

N71-35389



DBA SYSTEMS, INC.

DOCUMENTATION OF PHOTOGRAMMETRIC
MISSION SIMULATOR PROGRAM

VOLUME 4
MISSION ANALYSIS MODULE

CASE FILE
COPY

Photogrammetry

Computer Services

Mathematical Research

Electro-Optical Instrumentation

GENERAL OFFICES
Post Office Drawer 550
Melbourne, Florida 32901
Phone: 305 727-0660

ELECTRO-OPTICAL DIV.
Post Office Drawer 550
Melbourne, Florida 32901
Phone: 305 727-2020

WASH. D.C. OFFICE
9301 Annapolis Road
Lanham, Maryland 20801
Phone: 301 577-3001

CR-115174

Final Technical Report

DOCUMENTATION OF PHOTOGRAMMETRIC
MISSION SIMULATOR PROGRAM
VOLUME 4
MISSION ANALYSIS MODULE

August 1971

Prepared By:

DBA Systems, Inc.
P.O. Drawer 550
Melbourne, Florida 32901

Prepared For:

NASA Manned Spacecraft Center
Mapping Science Laboratory
Houston, Texas 77058

Under:

Contract Number: NAS9-10256

TABLE OF CONTENTS

	<u>Page</u>
1.0 INTRODUCTION	1
1.1 Ground Coverage Computations	1
1.2 Exposure Computations	1
2.0 MATHEMATICAL DEVELOPMENT	
2.1 Data Simulator	2
2.2 Exposure Computations	2
2.2.1 Computation of Exposure for a Single Target	3
2.2.2 Selection of Optimum Exposure for an Entire Frame	19
3.0 PROGRAM USER PROCEDURES	21
3.1 System Control Deck Setup Description	21
3.2 Input Data Description	21
3.2.1 Data Deck Setup	21
3.2.2 Input Card Description	23
3.3 Output Description	23
4.0 PROGRAM DESCRIPTIONS	37
4.1 Main Control Program - MANCE	37
4.2 Ground Coverage Program - MISSIM	37
4.3 Exposure Computation Program - APETUR	37
4.4 Utility Subroutines	43

LIST OF FIGURES

	<u>Page</u>
FIGURE 2.1 Albedo Table Illustration	6
FIGURE 2.2 Reflection Characteristics of the Earth and the Moon	8
FIGURE 2.3 Geometry of the Phase Angle	10
FIGURE 2.4 Geometry of Luminance Longitude	15
FIGURE 2.5 Fedorets' Photometric Curves	17
FIGURE 4.1 Tree Content Diagram Mission Analysis Module	38
FIGURE 4.2 Flow Diagram for Main Control Program Mission Analysis Module	39
FIGURE 4.3 Flow Diagram of Simulation Program - MISSIM Mission Analysis Module	40
FIGURE 4.4 Flow Diagram of Exposure Computations - APETUR Mission Analysis Module	42

1.0

INTRODUCTION

The Mission Analysis Module, acronym MANCE, is a general purpose simulator designed to assist NASA-MSC in planning photogrammetric missions of the lunar surface. This module has the capability of determining the theoretical coverage of frame photography acquired during a lunar orbiting mission. In addition, a number of exposure computations are performed for the purpose of determining an optimum exposure for each photograph.

1.1

Ground Coverage Computations

This sub-module is capable of simulating a wide variety of photogrammetric schemes. For global missions, the input to the sub-module is generated by the gravity recovery module, acronym RSTATE. For short arc or multi-mission simulations, the input to MANCE is by means of punched cards.

1.2

Exposure Computations

Exposure is defined as the quantity of light which impinges upon the photographic emulsion. The exposure at any given point in image space is dependent on a number of factors, but can be simply expressed as the product of illuminance and time (shutter speed).

Of all the factors involved in determining exposure, there are only two camera-setting variables. These are the lens aperture and the exposure duration (shutter speed). In selecting the settings for taking a photograph, the shutter speed will be chosen on the basis of image motion consideration. The aperture is then chosen in a manner which will result in an acceptable exposure.

A required input for these computations is a range of exposure values called the optimum exposure range. This range corresponds to all exposures within one half stop either side of optimum film exposure, where optimum exposure is a characteristic of the specified film.

Exposures are computed for a pre-selected array of points over the photographic plate, and for each of the specified T-values (calibrated f/ numbers). Thus for each T-value, there is a distinct numerical value for exposure for every array point. A minimum, maximum and mean exposure are obtained for each T-value. If the minimum, maximum or mean exposure so determined is within the optimum film exposure range, then the T-value associated with that exposure is said to be an optimum exposure setting. It is possible that there will be no optimum exposure for a given photograph.

2.0 MATHEMATICAL DEVELOPMENT

The Mission Analysis Module consists of two sub-modules. The first sub-module generates the photogrammetric data and performs the ground coverage computations. The second performs all exposure computations.

2.1 Data Simulator

The mathematical development of the data generator and ground coverage computations is the same as that used for the Multi-Mission Block Adjustment Module, MBLOCK. This development is given in Volume 3, Section 2.1, and will not be repeated here.

2.2 Exposure Computations

The exposure computations performed by MANCE utilize parameters describing the luminance characteristics of the target, the location of the sun, the exposure duration (shutter speed), the optimum exposure range of the film, and relevant camera characters. These parameters are defined in the program in the form of pre-stored empirical constants, mathematical relationships, and input data.

Computation of the exposure of a single target will be presented first, followed by a description of the manner in which optimum exposures for an entire frame are obtained.

2.2.1 Computation of Exposure for a Single Target

2.2.1.1 The Basic Equation

The exposure (E) is computed from the well-known formula:

$$E = B(\rho, \psi) V \delta t / 4T \quad (1)$$

where,

$B(\rho, \psi)$	is the brightness (given in meter-candles) as a function of the albedo (ψ) and the photometric constant (ρ) at the target point
V	is the irradiance of the camera system (percent transmittance of the lens system, including the spacecraft window)
δt	is the duration of exposure
T	is the calibrated f/number (input to program).

2.2.1.2 Computation of Exposure Duration

The exposure duration required for lunar orbiting photography is a function of the camera attitude, altitude, speed, IMC mechanism and the maximum allowable image smear.

The equations for computing the exposure duration are found by differentiating the projective equations with respect to time. These differentials are:

$$\frac{dx}{dt} = -\frac{cA + xD}{fh^c} V^c - \frac{cV^c}{h^c} P \quad (2)$$

$$\frac{dy}{dt} = -\frac{cA' + yD}{fh^c} V^c$$

where,

c	is the calibrated focal length of the camera
x, y	are the image coordinates
h^c	is the satellites altitude
V^c	is the satellites speed
P	is the percentage of effective image motion compensation
A, A', D, F	are elements of the local vertical orientation matrix given by

$$A = \cos \hat{\phi}$$

$$A' = -\sin \hat{\phi} \sin \hat{\omega}$$

$$D' = -\sin \hat{\phi} \cos \hat{\omega}$$

$$F = \cos \hat{\phi} \cos \hat{\omega}$$

where $\hat{\phi}$ and $\hat{\omega}$ are the local vertical pitch and roll of the camera. The exposure duration (δt) is computed using the maximum $\frac{dx}{dt}$ or $\frac{dy}{dt}$ for all image points on a photograph. This maximum (m) is then used to compute δt by:

$$\delta t = \frac{\Delta r}{m} \quad (3)$$

where,

Δr denotes the maximum allowable image smear.

For vertical photography equations (2) and (3) reduce to the following simplified form:

$$\delta t = \frac{h^c}{cV^c(1-P)} \Delta r \quad (4)$$

2.2.1.3 Computation of Brightness

As outlined in Section 2.2.1.2 brightness is a function of two parameters ρ and ψ (the albedo constant and the photometric constant). The relationship is:

$$B(\rho, \psi) = 13.4 \times 10^4 \rho \psi(\alpha, g)$$

where 13.4×10^4 is an empirically derived constant representing the average solar illumination at the lunar surface in meter-candles and

α = phase angle

g = luminance longitude angle.

The albedo and photometric constants are computed by interpolative techniques from empirically derived tables. The procedures used are outlined in the two ensuing sub-sections.

2.2.1.3.1 Computation of Albedo Constant

The albedo constant for a surface is the ratio of incident light to reflected light. Obviously this varies from point to point on the lunar surface depending on the spectral reflectivity of the targets geological characteristics. An empirical table of albedo constants has been developed for the lunar surface. These values have been tabulated in five degree (5°) increments of latitude and longitude for a 40° latitude band encircling the moon. This band lies between $+20^\circ$ and -20° latitude. The program stores the albedo constants in an internal table. The entries in this table are illustrated in Figure 2.1.

Given the latitude and longitude of a ground point the albedo constant for the point is computed by a weighted averaging scheme. For illustrative purposes, assume the target has a latitude and longitude of φ and λ , respectively. As indicated in Figure 2.1,

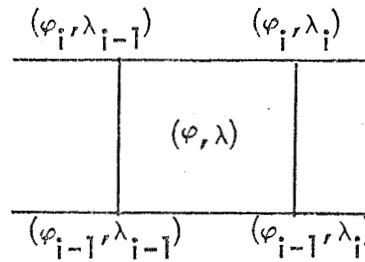


FIGURE 2.1
Albedo Table Illustration

φ_i, λ_i are the coordinates which have albedo values stored in the program. The distances between the target and each of the corners of the 5° squares are computed. These distances are used as weights to average the four albedo values. That is:

$$\rho(\varphi, \lambda) = \left(d_{i-1, i-1} + d_{i-1, i} + d_{i, i-1} + d_{i, i} \right) \cdot \left(\frac{\rho(\varphi_{i-1}, \lambda_{i-1})}{d_{i-1, i-1}} + \frac{\rho(\varphi_{i-1}, \lambda_i)}{d_{i-1, i}} + \frac{\rho(\varphi_i, \lambda_{i-1})}{d_{i, i-1}} + \frac{\rho(\varphi_i, \lambda_i)}{d_{i, i}} \right)$$

wherein,

$\rho(\varphi, \lambda)$ denotes the albedo value at latitude φ and longitude λ

d_{ik} denotes the distance between the target and the albedo corner (φ_i, λ_k) .

