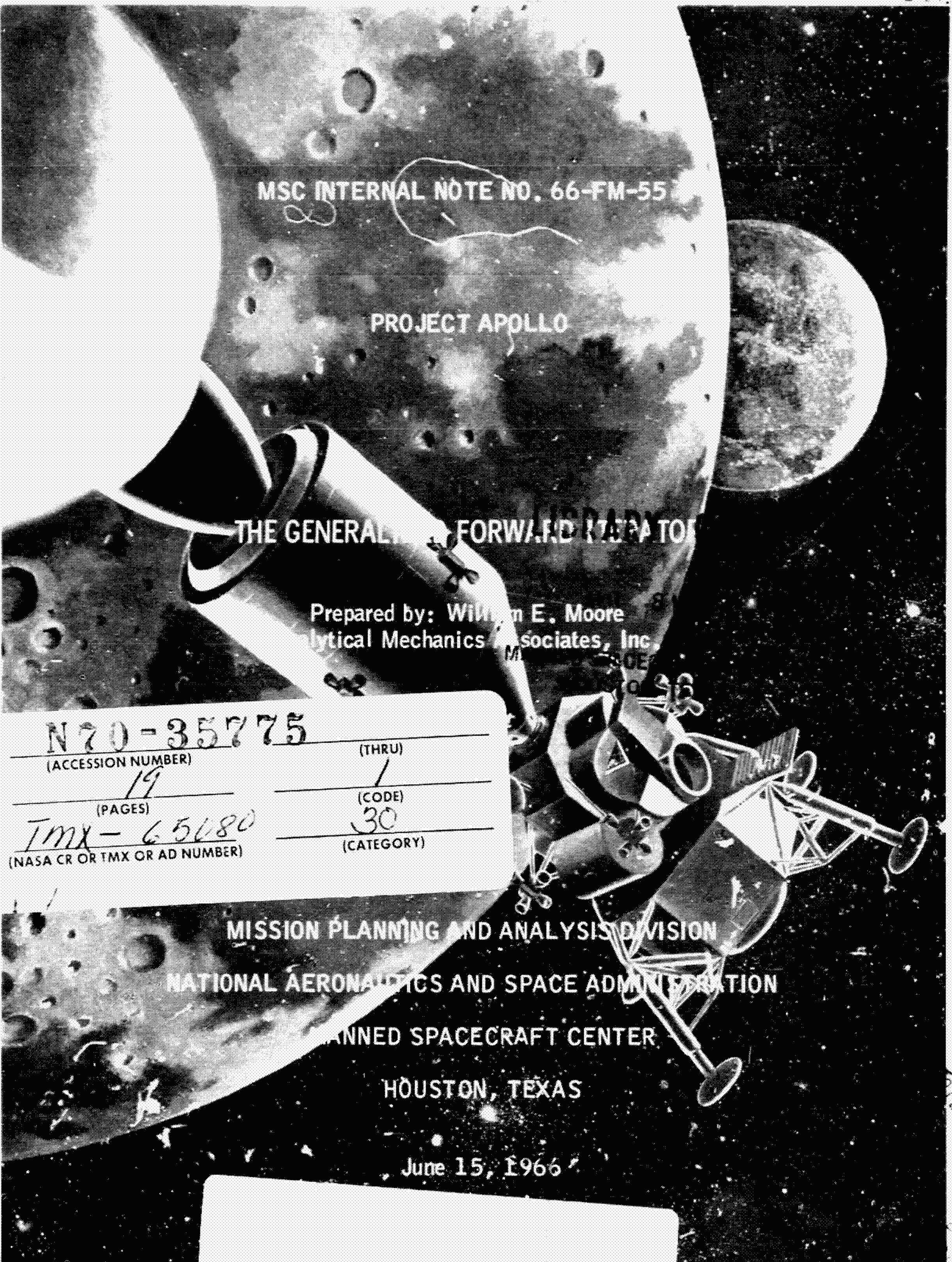

MSC INTERNAL NOTE NO. 66-FM-55

PROJECT APOLLO

THE GENERALIZED FORWARD ITERATOR

Prepared by: William E. Moore
Analytical Mechanics Associates, Inc.

MISSION PLANNING AND ANALYSIS DIVISION
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
MANNED SPACECRAFT CENTER
HOUSTON, TEXAS

June 15, 1966

MSC INTERNAL NOTE NO. 66-FM-55

PROJECT APOLLC

THE GENERALIZED FORWARD ITERATOR

Prepared by: William E. Moore
Analytical Mechanics Associates, Inc.

