To the Graduate Council:

I am submitting herewith a thesis written by Robert Joseph Vosburgh entitled "Abort Strategies for Interplanetary Missions." I recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Aerospace Engineering.

T. Conley Powell
T. Conley Powell, Major Professor

We have read this thesis
and recommend its acceptance:

J. Dea

H C Reddy

Accepted for the Council:

L Evans Cook
Vice Chancellor
Graduate Studies and Research

Thesis
T1
.N539
cop. 2

ABORT STRATEGIES FOR INTERPLANETARY MISSIONS

A Thesis

Presented for the

Master of Science

Degree

The University of Tennessee, Knoxville

Robert Joseph Vosburgh

December 1977

1346433

DEDICATION

Abort Strategies for Interplanetary Missions is dedicated to Mr. and Mrs. W. J. Vosburgh, whose support and reinforcement have been of immense help to the author in completing this thesis.

ACKNOWLEDGMENTS

The author is grateful for the unending support rendered by Dr. T. Conley Powell, and the remaining members of the committee, Dr. T. W. Feagin and Dr. K. C. Reddy.

Thanks are also due to Mrs. June Jarrell for her assistance in typing this thesis.

ABSTRACT

The purpose of this study was to develop an algorithm for determining the minimum ΔV transfer between any two orbits. Earth-Mars missions were chosen for illustrating the algorithm and its actual application to abort situations.

Derivations were done to illustrate the large number of methods available for solving the problem and the best situations for each method to be used.

Numerical results showed the particular solutions to the problem for a number of likely situations.

The appendices provide subroutines, with user instructions, that can be blocked together to solve a variety of similar problems.

TABLE OF CONTENTS

CHAPTER		PAGE
I.	INTRODUCTION	1
II.	DERIVATION OF ALGORITHMS	13
	1. Introduction.	13
	2. Orbit Determination	17
	3. Gauss Method.	20
	4. Semi-Latus Rectum Iteration Method.	29
	5. True Anomaly Iteration Method	35
	6. F and G Series.	37
	7. Universal Variables	44
	8. Velocity Differential Determination	46
	9. Newton-Raphson Minimization	48
	10. Final Problem Algorithm	50
III.	NUMERICAL RESULTS.	51
	1. Introduction.	51
	2. Nominal Earth-Mars Trajectories	52
	3. Early Manned Missions	52
	4. Advanced Manned Missions.	57
	5. Freighter Missions.	60
BIBLIOGRAPHY		63
APPENDICES		65
APPENDIX A.	SUBROUTINE OD(R,V,A,XMA,E,XI,XLN,AP,TA)	66
APPENDIX B.	SUBROUTINE PITR(R1,R2,T,A,E,XI,XLN,AP, XMA,TA)	69
APPENDIX C.	SUBROUTINE TAN(R1,R2,T1,T2,I,XØ,XK,E, P,AP,XLN,XI,XMA).	73
APPENDIX D.	SUBROUTINE FGS(R1,R2,V1,T1,T2,E,A,AP, XLN,XI,XMA)	79
APPENDIX E.	SUBROUTINE UV(R1,R2,T1,T2,A,XMA,E,XI, XLN,AP)	81
APPENDIX F.	SUBROUTINE RV(A,E,XI,XLN,AP,XMO,T,R, TA,U)	85
APPENDIX G.	SUBROUTINE VV(A,E,XI,XLN,AP,TA,V)	87

CHAPTER PAGE

APPENDICES (Continued)

APPENDIX H. SUBROUTINE ET(PSI,THETA,PHI,A). 89

APPENDIX I. NOMINAL EARTH-MARS TRAJECTORIES 91

APPENDIX J. MINIMIZATION PROGRAM. 103

VITA. 117

LIST OF FIGURES

FIGURE		PAGE
1.1.	Shuttle Based Rescue	5
1.2.	The Buddy System	7
2.1.	Orbital Elements	15
2.2.	Sector and Triangle Area	25
2.3.	Radius Vectors	30
2.4.	Position Vectors	39
3.1.	Earth-Mars Transfer.	53
3.2.	Early Manned Mission, Mid-Transfer Turn Back	55
3.3.	Early Manned Mission, Mid-Transfer Abort to Mars.	56
3.4.	Advanced Manned Mission Mid-Transfer Abort to Station #1	58
3.5.	Advanced Manned Mission Mid-Transfer Abort to Station#2	59
3.6.	Freighter Mission Mid-Transfer Abort to Station Mission Length vs. ΔV.	61

CHAPTER I

INTRODUCTION

As exploration of the solar system becomes more complex and far reaching, the systems used for space missions are accompanied by new safety hazards. Many of these hazards can be eliminated by meticulous inspection during the conceptual and design phases. But there will always be hazards that cannot be eliminated, and some provision must be made for these.

There are many possibilities to consider when trying to plan for all of the possible en route hazards. The first of these is a system or subsystem malfunction or failure. They can usually be circumvented in some way because of the meticulous planning that went into all of these subsystems. Crew illness or injury is not too common, due to the stringent physical requirements and safety guidelines laid out. At least, during this sort of emergency the spacecraft is generally still in excellent condition and can be used to its full potential. The last group of hazards all fall together in that they all involve the disablement of the craft in some manner, as does the first hazard mentioned above. These hazards include fire, collisions, explosions, implosions, meteoroid penetration, and anything else that might reduce the capability of the vehicle.

When one is confronted with a hazard of some sort there are three general categories which one's response might fall into. First, there is the possibility of repair and continuation of the mission as planned previous to the incident. Escape from the craft accomplished solely by the crew, with available resources, is another possibility. The use of assistance or rescue from a source other than their own craft is the ultimate hazard response. This response can be broken down into more detailed categories such as combined efforts between rescue craft and crippled but still maneuverable craft or just surviving in the affected craft until help can arrive.

The majority of escape studies have dealt with the concept of the escape capsule. Two possible types are generally considered: first, the capsule that waits in an orbit for a rescue craft, and, second, the capsule with the capability of reentry. In addition to the aforementioned capabilities, provision has to be made for travel time, and in orbit, waiting time. All of these add up to a substantial weight penalty to the spacecraft and the problem of determining how much each requirement should influence the design of the capsule. Should the vehicle be able to return the crew from any point of this mission? Should the capsule be able to sustain the crew until another craft can reach the disabled vehicle? These are the questions that make the concept so complex. Another problem is the possibility of

the rescue capsule falling prey to the same difficulty as necessitated the rescue in the first place.

The use of pre-positioned rescue systems requires one of two things. Either the system is capable of reaching many different rendezvous points (i.e., has a high ΔV capability) or there are going to be a large number of vehicles pre-positioned. These both seem quite unattractive and would require a high budget. However, the possibility does exist of using a pre-positioned vehicle that is convenient for a particular mission sequence. Compromises would be necessary in any instance. Another decision to be made concerns the function of the rescue vehicle after rendezvous. Will the vehicle serve as a waiting platform for a rescue craft or will it return the crew to an Earth orbit or even to the surface?

Earth Launched Rescue, in the near future, will undoubtedly utilize the capabilities of a modified space shuttle. The shuttle can be used for any low-Earth-orbit rescue but beyond low Earth orbit is relatively useless. Modifications are possible, of course, and would probably involve the deployment of a rescue vehicle from the shuttle's cargo bay.

For interplanetary missions it is highly unlikely that an Earth based rescue would even be useful, except in the case of a very early, near-Earth, emergency. The major role for the shuttle system will be in supplement to another

rescue approach. The design of rescue systems becomes much easier if it can be assumed that a reliable Earth reentry system will be readily available once low Earth orbit is achieved.

A question often arises as to whether a manned or unmanned system would be better. It is true that unmanned systems require less preparation time to launch. But a stronger point is the need for human control to deal with unanticipated events. Also, a faster rendezvous, braking, and docking sequence can be accomplished since there would be no delay time for Earth-to-craft commands. The final thought on manned-versus-unmanned rescue concerns the possibility of the distressed crew needing assistance. It seems obvious that an unmanned vehicle would be of little help in such a situation.

In this problem, it will be assumed that a reliable system exists, namely the Space Shuttle, as in Figure 1.1, or a Space Station, in a low Earth orbit that could either return the crew to Earth or provide the necessary care in orbit.

