A PRELIMINARY STUDY OF A VERY LARGE SPACE RADIOMETRIC ANTENNA

PRADEEP K. AGRAHAL

JANUARY 1979
A preliminary study to compute the size of a special radiometric reflector antenna is presented. Operating at 1 GHz, this reflector is required to produce 200 simultaneous contiguous beams, each with a 3 dB footprint of 1 km from an assumed satellite height of 650 km. The overall beam efficiency for each beam is required to be more than 90%.
1. **Summary**

A preliminary study to compute the size of a special radiometric reflector antenna is presented. Operating at 1 GHz, this reflector is required to produce 200 simultaneous contiguous beams, each with a 3 dB footprint of 1 km from an assumed satellite height of 650 km. The overall beam efficiency for each beam is required to be more than 90%.
2. Introduction

The purpose of this report is to present a preliminary study for the design of a large radiometric reflector antenna system. When orbiting at a height of 650 km, this antenna system is required to produce simultaneously 200 contiguous 3 dB circular footprints on the ground, each having a diameter of 1 km. The lowest frequency of operation is 1 GHz. The footprints are required to be as identical to each other as possible. The single most important requirement on the system is that the overall beam efficiency for the copolarized component in each of the 200 beams be better than 90% within the two and a half 3 dB beamwidths. This means that among other things, the cross polarization be minimum too (<25 dB).

When in orbit, the reflector may undergo considerable thermal distortions and its performance may change. A method is therefore needed to predict the performance of even a distorted reflector. Such a technique is discussed in Section 6 of this report.
3. **Spherical Reflector Approach**

A solution that meets the requirements set in Section 2 is schematically shown in Figure 1. The 200 beams are simultaneously obtained by stacking 200 identical feed antennas along a concentric circular arc in front of a spherical reflector such that each feed is pointing radially towards the spherical reflector surface. Each feed thus creates its own independent footprint. And since each feed antenna essentially sees an identical segment of the spherical reflector, the resulting 200 footprints are also practically identical. Observe that the angular separation between any two consecutive feed antennas (called $\theta$) is the same as the angular separation between the two adjacent footprints. This, for an altitude of 650 km and a footprint size of 1 km, turns out to be $0.088^\circ$ and the 200 feeds stacked along the feed arc thus subtend a total of $16.6^\circ$ angle at the center of the spherical reflector (Figure 2).

Notice that the angular separation between any two consecutive feeds depends only upon the altitude and the footprint size; the physical separation, however, is the product of the angular separation ($\theta = 0.088^\circ$) and the feed arc radius, and therefore, depends upon the radius of the feed arc also. The radius of the feed arc, therefore, should be large enough to provide enough physical room for each of the 200 feed antennas. It is assumed
Figure 1 -- Multibeam Spherical Reflector Concept
Figure 2 -- Geometry of Spherical Reflector Antenna
at this stage that each feed antenna should have, on an average, a room of at least 88 cm, which leads to a feed arc radius of 577 m. And since for spherical reflectors, the feed arc is generally located about halfway between the reflector and its center of curvature (nearer to the reflector), the radius of curvature of the spherical reflector is chosen to be 1175 m. Note that there is no specific reason to pick 88 cm for feed spacing except that the estimations of practical feed horn sizes suggest that a room of about a meter be available for each feed antenna. And of course, whether a feed horn limited in size to 88 cm at 1 GHz feeding a reflector with dimensions chosen above can give a satisfactory secondary far field pattern or not, remains to be checked.

Let us now consider an individual footprint which is caused by an individual feed antenna located at the feed arc. Each feed antenna illuminates a portion of the spherical reflector and it is the far field of this illuminated reflector aperture which must have (a) a 3 dB beamwidth of 0.088°, and (b) a beam efficiency of better than 90% within the two and a half 3 dB beamwidths. The later requires that the highest side lobe of the reflected pattern be less than -32 dB with wide angle side lobes below -80 dB. A study of various aperture distributions [1] indicates that for an operating frequency of 1 GHz, an aperture diameter of about 300 m (say, 305 m) with a rotationally symmetric cosine
squared field distribution produces both a 3 dB beamwidth of 0.088° and a side lobe at -32 dB, the side lobe fall off being -18 dB/Octave.

Returning to Figure 2, an illuminated aperture with a diameter of 305 m on the reflector corresponds to a cone of 15° half angle emanating from each feed antenna. Therefore, each feed antenna whose nominal diameter has been fixed to 88 cm at 1 GHz has to be able to produce a rotationally symmetric cosine squared far field pattern over ± 15°. The overall diameter of the spherical reflector dish to produce 200 beams then turns out to be 660 m as shown in Figure 2.
4. Feed Considerations

In the previous section, it was assumed that after reflection, each feed pattern gave rise to a rotationally symmetric aperture field which varied as cosine squared in the radial direction. For the reflector dimensions under consideration, the portion of the reflector illuminated by a feed is such a small fraction of the full sphere, that it is practically flat and therefore a rotationally symmetric cosine squared aperture distribution is easily achieved by a feed which too has a rotationally symmetric cosine squared pattern.

