AST/RO Technical Memorandum #16 (Preliminary Version)
13 October 1993

A SURFACE OF REVOLUTION METHOD FOR FABRICATION OF ELLIPSOIDAL MIRRORS AT BOSTON UNIVERSITY

Richard A. Chamberlin
Boston University, Dept. of Astronomy, 725 Commonwealth Ave
Boston, MA 02215

Al Smith
Boston University, Scientific Instrumentation Facility
590 Commonwealth Ave, Boston, MA 02215

Abstract

A submillimeter mirror using a surface of revolution method was produced on a Boston University owned Gidding, Lewis, & Fraser Horizontal boring mill (G&L mill). The mill has 60 inches of vertical travel and 80 inches of horizontal travel, and is equipped with ball lead screws and glass encoder scales. This mill is is controlled by a Dynapath Delta 20M numerical controller with a RS232 interface. One of the advantages of the surface of revolution method is the ease and simplicity of the programming. Thus, numerical control code for the mill was generated from a short VAX-11 FORTRAN program which is listed in this document. The mirror we produced has a figure deviation of less than 0.00067 inches rms from the theoretical ellipsoid surface as measured with a Brown and Sharpe profilometer. Although the mirror we produced is only 6 X 8 inches, its successful production does indicate that the G&L mill can be used to produce much larger mirror panels while maintaining accurate figure tolerances.

1 Introduction

The AST/RO telescope optical setup requires the use of ellipsoidal mirrors to transfer the focus of the optical system from one location to another. For example, the ellipsoidal secondary mirror transfers the focus of the primary mirror to a position 80 cm above the receiver table. The ellipsoidal receiver matching mirror we fabricated was designed by J. Bally to transfer this focus to the receiver optics, see AST/RO memo No. 12[?].

A schematic representation of an off axis ellipsoidal mirror is presented in Fig. 1. This ellipsoid has two foci located at points $x = -c$ and $x = +c$. Rays originating from $-c$ are refocused at $+c$ with a total reflection angle of 90° at the ellipsoidal mirror surface. The mirror center is at point P and the semi-major and semi-minor axes of the ellipse are a and b respectively. $R1$ is the distance from $-c$ to P and $R2$ is the distance from P to $+c$ (assuming geometrical optics - if gaussian optics are used, then $R1$ and $R2$ are the radii of the wave fronts at the mirror surface). Since the rays originating at $-c$ are in a focused beam, only a small section of the ellipsoid is required to reflect all of the rays originating at $-c$ and make an efficient mirror. This ellipsoidal mirror obviously can be made by rotating the ellipse shown in Fig. 1 around the x-axis. That is to say, we used

$$1 = \frac{x^2}{a^2} + \frac{r^2}{b^2} \tag{1}$$

with

$$r^2 = y^2 + z^2 \tag{2}$$

where x, y and z are in the coordinate system of the milling machine.

In the setup described below, the mill was programmed to cut arcs in the y-z plane. The arc radii r in the y-z plane were a function of the x position of the mill table such that the ball endmill contact point cuts arcs that describe the surface of the ellipsoid of Eq. 1. This method of setting up and programming the mill exploits the ability of the controller to cut arcs of arbitrary radius in the y-z plane, thus the ellipsoidal surface can be produced with relatively short and easy to understand numerical control (NC) code. No spline fitting, approximations, or computationally intensive routines are required. Other workers have used the surface of revolution method for producing millimeter wavelength mirrors, for example see Reference [?].

We fabricated an elliptical mirror having R1 = 145cm and R2=333.5 cm. A mechanical drawing of this mirror in shown in Fig. 2. Fig. 2 was made with grid coordinates of the mirror surface generated from the a program by J. Bally, ELLIPSOID.FOR, which is based on Reference [?]. ELLIPSOID.FOR is included at the end of this memo for reference purposes.

