Bellcomm

955 L'Enfant Plaza North, S.W.
Washington, D. C. 20024

date: August 5, 1971

to: Distribution

from: J. W. Powers

B71 08006

subject: Incorporation of Figure Intersection Determination Capability into the MSFC RAVFAC and VUFACT View Factor Computer Programs - Case 620

ABSTRACT

The MSFC view factor computer programs RAVFAC and VUFACT have been and are being much used in Skylab thermal analysis. Bellcomm has developed the subroutine EANCAL which increases the utility of these two programs by eliminating the need for any preliminary calculation of input parameters. The usefulness of these two MSFC programs has been further increased by incorporation of figure intersection and angular orientation determination capabilities in EANCAL.

In general, when a figure or the plane of the figure intersects another figure, different view factors exist from either surface of the intersecting figure to the intersected figure. For figures in general spatial orientations, the occurrence of intersections and the consequent existence of other view factors may not be readily detected from the figure definition geometry. EANCAL now provides a diagnostic message which alerts the program user when figures intersect and other view factors thus exist. Output messages specifying the angular orientations of all pairs of non-spherical figures with each other are also provided as an input data check.

Analysis developing the figure intersection and angular orientation determination capabilities of EANCAL for the various figure options in the two programs is presented. The use of EANCAL with these capabilities is explained and a subroutine listing is provided.

(NASA-CR-121313) INCORPORATION OF FIGURE INTERSECTION DETERMINATION CAPABILITY INTO THE MSFC RAVFAC AND VUFACT VIEW FACTOR COMPUTER PROGRAMS (Bellcomm, Inc.) 27 p

N79-73162

Unclas

00/61 12087

FF No. [redacted] (NASA CR OR TMX OR AD NUMBER) (CATEGORY)

SEP 1971
RECEIVED
NASA STI FACILITY
INPUT BRANCH

Bellcomm

955 L'Enfant Plaza North, S.W.
Washington, D. C. 20024

date: August 5, 1971

to: Distribution

from: J. W. Powers

B71 08006

subject: Incorporation of Figure Intersection Determination Capability into the MSFC RAVFAC and VUFACT View Factor Computer Programs - Case 620

MEMORANDUM FOR FILE

1.0 INTRODUCTION

Subroutine EANCAL (Euler ANgle CALculation) was developed by Bellcomm to facilitate use of the MSFC RAVFAC[1] and VUFACT[2] view factor computer programs which are much used in Skylab thermal analysis.[3],[4] EANCAL performs all the prerequisite calculations to prepare input for figures in any general spatial orientation. The versatility of EANCAL has been enhanced by the development of figure intersection determination and figure orientation determination capabilities. This memorandum describes the need for figure intersection determination capability in view factor programs and presents the analysis for this determination in RAVFAC and VUFACT. The angular orientation determination capability between all pairs of non-spherical surfaces is also developed as a check on EANCAL input. A listing of EANCAL with these capabilities is also provided.

1.1 COORDINATE SYSTEMS

The surface options available in RAVFAC and VUFACT and the corresponding EANCAL defining points are shown in Figures 1, 2 and 3. With subroutine EANCAL these surfaces are defined in some common central Cartesian coordinate system (CCS) by from three to six points. Selection of the CCS is by the user based upon how the surface elements of a particular configuration are most simply defined.

EANCAL defines a local surface coordinate system (SCS) as required for the two programs from the defining points for each surface type (Figures 1, 2 and 3). The various SCS and the listed equations which define the SCS parameters in terms of the CCS points are in general unnecessary for the use of EANCAL. These equations, which were developed in Reference (3), are required in the figure intersection determination analysis.

2.0 NEED FOR FIGURE INTERSECTION DETERMINATION CAPABILITY

In general when a figure or the plane of the figure intersects another figure, different view factors exist from either surface of the intersecting figure to the intersected figure. For figures in general spatial orientations, the occurrence of intersections and the consequent existence of other view factors may not be readily detected from the figure definition geometry. If the computer program can alert the user by an appropriate output message when other view factors are possible, spatial comprehension will be enhanced and fewer errors will result. For less involved configurations when it is known that intersections occur or not, the presence or absence of the message serves as an input data check.

With EANCAL the geometric visualization and calculation required to prepare Euler angle input for RAVFAC and VUFACT is eliminated. The ability to visualize the directions of normals to plane figures, however, is still a prerequisite. In general, if the incorrect active face direction of a figure is specified, either no view factor or an incorrect view factor will result.

3.0 SURFACE ACTIVE FACE DEFINITION

Program input ILK defines both the figure type and the active surface normal direction. ILK is a different integer for each figure type prefixed by +/- for the active normal direction. This input presents no visualization problem for three dimensional surfaces of revolution since +/- ILK indicates that the outside/inside surface is active. For plane figures +/- ILK indicates that the active surface is in the direction of the +/- Z SCS axis. When + ILK is called with the rectangle or triangle/trapezoid, EANCAL positions the SCS + Z-axis in the direction of a right hand rotation with respect to the figure vertices selected.

When the plane of a figure does not intersect the boundary of another figure, a sign error in ILK from faulty geometric visualization presents no problem other than having to rerun the program with an input change. If the plane of the figure does intersect another figure's boundary, however, an erroneous view factor will probably result from an erroneous ILK sign.

Consider two plane figures of Area A_1 and A_2 oriented such that the plane of A_1 intersects figure A_2. Let the line of intersection divide A_2 into segments of area A_3 and A_4. The sign of ILK for figure A_1 can in general cause either of two different view factors to be calculated by the program. For one sign of ILK the view factor from A_1 to A_3 is determined, while the other sign of ILK yields the view factor from A_1 to A_4.

4.0 INTERSECTION ANALYSIS

To determine if intersections occur within any figure boundaries, the planes of all planar figures (j-systems) are established in the SCS of all the other figures (i-systems). These j-system planes are established by determining their intercepts on the axes of the i-system figure SCS. An intersection of interest occurs if the j-system plane and the i-system figure equations have a real solution within the i-system surface boundary. If two planar figures are considered and if the j-system plane does not intersect the i-system surface, the possible intersection of the i-system plane with the j-system surface must also be investigated.

Only the intersections of plane figures with other figures will be considered. There should be minimum confusion with intersections of three dimensional figures because +/-ILK refer to the outside/inside surfaces.

4.1 NUMBER OF INTERSECTION AND ANGULAR ORIENTATION DETERMINATIONS

If a particular geometric configuration consists of a large number of figure elements including planar surfaces, many determinations for possible intersections will be made. This section considers the number of figure intersection tests and the number of figure orientation determinations made by EANCAL.

In a geometric system of N figures of which M are plane figures ($2 \leq M \leq N$), the number of pairs of figures to be considered for possible intersections is $M(N-1)-Q$. Integer Q is ≥ 0 depending upon both the number of intersections between plane figures of a system and at what phase of the search these intersections are found. If the planes of surfaces 1 and 2 of the M plane surfaces in a geometric system intersect surface M in the first tests conducted, the plane of surface M cannot intersect surfaces 1 and 2, thus Q=2. If the tests to determine if the plane of surface M

intersects surfaces 1 and 2 were conducted first, no intersections would be found and the tests to determine if the planes of surfaces 1 and 2 intersect surface M would have to be conducted, thus Q=0.

Messages specifying the angular orientations of all pairs of figures with respect to each other excluding spherical surfaces are also provided to facilitate checking figure definition input. The total number of these messages in a system of N figures of which S are spherical surfaces is (N-S)(N-S-1)/2.

4.2 DETERMINATION OF THE j-SYSTEM PLANE INTERCEPTS IN THE i-SYSTEM

All figures are defined by from three to six points P_g which have CCS rectangular coordinates X_g, Y_g, Z_g and a position vector $\bar{r}_g$ from the CCS origin to the point. For a particular figure pair consisting of i and j-system figures, the coordinates of the j-system figure are referred to the i-system SCS by

$$\begin{pmatrix} X_i \\ Y_i \\ Z_i \end{pmatrix} = T_i \begin{pmatrix} X_j - X_1 \\ Y_j - Y_1 \\ Z_j - Z_1 \end{pmatrix} \quad , \quad (1)$$

T_i is (1) of Reference (3) using the direction cosines for the particular type of i-system figure and X_j, Y_j, Z_j are the CCS coordinates of three points defining the j-system figure plane. T_i is the coordinate transformation for each i-system figure which rotates the CCS into angular correspondence with each respective SCS via the required Euler angle sequence and directions. X_1, Y_1, Z_1 are the CCS coordinates of P_1, the origin of each respective i-system figure to be tested for intersection with all the plane figures of the system. In the following analysis, all coordinate subscripts refer to j-system CCS points in the i-system SCS after transformation by (1).

The j-system plane defined by points $P_{1,2,3}$ which are not colinear is

$$\begin{vmatrix} X & Y & Z & 1 \\ X_1 & Y_1 & Z_1 & 1 \\ X_2 & Y_2 & Z_2 & 1 \\ X_3 & Y_3 & Z_3 & 1 \end{vmatrix} = 0$$

After expansion by minors and co-factors,

$$X\begin{vmatrix} Y_1 & Z_1 & 1 \\ Y_2 & Z_2 & 1 \\ Y_3 & Z_3 & 1 \end{vmatrix} + Y\begin{vmatrix} Z_1 & X_1 & 1 \\ Z_2 & X_2 & 1 \\ Z_3 & X_3 & 1 \end{vmatrix} + Z\begin{vmatrix} X_1 & Y_1 & 1 \\ X_2 & Y_2 & 1 \\ X_3 & Y_3 & 1 \end{vmatrix} = \begin{vmatrix} X_1 & Y_1 & Z_1 \\ X_2 & Y_2 & Z_2 \\ X_3 & Y_3 & Z_3 \end{vmatrix}, \quad (2)$$

If A,B,C and D are substituted for the determinants in (2), the equation of the j-system plane in the i-system is

$$AX + BY + CZ = D, \quad (2A)$$

The intercept form of the equation of a plane is

$$X/a + Y/b + Z/c = 1, \quad (3)$$

where the intercepts are determined from the j-system coordinates after transformation by (1). If the corresponding coefficients of (2A) and (3) are compared, the intercepts a,b, and c, of the j-system plane on the SCS i-system X,Y, and Z axes are

$$a=D/A, \; b=D/B, \; c=C/D$$

Special cases occur when the j-system plane is either perpendicular to a SCS i-system axis or is parallel to a SCS axis. These special cases occur when any one or any two of the j-system plane intercepts are infinite, which are indicated by the respective X,Y, and Z determinant coefficients in (2A) being zero. If one intercept is infinite, the j-system plane is parallel to the SCS-axis corresponding to the infinite intercept. If two intercepts are infinite, the j-system plane is perpendicular to the SCS i-system axis corresponding to the finite intercept.

A special case of particular interest occurs when the i-system figure is also a plane surface and is coplanar with the j-system figure. For this geometry with both plane figures in the i-system SCS X-Y plane, A=B=D=o, and no view factors exist between the two coplanar figures.

Additional special cases occur when one or two of the determinant coefficients A,B and C are zero and coefficient D is also zero. The following tabulation of all j-system plane orientations with respect to the i-system SCS relates position and finite intercepts to zero and non-zero magnitudes of the four determinant coefficients of (2A).

Determinant coefficients	j-system plane orientation in i-system SCS	Finite SCS axes intercepts
A,B,C and D $\neq$ O	general orientation	a,b,c
B and C = O	perpendicular to X_i-axis	a
A and C = O	perpendicular to Y_i-axis	b
A and B = O	perpendicular to Z_i-axis	c
A = O	parallel to X_i-axis	b,c
B = O	parallel to Y_i-axis	a,c
C = O	parallel to Z_i axis	a,b
B,C and D = O	Y_i-Z_i plane	a = o
A,C and D = O	X_i-Z_i plane	b = o
A,B and D = O	X_i-Y_i plane	c = o
A and D = O	contains X_i-axis	b = c = o
B and D = O	contains Y_i-axis	a = c = o
C and D = O	contains Z_i-axis	a = b = o

In EANCAL the four determinants of (2A) are calculated and one of the 13 cases listed above is established. Since the intersection geometry is most involved for the general case of all non-zero determinants, only this analysis is presented. The other special case results are readily obtained from the general analysis.

In (2) the CCS points $P_{1,2,3}$ shown in Figure 1 define the plane of the triangle/trapezoid. For the rectangle and the disc, also shown in the same figure, the corresponding points which define the two figure planes are $P_{1,2,4}$ and $P_{1,6,3}$. Disc point P_6, on the outside diameter at the SCS X-axis is not an EANCAL input but is used to define a third point in the disc plane for use in (2). The coordinates of P_6 in terms of the disc input CCS coordinates are calculated in EANCAL.

4.3 PLANE FIGURE INTERSECTIONS

If the surface to be tested for intersection is a plane figure, it lies in the SCS i-system X-Y plane (Figure 1). In this plane the trace of the j-system plane is the line $X/a + Y/b=1$. Intersection occurs when the simultaneous solution of this line and the i-system defining equation yields roots on the i-system figure boundary. The intersection criteria are determined from the SCS parameters in terms of the CCS coordinates and the j-system plane intercepts. The SCS parameters in Figures 1,2 and 3 are surface dimensions and angles. Since the magnitudes of these are invariant with coordinate system transformations, the i-figure CCS coordinates which define the SCS parameters do not require transformation.

4.3.1 RECTANGLE

The rectangle is defined by the boundaries $X_i=\gamma_{max}, Y_i=\beta_{max}$ and the SCS X_i and Y_i axes. No intersection occurs if a and b are both <0. Intersection will occur for any of the following conditions

$$\gamma_{max}>a>o, \text{where } \gamma_{max}=|\bar{r}_2-\bar{r}_1|$$

$$\beta_{max}>b>o, \text{where } \beta_{max}=|\bar{r}_4-\bar{r}_1|$$

$$\beta_{max}>b\left(1-\frac{\gamma_{max}}{a}\right)>o$$

4.3.2 DISC

The distance from the i-system origin to the line of the j-system plane in the i-system X-Y plane is $ab/\sqrt{a^2+b^2}$. Since the disc radius is β_{max}, the condition for intersection with a complete disc is

$$\beta_{max}>|ab|/\sqrt{a^2+b^2}, \text{ where } \beta_{max}=|\bar{r}_3-\bar{r}_1|$$

If the disc is a partial surface of revolution ($\gamma_{max}<360°$, Figure 1), the intersection determination will be for the complete figure as discussed in Section 4.4.

