Lh. 12 x 2

LA-6864-MS
Informal Report

UC-21

Issued: July 1977

Computer-Generated Interferograms to Characterize Microballoons

Barry S. Newberger
W. Stanley Hall

los alamos scientific laboratory of the University of California
LOS ALAMOS, NEW MEXICO 87545

An Affirmative Action/Equal Opportunity Employer

MASTER

UNITED STATES
ENERGY RESEARCH AND DEVELOPMENT ADMINISTRATION
CONTRACT W-7405-ENG. 36

Printed in the United States of America. Available from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road
Springfield, VA 22161
Price: Printed Copy $4.00 Microfiche $3.00

COMPUTER-GENERATED INTERFEROGRAMS TO CHARACTERIZE MICROBALLOONS

by

Barry S. Newberger
W. Stanley Hall

ABSTRACT

A code for the calculation and display of interferograms used for the characterization of laser fusion microballoons is described. Details of the analysis used in the algorithms are given and some unique features of the code are pointed out. The result is a code which is efficient and fairly versatile.

I. INTRODUCTION

One of the components of targets used in experiments to obtain thermonuclear fusion by laser-initiated implosions is a small hollow glass sphere typically called a microballoon. Prior to performing an experiment, one would like to be able to select the microballoons with respect to outer radius, wall thickness, uniformity, and freedom from nonspherical distortions based on specifications provided by a target designer. A method used widely to characterize these microballoons is interference microscopy.[1-3] Here light rays passing through a microballoon are beat with a reference ray to produce an interference pattern whose specific shape and fringe structure is a function of all of the parameters listed above. Once an interference pattern has been generated experimentally, one would like to be able to interpret the results, both qualitatively and quantitatively in terms of parameters of interest. Furthermore, one would like to be able to provide a technician who is selecting targets with a simple means of choosing targets which meet design criteria by simple visual inspection.

To provide the experimentalist with a means of obtaining such characterizations, we have developed a ray-tracing code, INTERF, to calculate interference patterns for a microballoon of given structure. The code is fully three-dimensional and does not require that the object be oriented relative to the optical axis of the system. Indeed, the code allows one to take arbitrary views through the microballoon as might be obtained by rolling it across the microscope stage. The output gives the interference pattern as viewed through the instrument.

In Sec. II we describe the ray transport equations and in Sec. III, the representation of the microballoons is discussed. A brief description of the code itself is given in Sec. IV and a summary comprises Sec. V. A listing of the code is in the Appendix.

II. RAY TRANSPORT EQUATIONS

A large part of the problem of the calculation of interferograms is solved, when the ray transport algorithms are constructed. Geometry is the bulk of the effort, optics being limited to little more than Snell's Law. Vector methods are clearly advantageous in three dimensions. We have used a parametric representation of the rays, which are, of course, straight line segments. The utility of this approach will be evident when we discuss the ray-surface intersections, particularly when dealing with the distorted spheroids. The equation for a ray is then

$$x = \frac{As}{(A^2+B^2+C^2)^{1/2}} + x_o \quad , \tag{1a}$$

$$y = \frac{Bs}{(A^2+B^2+C^2)^{1/2}} + y_o \quad , \tag{b}$$

and

$$z = \frac{Cs}{(A^2+B^2+C^2)^{1/2}} + z_o \quad , \tag{1c}$$

where

$$\frac{A}{(A^2+B^2+C^2)^{1/2}} , \quad \frac{B}{(A^2+B^2+C^2)^{1/2}} , \quad \frac{C}{(A^2+B^2+C^2)^{1/2}}$$

are the direction cosines of the ray (the dot product of the unit tangent to the ray with the cartesian coordinate directions), and (x_o,y_o,z_o) is the initial point on the ray; i.e. the point at which the arc length is zero.

In order to transport a ray as defined in Eq. (1), we must find its intersection with the surface it intercepts. We are concerned in general with distorted spherical surfaces; however, we defer the discussion of the intersection of rays with such surfaces until we discuss their representation in the next section. The two other surfaces of interest are segments of planes in three space and spheres.

A plane in three dimensions passing through the point (x',y',z') is defined by

$$a(x-x') + b(y-y') + c(z-z') = 0 \quad , \tag{2}$$

where, if the plane makes the angles α,β,γ with the x,y, and z axes, respectively,

$$a = \left(\frac{\cos^2\beta - \cos^2\alpha \sin^2\gamma}{1+\sin^2\gamma}\right)^{1/2} \quad , \tag{3a}$$

$$b = \left(\frac{\cos^2\alpha - \cos^2\beta \sin^2\gamma}{1+\sin^2\gamma}\right)^{1/2} \quad , \tag{3b}$$

and

$$c = \sin\gamma \quad . \tag{3c}$$

Alternatively, if the intercepts x_i, y_i, and z_i of plane and coordinate axes are specified,

$$a = \frac{z'}{[z'^2(x_i^2/y_i^2+1) + x_i^2(1-y'/y_i^2) - 2x_i x'(1-y'/y_i) + x'^2]^{1/2}} \quad , \tag{4a}$$

$$b = \frac{z'}{\left[z'^2(1+y_i^2/x_i^2)+y_i^{\,2}(1-y'/y_i) - \frac{2x'y_i^2}{x_i}(1-y'/y_i) + \frac{x'^2y_i^2}{z_i}\right]^{1/2}} \quad , \tag{4b}$$

and

$$c = \frac{(1-y'/y_i) - x'/x_i}{\left[z'^2(1/x_i^2 + 1/y_i^2) + (1-y'/y_i)^2 - \frac{2x'}{x_i}(1-y'/y_i) + x'^2/x_i^2\right]^{1/2}} \quad . \tag{4c}$$

Care must be exercised if the planes are parallel to one of the coordinate axes.
If $x_i = \infty$,

$a = 0$, $b/c = z'/(y_i - y')$, and

$$y'z' = (y_i - y')(z_i - z') \quad ; \tag{5a}$$

if $y_i = \infty$,

$b = 0$, $a/c = z'/(x_i - x')$, and

$$x'z' = (y_i - y')(z_i - z') \quad ; \tag{5b}$$

and if $z_i = \infty$,

$c = 0$, $a/b = y'/x_i - x'$,

and

$$x'y' = (x_i - x')(y_i - y) \quad . \tag{5c}$$

If the plane is parallel to two coordinate axes, we have $x_i \neq \infty$, $a = 1$, $b = c = 0$, and of course $x_i = x'$. Similarly for $y_i \neq \infty$ and $z_i \neq \infty$.
We can now find the point of intersection of the ray and plane. Substitution gives

$$s_p = \frac{(A^2+B^2+C^2)\,[a(x'-x_o)+b(y'-y_o)+c(z'-z_o)]}{aA+bB+cC} \quad , \tag{6}$$

where a, b, c are given by the Eq. (3), (4), or (5) as appropriate. And the intersection point P: (x_p, y_p, z_p) is given by

$$x_p = \frac{As_p}{(A^2+B^2+C^2)^{1/2}} + x_o \quad , \tag{7a}$$

$$y_p = \frac{Bs_p}{(A^2+B^2+C^2)^{1/2}} + y_o \quad , \tag{7b}$$

and

$$z_p = \frac{Cs_p}{(A^2+B^2+C^2)^{1/2}} + z_o \quad . \tag{7c}$$

Given the plane and the ray, Eqs. (6) and (7) are the desired result.

The intersection of a ray and a sphere of radius r_o can also be found in closed form. The parametric equation for a ray in spherical coordinates is

$$\begin{aligned} r(s) = \; & [(As + (A^2+B^2+C^2)^{1/2}\, x_o)^2 \\ & +(Bs + (A^2+B^2+C^2)^{1/2} y_o)^2 \\ & +(Cs + (A^2+B^2+C^2)^{1/2} z_o)^2]^{1/2} \cdot \frac{1}{(A^2+B^2+C^2)^{1/2}} \quad , \end{aligned} \tag{8a}$$

$$\theta(s) = \arccos \left[\frac{Cs+(A^2+B^2+C^2)^{1/2} z_o}{r(s)(A^2+B^2+C^2)^{1/2}} \right] \quad , \tag{8b}$$

and

$$\phi(s) = \arctan \left[\frac{Bs+(A^2+B^2+C^2)^{1/2} y_o}{As+(A^2+B^2+C^2)^{1/2} x_o} \right] \quad . \tag{8c}$$

At the point of intersection $r^2(s) = r_o^2$ and Eq. (8a) yields a quadratic equation for s_o, the arclength to the intersection. We find

$$s_o = - \frac{Ax_o+By_o+Cz_o}{(A^2+B^2+C^2)^{1/2}} \pm \left\{ \left(\frac{Ax_o+By_o+Cz_o}{A^2+B^2+C^2} \right)^2 + r_o^2 - x_o^2 - y_o^2 - z_o^2 \right\}^{1/2} \quad . \tag{9}$$

The lower sign gives the near intersection and the upper gives the diametrically opposed intersection. The intersection point (x_p, y_p, z_p) is found from Eq. (1). The polar coordinates of the intersection are

$$r = r_o \quad , \tag{10a}$$

$$\theta = \theta_p = \arccos\left[\frac{Cs_p + (A^2+B^2+C^2)^{1/2}}{r_o (A^2+B^2+C^2)^{1/2}}\right] \quad , \tag{10b}$$

and

$$\phi = \phi_p = \arctan\left[\frac{Bs_p + (A^2+B^2+C^2)^{1/2} y_o}{As_p + (A^2+B^2+C^2)^{1/2} x_o}\right] \quad . \tag{10c}$$

It now remains only to apply Snells law to find the A', B', C' of the refracted ray in terms of the A,B,C of the initial ray and the intersection point (x_p, y_p, z_p).

If $\hat{n}$ is the outward unit normal to a surface at a point, then Snell's law for refraction of a ray through that point is

$$\hat{n} \times (n_1 \vec{S}_1 - n_2 \vec{S}_2) = 0 \quad , \tag{11}$$

where n_1 and n_2 are the refractive indices of the media separated by the surface and $\vec{S}_1$ and $\vec{S}_2$ are the unit tangents to the rays. $\dot{S}_1$ is given and we need to find $\vec{S}_2$.