June 15, 1966

MISSION PLANNING AND ANALYSIS DIVISION

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

MANNED SPACECRAFT CENTER

HOUSTCN, TEXAS

Approved: MPFrank
M. P. Frank, III, Chief
Mission Analysis Branch

Approved: JPMayer
John P. Mayer, Chief
Mission Planning and Analysis Division

THE GENERALIZED FORWARD ITERATOR

by: William E. Moore

SUMMARY

A computer program has been written which constructs optimized Apollo missions. It is highly versatile; it will produce a scan of several related missions or an accurate trajectory of reference quality. Speed and accuracy are purely dependent on the mathematical model chosen for each particular case. This model could consist of an array of two-body approximations, or of a precise numerical integration of the equations of motion going by the Encke method when applicable. The program is extremely flexible with regard to the mission that can be constructed, and the extent to which it can be optimized. This note indicates how the input describes the mission and the method of computation to the computer. It also presents the various mathematical trajectory models that are available.

INTRODUCTION

This Apollo mission design program has been designed to furnish the analyst with a reliable tool for the design of a great variety of missions. It is fast enough to be capable of producing a "scan" when no great accuracy is required, as in preliminary mission design, and accurate enough to produce a trajectory of reference quality when required. It is general and flexible enough to permit the analyst to assess the effects of many different mission configurations and the introduction or omission of numerous constraints.

The major constituents of the program are (1) input and initialization, (2) first guess generation, (3) trajectory, mission parameter and constraint computation, (4) iterative parameter correction for the satisfaction of constraints and eventual mission optimization procedure, and (5) program output. These constituent parts are described in detail in the following sections.

PROGRAM INPUT AND INITIALIZATION

The operation of the program is completely controlled by a set of input data and switches. Each input number has associated with it on a card its location in the full input array. This location appears on the card and the input processor then stores this number in its proper location. Thus, it is necessary only to load those input numbers which are needed and which differ from previously loaded value. Physical quantities may be read in a number of different units. The unit is indicated by a code letter on the input card, which enables the processor to apply the proper scale factor for conversion to internal units. The input processor further has the capability of using a previously loaded number, of incrementing it by a specified amount, or of using a specified multiple or fraction of the present value.

A first guess for the independent variables obtained previously may be retained, be incremented or else be recomputed for every case. In addition, the converged answer of the previous case may be used as a first guess.

The input array further contains a set of switches which substantially control the flow of the program.

One set of these describes the situation at the beginning of the trajectory, and thus indicates which maneuvers are still to be performed. For example, the trajectory might start at the launch site, or at a later time, such as some position and velocity in earth parking orbit, or along the translunar leg, in lunar parking orbit, or along the transearth leg.

Another set of switches describes the independent variables of the search; i.e., the variables to be determined, and the constraints to be satisfied. Table II gives a list of all the inputs to the program.

During initialization all these switches are used to set up a path through the trajectory computer, so that only the trajectory segments needed in the problem at hand are computed, and all necessary constraints evaluated. It further guarantees that all necessary partial derivatives are computed. Simultaneously, in addition, all these switches are checked for consistancy. If an obvious inconsistency is detected, the program proceeds to the next case. Because of the great generality of the program, a complete consistency check is not feasible. Therefore, even if the input passes all the consistency tests, this "consistent" trajectory is not guaranteed to represent a reasonable problem, but as much checking of the switches as possible is done.

A third set of switches controls mode of computation of each of the various sub-arcs in the trajectory. The alternatives here are computation by numerical integration or by conic approxiations for the coast phases and idealized arcs for the powered portions.

The choice here will depend on the accuracy required as well as on the computer time available.

FIRST GUESS GENERATION

In order to insure the most efficient behavior of the iteration scheme, in some cases, first guesses for one or more of the independent variables may be computed. No effort has been made to fill up the program with all the known direct methods for getting these values; rather only those which make the trajectory computer program using or the iterator operation more efficient are included.

One can choose to compute first guesses for time of launch and time in earth parking orbit if starting at the launch site. Falling out, also, is the velocity increment at translunar injection. These three values are based on the desired translunar flight time, or a nominal time if a free-return trajectory is desired. From this value and the base time, the position of the moon at some point after arrival is computed as a target. Now the time of launch is chosen so that the plane of the earth parking orbit contains this target point. Then the time in that orbit is adjusted so that the translunar trajectory passes through the target point. These times are close enough to the correct ones to be quickly adjusted by the parameter search procedure.