2 Milling machine coordinates

Referring to the figures, the center of the mirror is indicated by the point P which is the ellipsoid tangent point with tangent angle θ as indicated in Fig. 1. The back of the mirror is plane parallel to the tangent plane at point P. At the four corners of the mirror are shown holes for receiving tooling balls (Reid Tool Co., Model TCB-26875) so that the mirror can be precisely and repeatably located. These tooling balls were used during the mill setup to correctly align the mirror and zero the machine coordinates.

The G & L mill has a four axis controller and is capable of simultaneous motions in the x, y, z and A directions. The A axis and the z axis move in parallel directions. Fig. 3 and Fig. 4 show how the G&L coordinates systems are set up. Fig. 3 depicts what a user would see facing the G&L mill will the spindle pointed toward him and the mill control console on the right. The table moving to the left with respect to (w.r.t.) to spindle axis is the positive x direction and the spindle moving up w.r.t to the table is the positive y direction. The spindle moving inward, away from the user, is the positive z direction; thus, the mill controller uses a "left" handed coordinate system. Fig. 4 is shown from the vantage of a user looking down onto the mill table, again with the mill control console on the right. A fourth "A" axis is also shown. Positive motion of the "A" axis moves the mill table away(?) from the spindle. Also shown in these figures is a right angle support called a "knee" which supports the mirror while it is being cut. The knee is set up to support the mirror at the correct tangent angle, θ w.r.t., the x-y plane, see figures 1 and 2.

3 Generating the NC code

The attached FORTRAN program, NC.FOR, produces the numerical control code specific to the DYNAPATH Delta 20M controller installed on the Boston University G&L 4 inch horizontal boring mill. The program is set up so that user merely needs to alter a few specification lines at the beginning, recompile, and execute. When the program is running, no user inputs are required, but several of the ellipsoid parameters, such as the tangent angle θ are typed out to the screen. The NC program is written out to a file which is named by the character variable FILE with the extension ".NC". A setup file is also written with the extension ".SETUP". This program should work to produce any ellipsoidal surface which does not include a vertex: i. e., one of the points where the ellipsoid intersects the x axis. This program could easily be adapted to producing off-axis parabolic mirrors, with the same condition that the mirror not include the vertex. Also, the program will probably fail to work satisfactorily in the cases where θ is too close to $0°$, or too close to 90°. The program is designed to make the mill cut using only climb-milling because we found that when taking off chip during the rough cut, climb-milling produced no chatter

whereas conventional milling produced substantial chatter. On the final cut, where the chip is small, probably both types of milling could be done in a zig-zag pattern with no degradation in surface quality. NC.FOR could easily be modified to produce a zig-zag cut if required.

The ellipsoid parameters are specified by program variables R1, R2, THEATI, MirrorLength, MirrorWidth, P_TB_H, P_TB_W, P_TB_L, and R_TB. R1, and R2 are the distances from the foci to point P **as shown in Fig. 1 and described above. THEATI is the angle of incidence of the central ray at** P**. (If THETAI = 45°, then the angle between the incident and reflected central rays is 90°.) MirrorLength, and MirrorWidth are the rectangular dimensions of the mirror boundary, see Fig. 2. P_TB_H is the "height" dimension from** P **to the center of the tooling ball: P_TB_H = 0.4225 inches in Fig. 2. P_TB_W is the "width" dimension from** P **to the center of the tooling ball: P_TB_W = 2.7500 inches. P_TB_L is the "length" dimension from** P **to the tooling ball center: P_TB_L = 3.7500. R_TB is radius of the Reid tooling ball: R_TB = 0.2500 inches.**