Crossing on the left with $\hat{n}$ yields

$$-\vec{S}_2 + \hat{n}(\hat{n} \cdot \vec{S}_2) = -\frac{n_1}{n_2} \vec{S}_1 + \frac{n_1}{n_2} \hat{n} (\hat{n} \cdot \vec{S}_1) \quad . \tag{12}$$

Dotting with $\dot{S}_2$ and using $|\vec{S}_2| = 1$ we get an equation for $\hat{n} \cdot S_2$ in terms of $\vec{S}_1 \cdot \vec{S}_2$:

$$(n \cdot \vec{S}_2)^2 = 1 - \frac{n_1}{n_2} \vec{S}_2 \cdot \vec{S}_1 + \frac{n_1}{n_2} (\hat{n} \cdot \vec{S}_1) (\hat{n} \cdot \vec{S}_2) \quad . \tag{13}$$

Now

$$(\hat{n} \times \vec{S}_1) \cdot (\hat{n} \times \vec{S}_2) = \vec{S}_1 \cdot \vec{S}_2 - (\hat{n} \cdot \vec{S}_2) (\hat{n} \cdot \vec{S}_1) \quad , \tag{14}$$

but

$$(\hat{n} \times \vec{S}_1) \cdot (\hat{n} \times \vec{S}_2) = n_1/n_2 \ |\hat{n} \times \vec{S}_1|^2 \quad ,$$

so

$$\vec{S}_1 \cdot \vec{S}_2 = n_1/n_2 \ |\hat{n} \times \vec{S}_1|^2 + (\hat{n} \cdot \vec{S}_2) (\hat{n} \cdot \vec{S}_1) \quad , \tag{15}$$

and thus

$$(\hat{n} \cdot \vec{S}_2)^2 + \frac{n_1}{n_2} (n \cdot \vec{S}_1)\ (n \cdot \vec{S}_2)$$

$$= 1 - \frac{n_1^{\ 2}}{n_2^{\ 2}} \left|\hat{n} \times \vec{S}_1\right|^2 + \frac{n_1}{n_2} (\hat{n} \cdot \vec{S}_1)(\hat{n} \cdot \vec{S}_2) \quad . \tag{16}$$

Dotting Eq. (12) with $\vec{S}_1$ using $|\vec{S}_1| = 1$, and Eq. (15) for $\vec{S}_1 \cdot \vec{S}_2$ shows $|\hat{n} \times \vec{S}_1|^2 = 1 - (\hat{n} \cdot S_1)^2$ so Eq. (16) becomes

$$(\hat{n} \cdot \vec{S}_2)^2 = 1 - \left(\frac{n_1}{n_2}\right)^2 (1 - (\hat{n} \cdot \vec{S}_1)^2) \tag{17a}$$

and

$$(\hat{n} \cdot \vec{S}_2) = \pm \left[1 - \left(\frac{n_1}{n_2}\right)^2 (1 - (\hat{n} \cdot \vec{S}_1)^2)\right]^{1/2} \quad . \tag{17b}$$

We now substitute into Eq. (12) for the desired expression for $\vec{S}_2$:

$$\vec{S}_2 = \frac{n_1}{n_2} \vec{S}_1 + \left\{ \pm \left[1 - \left(\frac{n_1}{n_2}\right)^2 (1 - (\hat{n} \cdot \vec{S}_1)^2)\right]^{1/2} - \frac{n_1}{n_2} (\hat{n} \cdot \vec{S}_1) \right\} \hat{n} \quad . \tag{18}$$

The positive bra~ch is appropriate for a ray leaving a surface and the negative appropriate for a ray entering a surface. If reflection occurs at the surface we have simply

$$\vec{S}_2 = \vec{S}_1 - 2(\hat{n} \cdot \vec{S}_1)\ \hat{n} \quad . \tag{18a}$$

The direction cosines A', B', C' are then

$$A' = \vec{S}_2 \cdot \hat{x} \quad , \tag{19a}$$

$$B' = \vec{S}_2 \cdot \hat{y} \quad , \tag{19b}$$

and

$$C' = \vec{S}_2 \cdot \hat{z} \quad . \tag{19c}$$

$A'^2 + B'^2 + C'^2 = 1$ since $\vec{S}_2$ is a unit vector.

We now have all the equations required to transport the rays and turn our attention to the description of the distorted surfaces.

III. NONSPHERICAL SURFACES

In characterizing glass microballoons for laser fusion target applications, we are interested in deviations of the walls from true spherical annuli, as well as determining wall thickness to a high degree of accuracy. Some analytic description of the distorted surfaces is desired keeping in mind that the microballoons are loaded into the optical instrument in an arbitrary way. That is, we cannot specify an orientation for a microballoon relative to the optical axis of the instrument (Fig. 1). Since the objects are prescreened,[4,5] however, we do know that we do not have to deal with severely distorted surfaces.

We have chosen to represent the surfaces by a "Fourier" expansion. In particular, since our surfaces are nominally spherical, an expansion in spherical harmonics is used. Furthermore, if δr_ℓ is the expansion coefficient of the ℓth term in the series and if r_o is the mean radius of the surface, we have $\delta r_\ell / r_o < 1$. We are thus led to a perturbation theory which lets us obtain the ray-surface intersections analytically. We thus avoid numerical inversions of transcendental equations.

The distorted surface is given in its local coordinate system by

$$r = r_o + \frac{1}{(2\pi)^{1/2}} \sum_{\ell=0}^{P} \delta r_\ell \, P_\ell^0 (\cos\theta) \quad , \tag{20}$$

P_ℓ^0 is the Legendre polynomial of order ℓ and we have taken the surface to be axisymmetric in its *local* coordinate system. In the laboratory system, the equation for the surface becomes

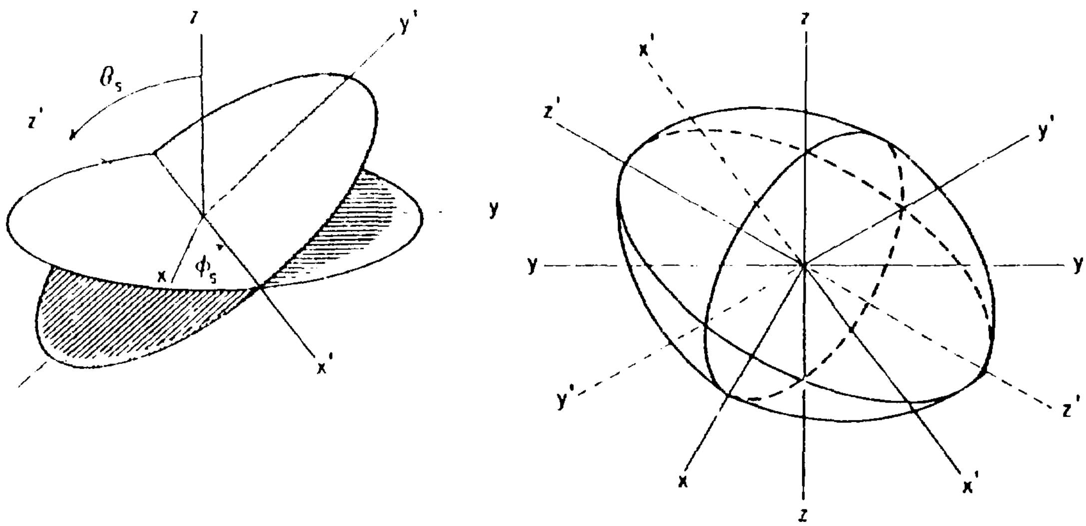


Fig. 1.
(a) Coordinate system. The angles θ_s and ϕ_s are the angles of the body system relative to the space system. The z-axis is the optical axis. (b) An ellipsoidal surface as it might appear relative to the space (unprimed) axis.

$$r = r_o + \sum_{\ell=0}^{P} \delta r_\ell \left(\frac{\ell+1/2}{2\pi}\right) \Big\{ P_\ell(\cos\theta)\, P_\ell(\cos\theta_s)$$

$$+ 2 \sum_{m=1}^{\ell} \frac{(\ell-m)!}{(\ell+m)!} P_\ell^m(\cos\theta)\, P_\ell^m(\cos\theta_s) \cos(m(\phi-\phi_s)) \Big\} \qquad (21)$$

by using the addition theorem for the spherical harmonics. We could relax the assumption of axisymmetry in the local frame by employing the more complicated rotation matrices familiar in quantum mechanics. The additional complexity was not warranted for our purposes. The analysis to be carried out to find the ray-surface intersections would proceed, in the more complicated case, in a similar fashion.

To find the ray-surface intersection, we simply substitute Eq. (20) into the parametric equation for a ray in spherical coordinates, Eq. (8), where we write $s = s_o + \delta s$. We want to find an expression for δs in terms of the given quantities r_o, δr defining the ray. The intersection of the ray with the sphere $r = r_o$ defines s_o. The parametric equation for a ray is

$$\left(\frac{r(s)}{r_o}\right)^2 = 1 + \frac{2\delta s\left((A^2+B^2+C^2)s_o + (A^2+B^2+C^2)^{1/2}(Ax_o+By_o+Cz_o)\right)}{(A^2+B^2+C^2)\, r_o^2}$$

$$+ \left(\frac{\delta s}{r_o}\right)^2 \quad . \qquad (22)$$

This equation is a transcendental equation for δs since the left hand side, $r(s)$, is given by Eq. (21) and θ and ϕ, the polar coordinates of the intersection point, are implicit functions of s. The solution of this equation is simplified because we can treat the transcendental part of Eq. (22) perturbatively. As $\delta r_\ell / r_o$ is a small coefficient, we can approximate the polar coordinates θ, ϕ by the values given by the intersection of the ray with the sphere $r = r_o$. In this case Eq. (22) simply reduces to a quadratic for δs. It is now a simple matter to iterate until δs has the desired accuracy. If a subscript is used to denote N-th iterant, we have

$$\delta s_N = \frac{-r_o^2 \sigma}{2} \mp \frac{r_o^2}{2} \Bigg[\sigma^2 + \frac{\delta}{r_o} \sum_{\ell m} \delta r_\ell\, Y_\ell^m \Big|_{s=s_o+\delta s_{N-1}}$$

$$+ \frac{4}{r_o^2} \left(\sum_{\ell m} \delta r_\ell\, Y_\ell^m \Big|_{s=s_o+\delta s_{N-2}} \right)^2 \Bigg] \qquad (23)$$

where

$$\sigma = \frac{2}{r_o^2} \left(s_o + \frac{Ax_o+By_o+Cz_o}{(A^2+B^2+C^2)} \right) \quad .$$

In general, a ray can intersect the distorted surface in two points. The near intersection is given by the upper sign and the far by the lower. If the radicand is negative, the ray misses the sphere and the degenerate case of a ray tangent to the sphere occurs when the radicand is equal to zero.

