by a backward recursion relation when $0.4 \leq HX < 100.0$, and by a forward recursion relation when $HX \geq 100.0$. These expressions are given in Ref. 6 as 10.1.2 and 10.1.19. The accuracy in $j_n\,(HX)$ is greater than 20 significant figures for the entire range of HX necessary for OBRAD.

4. The spherical Bessel function of the second kind is calculated in the subroutine SPHYN by a forward recursion relation given as Eq. 10.1.19 in Ref. 6. If $X > 1.0$ and $HX > 10.0$, SPHYN is called after H and X have been set and when L is equal to $L1$. It returns values of $y_n\,(HX)$ for $N = 0$ to $N = 143 + M$. The accuracy in $y_n\,(HX)$ when $HX > 1.0$ is greater than 20 significant figures.

Eigenvalues

Before the expansion constants can be evaluated, it is necessary to know the eigenvalues or separation constants for which solutions to Eqs. (5) and (6) exist.

Starting values or numbers agreeing to at least two places with the correct values are obtained for the eigenvalues. These starting values are solutions to an eigenvalue equation which when expressed in matrix form reduces to the problem of diagonalization of the matrix. The eigenvalues then appear as the resulting diagonal elements when ordered numerically from lowest to highest. Although the exact determination of the eigenvalues would require a matrix of order infinity, good starting values are obtained using matrices of modest proportions. The minimum size matrix used in OBRAD is of order 50, giving 50 possible starting values. When $H \leq 20$, all 50 values are adequate as starting values, with the lowest eigenvalue corresponding to $L = M$. However, as H increases, the order N of the matrix must also be increased to maintain good starting values for the 50 lowest eigenvalues. The order N as determined in statement 3 + 14 lines through statement 3 + 18 lines is adequate to give good starting values for the 50 lowest eigenvalues when H is less than or equal to 75.

The matrix elements A are obtained in statement 3 + 19 lines through statement 43. Subroutine EIGEN, which diagonalizes the matrix A, is then called. Details of the matrix and its diagonalization are given in Ref. 7. EIGEN returns the N diagonal elements in ascending numerical order. $LF - 1 + M$ of these, where LF is the highest L desired, are now used as starting values in a variational procedure devised by Bouwkamp (8) and Blanch (9). (When H is greater than 75, good starting values are obtained instead from a formula given by Meixner (2). This formula is programmed in statements 35 through 37.) This variational method adds corrections to the starting values, the corrections becoming successively smaller as the correct eigenvalue is approached. Good starting values are necessary to assure the convergence to correct eigenvalues. Convergence is assumed when the relative contribution of the correction is less than 10^{-24}. Because of the limited word length in the CDC 3800 at NRL a more stringent test does not give more accurate eigenvalues, but the corrections oscillate around 10^{-25}. This variational method is programmed in statement 6 + 3 lines through statement 22 + 1 line.

Expansion Constants

1. The first expansion constants that are used $d_n\ (i\lambda|m\ell)$ are calculated in statements 31 through 32 + 1 line. These calculations make use of the single subscripted variable ENR that has been obtained above in the eigenvalue correction. Although the method is disguised by the intermediate variable ENR, basically the expansion constants $d_n\ (i\lambda|m\ell)$ (called DLIST(J)) are calculated using Eqs. (8) through (10). Here the index J runs consecutively from 1 to 72. For example, DLIST (3) represents $d_5\ (i\lambda|m\ell)$ when $L - M$ is odd but $d_4(i\lambda|m\ell)$ when $L - M$ is even.

2. The expansion constants $\mathfrak{A}^{m\ell}_{n}/\mathfrak{A}^{m\ell}_{-m}$, used in the calculation of the radial function of the second kind, are obtained in statements 252 through 291.

First uncorrected values RATIO (J) are successively calculated by use of the reverse recursion form of Eq. (18) until they begin to decrease ($J = IND + 1$). Here RATIO ($123 + 2M$) is chosen equal to 0, and RATIO ($122 + 2M$) is chosen equal to 1.

Next using the fact that $\mathfrak{A}^{m\ell}_{-m-1}/\mathfrak{A}^{m\ell}_{-m} \equiv$ ARATIO (1) = 0 and $\mathfrak{A}^{m\ell}_{-m}/\mathfrak{A}^{m\ell}_{-m} \equiv$ ARATIO (2) = 1, true values ARATIO (J) are obtained by use of the forward recursion form of Eq. (18) until $J = IND + 1$.

Finally RATIO (J) is corrected by matching to ARATIO (J) at $J = INL' + 1$.

Evaluation of the Radial Functions

The expansion constants and functions are now combined to give the radial functions. The radial functions of both kinds and their first derivatives are calculated for $X = 0$ in statement 237 through statement 246 + 3 lines. For $X \neq 0$ the radial function of the first kind and its first derivative are calculated in statements 211 through 236 using the expansions given in Eqs. (11) through (14) and the radial function of the second kind and its first derivative are calculated in statement 291 + 1 line through statement 311 + 1 line using Eqs. (28) and (29).

A Wronskian check is made on the two radial functions and their first derivatives in statements 311 + 2 lines through 311 + 4 lines. Here the calculated Wronskian CWRON is compared to the theoretical Wronskian TWRON to give the number of significant figures that agree NIAC. When $X = 0$ the accuracy is determined instead by subtracting from 25 the number of accurate figures that are lost during subtraction of nearly equal numbers.

When $X > 1.0$ and $XH > 10.0$, Eqs. (15) and (16) are also used to calculate the radial function of the second kind and its first derivative. This is done in statement 311 + 7 through statement 324. A Wronskian check is made using these values, yielding the integer IAC.

IAC is now compared with NIAC in order to choose between the two sets of values for the radial function of the second kind and its first derivative. If IAC > NIAC, the results obtained using Eqs. (15) and (16) are printed. If NIAC $\geq$ IAC, the results of Eqs. (28) and (29) are printed instead.