The milling machine toolpath parameters are specified by the program variables R, SO, RF, CF, DWELL, S, and T01. Variable R is the radius of the ball end mill: R = 0.7500 inches in this case. SO is the mill step-over distance; i.e., it is the distance between adjacent arcs on the surface of revolution. SO is selected so as to

give the desired scallop height on the surface. Simple geometry shows that

$$SO \approx 2\sqrt{2HR} \tag{3}$$

where H is the desired scallop height and R is the radius of the ball end mill. We used SO = 0.1 (H = 0.002) for the rough cuts, and SO = 0.020 (H = 0.0001) inches for the finish cut. RF is the rapid feed speed in inches per minute that the mill uses when it is translating its axes and not cutting any metal. CF is the cutting speed inches per minute. We found that CF = 32 inches per minutes with a carbide 3/4 radius ball end mill (Bay State Carbide, 4 flutes, model 106B #124) provided as good as surface finish as CF = 16 inches per minute: therefore, we used the higher cutting speed. S is a code designating the spindle rotation speed. S = 16 is the highest spindle rotation accepted by the G & L and it corresponds to 1527 RPM. DWELL is a dwell time interval in seconds which we found necessary to program into the the "turning points" of the mill's y axis motion. That is, the mass of the spindle headstock and counter weight is several tons and sudden reversal of the direction headstock motion at CF = 32 inches per minute would occasionally cause the mill controller to shut down with a "Y SERVO ERROR" message. By programming in a short dwell time interval of 0.3 seconds at the end of the y axis motions, the mill controller was satisfied and no more "Y SERVO ERRROR" faults were encountered. T01 is the value of the tool holder offset

which is adjusted by the user to minimize the spindle extension so as to reduce chatter. In the case of this mirror, we used T01 = +49.0000 inches.

NC.FOR was written in VAX-11 FORTRAN which has several language extensions which may not be allowed on other FORTRAN compilers. Non-standard FORTRAN extensions we used are IMPLICIT NONE (line 0018), REAL∗8 (lines 0021-0024), variable names longer than eight characters, TYPE statements, and comments embedded in program statements (delineated with the "!" character). In order to make the input of user parameters easier, the version of NC.FOR presented in this memo is slightly modified from the original version we used to produce the mirror. The output NC program gives the coordinates of the ball end mill center point: thus, no tool offsets were added to the Dynapath tool table except as explained below.

4 Setting up the part and running the NC program

Before starting, the mirror blank was cut to size, mounting holes on the mirror backside where drilled and tapped, and the holes for the Reid tooling balls drilled and reamed, see Fig. 2. The mirror blank was held a Kurt vise which as mounted on the "knee" as shown in figures 3 and 4. The angle of the "knee", θ w.r.t. the x-y plane was set with a 5" sine bar. The x, y coordinates of the

milling machine where zeroed w.r.t. the center of a Reid tooling ball inserted in location 1, see figure 2. The z and A coordinates where zeroed by touching the tip of the ball end mill off of the Reid tooling ball inserted in location 1. The Dynapath controller was then pre-configured with the following parts offsets: PX = +52.0788 in, and PY = -2.7500 in. In order to minimize extension of the spindle, the mill tool table was preconfigured to give tool T01 an offset of +49.0000 in. The NC program was downloaded to the Dynapath controller from a host computer using the Dynapath RS232 interface. (The offsets PX and PY were calculated in NC.FOR.) Before cutting, the tooling ball was removed. The A axis was used to set the depth of the cut. The first rough chip was taken with A = 0.8000, the second rough chip was taken with A = 0.7500, and the final chip was taken with A = 0.7277. (The appropriate final value of A was found by NC.FOR taking into account the preconfigured value of T01: +49.0000 inches in this case.)

5 Measurement of mirror figure

After milling and before polishing, the surface quality of the mirror was judged to be between 63 and 125 microinches. A slight ridge, in the x-z plane could be seen along the centerline of the mirror. The ridge was due to a small amount of backlash in the z axis when the mill spindle reversed its direction of motion on each cutting pass. The height of the ridge was estimated to be less than 0.0002 inches.

The coordinates of a grid of points on the mirror surface was calculated with the aid of the program ELLIPSOID.FOR. Points on the calculated grid were measured with a Brown and Sharpe coordinate measurement machine. (These measurements were made before machine marks were polished out of the mirror surface.) We found that the best fit of the data to the grid of points generated by ELLIPSOID.FOR had a residual error of 0.00067 inches RMS with 111 points on the surface measured. This mirror was intended to operate with $\lambda = 600$ μm radiation; therefore, these measurements indicate that the mirror had a figure tolerance of better than λ /20 over its entire surface.