2.2.1.3.2 Computation of the Photometric Constant

The albedo constant gives the total percentage of incident light which is reflected irrespective of direction. The photometric constant is a percentage of reflected light in a given direction. Before describing how this photometric constant is computed, it is felt to be advantageous to digress slightly and review the nature of reflecting surfaces.

When light is incident upon a surface, the amount of reflected light in any direction is a function of the structure of the surface. For instance, a plane mirror reflects all incident light such that the angle of incidence equals the angle of reflectance. Such a surface is called specular. If the incident light is completely diffuse such that the amount of reflected light is independent of direction the surface is called Lambertian. A sheet of white paper is an example of a Lambertian surface. Figure 2.2 depicts the reflective characteristics of a Lambertian surface, of the earth, and of the moon.

Obviously the percentage of light reflected to the camera depends on the angle of incidence and the angle between the incident light ray and the line from the camera to the target, as well as the reflectivity of the target. Geometrically the relationship between the positions of the sun, the lunar target and the camera is depicted in Figure 2.3. In this figure T is the target location which has selenographic coordinates $(\varphi_T, \lambda_T, h_T)$. The latitude and longitude of the subsolar point (S) are φ_s, λ_s respectively, and the coordinates of the camera are φ^C, λ^C , and h^C . The photometric constant ψ is tabulated in matrix form in terms of two angles, namely phase angle (g) and luminance longitude (α). An interpolative scheme is used to evaluate (α, g) .

The phase angle (g) is defined as the angle between the vector from the sun to the lunar target and the vector from the target to the camera. Using vector algebra this angle is computed as:

$$\cos g = \frac{\vec{S} \cdot \vec{E}}{|\vec{E}|}$$

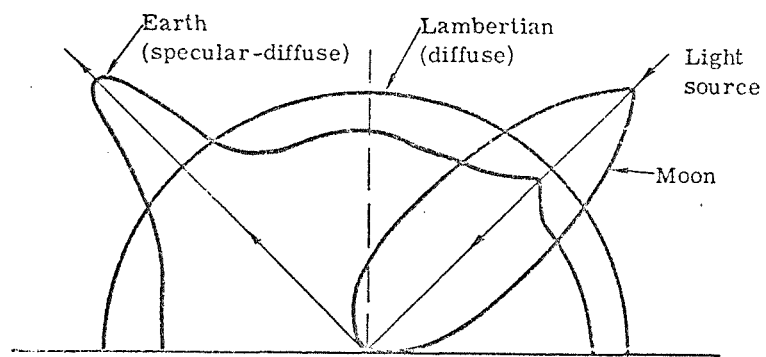


FIGURE 2.2

Reflection Characteristics of the Earth and the Moon

where,

$\vec{S}$ is the unit vector towards the sun

$\vec{E}$ is the vector from the target to the camera

$$\vec{S} = \begin{bmatrix} \cos\varphi_s \cos\lambda_s \\ \cos\varphi_s \sin\lambda_s \\ \sin\varphi_s \end{bmatrix}$$

$$\vec{E} = \vec{C} - \vec{T}.$$

$\vec{T}$ denotes the vector from the moons' center to the target and $\vec{C}$ is the vector from the lunar origin to the exposure station. These vectors are computed as:

$$\vec{C} = \frac{R + h_c}{R} \begin{bmatrix} \cos\varphi_c \cos\lambda_c \\ \cos\varphi_c \sin\lambda_c \\ \sin\varphi_c \end{bmatrix}$$

$$\vec{T} = \frac{R + h_T}{R} \begin{bmatrix} \cos\varphi_T \cos\lambda_T \\ \cos\varphi_T \sin\lambda_T \\ \sin\varphi_T \end{bmatrix}$$

where R is the lunar radius.

The selenographic latitude and longitude of the subsolar point (φ_s, λ_s) are computed from the Julian date (JD) at the time of exposing the photograph. The ephemeris of the selenographic subsolar point was curve fit in order to avoid the use of an ephemeris tape. The accuracy of these fits are completely

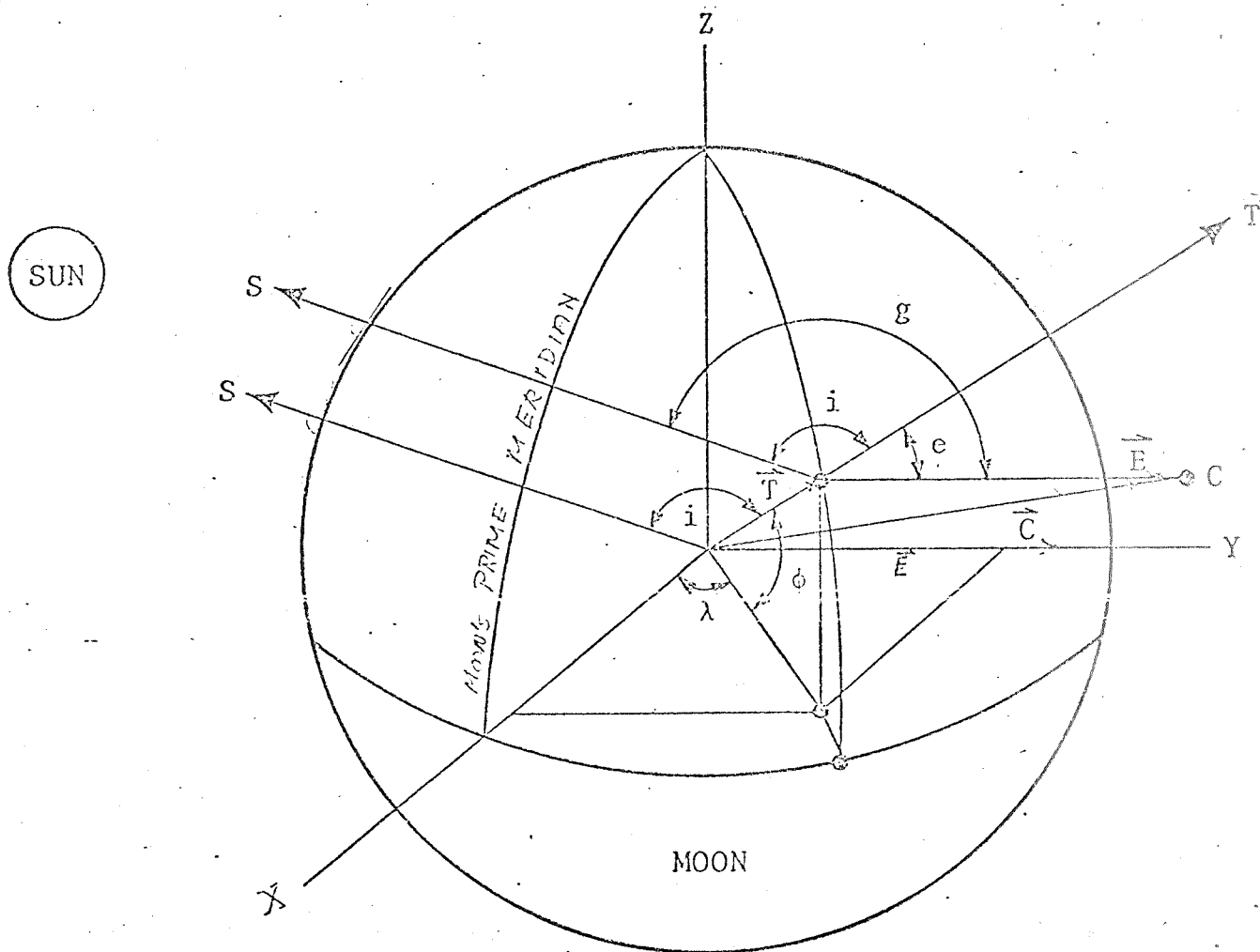


FIGURE 2.3

Geometry of the Phase Angle

adequate for exposure computations. The latitude is computed using the following algorithm:

If $2440587.5 \leq JD < 2440997.0$; then

$$\varphi_s = 1.55 \sin \left[\frac{360}{349} (JD - 2440648.5) \right]$$

If $2440997.0 \leq JD < 2441347.5$; then

$$\varphi_s = 1.56 \sin \left[\frac{360}{349.5} (JD - 2440997.0) \right]$$

If $2441347.5 \leq JD < 2441713.5$; then

$$\varphi_s = 1.56 \sin \left[\frac{360}{349.5} (JD - 2441347.5) \right]$$

If $JD < 2440587.5$ or $JD \geq 2441713.5$; then terminate the run printing an error message indicating that the solar ephemeris was only curve-fit for three one-year intervals.