REFERENCES

1. Hanish, S., Baier, R.V., Buckler, W.H., and Carmichael, B.K., "Tables of Spheroidal Wave Functions, Part 3 – Second Volume of Oblate Radial Functions of the First and Second Type," NRL Report 6543, Apr. 1967

2. Meixner, J., "Asymptotische Entwicklung der Eigenwerte und Eigenfunktionen der Differentialgleichungen der Sphäroid–Funktionen und der Mathieuschen Funktionen," Zeitshrift Angew. Math. Mech. 28:304–310 (1948)

3. Morse, P.M., and Feshbach, H., "Methods of Theoretical Physics," Vol II, New York: McGraw-Hill, 1953, p. 1502

4. Baber, W.G., and Hassé, H.R., "The Two Centre Problem in Wave Mechanics," Proc. Cambr. Phil. Soc. 31:564–581 (1935)

5. "Tables of Associated Legendre Functions," National Bureau of Standards, Mathematical Tables Project, New York: Columbia University Press, 1945

6. "Handbook of Mathematical Functions," National Bureau of Standards, Applied Mathematical Series 55, Washington: U.S. Govt. Printing Office, 1964

7. Hanish, S., and King, B., "Tables of Eigenvalues of the Wave Equation in Prolate Spheroidal Coordinates," NRL Report 6856, Apr. 1969

8. Bouwkamp, C.J., "Theoretische en Numerieke Behandeling van der Buiging door een Rone Opening," Diss. Groningen, Grongingen – Batavia, 1941

9. Blanch, G., "On the Computation of Mathieu Functions," J. Math. Phys. (M.I.T.) 25:1–20 (1946)

10. Flammer, C., "Spheroidal Wave Functions," Stanford University Press, 1957, pp. 42-43

11. Scarborough, J. B., "Numerical Mathematical Analysis," The Johns Hopkins Press, 1966, p. 12

Appendix A

Major Calculation Blocks of OBRAD Listed According to Statement Numbers

Calculation Block	Statement Number From	Statement Number To
Calculate factorials	*96 + 2	53
Read data	1	3 + 3
Do loops		
Set M	3 + 4	
Set H	3 + 10	
Calculate the starting eigenvalues		
by Subroutine EIGEN	3 + 14	7
by Meixner's formula	35	37
Set X	38	
Generate Q's Using Subroutine QLEG	38 + 5	5
Set L	6 + 1	
Correct the eigenvalues	6 + 3	22 + 1
Calculate constants d	31	32 + 1
Calculate radial functions $R1$, $R1D$, $R2$, $R2D$ for $X = 0$	237	246 + 3
Calculate $R1$, $R1D$ using SBESF	211	236
Calculate $\mathcal{Q}$ ratios	252	291
Calculate $R2$, $R2D$ by use of Q functions	291 + 1	311 + 1
Decide whether to use Neumann functions to calculate $R2$, $R2D$	311 + 5	

*Note: The symbol 96 + 1 signifies statement number 96 plus one line.

Calculation Block	From	To
Calculate $R2$, $R2D$ by means of spherical Neumann functions, subroutine SPHYN	311 + 7	324
Calculate Wronskian (plus accuracy check)	311 + 2 and 324 + 1	311 + 5 324 + 2
Decide to print out final result based on Legendre function approach or on Bessel function approach	324 + 3	
Final printout	326 + 1	