4.3.3 TRIANGLE/TRAPEZOID

For this class of figures the line of intersection of the i and j-system X-Y planes in the i-system is solved with the equations of the lines P_1P_2 and P_1P_3. These lines, which define the figure sides, intersect at P_1, the SCS origin:

P_1P_2, $Y=m_2X$, where $m_2=Y_2/X_2 = 1/\tan\gamma_{max}$

P_1P_3, $Y=m_3X$, where $m_3=Y_3/X_3 = 1/\tan\gamma_{min}$

Solving each of the line equations with the trace of the j-system plane in the X_i-Y_i plane yields

$$Y_i = ab/(b\tan\gamma_{max,\ min}+a),$$

where γ_{max} and γ_{min} refer respectively to lines P_1P_2 and P_1P_3. Intersection of the triangle/trapezoid and the j-system plane occurs with either of the two following conditions

$$\beta_{max} > ab/(b\ \tan\gamma_{max,min}+a) \quad \begin{array}{l} >\beta_{min},\ \text{trapezoid} \\ >o,\ \text{triangle} \end{array}$$

4.4 THREE-DIMENSIONAL FIGURE INTERSECTIONS

The analysis for intersections of the j-system plane with three-dimensional rather than planar i-system figures is more involved because intersections are not limited to the X_i-Y_i plane. In this analysis any i-system figures that are surfaces of revolution are assumed to be complete figures. If the i-system figure is a partial surface of revolution, the possible intersection will be determined as for a complete surface of revolution. When the partial surface is not intersected but the total surface is, the program yields the intersection message. This case can be detected by rerunning the example with the other ILK value for the planar figure, which will yield no view factor.

The distance ρ from the SCS i-system origin to the j-system plane in the i-system with intercepts a,b and c is

$$\rho = |abc| \Big/ \sqrt{(ab)^2+(ac)^2+(bc)^2}$$

Consider the plane defined by ρ and the SCS Z_i-axis. Let the intersection of this plane and the SCS X_i-Y_i plane define the X* direction.† The trace of the j-system plane in the X^*-Z_i plane is the line

$$\left(\frac{\frac{X^*}{ab}}{\sqrt{a^2+b^2}}\right) + \frac{Z_i}{c} = 1, \tag{4}$$

Intersections of the j-system plane with three-dimensional figures exist when (4) and the equation of the trace of the figure of interest in the X^*-Z_i plane yield real roots which are on the i-system surface boundary.

4.4.1 SPHERE

Intersection of this figure and the plane occurs when the sphere radius is greater than the distance ρ from the i-system origin to the j-system plane in the i-system. Since the radius is $|\bar{r}_2-\bar{r}_1| = \alpha$, the condition for intersection with a complete sphere is $\alpha > \rho$.

4.4.2 RIGHT CIRCULAR CYLINDER, RIGHT CIRCULAR CONE AND RIGHT CIRCULAR PARABOLOID

These three i-system surfaces of revolution are conveniently considered together because their axes are the SCS Z_i-axis. For altitude H and base radius R, the traces of the surfaces in the X^*-Z_i plane for $X^*>0$ are

$$\begin{aligned} X^* &= R, && \text{cylinder,} \\ Z_i &= HX^*/R, && \text{cone} \\ Z_i &= HX^{*2}/R^2, && \text{paraboloid} \end{aligned} \tag{5}$$

† The j-system plane and the X^*-Z_i plane geometry in the SCS i-system are shown in Figure 4.

An intersection of the j-system plane with any of (5) will occur when $o<c<H/(1-RV)$, where $V = \sqrt{a^2+b^2}/|ab|$. For values of $c<o$, possible intersections are determined from the simultaneous solution of (4) and (5) for Z_i or X^*. If the resultant Z_i is $<H$ or X^* is $<R$, intersection of the j-system plane with the i-system complete surface of revolution occurs. The conditions for intersection of the j-system plane with the respective three-dimensional figures of revolution in terms of the figure dimensions and plane intercepts are

$$1/V<R, \quad \text{cylinder}$$

$$o<[1/(1/c+RV/H)]< H, \quad \text{cone}$$

$$[-\frac{cVR^2}{2H}(1-\sqrt{1+4H/cV^2R^2})]< R, \quad \text{paraboloid}$$

where

$$H = |\bar{r}_2-\bar{r}_1|$$

$$R = |\bar{r}_3-\bar{r}_1|, \quad \text{cylinder}$$

$$R = |r_3-r_2|, \quad \text{cone and paraboloid}$$

4.4.3 RECTANGULAR PRISM

The geometry for intersection of this figure is more involved than for the other three-dimensional figures because it is not symmetrically positioned with respect to the Z_i axis. As in Section 4.4.2, the cross section of the figure and the trace of the j-system plane are considered in the X^*-Z_i plane. The j-plane intersection of the prism, however, is not limited to the cross section of the figure in the X^*-Z_i plane. The intersection may also occur along the projected view of the figure as observed normal to the X^*-Z_i plane. This triangle shaped intersection, not in

the X^*-Z_i plane, is shown by points $P_{I,J,K}$ in Figure 5. The geometry for intersection determination when a, b and c are all >0 is shown in Figure 5. The two parts of this figure show true views respectively of the X_i-Y_i and X^*-Z_i planes relative to the rectangular prism orientation.

The projected figure dimension d measured from the SCS origin in the $+X^*$ direction in (4) determines the Z_i which establishes if intersection occurs.

$$d = (a\beta_{max} + b\gamma_{max}) \Big/ \sqrt{a^2 + b^2}$$

The criterion for intersection referred to the figure altitude (α) for a, b, and c all >0 is $\alpha > Z_i > 0$ which in terms of the parameters of the system is

$$\alpha > c\left(1 - \frac{\gamma_{max}}{a} - \frac{\beta_{max}}{b}\right) > 0$$

γ_{max} and β_{max} are defined in terms of the CCS coordinates as for the rectangle and $\alpha = \bar{r}_5 - \bar{r}_1$.

The criteria for intersection when not all of the j-plane intercepts are >0 must also be considered. The following table tabulates the intersection criteria for the rectangular prism with the eight possible sign combinations with non-zero, finite intercepts.

a	+	-	-	+	+	-
b	-	+	-	+	+	-
c	+ or -	+ or -	-	-	+	+
Criteria for Inter-section	$\gamma_{max} > a$	$\beta_{max} > b$	no inter-section	$d > \frac{ab}{\sqrt{a^2+b^2}}$	$a > c\left(1 - \frac{\gamma_{max}}{a} - \frac{\beta_{max}}{b}\right)$	$\alpha > c$

The first two criteria columns are those for the intersection of the j-system plane with an i-system rectangle, (the rectangular prism lower base) for b or a <0.

4.5 EANCAL INPUT

Input to RAVFAC and VUFACT using EANCAL with the figure intersection determination capability is as described in Section 3.0 of Reference 4, except that the total number of figures in a given configuration must be specified. Integer ITNS is the total number of figures in the configuration and is added to the second surface input card of the first input figure. ITNS is right adjusted in columns 18-20 following program variable N.

4.6 FIGURE INTERSECTION MESSAGE

If any surface in a particular geometric configuration is intersected by the plane of a planar figure, the following message will be printed in the input data summary of the last figure in the configuration.

@@@ THE PLANE OF SURFACE j INTERSECTS SURFACE i.

@@@ ANOTHER VIEW FACTOR FROM SURFACE j to SURFACE i MAY BE OBTAINED BY USING THE OTHER VALUE OF ILK WITH SURFACE j.

In the above message i and j are the surface identification numbers (ISF) assigned to the respective figures involved. Only plane figures will occur as j while all types of surfaces can occur as i. If more than one intersection occurs, the message will be repeated in the last figure's data summary for all pairs of figures involved.

5.0 ORIENTATION OF FIGURES WITH RESPECT TO EACH OTHER

With EANCAL the need for spatial visualization to determine Euler angles and other input parameters is eliminated. As discussed in Sections 2.0 and 3.0, the ability to visualize the active face direction when one figure or its plane intersects another figure is a prerequisite to obtaining correct view factors both with and without EANCAL. Obtaining correct view factors with EANCAL is strongly dependent upon inputing the correct figure coordinates. For a configuration geometry with many elements, there will be a large number of coordinates with many attendant chances for input error.

To further facilitate visualization of figure elements and to help check program input, EANCAL also yields several

different kinds of figure orientation messages. These messages occur independently of the intersection message of section 4.6 and are described in the following sections.

5.1 ORIENTATIONS OF PLANE FIGURES WITH RESPECT TO OTHER PLANE FIGURES

With an i-system plane figure, zero values of one or two of the coefficients A,B, or C of the j-system plane indicate either parallism or perpendicularity between the particular pair of plane figure elements. If A,B, and C are all finite and $\neq 0$, the i and j-system planes are skewed.

If the planes of the i and j-systems are skewed, knowing the included angle may also be of value to check the program input. The angle between the active faces of two planar figures is found by considering the determinant coefficients of (2A). The unit normal vector $\bar{M}$ to the j-system plane active face for ILK in terms of these coefficients is

$$\bar{M} = (A\bar{i} + B\bar{j} + C\bar{k})/\sqrt{A^2+B^2+C^2}$$

The unit normal to the i-system plane active face in its SCS for +ILK is along the Z_i-axis. Since the angle between the planes is equal to the angle θ, between their respective active face normals,

$$\theta = \cos^{-1} (C/\sqrt{A^2+B^2+C^2})$$

For j-figures with intersecting planes and -ILK values, the angle between the normals to the active faces of the two planes is $180-\theta$.

For configurations with two or more plane figures, the following message will be listed for each pair of plane figures in the input data summary of the last figure in the configuration.

*** THE ANGLE BETWEEN THE ACTIVE FACES OF THE
PLANES OF FIGURES _____ AND _____ IS _____
DEG. FOR +ILK VALUES.

5.2 ORIENTATIONS OF NON-PLANAR FIGURES

The axes of the cylinder, cone and parabloid and one of the prism edges are specified by their respective CCS vectors $\bar{r}_2-\bar{r}_1$. If a geometric system of interest contains figures u and v of these types, the included angle, θ, between their respective axes, axis and edge, or edges is

$$\theta = \cos^{-1}\left[\frac{(X_2-X_1)_u(X_2-X_1)_v+(Y_2-Y_1)_u(Y_2-Y_1)_v+(Z_2-Z_1)_u(Z_2-Z_1)_v}{\sqrt{(X_2-X_1)_u^2+(Y_2-Y_1)_u^2+(Z_2-Z_1)_u^2}\sqrt{(X_2-X_1)_v^2+(Y_2-Y_1)_v^2+(Z_2-Z_1)_v^2}}\right],$$

where CCS coordinates are used for both figures.

For configurations with two or more non-spherical three-dimensional figures, one of the following messages that is appropriate will be listed for each pair of figures in the input data summary of the last figure in the configuration.

*** THE ANGLE BETWEEN THE AXES OF FIGURES _____ AND _____ IS _____ DEG.

*** THE ANGLE BETWEEN THE AXIS OF FIGURE _____ AND THE EDGE OF PRISM _____ DEFINED BY POINTS P1 AND P2 _____ DEG.

*** THE ANGLE BETWEEN THE EDGES OF PRISMS _____ AND _____ WHICH ARE DEFINED BY THEIR RESPECTIVE POINTS P1 AND P2 IS _____ DEG.

5.3 ORIENTATIONS OF PLANE FIGURES WITH NON-PLANAR FIGURES

If one of the figures is a plane figure and the other is neither a plane figure nor a sphere, the angle, θ, between the unit normal $\bar{M}$ to the plane active face for +ILK and the other figure axis or edge is as listed in Section 5.1 This is a consequence of the figure axis or edge P_1P_5 being along the Z_i SCS axis of the non-planar figure of the pair. If -ILK is used for the plane figure, the corresponding angle is 180-θ.

The appropriate one of the following messages will be listed in the input data summary of the last figure in the configuration.

*** THE ANGLE BETWEEN THE ACTIVE FACE OF THE
PLANE OF FIGURE _____ FOR +ILK AND THE
AXIS OF FIGURE _____ IS _____ DEG.

*** THE ANGLE BETWEEN THE ACTIVE FACE OF THE
PLANE OF FIGURE _____ FOR +ILK AND THE
EDGE OF PRISM _____ DEFINED BY POINTS P1
AND P5 IS _____ DEG.

6.0 EANCAL SUBROUTINE LISTING

Figure 6 is a listing of EANCAL with the intersection determination and orientation determination capabilities.

7.0 ACKNOWLEDGEMENT

Miss L. K. Hawkins incorporated the figure intersection and figure orientation determination capabilities into EANCAL. Her programming of these additional capabilities is appreciated.

J. W. Powers

1022-JWP-tla

J. W. Powers

Attachments
Figures 1 - 6

REFERENCES

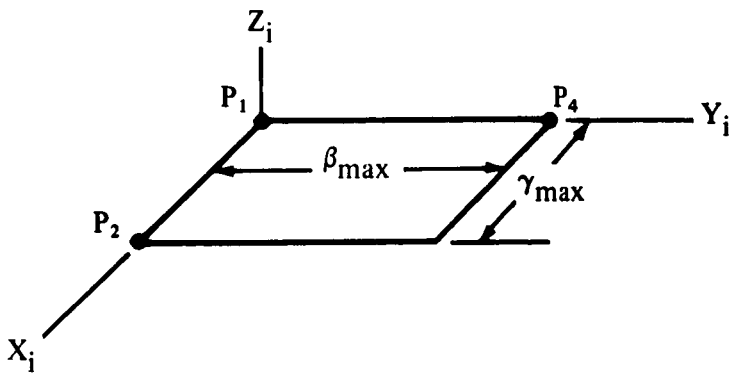
1. User's Manual for 'RAVFAC', A Radiation View Factor Digital Computer Program, Lovin, J. K., Lubkowitz, A. W., Lockheed Missiles and Space Co., NASA Contractor Report CR-6132, Nov. 1969.

