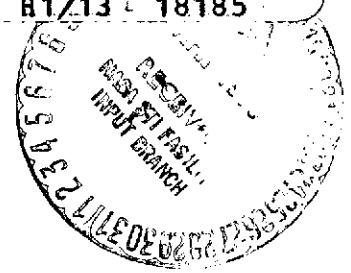
NASA TECHNICAL NOTE

NASA TN D-7882

NASA TN D-7882



DEVELOPMENT OF A METHOD FOR OPTIMAL MANEUVER ANALYSIS OF COMPLEX SPACE MISSIONS

Stewart F. McAdoo, Jr., Donald J. Jezewski, and G. S. Dawkins

Lyndon B. Johnson Space Center
Houston, Texas 77058

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • APRIL 1975

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA TN D-7882		

4. Title and Subtitle	5. Report Date
DEVELOPMENT OF A METHOD FOR OPTIMAL MANEUVER ANALYSIS OF COMPLEX SPACE MISSIONS	April 1975
	6. Performing Organization Code JSC-09186
7. Author(s) Stewart F. McAdoo, Jr., and Donald J. Jezewski, JSC, and G. S. Dawkins, University of Houston	8. Performing Organization Report No. JSC S-427
9. Performing Organization Name and Address Lyndon B. Johnson Space Center Houston, Texas 77058	10. Work Unit No. 986-16-20-00-72
	11. Contract or Grant No.
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D.C. 20546	13. Type of Report and Period Covered Technical Note
	14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

A system has been designed that allows mission planners to find optimal multiple-burn space trajectories easily. Two previously developed methods with different gravity assumptions perform the optimization function. The power of these programs is extended by a method of costate estimation. A penalty function method of constraining coast arc times to be positive is included. The capability of the method is demonstrated by finding the optimal control for three different space missions. These include a Shuttle abort-once-around mission and two- and three-burn geosynchronous satellite-placement missions.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
*OPBURN *Optimum *Multiple Burn *Shuttle Abort-Once-Around *Geosynchronous Transfer *Costate Estimation *Finite Thrust	STAR Subject Category: 13 (Astrodynamics)

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	29	$3.75

*For sale by the National Technical Information Service, Springfield, Virginia 22151

CONTENTS

TABLES

FIGURES

DEVELOPMENT OF A METHOD FOR OPTIMAL MANEUVER ANALYSIS OF COMPLEX SPACE MISSIONS

By Stewart F. McAdoo, Jr., Donald J. Jezewski, and G. S. Dawkins*
Lyndon B. Johnson Space Center

SUMMARY

Several approaches have been applied to the problem of analyzing missions for spacecraft that have thrust levels low enough that the impulsive thrusting approximation is not valid. The principles of optimal control were applied to the problem of multiple-burn spacecraft trajectories, which resulted in a method (called herein the inverse-square program) by which optimum multiple-burn trajectories could be found by determining the times at which the rocket engine should be switched on and off and by determining initial values for the differential equations describing the behavior of the costate vector. An optimum impulsive solution to the problem was computed and thrust arcs formed around the impulses. This program successfully found solutions to problems, but it was of limited usefulness because numerical integration of state, costate, and perturbation differential equations across thrust arcs required a significant amount of computer time for many problems and because the scheme for determining starting iterates was not suitable for problems in which the spacecraft thrust level was low.

To reduce the time required for computing multiburn trajectories, a second program (called herein the ω program) was based on the assumption that, on thrust arcs, the gravitational acceleration vector varies linearly with the radius vector. This assumption resulted in a closed-form solution to the state and costate differential equations across the thrust arcs that greatly reduced the trajectory computation time.

The inverse-square and ω programs are similarly structured. A starting iterate for the inverse-square program obtained from the ω program would not have limitations imposed by the thrust level; and, because it is obtained from a finite thrust program, it would be more accurate than the method of forming thrust arcs around impulses. The most obvious result of the increased accuracy would be the reduction in computer time required. Another advantage would be the ability of the combined programs to find solutions to missions performed by spacecraft having low thrust levels, for which solutions were previously not possible. On the other hand, the ω program is in itself sufficiently accurate for a large number of applications, and the inverse-square program could be used to verify that accuracy when necessary.

It is expected that the mission planner can estimate with reasonable accuracy the array of engine switch times. On an optimal trajectory, the spacecraft thrust vector is alined with the first three components of the costate vector (called the primer vector); so, with this in mind, the engineer can make some estimate of the direction of the primer vector. The last three components of the costate vector, which are called the primer vector derivative, cannot be estimated by association with physical properties of the mission under consideration. In this study, a scheme is presented for estimating the primer vector derivative. This scheme requires the planner to make some estimate of engine switch times and estimate the approximate direction the thrust vector should be oriented at the start of each thrust arc. An initial costate vector based on these estimates will be generated and will be passed on to the optimization program, which is the previously described combination of programs. As a result, a program that does not require the user to have knowledge of optimal control principles is now available for optimal analysis of multiple-burn space missions. Several examples of solutions to missions of interest are presented to demonstrate the versatility of the program. The optimal controls for three different space missions are given.

INTRODUCTION

To reduce the high cost of space flight, thrusting maneuvers and orbit parameters must be planned so that missions are performed in an efficient manner. As missions become more complex and require more maneuvers to accomplish the mission goals, and as

*University of Houston.

spacecraft thrust levels are reduced to lower vehicle cost, it becomes increasingly difficult for mission planners to determine an efficient mission plan.

Numerous computer programs have been developed which, under various assumptions and restrictions, are intended to aid the mission planner. In many of these programs, the assumption is made that maneuvers to change the shape and size of the spacecraft orbit are made impulsively (i.e., the maneuver is assumed to be an instantaneous change in velocity). The assumption of an impulsive maneuver is valid when the spacecraft thrust level is high enough that the actual time required to make the maneuver is small with respect to the total mission time. Perhaps the most general of these programs is the one described by Jezewski and Rozendaal (ref. 1). This program, which requires only initial and final conditions, will produce a mission profile that gives the optimum number, placements, directions, and sizes of the impulsive maneuvers.

Various approaches have been applied to the problem of analyzing missions for spacecraft that have thrust levels low enough that the impulsive-thrusting approximation is not valid. One approach is to assume a near-optimum guidance algorithm and numerically integrate the equations of motion through thrusting maneuvers (e.g., ref. 2). Another approach is to assume a behavior for the vehicle controls and directly optimize the parameters that describe that behavior (e.g., ref. 3).

Brown, Harrold, and Johnson (ref. 4) applied the principles of optimal control to the problem of multiple-burn spacecraft trajectories. The result was a method by which optimum multiple-burn trajectories could be found by determining the times at which the rocket engine should be switched on and off and by determining initial values for the differential equations describing the behavior of the costate vector. The costate vector is the vector of Lagrange multipliers that adjoins the equations of motion of the state vector to the performance functional and that describes the optimal thrust vector control through each maneuver. Tarbet (ref. 5) extended the versatility of the Brown, Harrold, and Johnson program by applying a conjugate gradient algorithm as the iteration scheme for determining the optimal control. The control consists of an initial costate vector and an engine-switch-time array. To assist mission planners in determining starting values for the costate and switch times for the iteration process, Tarbet first computed an optimum impulsive solution to the problem by the method described in reference 1 and formed thrust arcs around the impulses according to the technique described in reference 6. Although Tarbet's program successfully found solutions to problems, it was of limited usefulness because numerical integration of state, costate, and perturbation differential equations across thrust arcs required a significant amount of computer time for many problems and because the scheme for determining starting iterates was not suitable for problems in which the spacecraft thrust level was low.

In an effort to reduce the time required for computing multiburn trajectories, Jezewski (ref. 7) recently produced a program based on the principles of optimal control similar to the Brown, Harrold, and Johnson program except that he made the assumption that the gravitational acceleration vector varies linearly with the radius vector. This assumption resulted in a closed-form solution to the state and costate differential equations across the thrust arcs, which greatly reduced the trajectory computation time.

Tarbet's version of the Brown, Harrold, and Johnson program and Jezewski's program, if combined, would complement each other. Jezewski's program can be used to provide a starting iterate for Tarbet's program, because the two programs are similarly structured. A starting iterate obtained from Jezewski's program would not have limitations imposed by the thrust level; and, because it is obtained from a finite-thrust program, it would be more accurate than the method of forming thrust arcs around impulses. The most obvious result of the increased accuracy would be the reduction in computer time required. Another advantage would be the ability of the combined programs to find solutions to missions performed by spacecraft having low thrust levels, for which solutions were previously not possible. On the other hand, Jezewski's program is in itself sufficiently accurate for numerous applications, and Tarbet's program could be used to verify that accuracy when necessary. Thus, the use of Jezewski's program followed by verification of certain important solutions with Tarbet's program would allow parametric studies related to mission planning without entailing protracted computer time. The development of such a combination is described in this report.