UNCLASSIFIED

Appendix B

LISTING IN FORTRAN IV OF OBRAD,

A PROGRAM TO CALCULATE OBLATE

SPHEROIDAL RADIAL WAVE FUNCTIONS,

AS COMPILED FOR CDC 3800

```
      PROGRAM OBRAD
      TYPE DOUBLE AA,AAA,AH,AR,ARATIO,ARG,ARRAY,ATERM,BLIST,BOOK,CL,
     1COEFF1,COEFF2,COEFF3,CORA,CORB,CWRON,DE,DH,DL,DLIST,DN,DX,E1,E2,
     2E3,EJN,EM,ENR,ENRC,ESTORE,ESUM,EYE,FACT,FN1,FN2,FNM,GLIST,H,PI,
     3OUTPUT,PAPER,PCL,PEN,PL8,PL9,PLA,PLB,PLC,PLD,R1,R2,R3,RAD1,RAD1D,
     4RAD2,RAD2D,RATIO,RSTORE,RSUM,S,SSUM1,SUBSUM,TEMP,TERM,TERM1,TERM2,
     5TERM3,TWRON,W,X,X1,XL,XX,EIG,Q,CRAD2,CRAD2D,DW,FSTRAT,DRATIO,DNEG
      DIMENSION A(80,80),ARATIO(250),ARRAY(250),BLIST(250),BOOK(250),
     1COEFF1(250),COEFF2(250),COEFF3(250),DLIST(200),DN(200),EIG(80),
     2ENR(250),FNM(250),GLIST(250),OUTPUT(170,40),PAPER(250),PCL(250),
     3PEN(250),RATIO(250),Q(200),AIG(80),DRATIO(30)
      COMMON FACT(300)
   86 FORMAT(1X,I3,2X,5(D24.17,1X),1X,I3)
   87 FORMAT(1H1,56X,*OBLATE RADIAL FUNCTIONS*,///,47X,*H =*,F5.1,9X,
     1*X =*,F6.2,9X,*M =*,I3,///,3X,*L*,13X,*R1*,23X,*R1D*,22X,*R2*,
     223X,*R2D*,18X,*EIGENVALUE* ,9X,*ACC*/)
   90 FORMAT(8I4)
   96 FORMAT(D32.25)
      PI=3.14159265358979323846264340
      FACT(1)=1.0D
      DO 51 J=1,170
   51 FACT(J+1)=J*FACT(J)
      FACT(171)=FACT(171)*(1.D-300)
      DO 53 J=171,296
   53 FACT(J+1)=J*FACT(J)
    1 READ 90,M1,IDM,NM,L1,IDL,NL,NH,NX
      IF(EOF,60) 502,3
    3 READ 96, AH
      READ 96,DH
      READ 96, X1
      READ 96,DX
      DO2IM=1,NM
      M=M1+(IM-1)*IDM
      EM=M
      IF(L1.LT.M) L1=M
      LF=L1+(NL-1)*IDL
      H=AH-DH
      DO2IH=1,NH
      H=H+DH
      AA=-H*H
      AAA=AA*AA
      N=N1=LF-M+1
      IF(H.GT.75.D)  GO TO 35
      IF(H.GT.20.D) N=N1+(H-20.D)/2.D
      IF(N.LT.50) N=50
      IF(N.GT.80) N=80
      DO41 J = 1, N
      DO41 I = 1,N
   41 A(I,J) = 0.
      DO42 I = 1,N
      XL = M + I - 1
   42 A(I,I) =XL*(XL+1.)+AA*(2.*XL*(XL+1.)-2.*EM*EM-1.)/((2.*XL-1.)*
     1(2.*XL+3.))
      NM2 = N-2
      DO43 I = 1,NM2
      XL = M + I - 1
      A(I,I+2) = (AA/(2.*XL+3.))* SQRT(((XL+2.+EM)*(XL+1.+EM)*
     1(XL+2.-EM)*(XL+1.-EM))/((2.*XL+5.)*(2.*XL+1.)))
   43 A(I+2,I) = A(I,I+2)
```

```
      ANORM1=AA
      ANORM2=LF*(LF+1.)
      CALL EIGEN(A,AIG,N,N1,ANORM1,ANORM2)
      DO 7I=1,N1
    7 EIG(I)=AIG(I)
      GO TO 38
   35 DO 37 L=L1,LF,2
      P=2.*((L-M)/2)+M+1.
      EIG(L-M+1)=-H*H+2.*H*P-.5*(P*P-M*M+1)-P*(P*P-M*M+1)  /(8.*H)
     1-(5.*P**4+10.*P*P+1.-2.*M*M*(3.*P*P+1.)+M**4)/(64.*H*H)
     2-P*(33.*P**4+114.*P*P+37-2.*M*M*(23.*P*P+25.)+13.*M**4)/(512.*H**3
     3) -(63.*P**6+340.*P**4+239.*P*P+14.-10.*M*M*(10.*P**4+23.*P*P+3.)
     4+3.*M**4*(13.*P*P+6.)-2.*M**6)/(1024.*H**4)
     5-P*(527.*P**6+4139.*P**4+5221.*P*P+1009.-M*M*(939.*P**4+3750.*P*P
     6+1591.)+M**4*(465.*P*P+635.)-53.*M**6)/(8192.*H**5)
      IF(2*((L-M)/2).NE.L-M)  EIG(L-M)=EIG(L-M+1)
   37 EIG(L-M+2)=EIG(L-M+1)
   38 DO2IX=1,NX
      X=X1+(IX-1)*DX
      XX=X*X+1.D
      ARG=H*X
      IF(H.NE.AH.OR.X.EQ.0.D) GO TO 6
      LQ=125+2*M
      CALL QLEG(M,LQ,X,Q)
      DO 8I=1,LQ
    8 OUTPUT(I+1,IX)=Q(I)
      IF(M.EQ.0)4,5
    4 OUTPUT(1,IX) =0.D
      GO TO 6
    5 OUTPUT(1,IX)=-X*OUTPUT(2,IX)/EM+(EM-1.D)*OUTPUT(3,IX)/EM
    6 PRINT 87,H,X,M
      DO2IL=1,NL
      L=L1+(IL-1)*IDL
      PLB=2.D*EM+1.D
      IFC=0
      IUCT=(L-M)/2
      IRIO=IUCT+1
      IR=IRIO+1
      CL=EIG(L-M+1)
      IF(2*IUCT.NE.(L-M)) GO TO 10
   11 ID=2
      IB=75
      IC=2*M
      GLIST(1)=EM*(EM+1.D)+AA*(PLB-2.D)/((PLB-2.D)*(PLB+2.D))
      GO TO 12
   10 ID=3
      IB=74
      IC=2*M+1
      GLIST(1)=(EM+1.D)*(EM+2.D)+(6.D*EM+3.D)/(PLB*(PLB+4.D))*AA
   12 LIM=150
      IIB=IB-1
      DO 13I=ID,LIM,2
      EYE=I
      BLIST((I-ID+2)/2)=EYE*(EYE-1.D)*(PLB+EYE-1.D)*(PLB+EYE-2.D)*AAA/
     1((PLB+2.D*EYE-4.D)*(PLB+2.D*EYE)*(PLB+2.D*EYE-2.D)*(PLB+2.D*EYE-