2. User's Manual Lockheed Orbital Heat Rate Package (LOHARP), Lovin, J. K., Spradley, L. W., Lockheed Missiles & Space Co., Huntsville Research & Engineering Center, Huntsville Research Park, Ala., April 1970.

3. Powers, J. W., Analysis to Facilitate Use of the MSFC RAVFAC and VUFACT View Factor Programs, Bellcomm Memorandum for File, March 31, 1971 - Case 620.

4. Hawkins, L. K., Powers, J. W., Subroutine EANCAL to Facilitate Use of the MSFC RAVFAC and VUFACT View Factor Computer Programs, Bellcomm Memorandum for File, June 18, 1971 - Case 620.

Surface Type ± 1, Rectangle

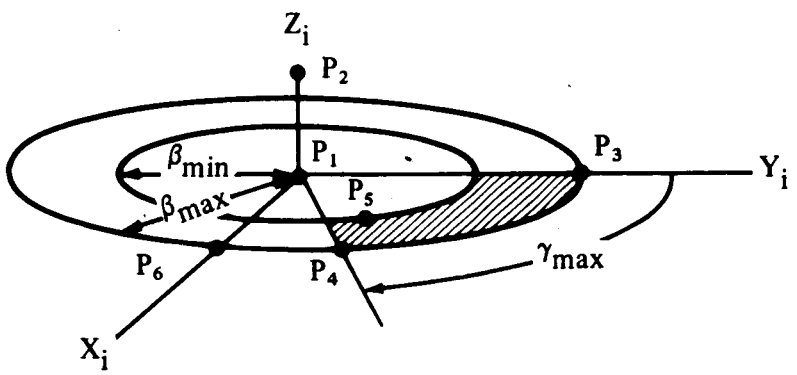


$\alpha = \beta_{min} = \gamma_{min} = 0$

$\beta_{max} = |\bar{r}_4 - \bar{r}_1|$

$\gamma_{max} = |\bar{r}_2 - \bar{r}_1|$

Surface Type ± 2, Disc



$\alpha = \gamma_{min} = 0$

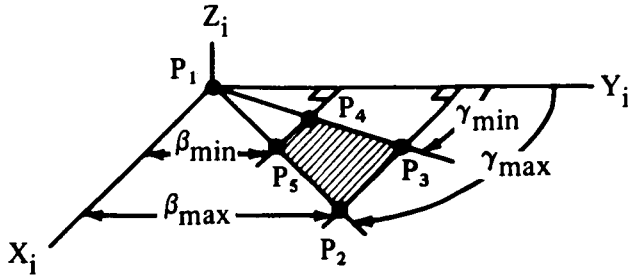
$\beta_{min} = |\bar{r}_5 - \bar{r}_1|$ or 0

$\beta_{max} = |\bar{r}_3 - \bar{r}_1|$

$\gamma_{max} = 2\sin^{-1}(|\bar{r}_4 - \bar{r}_3|/2|\bar{r}_3 - \bar{r}_1|)$

$0 < \gamma_{max} \leqslant 360°$

Surface Type ± 3, Trapezoid and Triangle



$\alpha = 0$

$\beta_{min} = 0$, Triangle

$\beta_{min} = |\bar{r}_4 - \bar{r}_1|\sin\theta_3$, Trapezoid

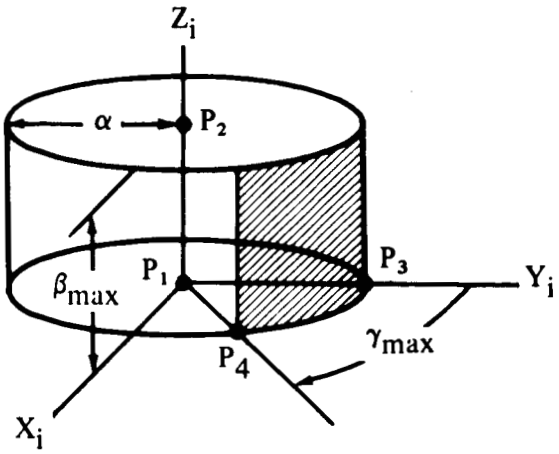
$\beta_{max} = |\bar{r}_3 - \bar{r}_1|\sin\theta_3$

$\gamma_{min} = \theta_3 - 90$

$\gamma_{max} = \theta_1 + \theta_3 - 90$

FIGURE 1 - SURFACE TYPES AND THEIR COORDINATE SYSTEMS

Surface Type ± 4, Cylinder Lateral Surface



$\alpha = |\bar{r}_3 - \bar{r}_1|$

$\beta_{min} = 0$

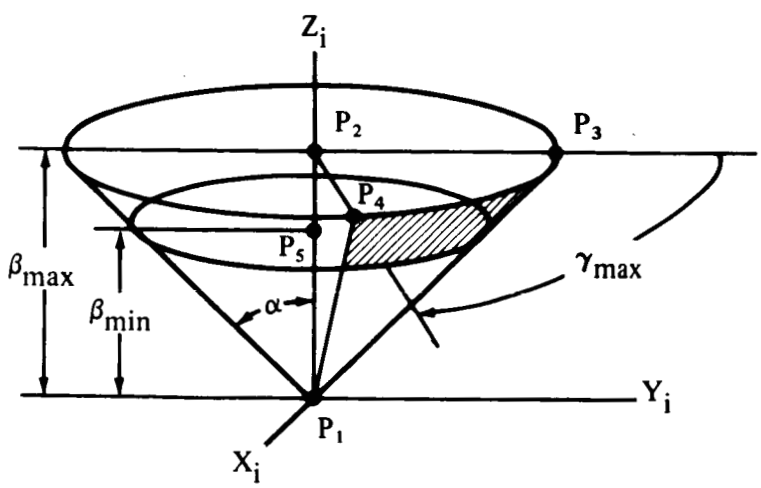
$\beta_{max} = |\bar{r}_2 - \bar{r}_1|$

$\gamma_{min} = 0$

$\gamma_{max} = 2\sin^{-1}(|\bar{r}_4 - \bar{r}_3|/2|\bar{r}_3 - r_1|)$

$0 < \gamma_{max} \leqslant 360°$

Surface Type ± 5, Cone and Cone Frustum Lateral Surface



$\alpha = \tan^{-1}(|\bar{r}_3 - \bar{r}_2|/|\bar{r}_2 - \bar{r}_1|)$

$\beta_{min} = |\bar{r}_5 - \bar{r}_1|$, frustum

$\beta_{min} = 0$, cone

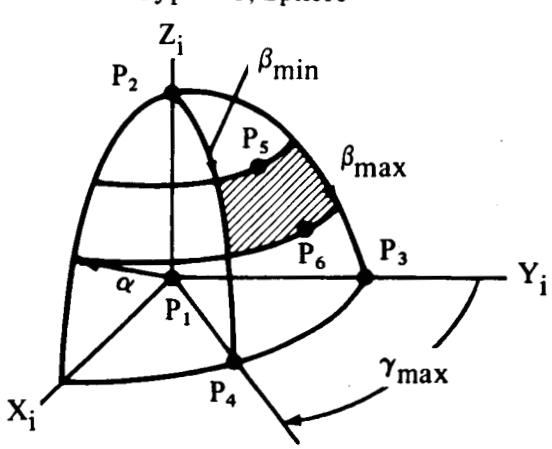
$\beta_{max} = |\bar{r}_2 - \bar{r}_1|$

$\gamma_{min} = 0$

$\gamma_{max} = 2\sin^{-1}(|\bar{r}_4 - \bar{r}_3|/2|\bar{r}_3 - \bar{r}_2|)$

$0 < \gamma_{max} \leqslant 360°$

Surface Type ± 6, Sphere



$\alpha = |\bar{r}_2 - \bar{r}_1|$

$\beta_{min} = 2\sin^{-1}(|\bar{r}_5 - \bar{r}_2|/2|\bar{r}_2 - \bar{r}_1|)$

$\beta_{max} = 2\sin^{-1}(|\bar{r}_6 - \bar{r}_2|/2|\bar{r}_2 - \bar{r}_1|)$

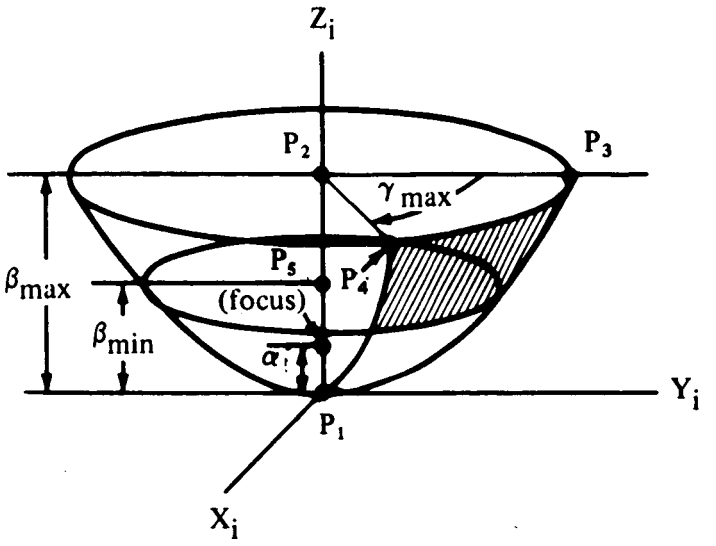
$\gamma_{min} = 0$

$\gamma_{max} = 2\sin^{-1}(|\bar{r}_4 - \bar{r}_3|/2|\bar{r}_2 - \bar{r}_1|)$

$0 < \gamma_{max} \leqslant 360°$

FIGURE 2 - SURFACE TYPES AND THEIR COORDINATE SYSTEMS

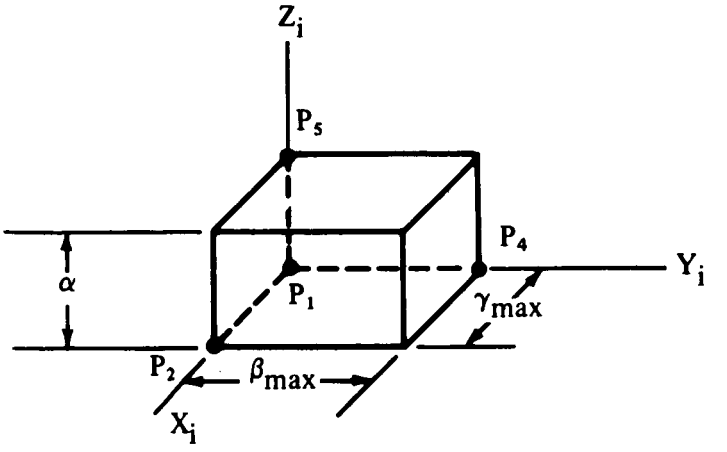
Surface Type ± 7, Circular Paraboloid



$$\alpha = |\bar{r}_3 - \bar{r}_2|^2/4|\bar{r}_2 - \bar{r}_1|$$
$$\beta_{min} = |\bar{r}_5 - \bar{r}_1|$$
$$\beta_{max} = |\bar{r}_2 - \bar{r}_1|$$
$$\gamma_{min} = 0$$
$$\gamma_{max} = 2\sin^{-1}(|\bar{r}_4 - \bar{r}_3|/2|\bar{r}_3 - r_2|)$$
$$0 < \gamma_{max} \leqslant 360°$$

Surface Type ± 15 or ± 16, Rectangular Prism *

$$\alpha = |\bar{r}_5 - \bar{r}_1|$$
$$\beta_{min} = \gamma_{min} = 0$$
$$\beta_{max} = |\bar{r}_4 - \bar{r}_1|$$
$$\gamma_{max} = |\bar{r}_2 - \bar{r}_1|$$