Note: This document was produced by LaTeXver. 2.09 running on a VAXstationII with VAX/VMS ver. 4.6. Figures 1, 3, 4, and 5, and FORTRAN code described here were also developed on the VAXstationII.

Listing of program: NC.FOR

This program was used for producing NC code for the G & L mill in accordance the surface of revolution method.

```
C        PRGM: NC.FOR
C        PROGRAMMER: RAChamberlin
C        LOCATION: BOSTON UNIVERSITY
C        LANGUAGE: VAX-11 FORTRAN
C        DATE: 2-JUL-93
C
C        Purpose:  calculates elliptical mirror surface and tool center. Uses
C        figure of revolution method.  Writes NC code for G&L 4"
C        horizontal boring mill Dynapath Delta 20M controller.
C
C        REFERENCE1 = IEEE TRANS. ON ANTENNAS & PROPAGATION, VOL. AP-31, NO. 4
C                                                                     (1983)
c

        IMPLICIT NONE
        INTEGER NXSTEP, I, RF, CF, S, IMAX
        INTEGER NUM                             !NC STATEMENT NUMBER
        REAL*8 PHI,  R,dx, XSTART, CX,CZ,THETA
        REAL*8  CGSA, A, CGSB, B, X2, B2,Y, DCX, DCY, TCX(1000)
        REAL*8 CRADIUS(1000)
        REAL*8 DPHI, XROT, ZROT, tphi, X(1000), RADIUS(1000)
        REAL R1, R2, THETAI, MirrorLength, MirrorWidth, Rad, Pi
        REAL P_TB_H, P_TB_L, P_TB_W, R_TB
        REAL Xoffset, Zoffset, FinalA, T01, PX, PY
        REAL INCH, F , F0, E, A0, THETAP, PSI
        REAL DWELL, S0, WIDTH, XEND, Z, TIME, GAMMA
        CHARACTER*30 NCFILE, FILE, SETUPFILE

C        ..."Hardwire" ELLIPSOID INPUT PARAMETERS, see Fig. 2 AST/RO memo 16....
        R1        = 145.454     !distance from first focus to P (cm)
        R2        = 333.559     !distance form 2nd  focus to P (cm)
        THETAI       = 45.0     !angle of incidence of ray at P (deg)
        MirrorLength = 8.0      !Length of mirror (parallel to x-axis) (inch)
        MirrorWidth  = 6.0      !Width of mirror (parallel to y-axis) (inch)
        P_TB_H  = 0.4225        !Height Distance from P to tooling ball center
        P_TB_L  = 3.7500        !Length Distance from P to tooling ball center
        P_TB_W  = 2.7500        !Width  Distance from P to tooling ball center
        R_TB    = 0.2500        !radius of tooling ball

c
C
c        ..."Hardwire" TOOLPATH PARAMETERS...

        R       = 0.75                !RADIUS OF BALL END MILL CUTTER (inches)
        S0      = 0.02                !STEP OVER DISTANCE FOR TOOLPATH (inches)
        RF        = 64                !RAPID FEED SPEED  (inches/min)
        CF        = 32                !CUTTING SPEED FEED (inches/min)
        DWELL    = 0.3                !PAUSE INTERVAL FOR SERVOS TO STABLIZE (sec.)
```

```
        S        = 16                 !SPINDLE SPEED (16 = 1527 RPM ON G&L)
        T01     = 49.0                !Tool holder offset preset into G&L controller
c                                 to prevent excessive spindle extension (inches)


