

General Disclaimer

One or more of the Following Statements may affect this Document

- This document has been reproduced from the best copy furnished by the organizational source. It is being released in the interest of making available as much information as possible.
- This document may contain data, which exceeds the sheet parameters. It was furnished in this condition by the organizational source and is the best copy available.
- This document may contain tone-on-tone or color graphs, charts and/or pictures, which have been reproduced in black and white.
- This document is paginated as submitted by the original source.
- Portions of this document are not fully legible due to the historical nature of some of the material. However, it is the best reproduction available from the original submission.

NASA CR-92457

Contract No:

NAS 9-8166

TRW Report No:

11176-H046-RO-00

COMPUTATION OF LIGHTING AND VIEWING ANGLES

4 October 1968

Prepared for
SYSTEMS ENGINEERING DIVISION
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
MANNED SPACECRAFT CENTER
HOUSTON, TEXAS

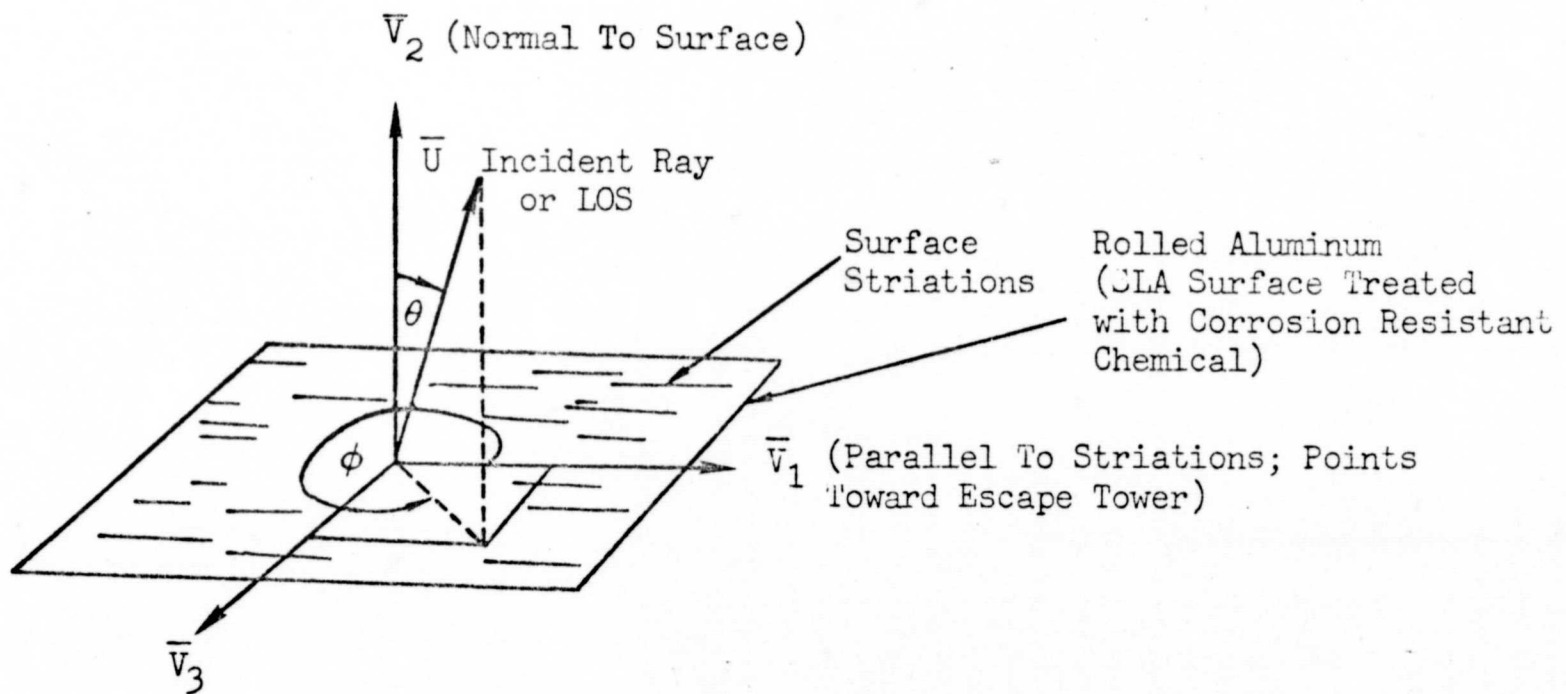
By
TRW SYSTEMS GROUP
HOUSTON, TEXAS

FACILITY FORM 602

N 69-14815	(ACCESSION NUMBER)	8	(PAGES)	1	(THRU)
NASA CR-92457	(NASA CR OR TMX OR AD NUMBER)	23	(CATEGORY)		