In the case where the parking orbit is already fixed, the procedure is slightly modified in that the translunar trajectory--still assumed to be in the same plane as the parking orbit--commonly will be chosen as close as possible to the target point, but will not, in general, contain it. In this case, too, the search procedure quickly adjusts the first guess to their correct value.

Another case involves a point after translunar injection, but not on the nominal translunar trajectory. In this situation, the most efficient procedure turns out to be to apply the search technique in a "backward" manner first. This consists of searching for a time of pericynthion and a pericynthion velocity vector, given a fixed pericythion position, which will lie on a trajectory containing the original given point. At the latter point, the computed velocity vector provides first guesses for the maneuver to be executed to return to nominal condition.

Finally, for each of the major burn phases, provision is made to compute first guesses for the guidance parameters. To use this provision, the analyst must furnish a vector approximating the state at the end of burning.

TRAJECTORY COMPUTER

The variables to be determined are, of course, initial and control variables; these are called independent variables. The constraints to be satirfied by the mission are represented by parameters called dependent variables. They are quantities which can be derived from states at significant points along the trajectory. Thus, by the trajectory computer we mean the program which transforms the independent variables into state variables, follows the values of the state variables along the mission path, and transforms the state variables into the dependent variables. The transformations are usally simple computations based on geometrical or physical relations and may on occasion involve a simple iteration. The trajectory proper - the propagation of the state variables along the mission path - is more complicated. Computing the state variables accurately is a time-consuming process. Thus, the user has a choice between an accurate but slow, and a fast but approximate method of computing each arc. These arcs are divided into two classes, according as power (that is, thrusting) is off or on. The arcs without power are called coasting arcs and include earth parking orbit, translunar coast, linear parking orbit, free-return coast, and transearth coast. Each of these may be integrated, if accuracy is desired, by the Encke method with the "universal incremental anomaly" as independent variable. Where speed is more important, as happens when there are many cases to be run(e.g. in a scan of initial or final parameters) the geometrical-dynamical equations of two-body motion are used.

The Encke method is, by, now, reasonably familiar. It consists of separating the accelerations into two-parts--one of which gives a differential equation solvable in closed form (the "two-body" solution), and the other of which is small, (the "perturbations") so that, even though they must be integrated numerically, the integrations can proceed in relatively large steps without the introduction of appreciable errors. The programming for the two-body solution is available anyway, in connection with the approximate trajectory methods; hence, no extra space is used to handle it. Moreover, each integrated arc of the trajectory has a certain termination condition, which is the product of an iteration in step size. First guesses for stopping conditions follow easily from the two-body formulas.

Both the two-body and perturbation equations are written in terms of the anomaly β and derivatives with respect to β. Doing this allows the integration step size to adapt itself to changes in distance from the central attracting body. Furthermore, the computation of the time, t, from β is decidedly easier than computing β from t. This is because Kepler's equation is transcendental in β, but linear in t. For a description of the equations in terms of β, see the Appendix.

The equations for the two-body motion are formulated in terms of two different anomalies. The "universal incremental anomaly", β, and the "eccentric incremental anomaly", Θ. It would be possible to write all the equations in terms of β but in this way computing time has to be sacrificed to elegance; hence, the "universal" approach has been abandoned where necessary. The current method is popularly referred to as the "Herrick-Beta" method.

The formulas are divided into two classes. The first class contains procedures for determining the value of the universal anomaly through which the position and velocity and time must be propagated. Here β is generally described in terms of the eccentric anomaly, Θ. The second class contains the procedures for propagating the position and velocity vectors, and the time. In all that follows X_o, $\dot{X}_o$, and t are, respectively, the position vector, the velocity vector, and the time at a known point on the trajectory. Let μ be the mass parameter of the attracting body.