C        ..."Hardwire" NC output file name...
        FILE = '492FIN' !SIX CHARACTERS
NCFILE = FILE
NCFILE(7:9) = '.NC'
SETUPFILE = FILE
SETUPFILE(7:12) = '.SETUP'

        OPEN(10,FILE=NCFILE,STATUS='NEW')
OPEN(11,FILE=SETUPFILE,STATUS='NEW')

        RAD = 57.295779
        PI  = 3.1415926535
        INCH = 2.54

        thetai = thetai/RAD

c          **********  Compute ellipse parameters
C          (SEE REFERENCE 1)

        f = R1*R2 / (R1 + R2)
        f0 = ABS(R1 + R2) - SQRT(R1*R1 + R2*R2 -
     $ 2.0*R1*R2*COS(2.0*thetai))
        f0 = f0 / 2.0
        e = 1.0 - (2.0*f0 / (ABS(R1+R2)))
        a = f0 / (1.0 - e)                          !Semi-major axis
        b = f0*SQRT((1.0+e)/(1.0-e))                  !Semi-minor axis

        TYPE1001, A, A/2.54, B, B/2.54
        WRITE(11, 1001) A, A/2.54, B, B/2.54
 1001   format( ' ellipsoid major axis = ', f12.4,' cm ', F12.4, ' in'/,
     $          ' ellipsoid minor axis = ', f12.4,' cm ', F12.4, ' in')

c         ....Calculate tangent angle THETA at P....
C          (SEE REFERENCE 1)

        A0 = SQRT(R1*R1 + R2*R2 - 2.0*R1*R2*COS(2.0*thetai))
        thetap = PI - ACOS((R1*R1 + A0*A0 - R2*R2) / (2.0*R1*A0))
        psi = thetap - thetai
        THETA =  (pi/2.0 - ABS(psi))
        type 1031, THETA*180./pi
        WRITE(11, 1031) THETA*180./pi
 1031        FORMAT(' TANGENT ANGLE = :', F12.4, ' Deg.')

c         ...calculate mirror center, (CX,CZ)...
        CX =   (SQRT(a*a-b*b) +  R1*COS(thetap))
        CZ = R1*SIN(thetap)
        write(6,1011) CX, -CZ, CX/2.54, -CZ/2.54
        write(11,1011) CX, -CZ, CX/2.54, -CZ/2.54
 1011   format( ' mirror center (cm): (x, z; ', f12.4, f12.4 ,' )' /,
     $          ' mirror center (in): (x, z: ', f12.4, f12.4 ,' )' )
```

```
      DX = mirrorLength + R       !overall range of toolpath IN X COORDINATES
      CZ = CZ/2.54                !convert to inches
      CX = CX/2.54                !convert to inches
      WIDTH =        mirrorWidth + R                !WIDTH OF TOOLPATH
c      ...Note: R is added to DX and WIDTH so tool cuts a little bit
c               of air beyond edges of mirror.

c      ...Calculate G&L setup offsets w.r.t. tooling ball No. 1 center...
c      ...Assumes G&L spindle zeroed w.r.t. tooling ball No. 1 as in memo...
      Xoffset = -ABS(P_TB_L)*COS(ABS(THETA)) +
     $                              ABS(P_TB_H)*SIN(ABS(THETA))
      Zoffset =  ABS(P_TB_L)*SIN(ABS(THETA)) +
     $                              ABS(P_TB_H)*COS(ABS(THETA))
      PX = CX + Xoffset
      PY = -ABS(P_TB_W)
      FinalA = (CZ - Zoffset) - (T01 + R_TB + R)
      TYPE1017, ABS(PX), PY, FinalA
      WRITE(11,1017) ABS(PX), PY, FinalA
 1017      FORMAT( /, ' ...SETUP PARAMETERS FOR G&L',
     $              ' WHEN ZEROED AS DESCRIBED IN MEMO...',/,
     $       ' PX =             ', F12.4,/,
     $       ' PY =             ', F12.4,/,
     $       ' A on final cut = ', F12.4 )

c
c      ...convert to inch units...