COMPUTATION OF LIGHTING AND VIEWING ANGLESINTRODUCTION

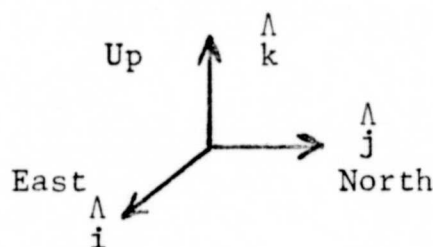
Bidirection reflectivity measurements of samples of spacecraft materials are done for a variety of lighting and photometer viewing angles. The resulting reflectivity values are presented as a function of two pairs of angles θ_1 , ϕ_1 , and θ_2 , ϕ_2 ,. The first pair defines the direction of the incident light with respect to the illuminated surface. The second pair defines the direction of the observed reflected light with respect to the sample surface. The sketch below illustrates this geometry for a sample of rolled aluminum. It should be noted, however, that the direction of vector $\bar{V}_1$, lying in the sample surface is essentially arbitrary.



This report describes a digital computer program that computes the θ , ϕ pairs necessary to utilize bidirectional reflectivity data for determining relative brightness of points on the exterior of the Apollo launch vehicle. The computations are made for a given vehicle attitude, direction of the sun or other single illumination source, and observer line-of-sight (LOS), all with respect to a fixed coordinate system, East, North and up.

EQUATIONS

Consider the triad of unit vectors $\hat{i}$, $\hat{j}$, $\hat{k}$, where $\hat{i}$ points East, $\hat{j}$ points North, and $\hat{k}$ points up (away from the earth's center).



- Let:
- (i) A be the azimuth angle of the vehicle longitudinal axis from North, positive towards the East.
 - (ii) E be the elevation angle of the vehicle longitudinal axis with respect to the $\hat{i}$, $\hat{j}$ plane, positive for vehicle nose up.
 - (iii) the vehicle roll attitude angle be zero, i.e., the vehicle pitch axis is parallel to the $\hat{i}$, $\hat{j}$ plane.
 - (iv) δ be the angle between the vehicle skin and its longitudinal axis, measured in the plane containing the longitudinal axis and the illuminated point in question, positive if adjacent points are closer to the longitudinal axis on the nose side and farther away on the tail side of the point in question.
 - (v) A_c be the azimuth of the observer LOS as viewed from the vehicle, measured as in (i).
 - (vi) E_c be the elevation of the observer LOS as viewed from the vehicle, measured as in (ii). NOTE: E_c is negative if vehicle is higher than observer.
 - (vii) A_s be the azimuth of the sun (or other single source of parallel rays) as viewed from the vehicle, measured as in (i).
 - (viii) E_s be the elevation of the sun as viewed from the vehicle, measured as in (ii).
 - (ix) φ be the vehicle roll angular position of the plane through the longitudinal axis containing the illuminated point in question.

Then, unit vectors $\bar{V}_1, \bar{V}_2, \bar{V}_3$ can be defined in terms of these angles. The tails of these unit vectors are tied to the illuminated point in question; $\bar{V}_1$ lies in the vehicle skin, pointing towards the nose, and is contained in the plane through the longitudinal axis and the illuminated point; $\bar{V}_2$ is normal to the skin; $\bar{V}_3 = \bar{V}_1 \wedge \bar{V}_2$.

Furthermore, unit vectors $\bar{U}_c$ and $\bar{U}_s$ can be defined. These vectors also emanate from the illuminated point in question, and $\bar{U}_c$ points towards the observer, and $\bar{U}_s$ points towards the sun.