At this point, we have all the equations necessary to do the ray transport. A brief description of the code itself will now be given. Figures 2-6 are simple interferograms. The parameters used are shown.

IV. PROGRAM DESCRIPTION

INTERF was written for the CDC-7600 and is executed under the Livermore Time Sharing System at a Tektronix 4015 Computer Display Terminal. The programming language used is LRLTRAN which is quite similar to FORTRAN IV. LRLTRAN has many unique features, but only a few are used in INTERF.

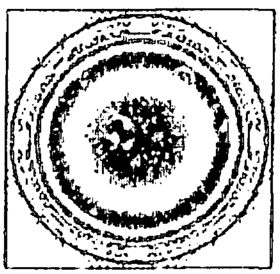
INTERF does the calculation in a fairly straightforward manner. First, all the data are read in using a NAMELIST read statement. There are 34 variables in the NAMELIST statement, but due to extensive use of defaults, one can get by with less than six for a simple problem. Then, the array VIEW is initialized to 0.5. The optical path length for the reference ray is calculated. A ray is then started from each point of a 50 by 50 grid from each of the specified quadrants.

One unique feature used in INTERF is the ability to use the following statement:

DIMENSION ARRAY ((i,j))

where i and j are integer constants ($i \neq 1$) specifying the subscript range of ARRAY.

```
» 10/12/76      09:26:25

e'l= 36.00      d=48.00      p= 2          lambda=0.5000    phasdif=0.
test=0.050      ns=1.0003    nv=1.0003

bod= 0.         cod= 0.      rod=48.09     phisod=  0.      thetsod=  0.
xso= 0.         yso= 0.      zso=48.00

bid= 0.         cid= 0.      rid=46.00     phisid=  0.      thetsid=  0.
xsi= 0.         ysi= 0.      zsi=48.00     ninner=1.5000

plot bkgd. pts.?
n
```



```
outline?
y
make film file?
n
give namelist input
```

Fig. 2.

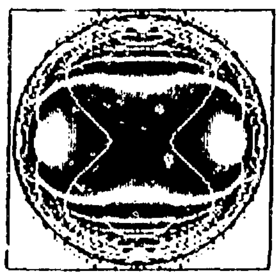
Pattern produced by a shell formed from two concentric spheres. Please refer to the text for the definitions of the symbols.

```
r 11/08/76      12:57:25

ell= 96.00      d=48.00        p= 2        lambda=0.5000    phasdif=0.
test=0.050      na=1.0003      nv=1.0003

bod=47.00     cod=48.00     rod=48.00     phisod=   0.      thetsod= 90.00
xso= 0.       yso= 0.       zso=48.00

bid=45.00     cid=46.00     rid=46.00     phisid=   0.      thetsid=   0.
xsi= 0.       ysi= 0.       zsi=48.00     ninner=1.5000

plot bkgd. pts.?
n
```

```
outline?
y
make film file?
n
give namelist input
```

(a)

```
s 11/07/76      15:54:14

ell= 96.00      d=48.00        p= 2        lambda=0.5000    phasdif=0.
test=0.050      na=1.0003      nv=1.0003

bod=47.00     cod=48.00     rod=48.00     phisod=   0.      thetsod=   0.
xso= 0.       yso= 0.       zso=48.00

bid=45.00     cid=46.00     rid=46.00     phisid=   0.      thetsid= 90.00
xsi= 0.       ysi= 0.       zsi=48.00     ninner=1.5000

plot bkgd. pts.?
n
```



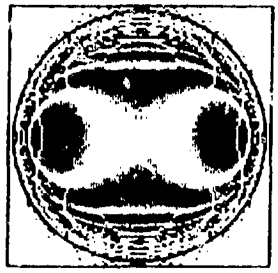
```
outline?
y
make film file?
n
give namelist input
```

(b)

Fig. 3(a)-(e).

Pattern produced by a shell formed from two ellipsoidal surfaces. In (e), the surfaces are not concentric.

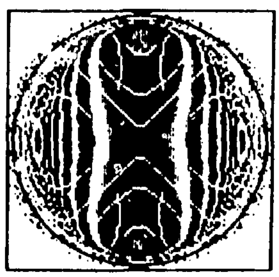
```
s 11/07/76      16:10:24

ell= 96.00      d=48.00        p= 2       lambda=0.5000    phasdif=0.5000
test=0.050      na=1.0003      nv=1.0003

bod=47.00     cod=48.00     rod=48.00     phisod=   0.      thetsod=   0.
xso= 0.       yso= 0.       zso=48.00

bid=45.00     cid=46.00     rid=46.00     phisid=   0.      thetsid= 90.00
xsi= 0.       ysi= 0.       zsi=48.00     ninner=1.5000

plot bkgd. pts.?
n
```



```
outline?
y
make film file?
n
give namelist input
```

(c)

```
s 11/07/76      15:08:03

ell= 96.00      d=48.00        p= 2       lambda=0.5000    phasdif=0.
test=0.050      na=1.0603      nv=1.0003

bod=47.00     cod=48.00     rod=48.00     phisod=   0.      thetsod= 90.00
xso= 0.       yso= 0.       zso=48.00

bid=45.00     cid=46.00     rid=46.00     phisid= 90.00     thetsid= 90.00
xsi= 0.       ysi= 0.       zsi=48.00     ninner=1.5000

plot bkgd. pts.?
n
```

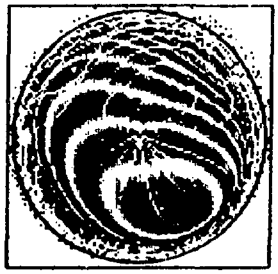


```
outline?
y
make film file?
n
give namelist input
```

(d)

Fig. 3 (cont)

```
r 11/08/76      13:43:15

ell= 96.00      d=48.00         p= 2            lambda=0.5000     phasdif=0.
test=0.050      na=1.0003       nv=1.0003

bod=47.00       cod=48.00       rod=48.00       phisod= 45.00     thetsod= 45.00
xso= 0.         yso= 0.         zso=48.00

bid=45.00       cid=46.00       rid=46.00       phisid=-45.00     thetsid=135.00
xsi= 0.         ysi= 0.         zsi=48.00       ninner=1.5000

plot bkgd. pts.?
n
```



```
outline?
y
make film file?
n
give namelist input
```

(e)

Fig. 3 (cont)

Another unique feature is the ability to use the following statement

DO N i = m_1, m_2, m_3

where m_1 may be 0 or negative. Also, if m_1 initially is greater than m_2, control is transferred to the statement following statement N.

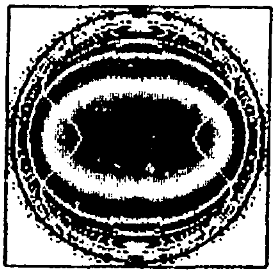
A point of intersection with the outer ellipsoidal surface is determined for each ray leaving the microscope stage. The optical path length that far is calculated using the index of refraction for air. Next, the new direction cosines are calculated so that the point of intersection with the first inner ellipsoidal surface can be determined along with the distance from the last intersection. The optical path length is again calculated using the index of refraction for the microballoon and this is added to the previous optical path length. This process is repeated until the innermost surface is reached. The ray is then followed back to the outer ellipsoidal surface adding optical path lengths at each intersection. At this point, the distance to and the point of intersection with the reference plane is determined. Finally the rays are projected down to the focal plane. At this point,

$$0.5 \times \text{cosine}\ [(\text{OPLRI}-\text{OPLRR}) \times 2\pi/\text{LAMBDA}] \qquad (24)$$

is added to the proper element of the array VIEW. OPLRR is the optical path length for the reference ray and OPLRI is the total optical path length for the current ray.

The above is of course quite simplified. For a ray leaving a starting point or last intersection, there may not be another intersection or the ray may be internally reflected. These and other eventualities are provided for as will be seen from the listing of the FORTRAN in the Appendix. A version is also available which displays a selected quadrant magnified by a factor four.

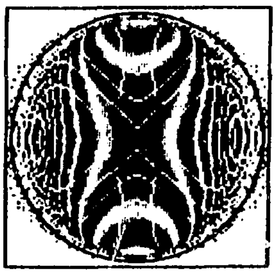
```
r 11/05/76      15:44:52

ell= 96.00     d=48.00        p= 2       lambda=0.5000    phasdif=0.
test=0.050     na=1.0003     nv=1.0003

bod=47.00    cod=48.00    rod=48.00    phisod=   0.      thetsod= 90.00
xso= 0.      yso= 0.      zso=48.00

bid=45.00    cid=46.50    rid=46.50    phisid=   0.      thetsid= 90.00
xsi= 0.      ysi= 0.      zsi=48.00    ninner=1.5000

plot bkgd. pts.?
n
```



```
outline?
y
make film file?
n
give namelist input
```

(a)

```
r 11/05/76      15:36:06

ell= 96.00     d=48.00        p= 2       lambda=0.5000    phasdif=0.
test=0.050     na=1.0003     nv=1.0003

bod=47.00    cod=48.00    rod=48.00    phisod=   0.      thetsod= 90.00
xso= 0.      yso= 0.      zso=48.00

bid=45.00    cid=46.50    rid=46.50    phisid= 90.00     thetsid= 90.00
xsi= 0.      ysi= 0.      zsi=48.00    ninner=1.5000

plot bkgd. pts.?
n
```



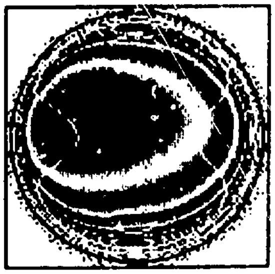
```
outline?
y
make film file?
n
give namelist input
```

(b)

Fig. 4.

Pattern produced by a shell formed from two concentric ellipsoids. This case is similar to Fig. 3 with the major radius of the inner ellipsoid increased by 0.5 μm. Figures 4(b) and 3(d) are comparable.

```
r 11/05/76      15:58:49

ell= 96.00     d=48.00      p= 2      lambda=0.5000    phasdif=0.
test=0.050     na=1.0003    nv=1.0003

bod=47.00    cod=43.00    rod=48.00    phisod=  0.      thetsod= 90.00
xso= 0.      yso= 0.      zso=48.00

bid=45.00    cid=46.50    rid=46.50    phisid=  0.      thetsid= 90.00
xsi=-0.25    ysi= 0.25    zsi=48.00    ninner=1.5000

plot bkgd. pts.?
n
```



```
outline?
y
make film file?
n
give namelist input
```

(c)

Fig. 4 (cont)

Description of Selected Variables Used in INTERF

A - defined by Eqs. (2), (3a), or (4a). (Namelist variable, default = 0.0.)