We start with the second class of formulas, that is, the propagation formulas. Let

$$(X_o \cdot X_o)^{\frac{1}{2}} = r_o \qquad (1)$$

$$(\dot{X}_o \cdot \dot{X}_o) = v_o^2 \qquad (2)$$

$$X_o \cdot \dot{X}_o = d_o \qquad (3)$$

$$\frac{1}{a} = \frac{2}{r_o} - \frac{v_o^2}{\mu} \qquad (4)$$

$$\alpha = -\frac{1}{a} \beta^2 \qquad (5)$$

$$G_i = i \sum_{k=o}^{\infty} \frac{\alpha^k}{(2k+i)!} \; (i = 0, 1, 2, 3) \qquad (6)$$

$$r = r_o G_o + \frac{d_o G_1}{\sqrt{\mu}} + G_2 \qquad (7)$$

$$d = \sqrt{\mu}\left(1 - r_o\left(\frac{1}{a}\right)\right) G_1 + d_o G_o \qquad (8)$$

$$t = t_o + \frac{1}{\sqrt{\mu}}\left(r_o G_1 + \frac{d_o G_2}{\sqrt{\mu}} + G_3\right) \qquad (9)$$

$$f = 1 - \frac{1}{r_o} G_2 \qquad (10)$$

$$g = \frac{r_o}{\sqrt{\mu}} G_1 + \frac{d_o}{\mu} G_2 \qquad (11)$$

$$\dot{f} = -\frac{\sqrt{\mu}}{r r_o} G_1 \qquad (12)$$

$$g = 1 - \frac{1}{r} G_2 \qquad (13)$$

$$x = f x_o + g \dot{x}_o \qquad (14)$$

$$\dot{x} = \dot{f} x_o + \dot{g} \dot{x}_o \qquad (15)$$

Formulas (9), (14), and (15) perform the propagations, but formulas (7) and (8) are also convenient in some applications.

The equation for β depend on some specified configuration of the state such as periapsis, specified distance, and specified flight-path angle. For periapsis, compute r_o, v_o^2, d_o, and $\frac{1}{g}$ as in equation (1) - (4) above.

Then set $s = \frac{d_o}{\sqrt{\mu}}\sqrt{\left|\frac{1}{a}\right|}$ (16)

$$c = 1 - r_o \frac{1}{a} \qquad (17)$$

$$e = \sqrt{c^2 + \frac{d_o^2}{\mu} \frac{1}{a}} \qquad (18)$$

If $\frac{1}{a} > 0$, set $E = \tan^{-1}\left(\frac{s}{c}\right)$ taking all four quadrants into considerations (19)

If $\frac{1}{a} < 0$, set

$$E = \log \frac{s + c}{e} \qquad (19')$$

$$\beta = -\frac{1}{\sqrt{|1/a|}} \quad E \qquad (20)$$

This arrangement breaks down if $\frac{1}{a} = 0$, but this case is impossible to create in the computer.

To determine the β for a given distance, it is first required that the current distance r_o is less than, or equal to, the desired distance s. It is assumed that, in the case of elliptic orbits, no more than one revolution is traversed from input state to output state. With these two assumptions, for each direction, incoming and outgoing, there can be no more than one place on the orbit at the distance r. Thus a switch, W, is provided.

$$W = \begin{cases} +1 & \text{if forward} \\ -1 & \text{if backward} \end{cases} \qquad (20)$$

Again formulas (1) - (4) provide d_o and $\frac{1}{a}$, and (17) and (18) are used to get c and e.

Now set $C_o = \frac{c}{e}$ (21)

$$C = \frac{1}{e}\left(1 - r\left(\frac{1}{a}\right)\right) \qquad (22)$$

If $\frac{1}{a} > 0$, and $C < -1$, the distance r is impossible, and calculation is suspended; otherwise, compute

$$S_o = \sqrt{|1 - C_o^2|} \qquad (23)$$

$$S = \sqrt{|1 - C^2|} \qquad (24)$$

Now if $\frac{1}{a} > 0$

$$E_o = \tan^{-1}\left(\frac{S_o}{C_o}\right) \qquad (25)$$

$$\text{and } E = \tan^{-1}\left(\frac{S}{C}\right) \qquad (26)$$

with the results allocated between quadrants I and II.

(Four-quadrant allocation does this automatically because (23) and (24) give non-negative results.)

If $\frac{1}{a} < 0$

$$E_o = \log (C_o + S_o) \qquad (25')$$

$$\text{and } E = \log (C + S) \qquad (26)$$

$$\text{Now} \quad |\theta| = |(\operatorname{sgn} d_o) E_o - W E| \qquad (27)$$

represents the absolute value of the total eccentric anomaly. Thus

$$= \frac{W\ |\theta|}{\sqrt{|\frac{1}{a}|}} \qquad (28)$$

To determine the β for a given flight-path angle we require that the input state must be at periapsis. Formulas (1), (2), and (4) are again used to get r_o and $\frac{1}{a}$. Now the eccentricity, e, is given by