Because a requirement still exists to estimate the initial costate vector and the engine switch times, mission planners may have difficulty applying the program successfully. It is expected that the mission planner can estimate with reasonable accuracy the array of engine switch times. The theory applied in references 5 and 7 proves that on an optimal trajectory the spacecraft thrust vector is alined with the three-dimensional vector of Lagrange multipliers associated with the velocity vector. This vector comprises the first three components of the six-dimensional costate vector and is called the

primer vector. With this in mind, the engineer can make some estimate of the direction of the primer vector. The last three components of the costate vector, which are called the primer vector derivative, cannot be estimated by association with physical properties of the mission under consideration. In this study, a scheme will be presented that is derived from Jezewski's work for estimating the primer vector derivative. This scheme will require the planner to make some estimate of engine switch times and estimate the approximate direction in which the thrust vector should be oriented at the start of each thrust arc (e.g., along the velocity vector, normal to the orbit plane, etc.). An initial costate vector based on these estimates will be generated and will be passed on to the optimization program, which is composed of Jezewski's and Tarbet's programs. To demonstrate the versatility of the program, several examples of solutions to missions of interest will be presented. The optimal control for three different space missions will be given, a Shuttle abort-once-around mission and two- and three-burn geosynchronous satellite-placement missions.

As an aid to the reader, where necessary the original units of measure have been converted to the equivalent value in the Système International d'Unités (SI). The SI units are written first, and the original units are written parenthetically thereafter.

SYMBOLS

A	propagation matrix across burn-coast arc
C	value of $J'(\alpha)$ from previous iteration
c	effective exhaust velocity
d	penalty magnitude
f	function
G	gravitational vector
H	Hamiltonian function
h	transversality condition
I	identity matrix
J	cost function
L	thrust direction vector
M	functional relationship for initial state vector
m	mass; number of components in α
N	functional relationship for final state vector
n	number of thrust arcs
P	primer vector, Lagrange multiplier for acceleration constraint
p	magnitude of primer vector
Q	negative of primer vector derivative, Lagrange multiplier for velocity constraint
R	radius vector
S	switch function
T	thrust magnitude
T/w_I	thrust-to-weight ratio
t	time
V	velocity vector
W	vector of thrust arc integrals; weighting matrix
w	weight
X	state vector, $X^T = (R^T:V^T)$
α	control variable for thrust magnitude; vector of control variable, $\alpha^T = (P^T, -Q^T, t_1, ..., t_{2n+1})$
γ	number in equation (A12) chosen to ensure that matrix is positive definite
Δ	incremental
δ	variation
ϵ	Lagrange multiplier for mass
η	Lagrange multiplier for thrust magnitude constraint
λ	costate vector, $\lambda^T = (P^T:-Q^T)$
μ	gravitational constant

ν Lagrange multiplier for thrust direction constraint

Σ sum

τ time interval

Υ coast arc state transition matrix

Φ coast arc costate transition matrix

Ψ thrust arc transition matrix

Ω thrust arc transition matrix for thrust integrals

ω Schuler frequency, the constant in gravity approximation

∇ gradient operator

Subscripts:

b burn

c coast

F final

I initial

i arc index

max maximum

p penalty

x independent variables

y dependent variables

Superscripts:

* optimum

-1 inverse

[i] arc index for matrices

j variable

k any integer

T transpose

Operators:

$\dot{}$ time derivative

$\tilde{}$ used to distinguish between cost functions

FORMULATION OF THE MULTIBURN OPTIMIZATION PROBLEM

The multiburn space trajectory optimization problem is to find a set of thrusting and coasting arcs that minimizes some performance functional while transferring between two physical boundary conditions,

$$M\left(R_I, V_I, t_I\right) = 0 \tag{1}$$

and

$$N\left(R_F, V_F, t_F\right) = 0 \tag{2}$$

subject to the differential constraints

$$\dot{R} = V \tag{3}$$

$$\dot{V} = G + \frac{T}{m} L \tag{4}$$

$$\dot{m} = -\frac{T}{c} \tag{5}$$

where R, V, and m are the state variables and T and L are the control variables. The symbols R and V are,

respectively, radius and velocity vectors in Cartesian coordinates. The constraint on thrust magnitude

$$0 \le T \le T_{max} \tag{6}$$

is required as is the constraint on the thrust direction vector L that

$$L^T L = 1 \tag{7}$$

References 4 and 7 base their solution to the n-burn optimization problem on the functional

$$J = \int_{t_I}^{t_F} -\dot{m}\, dt \tag{8}$$

that is to be minimized. It should be noted that minimizing this functional is equivalent to minimizing mass loss or, alternatively, maximizing final mass. The conditions needed for optimal control of a multiburn trajectory are developed in the literature (refs. 4, 5, and 7), and the development is presented herein for the convenience of the reader.

The inequality constraint (eq. (6)) is first rewritten

$$T(T_{max} - T) - \alpha^2 = 0 \tag{9}$$

The constraints (eqs. (3), (4), (5), (7), and (9)) are adjoined to equation (8) to form the variational Hamiltonian function

$$H = -\dot{m} + Q^T V + P^T\left(G + \frac{T}{m}L\right) + \epsilon\dot{m} + \nu\left(1 - L^T L\right) + \eta\left[T(T_{max} - T) - \alpha^2\right] \tag{10}$$

where Q, P, ϵ, ν, and η are Lagrange multipliers adjoining the constraints to the functional. The necessary conditions for optimality with respect to the control L are given by

$$\frac{\partial H}{\partial L} = 0 = \frac{T}{m} P^T - 2\nu L^T \tag{11}$$

By solving the set of linear equations (11), the necessary conditions are satisfied if

$$L = \pm \frac{P}{p} \tag{12}$$

The choice of which sign to use in equation (12) is determined by applying the Weierstrass E-condition (ref. 8). It follows that

$$L = +\frac{P}{p} \tag{13}$$

Substituting equation (13) into equation (10), $-c\dot{m}$ for T, and rearranging, the Hamiltonian function becomes

$$H = -\dot{m}\left(1 - \epsilon + p\frac{c}{m}\right) + Q^T V + P^T G + \eta\left[T(T_{max} - T) - \alpha^2\right] \tag{14}$$

The term α is a control variable, so $\frac{\partial H}{\partial \alpha} = 0$ is a necessary condition.

$$\frac{\partial H}{\partial \alpha} = -2\eta\alpha = 0 \tag{15}$$

Either η or α must equal 0. The Lagrange multiplier η cannot in general be assumed to be 0. From equation (9), for T = 0 and $T = T_{max}$, α must equal

0. By making the assumption that the thrust can only take on values of 0 and T_{max}, the quantity in brackets becomes equal to 0, and the last term of equation (14) then can be deleted so that the Hamiltonian function becomes

$$H = -\dot{m}\left(1 - \epsilon + p\frac{c}{m}\right) + Q^T V + P^T G \tag{16}$$

The necessary conditions for optimality with respect to the multipliers P, Q, and ϵ are

$$\dot{P}^T = -\frac{\partial H}{\partial V} = -Q^T \tag{17}$$

$$\dot{Q}^T = -\frac{\partial H}{\partial R} = -\left(P^T \nabla\right) G \tag{18}$$

and

$$\dot{\epsilon} = -\frac{\partial H}{\partial m} = T\frac{p}{m^2} \tag{19}$$

Differential Equation Solution

Equations (3) to (5) and (17) to (19) comprise a set of differential equations that must be solved to find the optimal multiple-burn trajectory.

Coast arcs.- On coasting arcs, both references 4 and 7 assume that the gravitational vector G is given by the inverse-square relation

$$G_c = -\frac{\mu R}{|R|^3} \tag{20}$$

For an inverse-square gravitational field, the solutions for the state and costate vectors are known in closed form (refs. 9 and 4) and may be expressed as

$$X(\tau) = \Upsilon X(0) \tag{21}$$

$$\lambda(\tau) = \Phi\lambda(0) \tag{22}$$

where Υ and Φ are known 6-by-6 matrices.

Thrust arcs.- Reference 4 assumes that the gravitational vector G is also given by equation (20). No solution for the equations of motion in closed form exists assuming this gravitational vector. The solution across thrust arcs is found by using numerical integration.

Reference 7 makes a different assumption for G on thrust arcs. To facilitate the solution of the differential equations, the assumption that

$$G_b = -\omega^2 R \tag{23}$$

is made so that equations (3), (4), (17), and (18) can be solved in closed form. The term ω in equation (23) is defined by

$$\omega = \sqrt{\frac{\mu}{|R|^3}} \tag{24}$$

where $|R|$ is evaluated only at the start of the thrust arc. The solution for equations (3) and (4) is

$$X(\tau) = \Psi X(0) + \Omega W \tag{25}$$

and the solution for equations (17) and (18) is

$$\lambda(\tau) = \Psi\lambda(0) \tag{26}$$

where

$$\Psi = \begin{bmatrix} I\cos\omega\tau & I\frac{1}{\omega}\sin\omega\tau \\ -I\omega\sin\omega\tau & I\cos\omega\tau \end{bmatrix} \tag{27}$$

$$\Omega = \frac{T}{\omega}\begin{bmatrix} I\sin\omega\tau & -I\cos\omega\tau \\ I\omega\cos\omega\tau & I\sin\omega\tau \end{bmatrix} \tag{28}$$

where I is an identity matrix of dimension 3, and

$$W = \begin{bmatrix} \int_0^T \frac{P}{m(t)p} \cos \omega\tau \, dt \\ \int_0^T \frac{P}{m(t)p} \sin \omega\tau \, dt \end{bmatrix} \tag{29}$$

Boundary Conditions

To complete the solution to the multiburn optimization problem, a set of initial values for the state variables R, V, m and the variables P, Q, ϵ must be obtained. In addition, another set of variables, the set of engine switch times $t_1, \ldots, t_{2n+1}$, where n is the number of thrust arcs, must be determined. Indexing for the switch times is as follows: t_0 is the time at which the vehicle is in the initial state given by equation (1); t_{2i-1} is the time of the start of the ith thrust arc; t_{2i} is the time of the termination of the ith thrust arc; and t_{2n+1} is the time of termination of a final coast arc.