     22.D))
   13 GLIST((I-ID+4)/2)=(EM+EYE)*(EM+EYE+1.D)+.5D*AA*((1.D)-(PLB*PLB-2.D
     1*PLB)/((PLB+2.D*EYE-2.D)*(PLB+2.D*EYE+2.D)))
   17 ENR(1)=CL-GLIST(1)
```

```
      DO 18I=1,IUCT
   18 ENR(I+1)=-BLIST(I)/ENR(I)-GLIST(I+1)+CL
      ENR(IB)=-BLIST(IB)/(GLIST(IB+1)-CL)
      IP=IIB+IR
      DO 19 I=IR,IIB
      IPI=IP-I
   19 ENR(IPI)=-BLIST(IPI)/(GLIST(IPI+1)-CL+ENR(IPI+1))
      ENRC=-BLIST(IRIO)/(GLIST(IR)-CL+ENR(IR))
      DE=ENRC*ENRC/BLIST(IRIO)
      CORB=DE
      DO 20 I=IR,IB
      DE=ENR(I)*ENR(I)/BLIST(I)*DE
      CORB=CORB+DE
   20 IF(DABS(DE/CORB).LT.1.D-27) GO TO 23
   23 CORA = 1.D
      DE=1.D
      DO 26 I=1,IUCT
      DE=BLIST(IRIO-I)/(ENR(IRIO-I)*ENR(IRIO-I))*DE
      CORA=CORA+DE
   26 IF(DABS(DE/CORA).LT.1.D-27) GO TO 27
   27 DL=(ENRC-ENR(IRIO))/(CORA+CORB)
      CL=CL+DL
      IF(DABS(DL/CL).LT.1.D-24) GO TO 22
      IFC=IFC+1
      IF(IFC.LT.50) GO TO 17
   22 CONTINUE
      EIG(L-M+1)=CL
   31 AR=ID
      DN(1)=((2.D*EM+2.D*AR-1.D)*(2.D*EM+2.D*AR+1.D)*ENR(1))/
     1((2.D*EM+AR)*(2.D*EM+AR-1.D)*AA)
      W=DN(1)*FACT(2*M+ID+1)/FACT(ID+1)
      DO30J=2,IB
      AR=ID+2*(J-1)
      DN(J)=DN(J-1)*((2.D*EM+2.D*AR-1.D)*(2.D*EM+2.D*AR+1.D)
     1*ENR(J))/((2.D*EM+AR)*(2.D*EM+AR-1.D)*AA)
      DW= DN(J)*(FACT(2*(M+J)+ID-1)/FACT(ID+2*J-1))
      IF((2*(M+J)+ID-1).GT.170) DW=DW*1.D+300
   30 W=W+DW
      DLIST(1)=FACT(L+M+1)/(FACT(L-M+1)*(W+FACT(IC+1)))
      DO32J=1,70
   32 DLIST(J+1)=DN(J)*DLIST(1)
      DLIST(72)=0.D
      IF(X.NE.0.D)   GO TO  200
      DRATIO(1)=0.D
      DNEG=DLIST(1)
      DO 33 I=1,M
      DRATIO(I+1)=-(I+I+ID-2.D)*(I+I+ID-3.D)*AA/((4*I+ID+ID-M-M-
     15.D)*(4*I+ID+ID-M-M-3.D))/(((I+I+ID-M-4.D)*(I+I+ID-M-3.D)-CL+(2.D*
     2(I+I+ID-M-4.D)*(I+I+ID-M-3.D)-M*M-M*M-1.D)*AA/((4*I+ID+ID-M-M-9.D)
     3*(4*I+ID+ID-M-M-5.D))+AA*(I+I+ID-M-M-4.D)*(I+I+ID-M-M-5.D)*
     4DRATIO(I)/((4*I+ID+ID-M-M-11.D)*(4*I+ID+ID-M-M-9.D)))
   33 DNEG=DNEG*DRATIO(I+1)
      TERM=FACT(M+M+ID+ID-3)/(2.D**(ID-2)*FACT(M+ID-1))
      FSTRAT=TERM*DLIST(1)
      TERM2=DABS(FSTRAT)
      DO 34 I=2,71
      TERM=-TERM*(M+M+I+I+ID+ID-6)*(M+M+I+I+ID+ID-7)/(4.D*(I-1)*(M+I+ID
     1-3))
      TERM1=TERM*DLIST(1)
```

```
      TERM2=DMAX1(TERM2,DABS(TERM1))
   34 FSTRAT=FSTRAT+TERM1
      IAC=26-DLOG10(TERM2/DABS(FSTRAT))
      IF(IAC.GT.25)  IAC=25
      GO TO 237
  200 E1=DSIN(ARG)/(H*FACT(M+1))
      R1=DCOS(ARG)/(H*FACT(M+1))
      GO TO (201,203,205,206),L-(L/4)*4+1
  201 E2 = - R1
      R2 = E1
      GO TO 207
  203 R2 = - R1
      E2 = - E1
      GO TO 207
  205 E2 = R1
      R2 = - E1
      GO TO 207
  206 R2 = R1
      E2 = E1
  207 GO TO (208,209,208,209),M-(M/4)*4+1
  208 E3 = R2
      R3 = - E2
      GO TO 210
  209 E3 = - R2
      R3 = E2
  210 IF(L.NE.L1) GO TO 211
      LNE=145
      IF(X*H.GE.100.D) LNE=LNE+M
      CALL SBESF(ARG,LNE,ARRAY)
  211 PLA=DSQRT((XX/(X*X))**M)*FACT(L-M+1)/FACT(L+M+1)
      IA=1
      IF(2*IUCT.NE.(L-M)) IA=IA+1
      IC=IA+142-M
      IF(X*H.GE.100.D) IC=IC+M-4
  214 SUBSUM = 0.D
  215 DO 217  K = IA, IC, 2
      IBOX3=IABS((K+M-L-1)/2)
      BOOK(K)=DLIST((K+1)/2)*ARRAY(K+M)*(FACT(K+2*M)/FACT(K))
      IF((K+2*M).GT.170)  BOOK(K)=BOOK(K)*1.D+300
      IF(2*(IBOX3/2).NE.IBOX3)BOOK(K)=-BOOK(K)
  216 SUBSUM = BOOK (K) + SUBSUM
      IF(DABS(BOOK(K)/SUBSUM).LT.1.D-27) GO TO 219
  217 CONTINUE
  219 RAD1=PLA*SUBSUM
      SSUM1=0.D
      DO 234 K=IA,IC,2
      IBOX3=IABS((K+M-L-1)/2)
      IDS = (K + 1)/2
      PCL(K)=DLIST(IDS)*(FACT(K+2*M)/FACT(K))
      IF((K+2*M).GT.170)  PCL(K)=PCL(K)*1.D+300
      IF(2*(IBOX3/2).NE.IBOX3) PCL(K)=-PCL(K)
      PLD=K+M-1
      IF((K+M-1).EQ.0) 230,231
  230 PEN (K) = - ARRAY (2)
      GO TO 232
  231 PEN(K)=(PLD*ARRAY(K+M-1)-(PLD+1.D)*ARRAY(K+M+1))/(2.D*PLD+1.D)
  232 PAPER (K) = PCL (K) * PEN (K)
  233 SSUM1 = PAPER (K) + SSUM1
      IF(DABS(PAPER(K)/SSUM1).LT.1.D-27) GO TO 236