A rotationally symmetric cosine squared feed pattern can be generated by any one of the several types of horns. In the present study, however, a circular corrugated horn [2] is considered. The feed pattern of the corrugated horn used for the computations presented in the following sections is shown in Figure 3. Observe that the horn diameter at the mouth is 2 m which is larger than 88 cm, the space designated for each feed at the feed arc. Therefore, the feed horns will have to be staggered around the feed arc so that they are still on an average 88 cm or 0.088° apart. The beam efficiency of the feed pattern within 15°, which corresponds to the edge of the reflector is 98.3%. This number is important because the overall beam efficiency of the antenna system is the product of this efficiency and the beam efficiency of the secondary pattern.
Figure 3 -- Feed Pattern of a Circular Corrugated Horn
5. **Secondary Pattern**

For the reflector geometry shown in Figure 2 and using the feed pattern presented in Section 4, the computed secondary radiation pattern is shown in Figure 4. It has a 3 dB beamwidth of 0.080° and a maximum cross polarization level of less than -200 dB. The beam efficiency of the secondary pattern at two and a half 3 dB beamwidths is 93.4%, the overall beam efficiency, therefore, being better than 91%.

One of the concerns in spherical reflector applications is the resulting spherical aberration. It is of interest to note that in the present case, such a small segment of the sphere is being used as reflector that the maximum spherical aberration near the edge of the illuminated aperture (where the field strength is -26.7 dB, Figure 3) is equivalent to a phase error of only about 18°. Such a small phase error causes a negligible degradation in the antenna gain.
Figure 4 -- Computed Secondary Pattern. Feed Pattern was used over \( \pm 16^\circ \) with \( 1^\circ \) increment.
6. Thermal Distortion Considerations

The performance of a reflector antenna in space is sometimes not the same as predicted by the initial design because the reflector undergoes severe distortions due to thermal variations. If the distorted shape of the reflector is quite a bit different from the original spherical shape, the reflector performance may change significantly and may even become unsatisfactory. Therefore, it will be desirable to be able to predict the performance of even the distorted reflector. If the distorted reflector surface could be known analytically, then the reflector performance of course could be accurately predicted. It is not generally possible to know an analytic expression for the entire distorted reflector surface at all times. Alternatively, a sampling scheme can be implemented such that the coordinates of many discrete target points located along a rectangular grid on the reflector surface are known. Then, a smooth tight cubic surface can be fitted through the four corners of each of the rectangular grid patches such that the whole composite reflector surface is continuous and has continuous partial derivatives. Using this piecewise analytic expression for the reflector surface, the reflector properties can be computed. Needless to say, the target points on the reflector surface must be dense enough to sample the distortions and such that the surface between the measured points could be assumed tightly stretched.
To demonstrate that the far field radiation pattern can indeed be accurately computed even when the reflector surface is known only at certain discrete points, the following example is presented.

Computations presented in Figure 4 are made again, this time, instead of using a single analytic expression for the entire spherical reflector surface, though, the x-, y-, z- coordinates of 45 equispaced points on the reflector surface are used. These surface points are located on the reflector surface along a rectangular grid as shown in Figure 5, the points being 40 m or 133.3 wavelengths apart. For computational purposes, the reflector surface over any rectangular patch is expressed as a bi-spline under tension [4]. In Figure 6, the far field radiation pattern computed by using a single analytic expression for the entire spherical reflector surface (as in Figure 4) is shown by solid lines. On the same figure, the far field radiation pattern computed by using the piecewise analytic composite surface through 45 target points on the reflector is plotted with solid dots. The field values not shown by solid dots are too close to the solid line curve values to distinguish. The modifications needed in the computer program in Reference 3 to make the present surface fitting computations are shown in the Appendix.

In conclusion, for actual distorted reflector conditions, where the whole distorted reflector surface is not known
Figure 5 -- Location of Surface Target Points used for Surface Fitting Calculations
analytically, accurate far field computations can be made by using our computer program which will accept for the reflector surface geometry, a set of discrete reflector surface points. The basic underlying assumption is that the surface is smooth between the target sample points.
7. References


Appendix

The computer program documented in Reference 3 has been modified such that now it can be used for making also the surface fitting reflector calculations similar to the ones presented in Section 6. The purpose of this appendix is to document the corresponding modifications, changes, and additions.

Two new features have been added to the computer program REFLCTR (documented in Reference 3). The first feature, which is not of direct concern to the subject matter of this report is that in addition to parabolic, spherical, and ellipsoidal reflectors, the program REFLCTR can now also handle planar reflectors. A planar reflector is specified by (a) three cartesian coordinates of a point on the surface of the planar reflector - PLNPNT(1), PLNPNT(2), and PLNPNT(3), and (b) the three cartesian components of a unit vector normal to the reflector surface - PLNORM(1), PLNORM(2), and PLNORM(3). The value of the integer variable SURFACE must be set to 4 for a planar reflector.