Equation (1) gives values for the state variables R and V, and the initial value of m will be specified according to the characteristics of the vehicle performing the mission. The variables P, Q, ϵ, $t_1, \ldots, t_{2n+1}$ (2n + 8 in all) must be determined by satisfaction of boundary conditions.

Six boundary conditions are obtained from equation (2). Of the remaining 2n + 2 conditions, 2n + 1 are obtained from conditions of optimality that are imposed on the Hamiltonian function, equation (16). The conditions of optimality on H are (1) H must be constant across the trajectory; (2) H must be maximized; and (3), for a time-open solution,

$$H_{2n+1} = 0 \tag{30}$$

Finally, the condition

$$\lambda_I{}^T \lambda_I = \text{constant} \neq 0 \tag{31}$$

is required because of the homogeneous property of the costate differential equations.

Equation (16) is rewritten

$$H = -\dot{m}S + h \tag{32}$$

where

$$h = P^T G + Q^T V \tag{33}$$

is called the transversality condition and

$$S = 1 - \epsilon + \frac{cp}{m} \tag{34}$$

The term S is called the switch function because the decision whether $\dot{m} = 0$ or $\dot{m} = \dot{m}_{max}$ may be based on the sign of S. When S is negative, a value of $\dot{m}$ equal to $\dot{m}_{max}$ would reduce the value of H (it must be remembered that $\dot{m}_{max}$ is a negative number), which is to be maximized. When S is negative, then $\dot{m}$ must equal 0. It follows, then, that at each engine switch time, S must equal 0. The following boundary conditions result:

$$S_i = 0 \quad i = 1, \ldots, 2n \tag{35}$$

Equations (2), (30), (31), and (35) then define the required 2n + 8 boundary conditions. Experience, however, indicates that these conditions are unsatisfactory because of the sensitivity of the switch function and because of the requirement to integrate the equation for ϵ (eq. (19)). Manipulation of the conditions of equation (35) along with the constancy of H will eliminate ϵ from the solution and reduce the number of unknown variables and of boundary conditions required to 2n + 7.

Equations (5) and (19) show that, because T = 0 on coast arcs, m and ϵ are constant on coast arcs. Solving

the equation for S_{2i} for ϵ and substituting the result into the equation for S_{2i+1} gives

$$S_{2i+1} = 0 = 1 - \left(1 + \frac{cp_{2i}}{m}\right) + \frac{cp_{2i+1}}{m} \quad i = 1, \ldots, n-1 \quad (36)$$

Simplification of equation (36) gives the conditions

$$p_{2i} = p_{2i+1} \quad i = 1, \ldots, n-1 \quad (37)$$

which states that the magnitude of the primer vector at the end of each interior coast arc is the same as at the beginning of that coast arc.

As can be easily shown, on coast arcs, H is constant; however, only on an optimum trajectory is H constant across the entire multiburn trajectory. On an optimum trajectory then, H must have the same value at the beginning and end of each thrust arc to satisfy the condition that H be a constant.

$$H_{2i} - H_{2i-1} = -\dot{m}S_{2i} + h_{2i} + \dot{m}S_{2i-1} - h_{2i-1} = 0 \quad i = 1, \ldots, n \quad (38)$$

At the beginning and end of each thrust arc, S_{2i} and S_{2i-1} must be equal to 0; thus, equation (38) reduces to

$$h_{2i} - h_{2i-1} = 0 \quad i = 1, \ldots, n \quad (39)$$

Equation (37) satisfies the condition on the switch function given in equation (35) for i = 2, ..., 2n - 1. To verify the satisfaction of equation (35) for i = 1, equation (38) should be examined for i = 1.

$$-\dot{m}S_2 + h_2 + \dot{m}S_1 - h_1 = 0 \quad (40)$$

Because S_2 is forced to be equal to 0 by the condition of equation (37), equation (40) can be rewritten

$$\dot{m}S_1 + (h_2 - h_1) = 0 \quad (41)$$

According to equation (39), $h_2 - h_1 = 0$, so S_1 must equal 0. Similarly, because $S_{2n-1} = 0$ by equation (37), equation (38) can be rewritten for i = n,

$$\dot{m}S_{2n} + (h_{2n} - h_{2n-1}) = 0 \quad (42)$$

Equation (39) requires that $(h_{2n} - h_{2n-1}) = 0$; therefore, S_{2n} must also equal 0. The boundary conditions of equations (37) and (39) then satisfy the conditions stated in equation (35). Equations (2), (30), (31), (37), and (39) form a complete set of 2n + 7 boundary conditions that may be used to determine the 2n + 7 variables P, Q, $t_1, \ldots, t_{2n+1}$.

When the gravity approximation (eq. (23)) is made on thrust arcs, the switch function at the end of each thrust arc must be modified to account for the discontinuity in the gravitational vector, so

$$S = 1 + \epsilon + \frac{cp}{m} + \frac{p^T(G_b - G_c)}{\dot{m}_{max}} \quad (43)$$

and the boundary condition (eq. (37)) becomes

$$0 = p_{2i} - p_{2i+1} - \frac{m_{2i}}{c\dot{m}_{max}} p_{2i}^T (G_b - G_c) \quad (44)$$

In summary, the multiburn trajectory optimization problem has been formulated as a boundary value problem in the 2n + 7 unknown quantities P, Q, $t_1, \ldots, t_{2n+1}$ with the differential equations (3) to (5), (17), and (18) and with the 2n + 7 boundary conditions given in equations (2), (30), (31), (37), and (39).

CONVERGENCE TO BOUNDARY CONDITIONS

After solutions to the differential equations have been obtained, what remains is a set of $2n + 7$ nonlinear equations in the $2n + 7$ unknown parameters P, Q, $t_1, \ldots, t_{2n+1}$. Evaluating these nonlinear equations requires the computation of a trajectory for which a set of values for the unknowns is given. In general, an initial estimate for the unknowns will not yield the desired solution to the nonlinear equations. An iterative convergence process is required to obtain a set of values for the $2n + 7$ unknowns, which produces the desired solution. One method of accomplishing this is to create a cost function and determine values of the parameters for which the cost function is minimized. One cost function is

$$J' = f(a)^T f(a) \tag{45}$$

where $f(a)$ is the vector function of the unknown parameters

$$a^T = \left(P^T, -Q^T, t_1, \ldots, t_{2n+1}\right) \tag{46}$$

For the solution in which the inverse-square gravity model of equation (20) is assumed on all arcs, the matrix of partial derivatives $\frac{\partial f}{\partial a}$ is known (in the sense that differential equations for $\delta\dot{X}$ and $\delta\dot{\lambda}$ must be numerically integrated on thrust arcs) and is described in reference 4. Reference 7 gives the matrix $\frac{\partial f}{\partial a}$ for the solution in which the gravity approximation of equation (23) is made on thrust arcs. With this information, numerous algorithms are applicable to this problem. Tarbet (ref. 5) applied a conjugate gradient algorithm to the full inverse-square problem of reference 4 with good results.

The algorithm described in reference 10 was used in the program of reference 7, which will be referred to as the ω program. That algorithm has been modified as shown in the appendix to increase the capability of the program. This enhancement was motivated in part by the requirement that the thrusting arcs and intermediate coasting arcs be nonnegative, which results in a parameter inequality constraint. The constraint capability was extended to initial and final coast arcs as mission requirements dictated.

STARTING ITERATES

To find a solution to the problem, an initial estimate for a must be supplied. In the program described in reference 5, an impulsive solution and finite thrust arcs created about each impulse comprise the starting iterate for the multiburn optimization program. Because of the inaccuracies inherent in approximating an impulse with a finite thrust arc, this method is restricted to relatively high (greater than 0.3) thrust-to-weight ratios (T/w_I). However, because appreciable amounts of computer time are required to find a solution, particularly for problems having long thrust arcs, an accurate estimate for a must be obtained. The ω program can be used to determine a starting value for a for the full inverse-square program because its closed-form trajectory computation permits rapid convergence. When these two programs are combined so that the ω program is first used to obtain a starting iterate for the full inverse-square program, a very versatile optimal maneuver analysis program results. The ω program can be used for preliminary mission studies and performance scans, because the rapid convergence allows such studies without excessive computer time requirements. When more accurate data or verification of data from the ω program is required, then the inverse-square program can be used.