For a near-Earth orbit of 100-500 nautical miles at inclinations from 20-90 degrees, rescue times would vary from 36 to 62 hours, plus docking time, for an Earth-based Shuttle launch.[1] Studies which looked at possible Space Station deployment of a rescue tug estimate rendezvous time

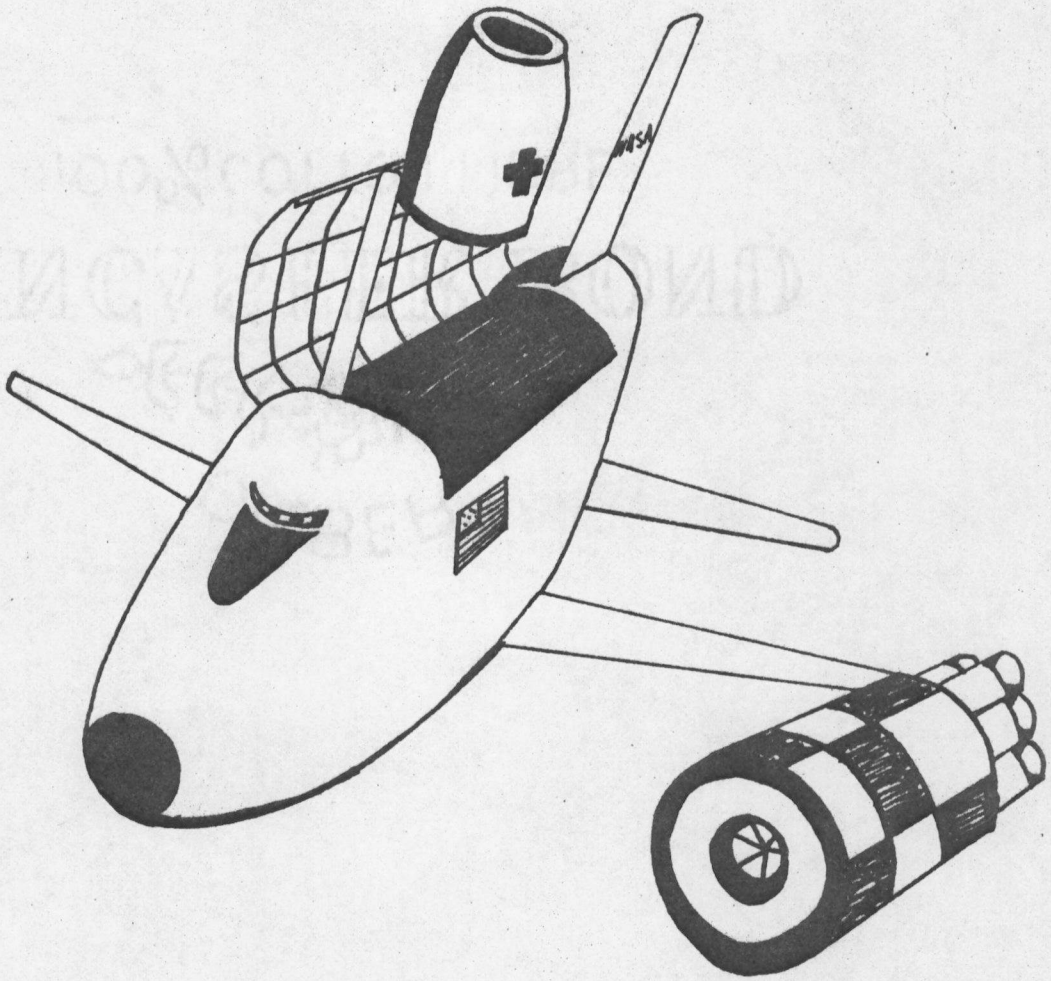


Figure 1.1. Shuttle Based Rescue.

of between 1 and 24 hours depending on the mission complexity.

Most studies which have examined long interplanetary missions have come up with a concept called the "buddy system" as the best safety precaution. In a "buddy system" mission, each spacecraft would have the capability of carrying the total complement of both spacecraft. The two craft would travel together to and from their destination with the capability of completion or abort if one of the craft were uninhabitable or disabled; see Figure 1.2. With missions going far beyond the range of an Earth-based rescue mission, this may be the only possible method for early manned missions.

The purpose of this study is to determine what possibilities are available for dealing with situations that might be encountered on an Earth-Mars mission.

Three basic classes of orbits will be examined: an orbit resembling what an early manned mission might use, a slower freighter-type mission, and an advanced manned mission. The early manned mission and the freighter mission will undoubtedly use elliptical transfer orbits. The advanced manned mission will use a hyperbolic orbit with a transfer time of 40-60 days.

In the past, not much has been done concerning manned missions that would last over two weeks. Most work in examining abort situations has been done in correlation with

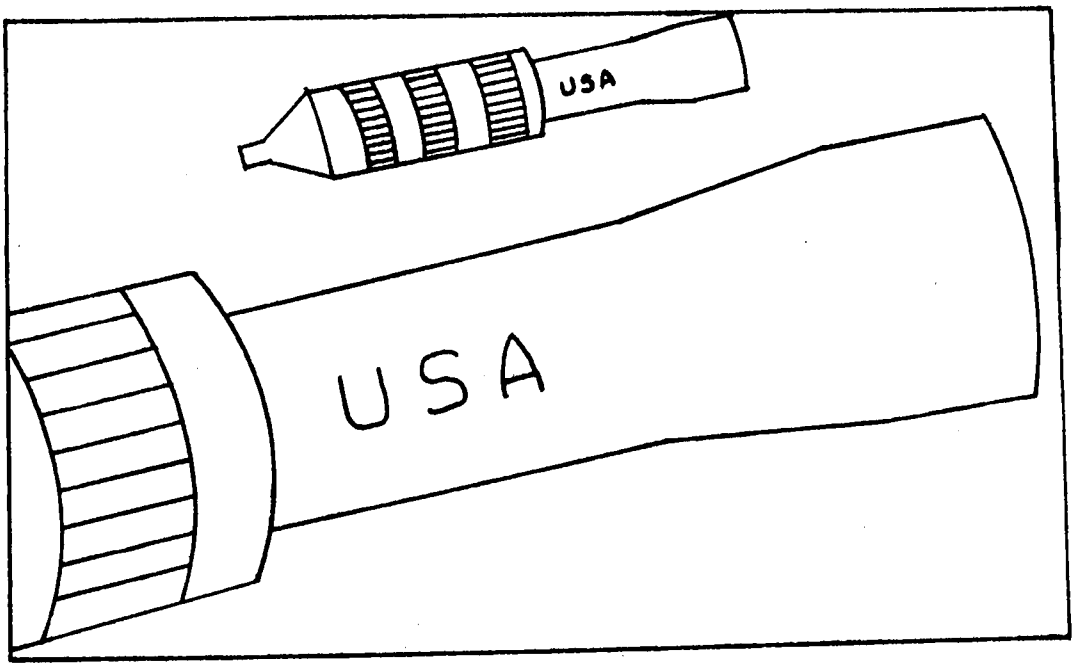


Figure 1.2. The Buddy System.

the Apollo program and thus has not really touched the problems that arise in a longer mission. During the Apollo studies, almost all work followed the theme of maintaining the original orbit through to its completion, or sending an Earth-based rescue craft. An Earth-based rescue craft is a very logical solution, but only for near-Earth missions. Besides range problems, it seems highly improbable that a full-scale Earth-based rescue system would ever be funded.

By the time of frequent Mars missions, there will very likely be space colonies or stations established which would provide targets for abort rendezvous. With the possibility of actually mining the asteroid belts and having people spend lifetimes on space-colonies, it is highly probable that stations will exist between the Earth and Mars. Thus, this study will assume the availability of such stations.

There exists the possibility of having two craft travel the same orbit to Mars providing the possibility of crew transfer in case of failure in one craft. A substantial cost is involved in providing the necessary capabilities for two such sister craft. Both craft would be required to have facilities for carrying twice the normal crew plus the extra life support apparatus and supplies. Because of these difficulties and because this concept is an area of study on its own, it will not be examined in any depth.

Preliminary work is directed at creating a computer program that would determine the necessary orbital elements for an Earth-Mars trajectory. The resultant program uses the True Anomaly Iteration technique to determine the necessary orbit, given the time of departure from Earth and the time of arrival at Mars. The program is easily modified to compute the orbit passing through any two given points at given times, and will, therefore, also show up in the determination of new orbits for the abort situation simulated or created.

The optimization program minimizes the ΔV necessary for rendezvous with a rescue base of some sort. Given the day of the emergency, a minimum - ΔV transfer is determined between the nominal transfer orbit and the chosen target.

It should be clear that a total loss of either ΔV or life support would leave no options to be considered. It will be necessary to establish hypothetical orbits for the space stations in order to make calculations.

It may seem premature to consider manned missions to Mars since none are currently in the planning at NASA. Since the Viking Missions there has been a decided lessening of interest in Mars. At present, the Space Shuttle Program is in the spotlight. However, when the effectiveness of the shuttle has been proved and space travel becomes more routine, the establishing of space stations and extension of space exploration will be much more likely.