The second new feature is that in addition to specifying a reflector surface as being parabolic, spherical, ellipsoidal, or planar by setting SURFACE = 1, 2, 3 or 4 respectively, one can now also choose to specify any of the above reflector surfaces in terms of only a finite number of discrete target points located along a rectangular grid (assumed square grid here for simplicity) on the reflector surface. This is done by setting the integer variable NEEDFIT to a nonzero positive value. SIGMA is the tension factor (defined later) used for fitting the surface through the
grid points, the grid spacing being DISFAC for both the y-
and the z- directions. All these newly defined variables
along with the ones already defined in Reference 3 are read,
as before, in the subroutine NPUT. A listing of modified
NPUT is given in Figure A-1.

As a result of implementing the above two features, the
subroutine APERTUR also changes. A listing of the new APERTUR
is given in Figure A-2. Notice that in this subroutine,
before statement number 100, the coordinates of surface
target points are first stored in dimensioned arrays called
EXTRAX, EXTRAY, and EXTRA2 and then the subroutine SURF1 is
called to compute the parameters necessary to compute an
interpolatory surface passing through the surface grid
points. Later on in the subroutine APERTUR (before statement
number 130), subroutine SURFD2 is called to interpolate
the reflector surface at the given coordinate pair and to
compute the components of a normal vector at the inter-
polated point.

The subroutines SURF1 and SURFD2 are from "A Spline
Under Tension Package for Curve and Surface Fitting" by
A. K. Cline, Department of Computer Science, University of
Texas, Austin. This package of subroutines is an extension
of Cline's work reported in Reference 4. A listing of
subroutines SURF1 and SURFD2 which also includes definition
of parameters used in the subroutines is presented in
Figure A-3.
Figure A-1
Figure A-1 (Continued)
Figure A-2
**Figure A-2 (Continued)**
IF(NFEQF.LT.1) GO TO 130
X=SQRND(YO,ZO,YNRM,ZNRM,IFIT,IF1,FXYAR,FXTRZ,FXTRX,IFIT,
FXTRA, SigMA)
R=SORT(XO**2+YO**2+ZO**2+70**2)
XMAG=SQRND(XO**2+YO**2+ZNRM**2)
NHAT(1)=1.0/XMAG
NHAT(2)=SIGN(YNRM,YO)/XMAG
NHAT(3)=SIGN(ZNRM,ZO)/XMAG
GO TO 130
130 GO TO (131,132,133,134,135)

131 NHAT(1)=PLNRM(1)
NHAT(2)=PLNRM(2)
NHAT(3)=PLNRM(3)
GO TO 134
132 NHAT(1)=-X0/ANTNF
NHAT(2)=-YO/ANTNF
NHAT(3)=-ZO/ANTNF
GO TO 134
134 NHAT(1)=-XO/ANTNF
NHAT(2)=-YO/ANTNF
NHAT(3)=-ZO/ANTNF
GO TO 134
136 NHAT(1)=2.0/ANTNF/SORT(XO**2+Y0**2+Z0**2)
NHAT(2)= -YO/SORT(XO**2+Y0**2+Z0**2)
NHAT(3)= -ZO/SORT(XO**2+Y0**2+Z0**2)
138 SCALAR=2.0*(2.1)*NHAT(1)+R(2.1)*NHAT(2)+R(3.1)*NHAT(3)
ON 1500 I=1,3
1500 SR(1)=R(1.1)*SCALAR+NHAT(1)
FT1=P(1.1,IT,IP)/R
FPI=P(1.1,IP)/R
C(1)=CST*CSSP*FT1-SINP*EPI
C(2)=CST*SINP*FT1+CSSP*EPI
C(3)=SINT*FT1
ON 2000 I=1,3
F(I)=0.0
ON 2000 J=1,3
2000 F(I)=F(I)+A(I,J)*C(J)
SCALAR=2.0*(E(I)*NHAT(1)+F(I)*NHAT(2)+NHAT(3)+F(I)*NHAT(3)
ON 2500 I=1,3
2500 F(I)=SCALAR+NHAT(1)-F(I)
Y=Y0+(X0-XC)*SR(J)/SR(1)
Z=Z0+(X0-XC)*SR(1)/SR(1)
O=SQRND(XC-X0)+(Y-Y0)*(Y-Y0)+(Z-Z0)*(Z-Z0)
PHASE=P12*(3.1)*S1NAM+PR(3,1,IP)
PNFW(1)=PNFW(1)+Y
PNFW(2)=PNFW(2)+Z

Figure A-2 (Continued)
Fig: A-2 (Continued)
Figure A-2 (Continued)
SUBROUTINE SIRE1 (M,N,X,Y,Z, ZX1,ZY1,ZM,ZXY1,
                   7XYM,7XY1,7XVM,TSLSW,7P,TEMP,
                   SIGMA,IER)