Note: 1 January 1971, O^h U. T. is Julian Day No. 2440952.5

The longitude is computed by:

$$\lambda_s = m \left[JD - JD_o \right]$$

where JD is the current Julian Date, and m and JD_o are obtained from the Julian Date Table (Table 2.1) as follows:

- 1) Find JD_1 and JD_2 such that $JD_1 \leq JD \leq JD_2$;
- 2) set m equal to the tabulated value of m at JD_1 ;
- 3) set JD_o equal to JD_1 .

TABLE 2.1
Julian Date Table

<u>JD₁</u>	<u>JD₂</u>	<u>m</u>
2440572.032	2440601.643	360/29.611
601.643	631.244	360/29.601
631.244	660.818	360/29.574
660.818	690.354	360/29.536
690.354	719.851	360/29.497
719.851	749.321	360/29.470
749.321	778.776	360/29.455
778.776	808.235	360/29.459
808.235	837.713	360/29.478
837.713	867.228	360/29.515
867.228	896.791	360/29.563
896.791	926.379	360/29.588
926.379	955.989	360/29.610
955.989	985.598	360/29.609
985.598	1015.181	360/29.583
1015.181	1044.729	360/29.548
1044.729	1074.241	360/29.512
1074.241	1103.721	360/29.480
1103.721	1133.177	360/29.456
1133.177	1162.634	360/29.457
1162.634	1192.107	360/29.473
1192.107	1221.611	360/29.504
1221.611	1251.152	360/29.541
1251.152	1280.731	360/29.579

Table 2.1 (Cont'd.)

<u>JD₁</u>	<u>JD₂</u>	<u>m</u>
2441280.731	2441310.336	360/29.605
1310.336	1339.948	360/29.612
1339.948	1369.543	360/29.595
1369.543	1399.109	360/29.566
1399.109	1428.637	360/29.528
1428.637	1458.128	360/29.491
1458.128	1487.593	360/29.465
1487.593	1517.048	360/29.455
1517.048	1546.510	360/29.462
1546.510	1575.996	360/29.486
1575.996	1605.524	360/29.528

The remaining angle required to evaluate the photometric function at a point is the luminance longitude (α). This is the angle in the phase plane at the target between the normal of the projection of the target vector $\vec{T}$ into the phase plane and a vector from the target to the camera ($\vec{E}$). This angle (α) is illustrated in Figure 2.4. Let $\vec{B}$ represent the vector perpendicular to the phase plane at the target. This vector may therefore be defined by:

$$\vec{B} = \vec{E} \times \vec{S}.$$

A vector perpendicular to both the target vector $\vec{T}$ and $\vec{B}$ lies in the phase plane. Denoting this vector as $\vec{A}$, it may be computed as:

$$\vec{A} = \vec{T} \times \vec{B}.$$

Vector $\vec{A}$ is normal to the desired projection vector, hence, a vector perpendicular to $\vec{A}$ and to $\vec{B}$ is the desired vector $\vec{P}$:

$$\vec{P} = \vec{B} \times \vec{A}.$$

The luminance longitude is the angle between $\vec{P}$ and $\vec{E}$; hence,

$$\cos \alpha = \frac{\vec{P} \cdot \vec{E}}{|\vec{P}| |\vec{E}|}$$

The emittance angle e is defined as the angle at the target, between the target vector $\vec{T}$, and the vector from the target to camera, $\vec{E}$. This angle is always in the range ($0 \leq e \leq \pi/2$) and is computed by:

$$\cos e = \frac{\vec{T} \cdot \vec{E}}{|\vec{E}| |\vec{T}|}$$

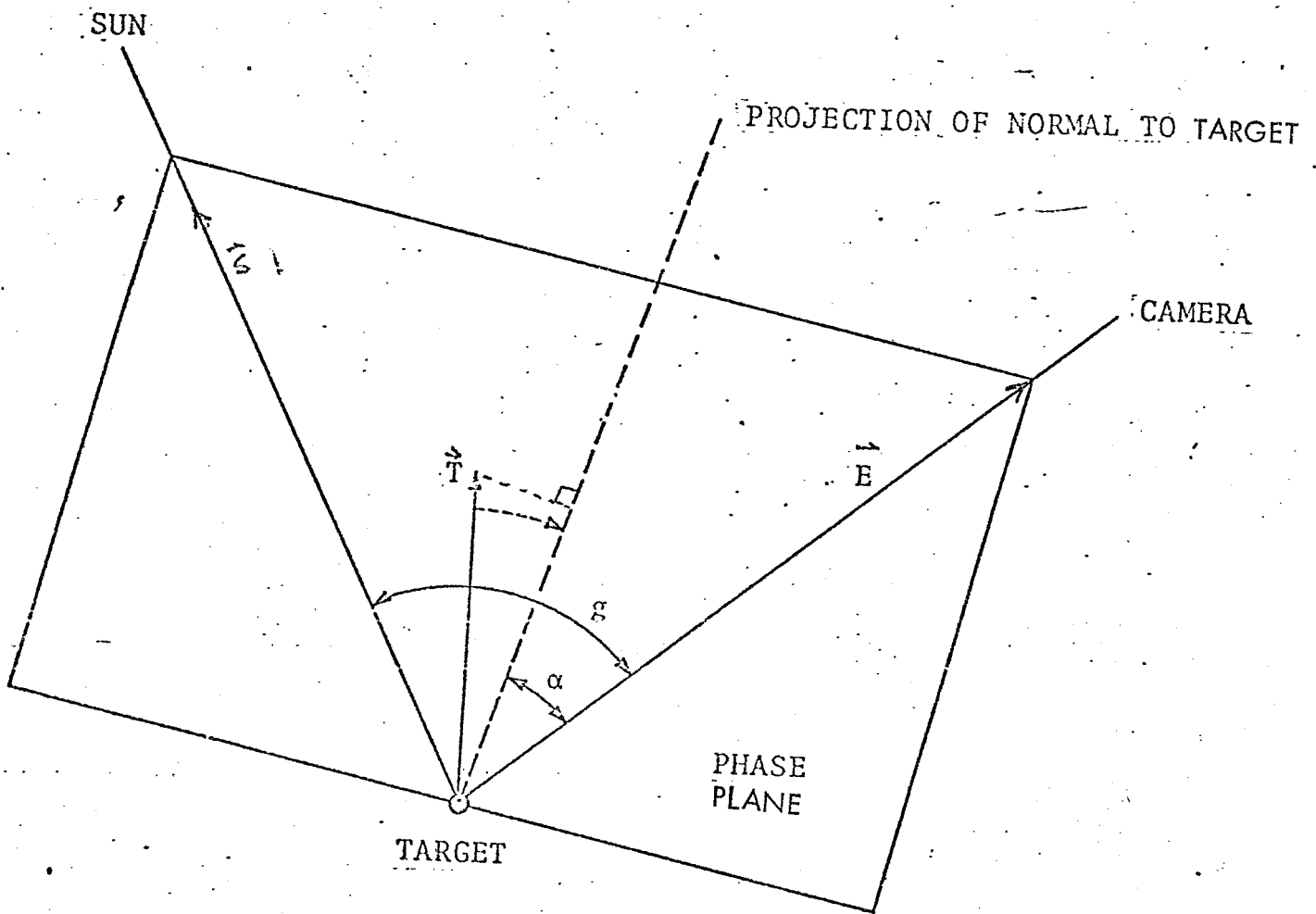


FIGURE 2.4 .

Geometry of Luminance Longitude

The angle of incidence (i) is the angle at the target between the sun vector, $\vec{S}$, and the target vector, $\vec{T}$. This angle is also in the range $(0 \leq i \leq \pi/2)$ and is computed by:

$$\cos i = \frac{\vec{S} \cdot \vec{T}}{|\vec{T}|}$$

The emittance angle and the incidence angle are used only to determine the sign of the luminance longitude (α) in accordance with the following algorithm:

- (i) $\alpha < 0$, if $g > \pi/2$
- (ii) $\alpha < 0$, if $\cos i < \cos e \cos g$
- (iii) otherwise $\alpha > 0$.

The photometric function is best described as a family of curves, one curve for each phase angle. This family of curves is illustrated in Figure 2.5. Based on these curves, values in 5° increments of α and g over the range 0 to π have been tabulated. From this table a two dimensional interpolation scheme is used to compute the photometric constant for any given ρ, ψ combination.

2.2.1.4 Computation of Camera System Irradiance (Lens Transmission Factor)

The percent of light transmitted through the lens system of the camera is determined from an optical bench procedure. A grid of light of known intensity is directed through the lens and the intensity measured upon exit. The ratio between the source intensity and the measured intensity is the percent of transmission. These lights are set up so that a square or rectangular grid is formed in the image plane. This permits one to determine the percent transmission for a given area of the imagery.

MANCE requires a table of values representing percent transmissions be input. To give NASA maximum flexibility in this the utilization of program, the nominal image space coordinates for each point at which the intensity has been measured are also input. Using these values and the image coordinates of the desired

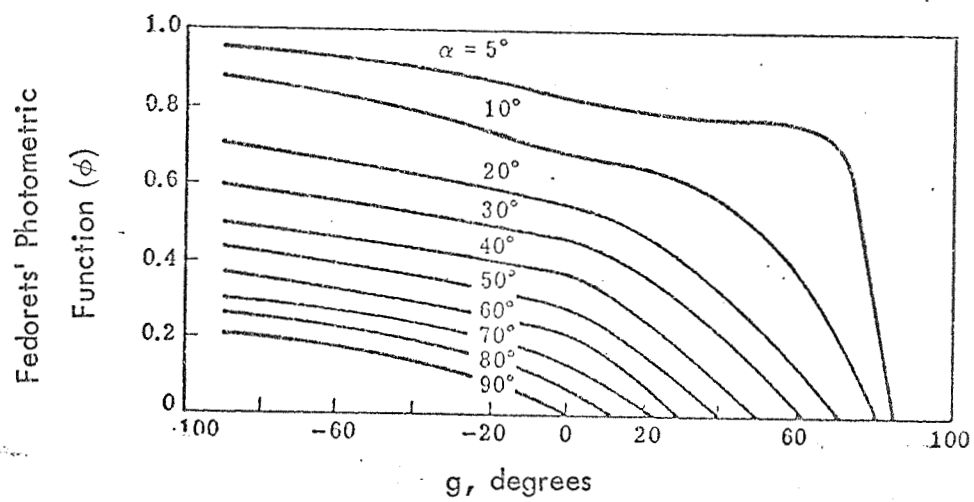


FIGURE 2.5

Fedorets' Photometric Curves

lunar surface point the percent of light transmitted at the image point is computed using a weighted vectorial averaging scheme.