* Type 16 Surface Has Open Face in $X_i - Y_i$ Plane

FIGURE 3 - SURFACE TYPES AND THEIR COORDINATE SYSTEMS

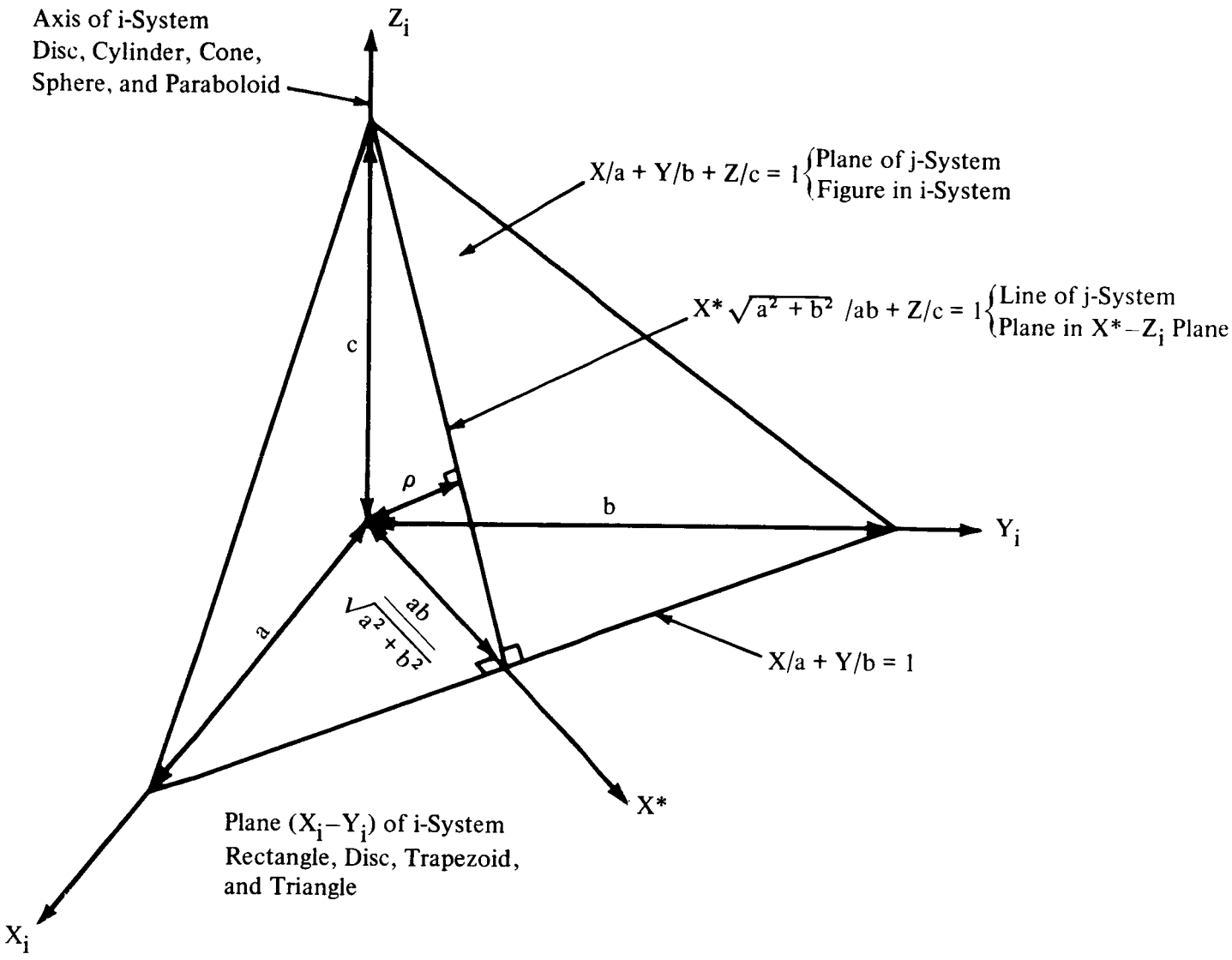


FIGURE 4 - ORIENTATION OF j-SYSTEM PLANE IN COORDINATE SYSTEM OF i-SYSTEM FIGURE, a, b, AND $c > 0$

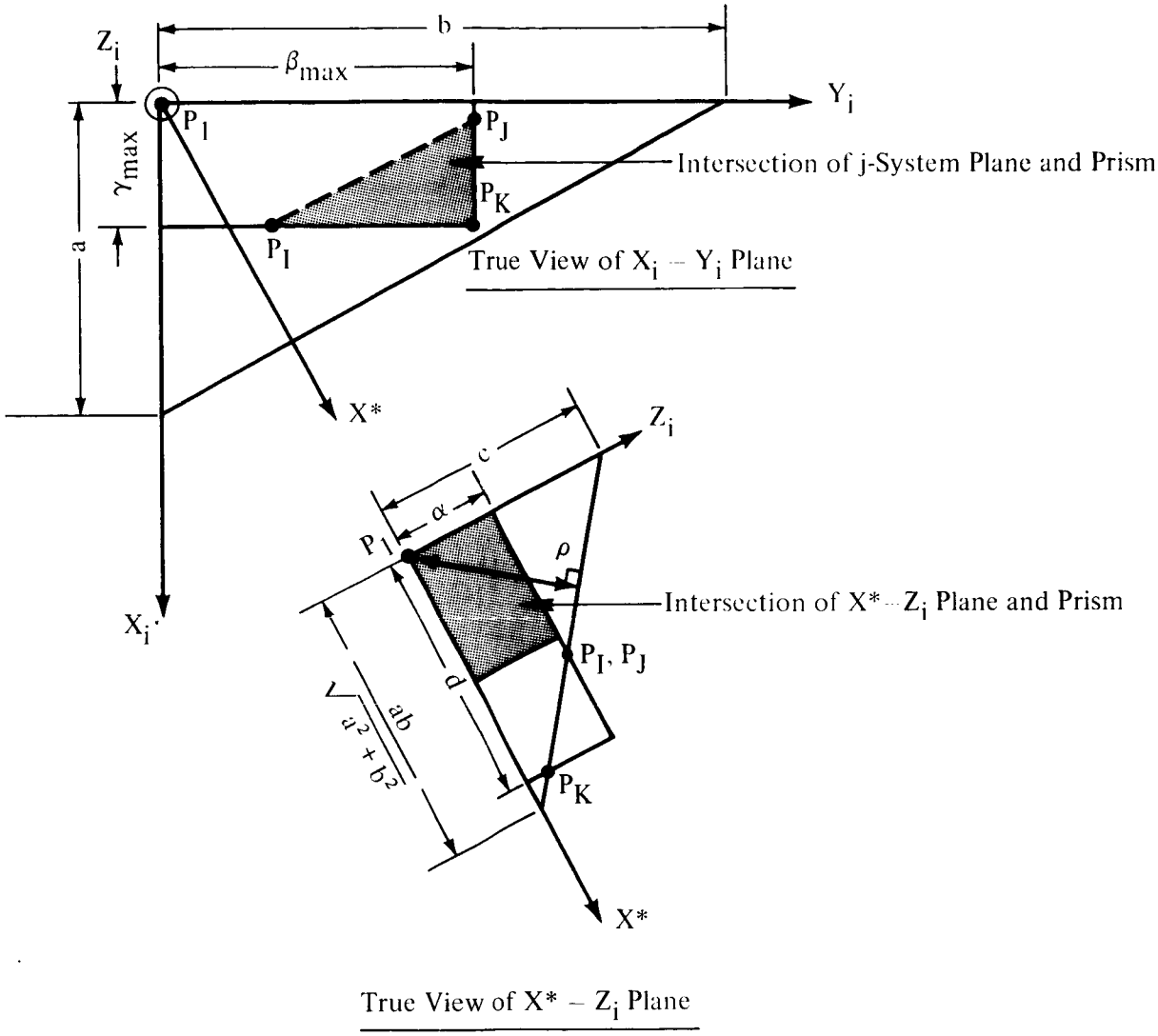


FIGURE 5 - ORIENTATION OF i-SYSTEM RECTANGULAR PRISM AND INTERSECTING j-SYSTEM PLANE, a, b, and c > 0

```
      SUBROUTINE EANCAL(PHI,PSI,OMEGA,ISF,ITNS)
C
C AUTHOR        L.K. HAWKINS
C
C               SUBROUTINE EANCAL WITH FIGURE INTERSECTION AND FIGURE
C               ORIENTATION DETERMINATION CAPABILITIES
C PURPOSE       SUBROUTINE EANCAL IS USED TO CALCULATE THE INPUT FOR
C               RAVFAC/VLFACT FOR THE VARIOUS FIGURE TYPES
C
C               READ DATA CARD CONTAINING FIGURE TYPE AND DIRECTION OF
C               ACTIVE NORMAL OF SURFACE (PLUS OR MINUS ILK) AND NUMBER OF
C               CARDS FOLLOWING WITH X,Y,Z COORDINATES FOR THAT FIGURE
C               TYPE (N).  ITNS=THE TOTAL NUMBER OF SURFACES IN THE SYSTEM
C
C
      COMMON/LOCS/TI1(150),TI2(150),TI3(150),TJI(150),TJ2(150),TJ3(150)
     .,TK1(150),TK2(150),TK3(150),RX(150),RY(150),RZ(150)
      COMMON/LIMT/EMIN(150),EMAX(150),GMIN(150),GMAX(150),A(150),ILK(15
     .0),KS(150),NSHADE(150),NS,DEGRAD,PI,PI2
C
      DIMENSION X1(150),X2(150),X3(150),X4(150),X6(150),
     . Y1(150),Y2(150),Y3(150),Y4(150),Y6(150),Z1(150),
     . Z2(150),Z3(150),Z4(150),Z6(150),P11(150),P12(150),
     . P13(150),P21(150),P22(150),P23(150),P31(150),P32(150),
     . P33(150),R(150),F(150),ISAVE(150),NSAVE(150),ILSAV(150)
      DIMENSION D(3,3),V(2),AD(3,3),BD(3,3),CD(3,3),JC(1)
      DIMENSION X(6),Y(6),Z(6)
      DIMENSION SA1(50),SA2(50),SA3(50),SA4(50),SA5(50),SA6(50)
      DIMENSION SY1(50),SY2(50),SY3(50),SY4(50),SY5(50),SY6(50)
      DIMENSION SZ1(50),SZ2(50),SZ3(50),SZ4(50),SZ5(50),SZ6(50)
      REAL OMEGA
      II=-1
      RTOD=.572957E0E02
      IF(NS.NE.1) GO TO 9998
      READ(5,10) ILK(NS),N,ITNS
      GO TO 9997
 9998 READ(5,225) ILK(NS),N
C
C               INITIALIZATION
C
 9997 DO 1 L=1,6
      X(L)=0.
      Y(L)=0.
 1    Z(L)=0.
C
      LP4=0
      LP5=0
      LP6=0
C
C
      IF(ILK(NS)) 4,5,5
 4    LLK=-ILK(NS)
      GO TO 6
 5    LLK=ILK(NS)
 6    GO TO(7,8,9,11,13,14,15,2,2,2,2,2,2,17,23),LLK
C
C               READ N CARDS CONTAINING THE POINT NUMBER (NP) AND
C               THE COORDINATES (X(NP),Y(NP),Z(NP)). THE UPPER
C               LIMIT OF 'I' DEPENDING ON THE TYPE OF FIGURE
C
C
 100  WRITE(6,160)
      DO 3 I=1,N
      READ(5,20) NP,X(NP),Y(NP),Z(NP)
      IF(NP.EQ.4) LP4=1
      IF(NP.EQ.5) LP5=1
      IF(NP.EQ.6) LP6=1
      IF(NP.GT.3.AND.Y(I).EQ.999.) GO TO 81
      WRITE(6,220) NP,X(NP),Y(NP),Z(NP)
      GO TO 82
 81   WRITE(6,170) NP,X(NP)
 82   IF(NP.EQ.4.AND.Y(4).EQ.999.) WRITE(6,190)
      IF(LLK.EQ.15.OR.LLK.EQ.16) GO TO 84
      IF(NP.EQ.5.AND.Y(5).EQ.999.) WRITE(6,200)
      GO TO 85
 84   IF(NP.EQ.5.AND.Y(5).EQ.999.) WRITE(6,215)
 85   IF(NP.EQ.6.AND.Y(6).EQ.999.) WRITE(6,210)
 3    CONTINUE
C
      AL=SQRT((X(5)-X(1))**2+(Y(5)-Y(1))**2+(Z(5)-Z(1))**2)
      CL=SQRT((X(4)-X(1))**2+(Y(4)-Y(1))**2+(Z(4)-Z(1))**2)
      L1=SQRT((X(3)-X(2))**2+(Y(3)-Y(2))**2+(Z(3)-Z(2))**2)
      L2=SQRT((X(3)-X(1))**2+(Y(3)-Y(1))**2+(Z(3)-Z(1))**2)
      L3=SQRT((X(2)-X(1))**2+(Y(2)-Y(1))**2+(Z(2)-Z(1))**2)
      L4=SQRT((X(4)-X(3))**2+(Y(4)-Y(3))**2+(Z(4)-Z(3))**2)
      L5=SQRT((X(5)-X(2))**2+(Y(5)-Y(2))**2+(Z(5)-Z(2))**2)
      L6=SQRT((X(6)-X(2))**2+(Y(6)-Y(2))**2+(Z(6)-Z(2))**2)
C
C               ORIGIN OF SCS FOR ALL FIGURE TYPES
C               ARE THE COORDINATES OF POINT P1
C
      RX(NS)=X(1)
      RY(NS)=Y(1)
      RZ(NS)=Z(1)
C
C
      GO TO(71,72,27,73,74,75,76,2,2,2,2,2,2,18,18),LLK
C
C               SURFACE TYPE  1,RECTANGLE
C
 7    WRITE(6,30)
      GO TO 100
C
C               SCS DEFINING PARAMETERS FOR RECTANGLE
C
 71   A(NS)=0.
      EMIN(NS)=0.
      EMAX(NS)=CL
      GMIN(NS)=0.
      GMAX(NS)=L3
C               DIRECTION COSINES FOR RECTANGLE
C
 19   E=GMAX(NS)
      C=EMAX(NS)
C
      P11(NS)=(X(2)-X(1))/E
      P12(NS)=(Y(2)-Y(1))/E
      P13(NS)=(Z(2)-Z(1))/E
      P21(NS)=(X(4)-X(1))/C
      P22(NS)=(Y(4)-Y(1))/C
      P23(NS)=(Z(4)-Z(1))/C
      P31(NS)=((Y(2)-Y(1))*(Z(4)-Z(1))-(Z(2)-Z(1))*(Y(4)-Y(1)))/(E*C)
      P32(NS)=((Z(2)-Z(1))*(X(4)-X(1))-(X(2)-X(1))*(Z(4)-Z(1)))/(E*C)
      P33(NS)=((X(2)-X(1))*(Y(4)-Y(1))-(Y(2)-Y(1))*(X(4)-X(1)))/(E*C)
C
      GO TO 99
C
C               SURFACE TYPE  2,DISC
C
 8    WRITE(6,40)
      GO TO 100
C
C               SCS DEFINING PARAMETERS FOR DISC
C
 72   IF(LP4.EQ.0) GO TO 36
      IF(LP4.EQ.1.AND.Y(4).EQ.999.) GO TO 33
      GMAX(NS)=2.*ASIN(L4/(2.*L2))*RTOD
      GO TO 34
 36   GMAX(NS)=360.
      GO TO 34
 33   GMAX(NS)=X(4)
 34   A(NS)=0.
      IF(LP5.EQ.0) GO TO 37
      IF(LP5.EQ.1.AND.Y(5).EQ.999.) GO TO 39
      EMIN(NS)=AL
      GO TO 39
 37   EMIN(NS)=0.
      GO TO 39
 38   EMIN(NS)=X(5)
 39   EMAX(NS)=L2
      GMIN(NS)=0.
C
C
C
 12   R(NS)=L2
      F(NS)=L3
      P11(NS)=((Y(3)-Y(1))*(Z(2)-Z(1))-(Z(3)-Z(1))*(Y(2)-Y(1)))/
     . (R(NS)*F(NS))
      P12(NS)=((Z(3)-Z(1))*(X(2)-X(1))-(X(3)-X(1))*(Z(2)-Z(1)))/
     . (R(NS)*F(NS))
      P13(NS)=((X(3)-X(1))*(Y(2)-Y(1))-(Y(3)-Y(1))*(X(2)-X(1)))/
     . (R(NS)*F(NS))
      IF(X(4).EQ.0.) GO TO 28
      X14=(X(4)-X(1))*P11(NS)+(Y(4)-Y(1))*P12(NS)+(Z(4)-Z(1))*
     . P13(NS)
      IF(X14.LT.0.) GMAX(NS)=360.-GMAX(NS)
 28   P21(NS)=(X(3)-X(1))/F(NS)
      P22(NS)=(Y(3)-Y(1))/F(NS)
      P23(NS)=(Z(3)-Z(1))/F(NS)
      P31(NS)=(X(2)-X(1))/F(NS)
      P32(NS)=(Y(2)-Y(1))/F(NS)
      P33(NS)=(Z(2)-Z(1))/F(NS)
      GO TO 99
C
C               SURFACE TYPE  3,TRIANGLE OR TRAPEZOID
C
 9    IF(X(4).EQ.0..AND.Y(4).EQ.0..AND.Z(4).EQ.0.) GO TO 21
      WRITE(6,80)
      GO TO 100
 21   WRITE(6,50)
      GO TO 100
C
C               ORIGIN OF SCS FOR TRAPEZOID
C               COORDINATES OF POINT P1 ARE CALCULATED
C
 27   IF(X(4).EQ.0..AND.Y(4).EQ.0..AND.Z(4).EQ.0.) GO TO 22
      XL=(X(4)-X(3))/L4
      XM=(Y(4)-Y(3))/L4
      XN=(Z(4)-Z(3))/L4
      T=(X(5)-X(2))/L5
      U=(Y(5)-Y(2))/L5
      V1=(Z(5)-Z(2))/L5
      X(1)=(Y(5)-Y(4)+(XM/XL)*X(4)-(U/T)*X(5))/(XM/XL-U/T)
      RX(NS)=X(1)
      Y(1)=(X(5)-X(4)+(XL/XM)*Y(4)-(T/U)*Y(5))/(XL/XM-T/U)
      RY(NS)=Y(1)
      Z(1)=(Y(5)-Y(4)+(XM/XL)*Z(4)-(U/VT)*Z(5))/(XM/XL-U/VT)
      RZ(NS)=Z(1)
      L2=SQRT((X(3)-X(1))**2+(Y(3)-Y(1))**2+(Z(3)-Z(1))**2)
      L3=SQRT((X(2)-X(1))**2+(Y(2)-Y(1))**2+(Z(2)-Z(1))**2)
      CL=SQRT((X(4)-X(1))**2+(Y(4)-Y(1))**2+(Z(4)-Z(1))**2)
C
C               SCS DEFINING PARAMETERS FOR TRIANGLE AND TRAPEZOID
C
 22   CONTINUE
      THETA1=ACOS(((X(3)-X(1))*(X(2)-X(1))+(Y(3)-Y(1))*
     . (Y(2)-Y(1))+(Z(3)-Z(1))*(Z(2)-Z(1)))/(L2*L3))
      SINTH1=SIN(THETA1)
      THETA3=ACOS(((X(3)-X(2))*(X(3)-X(1))+(Y(3)-Y(2))*
     . (Y(3)-Y(1))+(Z(3)-Z(2))*(Z(3)-Z(1)))/(L1*L2))
      SINTH3=SIN(THETA3)
C
*DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
      IF(X(4).EQ.0..AND.Y(4).EQ.0..AND.Z(4).EQ.0.) EMIN(NS)=0.
*DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
      IF(X(4).NE.0..OR.Y(4).NE.0..OR.Z(4).NE.0.) EMIN(NS)=CL*SINTH3
      A(NS)=0.
      EMAX(NS)=L2*SINTH3
      GMIN(NS)=THETA3*RTOD-90.
      GMAX(NS)=THETA1*RTOD+THETA3*RTOD-90.
C
C               DIRECTION COSINES FOR TRIANGLE AND TRAPEZOID
C
      XNX=((Y(2)-Y(1))*(Z(3)-Z(1))-(Z(2)-Z(1))*(Y(3)-Y(1)))/(L2*L3*SINT
     .H1)
      XNY=((Z(2)-Z(1))*(X(3)-X(1))-(X(2)-X(1))*(Z(3)-Z(1)))/(L2*L3*SINT
     .H1)
      XNZ=((X(2)-X(1))*(Y(3)-Y(1))-(Y(2)-Y(1))*(X(3)-X(1)))/(L2*L3*SINT
     .H1)
      P11(NS)=-(X(3)-X(2))/U1
      P12(NS)=-(Y(3)-Y(2))/U1
      P13(NS)=-(Z(3)-Z(2))/U1
      P21(NS)=(XNY*(Z(2)-Z(3))-XNZ*(Y(2)-Y(3)))/U1
      P22(NS)=(XNZ*(X(2)-X(3))-XNX*(Z(2)-Z(3)))/U1
      P23(NS)=(XNX*(Y(2)-Y(3))-XNY*(X(2)-X(3)))/U1
      P31(NS)=XNX
      P32(NS)=XNY
      P33(NS)=XNZ
      GO TO 99
C
C               SURFACE TYPE  4,RIGHT CIRCULAR CYLINDER
C
 11   WRITE(6,70)
      GO TO 100
 73   A(NS)=L2
      EMIN(NS)=0.
      EMAX(NS)=L3
      GMIN(NS)=0.
      IF(LP4.EQ.0) GO TO 41
*DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
      IF(LP4.EQ.1.AND.Y(4).EQ.999.) GO TO 42
      GMAX(NS)=2.*ASIN(L4/(2.*L2))*RTOD
```