B - defined by Eqs. (2), (3b), or (4b). (Namelist variable, default = 0.0.)

BID - is B in the equation for an ellipsoid,

$$\frac{X^2}{A^2} + \frac{Y^2}{B^2} + \frac{Z^2}{C^2} = 1 . \tag{25}$$

BID pertains to one of the inner ellipsoids. (Namelist variable, default = 0.0.)

BOD - is also B in Eq. (25) and pertains to the outer ellipsoid. (Namelist variable, default = 0.0.)

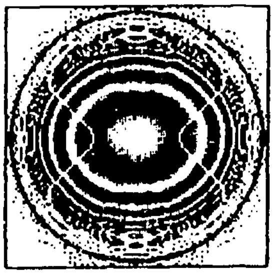
C - defined by Eqs. (2), (3c), or (4c). (Namelist variable, default = 0.0.)

CID - is C in Eq. (25) and pertains to one of the inner ellipsoids. (Namelist variable, default = 0.0.)

COD - id C in Eq. (25) and pertains to the outer ellipsoid. (Namelist variable, default = 0.0.)

D - is the distance from the focal plane to the reference plane. (Namelist variable, default = ROD).

```
s 11/07/76      17:14:36

ell= 96.00    d=48.00       p= 2          lambda=0.5000    phasdif=0.
test=0.050    na=1.0003     nv=1.0003

bod=47.00     cod=48.00     rod=48.00     phisod= 90.00    thetsod= 90.00
xso= 0.       yso= 0.       zso=48.00

bid=45.00     cid=46.00     rid=46.00     phisid=  0.      thetsid= 90.00
xsi= 0.       ysi= 0.       zsi=48.00     ninner=1.5000

for the next inner ellipsoid
bid= 0.       cid= 0.       rid=44.00     phisid=  0.      thetsid=  0.
xsi= 0.       ysi= 0.       zsi=48.00     ninner=1.6500

plot bkgd. pts.?
n
```



```
outline?
y
make film file?
n
give namelist input
```

Fig. 5.
Pattern produced by two shells formed from two concentric ellipsoids and a sphere.

DCA - is a direction cosine of a ray. See Eq. (1a).

DCAA - is the starting value of DCA. (Namelist variable, default = 0.0.)

DCB - is a direction cosine of a ray. See Eq. (1b).

DCBB - is the starting value of DCB. (Namelist variable, default = 0.)

DCC - is a direction cosine of a ray. See Eq. (1c).

DCCC - is the starting value of DCC. (Namelist variable, default = 1.0.)

DELSIT - is the δs_N of Eq. (23) (used only in subroutine PTTOOID).

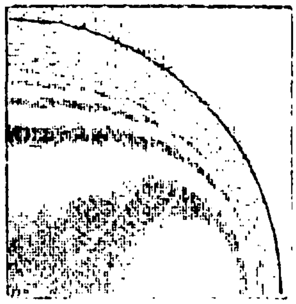
ELL - is the distance from the microscope stage to the reference plane. (Namelist variable, default = 2 x ROD.)

EPSID - is the eccentricity of a inner ellipsoid squared.

EPSOD - is the eccentricity of the outer ellipsoid squared.

END - is the variable used to end the run. (Namelist integer variable, default = 0.)

```
s 05/11/77     11 46 4-

ell= 96.00    d=48.00      p= 2      lambda=0.5000   phasdif=0.
test=0.050    na=1.0003    nv=1.0003

bod=47.00   cod=48.00   rod=48.00   phisod=  0.     thetsod=  0.
xso= 0.     yso= 0.     zso=48.00

bid=45.00   cid=46.00   rid=46.00   phisid=  0.     thetsid= 90.00
xsi= 0.     ysi= 0.     zsi=48.00   ninner=1.5000

plot bkgd. pts.?
no

outline?
yes
make film file?
no
```



(a)

```
r 05/11/77     14:20:42

ell= 96.00    d=48.00      p= 2      lambda=0.5000   phasdif=0.
test=0.050    na=1.0003    nv=1.0003

bod=47.00   cod=48.00   rod=48.00   phisod=  0.     thetsod= 90.00
xso= 0.     yso= 0.     zso=48.00

bid=45.00   cid=46.50   rid=46.50   phisid= 90.00   thetsid= 90.00
xsi= 0.     ysi= 0.     zsi=48.00   ninner=1.5000

plot bkgd. pts.?
yes

outline?
yes
make film file?
no
```

(b)

Fig. 6.
An expanded scale interferogram, for an object as in Figs. 3(c) and 4(b), respectively.

IC - controls the intensity of each point that is plotted to produce the desired interferogram. Minimum intensity occurs with IC = 68 and maximum intensity occurs with IC = 125.

LAMBDA - is the wavelength of light which produces the interference pattern. (Namelist real variable, default = 0.5.)

MAXPASS - gives the maximum number of iterations in subroutine PTTOOID for any ray.

NA - is the index of refraction for air. (Namelist real variable, default = 1.0003.)

NINNER - is an array containing the indices of refraction for each inner surface. [Namelist real variable, default for NINNER(1) = 1.5.]

NISURF - is the number of inner surfaces. (Namelist variable, default = 1, maximum = 6.)

NPASS - is the limit on the number of iterations in subroutine PTTOOID. See text pertaining to Eq. (23). Current limit is 20.

NQUAD - is the number of quadrants in which rays are to be traced. To save time, one may specify less than four quadrants. (Namelist variable, default = 4.)

NV - is the index of refraction for the microballoon void. (Namelist real variable, default = 1.0003.)

OPL - is the optical path length of the current ray.

P - is the number of terms in the Legendre summation. (Namelist integer variable, default = 2.) The program has not been tested for P > 2.

PHASDIF - is the phase of the reference rays in units of the wavelengths of the light producing the pattern.

PHI - is one of the parameters in spherical coordinates specifying a point of intersection.

PHISID - is the angle between the major axis of an inner ellipsoid and the Y-axis. (Namelist variable, default = 0.)

PHISOD - is the angle between the major axis of the outer ellipsoid and the Y-axis. (Namelist variable, default = 0.)

PLMTHET - is the value returned by function PLM when called with cosine of THET as an argument.

PLMTHTS - is the value returned by function PLM when called with cosine of THETSID (or THETSOD) as an argument.

R - is one of the parameters in spherical coordinates specifying a point of intersection.

RID - is the diameter of an inner spherical surface. Input only when EPSID = 0. (Namelist variable; default = CID, if EPSID ≠ 0.0.)

ROD - is the diameter of the outer spherical surface. (Namelist variable; default = COD, if EPSOD ≠ 0.)

S - is the distance a ray travels from one intersection (or starting point) to the next intersection.

SGN - is used in subroutine PTTOOID and gives the appropriate sign as required in Eq. (23).

SIGN - is used in subroutines DIRCOS and PTTOOID and gives the appropriate sign as required in Eq. (9) and Eq. (17b).

TEST - is the convergence criterion discussed in connection with Eq. (23). (Namelist variable, default = 0.1 x LAMBDA.)

THET - is one of the parameters in spherical coordinates specifying a point of intersection.

THETSID - is the angle between the major axis of an inner ellipsoid and the X-axis. (Namelist variable, default = 0.)

THETSOD - is the angle between the major axis of the outer ellipsoid and the Y-axis. (Namelist variable, default = 0.)

VIEW - is the array (100 x 100) containing the information which is used to produce the interference pattern on the Tektronix 4015 screen. Each element of VIEW produces a dot whose intensity is varied in proportion to the value of the element. (Fifty-eight different intensities are possible.)

XO - is one of the parameters in Cartesian coordinates specifying a point of intersection (or the starting point).

XSI - is the X-coordinate of the center of an inner ellipsoidal surface. (Namelist variable, default = 0.)

XSO - is the X-coordinate of the center of the outer ellipsoidal surface. (Namelist variable, default = 0.)

YO - is one of the parameters in Cartesian coordinates specifying a point of intersection (or the starting point).

YSI - is the Y-coordinate of the center of an inner ellipsoidal surface. (Namelist variable, default = 0.)

YSO - is the Y-coordinate of the center of the outer ellipsoidal surface. (Namelist variable, default = 0.)

ZO - is one of the parameters in Cartesian coordinates specifying a point of intersection (or the starting point).

ZSI - is the Z-coordinate of the center of an inner ellipsoidal surface. (Namelist variable, default = ZSO.)

ZSO - is the Z-coordinate of the center of the outer ellipsoidal surface. (Namelist variable, default = ROD.)

Description of Subroutines Used in INTERF

ABCSPHR - a short routine to calculate A, B, and C.

CCFSC - another short routine to calculate Cartesian coordinates when given spherical coordinates.

DIRCOS - is a routine the purpose of which is to calculate the new direction cosines given the old direction cosines, A, B, C, and the indices of refraction.

FACTORL - a short function to calculate factorials.

PLM - is a routine whose purpose is to compute normalized associated Legendre polynomials by recursion. [The routine obtained was written by R. Nerf and modified by S. Green. The version used was then modified again by S. Hall.]

PTTOOID - has a purpose which is to calculate the point of intersection of a ray with an ellipsoid and to calculate the distance from the starting point to the ellipsoid.

PTTOPLN - has a purpose which is to calculate the point of intersection of a ray with a plane and the distance from the starting point to the plane.

ANMODE
BELL
DRAWA
DRWABS
DWINDO
FINITT
MOVABS
MOVEA
NEWPAG
PTMOD
START
TERM
TOUTPT
TSEND
TWINDO
XYCNVT

These are all special routines that pertain to the Tektronix PLOT-10 system. For further information see Documents 062-1474-00 and 062-1530-00 published by the Information Display Division of Tektronix, Inc., P.O. Box 500, Beaverton, Oregon 97005.

ACKNOWLEDGMENTS

We wish to thank R. Jay Fries, Eugene H. Farnum, David M. Stupin, David R. Kohler, and John R. Miller for many useful discussions. The Legendre Function subroutine was provided by R. T. Pack.

REFERENCES

1. F. Skoberne, ed., "Laser Fusion Program at LASL, July 1—December 31, 1975," Los Alamos Scientific Laboratory report LA-6245-PR (July 1976) p. 77.

2. R. R. Stone, D. W. Gregg, and P. C. Souers, "Nondestructive Inspection of Transparent Microtargets for Laser Fusion," J. Appl. Phys., **46**, 2693 (1975).