```

```
234 CONTINUE
236 RAD1D=(H*SSUM1-EM*SUBSUM/(X*XX))*PLA
    GO TO 252
237 DO 239 I=2,M
239 PLB=PLB*(2*M-2*I+3)
    PLC=FACT(L-M+1)*FACT(2*M+1)*DLIST(1)*H**M/(FACT(L+M+1)*PLB)
    GO TO (240,242,244,246),(L-M)-((L-M)/4)*4+1
240 RAD1=PLC
    RAD1D=0.D
    RAD2=(M+M-1)*PI*FACT(M+1)*FACT(L-M+1)*(H/2.D)**M*FSTRAT*FSTRAT
   1/(+FACT(M+M+1)*FACT(L+M+1)*2.D*H*DNEG)
    RAD2D=1.D/(H*RAD1)
    GO TO 326
242 RAD1=0.D
    RAD1D=PLC*(2.D*EM+1.D)*H/(2.D*EM+3.D)
    RAD2=-1.D/(H*RAD1D)
    RAD2D= (M+M-3)*(M+M-1)*FACT(M+1)*FACT(L-M+1)*PI*(H/2.D)**M*
   1FSTRAT*FSTRAT/(FACT(M+M+1)*FACT(L+M+1)*2.D*H*H*DNEG)
    GO TO 326
244 RAD1=-PLC
    RAD1D=0.D
    RAD2=(M+M-1)*PI*FACT(M+1)*FACT(L-M+1)*(H/2.D)**M*FSTRAT*FSTRAT
   1/(-FACT(M+M+1)*FACT(L+M+1)*2.D*H*DNEG)
    RAD2D=1.D/(H*RAD1)
    GO TO 326
246 RAD1=0.D
    RAD1D=-PLC*(2.D*EM+1.D)*H/(2.D*EM+3.D)
    RAD2=-1.D/(H*RAD1D)
    RAD2D=-(M+M-3)*(M+M-1)*FACT(M+1)*FACT(L-M+1)*PI*(H/2.D)**M*
   1FSTRAT*FSTRAT/(FACT(M+M+1)*FACT(L+M+1)*2.D*H*H*DNEG)
    GO TO 326
252 MA=2*M+123
    GO TO 256
254 MA=MA-20
256 JS=MA-M-2
    S=JS
    RATIO(MA)=0.D
    IMA = MA - 1
    RATIO (IMA) = 1.D
    DO 253     J = 1, IMA
    I=IMA-J+1
    COEFF1(I)=2.D*H*(S+EM+1.D)*(S+2.D*EM+1.D)/(2.D*S+2.D*EM+3.D)
    COEFF2 (I) = (S + EM) * (S + EM + 1.D) - CL - H * H
    COEFF3 (I) = 2.D*H*S*(S + EM)/(2.D*S + 2.D*EM -1.D)
253 S = S - 1.D
    ARATIO (2) = 1. D
    ARATIO ( 1) = 0. D
269 DO 280 I=1,JS
    K=JS-I+1+M
    FN1 = COEFF1 (K + 1) * RATIO (K + 2)
    FN2 = COEFF2 (K + 1) * RATIO (K + 1)
270 RATIO (K) = (FN1 - FN2)/COEFF3 (K + 1)
    IF(DABS(RATIO(K)).GT.1.D+300) GO TO 254
    IF(DABS(RATIO(K+1)).GT.DABS(RATIO(K))) GO TO 272
280 CONTINUE
    IND=K
    GO TO 274
272 IND=K+1
274 IF(IND.LT.2) IND=2
```

```
281 DO 289 J=1,K
    IJ = J + 1
    FN1 = COEFF2 (J) * ARATIO (IJ)
    FN2 = COEFF3 (J) * ARATIO (IJ - 1)
    ARATIO (IJ + 1) = (FN1 + FN2)/COEFF1 (J)
289 CONTINUE
    RATIO(1)=RATIO(IND)/ARATIO(IND+1)
    DO 291 I=IND,MA
291 ARATIO(I+1)=RATIO(I)/RATIO(1)
    RSUM=OUTPUT(3,IX)*ARATIO(3)
    ESUM=OUTPUT(2,IX)
    RSTORE=-OUTPUT(2,IX)*((EM+1.D)*ARATIO(3)+H)
    ESTORE=EM*OUTPUT(1,IX)/XX+OUTPUT(3,IX)*ARATIO(3)*H
    IF(M.EQ.0) ESTORE=ESTORE+1.D/XX
    DO 306 K=4,MA
    AR=K
    TERM1=OUTPUT(K,IX)*ARATIO(K)
    TERM3=-ARATIO(K)*(EM+AR-2.D)*OUTPUT(K-1,IX)/XX
    IF(2*(K/2).EQ.K) GO TO 304
    RSUM=RSUM+TERM1
    ESTORE=ESTORE+TERM1*H
    RSTORE=RSTORE+TERM3
    IF(ABS(TERM1/ESTORE).LT.1.D-27.AND.ABS(TERM3/RSTORE).LT.1.D-27
   1.AND.ABS(TERM1/RSUM).LT.1.D-27) GO TO 309
    GO TO 306
304 TERM2=TERM1*X*(AR-2.D)/XX
    ESUM=ESUM+TERM1
    RSTORE=RSTORE-TERM1*H
    ESTORE=ESTORE+TERM2-TERM3
306 CONTINUE
309 RSUM = (RAD1 + E3 * ESUM)/R3
    RSTORE = (RAD1D + E3 * ESTORE)/R3
311 CRAD2=R3*ESUM+E3*RSUM
    CRAD2D=R3*ESTORE+E3*RSTORE
    TWRON=1.D/(H*XX)
    CWRON=RAD1*CRAD2D-CRAD2*RAD1D
    NIAC=-DLOG10(DABS((TWRON-CWRON)/TWRON)+1.D-26)
    IF(X.LE.1.D.OR.(X*H).LE.10.D) GO TO 325
    IF(L.NE.L1) GO TO 313
    LNE=141+M
    CALL SPHYN(ARG,LNE,FNM)
313 JN=1
    IF(2*((L-M)/2).EQ.(L-M))JN=0
    RAD2=0.D
315 K=1+JN/2
    TERM =FACT(JN+2*M+1)/FACT(JN+1)*DLIST(K)*FNM(JN+M+1)
    IF((JN+2*M+1).GT.170)   TERM=TERM*1.D+300
    IF(4*((JN+M-L)/4).NE.(JN+M-L)) TERM =-TERM
    RAD2=RAD2+TERM
    IF(K.LT.5) GO TO 316
    IF(DABS(TERM /RAD2).LT.(1.D-27)) GO TO 318
316 JN=JN+2
    GO TO 315
318 RAD2=RAD2*PLA
    PL9=-EM/(X**3+X)
    RAD2D=0.D
    JN=JN-2*(JN/2)
320 K=1+JN/2
    EJN=JN
```

```
      PL8=H/(2.D*EJN+2.D*EM+1.D)
      TERM =FACT(JN+2*M+1)/FACT(JN+1)*DLIST(K)*(PL8*((EJN+EM)*
     1 FNM(JN+M)-(EJN+EM+1.D)*FNM(JN+M+2))+PL9*FNM(JN+M+1))
      IF((JN+2*M+1).GT.170)   TERM=TERM*1.D+300
      IF(4*((JN+M-L)/4).NE.(JN+M-L))  TERM =-TERM
      RAD2D=RAD2D+TERM
      IF(K.LT.5) GO TO 322
      IF(DABS(TERM /RAD2D).LT.(1.D-27)) GO TO 324
  322 JN=JN+2
      GO TO 320