FIGURE 6 - EANCAL LISTING

```
251*          GO TO 43
252*    41    GMAX(NS)=360.
253*          GO TO 43
254*    42    GMAX(NS)=X(4)
255*    43    CONTINUE
256*          GO TO 12
257*   C
258*   C              SURFACE TYPE  5,RIGHT CIRCULAR CONE
259*   C
260*    13    WRITE(6,80)
261*          GO TO 100
262*   C
263*   C              SCS DEFINING PARAMETERS FOR RIGHT CIRCULAR CONE
264*   C
265*    74    A(NS)=ATAN2(U1,U3)*RTOD
266*    16    IF(LP4.EQ.0) GO TO 44
*DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
267*          IF(LP4.EQ.1.AND.Y(4).EQ.999.) GO TO 46
268*          GMAX(NS)=2.*ASIN(L4/(2.*L1))*RTOD
269*          GO TO 47
270*    44    GMAX(NS)=360.
271*          GO TO 47
272*    46    GMAX(NS)=X(4)
273*    47    CONTINUE
274*          IF(LP5.EQ.0) GO TO 48
*DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
275*          IF(LP5.EQ.1.AND.Y(5).EQ.999.) GO TO 49
276*          BMIN(NS)=A0
277*          GO TO 51
278*    48    BMIN(NS)=0.
279*          GO TO 51
280*    49    BMIN(NS)=X(5)
281*    51    BMAX(NS)=U3
282*          CMIN(NS)=0.
283*   C
284*   C              DIRECTION COSINES FOR RIGHT CIRCULAR CONE
285*   C
286*          R(NS)=U1
287*          H(NS)=U3
288*          P11(NS)=((Y(3)-Y(2))*(Z(2)-Z(1))-(Z(3)-Z(2))*(Y(2)-Y(1)))/
289*        . (R(NS)*H(NS))
290*          P12(NS)=((Z(3)-Z(2))*(X(2)-X(1))-(X(3)-X(2))*(Z(2)-Z(1)))/
291*        . (R(NS)*H(NS))
292*          P13(NS)=((X(3)-X(2))*(Y(2)-Y(1))-(Y(3)-Y(2))*(X(2)-X(1)))/
293*        . (R(NS)*H(NS))
*DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
294*          IF(X(4).EQ.0.) GO TO 29
295*          XI4=(X(4)-X(1))*P11(NS)+(Y(4)-Y(1))*P12(NS)+(Z(4)-Z(1))*
296*        . P13(NS)
297*          IF(XI4.LT.0.) GMAX(NS)=360.-GMAX(NS)
298*    29    P21(NS)=(X(3)-X(2))/R(NS)
299*          P22(NS)=(Y(3)-Y(2))/R(NS)
300*          P23(NS)=(Z(3)-Z(2))/R(NS)
301*          P31(NS)=(X(2)-X(1))/H(NS)
302*          P32(NS)=(Y(2)-Y(1))/H(NS)
303*          P33(NS)=(Z(2)-Z(1))/H(NS)
304*          GO TO 99
305*   C
306*   C              SURFACE TYPE  6,SPHERE
307*   C
308*    14    WRITE(6,90)
309*          GO TO 100
310*   C
311*   C              SCS DEFINING PARAMETERS FOR SPHERE
312*   C
313*    75    A(NS)=U3
314*          IF(LP4.EQ.0) GO TO 52
*DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
315*          IF(LP4.EQ.1.AND.Y(4).EQ.999.) GO TO 53
316*          GMAX(NS)=2.*ASIN(L4/(2.*L2))*RTOD
317*          GO TO 54
318*    52    GMAX(NS)=360.
319*          GO TO 54
320*    53    GMAX(NS)=X(4)
321*    54    CONTINUE
322*          IF(LP5.EQ.0) GO TO 56
*DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
323*          IF(LP5.EQ.1.AND.Y(5).EQ.999.) GO TO 57
324*          BMIN(NS)=2.*ASIN(L5/(2.*L3))*RTOD
325*          GO TO 58
326*    56    BMIN(NS)=0.
327*          GO TO 58
328*    57    BMIN(NS)=X(5)
329*    58    CONTINUE
330*          IF(LP6.EQ.0) GO TO 59
*DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
331*          IF(LP6.EQ.1.AND.Y(6).EQ.999.) GO TO 61
332*          BMAX(NS)=2.*ASIN(L6/(2.*L3))*RTOD
333*          GO TO 62
334*    59    BMAX(NS)=180.
335*    61    BMAX(NS)=X(6)
336*    62    CMIN(NS)=0.
337*   C
338*   C              DIRECTION COSINES FOR SPHERE
339*   C
340*          R(NS)=A(NS)
341*          P11(NS)=((Y(3)-Y(1))*(Z(2)-Z(1))-(Z(3)-Z(1))*(Y(2)-Y(1)))/
342*        . R(NS)**2
343*          P12(NS)=((Z(3)-Z(1))*(X(2)-X(1))-(X(3)-X(1))*(Z(2)-Z(1)))/
344*        . R(NS)**2
345*          P13(NS)=((X(3)-X(1))*(Y(2)-Y(1))-(Y(3)-Y(1))*(X(2)-X(1)))/
346*        . R(NS)**2
*DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
347*          IF(X(4).EQ.0.) GO TO 31
348*          XI4=(X(4)-X(1))*P11(NS)+(Y(4)-Y(1))*P12(NS)+(Z(4)-Z(1))*
349*        . P13(NS)
350*          IF(XI4.LT.0.) GMAX(NS)=360.-GMAX(NS)
351*    31    P21(NS)=(X(3)-X(1))/R(NS)
352*          P22(NS)=(Y(3)-Y(1))/R(NS)
353*          P23(NS)=(Z(3)-Z(1))/R(NS)
354*          P31(NS)=(X(2)-X(1))/R(NS)
355*          P32(NS)=(Y(2)-Y(1))/R(NS)
356*          P33(NS)=(Z(2)-Z(1))/R(NS)
357*   C      IF(X(5).EQ.0.) GO TO 32
358*          GO TO 99
359*   C
360*   C              SURFACE TYPE  7,CIRCULAR PARABOLOID
361*   C
362*    15    WRITE(6,110)
363*          GO TO 100
364*    76    A(NS)=U1**2/(4.*U3)
365*          GO TO 16
366*   C
367*   C              SURFACE TYPE  15,RECTANGULAR PRISM (6 CLOSED SIDES)
368*   C
369*    17    WRITE(6,120)
370*          GO TO 100
371*   C
372*   C              SCS DEFINING PARAMETERS FOR 6 CLOSED SIDED BOX
373*   C
374*    18    IF(LP5.EQ.0) GO TO 83
*DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
375*          IF(LP5.EQ.1.AND.Y(5).EQ.999.) A(NS)=X(5)
376*          GO TO 86
377*    83    A(NS)=A0
378*    86    BMIN(NS)=0.
379*          BMAX(NS)=C0
380*          CMIN(NS)=0.
381*          IF(LP4.EQ.0) GMAX(NS)=360.
382*          IF(LP4.NE.0) GMAX(NS)=U3
383*          GO TO 19
384*   C
385*   C              SURFACE TYPE  16,RECTANGULAR PRISM (5 CLOSED SIDES)
386*   C
387*    23    WRITE(6,130)
388*          GO TO 100
389*   C
390*   C              COMPUTE EULER ANGLES IN DEGREES
391*   C
392*    99    CONTINUE
393*          PSI=ASIN(P13(NS))*RTOD
394*          IF(PSI.LT.0.) PSI=360.+PSI
395*          SINPSI=SIN(PSI/RTOD)
396*   C
397*   C              IF SINPSI = 1, WRITE DIAGNOSTIC MESSAGE
398*   C
*DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
399*          IF(ABS(SINPSI).EQ.1.) WRITE(6,180) ISF,ISF
400*   C
401*   C
402*          PHI=ATAN2(-P12(NS),P11(NS))*RTOD
403*          IF(PHI.LT.0.) PHI=360.+PHI
404*          OMEGA=ATAN2(P23(NS),P33(NS))*RTOD
405*          IF(OMEGA.LT.0.) OMEGA=360.+OMEGA
406*   C
407*   C
408*   C
409*    10    FORMAT(I5,5X,I2,5X,I3)
410*    20    FORMAT(I2,E14.8,2E16.8)
411*    30    FORMAT(1H1,' THE INPUT SURFACE IS A RECTANGLE')
412*    40    FORMAT(1H1,' THE INPUT SURFACE IS A DISC')
413*    50    FORMAT(1H1,' THE INPUT SURFACE IS A TRIANGLE')
414*    60    FORMAT(1H1,' THE INPUT SURFACE IS A TRAPEZOID')
415*    70    FORMAT(1H1,' THE INPUT SURFACE IS A RIGHT CIRCULAR CYLINDER')
416*    80    FORMAT(1H1,' THE INPUT SURFACE IS A RIGHT CIRCULAR CONE')
417*    90    FORMAT(1H1,' THE INPUT SURFACE IS A SPHERE')
418*    110   FORMAT(1H1,' THE INPUT SURFACE IS A CIRCULAR PARABOLOID')
419*    120   FORMAT(1H1,' THE INPUT SURFACE IS A 6 CLOSED SIDED BOX')
420*    130   FORMAT(1H1,' THE INPUT SURFACE IS A 5 CLOSED SIDED BOX')
421*    160   FORMAT(' POINT',7X,'X',15X,'Y',15X,'Z')
422*    170   FORMAT(3X,I1,2X,F16.8)
423*    180   FORMAT(' ***THE VIEW FACTOR(S) FROM SURFACE ',I2,' TO OTHER SURFA
424*        .CES IF LISTED ARE INCORRECT. ',/' ***THE EULER ANGLE SET IS INCORR
425*        .ECT. ',/' ***REDEFINE SURFACE ',I2,' POINTS SO THAT SCS X-AXIS IS
426*        .NOT PARALLEL TO CCS Z-AXIS ')
427*    190   FORMAT(1H+,58X,'X(4)=GAMMA(MAX)')
428*    200   FORMAT(1H+,58X,'X(5)=BETA(MIN)')
429*    210   FORMAT(1H+,58X,'X(6)=BETA(MAX)')
430*    215   FORMAT(1H+,58X,'X(5)=ALPHA')
431*    220   FORMAT(3X,I1,2X,3(F16.8))
432*    225   FORMAT(I5,5X,I2)
433*   C
434*   C
435*   C              STORE FOR INTERSECTION DETERMINATION
436*   C
437*    400   CONTINUE
438*          NSAVE(NS)=ISF
439*          SX1(NS)=X(1)
440*          SX2(NS)=X(2)
441*          SX3(NS)=X(3)
442*          SX4(NS)=X(4)
443*          SX5(NS)=X(5)
444*          SX6(NS)=X(6)
445*          SY1(NS)=Y(1)
446*          SY2(NS)=Y(2)
447*          SY3(NS)=Y(3)
448*          SY4(NS)=Y(4)
449*          SY5(NS)=Y(5)
450*          SY6(NS)=Y(6)
451*          SZ1(NS)=Z(1)
452*          SZ2(NS)=Z(2)
453*          SZ3(NS)=Z(3)
454*          SZ4(NS)=Z(4)
455*          SZ5(NS)=Z(5)
456*          SZ6(NS)=Z(6)
457*   C
458*          IF(NS.NE.ITNS) GO TO 2
459*   C
460*   C              INTERSECTION DETERMINATION
461*   C
462*   C
463*   C              CHECK FOR FIGURE TYPES 1, 2 OR 3
464*   C
465*          DO 301 I=1,ITNS
466*    301   IF(IABS(ILK(I)).LE.3) GO TO 302
467*          GO TO 2
468*   C
469*   C              CHECK FOR TYPE 2 FIGURES
470*   C
471*    302   DO 303 I=1,ITNS
472*          IF(IABS(ILK(I)).NE.2) GO TO 303
473*   C
474*   C              CALCULATE X6,Y6,Z6 FOR EACH DISC
475*   C
476*          SX6(I)=SX1(I)+P11(I)*R(I)
477*          SY6(I)=SY1(I)+P12(I)*R(I)
478*          SZ6(I)=SZ1(I)+P13(I)*R(I)
479*   C
480*    303   CONTINUE
481*   C
482*   C              SAVE SURFACE NUMBER (NS) OF PLANE FIGURES IN THE SYSTEM
483*   C
484*          U=0
485*          DO 304 I=1,ITNS
486*          IF(IABS(ILK(I)).LE.3) GO TO 305
487*          GO TO 304
488*    305   U=U+1
489*          ISAVE(U)=I
490*    304   CONTINUE
491*   C
492*          DO 317 K=1,U
493*          DO 317 I=1,ITNS
494*          IF(I.EQ.ISAVE(K)) GO TO 318
495*          GO TO 317
```