Considering the possibility of an early manned mission with limited energy to expend, we can expect a mission length of 150-200 days. There is a strong possibility of sending two sister craft on the mission for safety reasons. Another important consideration is that there will probably not be a station positioned at a point where it would be of much benefit in an abort situation. Also, the possibility of making it to the destination and having anything there waiting is remote. Chances are that there will be very limited rescue options for the early manned flights.

With the possible establishing of bases, of some sort, on Mars for manufacturing or colonization, advanced manned missions and freighter missions become probable. The freighter missions will use low energy orbits and may be up to 250 days in length. Freighter missions will very likely be unmanned, but the chance exists that a crew would be on board, depending on the cargo and the length of the mission.

The last mission considered would use a high-energy orbit. The mission length would be on the order of 40-60 days and hopefully the missions would be fairly frequent. For these missions the abort options would be much more plentiful. At the time of such missions the space base on Mars and also a station positioned midway between Earth and Mars are more likely than for previous missions. For these

missions, this analysis will have the most relevant application.

For the problem being confronted here, it is necessary to have a means of determining the orbit between two points with two given times. The problem is generally called the Gauss problem, since Gauss devised a solution for it in 1802. There are quite a number of methods that have been used to solve this problem; all of them have their good and bad points. The method used here utilizes the true anomaly iteration technique. This method was chosen for its relatively good convergence characteristics.

It is obvious that the first thing necessary for the solution of this problem is to have the positions of the two points wanted in the orbit. For this problem, dealing first with an Earth to Mars orbit, the orbital characteristics are established for a base day. From this base day and these original positions of Earth and Mars, the new positions can easily be determined for any other day wanted. With these elements determined, the transfer orbit can be determined.

After convergence with the true anomaly iteration, the orbital elements for the transfer orbit can be determined. The necessary velocities for the transfer orbit, at the departure and arrival points, can be found and are used to determine the necessary changes in velocity at departure and arrival (ΔV's).

This program can also be used for determining any necessary rendezvous orbits by a simple modification. The only necessary modification is to change the departure and arrival information. The new elements and necessary ΔV's will then be given for any necessary modifications.

Following are the derivations of several of the most common solutions to the two-radius-vectors-and-time-of-transfer problem. Also included is a simple derivation of the Newton-Raphson Extremization method, to be used in finding the minimum $-\Delta V$ transfer.

Numerical results follow the derivation section and the appendices include all of the subroutines needed for the calculations.

CHAPTER II

DERIVATION OF ALGORITHMS

1. Introduction

The derivations that follow are modifications of derivations found in texts by Bates[2] and Escobal[3]. The derivations are of a general nature and any differences between those following and other derivations of the same sort are minor.

The fundamental building block, for optimizing a given rendezvous problem, is a method for determining an orbit from two radius vectors and the time of transfer between them. After having derived an algorithm for solving this problem, a relatively simple approach can be taken to the optimization problem.

Although there are a variety of numerical methods available for solving the two-radius-vector problem, all have their good and poor qualities. One of the fastest and simplest methods is the Gauss method, derived in the early 19th century by Carl Friedrich Gauss. He was challenged by the problem of determining the orbit of Ceres. His largest problem came from the small amount of data available, since Ceres was hidden by the Sun after about one month of observation. The Gauss Method, however, has convergence problems when the transfer angle becomes greater than 90^{o}

and thus is not applicable for all the cases necessary for the problem here. The p-iteration method is relatively fast in convergence and simple to use; however, the method reaches a critical point as the transfer angle reaches 180° and a division by zero will occur in the process. It is easily seen that all of the necessary orientation elements for an orbit are not going to be found for the case where the transfer angle is exactly 180°, since the radii will be colinear. (An orbit can be completely described by six orbital elements. Three of the elements determine the in-plane conic section and the remaining three situate the plane in the heliocentric coordinate system. Although there are many choices for orbital elements, the eccentricity, semi-major axis, and true anomaly (for a given time) will be used for the in-plane variables. For the remaining elements, the inclination, argument of pericenter, and longitude of the ascending node will be used; see Figure 2.1.) However, some information about the orbit can be determined and a method that works for all transfer angles is called the True Anomaly Iteration method. More important than determining the actual elements at a transfer of 180° is avoiding a division by zero in the computational algorithm. If the time for transfer is not too large, the orbit determination problem can be solved by using the f and g series expressions. The method has rapid convergence and is limited only by the magnitude of the time interval. A

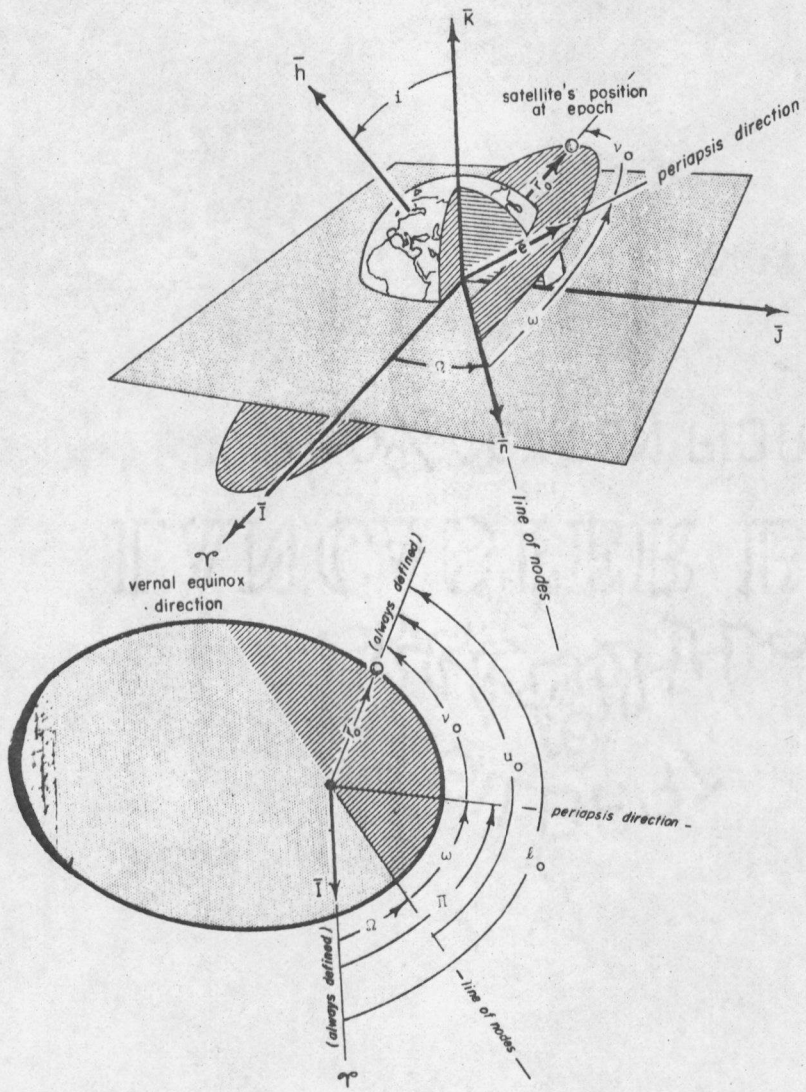


Figure 2.1. Orbital Elements.

simplified scheme that is good for all conics and all transfer angles, and is not limited by the time interval, is the Universal Variable method. In this method a series expansion is used to estimate the derivatives in the function, initially, avoiding the problem of evaluating the derivatives when the universal variable, z, is near zero.

In this chapter the aforementioned methods will be derived. As has been pointed out, all of the methods have their good and bad points which will be mentioned again during the development of each. Separate subroutines for each method are included in the appendices along with user instructions.

When a satisfactory method has been chosen for determining a solution to the two-radius-vector problem, it is relatively easy to determine the velocity change (ΔV) necessary for making a transfer between orbits. This determination is of greatest interest to the problem confronted in this thesis and will be used in computing the most effective transfer orbit for given abort situations.

A simple method for determining an extremum in a function is the Newton-Raphson method. A derivation of this method concludes this chapter and once again a subroutine, with user instructions, is supplied in the appendices.

The first problem considered will be that of determining an orbit, given the radius and velocity vectors for a given time. Once this derivation is complete, it

simplifies the other derivations since they will have to be carried out only through the determination of radius and velocity vectors.

2. Orbit Determination

Given a radius vector and velocity vector for any given time, it is possible to determine all of the elements of the orbit in question.

By first determining the angular momentum vector and the node vector, from the radius vector and velocity vector, it is possible to determine all of the orbital elements by simple relationships.