      a = a/2.54
      b = b/2.54


      XSTART = CX - dx/2.0


      XEND = XSTART + dx

      I = 1        !I IS STEPOVER INDEX
      X(1) = XSTART

 100      CONTINUE

C              ...COMPUTE LOCAL RADIUS FOR ELLIPSOIDAL SURFACE...
              X2 = X(I)**2
              RADIUS(I) =  B * Dsqrt( 1 - x2/(A*A) )

C
C      ...COMPUTE LOCAL SURFACE TANGENT ANGLE, PHI...
      PHI = ATAN ( -B*X(I)/(A*A) * 1.0/SQRT( 1 - X(I)*X(I)/(A*A) ) )
C
      tphi = 3.141593/2.0 - phi
      DCX  = R * DCOS (tPHI)
```

```
      DCY  = R * DSIN (tPHI)
      TCX(I)   = X(I) + DCX
      CRADIUS(I)   = RADIUS(I) - DCY

C        ...COMPUTE NEW X BASED ON STEPOVER...

      X(I+1) = X(I) + SO*COS(PHI)

      IF (X(I+1) .GT. XEND) THEN
              IMAX = I
              GOTO 200 !START WRITING NC PROGRAM
      ELSE
              I = I +1
              GOTO 100 !CONTINUE WITH STEPOVER
      ENDIF

C        ...START WRITING NC PROGRAM...
  200       CONTINUE

      TIME = 0.0

      I = 1


      NUM = 100        ! FIRST STATEMENT NUMBER


      WRITE(10, 1069) FILE
 1069       FORMAT ( ' (', A6,')')
      NUM = NUM + 2
      WRITE(10, 1070) NUM
 1070       FORMAT ( ' N', I4.4, '(9)E01$')
      NUM = NUM + 2
      WRITE(10, 1071) NUM
 1071       FORMAT ( ' N', I4.4, '(9)M08$')
      NUM = NUM + 2
      WRITE(10, 1072) NUM
 1072       FORMAT ( ' N', I4.4, '(9)M03T01S16$')


  300   CONTINUE

      IF ( I .GT. IMAX) GOTO 400

C        ...COMPUTE Z POSTION OF CUTTER CENTER BASED ON TOOLPATH WIDTH...
      Z = - SQRT( CRADIUS(I)**2 - (WIDTH/2.0)**2 )
      GAMMA = ABS( ATAN((WIDTH/2.0)/Z) )

C        ...NOTE: WE USE NEGATIVE Z because G&L HAS LEFT HAND COORDINATE
C        ...         SYSTEM...

C        ...WRITE G&L ARC START POSITION...

      NUM = NUM + 2
      WRITE(10,1000) NUM,RF,TCX(I),-WIDTH/2.0, Z
```

```
      TIME = TIME + WIDTH/RF
1000  FORMAT( ' N', I4.4, '(1)', 'F',I2,'X',F8.4,'Y',
     $              F7.4,'Z',F8.4,'$')
      NUM = NUM + 2
      WRITE(10,1010)NUM,DWELL
      TIME=TIME + DWELL/60.0
1010       FORMAT( ' N',I4.4,'(8)L',F3.1,'$')

C         ...EXECUTE ARC -  GIVE END POSTION (CLIMB MILL)...

      NUM = NUM + 2
      WRITE(10,1020) NUM,CF,WIDTH/2.0, Z
      TIME=TIME + CRADIUS(I)*2*GAMMA/CF
1020  FORMAT( ' N', I4.4, '(2)', 'F',
     $                      I2,'P1','Y',F6.4,'Z',F8.4,'J0.K0.D1$')
      NUM = NUM + 2
      WRITE(10,1030)NUM,DWELL
      TIME = TIME + DWELL/TIME
1030       FORMAT( ' N',I4.4,'(8)L',F3.1,  '$')

      I = I + 1

      GOTO 300

400       CONTINUE

C         ...WRITE END OF NC PGM...

      NUM = NUM + 2
      WRITE(10, 1040) NUM
1040       FORMAT ( ' N', I4.4, '(9)M09$')
      NUM = NUM + 2
      WRITE(10, 1050) NUM
1050       FORMAT ( ' N', I4.4, '(9)M30$')

      WRITE(10, 1060)
1060       FORMAT ( ' END')

      CLOSE(10)
      WRITE(6,1100) TIME
      WRITE(11,1100) TIME
1100  FORMAT( ' RUN TIME = ', F12.2, ' MINUTES')

      STOP
      END
```