FIGURE 6 - EANCAL LISTING (CONTINUED)

```
496*   C
497*   C                TRANSFORMATION OF COORDINATES
498*   C
499*    318   DO 319 L=1,ITNS
500*          IF(L.EQ.1) GO TO 319
501*          XDIF=RX(I)-RX(L)
502*          YDIF=RY(I)-RY(L)
503*          ZDIF=RZ(I)-RZ(L)
504*          X1(I)=XDIF*P11(L)+YDIF*P12(L)+ZDIF*P13(L)
505*          Y1(I)=XDIF*P21(L)+YDIF*P22(L)+ZDIF*P23(L)
506*          Z1(I)=XDIF*P31(L)+YDIF*P32(L)+ZDIF*P33(L)
507*          IF(IABS(ILK(I)).EQ.2) GO TO 320
508*   C
509*          XDIF=SX2(I)-RX(L)
510*          YDIF=SY2(I)-RY(L)
511*          ZDIF=SZ2(I)-RZ(L)
512*          X2(I)=XDIF*P11(L)+YDIF*P12(L)+ZDIF*P13(L)
513*          Y2(I)=XDIF*P21(L)+YDIF*P22(L)+ZDIF*P23(L)
514*          Z2(I)=XDIF*P31(L)+YDIF*P32(L)+ZDIF*P33(L)
515*          IF(IABS(ILK(I)).EQ.1) GO TO 321
516*   C
517*    320   XDIF=SX3(I)-RX(L)
518*          YDIF=SY3(I)-RY(L)
519*          ZDIF=SZ3(I)-RZ(L)
520*          X3(I)=XDIF*P11(L)+YDIF*P12(L)+ZDIF*P13(L)
521*          Y3(I)=XDIF*P21(L)+YDIF*P22(L)+ZDIF*P23(L)
522*          Z3(I)=XDIF*P31(L)+YDIF*P32(L)+ZDIF*P33(L)
523*          IF(IABS(ILK(I)).EQ.3) GO TO 323
524*          IF(IABS(ILK(I)).NE.1) GO TO 322
525*   C
526*    321   XDIF=SX4(I)-RX(L)
527*          YDIF=SY4(I)-RY(L)
528*          ZDIF=SZ4(I)-RZ(L)
529*          X4(I)=XDIF*P11(L)+YDIF*P12(L)+ZDIF*P13(L)
530*          Y4(I)=XDIF*P21(L)+YDIF*P22(L)+ZDIF*P23(L)
531*          Z4(I)=XDIF*P31(L)+YDIF*P32(L)+ZDIF*P33(L)
532*          IF(IABS(ILK(I)).EQ.1) GO TO 323
533*   C
534*    322   XDIF=SX6(I)-RX(L)
535*          YDIF=SY6(I)-RY(L)
536*          ZDIF=SZ6(I)-RZ(L)
537*          X6(I)=XDIF*P11(L)+YDIF*P12(L)+ZDIF*P13(L)
538*          Y6(I)=XDIF*P21(L)+YDIF*P22(L)+ZDIF*P23(L)
539*          Z6(I)=XDIF*P31(L)+YDIF*P32(L)+ZDIF*P33(L)
540*   C
541*   C                COMPUTE DETERMINANTS D,AD,BD,CD AND
542*   C                CALCULATE INTERCEPTS AI,BI,CI
543*   C
544*    323   CONTINUE
545*          IF(IABS(ILK(I)).EQ.1) GO TO 324
546*          IF(IABS(ILK(I)).EQ.2) GO TO 325
547*          GO TO 326
548*    324   X3(I)=X4(I)
549*          Y3(I)=Y4(I)
550*          Z3(I)=Z4(I)
551*          GO TO 326
552*    325   X2(I)=X6(I)
553*          Y2(I)=Y6(I)
554*          Z2(I)=Z6(I)
555*   C
556*   C
557*    326   V(1)=2.
558*          D(1,1)=X1(I)
559*          D(1,2)=Y1(I)
560*          D(1,3)=Z1(I)
561*          D(2,1)=X2(I)
562*          D(2,2)=Y2(I)
563*          D(2,3)=Z2(I)
564*          D(3,1)=X3(I)
565*          D(3,2)=Y3(I)
566*          D(3,3)=Z3(I)
567*   C
568*   C
569*   C                COMPUTE DETERMINANTS
570*   C
571*    367   CALL GOR(D,3,3,3,3,$341,JC,V)
572*          GO TO 343
573*   C
574*   C                THE DETERMINANT IS ZERO (D)
575*   C
576*    341   DETM=0.
577*          GO TO 501
578*   C
579*   C
580*    343   CONTINUE
581*          DETA=EXP(V(2))
582*          DETM=SIGN(DETA,V(1))
583*          IF(1.E-5.GT.ABS(DETM)) DETM=0.
584*   C
585*    501   V(1)=2.
586*          AD(1,1)=Y1(I)
587*          AD(1,2)=Z1(I)
588*          AD(1,3)=1.
589*          AD(2,1)=Y2(I)
590*          AD(2,2)=Z2(I)
591*          AD(2,3)=1.
592*          AD(3,1)=Y3(I)
593*          AD(3,2)=Z3(I)
594*          AD(3,3)=1.
595*          CALL GOR(AD,3,3,3,3,$344,JC,V)
596*          GO TO 348
597*   C
598*   C                DETERMINANT AD IS ZERO
599*   C
600*    344   DETM2=0.
601*          GO TO 366
602*    348   DET2A=EXP(V(2))
603*          DETM2=SIGN(DET2A,V(1))
604*          IF(1.E-5.GT.ABS(DETM2)) DETM2=0.
605*   C
606*    366   V(1)=2.
607*          BD(1,1)=Z1(I)
608*          BD(1,2)=X1(I)
609*          BD(1,3)=1.
610*          BD(2,1)=Z2(I)
611*          BD(2,2)=X2(I)
612*          BD(2,3)=1.
613*          BD(3,1)=Z3(I)
614*          BD(3,2)=X3(I)
615*          BD(3,3)=1.
616*          CALL GOR(BD,3,3,3,3,$346,JC,V)
617*          GO TO 349
618*   C
619*    346   DETM3=0.
620*          GO TO 402
621*    349   DET3A=EXP(V(2))
622*          DETM3=SIGN(DET3A,V(1))
623*          IF(1.E-5.GT.ABS(DETM3)) DETM3=0.
624*   C
625*    402   V(1)=2.
626*          CD(1,1)=X1(I)
627*          CD(1,2)=Y1(I)
628*          CD(1,3)=1.
629*          CD(2,1)=X2(I)
630*          CD(2,2)=Y2(I)
631*          CD(2,3)=1.
632*          CD(3,1)=X3(I)
633*          CD(3,2)=Y3(I)
634*          CD(3,3)=1.
635*          CALL GOR(CD,3,3,3,3,$347,JC,V)
636*          GO TO 352
637*   C
638*    347   DETM4=0.
639*          GO TO 5567
640*    352   DET4A=EXP(V(2))
641*          DETM4=SIGN(DET4A,V(1))
642*          IF(1.E-5.GT.ABS(DETM4)) DETM4=0.
643*   C
644*   C
645*   C                ONE INTERCEPT IS FINITE, D IS NOT ZERO
646*   C
647*    5567  LLK=IABS(ILK(L))
*DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
648*          IF(DETM.NE.0..AND.DETM2.NE.0..AND.DETM3.EQ.0..AND.DETM4.EQ.0.) GO
649*        . TO 382
650*          GO TO 385
651*   C
652*   C                X INTERCEPT IS FINITE
653*   C
654*    382   AI=DETM/DETM2
655*          IF(LLK.LE.3) GO TO 362
656*   C
657*          IF(LLK.EQ.4.OR.LLK.EQ.6) GO TO 391
658*          IF(LLK.EQ.5.OR.LLK.EQ.7) GO TO 392
659*          IF(LLK.EQ.15.OR.LLK.EQ.16) GO TO 393
660*          GO TO 2051
661*    391   IF(A(L).GT.ABS(AI)) GO TO 1000
662*          GO TO 2051
663*    392   IF(R(L).GT.ABS(AI)) GO TO 1000
664*          GO TO 2051
665*    393   IF(GMAX(L).GT.AI.AND.AI.GT.0.) GO TO 1000
666*          GO TO 2051
667*   C
668*    362   IF(LLK.EQ.1) GO TO 363
669*          IF(LLK.EQ.2) GO TO 364
670*          IF(LLK.EQ.3) GO TO 365
671*   C
672*    363   IF(GMAX(L).GT.AI.AND.AI.GT.0.) GO TO 1000
673*          GO TO 2051
674*    364   IF(BMAX(L).GT.ABS(AI)) GO TO 1000
675*          GO TO 2051
676*   C
677*   C                TRANSFORM TRIANGLE/TRAPEZOID COORDINATES OF THE
678*   C                INTERSECTED FIGURE
679*   C
680*    365   XDIF=SX2(L)-RX(L)
681*          YDIF=SY2(L)-RY(L)
682*          ZDIF=SZ2(L)-RZ(L)
683*          SX2(L)=XDIF*P11(L)+YDIF*P12(L)+ZDIF*P13(L)
684*   C
685*          XDIF=SX3(L)-RX(L)
686*          YDIF=SY3(L)-RY(L)
687*          ZDIF=SZ3(L)-RZ(L)
688*          SX3(L)=XDIF*P11(L)+YDIF*P12(L)+ZDIF*P13(L)
689*   C
690*          XDIF=SX4(L)-RX(L)
691*          YDIF=SY4(L)-RY(L)
692*          ZDIF=SZ4(L)-RZ(L)
693*          SX4(L)=XDIF*P11(L)+YDIF*P12(L)+ZDIF*P13(L)
694*   C
695*          XDIF=SX5(L)-RX(L)
696*          YDIF=SY5(L)-RY(L)
697*          ZDIF=SZ5(L)-RZ(L)
698*          SX5(L)=XDIF*P11(L)+YDIF*P12(L)+ZDIF*P13(L)
699*   C
700*   C
*DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
701*          IF(SX4(L).EQ.0..AND.SY4(L).EQ.0..AND.SZ4(L).EQ.0.) GO TO 503
702*          IF(SX4(L).GT.0..AND.SX5(L).GT.0.) GO TO 367
703*          GO TO 368
704*    367   IF(SX2(L).GT.AI.AND.AI.GT.SX4(L)) GO TO 1000
705*          GO TO 2051
706*    368   IF(SX4(L).LT.0..AND.SX5(L).GT.0.) GO TO 369
707*          GO TO 371
708*    369   IF(SX2(L).GT.AI.AND.AI.GT.SX3(L)) GO TO 1000
709*          GO TO 2051
710*    371   IF(SX4(L).LT.0..AND.SX5(L).LT.0.) GO TO 372
711*          GO TO 2051
712*    372   IF(SX5(L).GT.AI.AND.AI.GT.SX3(L)) GO TO 1000
713*          GO TO 2051
714*    503   IF(SX2(L).GT.0..AND.SX3(L).GT.0.) GO TO 373
715*          GO TO 374
716*    373   IF(SX2(L).GT.AI.AND.AI.GT.0.) GO TO 1000
717*          GO TO 2051
718*    374   IF(SX2(L).GT.0..AND.SX3(L).LT.0.) GO TO 375
719*          GO TO 376
720*    375   IF(SX2(L).GT.AI.AND.AI.GT.SX3(L)) GO TO 1000
721*          GO TO 2051
722*    376   IF(SX2(L).LT.0..AND.SX3(L).LT.0.) GO TO 377
723*          GO TO 2051
724*    377   IF(AI.LT.0..AND.AI.GT.SX3(L)) GO TO 1000
725*          GO TO 2051
726*   C
727*   C
728*   C                Y INTERCEPT IS FINITE, D IS NOT ZERO
729*   C
*DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
730*    385   IF(DETM.NE.0..AND.DETM2.EQ.0..AND.DETM3.NE.0..AND.DETM4.EQ.0.) GO
731*        . TO 383
732*          GO TO 386
733*    383   BI=DETM/DETM3
734*          IF(LLK.LE.3) GO TO 394
735*          IF(LLK.EQ.4.OR.LLK.EQ.6) GO TO 395
736*          IF(LLK.EQ.5.OR.LLK.EQ.7) GO TO 396
737*          IF(LLK.EQ.15.OR.LLK.EQ.16) GO TO 397
738*    395   IF(A(L).GT.ABS(BI)) GO TO 1000
739*          GO TO 2051
740*    396   IF(R(L).GT.ABS(BI)) GO TO 1000
741*          GO TO 2051
742*    397   IF(BMAX(L).GT.BI.AND.BI.GT.0.) GO TO 1000
743*          GO TO 2051
744*    394   IF(LLK.EQ.1) GO TO 398
745*          IF(LLK.EQ.2) GO TO 399
746*          IF(LLK.EQ.3) GO TO 401
```