The angular momentum vector and node vector, $\underline{h}$ and $\underline{n}$, are shown in Figure 2.1 and can be determined from

$$\underline{h} = \underline{r} \times \underline{v} \tag{2.2-1}$$

and

$$\underline{n} = \underline{k} \times \underline{h} \tag{2.2-2}$$

or

$$\underline{h} = \begin{vmatrix} I & J & K \\ r_x & r_y & r_z \\ v_x & v_y & v_z \end{vmatrix} = h_x\, I + h_y\, J + h_z\, K \tag{2.2-3}$$

and

$$\underline{n} = \begin{vmatrix} I & J & K \\ 0 & 0 & 1 \\ h_x & h_y & h_z \end{vmatrix} = n_x\, I + n_y\, J + n_z\, K \qquad (2.2\text{-}4)$$

By definition $\underline{n}$ must be perpendicular to both $\underline{k}$ and $\underline{h}$. To be perpendicular to $\underline{k}$ it must lie in the x-y plane and to be perpendicular to $\underline{h}$ must lie in the orbital plane. Thus the node vector must lie at the intersection of the ecliptic and orbital planes, i.e., the line of nodes as shown in Figure 2.1.

The semi-latus rectum, p, can be determined from the angular momentum vector, $\underline{h}$, and the gravitational constant, μ, as

$$p = \frac{h^2}{\mu} \qquad (2.2\text{-}5)$$

The eccentricity, e, may be determined from the radius vector, $\underline{r}$, the velocity vector, $\underline{v}$, and μ, by

$$e = (1 + (RV_T{}^2/\mu)(RV^2/\mu - 2))^{1/2} \qquad (2.2\text{-}6)$$

where

$$V_T = (V^2 - V_R{}^2)^{1/2} \qquad (2.2\text{-}7)$$

and

$$V_R = \underline{R} \cdot \underline{V}/|R| \qquad (2.2\text{-}8)$$

The inclination, i, being the angle between $\underline{h}$ and h_k can be determined from

$$i = \cos^{-1}(h/h) \tag{2.2-9}$$

The longitude of the ascending node, Ω, is the angle between the node vector, $\underline{n}$, and the x-axis, in the ecliptic, thus

$$\Omega = \cos^{-1}\left(\frac{n_x}{n}\right) \tag{2.2-10}$$

The argument of pericenter, ω, can be found as the angle between the node vector, $\underline{n}$, and the eccentricity vector, $\underline{e}$, which can be found from

$$\underline{e} = \left(\left(V^2 - \frac{\mu}{r}\right)(\underline{r}) - (\underline{r} \cdot \underline{v})(v)\right)/\mu \tag{2.2-11}$$

so,

$$\omega = \cos^{-1}(\underline{n} \cdot \underline{e}/ne) \tag{2.2-12}$$

The true anomaly, f, can be found, in a similar manner, as

$$f = \cos^{-1}\frac{\underline{e} \cdot r}{er} \tag{2.2-13}$$

The physical relationships for ω and f can be seen in Figure 2.1.

The final element, the semi-major axis, a, can be determined once the eccentric anomaly, E, or the hyperbolic anomaly, F, is determined from

$$E = 2 \tan^{-1} \left(\left(\frac{1 - e}{1 + e} \right)^{1/2} \tan f/2 \right) \qquad (2.2\text{-}14)$$

or

$$F = 2 \tanh^{-1} \left(\left(\frac{e - 1}{e + 1} \right)^{1/2} \tan f/2 \right) \qquad (2.2\text{-}15)$$

as

$$a = E - e \sin E \qquad (2.2\text{-}16)$$

or

$$a = e \sinh F - F \qquad (2.2\text{-}17)$$

With a, p, e, i, Ω, ω, and f the orbit is completely determined. (Actually, one of the variables a, p, e, is redundant; one customarily uses either a or p but not both.)

3. Gauss Method

Before proceeding with the derivation of the Gauss Method, a derivation of the f and g expressions would greatly simplify matters.

There are two common problems encountered when trying to solve Kepler's problem, for eccentricities near one. The first problem is diminishing numerical accuracy and the

second is a decreasing rate of convergence for the solution. By introducing a new variable, x, called the "Universal Variable," it is possible to avoid these two problems. This transformation was originally proposed in the early Twentieth Century and is known as the "Sundman Transformation."

The Universal Variable, x, is defined by

$$\dot{x} = \frac{\sqrt{\mu}}{r} \tag{2.3-1}$$

The so-called energy integral can be written as

$$\frac{1}{2}V_R^2 + \frac{1}{2}V_T^2 - \frac{\mu}{r} = -\frac{\mu}{2a} \tag{2.3-2}$$

or

$$\frac{1}{2}\dot{r}^2 + \frac{1}{2}(r\dot{f})^2 - \frac{\mu}{r} = -\frac{\mu}{2a} \tag{2.3-3}$$

where V_r and V_t are the radial and transverse components of $\underline{V}$. Solving for $\dot{r}$, and using the angular momentum equation

$$h = r^2\dot{f} = (\mu p)^{1/2} \tag{2.3-4}$$

$\dot{r}$ is determined as

$$\dot{r} = \left(-\frac{\mu p}{r^2} + \frac{2\mu}{r} - \frac{\mu}{a}\right)^{1/2} \tag{2.3-5}$$

Now dividing (2.3-5) by (2.3-1) it is seen that

$$\frac{\dot{r}^2}{\dot{x}^2} = \frac{(-\frac{\mu p}{r^2} + \frac{2\mu}{r} - \frac{\mu}{a})r^2}{\mu} = -p + 2r - \frac{r^2}{a} \qquad (2.3\text{-}6)$$

and it follows that

$$\dot{x} = \frac{\dot{r}}{(-p + 2r - \frac{r^2}{a})^{1/2}} \qquad (2.3\text{-}7)$$

To solve for x, (2.3-7) must be integrated, yielding

$$x + c_0 = \sqrt{a}\ \sin^{-1} \frac{(\frac{r}{a} - 1)}{(1 - \frac{p}{a})^{1/2}} \qquad (2.3\text{-}8)$$

where c_0 is the constant of integration.

The radius, r, can be determined, as a function of x, by utilizing the relation

$$e = \sqrt{1 - \frac{p}{a}} \qquad (2.3\text{-}9)$$

and substituting (2.3-9) into (2.3-8) to get

$$x + c_0 = \sqrt{a}\ \sin^{-1} \frac{(\frac{r}{a} - 1)}{e} \qquad (2.3\text{-}10)$$

and solving for r yields

$$r = a(1 + e \sin \frac{x + c_0}{\sqrt{a}}) \qquad (2.3\text{-}11)$$

From (2.3-1) r can be written as

$$r = \frac{\sqrt{\mu}}{\dot{x}} \tag{2.3-12}$$

so, from (2.3-11) and (2.3-12)

$$\sqrt{\mu}\, dt = a\left(1 + e \sin \frac{x + c_0}{\sqrt{a}}\right) dx \tag{2.3-13}$$

and integrating (2.3-13) yields

$$t = \left[ax - ae\sqrt{a}\left(\cos \frac{x + c_0}{\sqrt{a}} - \cos \frac{c_0}{\sqrt{a}}\right)\right]/\sqrt{\mu} \tag{2.3-14}$$

Without going through the entire derivation, it is known that a vector, being coplanar with two more, non-colinear, vectors, can be expressed as a linear combination of the two. Using the Universal Variable, x, it is possible to define variables f, g, $\dot{f}$, and $\dot{g}$, such that $\underline{r}$ and $\underline{v}$, for an orbit, can be determined in terms of $\underline{r}_0$ and $\underline{v}_0$. The relations necessary are

$$\underline{r} = f\,\underline{r}_0 + g\,\underline{v}_0 \tag{2.3-15}$$

and

$$\underline{v} = \dot{f}\,\underline{r}_0 + \dot{g}\,\underline{v}_0 \tag{2.3-16}$$

where

$$f = 1 - \frac{a}{r_0}\left(1 - \cos\frac{x}{\sqrt{a}}\right) \tag{2.3-17}$$

$$g = t - \frac{x^3}{\sqrt{\mu}} S \tag{2.3-18}$$

$$\dot{f} = \frac{\sqrt{\mu}}{r_0 r} x(zS-1) \tag{2.3-19}$$

$$\dot{g} = 1 = \frac{x^2}{r} C \tag{2.3-20}$$

and where

$$z = x^2/a \tag{2.3-21}$$

and

$$C = \sum_{K=0}^{\infty} \frac{(-z)^K}{(2K + 2)!} \tag{2.3-22}$$

while

$$S = \sum_{K=0}^{\infty} \frac{(-z)^K}{(2K + 3)!} \tag{2.3-23}$$

With an understanding of the Universal Variable formulation and the f and g expressions, the remaining derivations will be greatly simplified.