  324 RAD2D=RAD2D*PLA
      CWRON = RAD1 * RAD2D - RAD2 * RAD1D
      IAC=-DLOG10(DABS((TWRON-CWRON)/TWRON)+1.D-26)
      IF(IAC.GT.NIAC) GO TO 326
  325 RAD2=CRAD2
      RAD2D=CRAD2D
      IAC=NIAC
  326 IF(IAC.LT.0) IAC=0
      PRINT 86,L,RAD1,RAD1D,RAD2,RAD2D,EIG(L-M+1),IAC
    2 CONTINUE
      GO TO 1
  502 END
```

```
      SUBROUTINE SBESF(XH,LJ,RAY)
      DIMENSION RAY(250)
      TYPE DOUBLE CP,FACT,RAY,SUM,TERM,TK,TM,XH,XI,Z2H
      COMMON FACT(300)
      L = 0
    1 IF (XH.GE..4D) GO TO 4
      Z2H=XH*XH/2.D
      DO 3 N=L,LJ
      TM=FACT(N+1)*(XH +XH)**N/FACT(N+N+2)
      IF(N.GT.84) TM=TM*1.D-300
      IF(TM.EQ.0.D) GO TO 8
      SUM=1.D
      TERM = 1.D
      DO 2 I=1,50
      XI=I*(N+N+I+I+1)
      TERM=-TERM*Z2H/XI
      SUM = SUM + TERM
    2 IF(DABS(TERM/SUM).LT.1.D-26) GO TO 3
    3 RAY(N+1) =TM*SUM
      RETURN'
    4 N=170
      IF(XH.LT.100.D) GO TO 20
      RAY(1)=DSIN(XH)/XH
      RAY(2)=(RAY(1)-DCOS(XH))/XH
      DO 11 K=1,LJ
   11 RAY(K+2)=(K+K+1)*RAY(K+1)/XH-RAY(K)
      RETURN
   20 IF(XH.GT.10.D) N=210
      RAY(N+1)=1.D-250
      RAY(N+2)=0.D
      I = -N
      M = -1
```

```
      DO 5 KK=1,M
      K = -KK
      TK=K+K+1
    5 RAY(K)=TK*RAY(K+1)/XH-RAY(K+2)
      CP=DSIN(XH)/(XH*RAY(1))
      IF(DSIN(XH).LT.1.D-2) CP=(DSIN(XH)/XH-DCOS(XH))/(XH*RAY(2))
      DO 6 L=1,LJ
    6 RAY(L)=CP*RAY(L)
    8 DO 9 J=N,LJ
    9 RAY(J+1)=0.D
      END
```

```
      SUBROUTINE SPHYN(X,N,ARR)
      DIMENSION ARR(250)
      TYPE DOUBLE X,ARR,TKP1
      ARR(1)=-DCOS(X)/X
      ARR(2)=ARR(1)/X-DSIN(X)/X
      DO 2 K=1,N
      TKP1=K+K+1
    2 ARR(K+2)=TKP1*ARR(K+1)/X-ARR(K)
      END
```

```
      SUBROUTINE EIGEN(A,VALU,N,N1,ANORM1,ANORM2)
      DIMENSION A(80,80),VALU(80),DIAG(80),Q(80),VALL(85)
      NN=N-2
      DO 160 I=1,NN
      II=I+2
      DO 160 J=II,N
      T1=A(I,I+1)
      T2=A(I,J)
      IF(T2.EQ.0.)  GO TO 160
      T=1./SQRT(T1*T1+T2*T2)
      SIN=T2*T
      COS=T1*T
      DO 105 K=1,N
      T2=COS*A(K,I+1)+SIN*A(K,J)
      A(K,J)=COS*A(K,J)-SIN*A(K,I+1)
  105 A(K,I+1)=T2
      DO 125 K=1,N
      T2=COS*A(I+1,K)+SIN*A(J,K)
      A(J,K)=COS*A(J,K)-SIN*A(I+1,K)
  125 A(I+1,K)=T2
  160 CONTINUE
      DO15I=1,N
      DIAG(I)=A(I,I)
      Q(I)=A(I,I-1)*A(I,I-1)
      VALL(I)=ANORM1
   15 VALU(I)=ANORM2
      I=1
      MATCH = N
   18 TAU=(VALL(I)+VALU(I))/2.
```

```
      IF (MATCH.NE.I-1) LATCH = MATCH
      MATCH = 0
      T0=0.
      T1=1.E-100
      DO20J=1,N
      T2=(DIAG(J)-TAU)*T1-Q(J)*T0
      IF((T1.NE.0.).AND.((T2*T1).LE.0.))  MATCH=MATCH+1
      T0=T1
   20 T1=T2
      DO25K=I,MATCH
   25 VALU(K)=TAU
      NATCH=MATCH+1
      DO 30 K=NATCH,LATCH
   30 IF(TAU.GT.VALL(K))       VALL(K)=TAU
   40 IF((VALU(I)-VALL(I)).GT.(1.E-4))  GO TO 18
      I=I+1
      MATCH = N
      IF(I.LE.N1) GO TO 40
      END
```

```
      SUBROUTINE QLEG(M,LN,X,Q)
      DIMENSION Q(200)
      COMMON FACT(300)
      TYPE DOUBLE BE,COEF,DI,DK,DM,DN,FACT,GA,Q,SUM,TERM,X,XA,YA,Z,ZA
      DM=M
      NN=0
      YA=DSQRT(X*X+1.D)
      ZA=(YA+X)*(YA+X)
      XA=0.25D*(YA+X)
      Z=2.D/(YA+X)
      LNM=LN+1
      DO 135 N=NN,LNM
      DN=N
      BE=DN+1.D
      GA=DN+1.5D
      IF(N.GT.84) GO TO 500
      COEF=Z*XA**(-N)*(FACT(N+1)  /FACT(N+N+2) *FACT(N+1))
      GO TO 510
  500 COEF=Z*XA**(-N)*(FACT(N+1)  *(1.D-300)*FACT(N+1)/FACT(N+N+2))
  510 SUM=TERM=1.D
      DK=-1.D
  130 DK=DK+1.D
      IF(DK.GT.5000.) GO TO 135
  131 TERM=-TERM*(DK+.5D)*(BE+DK)/((DK+1.D)*ZA*(GA+DK))
      SUM=SUM+TERM
      IF(ABS(TERM/SUM).GT.1.E-27)  GO TO 130
  135 Q(N+1)=-COEF*SUM
      DO 30 I=1,M
      DI=I
      DO 30 N=NN,LN
      DN=N
   30 Q(N+1)=-((DI+DN)*X*Q(N+1)+(DN-DI+2.D)*Q(N+2))/YA
      DO 40 N=NN,LN
      IF((2*((N+1)/4)).NE.((N+1)/2)) Q(N+1)=-Q(N+1)
   40 IF(2*(N/2).NE.N) Q(N+1)=-Q(N+1)
      END
```

```
1   1   1   0   1  50   3   1
1.D
1.D
1.D
1.D
```