$$e = 1 - \frac{r_o}{a} \qquad (29)$$

Compute

$$S = \frac{\sqrt{|e^2 - 1|}}{e} \quad \frac{\sin \gamma}{\cos \gamma} \qquad (30)$$

$$C = \begin{cases} \sqrt{1 - S^2} & \text{if } \frac{1}{a} > 0 \\ \sqrt{1 + S^2} & \text{if } \frac{1}{a} < 0 \end{cases} \qquad (31)$$

$$= \begin{cases} \tan^{-1} \frac{S}{C} & \text{if } \frac{1}{a} > 0 \\ \log \frac{S + C}{e} & \text{if } \frac{1}{a} < 0 \end{cases} \qquad (32)$$

and finally,

$$\beta = \frac{\theta}{\sqrt{|\frac{1}{a}|}} \qquad (33)$$

Note that in the case of an elliptic orbit, formula (31) allows only that occurrence of the flight-path angle which is close to periapsis.

If any arc has to pass from the vicinity of one attracting body to the vicinity of another an iterative procedure is used to make sure that the arc is a continuous piecing together of two-body arcs. At the point where the arcs join, both the positions and velocities must agree. This "patch" point is defined as the intersection of the initial two-body arc with the surface around the moon where the ratio of the distance from the moon to the distance from the earth has a given value.

To start the procedure, the initial state is used, by means of the distance formula previously given, to find two values of the universal anomaly β for which the ratio described above will bracket the given value. If this is impossible, the precedure is terminated because the initial state is not such as to determine an arc intersecting the surface. From these first two values of β, a third value is obtained by linearly interpolating to get the desired ratio. Then these three values are used to start a quadratic iteration process, at each stage of which the outer two of three values of β used are chosen to bracket the value of the ratio, and the inner value is the last value computed. The process is stopped when this last value changes by a negligible amount.

For the burning arcs in the integrated mode, the thrust acceleration is incorporated among the terms to be integrated. It is computed from the guidance logic appropriate to each maneuver. In the approximate mode a geometrical representation of the effects of burning is used. That is, the changes in time, distance, velocity, and flight-path angle are applied, and the effects of rotation in the previous plane of motion, and out of it, are incorporated. The changes and amounts of rotation have to be calculated beforehand by studying the results of the integrations of the burn arcs. Let x_o, $\dot{x}_o$ be the position and velocity vectors at the beginning of the burn arc. Let Δr, Δv, and $\Delta\gamma$ be the changes in distance, velocity, and flight-path angle, respectively. Let Δs, ΔA, be the amounts by which the position vector is rotated in the plane of motion, and by which the velocity vector is rotated out of the plane of motion, respectively.

Let $r^2 = x_o \cdot x_o$

$$v^2 = \dot{x}_o \cdot \dot{x}_o$$

$$d = x_o \cdot \dot{x}_o$$

$$h = |x_o \times \dot{x}_o|$$

Then $x_1 = x_o \cos \Delta s + \dfrac{r^2\dot{x}_o - dx}{h} \sin \Delta x$

$$\dot{x}_1 = \dot{x}_o \cos (\Delta s - \Delta \nu) + \frac{d\dot{x}_o - v^2 x_o}{h} \sin (\Delta s - \Delta \gamma)$$

Let $d_1 = x_1 \cdot \dot{x}_1$

$$r_1 = (x_1 \cdot x_1)^{\frac{1}{2}}$$

Then $x_2 = x_1$

$$\dot{x}_2 = \frac{2d_1x_1}{r_1^2} \sin \frac{\Delta A}{2} + \dot{x}_1 \cos \Delta A - \frac{x_1 \cdot \dot{x}_1}{r_1} \sin \Delta A$$

$$r_2 = (x_2 \cdot x_2)^{\frac{1}{2}} = r_1 \text{ and } v_2 = (\dot{x}_2 \cdot \dot{x}_2)^{\frac{1}{2}}$$

Finally,

$$x_3 = x_2 \left(1 + \frac{\Delta r}{r_2}\right)$$

$$\dot{x}_3 = \dot{x}_2 \left(1 + \frac{\Delta v}{v_2}\right)$$

are the position and velocity vector at the end of the burning arc.

The mass ratio is exp $\left(-\dfrac{|\dot{x}_3 - \dot{x}_o|}{Ig}\right)$ where I is the specific impulse and g is the acceleration due to gravity at the earth's surface.