The "Gauss Method" reduces the two-radius vectors-and-transfer-time problem from three equations and three

unknowns to two equations and two unknowns by adding another independent equation and another unknown, y. Two of the three original unknowns can be eliminated from the four equations. With the two remaining, transcendental equations a solution can be found by trial and error.

As shown in Figure 2.2, the ratio of the sector area between two radius vectors and the triangular area between the same two radius vectors, can be called y and can be written as

$$y = \sqrt{\mu p}\, t/r_1 r_2 \sin\Delta f \qquad (2.3\text{-}24)$$

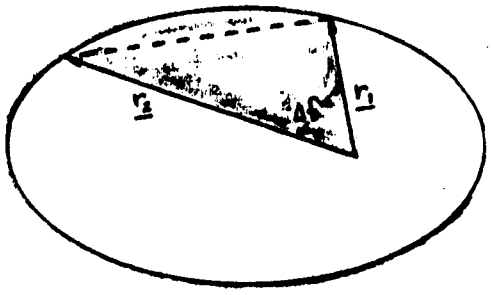


Figure 2.2. Sector and Triangle Area.

The semi-latus rectum, p, can be written as

$$p = \frac{r_1 r_2(1 - \cos\Delta f)}{r_1 + r_2 - 2\sqrt{r_1 r_2}\cos\frac{\Delta f}{2}\cos\frac{\Delta E}{2}} \qquad (2.3\text{-}25)$$

Using (2.3-25) and substituting it into (2.3-24) shows that

$$y^2 = \frac{\mu r_1 r_2 (1-\cos\Delta f) t^2}{(r_1 r_2 \sin\Delta f)^2 \left(r_1+r_2-2\sqrt{r_1 r_2}\cos\frac{\Delta f}{2}\cos\frac{\Delta E}{2}\right)} \tag{2.3-26}$$

(2.3-26) can be further reduced by using the trigonometric identity, $\sin^2 x/(1-\cos x) = 2\cos^2 x$, so that

$$y^2 = \frac{\mu t^2}{2r_1 r_2 \cos^2\frac{\Delta f}{2}\left(r_1+r_2-2\sqrt{r_1 r_2}\cos\frac{\Delta f}{2}\cos\frac{\Delta E}{2}\right)} \tag{2.3-27}$$

The introduction of two constants, S and W, can substantially reduce the complexity of (2.3-27) if

$$S = \frac{r_1 + r_2}{4\sqrt{r_1 r_2}\cos\frac{\Delta f}{2}} - \frac{1}{2} \tag{2.3-28}$$

and

$$W = \frac{\mu t^2}{\left(2\sqrt{r_1 r_2}\cos\frac{\Delta f}{2}\right)^3} \tag{2.3-29}$$

Making use of (2.3-28) and (2.3-29), (2.3-27) can be rewritten as

$$y = \left[\frac{W}{S + 1/2(1-\cos\frac{\Delta E}{2})}\right]^{1/2} \tag{2.3-30}$$

(2.3-30) is known as the "first equation of Gauss."

Another equation for determining y can be derived by using the relation

$$\frac{r_1 r_2 \sin\Delta f}{\sqrt{\mu p}} = t - \sqrt{\frac{a^3}{\mu}} (\Delta E - \sin\Delta E) \qquad (2.3\text{-}31)$$

and (2.2-24) to yield

$$1 - \frac{1}{y} = \frac{1}{t}\sqrt{\frac{a^3}{\mu}} (\Delta E - \sin\Delta E) \qquad (2.3\text{-}32)$$

(2.3.32) can be further reduced using the trigonometric identity, $\sin x - 2 \sin\frac{x}{2} \cos\frac{x}{2}$, to give

$$y = \frac{\sqrt{\mu p}\ t}{2 r_1 r_2 \sin\frac{\Delta f}{2} \cos\frac{\Delta f}{2}} \qquad (2.3\text{-}33)$$

or since,

$$f = 1 - \frac{r_2}{p} (1 - \cos\Delta f) = 1 - \frac{a}{r_1} (1 - \cos\Delta E) \qquad (2.3\text{-}34)$$

and following from (2.3-34)

$$1 - \cos\Delta f = \frac{ap}{r_1 r_2} (1 - \cos\Delta E) \qquad (2.3\text{-}35)$$

or

$$2 \sin^2 \frac{\Delta f}{2} = 2 \frac{ap}{r_1 r_2} \sin^2 \frac{\Delta E}{2} \qquad (2.3\text{-}36)$$

so that

$$\sin \frac{\Delta f}{2} = \sqrt{\frac{ap}{r_1 r_2}} \sin \frac{\Delta E}{2} \tag{2.3-37}$$

then (2.3-37) can be substituted into (2.3-33) to give

$$y = \frac{\sqrt{\mu}\, t}{2\sqrt{a r_1 r_2} \sin \frac{\Delta E}{2} \cos \frac{\Delta f}{2}} \tag{2.3-38}$$

In this form the semi-latus rectum, p, has been completely eliminated. To eliminate the semi-major axis, a, (2.3-32) and (2.3-38) can be combined to yield

$$y^3\left(1 - \frac{1}{y}\right) = \frac{\mu t^2}{\left(2\sqrt{r_1 r_2} \cos \frac{\Delta f}{2}\right)^3} \frac{\Delta E - \sin \Delta E}{\sin^3 \frac{\Delta E}{2}} \tag{2.3-39}$$

Making use of W, as defined in (2.3-29) and rearranging the first term leaves

$$y^2(y-1) = \frac{W(\Delta E - \sin \Delta E)}{\sin^3 \frac{\Delta E}{2}} \tag{2.3-40}$$

(2.3-40) can be combined with "the first equation of Gauss," (2.3-30), and then

$$y = 1 + \left(\frac{\Delta E - \sin \Delta E}{\sin^3 \frac{\Delta E}{2}}\right)\left(S + \frac{1 - \cos \frac{\Delta E}{2}}{2}\right) \tag{2.3-41}$$

which is known as "the second equation of Gauss."

Using the two Gaussian equations (2.3-30) and (2.3-41), an initial estimate for y is necessary for evaluation of the first equation to determine ΔE. With ΔE the second equation can be evaluated to give a new value for y. This new value of y can be put into the first equation once again to give a better value of ΔE. This process can be continued until the difference in values for either y or ΔE with consecutive iterations falls within the desired error tolerance.

With the solution for the two Gauss equations determined, the semi-latus rectum, p, can be found from (2.3-25) and then the f and g expressions can be evaluated. At this point the problem is solved, since the orbit determination process, using $\underline{r}$ and $\underline{v}$, can complete the process.

4. Semi-Latus Rectum Iteration Method

Given radius vectors $\underline{r}_1$ and $\underline{r}_2$, as shown in Figure 2.3, two unit vectors can be computed as

$$\mu_1 = \frac{\underline{r}_1}{r_1} \tag{2.4-1}$$

$$\mu_2 = \frac{\underline{r}_2}{r_2} \tag{2.4-2}$$

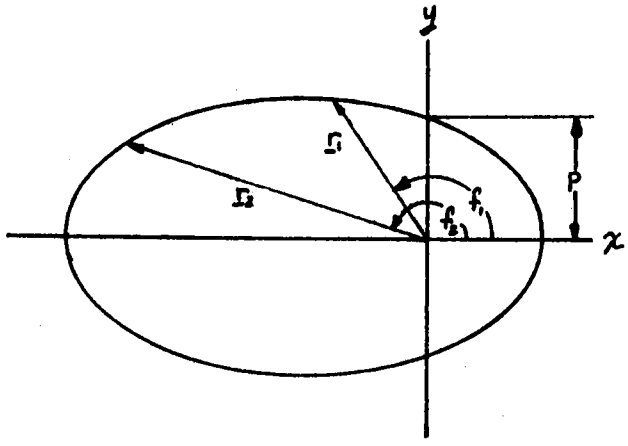


Figure 2.3. Radius Vectors.

Knowing these unit vectors, the angle between them can be easily computed as

$$f_2 - f_1 = \cos^{-1} (u_1 \cdot u_2) \qquad (2.4\text{-}3)$$

$$f_2 - f_1 = \sin^{-1} (u_1 \times u_2) \qquad (2.4\text{-}4)$$

Now, assuming it is known whether the orbit is direct or retrograde, the angle $f_2 - f_1$ is completely specified.

If the value of the semi-latus rectum, p, is guessed it is possible to determine the eccentricity once the values for f_1 and f_2 have been found.