The weight is the reciprocal of the distance from the image point to the point where the present transmission was measured.

2.2.1.5 Summary of Exposure Computation for Single Target

The basic formula (Equation 1) for computing exposure can be written functionally as the following series of parametric equations:

$$E = B V \delta t / 4T^2$$

$$B = f_1(\rho, \psi)$$

$$V = f_2(x, y)$$

$$\delta t = f_3(\hat{\phi}, \hat{\omega}, V^c, P, h^c, \Delta r)$$

$$\rho = f_4(\phi_T, \lambda_T)$$

$$\psi = f_5(\phi_s, \lambda_s, \phi_T, \lambda_T, \phi^c, \lambda^c)$$

in which,

$$B = \text{brightness}$$

$$\delta t = \text{duration of exposure}$$

$$\rho = \text{albedo of lunar surface point } (\phi_T, \lambda_T)$$

$$\psi = \text{photometric constant}$$

$$x, y = \text{image coordinates of target}$$

$$V = \text{percent transmission at coordinates } \hat{x}, \hat{y}$$

$$\hat{\phi}, \hat{\omega} = \text{true pitch and roll of the camera}$$

V^c	=	speed of the vehicle
P	=	percent forward motion compensation
h^c	=	height of satellite
Δr	=	maximum allowable image smear
φ_T, λ_T	=	selenographic latitude and longitude of the target
φ_s, λ_s	=	selenographic latitude and longitude of the subsolar point
φ^c, λ^c	=	selenographic latitude and longitude of the camera
T	=	calibrated f-number

The values of any parameters for which no derivation is given are either input variables or computed from input. For example $\varphi^c, \lambda^c, h^c, V^c$ are computed from the input orbital parameters.

2.2.2 Selection of Optimum Exposure for an Entire Frame

The illuminance varies from point to point over the area of each frame. The duration of the exposure (shutter speed) is determined by image motion considerations. It remains to select a lens aperture (T-value) which will result in maximum detail in the imagery. The procedure followed by this program is a two-step selection process.

For each T-value, three exposure values are selected from all of the values computed in the grid of sample ground points. These are the maximum, minimum, and mean of all exposure values computed for that T-value. These exposure values are tabulated by T-value.

Finally, an attempt is made to select one T-value from each of the three sets of T-values as follows. The exposures derived in Step 1 for each T-value are tested against the optimum film exposure range. When an exposure value is found which is within this range, that value and the T-value associated with it are tabulated, and the search ends. If no acceptable value is found, zeroes are entered in the table. The search is performed upon each set of the three sets of exposures.

It is recognized that for a given application, an alternative interpretation of "optimum exposure" may be desired. For example, one could say that optimum exposure is that T-value which results in the greatest number of image points

in the frame having exposure values which are within the optimum exposure range. The possibilities are limitless. However, until further studies are completed, the present definition seems to be a reasonable choice.

3.0 PROGRAM USER PROCEDURES

3.1 System Control Setup Description

MANCE is written entirely in FORTRAN IV for the UNIVAC 1108 operating under EXEC-2. The control cards are only those required for standard Fortran programs.

File Assignments

MANCE uses one internal scratch file (logical file #8). On option the program will require an input tape or logical unit 1. This file is generated by the Gravity Recovery Module and contains the times of all exposures if a global mission is to be simulated. If the program is operating in a short arc multi-mission mode this input file is not required.

If it is desired to plot the ground coverage of the photography then a magnetic tape must be assigned to logical unit 2. This tape is input to the NASA-MSD plotting program. If the plot option is not exercised for a given run then this file need not be assigned.

3.2 Input Data Description

The input for MANCE can be set up for either a global or short arc multi-mission case. Section 3.2.1 outlines the arrangement of the cards for the data deck. Section 3.2.2. describes the contents of each card in detail.

3.2.1 Data Deck Setup

CASE 1. Global Simulation

All the following cards are required.

<u>Card Type</u>	<u>Description</u>
1	Hollerith ID
2	Option card (1)
3	Cartesian state vector
4	Time of epoch
5	Reference constants
9	Exposure parameters
10	Number of T-values (maximum = 20)
11	Actual T-values (7 per card)
12	Number of entries in irradiance matrix (maximum = 625)
13	Irradiance matrix entry

Card 13 is repeated once for each element in the incidence matrix.

CASE 2. Short Arc Multi-Mission Simulation

<u>Card Type</u>	<u>Description</u>
1	Hollerith ID
2	Option card (0)
5	Reference constants
6	Keplerian state vector
7	Photo card 1
8	Photo card 2

Cards 6, 7, and 8 are repeated for all strips.

<u>Card Type</u>	<u>Description</u>
9	Exposure parameters
10	Number of T-values (maximum = 20)
11	Actual T-values (7 per card)
12	Number of entries in irradiance matrix (maximum = 625)
13	Irradiance matrix entry

Card 13 is repeated once for each entry in the incidence matrix.

3.2.2 Input Card Description

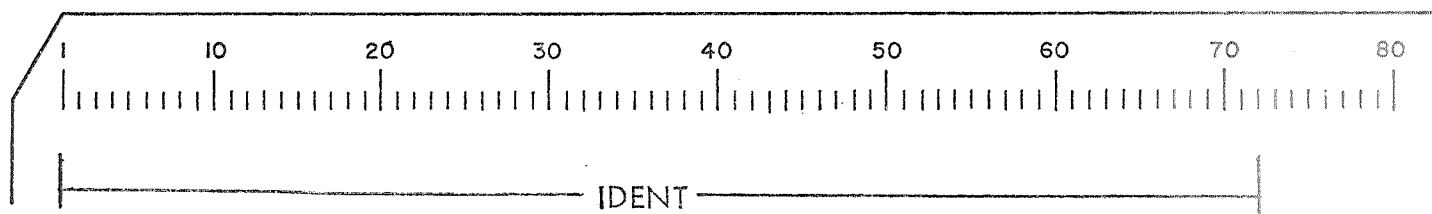
One card is described on each of the following sheets.

3.3 Output Description

The output of MANCE is in the form of hard copy printout. DBA has included sufficient Hollerith information to describe the output. Some of the output is optional and is under the control of the print option flag (OPRINT - reference card type number 9). The optional output is the table of exposure, brightness, albedo, photometric constants, etc. on a point by point basis. All other tables are output for each photograph.

Card Type 1

HOLLERITH IDENTIFICATION



FORMAT (12A6)

Word No.	Program Variable	Units	Format	Column	Math Symbol	Description
1-12	IDENT	-	12A6	1-72	-	72 character Hollerith identification

Card Type 2

OPTION CARD

1	10	20	30	40	50	60	70	80

I T Y P E								

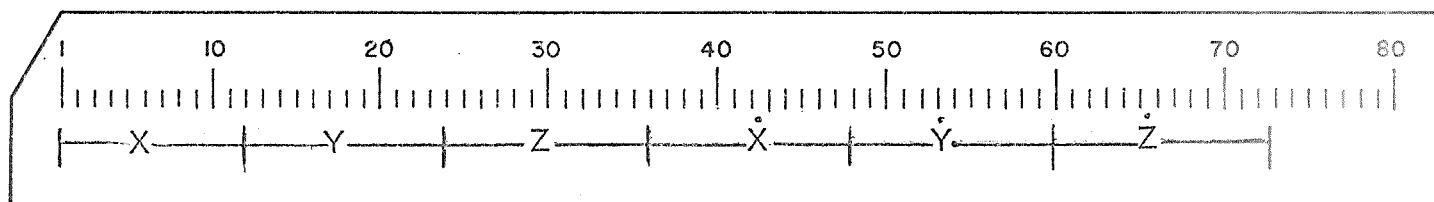
FORMAT (I5)

Word No.	Program Variable	Units	Format	Column	Math Symbol	Description
1	ITYPE	—	I5	1-5	—	Integer flag specifying the type of simulation = 0 - multi-mission 1 - global simulation

Card Type 3

CARTESIAN STATE VECTOR

(IF ITYPE = 1)



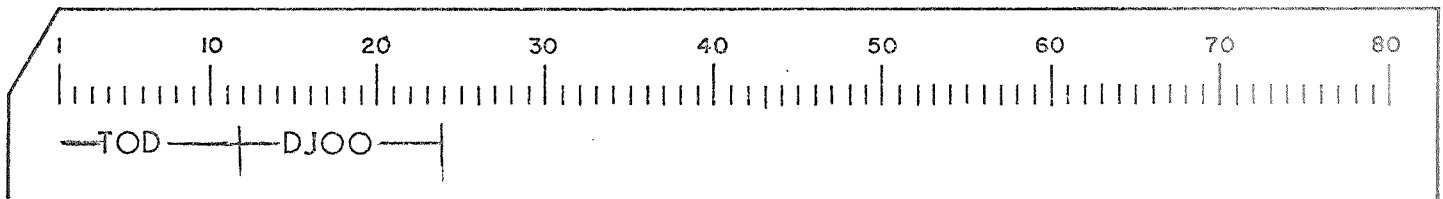
FORMAT (6 D12.6)