UNCLASSIFIED

Appendix C

SAMPLE OUTPUT FROM OBRAD

OBLATE RADIAL FUNCTIONS

H = 2.0   X = 1.00   M = 0

L	R1	R1D	R2	R2D	EIGENVALUE	ACC
0	2.34717669462782575-001	-6.58288574406821127-001	3.09016403145128916-001	1.98442804122401613-001	-1.59449321318545821+000	24
1	3.67173553483780457-001	1.28819152193030221-002	-1.70976569147469831-001	6.74878383752388669-001	-5.05243980880919536-001	26
2	2.40771171637638706-001	1.84044667374789346-001	-4.52760905266076743-001	6.92241010849324981-001	4.09150910215069872+000	26
3	8.86364805684672965-002	1.54632440508265760-001	-8.05268012504849698-001	1.41566363147967906+000	1.00038638047302243+001	23
4	2.54281750428786965-002	6.43457698859617486-002	-1.88042365248769147+000	5.07321866994627864+000	1.80022284627129392+001	23
5	5.81601165104193378-003	1.92394122025173659-002	-6.20517850578578085+000	2.24580040023796754+001	2.80009228911501940+001	24
6	1.11378010125745539-003	4.52312681078985935-003	-2.63954595964100689+001	1.17267255967295161+002	4.00004582434479693+001	26
7	1.83475384685069244-004	8.81299329264931221-004	-1.35941275649262589+002	7.09604970576092168+002	5.40002531250156766+001	23
8	2.65366896979663664-005	1.46952160366230246-004	-8.18297982790462813+002	4.88943214403854239+003	7.00001512463476067+001	23
9	3.42197042554218326-006	2.14442736942438669-005	-5.62656091355813121+003	3.77978638391991264+004	8.80000959547799885+001	23
10	3.98210773769606290-007	2.78414966975615725-006	-4.34652881013081528+004	3.23914220767704478+005	1.08000063823823568+002	23
11	4.22267749456369435-008	3.25724677087945179-007	-3.72471541440938677+005	3.04728048981624905+006	1.30000044108474179+002	23
12	4.11328827105372811-009	3.46893278046847912-008	-3.50516558414518713+006	3.12178854435098925+007	1.54000031463047484+002	23
13	3.70547030988010438-010	3.39104944222828536-009	-3.59258268877841546+007	3.45904120264660398+008	1.80000023047882738+002	23
14	3.10486111862511741-011	3.06388486332246096-010	-3.98291666468379795+008	4.12153762448065954+009	2.08000017270791205+002	23
15	2.43183069655820260-012	2.57370790404271681-011	-4.74854338029237281+009	5.25473888317544871+010	2.38000013197627469+002	23
16	1.78808327742584407-013	2.02014683681596607-012	-6.05772228121030901+010	7.13753075205308716+011	2.70000010258742425+002	22
17	1.23093863527105232-014	1.48814782452726730-013	-8.23299786969724495+011	1.02895169381654110+013	3.04000008094993374+002	23
18	8.11665366791027173-016	1.03279709022009685-014	-1.18753896510113587+013	1.56901263060598560+014	3.40000006473282625+002	22
19	5.04277503549408506-017	6.77587178256325405-016	-1.81180495084034804+014	2.52310322561233479+015	3.78000005238405637+002	22
20	2.97917012573669646-018	4.21512905327820970-017	-2.91501388275478324+015	4.26724213641452590+016	4.18000004284653553+002	22
21	1.67767782061018806-019	2.49304888441621540-018	-4.93244051490987146+016	7.57188568674903722+017	4.60000003538556806+002	22
22	9.02539662701525016-021	1.40536190700136214-019	-8.75621006356095655+017	1.40651502106534583+019	5.04000002948115435+002	23
23	4.64773027375452579-022	7.56740734718397457-021	-1.62720807058269013+019	2.72955893394116691+020	5.50000002475915113+002	23
24	2.29525575627620571-023	3.90019103001484264-022	-3.15912911561567091+020	5.52391293473410259+021	5.98000002094630985+002	22
25	1.08885668262753784-024	1.92756039390586780-023	-6.39570035653564023+021	1.16378045923119012+023	6.48000001784033088+002	22
26	4.96977340821910846-026	9.15063900724499112-025	-1.34793683127172710+023	2.54850988407564918+024	7.00000001528956024+002	22
27	2.18551979719457311-027	4.17923056353396727-026	-2.95279579654320652+024	5.79248267412740524+025	7.54000001317899774+002	22
28	9.27269665473345848-029	1.83895734931473313-027	-6.71355333685200475+025	1.36465822428473411+027	8.10000001142050915+002	22
29	3.80041402717961348-030	7.80657816058199280-029	-1.58213941603152800+027	3.32829657436019115+028	8.68000000994588412+002	22
30	1.50637590835473614-031	3.20114981681520372-030	-3.85983805504402844+028	8.39370840109181065+029	9.28000000870184918+002	22
31	5.78075591697548589-033	1.26944347493769760-031	-9.73689962613566094+029	2.18648849475932263+031	9.90000000764644261+002	22
32	2.14993962011119050-034	4.87306435381676046-033	-2.53704250280939366+031	5.87705174217233482+032	1.05400000067463500+003	22
33	7.75662549441202662-036	1.81332499798913639-034	-6.82100210593831264+032	1.62845381397687316+034	1.12000000059749256+003	23
34	2.71715439756530950-037	6.54469222310974869-036	-1.89045789042302650+034	4.64733802309754159+035	1.18800000053107092+003	22
35	9.24950109918111579-039	2.29344012252003658-037	-5.39627205269731847+035	1.36482746765963931+037	1.25800000047363039+003	22
36	3.06218769068564281-040	7.80977767437188674-039	-1.58511502202048207+037	4.12143387817195032+038	1.33000000042375197+003	22
37	9.86695043983993688-042	2.58637116072524149-040	-4.78762971885347132+038	1.27875503620457704+040	1.40400000038027156+003	22
38	3.09658134291986096-043	8.33628977133375743-042	-1.48574670083086465+040	4.07364883338614273+041	1.48000000034222889+003	22
39	9.47163996973461536-045	2.61695310843173599-043	-4.73393692552290192+041	1.33150215674703549+043	1.55800000030882775+003	21
40	2.82547152340496883-046	8.00673730322825485-045	-1.54759995530369776+043	4.46253788188978129+044	1.63800000027940471+003	23
41	8.22519019850240727-048	2.38908806686838369-046	-5.18768280063806508+044	1.53262947385611789+046	1.72000000025340442+003	22
42	2.33799557802946587-049	6.95652487022501523-048	-1.78196939228790145+046	5.39080814480933645+047	1.80400000023035992+003	22
43	6.49270631940771031-051	1.97783477603665562-049	-6.26881713482143060+047	1.94084174552426810+049	1.89000000020987691+003	23
44	1.76247414796831296-052	5.49372802191887714-051	-2.25729213072288096+049	7.14850256516628045+050	1.97800000019162114+003	22
45	4.67902071252573897-054	1.49160802004801817-052	-8.31526420695059755+050	2.69220548359027393+052	2.06800000017530816+003	23
46	1.21543873321760019-055	3.96071213449269270-054	-3.13206000852870079+052	1.03623585245786392+054	2.16000000016069503+003	21
47	3.09071347447803081-057	1.02904371176655176-055	-1.20569948356182334+054	4.07440721554817776+055	2.25400000014757353+003	22
48	7.69706633860029849-059	2.61720823626680082-057	-4.74134225458350528+055	1.63580765014348177+057	2.35000000013576459+003	22
49	1.87809627222594917-060	6.51898712822089606-059	-1.90380937516333849+057	6.70311281423456589+058	2.44800000012511373+003	23