3. R. R. Stone, P. C. Souers, G. C. Abell, and C. W. Reed, "Light Interference in Hollow Glass Microsphere Laser Targets," Lawrence Livermore Laboratory report UCRL-77487 (1975).

4. R. Jay Fries and Eugene H. Farnum, "Laser Fusion Target Fabrication," Los Alamos Scientific Laboratory report, LA-5703-SR Rev. (April 1974).

5. E. H. Farnum and R. J. Fries, *External Pressurization as a Means of Upgrading Glass Microballoons,* Bull. Am. Phys. Soc., Series 11, **20**, 1247 (1975).

APPENDIX

PROGRAM INTERF

LASL Identification No. LP-0769

```
$CHAT (TEKTLIB TEKLIB ORDERLIB SYMLIB) INTERF1  RUN1 RK LIST1 H77 M M
      PROGRAM INTERF(INPUT,OUTPUT,GRAYPLT,TAPE3=GRAYPLT)
C
C     PLEASE REPORT ERRORS TO
C     W. STANLEY HALL                       BARRY S. NEWBERGER
C     MS-532                          OR    MS-531
C     BOX 1663                              BOX 1663
C     LOS ALAMOS, N.M. 87545                LOS ALAMOS, N.M. 87545
C     PH 505 667 7381                       PH 505 667 5627
C
      INTEGER END, OPTION, P
      REAL LAMBDA, NA, NINNER, NV
      DIMENSION BID(6), CID(6), EPSID(6), ICT(10),
     1          NINNER(7), NTR(13), PHISID(6), RID(6),
     2          THETSID(6), XST(6), YST(6), ZST(6)
      COMMON A, B, C, DCA, DCB, DCC, R, THET, PHI, S,
     1       XO, YO, ZO, XP, YP, ZP, IFLAG, VIEW(100,100),
     2       DELS, T2, P, OPLINC, TEST, IER(9),
     3       IRFLAG, ITFLAG, MAXPASS, MAXTERM
      COMMON /LEGEND/PLMTHET(0,61,8), PLMTHIS(0,81,8)
      COMMON /CONST/COSTHET, COSTHIS, OOSQR2P, PHID, TERMS((0,2))
      DATA ((ICT(I),I=1,10)=125,119,112,106,100,93,87,81,74,68)
      DATA (RAD=57.2957795)
      NAMELIST /INPT1/A, B, C, BOD,
     1          XSO, YSO, ZSO, RID, XST, YST, ZST,
     2          NA, NINNER, NV, D, FILL, NTSURF, NQUAD,
     3          END, DCAA, DCBB, DCCC, LAMBDA, PHASDIF,
     4          BID, BOD, CID, COD, P, PHISID, PHISOD,
     5          THETSID, THETSOD, TEST
      CALL CHANGE(5H*RUN1)
      A=0.
      B=0.
      C=1.
      DCAA=0.
      DCBB=0.
      DCCC=1.
      D=0.0
      FILL=0.0
      LAMBDA=0.5
      NA=1.3003
      NINNER(1)=1.5
      NV=1.0003
      NTSURF=1
      NQUAD=4
      P=2
      PHASDIF=0.0
      TEST=0.0
      XO=0.
      YO=0.
      BOD=COD=0.0
      PHISOD=THETSOD=0.0
      DO 10 I=1, 6
      BID(I)=0.0
      CID(I)=0.0
      PHISID(I)=0.0
   10 THETSID(I)=0.0
      END=0
      XSO=0.
      YSO=0.
      ZSO=0.
```

```
      XSI=0.
      YSI=0.
      ZSI=0.
C
C     D   IS THE DIST. FROM THE FOCAL PLANE TO THE REF. PLANE.
C     ELL IS THE DIST. FROM THE MICR. STAGE TO THE REF. PLANE.
C     LAMBDA IS THE WAVELENGTH OF LIGHT PRODUCING PATTERN.
C     NA IS THE INDEX OF REFRACTION FOR AIR.
C     NV IS THE INDEX OF REFRACTION FOR MICROBALLOON VOID.
C     NINNER IS AN ARRAY CONTAINING THE INDICES OF REFRACTION FOR
C             EACH INNER SURFACE. (NINNER(1) FOR OUTERMOST MATERIAL)
C     NISURF IS THE NUMBER OF INNER SURFACES. (DEFAULT=1, MAX=6)
C     NQUAD IS THE # OF QUADRANTS IN WHICH RAYS ARE TO BE TRACED.
C     PHASDIF IS PHASE OF THE REF. RAYS IN UNITS OF THE WAVELENGTH
C              OF THE LIGHT PRODUCING THE PATTERN. (0. TO 1.)
C     RID IS DIAM. OF INNER SPHERICAL SURFACE.
C     ROD IS DIAM. OF OUTER SPHERICAL SURFACE.
C     XSI IS X-COORD. OF CENTER OF THE INNER ELLIPSOIDAL SURFACE.
C     XSO IS X-COORD. OF CENTER OF THE OUTER ELLIPSOIDAL SURFACE.
C     YSI IS Y-COORD. OF CENTER OF THE INNER ELLIPSOIDAL SURFACE.
C     YSO IS Y-COORD. OF CENTER OF THE OUTER ELLIPSOIDAL SURFACE.
C     ZSI IS Z-COORD. OF CENTER OF INNER ELLIPSOIDAL SURFACE.
C     ZSO IS Z-COORD. OF CENTER OF OUTER ELLIPSOIDAL SURFACE.
C
C                                    X**2    Y**2    Z**2
C      EQUATION FOR ELLIPSOID        ----  + ---- + ---- = 1.
C                                    A**2    B**2    C**2
C
C     BOD IS A CONST. IN EQ. FOR OUTER ELLIPSE (ASSUME AOD=BOD)
C     COD IS A CONST. IN EQ. FOR OUTER ELLIPSE
C     BID IS A CONST. IN EQ. FOR INNER ELLIPSE (ASSUME AID=BID)
C     CID IS A CONST. IN EQ. FOR INNER ELLIPSE
C   THETSOD IS ANGLE BETWEEN MAJOR AXIS OF OUTER ELLIPSE AND X-AXIS
C   THETSID IS ANGLE BETWEEN MAJOR AXIS OF INNER ELLIPSE AND X-AXIS
C    PHISOD IS ANGLE BETWEEN MAJOR AXIS OF OUTER ELLIPSE AND Y-AXIS
C    PHISID IS ANGLE BETWEEN MAJOR AXIS OF INNER ELLIPSE AND Y-AXIS
C    THE FOUR INPUT ANGLES DESCRIBED ABOVE TO BE GIVEN IN DEGREES.
C     EPSOD IS OUTER ELLIPSOID ECCENTRICITY SQUARED
C     EPSID IS INNER ELLIPSOID ECCENTRICITY SQUARED
C         P IS THE NUMBER OF TERMS IN LEGENDRE SUMMATION
C    IF P>6, INCREASE DIMENSIONS OF /LEGEND/ IN MAIN, DIRCOS, AND PTTODID.
C
      CALL INITT(960)
      CALL SETBUF(1)
      CALL TERM(3,4096)
      CALL CHRSIZ(1)
C
      TER(1)=0.5
      DO 20 I=2, 9
   20 TER(I)=(1.5-FLOAT(I))*TER(I-1)/FLOAT(I)
C     TER=.5, -1/8, 3/48, -15/384, ETC. (USED IN PTTODID)
   30 PRINT 330
      INPUT DATA INPT1, 59, 0
      IF (END.EQ.1) GO TO 500
       PHISOD= PHISOD/RAD
      THETSOD=THETSOD/RAD
      EPSOD=0.0
      IF (BOD.GT.0.0) EPSOD=1.0-COD**2/BOD**2
      IF (EPSOD.EQ.0.0.AND.BOD.GT.0.0) PRINT 453
      IF (EPSOD.EQ.0.0.AND.BOD.GT.0.0) GO TO 30
```

```
      IF (EPSOD.NE.0.) ROD=COD
      IF (ELL.EQ.0.0) ELL=ROD*2.0
      IF (D  .EQ.0.0) D  =ROD
      IF (ZSO.EQ.0.0) ZSO=ROD
      NINNER(NISURF+1)=NV
      DO 40 IS=1, NISURF
       PHISID(IS)= PHISID(IS)/RAD
      THETSID(IS)=THETSID(IS)/RAD
      EPSID(IS)=0.0
      IF (RID(IS).GT.0.0) EPSID(IS)=1.0-CID(IS)**2/RID(IS)**2
      IF (EPSID(IS).EQ.0.0.AND.RID(IS).GT.0.0) PRINT 460, IS, IS
      IF (EPSID(IS).EQ.0.0.AND.RID(IS).GT.0.0) GO TO 30
      IF (EPSID(IS).NE.0.0) RID(IS)=CID(IS)
      IF (ZSI(IS).EQ.0.0) ZSI(IS)=ZSO
   40 CONTINUE
      TERM1=6.2831853072/LAMBDA
      IF (TEST.EQ.0.0) TEST=0.1*LAMBDA
      ZP=ELL
      X0=Y0=Z0=0.
      MAXPASS=0
      DO 50 I=1, 10000
   50 VIEW(I)=0.5
      DCA=DCAA
      DCB=DCBB
      DCC=DCCC
      CALL PITOPEN
      REFRAY=NA*(S+PHASDIF*LAMBDA)
C
      DO 120 IQUAD=1, NQUAD
      IXO=IYO=1
      IF (IQUAD.EQ.2.OR.IQUAD.EQ.3) IXO=-1
      IF (IQUAD.GT.2) IYO=-1
      ISR=2*NISURF+1