FIGURE 6 - EANCAL LISTING (CONTINUED)

```
 747*    398   IF(BMAX(L).GT.BI.AND.BI.GT.0.) GO TO 1000
 748*          GO TO 2051
 749*    399   IF(BMAX(L).GT.ABS(BI)) GO TO 1000
 750*          GO TO 2051
 751*    401   IF(BMAX(L).GT.BI.AND.BI.GT.BMIN(L)) GO TO 1000
 752*          GO TO 2051
 753*   C
 754*   C              Z INTERCEPT IS FINITE, D IS NOT ZERO
 755*   C
*DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
 756*    386   IF(DETM.NE.0..AND.DETM2.EQ.0..AND.DETM3.EQ.0..AND.DETM4.NE.0.) GO
 757*        . TO 384
 758*          GO TO 504
 759*    384   CI=DETM/DETM4
 760*          IF(LLK.EQ.4) GO TO 601
 761*          IF(LLK.EQ.5) GO TO 601
 762*          IF(LLK.EQ.7) GO TO 601
 763*          IF(LLK.EQ.15.OR.LLK.EQ.16) GO TO 602
 764*          IF(LLK.EQ.6) GO TO 388
 765*          GO TO 2051
 766*    601   IF(BMAX(L).GT.CI.AND.CI.GT.0.) GO TO 1000
 767*          GO TO 2051
 768*    602   IF(A(L).GT.CI.AND.CI.GT.0.) GO TO 1000
 769*          GO TO 2051
 770*    388   IF(A(L).GT.ABS(CI)) GO TO 1000
 771*          GO TO 2051
 772*   C              AD IS NOT EQUAL TO ZERO, ALL OTHERS ARE ZERO
 773*   C
*DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
 774*    504   IF(DETM.EQ.0..AND.DETM3.EQ.0..AND.DETM4.EQ.0.) GO TO 505
 775*          GO TO 506
 776*    505   IF(LLK.EQ.1) GO TO 2051
 777*          IF(LLK.EQ.15.OR.LLK.EQ.16) GO TO 2051
 778*          IF(LLK.NE.3) GO TO 1000
 779*   C
 780*   C              TRANSFORM X2 AND X3 COORDINATES OF
 781*   C              INTERSECTED TRIANGLE/TRAPEZOID
 782*   C
 783*    605   XDIF=SX2(L)-RX(L)
 784*          YDIF=SY2(L)-RY(L)
 785*          ZDIF=SZ2(L)-RZ(L)
 786*          SX2(L)=XDIF*P11(L)+YDIF*P12(L)+ZDIF*P13(L)
 787*   C
 788*          XDIF=SX3(L)-RX(L)
 789*          YDIF=SY3(L)-RY(L)
 790*          ZDIF=SZ3(L)-RZ(L)
 791*          SX3(L)=XDIF*P11(L)+YDIF*P12(L)+ZDIF*P13(L)
 792*          IF(SX2(L).GT.0..AND.SX3(L).LT.0.) GO TO 1000
 793*          GO TO 2051
 794*   C
 795*   C              BD IS NOT EQUAL TO ZERO, ALL OTHERS ARE ZERO
 796*   C
*DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
 797*    506   IF(DETM.EQ.0..AND.DETM2.EQ.0..AND.DETM4.EQ.0.) GO TO 507
 798*          GO TO 508
 799*    507   IF(LLK.EQ.1) GO TO 2051
 800*          IF(LLK.EQ.3) GO TO 2051
 801*          IF(LLK.EQ.15.OR.LLK.EQ.16) GO TO 2051
 802*          GO TO 1000
 803*   C
 804*   C              CD IS NOT EQUAL TO ZERO, ALL OTHERS ARE ZERO
 805*   C
*DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
 806*    508   IF(DETM.EQ.0..AND.DETM2.EQ.0..AND.DETM3.EQ.0.) GO TO 509
 807*          GO TO 403
 808*    509   IF(LLK.EQ.6) GO TO 1000
 809*          GO TO 2051
 810*   C
 811*   C              TWO INTERCEPTS ARE FINITE
 812*   C              DETERMINE IF AD=0. AND C=0.
 813*   C
*DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
 814*    403   IF(DETM.EQ.0..AND.DETM2.EQ.0.) GO TO 422
 815*          GO TO 423
 816*    422   IF(LLK.EQ.1.OR.LLK.EQ.3) GO TO 2051
 817*          IF(LLK.EQ.2) GO TO 1000
 818*          IF(LLK.EQ.4) GO TO 1000
 819*          IF(LLK.EQ.6) GO TO 1000
 820*          IF(LLK.EQ.7) GO TO 1000
 821*          IF(LLK.EQ.5) GO TO 424
 822*          IF(LLK.EQ.15.OR.LLK.EQ.16) GO TO 425
 823*   C
 824*    424   ATNYOZ=ATAN(ABS(Y1(I))/ABS(Z1(I)))
 825*          IF(ATNYOZ.LT.A(L)) GO TO 1000
 826*          GO TO 2051
 827*    425   IF(Y1(I).GT.0..AND.Z1(I).GT.0.) GO TO 1000
 828*          IF(Y1(I).LT.0..AND.Z1(I).LT.0.) GO TO 1000
 829*          GO TO 2051
 830*   C
 831*   C              DETERMINE IF D AND BD ARE ZERO
 832*   C
*DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
 833*    423   IF(DETM.EQ.0..AND.DETM3.EQ.0.) GO TO 426
 834*          GO TO 427
 835*   C
 836*    426   IF(LLK.EQ.1) GO TO 2051
 837*          IF(LLK.EQ.3) GO TO 605
 838*          IF(LLK.EQ.2) GO TO 1000
 839*          IF(LLK.EQ.4) GO TO 1000
 840*          IF(LLK.EQ.6) GO TO 1000
 841*          IF(LLK.EQ.7) GO TO 1000
 842*          IF(LLK.EQ.5) GO TO 428
 843*          IF(LLK.EQ.15.OR.LLK.EQ.16) GO TO 429
 844*   C
 845*    428   ATNXOZ=ATAN(ABS(X1(I))/ABS(Z1(I)))
 846*          IF(A(L).GT.ATNXOZ) GO TO 1000
 847*          GO TO 2051
 848*    429   IF(X1(I).GT.0..AND.Z1(I).GT.0.) GO TO 1000
 849*          IF(X1(I).LT.0..AND.Z1(I).LT.0.) GO TO 1000
 850*          GO TO 2051
 851*   C
 852*   C              DETERMINE IF D AND CD ARE ZERO
 853*   C
*DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
 854*    427   IF(DETM.EQ.0..AND.DETM4.EQ.0.) GO TO 432
 855*          GO TO 433
 856*    432   IF(LLK.EQ.1) GO TO 430
 857*          IF(LLK.EQ.15.OR.LLK.EQ.16) GO TO 499
 858*          IF(LLK.EQ.3) GO TO 430
 859*          GO TO 1000
 860*   C
 861*    430   IF(X1(I).GT.0..AND.Y1(I).GT.0.) GO TO 431
 862*          IF(X1(I).LT.0..AND.Y1(I).LT.0.) GO TO 431
 863*          GO TO 498
 864*   C
 865*    431   TR2TAN=ATAN2(ABS(X1(I)),ABS(Y1(I)))
 866*          IF(GMIN(L).LT.TR2TAN.AND.TR2TAN.LT.GMAX(L)) GO TO 1000
 867*          GO TO 2051
 868*   C
 869*    499   IF(X1(I).GT.0..AND.Y1(I).GT.0.) GO TO 1000
 870*          IF(X1(I).LT.0..AND.Y1(I).LT.0.) GO TO 1000
 871*          GO TO 2051
 872*   C
 873*    498   TRTAN=ATAN2(-ABS(X1(I)),ABS(Y1(I)))
 874*          IF(GMIN(L).LT.TRTAN.AND.TRTAN.LT.GMAX(L)) GO TO 1000
 875*          GO TO 2051
 876*   C
 877*   C              IF AD IS ZERO, CALCULATE BI AND CI
 878*   C
*DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
 879*    433   IF(DETM2.EQ.0.) GO TO 404
 880*   C
 881*   C              IF BD IS ZERO, CALCULATE AI AND CI
 882*   C
*DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
 883*          IF(DETM3.EQ.0.) GO TO 405
 884*   C
 885*   C              IF CD IS ZERO, CALCULATE AI AND BI
 886*   C
*DIAGNOSTIC*  THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
 887*          IF(DETM4.EQ.0.) GO TO 406
 888*   C
 889*   C              IF ALL INTERCEPTS ARE FINITE, CALCULATE
 890*   C              AI,BI,CI
 891*   C
 892*    434   AI=DETM/DETM2
 893*          BI=DETM/DETM3
 894*          CI=DETM/DETM4
 895*          GO TO 500
 896*   C
 897*    404   BI=DETM/DETM3
 898*          CI=DETM/DETM4
 899*          GO TO 407
 900*    405   AI=DETM/DETM2
 901*          CI=DETM/DETM4
 902*          GO TO 413
 903*    406   AI=DETM/DETM2
 904*          BI=DETM/DETM3
 905*          GO TO 421
 906*    407   IF(LLK.EQ.3) GO TO 394
 907*    415   IF(LLK.EQ.4) GO TO 408
 908*          IF(LLK.EQ.5) GO TO 409
 909*          IF(LLK.EQ.6) GO TO 410
 910*          IF(LLK.EQ.7) GO TO 409
 911*          IF(LLK.EQ.15.OR.LLK.EQ.16) GO TO 412
 912*   C
 913*    408   ACVBI=A(L)/ABS(BI)
 914*          ACVBIM=1.-ACVBI
 915*          CACVM=CI*ACVBIM
 916*          IF(CACVM.GT.0..AND.CACVM.LT.BMAX(L)) GO TO 1000
 917*          GO TO 2051
 918*    409   BIPRC=ABS(BI)*(1.-BMAX(L)/CI)
 919*          IF(R(L).GT.BIPRC) GO TO 1000
 920*          GO TO 2051
 921*    410   SPERC=ABS(BI*CI)/SQRT(BI**2+CI**2)
 922*          IF(A(L).GT.SPERC) GO TO 1000
 923*          GO TO 2051
 924*   C
 925*    412   IF(BMAX(L).GT.BI.AND.BI.GT.0.) GO TO 1000
 926*          IF(A(L).GT.CI.AND.CI.GT.0.) GO TO 1000
 927*          PRSMP=BI*(1.-A(L)/CI)
 928*          IF(BMAX(L).GT.PRSMP.AND.PRSMP.GT.0.) GO TO 1000
 929*          GO TO 2051
 930*   C
 931*    413   IF(LLK.EQ.3) GO TO 362
 932*    414   IF(LLK.EQ.4) GO TO 416
 933*          IF(LLK.EQ.5) GO TO 417
 934*          IF(LLK.EQ.6) GO TO 418
 935*          IF(LLK.EQ.7) GO TO 417
 936*          IF(LLK.EQ.15.OR.LLK.EQ.16) GO TO 420
 937*   C
 938*    416   CYLFR=CI*(1.-A(L)/ABS(AI))
 939*          IF(BMAX(L).GT.CYLFR.AND.CYLFR.GT.0.) GO TO 1000
 940*          GO TO 2051
 941*    417   CNFR2=ABS(AI)*(1.-BMAX(L)/CI)
 942*          IF(R(L).GT.CNFR2) GO TO 1000
 943*          GO TO 2051
 944*    418   SPHFR=ABS(AI*CI)/SQRT(AI**2+CI**2)
 945*          IF(A(L).GT.SPHFR) GO TO 1000
 946*          GO TO 2051
 947*    420   IF(BMAX(L).GT.AI.AND.AI.GT.0.) GO TO 1000
 948*          IF(A(L).GT.CI.AND.CI.GT.0.) GO TO 1000
 949*          PRSMP2=AI*(1.-A(L)/CI)
 950*          IF(BMAX(L).GT.PRSMP2.AND.PRSMP2.GT.0.) GO TO 1000
 951*          GO TO 2051
 952*   C
 953*    421   IF(LLK.EQ.3) GO TO 500
 954*          IF(LLK.EQ.15.OR.LLK.EQ.16) GO TO 411
 955*          IF(LLK.EQ.4) GO TO 435
 956*          IF(LLK.EQ.6) GO TO 435
 957*          IF(LLK.EQ.5) GO TO 436
 958*          IF(LLK.EQ.7) GO TO 436
 959*   C
 960*    435   ABE=ABS(AI*BI)/SQRT(AI**2+BI**2)
 961*          IF(A(L).GT.ABE) GO TO 1000
 962*          GO TO 2051
 963*    436   ABE=ABS(AI*BI)/SQRT(AI**2+BI**2)
 964*          IF(R(L).GT.ABE) GO TO 1000
 965*          GO TO 2051
 966*    411   IF(BMAX(L).GT.AI.AND.AI.GT.0.) GO TO 1000
 967*          IF(BMAX(L).GT.BI.AND.BI.GT.0.) GO TO 1000
 968*          GO TO 419
 969*   C
 970*   C              CONSIDER OTHER FIGURES IN THE SYSTEM
 971*   C
 972*    500   CONTINUE
 973*          GO TO(306,307,308,309,310,311,312,2,2,2,2,2,2,2,313,313),LLK
 974*   C
 975*   C              SURFACE TYPE 1, RECTANGLE
 976*   C
 977*    306   IF(AI.GT.0..AND.AI.LT.BMAX(L)) GO TO 1000
 978*          IF(BI.GT.0..AND.BI.LT.BMAX(L)) GO TO 1000
 979*    419   RECFR=BI*(1.-BMAX(L)/AI)
 980*          IF(BMAX(L).GT.RECFR.AND.RECFR.GT.0.) GO TO 1000
 981*          GO TO 2051
 982*   C
 983*   C              SURFACE TYPE 2, DISC
 984*   C
 985*    307   AB=ABS(AI*BI)/SQRT(AI**2+BI**2)
 986*          IF(BMAX(L).GT.AB) GO TO 1000
 987*          GO TO 2051
 988*   C
 989*   C              SURFACE TYPE 3, TRIANGLE/TRAPEZOID
```