Word No.	Program Variable	Units	Format	Column	Math Symbol	Description
1	STATED(1)	M	D12.6	1-12	X	X component of state vector
2	STATED(2)	M	D12.6	13-24	Y	Y component of state vector
3	STATED(3)	M	D12.6	25-36	Z	Z component of state vector
4	STATED(4)	M	D12.6	37-48	X	X component of state vector
5	STATED(5)	M	D12.6	49-60	Y	Y component of state vector
6	STATED(6)	M	D12.6	61-72	Z	Z component of state vector

Card Type 4

TIME OF EPOCH

(IF ITYPE = 1)

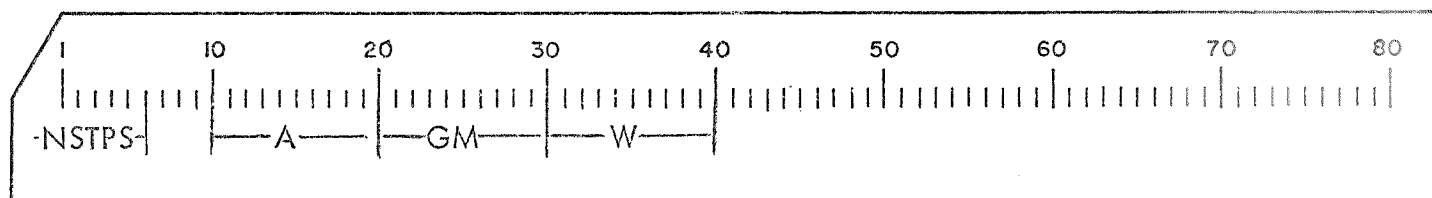


FORMAT (2 D12.6)

Word No.	Program Variable	Units	Format	Column	Math Symbol	Description
1	TOD	SEC	D12.6	1-12	—	Time of epoch
2	DJOO	—	D12.6	13-24	—	Julian Date for the state vector

Card Type 5

REFERENCE CONSTANT



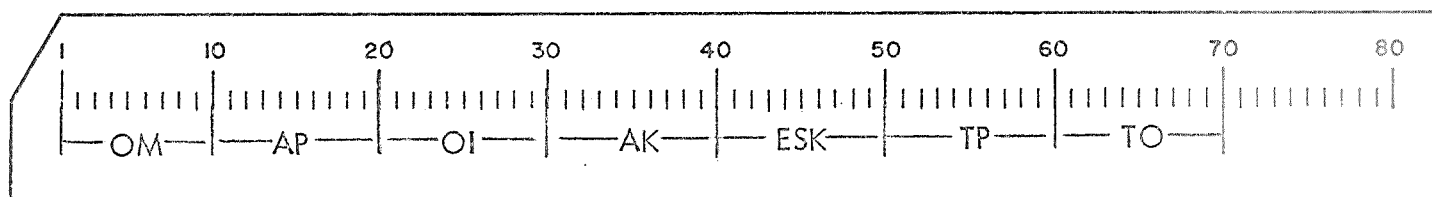
FORMAT (I5, 5X, 3 E10.5)

Word No.	Program Variable	Units	Format	Column	Math Symbol	Description
1	NSTPS	—	I5	1-5	—	Number of strips
2	A	M	E10.5	11-20	α	Semi-major axis of reference ellipsoid
3	GM	rad/s	E10.5	21-30	μ	Gravitational description of origin
4	W	rad/s	E10.5	31-40	ω	Rotational rate of reference ellipsoid

Card Type 6

KELPERIAN STATE VECTOR

(IF ITYPE = 0)



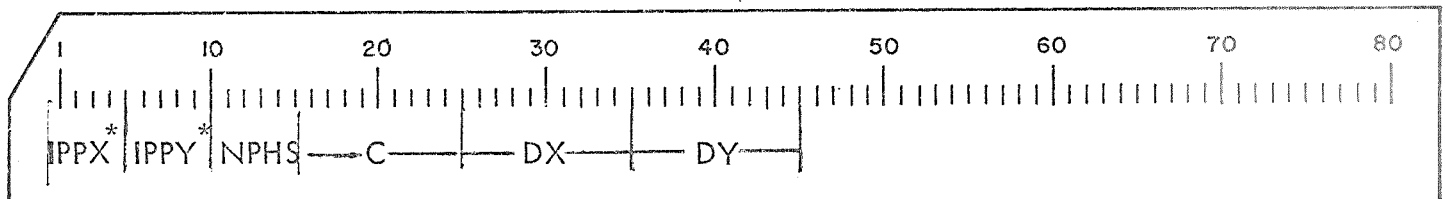
FORMAT (7 E10.5)

Word No.	Program Variable	Units	Format	Column	Math Symbol	Description
1	OM	Deg Deg	E10.5	1-10	Ω	Right ascension of ascending node of orbital plane
2	AP	Deg Deg	E10.5	11-20	ω'	Argument of pericyntion from equator measured in orbital plane
3	OI	Deg Deg	E10.5	21-30	i	Orbital inclination
4	AK	M	E10.5	31-40	a'	Semi-major axis of orbit
5	ESK		E10.5	41-50	e	Eccentricity of orbit
6	TP	Sec	E10.5	51-60	τ	Time of pericyntion
7	TO	Sec	E10.5	61-70	t_0	Time of state vector

Card Type 7

PHOTO CARD 1

(IF ITYPE = 0)



FORMAT (3I5, 3E10.5)

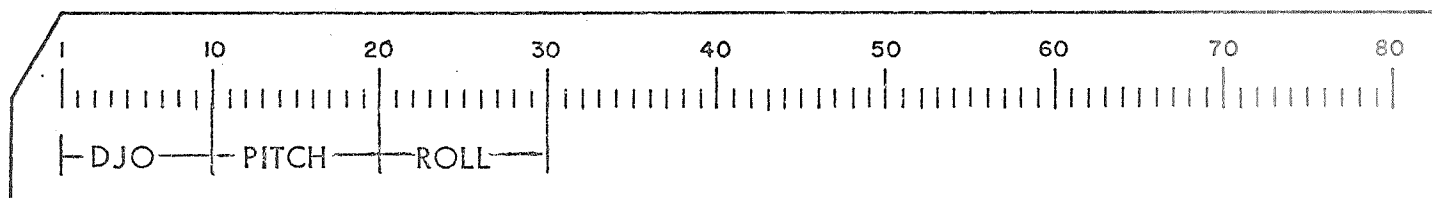
*The product of IPPX and IPPY must be less than 81.

Word No.	Program Variable	Units	Format	Column	Math Symbol	Description
1	IPPX	—	I5	1-5	—	Number of passpoints in X direction of plate (must be odd)
2	IPPY	—	I5	6-10	—	Number of passpoints in Y direction of plate (must be odd)
3	NPHS	—	I5	11-15	—	Number of photos in this strip
4	C	mm	E10.5	16-25	c	Focal length of camera
5	DX	mm	E10.5	26-35	δ_x	Film format in X
6	DY	mm	E10.5	36-45	δ_y	Film format in Y

Card Type 8

PHOTO CARD +2

(IF ITYPE = 0)

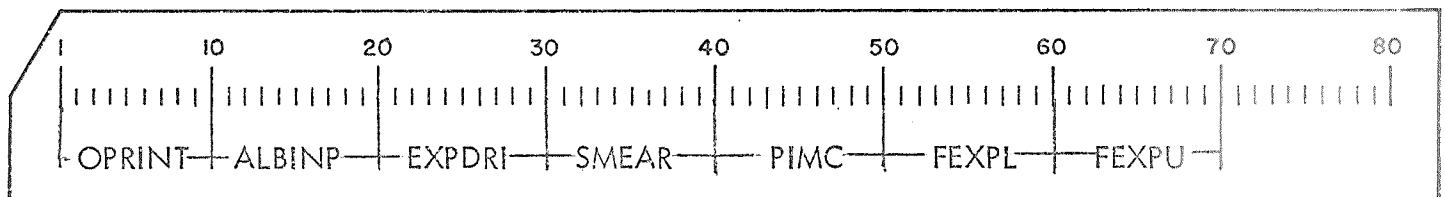


FORMAT (3 E10.5)

Word No.	Program Variable	Units	Format	Column	Math Symbol	Description
1	DJO		E10.5	1-10	—	Julian date at epoch
2	PITCH	Deg	E10.5	11-20	ϕ	Local pitch angle of camera
3	ROLL	Deg	E10.5	21-30	ω^*	Local roll angle of camera