To initialize the iteration loop in the ω program, an estimate of costate λ and the engine on and off times must be provided. It is expected that a mission engineer, relying on experience and a knowledge of the desired trajectory, can estimate the engine on and off times. The

costate is a six-dimensional vector comprised of a primer vector $\mathbf{P}$ and its derivative $\dot{\mathbf{P}}$.

$$\lambda = \begin{pmatrix} \mathbf{P} \\ \dot{\mathbf{P}} \end{pmatrix} \tag{47}$$

where $\dot{\mathbf{P}} = -\mathbf{Q}$. According to equation (13), the thrust acceleration vector is alined with and has the same direction as the primer vector $\mathbf{P}$. Thus, knowing what is to be accomplished in a particular maneuver, a mission engineer can estimate how the thrust vector should be directed (e.g., posigrade or retrograde) and, therefore, the direction of the primer vector $\mathbf{P}$. If some estimate for $|\mathbf{P}|$ at the start of each maneuver can be made, then the primer derivative $\dot{\mathbf{P}}$ is the only quantity for which an estimate is not readily obtained. Reference 11 has a scheme for the estimation of $\dot{\mathbf{P}}$ for two-burn maneuvers. The following is an extension of that scheme to n burns.

According to equation (26), the costate λ_2 at time t_2 at the end of a burn may be computed in terms of λ_1 at t_1 at the beginning of the burn by

$$\lambda_2 = \Psi\lambda_1 \tag{48}$$

where Ψ is a matrix that is a function of time only. Similarly, across coast arcs, λ_3 can be expressed in terms of λ_2 as

$$\lambda_3 = \Phi\lambda_2 \tag{49}$$

according to equation (22). Combining equations (48) and (49) gives

$$\lambda_3 = \Phi\Psi\lambda_1 \tag{50}$$

which is the solution for the costate at the beginning of the second burn arc of a multiburn solution, the time at which another estimate for $\mathbf{P}$ can be made. For conciseness, define

$$A^{[i]} = \Phi^{[i]}\Psi^{[i]} \tag{51}$$

on the ith burn-coast arc, and renumber λ so that λ_i is the value at the beginning of the ith burn-coast arc,

$$\lambda_{i+1} = A^{[i]}\lambda_i \tag{52}$$

and rewrite equation (52)

$$\begin{bmatrix} \mathbf{P}_{i+1} \\ \dot{\mathbf{P}}_{i+1} \end{bmatrix} = \begin{bmatrix} A_{11}^{[i]} & A_{12}^{[i]} \\ A_{21}^{[i]} & A_{22}^{[i]} \end{bmatrix} \begin{bmatrix} \mathbf{P}_i \\ \dot{\mathbf{P}}_i \end{bmatrix} \tag{53}$$

The derivative $\dot{\mathbf{P}}_i$, when $i = 1$, is the quantity to be estimated. Two equations involving $\dot{\mathbf{P}}_i$ can be written from equation (53)

$$\mathbf{P}_{i+1} = A_{11}^{[i]}\mathbf{P}_i + A_{12}^{[i]}\dot{\mathbf{P}}_i \tag{54}$$

and

$$\dot{\mathbf{P}}_{i+1} = A_{21}^{[i]}\mathbf{P}_i + A_{22}^{[i]}\dot{\mathbf{P}}_i \tag{55}$$

Equation (54) can be solved directly for $\dot{\mathbf{P}}_i$:

$$\dot{\mathbf{P}}_i = A_{12}^{[i]-1}\left(\mathbf{P}_{i+1} - A_{11}^{[i]}\mathbf{P}_i\right) \tag{56}$$

which gives an estimate for $\dot{P}_i$, for a two-burn problem, by setting $i = 1$. Solving equation (55) for $\dot{P}_i$ gives

$$\dot{P}_i = A_{22}^{[i]^{-1}}\left(\dot{P}_{i+1} - A_{21}^{[i]}P_i\right) \qquad (57)$$

With proper indexing, equation (56) is a solution for $\dot{P}_{i+1}$; so substituting equation (56) into equation (57) yields

$$\dot{P}_i = A_{22}^{[i]^{-1}}\left[A_{12}^{[i+1]^{-1}}\left(P_{i+2} - A_{11}^{[i+1]}P_{i+1}\right) - A_{21}^{[i]}P_i\right] \qquad (58)$$

For $i = 1$, equation (58) provides an estimate for $\dot{P}_i$ for a three-burn solution. Proceeding similarly, an estimate for a four-burn problem can be obtained by substituting equation (58) into equation (57):

$$\dot{P}_i = A_{22}^{[i]^{-1}}\left\{A_{22}^{[i+1]^{-1}}\left[A_{12}^{[i+2]^{-1}}\left(P_{i+3} - A_{11}^{[i+2]}P_{i+2}\right) - A_{21}^{[i+1]}P_{i+1}\right] - A_{21}^{[i]}P_i\right\} \qquad (59)$$

Examination of equations (56), (58), and (59) indicates that a form exists for a $\dot{P}_1$ estimate in terms of the estimates for the P vectors at the start of each burn-coast arc and that the equations lend themselves to a computer solution for $\dot{P}_1$ for a trajectory with a desired number of burn arcs. To implement this procedure, a trajectory must be calculated by using the time estimates and some guess made for P_1 and $\dot{P}_1$ to obtain the $A^{[i]}$ matrices. Usually, P_1 is specified to be a vector of unit magnitude in the desired thrust direction and $\dot{P}_1$ is chosen to be a unit vector in the gravity-acceleration direction. Given P_1 and $\dot{P}_1$, a trajectory is propagated and the state vectors, the costate vectors, and the $A^{[i]}$ matrices are computed and saved. The desired P_i vectors are computed based on the just-saved state vectors and the mission engineer's estimate of the required direction for the P vectors. These desired P vectors are then compared with those actually computed in the trajectory propagation. If any value of P_i is greater than a given tolerance in direction (about 0.6 radian), a new $\dot{P}_1$ estimate is calculated from the equations developed previously using the $A^{[i]}$ matrices and the desired P_i vectors. A new trajectory is then calculated by starting the iterative loop again. A functional flow chart for the estimation scheme is given in figure 1.

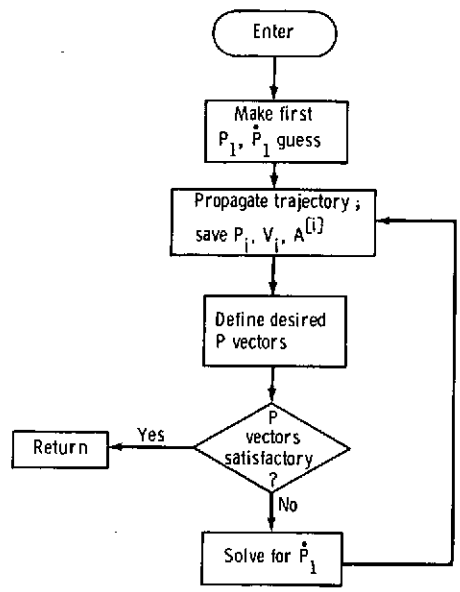


Figure 1.- Costate estimation flow chart.

EXAMPLE APPLICATIONS

The program (called OPBURN) described in the foregoing sections is very versatile. It allows mission planners to determine optimal multiple-burn trajectories for a large class of orbital-transfer problems. The mission planners need to provide the program only with data describing the initial and final conditions, the vehicle characteristics, and an approximate idea of some of the physical characteristics of the solution. To demonstrate the program capability, solutions to two types of example problems will be presented. The first type of

problem is a Shuttle abort-once-around (AOA) trajectory from the suborbital tank-staging point to the entry-interface conditions. The second type of problem for which solutions will be presented is a geosynchronous satellite-placement mission that is initiated from a low-altitude circular orbit.

Shuttle Abort-Once-Around Mission

Current plans for the launch of the Space Shuttle are to shut down the main engines before a safe orbit has been attained, separate the external tank from the orbiter so that the tank will return to Earth without needing a retrorocket, and then continue the injection of the orbiter with the orbit maneuvering system (OMS) engines. Should an abort become necessary at the tank-separation point, the current plan is to make a circumnavigation of the Earth by using OMS maneuvers to ensure that the proper entry interface for a safe landing is achieved. For certain missions, this will be the nominal profile. One of the design missions for the Shuttle is a single-orbit mission in which the Shuttle is launched and transfers to a suitable low-altitude circular orbit (e.g., approximately 185.2 kilometers (100 nautical miles) altitude), dispenses a payload, deorbits, reenters, and lands at its departure point. This mission is to be flown from the Western Test Range, and the orbit will have an inclination of 104°.