C THIS SUBROUTINE DETERMINES THE PARAMETERS NECESSARY TO
C COMPUTE AN INTERPOLATORY SURFACE PASSING THROUGH A RECT-
C ANGULAR GRID OF FUNCTIONAL VALUES. THE SURFACE DETERMINED
C CAN BE REPRESENTED AS THE TENSOR PRODUCT OF SPLINES UNDER
C TENSION, THE X- AND Y-PARTIAL DERIVATIVES AROUND THE
C BOUNDARY AND THE X-Y-PARTIAL DERIVATIVES AT THE FOUR
C CORNERS MAY BE SPECIFIED OR OMMITTED FOR ACTUAL MAPPING
C OF POINTS INTO THE SURFACE IT IS NECESSARY TO CALL THE
C FUNCTION SIRE2.

C IMPLICIT-
C
M IS THE NUMBER OF GRID LINES IN THE X-DIRECTION, I. E.
C LINES PARALLEL TO THE Y-AXIS (M.GF. 2).
C
N IS THE NUMBER OF GRID LINES IN THE Y-DIRECTION, I. E.
C LINES PARALLEL TO THE X-AXIS (N.GF. 2).
C
X IS AN ARRAY OF THE N X-COORDINATES OF THE GRID LINES
C IN THE X-DIRECTION. THESE SHOULD BE STRICTLY INCREASING.
C
Y IS AN ARRAY OF THE M Y-COORDINATES OF THE GRID LINES
C IN THE Y-DIRECTION. THESE SHOULD BE STRICTLY INCREASING.
C
Z IS AN ARRAY OF THE M*N FUNCTIONAL VALUES AT THE GRID
C POINTS. I. E. Z(I,J) CONTAINS THE FUNCTIONAL VALUE AT
C (X(I),Y(J)) FOR I = 1,...,M AND J = 1,...,N.
C
Z IS THE ROW DIMENSION OF THE MATRIX Z USED IN THE
C CALLING PROGRAM (7.GF. Z).
C
ZX1 AND ZXM ARE ARRAYS OF THE X-PARTIAL DERIVATIVES
C OF THE FUNCTION ALONG THE X(I) AND X(M) GRID LINES.
C RESPECTIVELY. THIS ZX1(I) AND ZXM(J) CONTAIN THE X-PART-
C IAL DERIVATIVES AT THE POINTS (X(I),Y(J)) AND
C (X(M),Y(J)), RESPECTIVELY, FOR J = 1,...,N. EITHER OF
C THESE PARAMETERS WILL BE IGNORRED (AND APPROXIMATIONS
C SUPPLIED INTERNALLY) IF ISPLSN 0 INDICATES.
C
ZY1 AND ZYN ARE ARRAYS OF THE Y-PARTIAL DERIVATIVES
C OF THE FUNCTION ALONG THE Y(1) AND Y(N) GRID LINES.
C RESPECTIVELY. THIS ZY1(I) AND ZYN(J) CONTAIN THE Y-PART-

Figure A - 3

27
{101. DERIVATIVES AT THE POINTS \((x(i), y(i))\) AND \\
\((x(i), y(n))\), RESPECTIVELY, FOR \(i = 1, \ldots, n\). EITHER OF \\
THESE PARAMETERS WILL BE IGNORED (AND ESTIMATIONS \\
SUPPLIED INTERNALLY) IF ISLPW = 0 INDICATES. \\

\((x(i), y(i)), (x(i), y(n)), (x(i), y(n))\), AND \((x(n), y(n))\), \\
RESPECTIVELY, ANY OF THESE PARAMETERS WILL BE IGNORED (AND \\
ESTIMATIONS SUPPLIED INTERNALLY) IF ISLPW = 0 INDICATES. \\

ISLPW CONTAINS A SWITCH INDICATING WHICH NORMALLY \\
DERIVATIVE INFORMATION IS USER-SUPPLIED AND WHICH \\
SHOULD BE ESTIMATED BY THIS SUBROUTINE. TO DETERMINE \\
ISLPW, LET \\

\[11 = 0 \text{ IF } x1 \text{ IS USER-SUPPLIED (AND) } 1 \text{ OTHERWISE} \]
\[12 = 0 \text{ IF } xM \text{ IS USER-SUPPLIED (AND) } 1 \text{ OTHERWISE} \]
\[13 = 0 \text{ IF } y1 \text{ IS USER-SUPPLIED (AND) } 1 \text{ OTHERWISE} \]
\[14 = 0 \text{ IF } yM \text{ IS USER-SUPPLIED (AND) } 1 \text{ OTHERWISE} \]
\[15 = 0 \text{ IF } xY11 \text{ IS USER-SUPPLIED} \]
\[16 = 0 \text{ IF } xYM1 \text{ IS USER-SUPPLIED} \]
\[17 = 0 \text{ IF } yY11 \text{ IS USER-SUPPLIED} \]
\[18 = 0 \text{ IF } yYM1 \text{ IS USER-SUPPLIED} \]