Card Type 9

EXPOSURE PARAMETERS

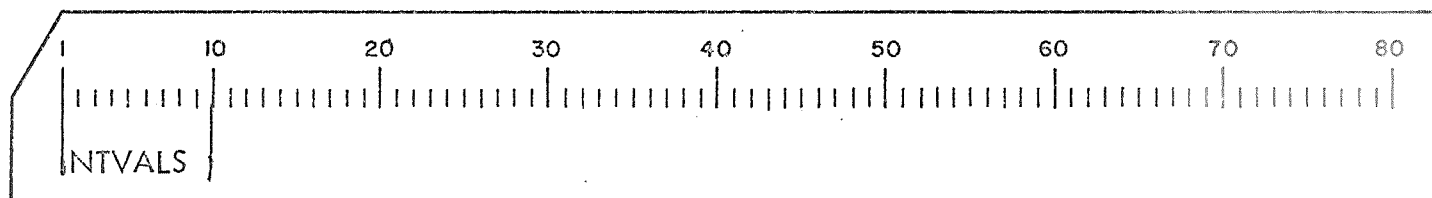


FORMAT (I10, 6F10.0)

Word No.	Program Variable	Units	Format	Column	Math Symbol	Description
1	OPRINT	—	I10	10	—	Print Flag = 1 optional print is desired = 2 optional print is not desired
2	ALBINP	—	F10.0	11-20	—	Default albedo value (this will be used if latitude is not between + and - 20°)
3	EXPDRI	Sec	F10.0	21-30	—	Exposure duration - (If this value is zero the program computes it)
4	SMEAR	mm	F10.0	31-40	—	Maximum allowable image smear
5	PIMC	—	F10.0	41-50	—	Percent image motion compensation
6	FEXPL	—	F10.0	51-60	—	Film exposure lower limit
7	FEXPU	—	F10.0	61-70	—	Film exposure upper limit

Card Type 10

NO. OF T-VALUES



FORMAT (I 10)

Word No.	Program Variable	Units	Format	Column	Math Symbol	Description
1	NTVALS	—	I 10	1-10	—	Number of T-values to be used (maximum of 20)

Card Type 11

T-VALUES

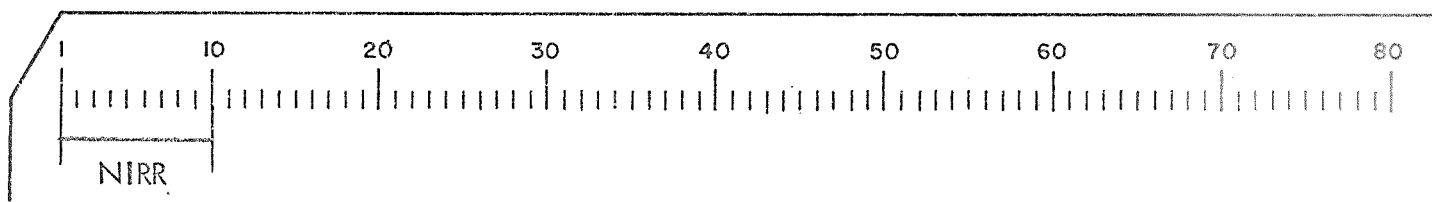
1	10	20	30	40	50	60	70	80
TVALUE(1)	TVALUE(2)	TVALUE(3)	TVALUE(4)	TVALUE(5)	TVALUE(6)	TVALUE(7)		

FORMAT (7 F10.0)

Word No.	Program Variable	Units	Format	Column	Math Symbol	Description
1	TVALUE (I) I = 1, NTVALS	—	7F10.0	1-70	T	The T-values (7 per card)

Card Type 12

NO. OF IRRADIANCE ENTRIES



FORMAT (I 10)

Word No.	Program Variable	Units	Format	Column	Math Symbol	Description
1	NIRR	—	I 10	10	—	Number of entries in irradiance matrix (maximum = 625)

Card Type 13

IRRADIANCE MATRIX

1	10	20	30	40	50	60	70	80
IRR(1,K)	IRR(2,K)	IRR(3,K)						

FORMAT (3 F10.0)

Word No.	Program Variable	Units	Format	Column	Math Symbol	Description
1	IRR(1,k)	mm	F10.0	1-10	—	X-coordinate for the K-th element in the irradiance matrix
2	IRR(2,k)	mm	F10.0	11-20	—	Y-coordinate of the K-th element in irradiance matrix
3	IRR(3,k)	mm	F10.0	21-30	—	Irradiance value for the K-th element

4.0 PROGRAM DESCRIPTIONS

The mission analysis program is composed of a small main program, two major subprograms (MISSIM and APETUR) and an assortment of special purpose and utility subroutines.

Figure 4.1 is a tree content diagram for the programs contained within this module.

4.1 Main Control Program - MANCE

The main program, MANCE, is only a director used to call the MISSIM and the APETUR routines. Figure 4.2 is the flow chart for this program. The comments included in the decks describe the functions performed by the program.

4.2 Ground Coverage Program

The MISSIM subprogram generates the data for the exposure calculations and the ground coverage plots. The mathematics for this subprogram are discussed in Section 2.1 of this volume. Figure 4.3 is a flow chart outlining the data flow for this program; as with all the routines, the comments in the program deck describe the computational techniques.

4.3 Exposure Computation Program

The APETUR subprogram computes the optimum exposure for each photograph simulated by the MISSIM subprogram. The computations performed in this subprogram are described in Section 2.2. The order of computations is presented in Figure 4.4. Comments interspersed in the source decks describe the various computations.