      DO 60 IS=1, ISR
   60 NIR(IS)=0
      DO 110 IXX=1, 53
      IF (IQUAD.EQ.1.OR.IQUAD.EQ.4) X0=0.5+IXO*(IXX-1)
      IF (IQUAD.EQ.2.OR.IQUAD.EQ.3) X0=0.5+IXO*IXX
      X00=X0
      NFLAG=0
      DO 100 IYY=1, 50
      IF (IQUAD.LE.2) Y0=0.5+IYO*(IYY-1)
      IF (IQUAD.GT.2) Y0=0.5+IYO*IYY
      X0=X00
      Z0=0.
      DCA=DCAA
      DCB=DCBB
      DCC=DCCC
      CALL PITOPID(XSO, YSO, ZSO, ROD, +1., COD, EPSOD, THETSOD, PHISOD,NA)
      IF (IFLAG.EQ.0) NFLAG=NFLAG+1
      IF (NFLAG.EQ.40) GO TO 120
      IF (IFLAG.EQ.0) GO TO 100
C     IF IFLAG=0, THEN THE RAY MISSED THE OUTER SPHERE.
      IF (IIFLAG.EQ.1) GO TO 30
      IF (S.LT.0.) PRINT 341
      IF (S.LT.0.) GO TO 30
      OPL=OPLINC
      IF (S.GT.0.) CALL CCESC
      IF (S.GT.0.) X0=X0+XSO
      IF (S.GT.0.) Y0=Y0+YSO
```

```
      IF (S.GT.0.) Z0=Z0+ZS0
C     (X0,Y0,Z0) ARE THE COORD. OF THE 1ST PT. ON OUTER SPHERE
C     HIT BY RAY
      CALL ARCSPHR
      IF (S.GT.0.)
     1 CALL DIRCOS(NA, NINNER(1), -1.0, THETS0D, PHIS0D, R0D, EPS0D, DUM)
      ISR=NISURF
      DO 70 IS=1, NISURF
      CALL PTTOPID(XSI(IS), YSI(IS), ZSI(IS), RID(IS), +1.0,
     1                    CID(IS), EPSID(IS), THETSID(IS),
     2                    PHISID(IS), NINNER(IS))
      IF (IFLAG.EQ.0) GO TO 80
C     IF IFLAG=0, THEN THE RAY MISSED THE INNER SPHERE.
      IF (IFLAG.EQ.1) GO TO 30
      IF (S.LE.0.) PRINT 342
      IF (S.LE.0.) GO TO 30
C     IF S=0 HERE, IT MEANS THAT AT THIS PT. THE WALL
C     THICKNESS BETWEEN INNER AND OUTER SPHERES = 0.
      OPL=OPL+OPLINC
      CALL CCESC
      X0=X0+XSI(IS)
      Y0=Y0+YSI(IS)
      Z0=Z0+ZSI(IS)
C     NOW (X0,Y0,Z0) IS THE 1ST PT. ON AN INNER ELLIPSOID HIT BY RAY
      CALL ARCSPHR
      CALL DIRCOS(NINNER(IS), NINNER(IS+1), -1.0, THETSID(IS),
     1                  PHISID(IS), RID(IS), EPSID(IS), NIR(IS))
      IF (IREFLAG.EQ.1) GO TO 80
   70 CONTINUE
      NIS=NISURF
   80 IF (IS.LE.NISURF) NIS=IS-1
      DO 90 IS=NIS, 1, -1
      ISR=ISR+1
      CALL PTTOPID(XSI(IS), YSI(IS), ZSI(IS), RID(IS), -1.0,
     1                    CID(IS), EPSID(IS), THETSID(IS),
     2                    PHISID(IS), NINNER(IS+1))
      IF (IFLAG.EQ.0) PRINT 343
      IF (IFLAG.EQ.0) GO TO 30
C     IF IFLAG=0 HERE, IT MEANS A RAY LEAVING THE INNER SPHERE
C     MISSED THE OTHER SIDE OF THE INNER SPHERE WHICH IS IMPOSSIBLE
      IF (IFLAG.EQ.1) GO TO 30
      IF (S.LT.0.) PRINT 344
      IF (S.LT.0.) GO TO 30
C     IF S=0 HERE, IT MEANS THAT THE RAY LEAVING THE 1ST PT. ON
C     INNER SPHERE TRAVELED 0 DIST. TO THE OTHER SIDE OF INNER SPH.
      OPL=OPL+OPLINC
      CALL CCESC
      X0=X0+XSI(IS)
      Y0=Y0+YSI(IS)
      Z0=Z0+ZSI(IS)
C     (X0,Y0,Z0) ARE NOW THE COORD. OF PT. ON FAR SIDE OF INNER SPH.
      CALL ARCSPHR
      CALL DIRCOS(NINNER(IS+1), NINNER(IS), +1.0, THETSID(IS),
     1                 PHISID(IS), RID(IS), EPSID(IS), NIR(ISR))
   90 CONTINUE
      CALL PTTOPID(XS0, YS0, ZS0, R0D, -1., C0D, EPS0D,
     1                   THETS0D, PHIS0D, NINNER(1))
      IF (IFLAG.EQ.0) PRINT 345
      IF (IFLAG.EQ.0) GO TO 30
C     IF IFLAG=0 HERE, IT MEANS A RAY LEAVING THE INNER SPHERE
```

```
C      MISSED THE FAR SIDE OF THE OUTER SPHERE WHICH IS IMPOSSIBLE.
       IF (ITFLAG.EQ.1) GO TO 5Ø
       IF (S.EQ.Ø.) PRINT 346
       IF (S.EQ.Ø.) GO TO 5Ø
C      IF S=Ø HERE, IT MEANS THAT AT THIS PT. THE WALL THICKNESS
C      BETWEEN INNER AND OUTER SPHERE = Ø.
       OPL=OPL+OPLINC
       CALL CCESC
       XØ=XØ+XSC
       YØ=YØ+YSC
       ZØ=ZØ+ZSC
C      (XØ,YØ,ZØ) ARE NOW THE COORD. OF PT. ON FAR SIDE OF OUTER SPH.
       CALL ARCSPHR
       ISR=2*NISURF+1
       CALL DIRCOS(NINNER(1), NA, +1.Ø, THETSOD, PHISOD, ROD, PSOD,
      1                 NIR(ISR))
       IF (IRFLAG.EQ.1) GO TO 1ØØ
C      I ASSUME HERE THAT IF THERE IS INTERNAL REFLECTION AT THIS
C      POINT, THE RAY IS LOST.
       A=B=Ø.
       C=1.
       CALL PTTOPLN
       OPL=OPL+S*NA
       ZP=D
       DCA=-DCA
       DCB=-DCB
       DCC=-DCC
       CALL PTTOPLN
       OPL=OPL+S*NA
       ZP=F(1)
       IX=XØ+51.
       IY=YØ+51.
       IF (IX.GT.1ØØ.OR.IY.GT.1ØØ) GO TO 1ØØ
       IF (IX.LE.  Ø.OR.IY.LE.  Ø) GO TO 1ØØ
       VIEW(IX,IY)=VIEW(IX,IY)+Ø.5*COS((OPL-REFRAY)*TERM1)
   1ØØ CONTINUE
   11Ø CONTINUE
   12Ø PRINT 35Ø, IQUAD, (NIR(IS), IS=1, ISR)
       PRINT 44Ø, MAXPASS
       DO 13Ø I=1, 1ØØØØ
       IF (VIEW(I).LT.Ø.Ø) VIEW(I)=Ø.Ø
       IF (VIEW(I).GT.1.Ø) VIEW(I)=1.Ø
   13Ø CONTINUE
       CALL CLOCK(TIME, DATE)
       PRINT 36Ø, DATE, TIME
C
C      FROM HERE ON, MANY OF THE SPECIAL PLOTTING SUBROUTINES USED
C      ARE PART OF THE PLOT-1Ø TERMINAL CONTROL SYSTEM DESCRIBED BY
C      TEKTRONIX IN DOCUMENTS NUMBERED Ø62-1474-ØØ AND Ø62-151Ø-ØØ.
C
C      THE FOLLOWING SECTION PLOTS POINTS OF 17 DIFFERENT INTENSITIES
C      SO ONE MAY ADJUST TEKTRONIX 4Ø15 TERMINAL FOR BEST CONTRAST.
   14Ø PRINT 38Ø
       READ(59,37Ø) OPTION
       IF (OPTION.NE.1HY) GO TO 17Ø
       IY=3ØØØ
   15Ø DO 16Ø I=1, 17
       IX=1ØØØ+I*2Ø
       CALL TOUTPT(27)
       CALL TOUTPT(28)
```

```
      CALL TOUTPT(ICT(I))
  160 CALL XYCNVT(IX, IY)
      CALL TSEND
      IY=IY-50
      IF (IY.GT.100) GO TO 150
      GO TO 140
C
  170  PHISOD= PHISOD*RAD
      THETSOD=THETSOD*RAD
      DO 180 IS=1, NISURF
       PHISID(IS)= PHISID(IS)*RAD
  180 THETSID(IS)=THETSID(IS)*RAD
      CALL NEWPAG
      CALL ANMODE
      PRINT 390, DATE, TIME,
     1             FIL, D, P, LAMBDA, PHASDIF,
     2             TEST, NA, NV
      PRINT 391, ROD, COD, ROD, PHISOD, THETSOD,
     1             XSO, YSO, ZSO
      PRINT 392, RID(1), CID(1), RID(1), PHISID(1), THETSID(1),
     1             XSI(1), YSI(1), ZSI(1), NINNER(1)
      IF (NISURF.EQ.1) GO TO 200
      DO 190 IS=2, NISURF
      PRINT 393
  190 PRINT 392, RID(IS), CID(IS), RID(IS), PHISID(IS), THETSID(IS),
     1             XSI(IS), YSI(IS), ZSI(IS), NINNER(IS)
  200 PRINT 395
      READ (59, 370) OPTION
C
C     THE FOLLOWING SECTION DEFINES THE PLOTTING AREA.
C     SCREEN SIZE IS 4095 X 3120 FOR THE ENHANCED GRAPHICS MODULE.
      CALL MOVABS(1548, 280)
      CALL DRWABS(2558, 280)
      CALL DRWABS(2558, 1290)