THEN ISLPW = \[11 + 2 \cdot 12 + 2 \cdot 13 + 2 \cdot 14 + 1 \cdot 15 + 2 \cdot 16 + 2 \cdot 17 + 1 \cdot 18 \]

THIS ISLPW = 0 INDICATES ALL DERIVATIVE INFORMATION IS 
USER-SUPPLIED AND ISLPW = 256 INDICATES NO DERIVATIVE 
INFORMATION IS USER-SUPPLIED. ANY VALUE BETWEEN THESE 
LIMITS IS VALID.

\(x1\) IS AN ARRAY OF AT LEAST \(2n + 1\) LOCATIONS.
\(xM\) IS AN ARRAY OF AT LEAST \(2n + 1\) LOCATIONS WHICH IS 
USED FOR SCRATCH STORAGE.

\(y1\) IS AN ARRAY OF AT LEAST \(2n + 1\) LOCATIONS.
\(yM\) IS AN ARRAY OF AT LEAST \(2n + 1\) LOCATIONS.

SIGMA CONTAINS THE TENSION FACTOR. THIS VALUE INDICATES 
THE CURVATURE DESIRED. IF \(\text{ANS}(\text{SIGMA})\) IS NEARLY ZERO 
\((\approx 0.001)\) THE RESULTING SURFACE IS APPROXIMATELY THE 
TENSOR PRODUCT OF CUBIC SPLINES, IF \(\text{ANS}(\text{SIGMA})\) IS LARGE 
\((\approx 50)\) THE RESULTING SURFACE IS APPROXIMATELY

\begin{figure}
\centering
\caption{A - 3 (Continued)}
\end{figure}
hi-linear. If sigma equals zero, tensor products of
Cubic splines result. A standard value for sigma is
approximately 1. in absolute value.

ON OUTPUT --

II CONTAINS THE VALUES OF THE XX-, YY-, AND XYXY-PARTIAL
DERIVATIVES OF THE SURFACE AT THE GIVEN NODES.

IERA CONTAINS AN ERROR FLAG.
   = 0 FOR NORMAL RETURN.
   = 1 IF N IS LESS THAN 2 OR V IS LESS THAN 2.
   = 2 IF THE X-VALUES OR Y-VALUES ARE NOT STRICTLY
     INCREASING.

AND

X, Y, Z, 17, ZX1, ZX2, ZY1, ZY2, ZYM, ZYV1, ZYV2, ZYVM.
ZXY1, ZXY2, ZXYM, ZSYM, AND SIGMA ARE UNALTEDRED.

THIS SUBROUTINE REFERENCES PACKAGE MODULES ,..., TERMS,
AND .

-------------------------------------------------------------------------

INTEGER N,M,17,15,15,SW
REAL X(N),Y(N),Z(N),XV(N),ZXY(N),ZYM(N),ZYM(N),ZYM(N),ZYM(N),
* Citation: X(N),ZXY(N),ZXY(N),ZXY(N),ZXY(N),ZXY(N),ZXY(N),
*      M N = M-1
*      MD1 = M+1
*      MD2 = M-1
*      MD3 = M+1
*      MDM = M+1
*      IER = 0
*      IF (N .LE. 1) GO TO 49
*      IF (Y(N) .LE. Y(1)) GO TO 47
*      SIGMA = ABS(SIGMA)*FLOAT(N-1)/(Y(N)-Y(1))
*      IF ((SIGMA/(2.*MDM)) .LE. (SIGMA/2)) GO TO 2
*      GO TO 1
1   Y(1,1,1) = Y(1,1)
   GO TO 5
2   DELY1 = Y(2,1,1)
   DELY2 = DELY1+DELV1
   IF (N .GT. 2) DELV2 = Y(3,1,1)
   IF (DELV1 .LE. 0 .AND. DELV2 .LE. DELY1) GO TO 47
   CALL CHR7 (DELY1,DELV2,SIGNA,C1,C2,C3,CM)

Figure A - 3 (Continued)
Figure A - 3 (Continued)
Figure A-3 (Continued)
Figure A - 3 (Continued)
Figure A - 3 (Continued)
FUNCTION SURFD (XX, YY, Z, Y, M, N, X, Y, Z, P, SIGMA)

C THIS FUNCTION INTERPOLATES A SURFACE AT A GIVEN COORDINATE
C PAIR USING A HI-SPLINE UNDER TENSION. THE SUBROUTINE SURF1
C SHOULD BE CALLED EARLIER TO DETERMINE CERTAIN NECESSARY
C PARAMETERS.

IN INPUT--

XX AND YY CONTAIN THE X- AND Y-COORDINATES OF THE POINT
TO BE MAPPED ONTO THE INTERPOLATING SURFACE.