A narrative description of the actual method of computing the trajectory follows. The independent and dependent variables are referred to as x_i and y_i, respectively. For example, x_{25} is the independent variable time of launch, and y_3 is the dependent variable time of launch. Many quantities required by the trajectory computer may be computed, once for all trajectories, as soon as the input is available. Thus the reader need not assume that every computation shown is performed every time the trajectory computer is passed through. Instead, the trajectory computer is intended to take care of the effects of changing values of any of the x_i.

We start at the launch site. Knowing the latitude and longitude of launch, and the time of launch, x_{25}, we know the position of the spacecraft at launch. An artificial velocity vector is made up out of the launch azimuth and the circular velocity at the height defined by the Δr for ascent. Now an application of the approximate burn formulas furnishes the state at the end of ascent (or at the beginning of earth parking orbit). In the process the deviations in flight-path angle and velocity from circular conditions, x_{22}, and x_{23}, are used. Note that there isn't any provision for integrating the ascent trajectory.

The time and azimuth at launch, y_3 and y_4, are simply transferred over as dependent variables. Two-body formulas are now applied to the state at the beginning of earth parking orbit to get y_5 and y_6, the perigee and apogee, heights, respectively, of the orbit. If the orbit is circular, the initial height is used. Next the state at the beginning of earth parking orbit is propagated to the state at the end of earth parking orbit, by using the time in earth parking orbit, x_{21}. This computation may be either accurate or approximate.

For the approximate translunar injection change in velocity and plane change, x_{20} and x_{19} , are used in the burn formulas to get the state at the end of translunar injection (or the state at the beginning of translunar coast). For the accurate translunar injection, the obsolete MIT guidance is used, pending an exact definition of something better. In either case, the mass after translunar injection, y_7, is available.

The state at the beginning of translunar injection is either propagated through the patching iteration, or integrated to pericynthion, where the height, y_8, and translunar flight time, y_{14}, are available.

The flight-path angle at the beginning of lunar orbit insertion, x_{14}, is used, together with the state at pericynthion, to get the state at the beginning of the lunar orbit insertion maneuver. This is done either by the approximate formula given above or by integrating accurately to a state which has the given flight-path angle. At the start of lunar orbit insertion, the height x_9, is available. The coordinates of the state are then transformed so that they are referenced to the earth-moon plane, and the inclination, latitude, longitude, and azimuth, y_{10}, y_{11}, y_{12}, and y_{13}, are computed in that reference frame.

If free-return constraints are to be computed, the state at the beginning of lunar orbit insertion is propagated back to the earth to get a state of perigee, from which the height and inclination, y_{15}, and y_{16}, can be obtained.

Next the state at the start of lunar parking orbit is used to compute the effects of the lunar orbit insertion maneuver. In the process, either x_{13} or x_{11} and x_{12} are control parameters for this manever. As a result, we get the state at the beginning of lunar parking orbit and the height y_{18} and mass y_{17} at that point. If the parking orbit is to be non-circular, x_9 and x_{10}, the flight-path angle and excess (above circular) velocity, are introduced. Then from the resulting state at the beginning of lunar parking orbit, the heights at pericynthion and apocynthion, y_{19} and y_{20}, are obtained. Now the state is propagated through the time x_8, resulting in a state supposedly directly over the landing site. Thus x_{21} and y_{22}, the latitude and longitude of that point, are determined. The time of staying on the moon is used to propagate this last state to the state of the CSM at the time of departure from the moon's surface. This state is used to derive the angle y_{23} by which the LEM is out of the lunar orbit plane.

From the last state, we propagate to the state at the end of lunar parking orbit, by introducing enough time to exhaust the total time in parking orbit x_7. This state is then operated on by the control parameters for transearth injection and the mass y_{24}.

The state after transearth injection is then propagated back toward the earth, obtaining the state at perigee. The transearth flight time y_{25} is the time to perigee. In approximate calculations, if the orbit is elliptical, but very close to the center of the earth, this time could be negative. To create smoother convergence, an orbital period is added to the time in this case. At perigee, we have the return inclination y_{27}. The next quantities are dependent on the state at reentry. This state is defined to have a certain flight-path angle, which is a function of the energy of the return orbit. From perigee we propagate the state back to this flight-path angle. At reentry, then, the height, y_{28}, and the velocity, y_{29}, are available.