It can be shown that, where $S \equiv \text{Sine}$, $C \equiv \text{Cosine}$

$$\begin{aligned}\bar{V}_1 = & [SACEC\delta + (CAC\varphi + SAS\varphi)S\delta]i \\ & + [CACEC\delta - (SAC\varphi - CASES\varphi)S\delta]j \\ & + [SEC\delta - CES\varphi S\delta]k\end{aligned}$$

$$\begin{aligned}\bar{V}_2 = & [SACE3\delta - (CAC\varphi + SAS\varphi)C\delta]i \\ & + [CACES\delta + (SAC\varphi - CASES\varphi)C\delta]j \\ & + [SES\delta + CES\varphi C\delta]k\end{aligned}$$

$$\bar{V}_3 = \bar{V}_1 \wedge \bar{V}_2,$$

and

$$\bar{U}_c = S A_c C E_c i + C A_c C E_c j + S E_c k$$

$$\bar{U}_s = S A_s C E_s i + C A_s C E_s j + S E_s k.$$

Finally, it is evident from the first sketch that:

$$\begin{aligned}\theta_1 &= C^{-1} \bar{U}_s \cdot \bar{V}_2, & \phi_1 &= \tan^{-1} \frac{\bar{U}_s \cdot \bar{V}_3}{\bar{U}_s \cdot \bar{V}_1} \\ \theta_2 &= C^{-1} \bar{U}_c \cdot \bar{V}_2, & \phi_2 &= \tan^{-1} \frac{\bar{U}_c \cdot \bar{V}_3}{\bar{U}_c \cdot \bar{V}_1}\end{aligned}$$

PROGRAM DESCRIPTION

The RUSH language name of the program is (AGPCUR,WALKER) and Mr. D. Dartez, TRW Systems Group, Houston Operations, is the programmer.

The program computes the vectors described above and the angles θ, ϕ . The program is written in the RUSH language for use on TRW's Conversational Computing Console Terminal.

Inputs:

Input of data refers to the transfer of information to the internal memory of the computer. RUSH language statements of the input type contained in (AGPCUR,WALKER) direct information to be transferred from the typewriter console to the internal memory of the computer. After each request of the computer, the user responds by typing in the assigned value for a particular variable. The carriage on the console is then returned and another request from the computer is made. This is the sequence used in transferring data to the internal memory of the computer. For a list and description of input values, see Table I.

TABLE I.

RUSH INPUT DATA

RUSH NOTATION	MATH NOTATION	DESCRIPTION	UNITS
A	A	Azimuth from true North (positive toward the East) of spacecraft axis	Degrees
E	E	Elevation of spacecraft axis from the horizon (i, j plane)	Degrees
AS	A _s	Azimuth of the sun line-of-sight	Degrees
ES	E _s	Elevation of the sun line-of-sight	Degrees
AC	A _c	Azimuth of the camera line-of-sight	Degrees
EC	E _c	Elevation of the camera line-of-sight	Degrees
D	δ	Inclination of surface with respect to the center line of vehicle	Degrees
P	ϕ	Roll position of illuminated point	Degrees

Outputs:

An output statement specifies by name the item or items which are to be retrieved from memory, referring to a format code within the program which describes the form this information is to take on the output or printed page. For a list and description of output information, see Table II.

TABLE II.

RUSH OUTPUT DATA

RUSH NOTATION	MATH NOTATION	DESCRIPTION	UNITS
THETA1	θ_1	Incidence angle from surface normal	Degrees
PHI1	ϕ_1	Incidence azimuth angle	Degrees
THETA2	θ_2	Reflection angle from surface normal	Degrees
PHI2	ϕ_2	Reflection azimuth angle	Degrees

DIRECTION FOR USE OF "RUSH" PROGRAM (AGPCUR,WALKER)

1. o/o Login (TRWHCC, XXXXXX)
2. RUSH request user auto-save (depress attention key)
3. o/o Load (AGPCUR,WALKER)
4. o/o XEQ 1 thru ...
5. Program request A, E, AS, ES, AC, EC, D, and P.
6. Program prints Thetal, Phil, Theta2 and Phi2 associated with above D and P.
7. Program request another D and P.
8. Program prints Thetal, Phil, Theta2 and Phi2 associated with latest input value of D and P.
9. This sequence is repeated until all necessary data has been calculated.