M AND N CONTAIN THE NUMBER OF GRID LINES IN THE X- AND
Y-DIRECTIONS, RESPECTIVELY, OF THE RECTANGULAR GRID
WHICH SPECIFICATED THE SURFACE.

X AND Y ARE ARRAYS CONTAINING THE X- AND Y-GRID VALUES,
RESPECTIVELY, EACH IN INCREASING ORDER.

Z IS A MATRIX CONTAINING THE M X N FUNCTIONAL VALUES
CORRESPONDING TO THE GRID VALUES (I.E., Z(I,J) IS THE
SURFACE VALUE AT THE POINT (X(I),Y(J)) FOR I = 1,...,M
AND J = 1,...,N).

I$ CONTAINS THE ROW DIMENSION OF THE ARRAY Z AS DECLARED
IN THE CALLING PROGRAM.

2P IS AN ARRAY OF REMAIN LOCATIONS STORED WITH THE
VARIOUS SURFACE DERIVATIVE INFORMATION DETERMINED BY
SURF1.

AND

SIGMA CONTAINS THE TENSION FACTOR (ITS SIGN IS IGNORED).

THE PARAMETERS M, N, X, Y, Z, P, AND SIGMA SHOULD BE
INPUT UNALTEDERED FROM THE OUTPUT OF SURF1.

IN OUTPUT--

SURFD CONTAINS THE INTERPOLATED SURFACE VALUE.

ZY IS PARTIAL DERIVATIVE WITH RESPECT TO Y.

ZY IS PARTIAL DERIVATIVE WITH RESPECT TO Y.

NONE OF THE INPUT PARAMETERS ARE ALTERED.

THIS FUNCTION REFERENCES PACKAGE MODULES INTRUL AND

Figure A - 3 (Continued)
Figure A - 3 (Continued)
1 DFLR1 = (DFL + DFLS) / 2.
  DFLR2 = (DFL2 + DFLS) / 2.
  CALL SNHC5H (STMP1, DUMMY, SIGMAX, DFL, 1, 0)
  CALL SNHC5H (STMP2, DUMMY, SIGMAX, DFL2, 0)
  CALL SNHC5H (STMP1, DUMMY, SIGMAX, DFLS, -1)
  CALL SNHC5H (STMP2, DUMMY, SIGMAX, DFLS2, -1)
  CALL SNHC5H (DUMMY, DUMMY, SIGMAX, DFLP2, -1)
  CALL SNHC5H (DUMMY, DUMMY, SIGMAX, DFLP3, -1)

  7[M] = HERMN7 (7(1,-1,1), 7(1,1,1), 7(1,1,1), DFLR1, DFLR2).
  7I = HERMN7 (7(1,1,-1), 7(1,1,1), DFLR1, DFLR2).

  7XXY1 = HERMN7 (7(1,1,1), 7(1,1,2), 7(1,1,1), DFLR1, DFLR2).
  7XYI = HERMN7 (7(1,1,1), 7(1,1,1), DFLR1, DFLR2).

  7YY1M = HERMN7 (7(1,1,1), 7(1,1,1), 7(1,1,1), DFLR1, DFLR2).

  7XYXY1 = HERMN7 (7(1,1,1), 7(1,1,1), DFLR1, DFLR2).

2 DFL1 = XX-X(11).
  DFL2 = X(1)-X.<
  DFLS = X(1)-XX(11).
  IF (SIGMAX = M, 0.) THEN 3.
  SHER12 = HERMN7 (7(1,1), 7(1,1), 7XY1, 7XXY).
  7X = HERMY (7XY1, 11, 7XXY1, 7XXY).
  7Y = HERMY (7XY1, 11, 7XXY1, 7XXY).

RETURN 3.
DFLR1 = (DFL1 + DFLS) / 2.
  DFLR2 = (DFL2 + DFLS) / 2.
  CALL SNHC5H (STMP1, DUMMY, SIGMAX, DFL, 1, 0)
  CALL SNHC5H (STMP2, DUMMY, SIGMAX, DFL2, 0)
  CALL SNHC5H (STMP1, DUMMY, SIGMAX, DFLS, -1)
  CALL SNHC5H (STMP2, DUMMY, SIGMAX, DFLS2, -1)
  CALL SNHC5H (DUMMY, DUMMY, SIGMAX, DFLP2, -1)
  CALL SNHC5H (DUMMY, DUMMY, SIGMAX, DFLP3, -1)

  SHER12 = HERMN7 (7(1,1,1), 7(1,1,1), 7XXY1, SIGMAX).
  7X = HERMY (7(1,1,1), 7(1,1,1), 7XXY1, SIGMAX).
  7Y = HERMY (7(1,1,1), 7(1,1,1), 7XXY1, SIGMAX).

RETURN.

END.