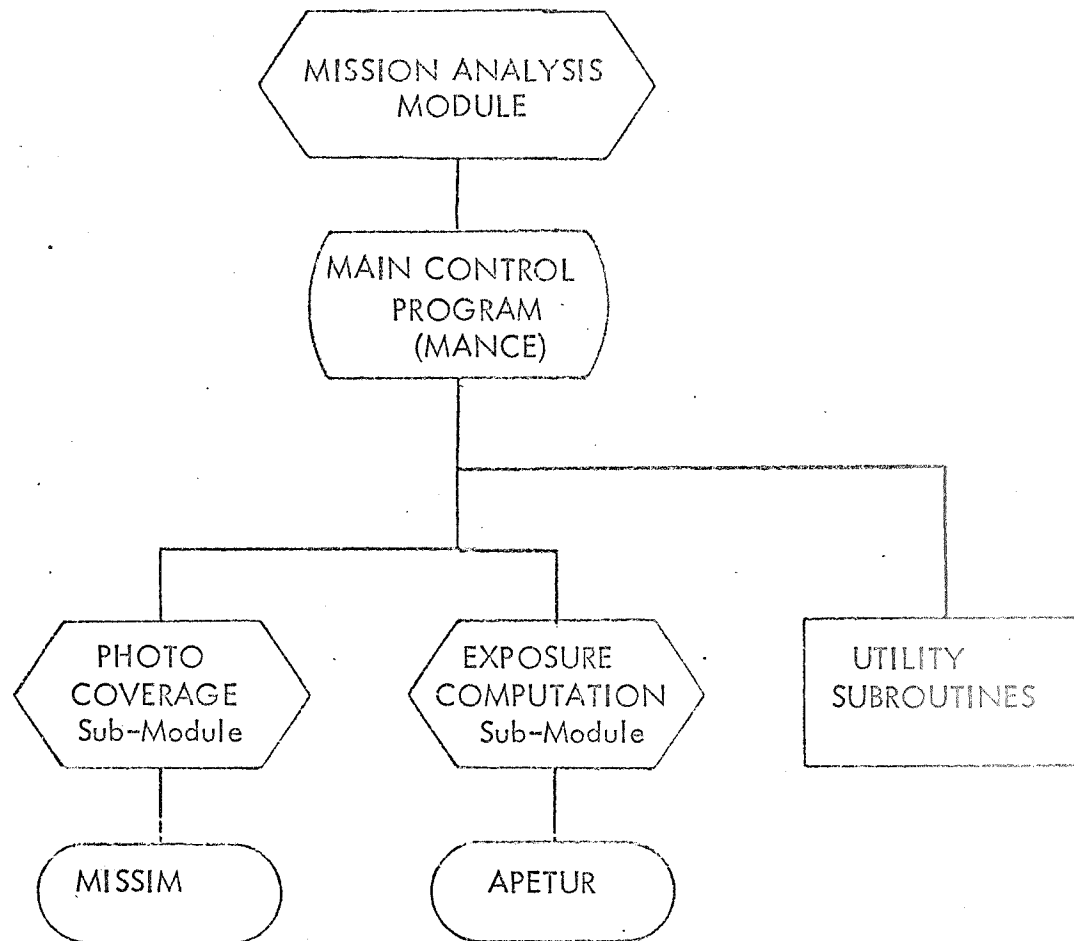


FIGURE 4.1

Tree Content Diagram Mission Analysis Module

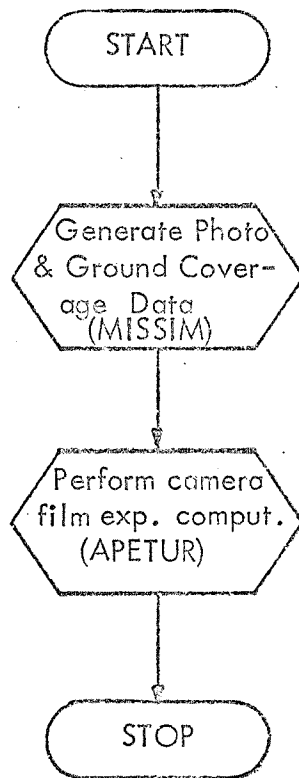


FIGURE 4.2
Flow Diagram for Main Control Program
Mission Analysis Module

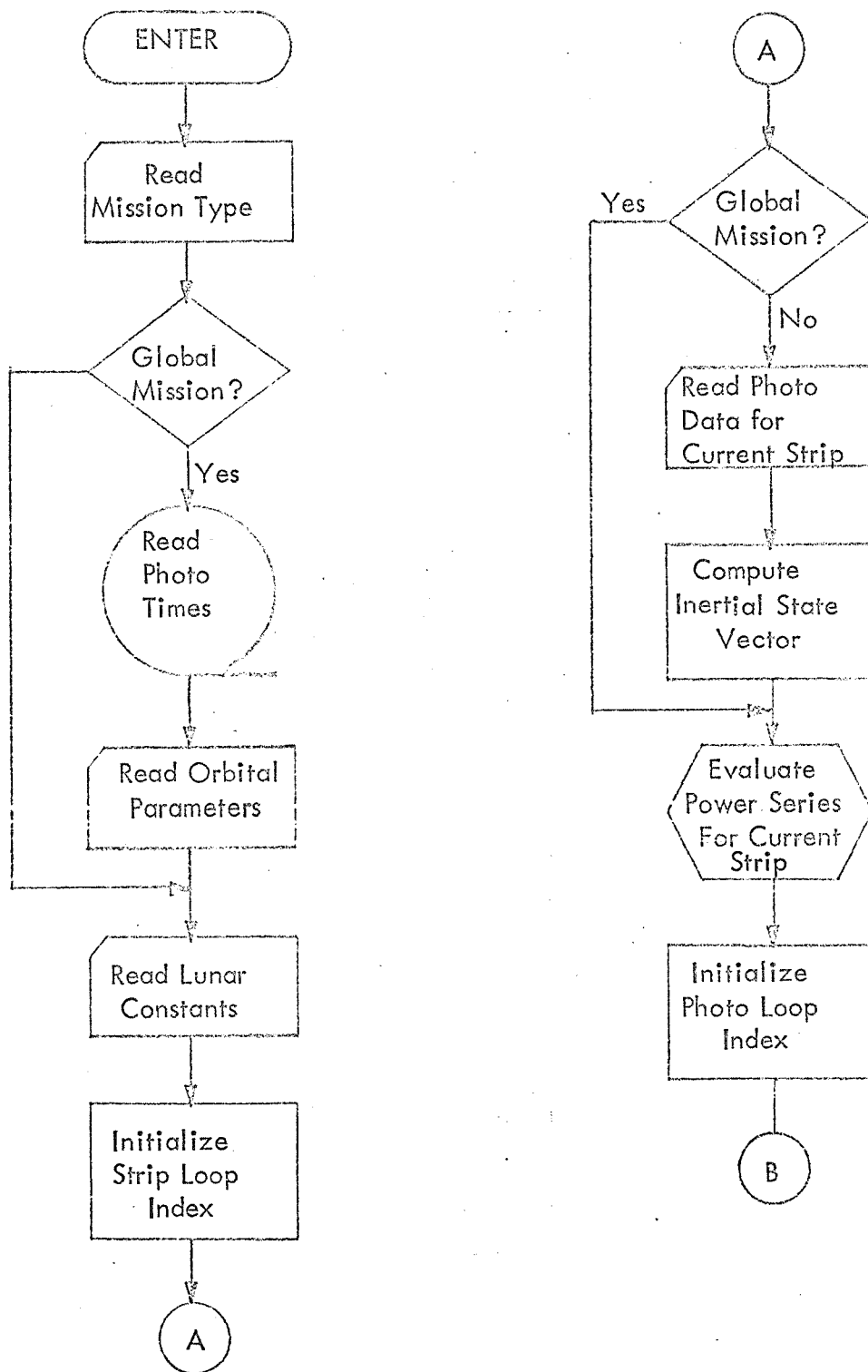


FIGURE 4.3 (Part 1 of 2)
Flow Diagram of Simulation Program - MISSIM
Mission Analysis Module

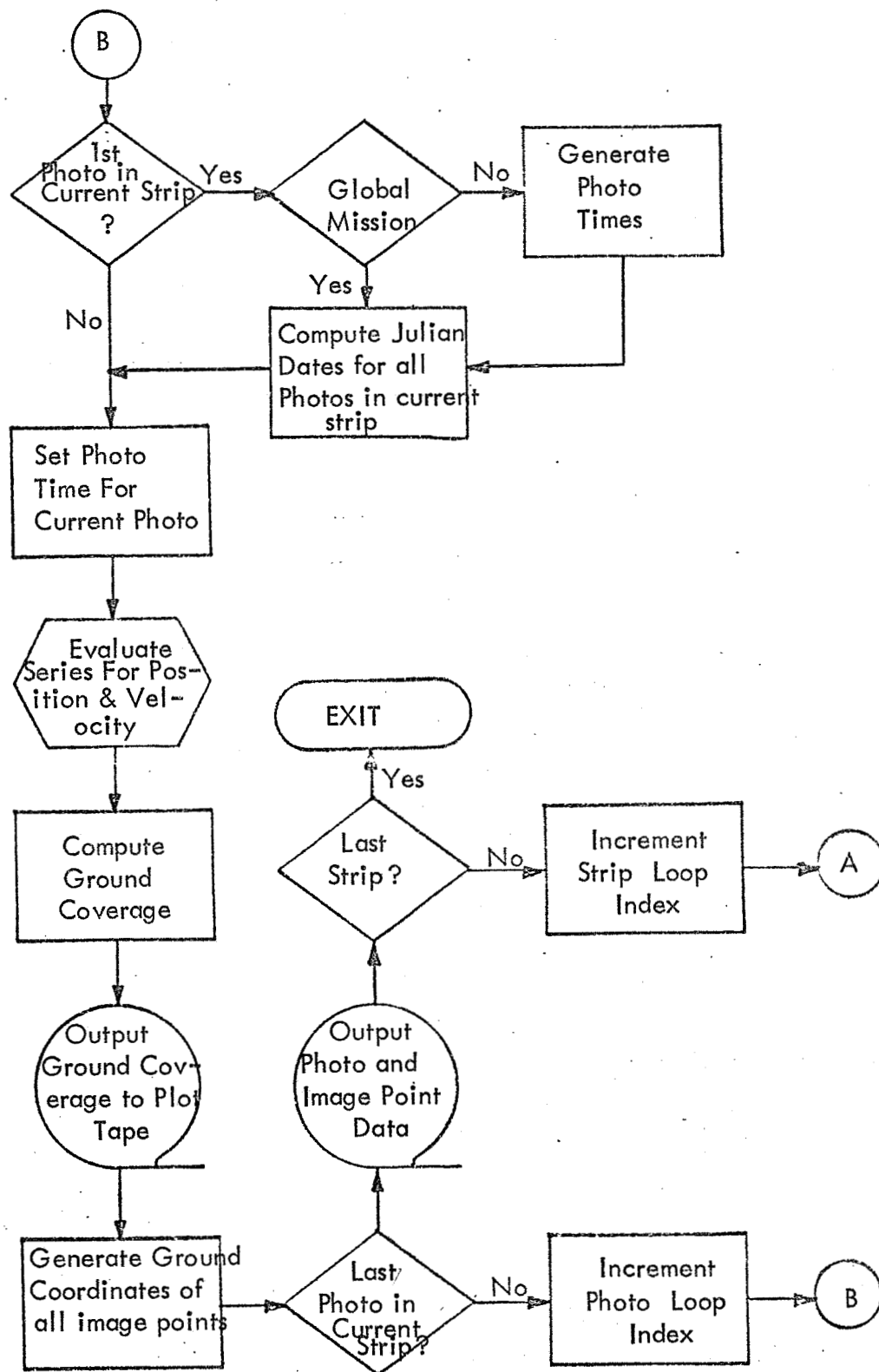


FIGURE 4.3 (part 2 of 2)