The calculation of the rest of the constraints is always approximate, and is based on the assumption that the motion of the spacecraft during reentry may be approximately represented by a circular orbit. Finding the landing site may be done in either of two ways, at the option of the user. First, he may specify a fixed reentry range. The circular orbit is calculated as to traverse this range. Thus we have a state at landing, from which azimuth at landing y_{29}, total mission time y_{31}, and latitude at landing y_{32} are easy to derive. Alternatively, the user may specify the longitude of the earth landing site, in which case an iteration is performed to choose the time elapsed so that the spacecraft and the landing site have the same longitude. (For partial-derivative computations, this time may be adjusted by one revolution, so as to remove discontinuities.) From the time of landing, we get azimuth at landing, y_{26}, reentry range, y_{30}, total mission time, y_3, and latitude of landing, y_{32}.

If it is desired to pick up an initial state vector, rather than starting at launch, this can easily be achieved by setting a switch to describe to the program how far into the mission the state is. If enough stages of the trajectory have been computed to derive all the dependent variables, then the remainder is by-passed. Thus, in all instances, only the desired parts of the trajectory are computed, so as to save computer time.

PARAMETER CORRECTION SCHEME

Generally speaking, the first guesses applied to the trajectory computer will not yield values of the dependent variables that satisfy the constraints. This program makes use of an iteration scheme to correct the independent variables until the constraints are, indeed, satisfied. The scheme is described in detail elsewhere (see ref. 1).

OUTPUT

The output section provides the value of all the converged input variable and the values of all the dependent variables. In addition, for each of the states appropriate to the converged trajectory, about a hundred parameters are displayed. These include coordinates of the spacecraft and the attracting bodies relative to several reference system, the polar angles corresponding to these coordinates, orbital elements, the orbital parameters, etc.

REFERENCES

1. Campbell, J. H.; Moore, W. E.; and Wolf, H.: A General Method for Selection and Optimization of Trajectories. AIAA/ION Astrodynamics Specialist Conference, Monterey. California, September 16-17, 1965, AIAA Paper No. 65-697

APPENDIX

AN ENCKE METHOD ADAPTED TO MISSION ANALYSIS

1. The Standard Encke Method

In the standard formulation of the trajectory problem, we are given

$$\ddot{x} = \frac{d^2x}{dt^2} = -\mu \frac{x}{|x|^3} + F, \qquad (1)$$

$$x = (t_o) = x_o \quad \dot{x}\,(t_o) = \dot{x}_o$$

where x is the vector from the central body to the spacecraft, μ is the attraction coefficient due to the central body, F is the sum of the other forces acting on the spacecraft, and x_o, $\dot{x}_o$ are the initial position and velocity vectors. According to the standard Encke method, we introduce another differential equation

$$\ddot{\rho} = \frac{-\mu\rho}{|\rho|^3} \qquad (2)$$

The solution $\rho = \rho\,(t)$ of this equation, with $\rho(t_o) = x_o$ and $\dot{\rho}(t_o) = \dot{x}_o$ can be found with extreme accuracy in closed form.

Thus we set

$$x = \rho + \xi \qquad (3)$$

so that, since $\xi = x - \rho$, we have to solve the differential equation

$$\ddot{\xi} = \ddot{x} - \ddot{\rho} = -\mu\left(\frac{-x}{|x|^3} - \frac{\rho}{|\rho|^3}\right) + F \qquad (4)$$

with the initial conditions

$$\xi\,(t_o) = x\,(t_o) - \varrho(t_o) = x_o - x_o = 0$$

and $\dot{\xi}\,(t_o) = \dot{x}\,(t_o) - \dot{\rho}\,(t_o) = \dot{x}_o - \dot{x}_o = 0$.

In the region where (1) is difficult to solve, that is, near $|x| = 0$, equation (4) is much easier to integrate, so that, in general, the same accuracy can be obtained with less computing time.

2. Change of Independent Variable

In the construction of the closed-form solution for (2), a parameter β arises, related to t by the equation

$$\beta = \sqrt{\mu} \int_{t_0}^{t} \frac{dt}{\rho} \qquad (5)$$

In terms of β, Kepler's equation takes the form

$$t = t_0 + \frac{f\ (\beta)}{\sqrt{\mu}} \qquad (6)$$

where f is a transcendental function of β, and is obtained by summing several power series.

If t is taken as the independent variable, eauation (6) has to be solved for β by an iterative method, requiring numerous time consuming evaluation of the function f for each integration step. Using β as the independent variable, however, only requires a single evaluation.