The OPBURN program was used to analyze the OMS propulsion requirements for this mission when the suborbital-tank-staging launch scheme was devised. The state vectors (R and V) were specified at the main engine cutoff and at the reentry interface, the vehicle characteristics were defined, and the program OPBURN was used to determine a sequence of two burns to achieve a transfer between the two states. The state vectors and the vehicle characteristics for the example are given in table I. It should be noted that two final state vectors are given. The solution to the AOA mission

TABLE I.- EXAMPLE SHUTTLE AOA MISSION DATA

(a) State vectors

Parameter	Initial	Final	
		Example 1	Example 2
Radius, m (ft)	6 470 763 (21 229 538)	6 388 721 (21 295 738)	6 388 721 (21 295 738)
Right ascension, deg	13.2	258.5	261.5
Declination, deg	0	0	0
Velocity, m/sec (ft/sec) . . .	7833 (25 700)	7818 (25 650)	7833 (25 700)
Flightpath angle, deg	.2	-.715	-.815
Azimuth, deg	90	90	90

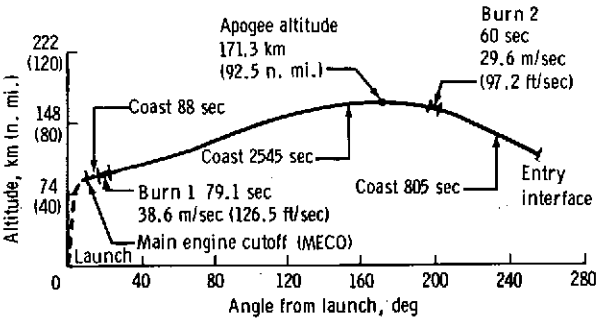
(b) Vehicle characteristics

Parameter	Value
Initial weight, kg (lb)	110 239 (243 031)
Thrust, N (lbf)	53 378 (12 000)
Specific impulse, sec	313.2

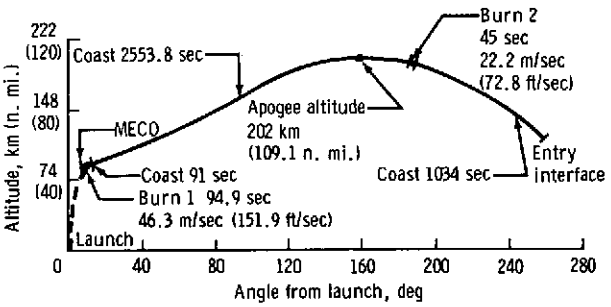
having the first given final state vector will include the full inverse-square solution from the final phase of the program. To demonstrate the inequality constraint capability of OPBURN, solutions to the AOA mission having the second given final state vector will be presented with and without imposition of an inequality constraint on the initial coast arc. Although the mission has a 104° orbital inclination, it is completely coplanar; for simplicity, the problem was formulated for OPBURN in the equatorial plane. It should be noted that the vehicle T/w_I ratio is very small (0.049). In addition to the data in table I, the thrust direction at the start of the first burn arc was specified to be posigrade, whereas the thrust direction at the start of the second burn arc was retrograde. Estimates of engine on and off times were provided.

Table II gives data describing the solution to the first AOA problem. The times shown in the "Initial estimate" column are those provided the program as input data. The costate vector in the "Initial estimate" column is produced by the costate estimation routine, and the orbital parameters that follow it describe the orbits and thrust arcs that result from state propagation using that costate estimate. Using the solution described as the starting iterate, the ω program converged in 13 iterations. Parameters describing that solution are given in the "ω approximation" column. It should be noted that the initial coast length was increased by approximately 87 seconds. The first thrust arc was decreased by 15 seconds, and the second thrust arc was delayed for approximately 300 seconds and increased 13 seconds in duration. The costate vector, however, was changed only slightly in obtaining the converged solution. When the converged solution from the ω program was used as a starting point, the inverse-square program converged in 11 iterations. Eleven iterations were required because the conjugate gradient iterator was forced to take as many iterations as unknowns in the problem (in this case, the six costate components and the four times shown in table II plus the time of termination of a final coast arc). Parameters describing that solution are also listed in table II. The similarity of the ω and inverse-square solutions indicates that, at least for this class of problem, the ω program has sufficient accuracy to preclude the need of the inverse-square program for every desired solution. An altitude profile of this mission is shown in figure 2(a).

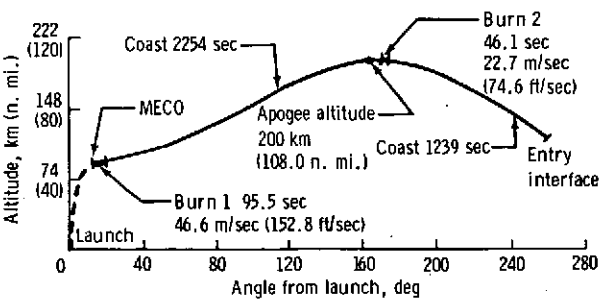
The ω program is formulated to determine the optimum transfer between the two orbits specified by the input initial and final state vectors. By determining the length of the coast arcs before the first burn and after the last burn while allowing their length to be



(a) Example 1.



(b) Example 2, no constraint on initial coast.



(c) Example 2, constrained initial coast.

Figure 2.- Shuttle AOA mission profiles.

TABLE II.- SHUTTLE AOA SOLUTION SUMMARY

Parameter	Initial estimate	ω approximation	Inverse-square solution
Engine-switch-time array, sec			
t_0	0	0	0
t_1	1	87.7	87.8
t_2	95	166.8	166.9
t_3	2413	2711.6	2711.6
t_4	2460	2771.7	2771.8
Costate vector			
λ_1	-0.226	-0.271	-0.271
λ_2	.974	.854	.854
λ_3	0	0	0
λ_4	-.385	-.443	-.443
λ_5	-.127	-.033	-.033
λ_6	0	0	0
Intermediate orbit			
Perigee altitude, km (n. mi.)	89.3 (48.2)	91.3 (49.3)	91.3 (49.3)
Apogee altitude, km (n. mi.)	197.8 (106.8)	171.1 (92.4)	171.3 (92.5)
ΔV_1, m/sec (ft/sec)	45.8 (150.4)	38.5 (126.5)	38.5 (126.4)
ΔV_2, m/sec (ft/sec)	23.1 (76.1)	29.6 (97.2)	29.7 (97.4)
$\Sigma\Delta V$, m/sec (ft/sec)	69.0 (226.5)	68.2 (223.7)	68.2 (223.8)

either positive or negative, departure from the initial orbit and arrival at the final orbit at the optimum position are ensured. In certain instances, however, mission considerations preclude the usefulness of solutions that have negative initial or final coasts. For example, in the foregoing description of the AOA mission, the initial state vector was given at the Shuttle main engine cutoff in a launch trajectory, making it impossible for the mission designer to initiate the transfer at a position on the initial orbit previous to the input state vector. If the final state vector in the previous problem is changed to that in the last column of table I, a negative initial coast of 91.0 seconds is required as indicated by the data in the "Unconstrained" column of table III. Imposing the

TABLE III.- INEQUALITY CONSTRAINT EXAMPLE

Parameter	Unconstrained	Constrained
Engine-switch-time array, sec		
t_0	0	0
t_1	-90.9	1.03
t_2	3.97	96.5
t_3	2557.8	2350.2
t_4	2602.8	2396.3
Costate vector		
λ_1	-0.184	-0.195
λ_2	.882	.888
λ_3	0	0
λ_4	-.427	-.413
λ_5	-.12	-.108
λ_6	0	0
Intermediate orbit		
Perigee altitude, km (n. mi.)	86.5 (46.7)	89.3 (48.2)
Apogee altitude, km (n. mi.)	202.1 (109.1)	200.0 (108.0)
ΔV_1, m/sec (ft/sec)	46.3 (151.9)	46.6 (152.8)
ΔV_2, m/sec (ft/sec)	22.1 (72.8)	22.7 (74.6)
$\Sigma \Delta V$, m/sec (ft/sec)	68.5 (224.7)	69.3 (227.4)

parameter inequality constraint developed in the appendix on the initial time yields the solution indicated in the last column. It should be noted that the initial coast time is now 1.03 second and that, as expected, the ΔV required for the transfer is higher. The constraint is stated such that the coast arc is only required to be nonnegative, and a 0-second coast arc can be expected. The actual constraint as implemented in the appendix is "hard" only in the limit of $k \to \infty$, so the program cannot satisfy the constraint precisely. A hard constraint is one in which the quantity being constrained may take on a value equal to the value of the constraint. In the example under discussion, the initial coast arc would have a 0-second duration if the constraint were hard.

Altitude profiles for the unconstrained and constrained example AOA missions are illustrated in figures 2(b) and 2(c). The profiles are very similar. It should be noted that the second burn arc occurs closer to apogee in the constrained solution.

Synchronous-Orbit Missions

A general-purpose rocket vehicle called a tug is being developed to provide the additional propulsion required for satellite-placement missions. In an effort to save weight, some of the several designs being considered entail the use of a rather small engine, which results in a low thrust-to-weight ratio. This sometimes complicates mission-planning efforts to ensure that the tug will be used as efficiently as possible. One particularly difficult mission, for which numerous payloads are possible, is the placement of a satellite in a geosynchronous orbit. This mission is difficult because of the large altitude change and the large plane change (29°) required between the initial and final circular orbits. The altitude of a circular geosynchronous orbit is 35 786 kilometers (19 323 nautical miles). As a further example of the capability of OPBURN, this mission was chosen for solutions with vehicle thrust-to-weight ratios as low as 0.1. Solutions for the mission will include two- and three-burn profiles.