Listing of program: ELLIPSOID.FOR

This program was used for producing Fig. 2 in this memo, and for check the mirror figure tolerance.

```
c       programmer: John Bally
c       location: Univ. of Colorado
c       date:       apr93
c       language: vax-11 Fortran, (sun f77 ?)
c
c       modified: 17jun93, by RAC, added A, B printout
c                30jun93: pi corrected; other ellipse parameters printed out;
c                "INCH" declared as real number.   Otherwise FORTRAN assumes
C                that "INCH" is integer and causes inaccurate output.
c
c       purpose: calculates grid for machining ellipsoidal reflectors
c       reference: ieee trans. on antennas & propagation, vol.ap-31,no.4,jul83
c
c  To use program:
c
c  f77 file.f
c
c  Run the resultuing a.out file and answer all the questions exactly as below.
c  The following parameters are appropriate for the 230GHz matching  mirror
c  for ASTRO.
c
c   Input shorter Radius, R1 (cm) =
c   31.677
c   Input longer Radius, R2 (cm) =
c   309.0
c   Angle of incidence on mirror, thetai (degrees) =
c   45.0
c   Major axis mirror diameter, Dx (cm) =
c   20.0
c   Minor axis mirror diameter, Dy (cm) =
c   15.0
c   mill step size in X, stepX (inches) =
c   0.2
c   mill step size in Y, stepY (inches) =
c   0.2
c
c  The results are written into a file called 'mirror'
c  You may need to change the stepX or stepY parameters to get a smoother
c  cut.  I recommend cutting the mirror in vax the first time.
c ****************************************************************
c       this program calculates the coeffs for an elliptical mirror
c       z=function of x and y. Coeffs are defined in TS's paper.
c ****************************************************************
       implicit none

       CHARACTER*30 MIRRORNAME
       double precision a, b, R1, R2, thetap, thetai
```

```
      double precision RAD, A0, PI, psi
      double precision AA, BB, B1, B2, B3
      double precision CC, C1, C2, C3
      double precision X, Y, Z
      real*8       dx, dy, stepx, stepy, f, f0, e,px, py, xin, yin
      real*8  zin, inch
      integer nx, ny, ix, iy

      RAD = 57.295779
      PI  = 3.1415926535
      INCH = 2.54

      open(10,FILE='mirror.DAT', STATUS = 'UNKNOWN')

      PRINT*,' ENTER MIRROR NAME'
      ACCEPT 1003, MIRRORNAME
 1003   FORMAT (  A30)
      WRITE(10,1004) MIRRORNAME
 1004       FORMAT (' ' , A30)
      print*,'Input shorter Radius, R1 (cm) ='
      read*, R1
      write(10, 1005) 'R1 (cm)        = ', R1
 1005       FORMAT( ' ', A11, F12.6)
      print*,'Input longer Radius, R2 (cm) ='
      read*, R2
      WRITE(10, 1005) 'R2 (cm)        = ', R2
      print*,'Angle of incidence on mirror, thetai (degrees) ='
      read*,thetai
      WRITE(10, 1005) 'THETAI         = ', THETAI
      print*,'Major axis mirror diameter, Dx (cm) ='
      read*, Dx
      print*,'Minor axis mirror diameter, Dy (cm) ='
      read*, Dy
      print*,'mill step size in X, stepX (inches) ='
      read*, stepX
      print*,'mill step size in Y, stepY (inches) ='
      read*, stepY

      stepX = stepX*INCH
      stepY = stepY*INCH
      X = -Dx / 2.0
      Y = -Dy / 2.0
      nx = (Dx / stepX) + 1
      ny = (Dy / stepY) + 1

      thetai = thetai/RAD

c **********  Compute ellipse parameters

      f = R1*R2 / (R1 + R2)
      f0 = ABS(R1 + R2) -
     $          SQRT(R1*R1 + R2*R2 - 2.0*R1*R2*COS(2.0*thetai))
      f0 = f0 / 2.0
      e = 1.0 - (2.0*f0 / (ABS(R1+R2)))
      a = f0 / (1.0 - e)
```

```
       b = f0*SQRT((1.0+e)/(1.0-e))

       write(6,1000) A, B
       write(10,1000) A, B

 1000        format( ' ellipsoid major axis = ', f12.4, ' cm', /,
    $                 ' ellipsoid minor axis = ',  f12.4, ' cm')

c **********  Transform coordinates to mirror center, tangent to mirror

       A0 = SQRT(R1*R1 + R2*R2 - 2.0*R1*R2*COS(2.0*thetai))
       thetap = PI - ACOS((R1*R1 + A0*A0 - R2*R2) / (2.0*R1*A0))
       psi = thetap - thetai
       type 1030, (pi/2.0 - ABS(psi))*180./pi
       WRITE(10, 1030) (pi/2.0 - ABS(psi))*180./pi
 1030        FORMAT(' TANGENT ANGLE = :', F12.4)

       AA = a*a*SIN(psi)*SIN(psi) + b*b*COS(psi)*COS(psi)

c         ...calculate mirror center in ellipse coordinates...
       px =  - (SQRT(a*a-b*b) +  R1*COS(thetap))
       py = R1*SIN(thetap)
       write(6,1010) px, py
       write(10,1010) px, py
 1010        format( ' mirror center (cm): (x, y; ', f12.4, f12.4 ,' )')


       do 200 iy=1,ny
       X = -Dx / 2.0
c

       do 100 ix=1,nx

       B1 = b*b*X*SIN(psi)*COS(psi) - a*a*X*SIN(psi)*COS(psi)
       B2 = b*b*COS(psi)*(SQRT(a*a-b*b)+R1*COS(thetap))
       B3 = a*a*R1*SIN(psi)*SIN(thetap)
       BB = 2.0*(B1 - B2 - B3)

       C1 = b*b*( X*SIN(psi) - (SQRT(a*a-b*b) +  R1*COS(thetap)))**2
       C2 = a*a*(X*COS(psi) + R1*SIN(thetap))**2
       C3 = a*a*Y*Y - a*a*b*b
       CC = C1 + C2 + C3

       Z = (-BB - SQRT(BB*BB - 4.0*AA*CC)) / (2.0*AA)

       Xin = X / INCH
       Yin = Y / INCH
       Zin = Z / INCH
       write(10,11) Xin, Yin, Zin
 11     format(3x,f8.4,5x,f8.4,5x,f8.4)

       X = X + stepX
 100    continue
       Y = Y + stepY
 200    continue
```

```
close(10)


stop
end
```