      CALL DRWABS(1548, 1290)
      CALL DRWABS(1548, 280)
      CALL TSEND
C
C  THE DO 230 LOOP WILL PLOT THE DESIRED POINTS QUADRANT BY QUADRANT
C  VARYING THE INTENSITY OF EACH POINT ACCORDING TO VALUE OF VIEW.
      CALL PTMOD
      DO 230 IQUAD=1, NQUAD
      GO TO(201, 202, 203, 204), IQUAD
  201 IXX=IYY=51
      LX=LY=100
      GO TO 210
  202 IXX=1
      LX=50
      GO TO 210
  203 IXX=IYY=1
      LX=LY=50
      GO TO 210
  204 IXX=51
      LX=100
  210 DO 220 I=IXX, LX
      CALL TSEND
      CALL TOUTPT(27)
      CALL TOUTPT(28)
      DO 220 J=IYY, LY
      IF (OPTION.NE.1HY.AND.VIEW(I,J).EQ.0.5) GO TO 220
```

```
      IX=I*10+1540
      IY=J*10+280
      IC=68+INT(VIEW(I, J)*57.)
C     IC IS THE INTENSITY CONTROL (MIN=68, MAX=125)
      CALL TOUTPT(IC)
      CALL XYCNVT(IX, IY)
  220 CONTINUE
  230 CONTINUE
C
      CALL BELL
      CALL MOVABS(1,600)
      CALL ANMODE
      PRINT 400
      READ(59,370) OPTION
      IF (OPTION.NE.1HY) GO TO 280
C     WILL NOW PLOT THE OUTLINE.
      CALL DWINDO(-49.5, 49.5, -49.5, 49.5)
      CALL TWINDO(1558, 2548, 290, 1280)
      C=COS(.01745329252)
      S=SIN(.01745329252)
      X0=ROD
      Y0=0.0
      X=X0+XS0
      Y=   YS0
      CALL MOVEA(X, Y)
      DO 240 J=1, 359
      X=X0*C-Y0*S
      Y=Y0*C+X0*S
      X0=X
      Y0=Y
      X=X+XS0
      Y=Y+YS0
  240 CALL DRAWA(X, Y)
C
  280 CALL MOVABS(1,400)
      CALL ANMODE
      PRINT 410
      READ(59,370) OPTION
      IF (OPTION.EQ.1HY) WRITE(3,420) DATE, TIME
      IF (OPTION.EQ.1HY) WRITE(3,430) VIEW
      CALL MOVABS(1, 200)
      CALL ANMODE
      GO TO 30
  500 END FILE 3
      CALL FINITT(0, 3000)
      CALL EXIT
C
  330 FORMAT (19HGIVE NAMELIST INPUT )
  341 FORMAT ("A RAY TRAVELED A NEG. DIST. TO REACH OUTER SURFACE.")
  342 FORMAT ("A RAY TRAVELED A NEG. DIST. GOING FROM OUTER SURF. TO INNER.")
  343 FORMAT ("A RAY WHICH ENTERED INNER SURF. MISSED OTHER SIDE.")
  344 FORMAT ("A RAY TRAVELED A NEG. DIST. GOING FROM INNER SURFACE TO OUTER")
  345 FORMAT ("A RAY LEAVING INNER SURFACE MISSED OUTER SURFACE.")
  346 FORMAT ("STOP 6")
  350 FORMAT ("IN QUAD", I2, ". NO. OF INTERNAL REFLECTIONS="/ (3I5)
  360 FORMAT (//A10, 3X, A10)
  370 FORMAT (A1)
  380 FORMAT ("ADJUST INTENSITY? (YES OR NO)")
  390 FORMAT (A10, 3X, A10//
     1           4HELL=, F6.2, 2X, 4H  D=, F5.2, 5X, 4H  P=, I2, 6X,
```

```
     2         7HLAMBDA=, F6.4, 3X, 8HPHASDIF=, F6.4/
     3         5HTEST=, F5.3, 3X, 4H NA=, F6.4, 3X, 4H NV=, F6.4 //)
 391 FORMAT (4HROD=, F5.2, 3X, 4HCOD=, F5.2, 3X, 4HROD=, F5.2, 3X,
     1         7HPHISOD=, F6.2, 3X, 8HTHETSOD=, F6.2/
     2         4HXSO=, F5.2, 3X, 4HYSO=, F5.2, 3X, 4HZSO=, F5.2 //)
 392 FORMAT (4HRID=, F5.2, 3X, 4HCID=, F5.2, 3X, 4HRID=, F5.2, 3X,
     1         7HPHISID=, F6.2, 3X, 8HTHETSID=, F6.2/
     2         4HXSI=, F5.2, 3X, 4HYSI=, F5.2, 3X, 4HZSI=, F5.2, 3X,
     3         7HNINNER=, F6.4//)
 393 FORMAT ("FOR THE NEXT INNER ELLIPSOID")
 395 FORMAT (16HPLOT BKGD. PTS.?)
 4ØØ FORMAT ("OUTLINE?")
 41Ø FORMAT ("MAKE FILM FILE?")
 42Ø FORMAT (2A1Ø)
 43Ø FORMAT (2ØF6.4)
 44Ø FORMAT ("MAX NPASS IN PTTOUID =", I2)
 45Ø FORMAT ("EPSOD=Ø. WHILE ROD.GT.Ø")
 46Ø FORMAT ("EPSID(", I1, ")=Ø. WHILE RID(", I1, ").GT.Ø")
     END
     SUBROUTINE ARCSPHE
     COMMON A, B, C, DCA, DCB, DCC, R, THET, PHI, S,
    1         XØ, YØ, ZØ, XP, YP, ZP, IFLAG, VIEW(1ØØ,1ØØ)
     T1=SIN(THET)
     A=T1*COS(PHI)
     B=T1*SIN(PHI)
     C=COS(THET)
     RETURN
     END
     SUBROUTINE CCESC
C    PURPOSE - TO OBTAIN CART. COORD. GIVEN SPHERICAL COORD.
     COMMON A, B, C, DCA, DCB, DCC, R, THET, PHI, S,
    1         XØ, YØ, ZØ, XP, YP, ZP, IFLAG, VIEW(1ØØ,1ØØ)
     T1=R*SIN(THET)
     XØ=T1*COS(PHI)
     YØ=T1*SIN(PHI)
     ZØ=R*COS(THET)
     RETURN
     END
     SUBROUTINE DIRCOS(ETA1, ETA2, SIGN, THETS, PHIS, RØ, EPS, NIR)
C    PURPOSE - TO CALC NEW DIRECTION COSINES
C    GIVEN OLD DIRECTION COSINES(DCA,DCB,DCC), A, B, C, AND
C    THE INDICES OF REFRACTION(ETA1,ETA2)
     INTEGER P
     COMMON A, B, C, DCA, DCB, DCC, R, THET, PHI, S,
    1         XØ, YØ, ZØ, XP, YP, ZP, IFLAG, VIEW(1ØØ,1ØØ),
    2         DELS, T2, P, OPLINC, TEST, IER(9),
    3         IRFLAG, ITFLAG, MAXPASS, HATTERM
     COMMON /LEGEND/PLMTHET((Ø,8),8), PLMTHIS((Ø,8),8)
     COMMON /CONST/COSTHET, COSTHIS, OOSQR2P, PHID, TERMS((Ø,21))
     DCINTRE(A1,A2)=T2*(A1-2.Ø*HATTERM*A2)
     IRFLAG=Ø
     T1=ETA1/ETA2
     T6=DCA**2+DCB**2+DCC**2
     T2=1.Ø/SQRT(T6)
     IF (EPS.NE.Ø.Ø) GO TO 5
     T3=(A*DCA+B*DCB+C*DCC)*T2
     T4=1.-T1**2*(1.-T3**2)
     IF (T4.LE.Ø.Ø) GO TO 5Ø
     T4=SIGN*SQRT  (4)-T1*T3
     T5=T1*T2
```

```
      DCA=T5*DCA+T4*A
      DCB=T5*DCB+T4*B
      DCC=T5*DCC+T4*C
      RETURN
C
    5 OORO=1.0/R0
      OORO2=OORO**2
C
      DCFUNCT(A1,A2)=T1*T2*A1+(SIGN*SQRT(T4)-T1*HATTERM*T2)*A2
      XYDOTN (A1,A2)=(SINTHET*A1-COSTHET*OORO*A1*SUM1+
     1                2.0*A2*OORO/SINTHET*SUM2)*DOTTERM
C
      SINTHET=SIN(THET)
      COTTHET=COSTHET/SINTHET
      SINPHI=SIN(PHI)
      COSPHI=COS(PHI)
C
      SUM1=0.0
      SUM2=0.0
      DO 30 L=0, P
      SUM3=0.0
      SUM4=0.0
      IF (L.EQ.0) GO TO 20
      DO 10 M=1, L
      SUM3=SUM3+FLOAT(M)*FACTORL(L-M)/FACTORL(L+M)*PLMTHET(L,M)*
     1      PLMTHTS(L,M)*SIN(FLOAT(M)*PHID)
   10 SUM4=SUM4+FACTORL(L-M)/FACTORL(L+M)*
     1      (FLOAT(M)*COTTHET*PLMTHET(L,M)-FLOAT(L+M)*FLOAT(L-M+1)
     2      *PLMTHET(L,M-1))*PLMTHTS(L,M)*COS(FLOAT(M)*PHID)
   20 SUM1=SUM1+TERMS(L)*SQRT(FLOAT(L)+0.5)*OOSQR2P*(FLOAT(L)*
     1      (PLMTHET(L-1,0)/SINTHET-COTTHET*PLMTHET(L,0))*PLMTHTS(L,0)
     2      +SUM4+SUM4)
   30 SUM2=SUM2+TERMS(L)*SQRT(FLOAT(L)+0.5)*OOSQR2P*SUM3
      DOTTERM=1.0/SQRT(1.0+OORO2*SUM1**2+
     1            (4.0*OORO2/SINTHET**2*SUM2)**2)
      XDOTN=XYDOTN(COSPHI, SINPHI)
      YDOTN=XYDOTN(SINPHI,-COSPHI)
      ZDOTN=(COSTHET+SINTHET*OORO*SUM1)*DOTTERM
      HATTERM=DCA*XDOTN+DCB*YDOTN+DCC*ZDOTN
      T4=1.0-T1**2*( 0-HATTERM**2/T6)
      IF (T4.LE.0.0) GO TO 60
      DCA=DCFUNCT(DCA, XDOTN)
      DCB=DCFUNCT(DCB, YDOTN)
      DCC=DCFUNCT(DCC, ZDOTN)
      HATTERM=DCA*XDOTN+DCB*YDOTN+DCC*ZDOTN
      RETURN
C