It remains, of course, to see what becomes of equations (2) and (4) if β is is the independent variable. We have from (5) that

$$\frac{dt}{d\beta} = \frac{\rho}{\sqrt{\mu}}$$

at any point along the solution of (2). A prime (') denotes differentiation with respect to β. Thus $\dot{\rho} = \rho' \frac{\sqrt{\mu}}{\rho}$ and $\rho' = \dot{\rho} \frac{\rho}{\sqrt{\mu}}$ at any point along the solution of (2). Thus the initial conditions become $\rho(\beta_0) = x_0$ and $\rho'(\beta_0) = \frac{x_0 \ |x_0|}{\sqrt{\mu}}$ when $\beta_0 = \beta(t_0) = 0$.

Now the solution of (2), ρ and ρ', can be written in clcsed form for any β. As auxiliary quantities in this solution we have $|\rho|$ and

$$D = \frac{\rho \cdot \rho'}{|\rho|}$$

They are computed as functions of β before ρ and ρ' known, that is, with accuracy at least as good as that of ρ and ρ'. Not only are they needed and easy to compute, but also they have the interesting property that

$$\frac{dt}{d\beta} = \frac{\rho}{\sqrt{\mu}} \text{ , as we saw above,} \qquad (7)$$

and $\frac{d^2t}{d\beta^2} = \frac{D}{\sqrt{\mu}}$

Thus equation (2) is solved more economically in terms of β than in terms of t.

Now we turn to equation (4). To treat it, we want to express ξ'' in terms of ξ. From (7) we have that

$$\xi' = \dot{\xi}\,\frac{dt}{d\beta} = \dot{\xi}\,\frac{|\rho|}{\sqrt{\mu}}$$

Differentiating with respect to β,

$$\begin{aligned}\xi'' &= \frac{d\dot{\xi}}{d\beta}\,\frac{|\rho|}{\sqrt{\mu}} + \dot{\xi}\,\frac{d}{d\beta}\,\frac{|\rho|}{\sqrt{\mu}}\\ &= \ddot{\xi}\left(\frac{|\rho|}{\sqrt{\mu}}\right)^2 + \xi'\,\frac{\sqrt{\mu}}{|\rho|}\,\frac{D}{\sqrt{\mu}}\\ &= \frac{\ddot{\xi}|\rho|^2}{\mu} + \xi'\,\frac{D}{|\rho|}\\ &= -|\rho|^2\left(\frac{x}{|x|^3} - \frac{\rho}{|\rho|^3}\right) + \frac{\rho^2}{\mu}\,F + \xi'\,\frac{D}{|\rho|} \qquad (8)\end{aligned}$$

Thus (8) is the equation to be integrated numericall--, instead of (4).

The coefficients $\frac{|\rho|^2}{\mu}$ and $\frac{D}{|\rho|}$ can be calculated with much more accuracy than the factors involving ξ, since they depend only on the two-body solution.

For analysis of error propagation, we write (8) as

$$\xi'' = \frac{-1}{|\rho|}\left[(\rho+\xi)\,\frac{|\rho|^3}{|\rho|^3} - \rho\right] + \frac{|\rho|^2}{\mu}\,F + \xi'\,\frac{D}{|\rho|}\,.$$

The mechanics of the procedure, then, are easy to enumerate. The initial conditions are x_o and $\dot{x}_o$. Let

$$\rho(t_o) = x_o$$

$$\rho'(t_o) = \frac{\dot{x}_o|x|_o}{\sqrt{\mu}}\,.$$

Using these initial conditions, evaluate t, $\frac{|\rho|^2}{\mu}$, $\frac{D}{|\rho|}$, ρ, ρ' for each value of β to be considered.

Let $\xi_0 = \xi_0' = 0$. Using these initial conditions, integrate equation (8) to get $\xi(\beta)$ and $\xi'(\beta)$. Note that the first two terms on the right-hand side of equation (7) are functions of x and possibly x'. These are obtained by

$$x(\beta) = \rho(\beta) + \xi(\beta)$$

$$x'(\beta) = \rho'(\beta) + \xi'(\beta).$$

If, at any point $\dot{x}$ is required, it can be found from

$$\dot{x}(t(\beta)) = x'(\beta) \frac{\sqrt{\mu}}{\sqrt{\rho(\beta)}}$$

Depending on the rectification control logic, there will be places where the solution to equation (2) must be started over.

At this point, β, ξ, and ξ' are reset to zero, while the value t, x, $\dot{x}$ become the new t_0, x_0, $\dot{x}_0$. In particular, then,

$$x_0' = \frac{x\,|\rho|}{|x|}.$$