TABLE IV.- EXAMPLE GEOSYNCHRONOUS-ORBIT MISSION DATA

(a) State vectors

Parameter	Initial	Final
Radius, m (ft)	6 655 965 (21 837 155)	42 164 047 (138 333 497)
Right ascension, deg	-90	180
Declination, deg	0	0
Velocity, m/sec (ft/sec)	7738.64 (25 389.25)	3074.67 (10 087.52)
Flightpath angle, deg	0	0
Azimuth, deg	90	61

(b) Vehicle characteristics

Parameter	Value
Initial weight, kg (lb)	45 360 (100 000)
Thrust, N (lbf)	133 447 (30 000) 88 964 (20 000) 44 482 (10 000)
Specific impulse, sec	420

The geosynchronous satellite-placement mission is initiated in a low-altitude circular orbit sized according to the capabilities of the Shuttle. For this example, a circular orbit of 278 kilometers (150 nautical miles) altitude was assumed. The initial state vector is given in table IV. For ease of input, a coordinate system was established so that the initial orbit was in the X-Y plane, and the line of intersection (i.e., the line of nodes) of the initial and final orbit planes was along the X-axis (fig. 3). The final state vector, also given in table IV, corresponds to a circular orbit with an altitude of 35 786 kilometers (19 323 nautical miles) and at an inclination of 29° from the X-Y plane. Vehicle characteristics shown were chosen to produce vehicle T/w_I of 0.3, 0.2, and 0.1. The specific impulse selected is that of a proposed oxygen difluoride/methane propellant tug.

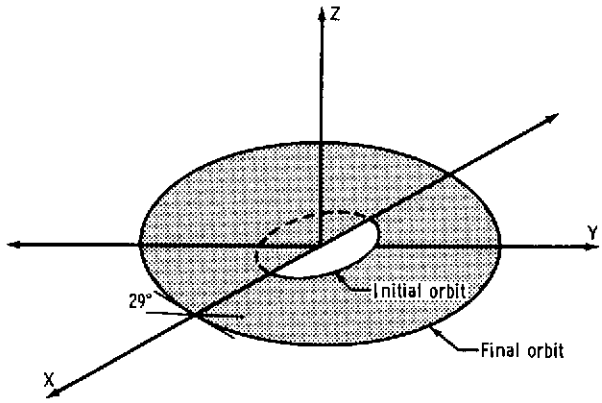
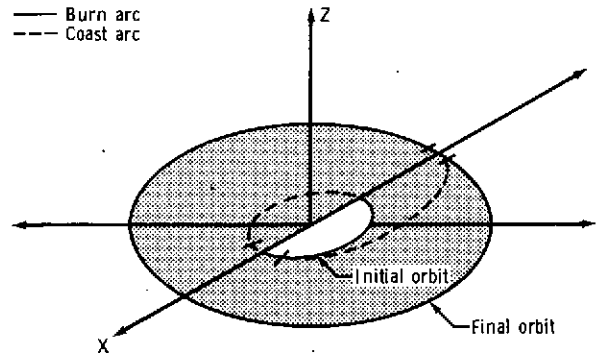
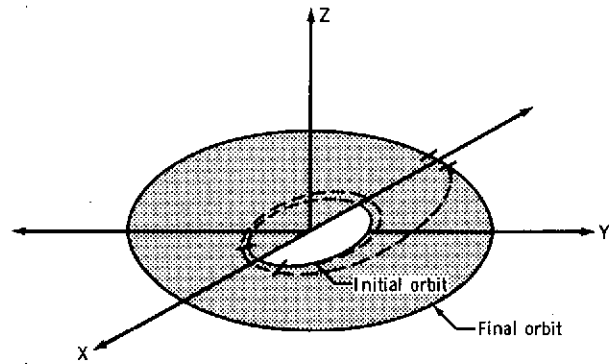


Figure 3.- Illustration of initial and final orbit geometry of geosynchronous-orbit mission.

The two-burn mission profile consists of two thrust arcs. The first, which is centered approximately on the line of nodes, places the spacecraft into an elliptical transfer orbit with an apogee altitude approximately that of the desired final circular orbit. The second thrust arc, which occurs at the apogee of the transfer orbit, circularizes the orbit at the desired altitude. Each thrust arc concurrently performs a portion of the required plane change; in other words, the first thrust arc achieves a small part of the required plane change, and the second completes it. In the three-burn profile, the function of the first thrust arc in the two-burn profile is achieved by the first two thrust arcs. These thrust arcs are separated by approximately one revolution in an elliptical orbit and centered approximately on the same node. Each mission profile is illustrated in figure 4.



(a) Two-burn arc.



(b) Three-burn arc.

Figure 4.- Illustration of geosynchronous-orbit mission profiles.

As in the previous example, estimates of the engine switch times were provided and the costate estimation routine was used to produce a costate vector estimate based on the knowledge that all the burns would be posigrade. These estimates for the control variables were then transferred to the ω program for convergence to an estimate for the optimal control. The optimal control estimate from the ω program was used as a starting iterate for the inverse-square program in the two- and three-burn cases.

Parameters describing optimal two- and three-burn geosynchronous-orbit missions for vehicles with a T/w_I of 0.3 are listed in table V. The time arrays listed in the "initial estimate" columns were the estimates provided the program; the costates in those columns are produced by the costate estimation routine. The columns labeled "ω solution" give the values of

TABLE V.- GEOSYNCHRONOUS MISSION SOLUTION PARAMETERS, $T/w_I = 0.3$

Parameter	Two thrust arcs			Three thrust arcs		
	Initial estimate	ω solution	Inverse-square solution	Initial estimate	ω solution	Inverse-square solution
Engine-switch-time array, sec						
t_0	0	0	0	0	0	0
t_1	900	956	1 005	1 180	1 180	1 175
t_2	1 650	1 591	1 637	1 520	1 518	1 514
t_3	20 320	20 252	20 250	10 660	10 662	10 654
t_4	20 600	20 522	20 521	10 950	10 953	10 944
t_5	--	--	--	29 600	29 656	29 756
t_6	--	--	--	29 880	29 928	29 848
Costate vector						
λ_1	0.362	0.699	0.691	1.14	0.680	0.687
λ_2	.227	.185	.202	-.064	.219	.211
λ_3	0	-.0007	.0021	.529	.00023	.00035
λ_4	.452	.173	.189	-.075	.205	.197
λ_5	.736	.651	.647	.478	.638	.645
λ_6	0	-.168	-.165	-.1	-.172	-.169
Intermediate orbits						
Perigee altitude, km (n. mi.)	420 (227)	374 (202)	352 (190)	293 (158)	289 (156)	287 (155)
	--	--	--	311 (168)	298 (161)	296 (160)
Apogee altitude, km (n. mi.)	126 725 (68 426)	35 784 (19 322)	35 784 (19 322)	6389 (3450)	6302 (3403)	6334 (3420)
	--	--	--	36 527 (19 723)	35 786 (19 323)	35 786 (19 323)
Δ inclination, deg						
Burn 1	0	2.13	2.15	0.8	1.16	1.17
Burn 2	0	26.87	26.85	.28	1.04	1.03
Burn 3	--	--	--	3.5	26.8	26.8
ΔV, m/sec (ft/sec)						
Burn 1	3163 (10 376)	2494 (8183)	2479 (8133)	1147 (3762)	1140 (3741)	1144 (3752)
Burn 2	2329 (7621)	1789 (5868)	1787 (5864)	1317 (4323)	1323 (4339)	1318 (4325)
Burn 3	--	--	--	1863 (6112)	1792 (5880)	1793 (5883)
$\Sigma\Delta V$, m/sec (ft/sec)	5485 (17 997)	4283 (14 052)	4269 (14 007)	4327 (14 197)	4255 (13 960)	4255 (13 960)

parameters describing the solutions obtained from the ω program. The values describing the solutions obtained from the full inverse-square program are listed in the columns labeled "Inverse-square solution." The two three-burn solutions are in close agreement, which indicates that the gravity approximation in the ω program is accurate in this case. The ω solution to the two-burn problem does not agree with the inverse-square solution as well as the ω solution does with the three-burn problem. It should be noted that the ΔV require-

TABLE VI.- GEOSYNCHRONOUS MISSION SOLUTION PARAMETERS, $T/w_I = 0.2$

Parameter	Two thrust arcs			Three thrust arcs		
	Initial estimate	ω solution	Inverse-square solution	Initial estimate	ω solution	Inverse-square solution
Engine-switch-time array, sec						
t_0	0	0	0	0	0	0
t_1	800	758	831	1 140	1 140	1 043
t_2	1 800	1 723	1 788	1 725	1 726	1 618
t_3	20 320	20 238	20 233	10 920	11 868	11 793
t_4	20 540	20 636	20 634	11 340	12 235	12 165
t_5	--	--	--	30 050	31 015	31 030
t_6	--	--	--	30 460	31 418	31 437
Costate vector						
λ_1	1.05	0.636	0.696	1.10	0.696	0.691
λ_2	-.053	.093	.190	-.064	.104	.202
λ_3	0	0	0	0	.00006	-.000088
λ_4	-.05	.087	.179	-.601	.098	.189
λ_5	1.02	.742	.651	1.05	.686	.647
λ_6	0	-.167	-.16	0	-.151	-.165
Intermediate orbits						
Perigee altitude, km (n. mi.)	576 (311)	472 (255)	446 (241)	317 (171)	322 (174)	311 (168)
	--	--	--	413 (223)	383 (207)	324 (175)
Apogee altitude, km (n. mi.)	44 818 (24 200)	35 784 (19 322)	35 784 (19 322)	6947 (3751)	8097 (4372)	7852 (4240)
	--	--	--	37 281 (20 130)	35 784 (19 322)	35 792 (19 326)
Δ inclination, deg						
Burn 1	0	2.03	2.08	0	1.33	1.29
Burn 2	0	26.97	26.92	0	.87	.88
Burn 3	--	--	--	--	26.8	26.83
ΔV, m/sec (ft/sec)						
Burn 1	2665 (8745)	2539 (8329)	2507 (8225)	1212 (3977)	1348 (4423)	1319 (4326)
Burn 2	920 (3017)	1785 (5857)	1786 (5858)	1288 (4226)	1145 (3756)	1152 (3779)
Burn 3	--	--	--	1827 (5995)	1787 (5864)	1792 (5879)
$\Sigma\Delta V$, m/sec (ft/sec)	3585 (11 762)	4324 (14 186)	4292 (14 082)	4328 (14 198)	4280 (14 043)	4262 (13 984)

ment for the ω solution is 13.7 m/sec (45 ft/sec) higher than that found by the inverse-square program. All the increase is required in the first thrust arc, which indicates that the gravity approximation does not have sufficient accuracy on that arc. Because the gravity approximation (eq. (23)) is a linear function of position, the accuracy of the approximation on a thrust arc will depend on the altitude excursion of the vehicle in that arc.