Figure A-3 (Continued)
SUBROUTINE SINHCSH (SINH,M,COSH,X,ISW)
C
THIS SUBROUTINE RETURNS APPROXIMATIONS TO
SINH(X) = SINH(X)-X
COSH(X) = COSH(X)-1
AND
COSH(X) = COSH(X)-1+X*X/2
WITH RELATIVE ERRORS LESS THAN 3.42E-14

ON INPUT--
X CONTAINS THE VALUE OF THE INDEPENDENT VARIABLE.
ISW INDICATES THE FUNCTION DESIRED
= -1 IF ONLY SINH IS DESIRED,
= 0 IF BOTH SINH AND COSH ARE DESIRED,
= 1 IF ONLY COSH IS DESIRED,
= 2 IF ONLY COSH IS DESIRED AND SINH IS UNALTERED IF ISW = 0 OR ISW = 2.

ON OUTPUT--
SINH CONTAINS THE VALUE OF SINH(X) IF ISW > 0 OR ISW = 2 (SINH IS UNALTERED IF ISW = 0 OR ISW = 2).
COSH CONTAINS THE VALUE OF COSH(X) IF ISW > 0 OR ISW = 2 AND CONTAINS THE VALUE OF COSH(X) IF ISW > 0 AND COSH IS UNALTERED IF ISW = 0 OR ISW = 2.

AND
X AND ISW ARE UNALTERED.

_____________________________________________________________________
INTEGER ISW
REAL SINH,M,COSH,X
DATA $247/0.60776938-1.338E-13/,
  $6.72528442183206E-07/,
  $5.7257204-1.76191452E-06/,
  $0.1434144463.1406E-12/,
  $5.11-5.24-5443017.110E-06/,
DATA 0.241.7341356762981410E-7/,
  0.23/3.47777277751104E-15/,
  0.227.1515119602924-12/,
  0.217.8215119602924-12/.

Figure A - 3 (Continued)
* C01/7.515151067947F-03/
DATA Z13/5.5927116264720F-07/
  Z17/1,77743683000046F-04/
  Z19/1,69340061694702F-07/
  Z24/1,33412535492375F-09/
  Z33/5.8095934138643F-07/
  Z39/1,2791494440363F-04/
  Z01/1,635237143731E-07/
XX = X
AX = ABS(XX)
XS = XX#AX
IF ( (AX # GF, 2.7)) G0 TO 2
IF ( (ISW # FO, 1.15) # 0) G0 TO 2
IF ( (AX # GF, 1.15) # 0) G0 TO 1
SINH = (((SP4#XS#SP2)#XS#SP2)#XS+SPL1)#XS+1.0)#XS#XX)
  /((SP1#XS+1.0)Z6.1)
G0 TO 2
1 SINH = -(((1.#/EXPX+AX)+AX)-EXPX)/2.
IF (XX # LT, 0.0) SINH = -SINH
2 IF ( (ISW # BF, 0.1) # 0) G0 TO 4
IF ( (AX # GF, 1.15) # 0) G0 TO 4
COSH = (((((CP4#XS+CP2)#XS+CP2)#XS+CP1)#XS+1.0)#XS)
  /((((CP1#XS+1.0)Z7.0)
G0 TO 4
3 COSH = (((1.#/EXPX-2.0)+EXPX)/2.
4 IF ( (ISW # LF, 1.0) # 0) RETURN
IF ( (AX # GF, 2.0) # 0) G0 TO 5
COSH = ((((((7.0#XS+7.0)#XS+7.0)#XS+7.0)#XS+7.0)#XS+7.0)#7.0)
  RETURN
5 COSH = (((1.#/EXPX-2.0)-XS)+EXPX)/2.
RETURN
END
SUBROUTINE TERMS (ALPHA, SOH, SIGMA, DEL)
C
C THIS SUBROUTINE COMPUTES THE DIAGONAL AND SUPERDIAGONAL
C TERMS OF THE TRIDIAGONAL LINEAR SYSTEM ASSOCIATED WITH
C SPLINE UNDER TENSION INTERPOLATION.
C
C IF INPUT--
C
C SIGMA CONTAINS THE TENSION FACTOR.
C
C IN,
C
Figure A - 3 (Continued)
C

DEFL CONTAINS THE STEP SIZE.
C

IN OUTPUT--
C

SIGMA*DEFL*COSH(SIGMA*DEFL) - SINH(SIGMA*DEFL)
C

DIAG = DEFL-------------------------------
C

(SIGMA*DEFL)**2 + SINH(SIGMA*DEFL)
C

SIGMA*DEFL - SIGMA*DEFL
C

S(l) = DEFL------------------------
C

(SIGMA*DEFL)**2 * SINH(SIGMA*DEFL)
C

AND
C

SIGMA AND DEFL ARE UNALTEDED.
C

THIS SUBROUTINE REFERENCES PACKAGE MODULE SNHCSH.
C

REAL DIAG,S(l)SIGMA,DEFL
IF (SIGMA .LE. 0.) GO TO 1
DIAG = DEFL/2.
S(l) = DEFL/6.
RETURN
1 SIGDEF = SIGMA*DEFL
CALL SNHCSH (SINH,COSH,SIGMA,DEFL,0)
DENOM = DEFL/((SINH+SIGMA)*SIGMA+SIGMA)
DIAG = DENOM*(SINH+COSH-SINH)
S(l) = DENOM*S(l)
RETURN
END