SEE ATTACHMENT FOR A LISTING OF PROGRAM.

```

1.  PROGRAM:-(ACPCUP,WALKER)
2.  DCL V1(3,1), V2(3,1), V3(3,1), US(3,1), UC(3,1);
3.  GET LIST(A,E,AS,ES,AC,EC);
4.  GET LIST(P);
5.  V1(1,1)=SINH(A)*COSD(E)*COSD(D)+(COSD(A)*COSD(P)+SINH(A)*SINH(P))*SINH(D);
6.  V1(2,1)=COSD(A)*COSD(E)*COSD(D)-(SINH(A)*COSD(P)-COSD(A)*SINH(E)*SINH(P))*SINH(D);
7.  V1(3,1)=SINH(E)*COSD(D)-COSD(E)*SINH(P)*SINH(D);
8.  V2(1,1)=SINH(A)*COSD(F)*SINH(D)-(COSD(A)*COSD(P)+SINH(A)*SINH(E)*SINH(P))*COSD(D);
9.  V2(2,1)=COSD(A)*COSD(E)*SINH(D)+(SINH(A)*COSD(P)-COSD(A)*SINH(E)*SINH(P))*COSD(D);
10. V2(3,1)=SINH(E)*SINH(D)+COSD(E)*SINH(P)*COSD(D);
11. V3(1,1)=V1(2,1)*V2(3,1)-V1(3,1)*V2(2,1);
12. V3(2,1)=-(V1(1,1)*V2(3,1)-V1(3,1)*V2(1,1));
13. V3(3,1)=V1(1,1)*V2(2,1)-V1(2,1)*V2(1,1);
14. UC(1,1)=SINH(AC)*COSD(EC);
15. UC(2,1)=COSD(AC)*COSD(EC);
16. UC(3,1)=SINH(EC);
17. US(1,1)=SINH(AS)*COSD(ES);
18. US(2,1)=COSD(AS)*COSD(ES);
19. US(3,1)=SINH(ES);
20. COTH1=US(1,1)*V2(1,1)+US(2,1)*V2(2,1)+US(3,1)*V2(3,1);
21. PHI2=ATANH(-(UC(1,1)*V3(1,1)+UC(2,1)*V3(2,1)+UC(3,1)*V3(3,1))/(UC(1,1)*V1(1,1)+UC(2,1)*V1(2,1)+UC(3,1)*V1(3,1));
22. COTH2=UC(1,1)*V2(1,1)+UC(2,1)*V2(2,1)+UC(3,1)*V2(3,1);
23. PHI1=ATANH(-(US(1,1)*V3(1,1)+US(2,1)*V3(2,1)+US(3,1)*V3(3,1))/(US(1,1)*V1(1,1)+US(2,1)*V1(2,1)+US(3,1)*V1(3,1));
24. UCV3=UC(1,1)*V3(1,1)+UC(2,1)*V3(2,1)+UC(3,1)*V3(3,1);
25. USV3=US(1,1)*V3(1,1)+US(2,1)*V3(2,1)+US(3,1)*V3(3,1);
26. UCV1=UC(1,1)*V1(1,1)+UC(2,1)*V1(2,1)+UC(3,1)*V1(3,1);
27. USV1=US(1,1)*V1(1,1)+US(2,1)*V1(2,1)+US(3,1)*V1(3,1);
28. IF UCV3>0. THEN GO TO TEST1;
29. IF UCV1>0. THEN GO TO SECOND;
30. PHI2=PHI2+180.;
31. GO TO SECOND;
32. TEST1: IF UCV1>0. THEN GO TO SECOND;
33. PHI2=PHI2-180.;
34. GO TO SECOND;
35. SECOND: IF USV3>0. THEN GO TO TEST2;
36. IF USV1>0. THEN GO TO PRINT;
37. PHI1=PHI1+180.;
38. GO TO PRINT;
39. TEST2: IF USV1>0. THEN GO TO PRINT;
40. PHI1=PHI1-180.;
41. THETA1=ATANH(SQRT(1/(COTH1*COTH1)-1));
42. THETA2=ATANH(SQRT(1/(COTH2*COTH2)-1));
43. PUT IMAGE(THETA1,PHI1,THETA2,PHI2)(PHI)SKIP(2);
44. IMAGE;
45. THETA1=----- PHI1=----- THETA2=----- PHI2=-----
    GO TO START;
    6120 END OF PROGRAM LISTING

```