Data describing the same missions for vehicles with a $T/w_I = 0.2$ are given in table VI. Again, acceptable agreement exists between the ω and inverse-square

solutions. Because of the lower T/w_I, longer thrust arcs are required to produce a given orbital change, so a larger altitude excursion results and the ω approximation is less accurate for these solutions than when the T/w_I was 0.3. This fact is evident by examining the characteristic velocity requirements for the two-burn solution in table VI. The ω program indicates a ΔV requirement 31.4 m/sec (103 ft/sec) greater than the requirement found by the inverse-square program. All this increased cost is required to accomplish the first thrust arc. In the three-burn case, the ω solution requires a total of 18 m/sec (59 ft/sec) more than in the inverse-square solution. The increased cost cannot be attributed as easily to any particular thrust arc because of the differences in the solutions. For example, the first thrust arc required 29.6 m/sec (97 ft/sec) more in the ω solution than in the inverse-square solution; however, the apogee of the first intermediate orbit is 244 kilometers (132 nautical miles) higher in the ω solution. The transfer to that orbit would require a larger ΔV.

Parameters describing two- and three-burn solutions to the geosynchronous-orbit mission for vehicles with a T/w_I of 0.1 are listed in table VII. Data describing the inverse-square solution are provided only for the two-burn case. No inverse-square solution is provided for the three-burn case because the ω solution proved to be unsatisfactory as a starting iterate for the inverse-square program. The large difference between the ω and inverse-square solutions in the two-burn case should be noted. The difference indicates that the ω program solution has little value because of the inaccuracy of the gravity approximation, although the ω program did converge given the time array shown and the costate estimate produced by the costate estimation routine. Also, although the inverse-square solution did converge, the ω solution was not suitable as a starting iterate in the inverse-square program because the ω solution produced a large initial error in boundary condition satisfaction in the inverse-square program and protracted computer time was required to attain convergence.

The performance data for the geosynchronous-orbit missions were plotted as a function of T/w_I (fig. 5) to illustrate the effects of lower thrust-to-weight ratios and the benefits of increasing the number of thrust arcs used to accomplish the mission. The graph indicates a 152 m/sec (500 ft/sec) penalty for reducing T/w_I from 0.3 to 0.1 when a two-burn transfer is used. The ω solution curve for three-burn transfers indicates a penalty of 91

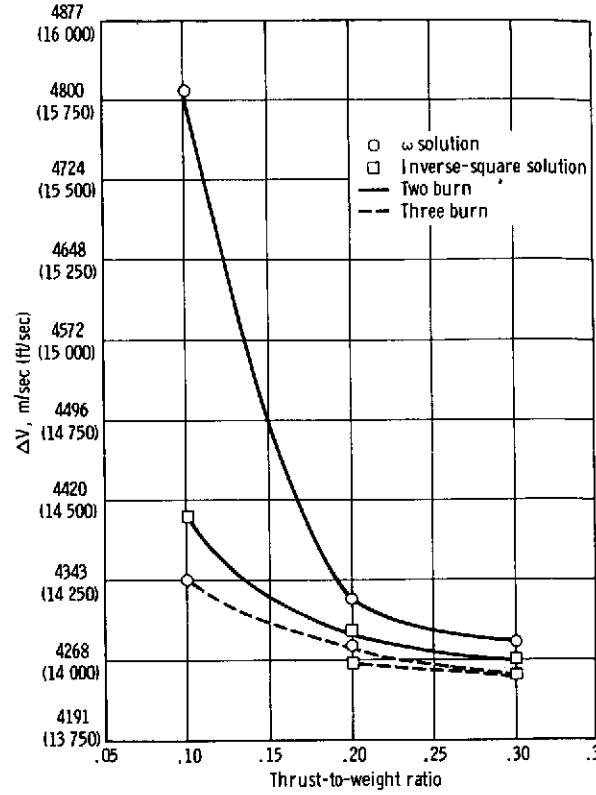


Figure 5.- Performance comparison for various geosynchronous-orbit transfers.

m/sec (300 ft/sec) for reducing T/w_I from 0.3 to 0.1. Some of that penalty is caused by the gravity assumption in the ω program, as is evident by comparing the penalties indicated for reducing T/w_I from 0.3 to 0.2. The ω solution curve shows a 25-m/sec (83 ft/sec) penalty, whereas the inverse-square solution curve indicates a penalty of only 7 m/sec (24 ft/sec). The three ω solutions for $T/w_I = 0.1$ show that a definite though diminishing performance improvement is possible by increasing the number of thrust arcs.

In all cases investigated, the costate estimation routine produced estimates for the costate vector from which the ω program converged. A more accurate time estimate was available in some cases than in others; this resulted in fewer iterations being required by the ω program. The time estimates given in tables V to VII do not necessarily indicate the accuracy required by the program but were simply the best available.

TABLE VII.- GEOSYNCHRONOUS MISSION SOLUTION PARAMETERS, $T/w_I = 0.1$

Parameter	Two thrust arcs			Three thrust arcs	
	Initial estimate	ω solution	Inverse-square solution	Initial estimate	ω solution
Engine-switch-time array, sec					
t_0	0	0	0	0	0
t_1	425	-641	304	1 076	1 076
t_2	2 375	1 567	2 294	2 050	2 051
t_3	20 360	20 628	20 231	9 872	9 872
t_4	21 150	21 313	21 001	10 850	10 850
t_5	--	--	--	29 162	29 163
t_6	--	--	--	29 947	29 948
Costate vector					
λ_1	1.02	0.901	0.710	1.09	1.02
λ_2	-.032	-.375	.148	-.063	-.012
λ_3	0	-.0007	.001	0	$-.3 \times 10^{-6}$
λ_4	-.029	-.352	.139	-.059	-.0144
λ_5	1.01	.962	.659	1.04	1.1
λ_6	0	-.218	-.143	0	-.200
Intermediate orbits					
Perigee altitude, km (n. mi.)	1472 (795)	1606 (867)	963 (520)	430 (232)	407 (220)
	--	--	--	733 (396)	648 (350)
	--	--	--	5600 (3024)	5560 (3005)
Apogee altitude, km (n. mi.)	21 172 (11 432)	35 782 (19 321)	35 782 (19 321)	36 058 (19 470)	35 782 (19 321)
Δ inclination, deg					
Burn 1	0	1.48	1.84	0	0.98
Burn 2	0	27.52	27.16	0	1.19
Burn 3	--	--	--	0	26.83
ΔV, m/sec (ft/sec)					
Burn 1	2723 (8934)	3073 (10 081)	2644 (8675)	1087 (3565)	1088 (3569)
Burn 2	1781 (5844)	1736 (5696)	1761 (5776)	1487 (4880)	1487 (4879)
Burn 3	--	--	--	1770 (5804)	1771 (5810)
$\Sigma\Delta V$, m/sec (ft/sec)	4352 (14 278)	4809 (15 777)	4405 (14 451)	4344 (14 251)	4346 (14 259)

CONCLUSIONS AND RECOMMENDATIONS

A program is now available for optimal analysis of multiple-burn space missions. The program does not require that the user have knowledge of optimal control principles. This new program is the result of the development of the costate estimation subroutine that requires the user to input only physical quantities that can be identified and estimated. Through the use of the ω/inverse-square combination, the author has become confident that the solutions produced by the ω program are accurate, so that the inverse-square phase of OPBURN is not generally required. It is retained in OPBURN as a valuable program for solution verification of particular missions, especially when a new class of solutions is being produced.

Although the program produces useful results, three improvements and additions are suggested as follows.

1. The capability to segment thrust arcs in the ω program should be added, reinitializing the gravity approximation each time. This addition would increase the accuracy of the ω program and reduce or eliminate the inaccuracies noted in the geosynchronous missions that result from large altitude excursions on long thrust arcs. Through the use of a segmented solution, the range of cases for which the ω solution would be a suitable starting iterate should be increased (e.g., the three-burn geosynchronous mission with $T/w_I = 0.1$).

2. Parameter inequality constraints on the switch times in the inverse-square program should be added. This addition would make the ω/inverse-square combination compatible for a wider range of cases.