Starting with the simple relation

$$\cos f_2 - \cos f_2 = 0 \qquad (2.4\text{-}5)$$

and using the trigonometric identity, $\sin^2 x + \cos^2 x = 1$, yields

$$(\sin^2 f_1 + \cos^2 f_1)\cos f_2 - \cos f_2 = 0 \qquad (2.4\text{-}6)$$

and expanding (2.4-6)

$$\sin^2 f_1 \cos f_2 + \cos^2 f_1 \cos f_2 - \cos f_2 = 0 \qquad (2.4\text{-}7)$$

Now, by adding $\sin f_1 \sin f_2 \cos f_1$ to both sides leaves

$$\sin f_1 \sin f_2 \cos f_1 - \sin^2 f_1 \cos f_2 = \cos^2 f_1 \cos f_2 + \sin f_1 \sin f_2 \cos f_1 - \cos f_2 \qquad (2.4\text{-}8)$$

Rearranging (2.4-8) leaves

$$\sin f_1(\sin f_2 \cos f_1 - \cos f_2 \sin f_1) = \cos f_1(\cos f_2 \cos f_1 + \sin f_2 \sin f_1) - \cos f_2 \qquad (2.4\text{-}9)$$

By using the trigonometric identities, $\sin(b-a) = \sin b \cos a - \cos b \sin a$, and $\cos(b-a) = \cos a \cos b + \sin a \sin b$, (2.4-9) reduces to

$$\sin f_1 \sin(f_2 - f_1) = \cos f_1 \cos(f_2 - f_1) - \cos f_2 \qquad (2.4\text{-}10)$$

Rearranging (2.4-10) and multiplying through by the eccentricity, e, yields

$$e \sin f_1 = \frac{e \cos f_1 \cos(f_2-f_1) - e \cos f_2}{\sin(f_2-f_1)} \qquad (2.4\text{-}11)$$

and similarly,

$$e \sin f_2 = \frac{- e \cos f_2 \cos(f_2-f_1) + e \cos f_1}{\sin(f_2-f_1)} \qquad (2.4\text{-}12)$$

Having guessed a value for the semi-latus rectum, and using the equation of a conic, gives

$$e \cos f_i = \frac{p}{r_i} - 1 \quad , \quad i = 1,2 \qquad (2.4\text{-}13)$$

Now, all of the values needed for determining f_1 and f_2 from (2.4-11) and (2.4-12) are available. The eccentricity can now be determined from

$$e = ((e \cos f_i)^2 + (e \sin f_i)^2)^{1/2} \quad , \quad i = 1,2 \qquad (2.4\text{-}14)$$

and the semi-major axis, p, may now be determined from

$$a = p/(1-e^2) \qquad (2.4\text{-}15)$$

Since the first guess for the semi-latus rectum will, most likely, be incorrect, it is necessary to compute a better estimate for p. The idea used to compute the new value is based on the difference between the known transfer time, between $\underline{r}_1$ and $\underline{r}_2$, and time determined from the guessed value of p.

It is first necessary to determine the mean anomalies, M_1 and M_2, at the two times from

$$M_i = E_i - e \sin E_i \quad , \quad i = 1,2 \tag{2.4-16}$$

and the eccentric anomalies are uniquely determined from

$$\sin E_i = \frac{\sqrt{1-e^2}\ \sin f_i}{1 + e \cos f_i} \quad , \quad i = 1,2 \tag{2.4-17}$$

and

$$\cos E_i = \frac{\cos f_i + e}{1 + e \cos f_i} \quad , \quad i = 1,2 \tag{2.4-18}$$

The mean motion, n, can be defined as

$$n = (K/a^3)^{1/2} \tag{2.4-19}$$

where

$$K = G(M_1+M_2) \tag{2.4-20}$$

The time for transfer can be found from

$$t_2 - t_1 = (M_2-M_1)/n \tag{2.4-21}$$

Since $t_2 - t_1$ is known, the difference between the computed time and the actual time is

$$dt = (t_2-t_1) - (M_2-M_1)/n \tag{2.4-22}$$

Once dt has been determined for the first estimate of p one should increment p by a small amount, Δp. The same process as originally used should be repeated for $p + \Delta p$ and $p - \Delta p$. It is now possible to determine a numerical derivative to be used in finding the next estimate for p. Using a central difference method

$$dt' = \frac{dt(p+\Delta p) - dt(p-\Delta p)}{2\Delta p} \qquad (2.4\text{-}23)$$

Now, dp can be defined as

$$dp = - \frac{dt(p_i)}{dt'(p_i)} \quad , \quad i = 1,2,\ldots, m \qquad (2.4\text{-}24)$$

where m is the iteration step. The i + 1 estimate for p is given by

$$p_{i+1} = p_i + dp \qquad (2.4\text{-}25)$$

The process is continued, through m steps, until

$$|dp| < \varepsilon \qquad (2.4\text{-}26)$$

Once the iterative procedure has converged, the f and g series can be evaluated, as outlined in Section II.3. The orbital elements can then be determined as detailed in Section II.2.

5. True Anomaly Iteration Method

One simplified method for determining a solution to the two radii and time problem is by iterating using the true anomaly, f.

It is, first of all, possible to determine a value for the transfer angle, $f_2 - f_1$, by first determining two unit vectors, U_1 and U_2, from

$$U_1 = \frac{r_1}{r_1} \tag{2.5-1}$$

and

$$U_2 = \frac{r_2}{r_2} \tag{2.5-2}$$

From (2.5-1) and (2.5-2), $f_2 - f_1$ can be determined as

$$f_2 - f_1 = \cos^{-1} (U_1 \cdot U_2) \tag{2.5-3}$$

and

$$f_2 - f_1 = \sin^{-1} (U_1 \times U_2) \tag{2.5-4}$$

Knowing $f_2 - f_1$ and guessing an initial value for f_1,

$$f_2 = f_1 + (f_2 - f_1) \tag{2.5-5}$$

Since,

$$r = \frac{p}{1 + e \cos f} \tag{2.5-6}$$

it follows that

$$r_1(1 + e \cos f_1) = r_2(1 + e \cos f_2) \tag{2.5-7}$$

From (2.5-7),

$$r_1 - r_2 = e(r_2 \cos f_2 - r_1 \cos f_1) \tag{2.5-8}$$

and the eccentricity can be found from (2.5-8) as

$$e = \frac{r_1 - r_2}{r_2 \cos f_2 - r_1 \cos f_1} \tag{2.5-9}$$

The semi-major axis, a, can be determined from

$$a = \frac{r(1 + e \cos f)}{(1-e^2)} \tag{2.5-10}$$

and then the semi-latus rectum follows directly from

$$p = a(1-e^2) \tag{2.5-11}$$

To avoid being redundant in deriving a method for determining the new estimate for f, it is only necessary to go through the procedure previously mentioned in Section II.4. Equations (2.4-15) through (2.4-21) should be used in the same form. Once dt has been determined for the first

estimate of f, a new estimate for f should be evaluated, at f + Δf and f - Δf. This process should yield another two values of dt to be used for determining

$$dt' = \frac{dt(f+\Delta f) - dt(f-\Delta f)}{2\Delta f} \qquad (2.5\text{-}12)$$

df can now be defined as

$$df = \frac{dt(f_i)}{dt'(f_i)} \quad , \quad i = 1,2,\ldots, m \qquad (2.5\text{-}13)$$

Now, the next estimate for the true anomaly is

$$f_{i+1} = f_i + df \qquad (2.5\text{-}14)$$

The iterations are continued until

$$|df| < \varepsilon \qquad (2.5\text{-}15)$$

where ε is the specified error tolerance for the true anomaly.

With a solution determined, the f and g series can be evaluated, as in Section II.3, and then the orbital elements can be determined from the procedure outlined in Section II.2.

6. F and G Series

Although this method has many similarities to the f and g relationships mentioned in Section II.2, it should be realized that this section deals with solving the Gauss

problem and not just the problem of determining orbital elements from a radius and velocity vector.

A solution for the vector equation

$$\ddot{\underline{r}} = \frac{-\mu \underline{r}}{r^3} \tag{2.6-1}$$

can be found by time series expansions. This method of orbit determination is of particular interest in problems where the solution might be elliptic, parabolic, or hyperbolic since it converges for all cases given a transfer angle less than $\pi/2$.