   50 HATTERM=DCA*A+DCB*B+DCC*C
      XDOTN=A
      YDOTN=B
      ZDOTN=C
   60 DCA=DCINTRF(DCA, XDOTN)
      DCB=DCINTRF(DCB, YDOTN)
      DCC=DCINTRF(DCC, ZDOTN)
      NIR=NIR+1
      IRFLAG=1
      RETURN
      END
      FUNCTION FACTORL(NIN)
      IF (NIN.EQ.0) FACTORL=1.0
```

```
      IF (NIN.EQ.0) RETURN
      FACTORL=NIN
      IF (NIN.LE.1) RETURN
      NI=IFIX(FACTORL)-1
      N=NI-1
      DO 10 I=1, N
      FACTORL=FACTORL*FLOAT(NI)
   10 NI=NI-1
      RETURN
      END
      FUNCTION PLM(L,MM,X)
C
C
C     PURPOSE
C     COMP. NORMALIZED ASSOCIATED LEGENDRE POLYNOMIALS BY RECURSION.
C
C     DESCRIPTION OF PARAMETERS
C        MM-MAGNETIC QUANTUM NUMBER.
C        L-SUBSIDIARY QUANTUM NUMBER.
C        X-COSINE OF THETA.
C
C     SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
C        NONE.
C
C     METHOD
C     IN ACCORD WITH MCLEAN AND YOSHEMINE CONVENTION.
C     EACH PLM IS NORMALIZED TO 1.0 ON (-1.,1.).
C     PROGRAM OF R. NERF MODIFIED BY S. GREEN.
C
      M=IABS(MM)
C
C     COMPUTE GENERAL CASE BY RECURSION UPWARDS IN L AND ACROSS IN M.
C
   10 FORMAT(//37H**** ERROR. ARG OUT OF RANGE FOR PLM(,2I5
     1       ,F12.5,2H).)
C
      XL=FLOAT(L)
      IF (M.GT.L.OR.L.LT.0) GO TO 90
      IF (L.EQ.0) PLM=1.0
      IF (L.EQ.0) RETURN
      P1=1.0
      P2=X
      DO 30 I=1, L
      XI=FLOAT(I)
      P3=((2.0*XI+1.0)*X*P2-XI*P1)/(XI+1.0)
      P1=P2
   30 P2=P3
      IF (M.EQ.0) GO TO 60
C
C     AT END OF LOOP P1=P(L,0,X)
C
      IF (ABS(X).GT.1.0) PRINT 10, L, MM, X
      Z=SQRT(ABS(1.0-X*X))
      IF (Z.GT.1.E-6) GO TO 40
C
C     IF Z=0, THEN X=1 AND PLM(1.)=0 FOR M.GT.0.
C
      PLM=0.0
      RETURN
   40 P2=(XL+1.0)*(P2-X*P1)/Z
```

```
      DO 5Ø I=1, M
      XI=FLOAT(I)
      P3=-2.Ø*X/Z*P2*XI-(XL+XI)*(XL-XI+1.Ø)*P1
      P1=P2
   5Ø P2=P3
C
C     NORMALIZATION.....
C
   6Ø XNORM=(2.Ø*XL+1.Ø)/2.Ø
      IF (M.LE.Ø) GO TO 8Ø
      XLM=XL+1.Ø
      XLP=XL
      DO 7Ø I=1, M
      XLM=XLM-1.Ø
      XLP=XLP+1.Ø
   7Ø XNORM=XNORM/(XLM*XLP)
   8Ø PLM=P1*SQRT(XNORM)
      PLM=PLM*SQRT(FACTORL(L+M)/(FACTORL(L-M)*(XL+Ø.5)))
      RETURN
   9Ø PRINT 1Ø, L, MM, X
      PLM=Ø.Ø
      RETURN
      END
      SUBROUTINE PTTOPT(XS, YS, ZS, RADIUS, SIGN, CD, EPS, THETS, PHIS,INDEX)
C   PURPOSE - TO CALC PT OF INTERSECTION OF RAY WITH ELLIPSOID
C   AND DISTANCE S FROM STARTING PT(XØ, YØ, ZØ) TO ELLIPSOID.
      INTEGER P
      REAL INDEX
      DIMENSION DELSIT(2Ø), SERIES(2Ø)
      COMMON A, B, C, DCA, DCB, DCC, R, THET, PHI, S,
     1          XØ, YØ, ZØ, XP, YP, ZP, IFLAG, VIEW(1ØØ,1ØØ),
     2          DELS, T2, P, OPLINC, TEST, TER(9),
     3          IRFLAG, ITFLAG, MAXPASS, MATTERM
      COMMON /LEGEND/PLMTHET((Ø,8),8), PLMTHTS((Ø,8),8)
      COMMON /CONST/COSTHET, COSTHTS, OOSQR2P, PHID, TERMS((Ø,2))
      DATA (SQRTPI =1.772453851)
      DATA (SQRPIO5= .7926654591)
      DATA (TWOO3  = .6666666667)
      DATA (OOSQR2P= .3989422801)
      IFLAG=1
      T1=DCA**2+DCB**2+DCC**2
      X=XØ-XS
      Y=YØ-YS
      Z=ZØ-ZS
      T3=DCA*X+DCB*Y+DCC*Z
      T4=T3**2/T1+RADIUS**2-X**2-Y**2-Z**2
      IF (T4.LE.Ø.Ø) IFLAG=Ø
      IF (T4.LE.Ø.Ø) RETURN
      T2=SQRT(T1)
      S=-T3/T2-SQRT(T4)*SIGN
      R=SQRT((DCA*S+T2*X)**2+(DCB*S+T2*Y)**2+(DCC*S+T2*Z)**2)/T2
      THET=ACOS((DCC*S+T2*Z)/(R*T2))
      PHI=ATAN2((DCB*S+T2*Y),(DCA*S+T2*X))
C   POINT OF INTERSECTION=(R,THETA,PHI) IN SPHERICAL COORD.
C   CENTER OF SPHERE IS AT PT (XS,YS,ZS)
C   IF IFLAG=1, THEN THE RAY DID NOT MISS THE SPHERE.
      IF (EPS.EQ.Ø.Ø) OPLINC=S*INDEX
      IF (EPS.EQ.Ø.Ø) RETURN
      R=RADIUS
      OS=S
```

```
      ITFLAG=Ø
      SØ=2.Ø*((T1*OS+T2*T3)/(T1*RADIUS**2))
      SGN=1.Ø
      IF (SØ.LT.Ø.Ø) SGN=-1.Ø
      SØSQ=SØ**2
      TWOOSØ=2.Ø/SØ
C
      TERMS(Ø)= TWOO3*EPS*CD*SQRTPI
      TERMS(1)= Ø.Ø
      TERMS(2)=-TWOO3*EPS*CD*SQRPIO5
      NPASS=Ø
      COSTHTS=COS(THETS)
    5 NPASS=NPASS+1
      IF (NPASS.EQ.2Ø) PRINT 2ØØ
      IF (NPASS.EQ.2Ø) ITFLAG=1
      IF (NPASS.EQ.2Ø) RETURN
    6 COSTHET=COS(THET)
      PHID=PHI-PHIS
C
      DO 2Ø L=Ø, P
      LODDCOR=1.Ø
      LODSCOR=1.Ø
      IF (COSTHET.LE.Ø.Ø.AND.MOD(L,2).EQ.1) LODDCOR=-1.Ø
      IF (COSTHTS.LE.Ø.Ø.AND.MOD(L,2).EQ.1) LODSCOR=-1.Ø
      DO 1Ø M=Ø, L
      IF (NPASS.EQ.1) PLMTHTS(L,M)=PLM(L,M,ABS(COSTHTS))*LODSCOR
   1Ø PLMTHET(L,M)=PLM(L,M,ABS(COSTHET))*LODDCOR
   2Ø CONTINUE
C
      IF (NPASS.EQ.1ØØØ) RETURN
C
      PREVSUM=SUM
      SUM=Ø.Ø
      IF (NPASS.EQ.1) PREVSUM=Ø.Ø
      IF (THETS.NE.Ø.Ø.OR.PHIS.NE.Ø.Ø) GO TO 4Ø
      DO 3Ø L=Ø, P
   3Ø SUM=SUM+TERMS(L)*SQRT(FLOAT(L)+Ø.5)*PLMTHET(L,Ø)
      SUM=SUM*OOSQR2P
      GO TO 9Ø
   4Ø DO 7Ø L=Ø, P
      SUM2=Ø.Ø
      IF (L.EQ.Ø) GO TO 6Ø
      DO 5Ø M=1, L
   5Ø SUM2=SUM2+FACTORL(L-M)/FACTORL(L+M)*PLMTHET(L,M)
     1     *PLMTHTS(L,M)*COS(FLOAT(M)*PHID)
   6Ø SUM=SUM+TERMS(L)*SQRT(FLOAT(L)+Ø.5)*OOSQR2P*(PLMTHET(L,Ø)
     1     *PLMTHTS(L,Ø)+SUM2+SUM2)
   7Ø CONTINUE
   9Ø T4=SØSQ+8.Ø*(SUM/RADIUS**3)+4.Ø*PREVSUM**2/RADIUS**4
      IF (T4.LE.Ø.Ø.AND.SIGN.EQ.-1.Ø) T4=AMAX1(2.ØE-6, ABS(T4))
      IF (T4.LT.Ø.Ø) TFLAG=Ø
      IF (T4.LT.Ø.Ø) RETURN
      IF (T4.LT.2.ØE-6.AND.SIGN.EQ.-1) T4=2.ØE-6
      DELSIT(NPASS)=(RADIUS**2/2.Ø)*(SGN*SQRT(T4)-SØ)
      S=OS+DELSIT(NPASS)
      R    =SQRT((DCA*S+T2*X)**2+(DCB*S+T2*Y)**2+(DCC*S+T2*Z)**2)/T2
      THET=ACOS((DCC*S+T2*Z)/(R*T2))
      PHI=ATAN2((DCB*S+T2*Y),(DCA*S+T2*X))
      IF (NPASS.EQ.1) GO TO 5
      DELS=DELSIT(NPASS)
```

```
      MAXPASS=MAX0(MAXPASS, NPASS)
      DPLINC=S*INDEX
      NPASS=1000
      GO TO 6
  200 FORMAT ("IN PITOOID, TEST TOO STRINGENT")
      END
C
      SUBROUTINE PITOPLN
C    PURPOSE - TO CALC PT OF INTERSECTION OF RAY WITH PLANE
C    AND DISTANCE S FROM STARTING PT(X0, Y0, Z0) TO PLANE
C    POINT OF INTERSECTION = (X0, Y0, Z0)
C    COORD OF STARTING PT OF RAY = (X0, Y0, Z0)
C    COORD OF SOME PT ON PLANE = (XP, YP, ZP)
      COMMON A, B, C, DCA, DCB, DCC, R, THET, PHI, S,
     1        X0, Y0, Z0, XP, YP, ZP, IFLAG, VIEW(100,100)
      T1=(A*(XP-X0)+B*(YP-Y0)+C*(ZP-Z0))/(A*DCA+B*DCB+C*DCC)
      X0=DCA*T1+X0
      Y0=DCB*T1+Y0
      Z0=DCC*T1+Z0
      S=SQRT(DCA**2+DCB**2+DCC**2)*T1
      RETURN
      END
```