3. An interactive program should be developed for mission planning. This program would accept input data, call the costate estimation program, and display the resulting trajectory. With such a program, the mission planner could interactively determine thrust-arc placement and size so that the resulting trajectory could be transferred to an execution of the OPBURN program with a greater chance of convergence to the desired answer.

Lyndon B. Johnson Space Center
National Aeronautics and Space Administration
Houston, Texas, September 20, 1974
986-16-20-00-72

APPENDIX

PARAMETER INEQUALITY CONSTRAINTS

To restrict the ω program so that coast and burn arcs of negative time duration are prohibited, it is necessary to implement an inequality constraint. The algorithm provided in the ω program (ref. 9) was modified to include an inequality constraint by the penalty function approach. The penalty function described in reference 12 was selected because it has the property that it and all its derivatives are continuous. The penalty function is given by

$$J_p(\alpha) = \sum_{i=1}^{m} d_i e^{\left[1-\left(1+\alpha_i\right)^k\right]} \tag{A1}$$

where α is the m component vector of parameters, d is a positive penalty constant, and k is some positive integer.

Given the vector function $f(\alpha)$, we wish to find α such that $f(\alpha) = 0$, subject to the constraints that some or all the parameters α_i remain nonnegative. First, form the function

$$J'(\alpha) = J(\alpha) + J_p(\alpha) \tag{A2}$$

where

$$J(\alpha) = \tfrac{1}{2} f(\alpha)^T W_y f(\alpha) \tag{A3}$$

and expand it in a Taylor series to second-order terms about α^*, the constrained solution to $f(\alpha) = 0$:

$$J'(\alpha) = J(\alpha^*) + J_p(\alpha^*) + \left(\frac{\partial J}{\partial \alpha}\bigg|_{\alpha^*} + \frac{\partial J_p}{\partial \alpha}\bigg|_{\alpha^*}\right)(\alpha - \alpha^*)$$

$$+ \tfrac{1}{2}(\alpha - \alpha^*)^T\left(\frac{\partial^2 J}{\partial \alpha^2}\bigg|_{\alpha^*} + \frac{\partial^2 J_p}{\partial \alpha^2}\bigg|_{\alpha^*}\right)(\alpha - \alpha^*) \tag{A4}$$

At α^*,

$$\frac{\partial J}{\partial \alpha}\bigg|_{\alpha^*} + \frac{\partial J_p}{\partial \alpha}\bigg|_{\alpha^*} = 0 \tag{A5}$$

so that

$$J'(\alpha) = J(\alpha^*) + J_p(\alpha^*) + \tfrac{1}{2}(\alpha - \alpha^*)^T\left(\frac{\partial^2 J}{\partial \alpha^2}\bigg|_{\alpha^*} + \frac{\partial^2 J_p}{\partial \alpha^2}\bigg|_{\alpha^*}\right)(\alpha - \alpha^*) \tag{A6}$$

Using this J', find the gradient

$$G'(\alpha) = \frac{\partial J'}{\partial \alpha} = (\alpha - \alpha^*)^T\left(\frac{\partial^2 J}{\partial \alpha^2}\bigg|_{\alpha^*} + \frac{\partial^2 J_p}{\partial \alpha^2}\bigg|_{\alpha^*}\right) \tag{A7}$$

It is now possible to solve for $\Delta\alpha$ by assuming that the second-order terms vary slowly $\left(\text{i.e., } \frac{\partial^2 J'(\alpha)}{\partial \alpha^2}\bigg|_{\alpha^*} \approx \frac{\partial^2 J'(\alpha)}{\partial \alpha^2}\bigg|_{\alpha}\right)$

$$\Delta\alpha = \alpha^* - \alpha = -\left(\frac{\partial^2 J}{\partial \alpha^2} + \frac{\partial^2 J_p}{\partial \alpha^2}\right)^{-1} G'(\alpha)^T \tag{A8}$$

All that remains is to determine $G'(\alpha)$ and $\frac{\partial^2 J}{\partial \alpha^2}$. Differentiating equation (A2) yields

$$G'(\alpha)^T = \frac{\partial J^T}{\partial \alpha} + \frac{\partial J_p^T}{\partial \alpha} = \frac{\partial f^T}{\partial \alpha} W_y f + \frac{\partial J_p^T}{\partial \alpha} \tag{A9}$$

and differentiating equation (A3) twice gives

$$\frac{\partial^2 J}{\partial \alpha^2} = \frac{\partial f^T}{\partial \alpha} W_y \frac{\partial f}{\partial \alpha} + f^T W_y \frac{\partial^2 f}{\partial \alpha^2} \qquad \text{(A10)}$$

which contains the undesirable term $\frac{\partial^2 f}{\partial \alpha^2}$. To circumvent the complex computation involved in the last term, it is approximated by.

$$f^T W_y \frac{\partial^2 f}{\partial \alpha^2} = \gamma W_x \qquad \text{(A11)}$$

so that

$$\Delta\alpha = -\left(\frac{\partial f^T}{\partial \alpha} W_y \frac{\partial f}{\partial \alpha} + \frac{\partial^2 J_p}{\partial \alpha^2} + \gamma W_x\right)^{-1}\left(\frac{\partial f}{\partial \alpha} W_y f + \frac{\partial J_p}{\partial \alpha}^T\right) \qquad \text{(A12)}$$

The symbol γ is a positive number chosen so that the matrix in the first set of parentheses of equation (A12) is positive definite. Note from equation (A1) that $J_p(\alpha)$ is a summation of terms each of which is a function of only one parameter. Because of this property, the gradient $\frac{\partial J_p}{\partial \alpha}$ is a row vector with m terms, and each term $\frac{\partial J_p}{\partial \alpha_i}$ is a function of the ith parameter only. Proceeding to the second derivative $\frac{\partial^2 J_p}{\partial \alpha^2}$, note that it is an m-by-m diagonal matrix with the i,ith element a function of the ith parameter only.

In the ω program, the vector α is composed of six costate components and a 2n +1 array of times. Because it is desired to constrain the times only, $d_i = 0$, $i = 1, ..., 6$. The value for the remaining d_i will be set to

$$d = 10^j C \qquad \text{(A13)}$$

where C is the value of $J'(\alpha)$ of the previous iteration. Initially, $j = 2$ with provision for its increase should the constraint be violated. The value of k in equation (A1) is nominally set to 2501, which produces a minimum arc duration of less than 1 second for each time constrained. The odd number 2501 is required to eliminate symmetry from equation (A1). If the penalty function is allowed to be symmetric, then the possibility exists for a solution that violates the constraint but for which no penalty is assessed (i.e., $\alpha_i < -2$).

The two weighting matrices W_x and W_y will normally be identity matrices. However, they can be defined as required for special situations so long as W_y is positive definite. An algorithm exists in the ω program to modify W_y so that the tolerance to which optimality conditions are satisfied is increased. This algorithm is provided because it may not be possible to satisfy the optimality conditions completely in cases in which the first or the last or both coast arcs are constrained.

REFERENCES

1. Jezewski, D. J.; and Rozendaal, H. L.: An Efficient Method for Calculating Optimal Free-Space N-Impulse Trajectories. AIAA J., vol. 6, no. 11, Nov. 1968, pp. 2160-2165.
2. McAdoo, S. F.; and Funk, J.: An Earth Departure Technique for Use in Manned Interplanetary Missions Using the NERVA Engine. AAS 70-039, presented at the AAS/AIAA Astrodynamics Conference, June 1970.
3. Johnson, I. L.; and Kamm, J. L.: Near Optimal Shuttle Trajectories Using Accelerated Gradient Methods. AAS/AIAA no. 328, presented at the AAS/AIAA Astrodynamics Specialists Conference, Aug. 1971.
4. Brown, K. R.; Harrold, E. F.; and Johnson, G. W.: Rapid Optimization of Multiple-Burn Rocket Flights. NASA CR-1430, 1969.
5. Tarbet, J. D.: Optimum Continuous Control by a Method of Parameterization. Ph. D. Thesis, Univ. of Houston, 1971.
6. Robbins, H. M.: An Analytical Study of the Impulsive Approximations. AIAA J., vol. 4, no. 8, Aug. 1966, pp. 1417-1423.
7. Jezewski, D. J.: N-Burn Optimal Analytic Trajectories. AIAA J., vol. 11, no. 10, Oct. 1973, pp. 1373-1376.
8. Lawden, D. F.: Optimal Trajectories for Space Navigation. Butterworths (London), 1963.
9. Goodyear, W. H.: Completely Closed-Form Solution for Coordinates and Partial Derivatives of the Two-Body Problem. Astron. J., vol. 70, no. 3, Apr. 1965, pp. 189-192.
10. Armstrong, E. S.: A Combined Newton-Raphson and Gradient Parameter Correction Technique for Solution of Optimal-Control Problems. NASA TR R-293, 1968.
11. Jezewski, D. J.: Optimal Analytic Multiburn Trajectories. AIAA J., vol. 10, no. 5, May 1972, pp. 680-685.
12. Jezewski, D. J.; and Faust, N. L.: Inequality Constraints in Primer-Optimal, N-Impulse Solutions. AIAA J., vol. 9, no. 4, Apr. 1971, pp. 760-763.