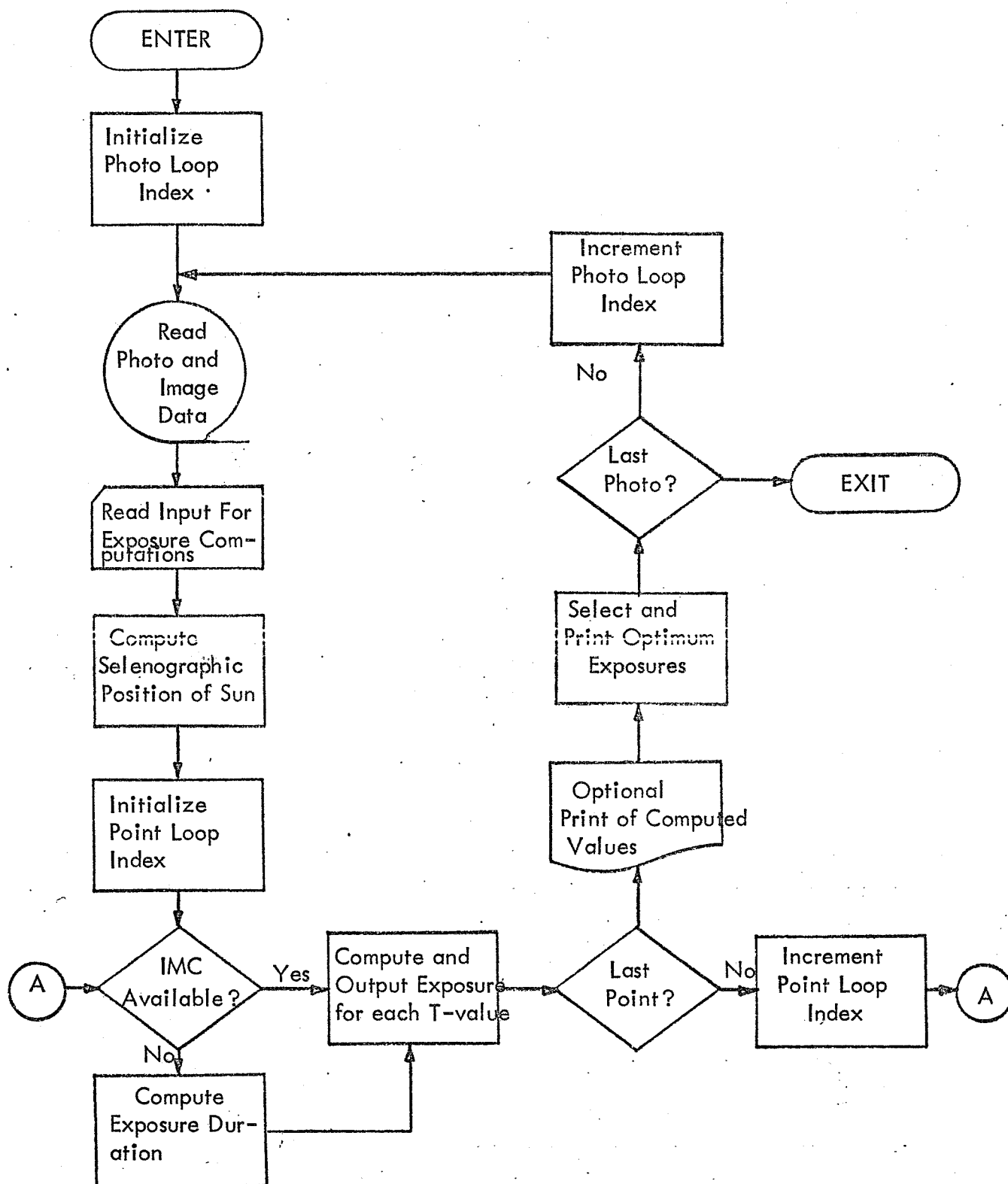


FIGURE 4.4
Flow Diagram of Exposure Computations - APETUR
Mission Analysis Module

4.4 Utility Subroutines

4.4.1 Subroutine ALBTAB

4.4.1.1 Entry ALBLOK

ALBLOK interpolates a table of internally stored albedo values.

4.4.1.2 Entry ALBTAB

ALBTAB is an initialization entry.

4.4.1.3 Entry IRRLOK

IRRLOK interpolates a table of internally stored irradiance values.

4.4.1.4 Entry PMLOOK

PMLOOK interpolates a table of internally stored photometric constants.

4.4.2 Subroutine ALTTAB

ALTTAB creates internally stored atmospheric drag tables from input data.

4.4.3 Subroutine CLEAR

CLEAR is used to set the elements of a double precision array to zero.

4.4.4 Subroutine CODERA

4.4.4.1 Entry CODERA

CODERA is the main integrator control program for evaluating the series expression of the orbit.

4.4.4.2 Entry CODERB

CODERB evaluates the series expression of the orbit at a specific time.

4.4.5 Subroutine COMMER

COMMER prints a header page for the hard copy output.

4.4.6 Subroutine CROSSP

CROSSP computes the cross product of two vectors.

4.4.7 Subroutine GEOTR2

GEOTR2 performs the conversion of selenocentric coordinates to selenographic coordinates.

4.4.8 Function INBCD

INBCD is set equal to the Fieldata (BCD) representation of the integer argument.

4.4.9 Subroutine LENGTH

4.4.9.1 Entry LENGTH

LENGTH constructs an array of subscript values for use by the integrator.

4.4.9.2 Entry SETADD

SETADD constructs actual core addresses for the integrator.

4.4.10 Subroutine MATMPY

MATMPY is a multi-purpose matrix multiplication package.

4.4.11 Subroutine MOVE

MOVE is a utility routine for moving data arrays within core.

4.4.12 Subroutine MOVERS

MOVERS is a multi-entry package of matrix utility routines.

4.4.12.1 Entry INCOIN

INCOIN increments each element of an integer array by a constant, and stores the result in another integer array.

4.4.12.2 Entry INMOVE

INMOVE moves elements from one integer array to another.

4.4.12.3 Entry INSTEP

INSTEP constructs a spacing array.

4.4.12.4 Entry KLEAR

KLEAR sets the elements of an integer array to zero.

4.4.12.5 Entry MOVERS

MOVERS is a dummy entry included to establish dimensions for the subroutines.

4.4.12.6 Entry REMOVE

REMOVE moves elements from one real array to another.

4.4.12.7 Entry SPRADD

SPRADD adds the elements of one array to certain elements of another array.

4.4.12.8 Entry SPREAD

SPREAD changes the spacing between array elements.

4.4.12.9 Entry SUBSET

SUBSET moves an array from or to a larger array.

4.4.13 Subroutine OMAT

OMAT computes the USR orientation matrix from orientation angles as follows:

$$T = R_3(\chi) R_2(-\varphi) R_1(\omega).$$

4.4.14 Subroutine OMLV

OMLV computes the local vertical orientation matrix from local vertical angles as follows:

$$\bar{T} = R_1(\bar{\omega}) R_2(-\bar{\varphi}) R_3(\pi/2 - \bar{\alpha}).$$

4.4.15 Subroutine PAGE

4.4.15.1 Entry CYCLE

CYCLE is called to advance hard copy to start for successive page and print headings.

4.4.15.2 Entry PAGE

PAGE is an initialization entry.

4.4.16 Subroutine PLHXYZ

PLHXYZ transforms selenographic coordinates to selenocentric coordinates.

4.4.17 Subroutine POTTAB

POTTAB constructs a potential model table for the integrator.

4.4.18 Subroutine PROJ

PROJ computes the USR coordinates of the intersection of a ray with the reference ellipsoid.

4.4.19 Subroutine RADDMS

RADDMS converts an angle given in radians to degrees, minutes, and seconds.

4.4.20 Subroutine RINUSR

RINUSR computes the transformation matrix for converting inertial to USR as follows:

$$R = R_3(\text{GHA})$$

4.4.21 Subroutine RLVUSR

RLVUSR computes the transformation matrix for converting local vertical to USR as follows:

$$R = R_1(\pi/2 - \varphi) R_3(\pi/2 + \lambda)$$

4.4.22 Functions SDOD, SDOP, SDOQ, SDOS, SDOT

These functions compute the dot product of variables in a series form.

The following notation is used:

$$a^2 = \sum_{i=0}^K a_i a_{k-i}$$

b^2 , c^2 , and d^2 are similarly defined.

4.4.22.1 Entry SDOD

$$\text{SDOD} = a^2$$

4.4.22.2 Entry SDOP

Initialization entry (Must be called once).

4.4.22.3 Entry SDOQ

$$\text{SDOQ} = a^2 + b^2 + c^2 - d^2$$

4.4.22.4 Entry SDOS

$$\text{SDOS} = a^2$$

4.4.22.5 Entry SDOT

$$\text{SDOT} = a^2 + b^2 + c^2$$

4.4.23 Subroutine SEDATA

SEDATA is a set-up routine to prepare data for the integrator.

4.4.24 Subroutine SERMAM

SERMAM performs a series oriented matrix multiplication.

4.4.25 Subroutine SEXDRG

SEXDRG computes the series expansion with respect to drag.

4.4.26 Subroutine SEXPOT

SEXPOT computes the series expansion with respect to potential coefficients.

4.4.27 Multiple Entry Function Package SNAP

This package contains a group of functional entries for computing series form cross products. The following notation is used:

t is a real variable

$$ab = \sum_{i=0}^K a_i b_{k-i}$$

$$cd = \sum_{i=0}^K c_i d_{k-i}$$

4.4.27.1 Function SMAM

$$SMAM = t - (ab - cd)$$

4.4.27.2 Function SMIP

$$SMIP = t - (ab + cd)$$

4.4.27.3 Function SMOT

$$SMOT = t - ab$$

4.4.27.4 Function SNAM

$$SNAM = - (ab - cd)$$

4.4.27.5 Function SNAP

SNAP is an initialization entry which must be called once.

4.4.27.6 Function SNIP

$$SNIP = - (ab + cd)$$

4.4.27.7 Function SNOT

$$SNOT = -ab$$

4.4.27.8 Function SPAM

$$\text{SPAM} = t + (ab - cd)$$

4.4.27.9 Function SPIP

$$\text{SPIP} = t + (ab + cd)$$

4.4.27.10 Function SPOT

$$\text{SPOT} = t + ab$$

4.4.27.11 Function SZAM

$$\text{SZAM} = (ab - cd)$$

4.4.27.12 Function SZIP

$$\text{SZIP} = (ab + cd)$$

4.4.27.13 Function SZOT

$$\text{SZOT} = ab$$

4.4.28 Subroutine SOLINT

SOLINT forms solutions at a given time for a previously generated power series. SOLINT is used by the integrator.

4.4.29 Subroutine SOSTAR

SOSTAR re-initializes power series for analytical continuation of the integration process.

4.4.30 Subroutine SOTIME

SOTIME solves for integration time step, Δt .

4.4.31 Function TABLOK

TABLOK is a two-way table look-up function.

4.4.32 Subroutine TIMDAY

TIMDAY extracts current time and date from the computer system.

4.4.33 Functions UDACC, UDJRK, UDPOS, UDVEL

These functions are used by the integrator to update variables.

4.4.33.1 Entry UDACC

UDACC solves power series for acceleration.

4.4.33.2 Entry UDJRK

UDJRK solves power series for jerk.

4.4.33.3 Entry UDPOS

UDPOS solves power series for position.

4.4.34 Entry UDVEL

UDVEL solves power series for velocity.