* SUBROUTINE CELL (DEFL,DEFL2,SIGMA,C1,C2,C3,M)
C

THIS SUBROUTINE DETERMINES THE COEFFICIENTS C1, C2, AND C3
C

USED TO DETERMINE ENHANCED SLOPES, SPECIFICALLY, IF
C

FUNCTION VALUES Y1, Y2, AND Y3 ARE GIVEN AT POINTS X1, X2,
C

AND X3, RESPECTIVELY, THE QUANTITY C1X1 + C2X2 + C3X3
C

IS THE VALUE OF THE DERIVATIVE AT X1 OF A SPLINE UNDER
C

TENSION (WITH TENSION FACTOR SIGMA) PASSING THROUGH THE
C

THREE POINTS AND HAVING THIRD DERIVATIVE EQUAL TO ZERO AT
C

X1. Optionally, only two values, C1 AND C2 ARE DETERMINED.
C

ON INPUT--
C

DEFL IS X2-X1 (GT. 0.).
C

Figure A - 3 (Continued)
DFL2 IS X3-X1 (G1 X2). IF N .EQ. 7, THIS PARAMETER IS
IGNORER.

SIGMA IS THE TENSION FACTOR.

AND

N IS A SWITCH INDICATING THE NUMBER OF COEFFICIENTS TO
BE RETURNED. IF N .EQ. 2 ONLY TWO COEFFICIENTS ARE
RETURNED, OTHERWISE ALL THREE ARE RETURNED.

ON OUTPUT--

c1, c2, AND c3 CONTAIN THE COEFFICIENTS.

NONE OF THE INPUT PARAMETERS ARE ALTERED.

THIS SUBROUTINE REFERENCES PACKAGE MODULE SNHCSH.

REAL DFL1,DFL2,SIGMA,C1,C2,C3
IF (N .EQ. 2) GO TO 2
IF (SIGMA .NE. 0.0) GO TO 1
DFL = DFL2-DFL1
C1 = -(DFL1+DFL2)/(DFL1*DFL2)
C2 = DFL2/(DFL1*DFL1)
C3 = -DFL1/(DFL2*DFL1)
RETURN
1 CALL SNHCSH (DUMMY,COSH1,SIGMA*DFL1,1)
CALL SNHCSH (DUMMY,COSH2,SIGMA*DFL2,1)
DFLP = SIGMA*(DFL2+DFL1)/2.
DFLM = SIGMA*(DFL2-DFL1)/2.
CALL SNHCSH (SINHMP,DUMMY,DFLP,-1)
CALL SNHCSH (SINHSM,DUMMY,DFLM,-1)
DENOM = COSH1*(DFL2-DFL1)*2.*DFL1*(SINHMP+DFL1)*
       (SINHSM+DFL1)
C1 = 2.*((SINHMP*DFLP)/(SINHSM+DFLM)/DENOM
C2 = -COSH2/DENUM
C3 = COSH1/DENUM
RETURN
2 C1 = -DFL1
C2 = -C1
RETURN
END

FUNCTION INTRVL (T,Y,N)

Figure A-3 (Continued)
C THIS FUNCTION DETERMINES THE INDEX OF THE INTERVAL
C (DETERMINED BY A GIVEN INCREASING SEQUENCE) IN WHICH
C A GIVEN VALUE LIES.
C
C ON INPUT--
C
C  T IS THE GIVEN VALUE.
C  X IS A VECTOR OF STRICTLY INCREASING VALUES.
C  AND
C
C  N IS THE LENGTH OF X (N ≥ 2).
C
C ON OUTPUT--
C
C  INTERVAL RETURNS AN INTEGER I SUCH THAT
C
C  1 ≤ I ≤ N-1 IF T ≤ X(I) ≤ X(I+1) +
C  I = N-1 IF X(N-1) ≤ T
C  OTHERWISE X(I) ≤ T ≤ X(I+1).
C
C MORE OF THE INPUT PARAMETERS ARE ALTERED.
C
C------------------------------------------------------------------
C INTEGER N
C REAL T, X(N)
C IF T = 1
C IF (T ≥ X(I)) GO TO 4
C IF (T ≤ X(I-1)) GO TO 6
C I = 2
C I = N-1
C I = I + 1
C END = I + 1
C GO TO 2
C T I = 1
C GO TO 1
C IF I(N) = 1
C "ERROR"
C IF I(N) = 1
C "ERROR"
C IF I(N) = N-1
C "ERROR"
C END.
C

Figure A - 3 (Continued)