FIGURES

Figure 1: The ellipsoidal surface is generated by revolving the ellipse in the x-z plane around the x axis. The edge of the mirror, centered on the point P is shown schematically in the figure.

Figure 2: This figure a mechanical drawing of the mirror. The depth and outline of the ellipsoidal mirror surface represented in the drawing was derived from a grid of points generated from ELLIPSOID.FOR.

Figure 3: This schematic representation is meant to show how the G & L mill coordinate system is arranged. In this representation the veiwer is standing facing the spindle axis with the control console on his right. The spindle moving up w.r.t the mill table is the $+\ y$ direction. The mill table moving to the left w.r.t the spindle axis makes the cutter move in the $+\ x$ direction. Also shown, is the backside of the "knee" which supports the mirror while it is machined.

Figure 4: In this representation of the G & L coordinate system, the viewer is above the mill looking down. The control console is to his right. The spindle moving inward is the $+\ z$ direction, and as before, the mill table moving to the left effectively makes the cutter move in the $+\ x$ direction. The knee is shown with its front

surface set at the appropriate tangent angle θ. In this view, the y axis would be orthogonal to plane of the paper.

Figure 5: This figure is a from the vantage of a veiwer looking down on the mill as in Fig. 4. This figure is a close-up veiw of how the mill spindle and ball endmill should be zeroed w.r.t. to the Reid tooling ball.