Given the two position vectors, as shown in Figure 2.4, and expanding from a radius vector halfway between $\underline{r}_1$ and $\underline{r}_2$ yields

$$\underline{r}_1 = \underline{r}_0 + \tau_1 \dot{\underline{r}}_0 + \frac{1}{2} \tau_1^2 \ddot{\underline{r}}_0 + \ldots \tag{2.6-2}$$

$$\underline{r}_2 = \underline{r}_0 + \tau_2 \dot{\underline{r}}_0 + \frac{1}{2} \tau_2^2 \ddot{\underline{r}}_0 + \ldots \tag{2.6-3}$$

and

$$\tau_1 = K(t_1 - t_0) \tag{2.6-4}$$

$$\tau_2 = K(t_2 - t_0) \tag{2.6-5}$$

where

$$t_o = \frac{t_2 + t_1}{2} \tag{2.6-6}$$

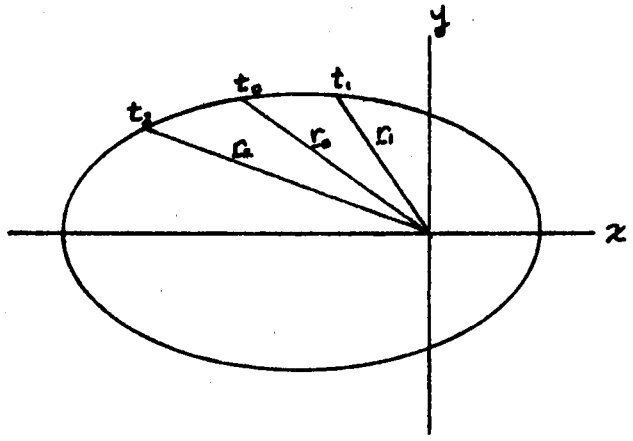


Figure 2.4. Position Vectors.

Now applying (2.6-1) as a dynamical constraint at t_0,

$$\ddot{\underline{r}}_0 = - \frac{\mu \, \underline{r}_0}{r_0^3} \tag{2.6-7}$$

into (2.6-2) and (2.6-3), yields

$$\underline{r}_1 = \underline{r}_0 \left(1 - \frac{\mu \, \tau_1^2}{2r_0^3}\right) + \tau_1 \, \dot{\underline{r}}_0 + \ldots \tag{2.6-8}$$

$$\underline{r}_2 = \underline{r}_0 \left(1 - \frac{\mu \, \tau_2^2}{2r_0^3}\right) + \tau_2 \, \dot{\underline{r}}_0 + \ldots \tag{2.6-9}$$

These two series expansions can be truncated and simplified to

$$\underline{r}_1 = A\,\underline{r}_0 + \tau_1\,\dot{\underline{r}}_0 \tag{2.6-10}$$

$$\underline{r}_2 = B\,\underline{r}_0 + \tau_2\,\dot{\underline{r}}_0 \tag{2.6-11}$$

where

$$A = \left(1 - \frac{\mu\,\tau_1^{\,2}}{2\,r_0^{\,3}}\right) \tag{2.6-12}$$

$$B = \left(1 - \frac{\mu\,\tau_2^{\,2}}{2\,r_0^{\,3}}\right) \tag{2.6-13}$$

By eliminating $\dot{\underline{r}}_0$ from (2.6-10) and (2.6-11) and solving for $\underline{r}_0$ as follows:

$$\frac{\underline{r}_1 - \underline{r}_0 A}{\tau_1} = \frac{\underline{r}_2 - \underline{r}_0 B}{\tau_2} = \dot{\underline{r}}_0 \tag{2.6-14}$$

then,

$$\underline{r}_0 = \frac{\tau_1\underline{r}_2 - \tau_2\underline{r}_1}{\tau_1 B - \tau_2 A} = \left(\frac{\tau_2}{A\tau_2 - B\tau_1}\right)\underline{r}_1 - \left(\frac{\tau_1}{A\tau_2 - B\tau_1}\right)\underline{r}_2 \tag{2.6-15}$$

Using the aforementioned approximation for r_0 from

$$r_0 = \frac{r_1 + r_2}{2} \tag{2.6-16}$$

A and B can be determined from (2.6-12) and (2.6-13). Therefore the starting value of $\underline{r}_0$ is known.

By eliminating $\underline{r}_0$ from (2.6-10) and (2.6-11) $\dot{\underline{r}}_0$ can be solved for as follows:

$$\frac{\underline{r}_1 - \tau_1 \dot{\underline{r}}_0}{A} = \frac{\underline{r}_2 - \tau_2 \dot{\underline{r}}_0}{B} = \underline{r}_0 \tag{2.6-17}$$

$$\dot{\underline{r}}_0 = \frac{A\,\underline{r}_2 - B\,\underline{r}_1}{A\,\tau_2 - B\,\tau_1} = \left(\frac{A}{A\,\tau_2 - B\,\tau_1}\right)\underline{r}_2 - \left(\frac{B}{A\,\tau_2 - B\,\tau_1}\right)\underline{r}_1 \tag{2.6-18}$$

Now, being able to determine $\underline{r}_0$ and $\dot{\underline{r}}_0$ from (2.6-15) and (2.6-18) allows the application of these first approximations as follows:

$$r_0 = \sqrt{\underline{r}_0 \cdot \underline{r}_0} \tag{2.6-19}$$

$$\dot{r}_0 = (\underline{r}_0 \cdot \dot{\underline{r}}_0)/r_0 \tag{2.6-20}$$

$$V_0 = \sqrt{\dot{\underline{r}}_0 \cdot \dot{\underline{r}}_0} \tag{2.6-21}$$

By use of the vis-viva equation,

$$\frac{1}{a} = \frac{2}{r_0} - \frac{V_0^2}{\mu} \tag{2.6-22}$$

it is possible to compute the semi-major axis, a, and from that determine whether the solution is elliptic, parabolic, or hyperbolic.

The next step is to evaluate the f and g series for $\underline{r}_0$ and $\dot{\underline{r}}_0$ from the following relations:

$$f = 1 - \frac{1}{2} u_0\tau^2 + \frac{1}{2} u_0p_0\tau^3 + \frac{1}{24} (3u_0q_0-15u_0p_0^2+u_0^2)\tau^4$$

$$+ \frac{1}{8}(7u_0p_0^3-3u_0p_0q_0-u_0^2p_0)\tau^5 + \frac{1}{720} (630\, u_0p_0^2q_0-$$

$$24\, u_0^2q_0-u_0^3-45u_0q_0^2-945u_0p_0^4+210u_0^2p_0^2)\tau^6 + \ldots \quad (2.6\text{-}23)$$

and

$$g = \tau - \frac{1}{6} u_0\tau^3 + \frac{1}{4} u_0p_0\tau^4 + 120(qu_0q_0-45u_0p_0^2+u_0^2)\tau^5$$

$$+ \frac{1}{360}(210u_0p_0^3-90u_0p_0q_0-15u_0^2p_0)\tau^6 +$$

$$\frac{1}{5040}(3150u_0p_0^2q_0$$

$$- 54\, u_0^2q_0-225u_0q_0^2-4725u_0p_0^4+630u_0^2p_0^2-u_0^3)\tau^7$$

$$+ \ldots \quad (2.6\text{-}24)$$

where

$$u = \mu/r^3 \quad (2.6\text{-}25)$$

$$p = \frac{r\dot{r}}{r^2} \quad (2.6\text{-}26)$$

$$q = \frac{V^2 - r^2u}{r^2} \quad (2.6\text{-}27)$$

From the general expressions

$$\underline{r}_1 = f_1\underline{r}_0 + g_1\,\dot{\underline{r}}_0 \tag{2.6-28}$$

$$\underline{r}_2 = r_2\underline{r}_0 + g_2\,\dot{\underline{r}}_0 \tag{2.6-29}$$

$\dot{\underline{r}}_0$ can be eliminated, leaving

$$\underline{r}_0 = \left(\frac{g_2}{f_1g_2-f_2g_1}\right)\underline{r}_1 - \left(\frac{g_1}{f_1g_2-f_2g_1}\right)\underline{r}_2 \tag{2.6-30}$$

and, similarly, $\underline{r}_0$ can be eliminated, leaving

$$\dot{\underline{r}}_0 = \frac{\underline{r}_2f_1-\underline{r}_1f_2}{f_1g_2-f_2g_1} = \left(\frac{f_1}{f_1g_2-f_2g_1}\right)\underline{r}_2 - \left(\frac{f_2}{f_1g_2-f_2g_1}\right)\underline{r}_1 \tag{2.6-31}$$

Using the initial estimates determined from (2.6-19), (2.6-20), and (2.6-21), the f and g series can be evaluated via (2.6-23) and (2.6-24). New values for $\underline{r}_0$ and $\dot{\underline{r}}_0$ can be immediately obtained from (2.6-30) and (2.6-31). A new value for V_0 can be obtained, once again from (2.6-21). This process can continue until a satisfactory value for the error involved is reached. As in the previous method, the problem is considered solved when the correct values of $\underline{r}_0$ and $\dot{\underline{r}}_0$ have been determined.