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)*

Naval Research Laboratory
Washington, D.C. 20390

2a. REPORT SECURITY CLASSIFICATION

Unclassified

2b. GROUP

3. REPORT TITLE

A FORTRAN COMPUTER PROGRAM FOR CALCULATING THE OBLATE SPHEROIDAL RADIAL FUNCTIONS OF THE FIRST AND SECOND KIND AND THEIR FIRST DERIVATIVES

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

An interim report on a continuing NRL Problem

5. AUTHOR(S) *(First name, middle initial, last name)*

A. L. Van Buren, R. V. Baier, and S. Hanish

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
January 20, 1970	34	11

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
NRL Problem S01-28	NRL Report 6959
b. PROJECT NO. RR 102-08-41-5225	
c.	9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*
d.	

10. DISTRIBUTION STATEMENT

This document has been approved for public release and sale; its distribution is unlimited

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	Department of the Navy (Office of Naval Research), Washington, D.C. 20360

13. ABSTRACT

The Helmholtz or scalar wave equation $(\nabla^2 + k^2)\,\Psi = 0$ is separable in oblate spheroidal coordinates $ih, -i\xi$, with solutions $\Psi = S\,(ih, \eta)\,R\,(ih, -i\xi)\,\Phi\,(\varphi)$. The subject of this report is a Fortran computer program called OBRAD which numerically evaluates the radial solutions $R\,(ih, -i\xi)$. The printed output from OBRAD consists of radial functions of the first and second kind, $R_{m\ell}^{(1),(2)}$ $(ih, -i\xi)$, their first derivatives $\partial R_{m\ell}^{(1),(2)}\,(ih, -i\xi)/\partial\xi$, the separation constants or eigenvalues $A_{m\ell}\,(ih)$, and an accuracy check. This report first describes the input data cards and the output format. The theory of the oblate spheroidal wave function is then discussed. A description of the principal internal features of OBRAD is then given. Finally a computer listing of OBRAD is attached as an appendix.

DD FORM 1473 (PAGE 1)
1 NOV 65

S/N 0101-807-6801

Security Classification

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Acoustic radiation						
Computer programming						
Mathematical models						
Analysis (mathematics)						
Wave equations						
Spheroidal wave functions						
Acoustic fields						
Sound transducers						
Sonar transducers						
Sonar arrays						
Submarines						

DD FORM 1 NOV 65 1473 (BACK)
(PAGE 2)

UNCLASSIFIED