FIGURE 6 - EANCAL LISTING (CONTINUED)

```
990*   C
991*    308   ABCVX=(AI*BI)/(BI*TAN(BMAX(L)/RTOD)+AI)
992*          ABCVN=(AI*BI)/(BI*TAN(BMIN(L)/RTOD)+AI)
993*    327   IF(BMAX(L).GT.ABCVX.AND.ABCVX.GT.BMIN(L)) GO TO 1000
994*          IF(BMAX(L).GT.ABCVN.AND.ABCVN.GT.BMIN(L)) GO TO 1000
995*          GO TO 2051
996*   C
997*   C              SURFACE TYPE 4, CYLINDER
998*   C
999*    309   IF(CI.GT.0.) GO TO 441
1000*         VC=SQRT(AI**2+BI**2)/ABS(AI*BI)
1001*         CYLD=1./VC
1002*         IF(R(L).GT.CYLD) GO TO 1000
1003*         GO TO 2051
1004*   C
1005*   C              SURFACE TYPE 5, CONE
1006*   C
1007*    310   IF(CI.GT.0.) GO TO 441
1008*         VC=SQRT(AI**2+BI**2)/ABS(AI*BI)
1009*         CIRVH=1./(1./CI+(R(L)*VC)/H(L))
1010*         IF(CIRVH.GT.0..AND.CIRVH.LT.H(L)) GO TO 1000
1011*         GO TO 2051
1012*   C
1013*   C              SURFACE TYPE 6, SPHERE
1014*   C
1015*    311   DSQR=SQRT((AI*BI)**2+(AI*CI)**2+(BI*CI)**2)
1016*         SPFRC=ABS(AI*BI*CI)/DSQR
1017*         IF(A(L).GT.SPFRC) GO TO 1000
1018*         GO TO 2051
1019*   C
1020*   C              SURFACE TYPE 7, PARABOLOID
1021*   C
1022*    312   IF(CI.GT.0.) GO TO 441
1023*         VC=SQRT(AI**2+BI**2)/ABS(AI*BI)
1024*         XHCRV=(4.*H(L))/(ABS(CI)*R(L)**2+VC**2)
1025*         IF(XHCRV.GT.1.) GO TO 2051
1026*         SRHVCR=1.-SQRT(1.+(4.*H(L))/(CI*R(L)**2*VC**2))
1027*         FARNFR=(-CI*VC*R(L)**2)/(2.*H(L))
1028*         FARWH=FARNFR*SRHVCR
1029*         IF(FARWH.GT.0..AND.FARWH.LT.R(L)) GO TO 1000
1030*         GO TO 2051
1031*   C
1032*    441   VC=SQRT(AI**2+BI**2)/ABS(AI*BI)
1033*         HRVH(L)/(1.-R(L)*VC)
1034*         IF(CI.LT.0..AND.CI.GT.HRV) GO TO 1000
1035*         GO TO 2051
1036*   C
1037*   C              SURFACE TYPES 15 AND 16, RECTANGULAR PRISM
1038*   C
1039*    313   IF(AI.GT.0..AND.BI.GT.0..AND.CI.GT.0.) GO TO 331
1040*         IF(AI.LT.0..AND.BI.LT.0..AND.CI.LT.0.) GO TO 2051
1041*         IF(AI.GT.0..AND.BI.GT.0..AND.CI.LT.0.) GO TO 335
1042*         IF(AI.LT.0..AND.BI.LT.0..AND.CI.GT.0.) GO TO 336
1043*         IF(AI.GT.0..AND.BI.LT.0.) GO TO 332
1044*         IF(AI.LT.0..AND.BI.GT.0.) GO TO 333
1045*   C
1046*    331   ZI=CI*(1.-BMAX(L)/AI-BMIN(L)/BI)
1047*         IF(ZI.GT.0..AND.ZI.LT.A(L)) GO TO 1000
1048*         GO TO 2051
1049*    332   IF(BMAX(L).GT.AI) GO TO 1000
1050*         GO TO 2051
1051*    333   IF(BMAX(L).GT.BI) GO TO 1000
1052*    335   CPF=(AI*BMAX(L)+BI*BMAX(L))/SQRT(AI**2+BI**2)
1053*         CPC=(AI*BI)/SQRT(AI**2+BI**2)
1054*         IF(CPF.GT.CPC) GO TO 1000
1055*         GO TO 2051
1056*    336   IF(CI.LT.A(L)) GO TO 1000
1057*         GO TO 2051
1058*   C
1059*   C
1060*   C
1061*    1000  ISV=I
1062*          LSV=L
1063*          DO 9999 LH=1,ITNS
1064*          IF(ISV.EQ.LH) ISV1=NSAVE(LH)
1065*    9999  IF(LSV.EQ.LH) LSV1=NSAVE(LH)
1066*          WRITE(6,340) ISV1,LSV1,,ISV1,LSV1,ISV1
1067*    340   FORMAT(//' &&&THE PLANE OF SURFACE ',I2,' INTERSECTS SURFACE ',I2,
1068*         .',',//,' &&&ANOTHER VIEW FACTOR FROM SURFACE ',I2,' TO SURFACE ',I2
1069*         .,' MAY BE OBTAINED BY USING THE OTHER VALUE OF ILK WITH SURFACE ',
1070*         .I2,',',)
1071*          GO TO 2050
1072*   C
1073*    2051  ISV=I
1074*          LSV=L
1075*          DO 2052 LH=1,ITNS
1076*          IF(ISV.EQ.LH) ISV1=NSAVE(LH)
1077*    2052  IF(LSV.EQ.LH) LSV1=NSAVE(LH)
1078*   C
1079*   C
1080*    2050  X1I=SX1(I)
1081*          X2I=SX2(I)
1082*          Y1I=SY1(I)
1083*          Y2I=SY2(I)
1084*          Z1I=SZ1(I)
1085*          Z2I=SZ2(I)
1086*          X1L=SX1(L)
1087*          X2L=SX2(L)
1088*          Y1L=SY1(L)
1089*          Y2L=SY2(L)
1090*          Z1L=SZ1(L)
1091*          Z2L=SZ2(L)
1092*          XMP=(X2I-X1I)*(X2L-X1L)
1093*          YMP=(Y2I-Y1I)*(Y2L-Y1L)
1094*          ZMP=(Z2I-Z1I)*(Z2L-Z1L)
1095*          XNUM=XMP+YMP+ZMP
1096*          SRI=SQRT((X2I-X1I)**2+(Y2I-Y1I)**2+(Z2I-Z1I)**2)
1097*          SRL=SQRT((X2L-X1L)**2+(Y2L-Y1L)**2+(Z2L-Z1L)**2)
1098*          XDEN=SRI*SRL
1099*          XFRCT=XNUM/XDEN
1100*   C
1101*   C
1102*          IF=2*ITNS-2
1103*          II=II+2
1104*          III=II+1
1105*          ILSAV(II)=ISV1
1106*          ILSAV(III)=LSV1
1107*   C
1108*          DO 2054 IR=1,II,2
1109*          IF(ILSAV(IR).EQ.ILSAV(III)) GO TO 319
1110*    2054  CONTINUE
1111*   C
1112*    2055  IF(ILK(I).EQ.6.OR.ILK(L).EQ.6) GO TO 319
1113*          IF(ILK(I).LE.3.AND.ILK(L).LE.3) GO TO 1500
1114*          IF(ILK(I).LE.3.OR.ILK(L).LE.3) GO TO 1502
1115*   C
1116*          THETA=ACOS(XFRCT)*RTOD
1117*   C
1118*          IF(ILK(I).GE.15.AND.ILK(L).GE.15) GO TO 1501
1119*          IF(ILK(I).EQ.4.OR.ILK(I).EQ.5.OR.ILK(I).EQ.7) GO TO 1509
1120*          IF(ILK(L).EQ.4.OR.ILK(L).EQ.5.OR.ILK(L).EQ.7) GO TO 1510
1121*    1500  THETA=ACOS(DETM4/SQRT(DETM2**2+DETM3**2+DETM4**2))*RTOD
1122*          WRITE(6,2000) ISV1,LSV1,THETA
1123*          GO TO 319
1124*    1502  IF(ILK(I).LE.3) GO TO 1503
1125*          IF(ILK(L).LE.3) GO TO 1505
1126*    1503  IF(ILK(L).EQ.4.OR.ILK(L).EQ.5.OR.ILK(L).EQ.7) GO TO 1504
1127*          IF(ILK(L).GE.15) GO TO 1506
1128*    1505  IF(ILK(I).EQ.4.OR.ILK(I).EQ.5.OR.ILK(I).EQ.7) GO TO 1504
1129*          IF(ILK(I).GE.15) GO TO 1506
1130*    1504  THETA=ACOS(DETM4/SQRT(DETM2**2+DETM3**2+DETM4**2))*RTOD
1131*          IF(ILK(L).LE.3) GO TO 1507
1132*          WRITE(6,2001) ISV1,LSV1,THETA
1133*          GO TO 319
1134*    1507  WRITE(6,2001) LSV1,ISV1,THETA
1135*          GO TO 319
1136*    1506  THETA=ACOS(DETM4/SQRT(DETM2**2+DETM3**2+DETM4**2))*RTOD
1137*          IF(ILK(L).LE.3) GO TO 1508
1138*          WRITE(6,2002) ISV1,LSV1,THETA
1139*          GO TO 319
1140*    1508  WRITE(6,2002) LSV1,ISV1,THETA
1141*          GO TO 319
1142*    1501  WRITE(6,2003) ISV1,LSV1,THETA
1143*          GO TO 319
1144*    1509  IF(ILK(L).EQ.4.OR.ILK(L).EQ.5.OR.ILK(L).EQ.7) GO TO 1511
1145*          IF(ILK(L).GE.15) GO TO 1512
1146*    1510  IF(ILK(I).GE.15) GO TO 1513
1147*    1511  WRITE(6,2004) ISV1,LSV1,THETA
1148*          GO TO 319
1149*    1512  WRITE(6,2005) ISV1,LSV1,THETA
1150*          GO TO 319
1151*    1513  WRITE(6,2005) LSV1,ISV1,THETA
1152*   C
1153*   C
1154*    2000  FORMAT(//' ***THE ANGLE BETWEEN THE ACTIVE FACES OF THE PLANES OF
1155*         . FIGURES ',I2,' AND ',I2,' IS ',F6.2,' DEG.FOR +ILK VALUES.')
1156*    2001  FORMAT(//' ***THE ANGLE BETWEEN THE ACTIVE FACE OF THE PLANE OF F
1157*         .IGURE ',I2,' FOR +ILK AND THE AXIS OF FIGURE ',I2,' IS ',F6.2,' DE
1158*         .G.')
1159*    2002  FORMAT(//' ***THE ANGLE BETWEEN THE ACTIVE FACE OF THE PLANE OF F
1160*         .IGURE ',I2,' FOR +ILK AND THE EDGE OF PRISM ',I2,' DEFINED BY POIN
1161*         .TS P1 AND P5 IS ',F6.2,' DEG.')
1162*    2003  FORMAT(//' ***THE ANGLE BETWEEN THE EDGES OF PRISM ',I2,' AND ',I2
1163*         .,' WHICH ARE DEFINED BY THEIR RESPECTIVE POINTS P1 AND P2 IS ',F6
1164*         .2,' DEG.')
1165*    2004  FORMAT(//' ***THE ANGLE BETWEEN THE AXES OF FIGURES ',I2,' AND ',I
1166*         .I2,' IS ',F6.2,' DEG.')
1167*    2005  FORMAT(//' ***THE ANGLE BETWEEN THE AXIS OF FIGURE ',I2,' AND THE
1168*         .EDGE OF PRISM ',I2,' DEFINED BY POINTS P1 AND P2 IS ',F6.2,' DEG.')
1169*         .)
1170*   C
1171*   C
1172*   C
1173*    319   CONTINUE
1174*   C
1175*   C
1176*    317   CONTINUE
1177*   C
1178*    2     RETURN
1179*          END
```

FIGURE 6 - EANCAL LISTING (CONTINUED)