7. Universal Variables

By estimating a value for the eccentric or hyperbolic anomaly and making use of universal variables, a solution to the Gauss problem can be found. First, define

$$x = \sqrt{a}\,(E - E_0) \tag{2.7-1}$$

$$x = \sqrt{-a}\,(F - F_0) \tag{2.7-2}$$

and

$$z = \Delta E^2 \tag{2.7-3}$$

$$z = -\Delta F^2 \tag{2.7-4}$$

Now, the expressions for the f and g terms become

$$f = 1 - \frac{r_2}{p}(1-\cos\Delta f) = 1 - \frac{x^2}{r_1}\,C \tag{2.7-5}$$

$$g = \frac{r_1 r_2 \sin\Delta f}{\sqrt{\mu p}} = t - \frac{x^3}{\sqrt{\mu}}\,S \tag{2.7-6}$$

where

$$C = 1 - \cos\sqrt{z} \tag{2.7-7}$$

$$S = \frac{\sqrt{z} - \sin\sqrt{z}}{\sqrt{z}^3} \tag{2.7-8}$$

or for hyperbolic cases

$$C = \frac{1 - \cosh\sqrt{-z}}{z} \tag{2.7-9}$$

$$S = \frac{\sinh\sqrt{-z} - \sqrt{-z}}{\sqrt{(-z)^3}} \tag{2.7-10}$$

also,

$$\dot{f} = -\frac{\sqrt{\mu}}{r_1 r_2} x\,(1-zS) \tag{2.7-11}$$

$$\dot{g} = 1 - \frac{x^2}{2} C \tag{2.7-12}$$

From (2.7-5) it can be seen that

$$x = \sqrt{\frac{r_1 r_2(1-\cos\Delta f)}{pC}} \tag{2.7-13}$$

Now, substituting (2.7-13) into (2.7-11) yields

$$\frac{r_1 r_2(1-\cos\Delta f)}{p} = r_1 + r_2 - \frac{\sqrt{r_1 r_2}\,\sin\Delta f}{\sqrt{1-\cos\Delta f}} \tag{2.7-14}$$

Let

$$A = \frac{\sqrt{r_1 r_2}\,\sin\Delta f}{(1-\cos\Delta f)} \tag{2.7-15}$$

and

$$y = r_1 + r_2 - \frac{A(1-zS)}{\sqrt{C}} \tag{2.7-16}$$

Now x can be expressed by

$$x = \sqrt{7/C} \tag{2.7-17}$$

Solving for t gives

$$\sqrt{\mu}\, t = x^3 S + A \sqrt{y} \tag{2.7-18}$$

By using the f and g expressions and simplifying,

$$f = 1 - y/r_1 \tag{2.7-19}$$

$$g = A \sqrt{y/\mu} \tag{2.7-20}$$

$$\dot{g} = 1 - y/r_2 \tag{2.7-21}$$

and solving for the velocity:

$$\underline{r}_2 = f\underline{r}_1 + g\underline{V}_1 \tag{2.7-22}$$

$$\underline{V}_1 = \frac{\underline{r}_2 - f\underline{r}_1}{g} \tag{2.7-23}$$

The orbit is now determined and the problem considered solved.

8. Velocity Differential Determination

Given the time of departure from Earth and the time of arrival at Mars it is possible to solve for the transfer orbit using one of the aforementioned methods. The problem simplifies to two radius vectors and the time in between for transfer, "the Gauss problem." With the orbital elements, for any particular orbit, and a given time it is a simple

task to determine the radius and velocity vectors at that time. Subroutines for doing that are listed in the appendices and are of a simple derivation. Thus, the heliocentric velocity of Earth, at departure time, and Mars, at arrival date are easily determined.

For this problem it is assumed that the transfer begins from a circular orbit above Earth and ends in a circular orbit above Mars. The velocity of a spacecraft, in a circular orbit, is found from the relation

$$V_{CS} = \sqrt{\mu / r_{CS}} \tag{2.8-1}$$

Once the transfer orbit has been determined the same basic procedure as mentioned previously for the Earth and Mars velocity vectors can be used for determining the velocity vectors at departure and arrival times.

All of the information necessary for computing the total ΔV is now available. For departure, the ΔV necessary is

$$\Delta V_1 = \left| \underline{V}_{TD} - \underline{V}_E \right| - V_{CS} \tag{2.8-2}$$

where V_{TD} is the transfer orbit's velocity at departure, V_E is the velocity of the Earth, heliocentrically, at departure, and V_{CS} is the velocity of the spacecraft in its orbit above Earth.

Determining the ΔV necessary for arrival at Mars is computed in a similar manner.

$$\Delta V_2 = \left|\underline{V}_{TA} - \underline{V}_M\right| - V_{CS} \qquad (2.8\text{-}3)$$

where V_{TA} is the transfer orbit's velocity at arrival at Mars, V_M is the heliocentric velocity of Mars, and V_{CS} is the circular velocity of the wanted parking orbit above Mars. The total ΔV for the nominal mission is then determined to be

$$\Delta V_{TOT} = \Delta V_1 + \Delta V_2 \qquad (2.8\text{-}4)$$

To determine the necessary ΔV for a transfer between two points, which are of negligible mass, the same approach holds. However, it is simplified, somewhat, since there is no need to determine the V_{CS} since most transfers would incorporate rendezvous rather than parking orbits.

9. Newton-Raphson Minimization

The Newton-Raphson Method is a method used for solving a set of algebraic equations. The method allows re-estimating a set of initial values for the solution of the set of algebraic equations. The process can be repeated until the change between the i^{th} and $i+1^{st}$ estimate is within acceptable tolerance.

In this particular application only one equation is to be minimized, so the procedure reduces to a simplified form.

Given a function f(x) it is possible to find an extremum for the function where the first derivative is exactly equal to zero.

$$\frac{df(x)}{dx} = 0 \qquad (2.9\text{-}1)$$

A necessary condition for a minimum in the function is

$$\frac{d^2f(x)}{dx^2} > 0 \qquad (2.9\text{-}2)$$

However, in this particular problem the method will not converge to anything but a minimum due to the form of the function.

Using a central difference method to approximate the correction term for the function at the first guess for the solution, x, is

$$\Delta x = \frac{df/dx}{d^2f/dx^2} = \frac{\delta}{2}\,\frac{f(x+\delta) - f(x-\delta)}{f(x+\delta)-2f(x)+f(2-\delta)} \qquad (2.9\text{-}3)$$

where δ is the incremental change in x.

A new guess for x can now be found from

$$x_{i+1} = x_i - \Delta x \qquad (2.9\text{-}4)$$

This procedure can be continued until the first derivative is within an acceptable tolerance of zero, to give a numerical solution to the problem. A computer

subroutine for the application of this method is included in the appendices.

10. Final Problem Algorithm

Use must be made of several of the preceding derivations to solve for an optimum transfer between two orbits. The building block for the entire program is the orbit determination program linked with one of the Gauss problem solutions. Also necessary are the radius and velocity determination subroutines RV and VV which are necessary not only for the orbit determination but also for the computation of the total ΔV required. The extremization subroutine is written in such a manner as to extremize any function F. The function F must be written so as to output total ΔV required for a given time input from the Newton-Raphson subroutine.

To sum the basic algorithm up, the Newton-Raphson subroutine calls a function F which gives a value for total ΔV, dependent on the time input. The function F calls the orbit determination subroutines and computes the ΔV for the input time, given from the Newton-Raphson subroutine. When the Newton-Raphson subroutine encounters a minimum it stops, having optimized the expenditure of minimum ΔV between two orbits.

CHAPTER III

NUMERICAL RESULTS

1. Introduction

Here the methods developed in the previous chapter will be used to investigate the optimal orbits for transferring between two orbits. The nominal orbits considered are those needed for Earth to Mars flights of various time lengths. In particular, three classes of Earth-Mars trajectories are considered:

1. Early Manned Missions
2. Advanced Manned Missions
3. Freighter Missions

The early manned flights considered are assumed to be trips of around 150 days. The advanced manned missions will be assumed hyperbolic and with flight times of near 50 days. The freighter missions will be assumed unmanned and having trip durations of near 200 days.

Before looking into the optimal aborts it is necessary to compute the nominal mission characteristics. The first section will deal with the nominal missions.

It should be mentioned that no optimum time for initiation time, for the transfer orbit, was found to exist. Therefore the optimization deals only with the actual transfer time.

2. Nominal Earth-Mars Trajectories

By using the true anomaly iteration technique, as derived in Section II.5, it is possible to solve for the necessary orbital elements for an Earth-Mars mission of given length, with given departure time. It is necessary to specify the orbital elements for both Earth and Mars so that the radius vectors, $\underline{r}_1$ and $\underline{r}_2$, can be determined for the departure date from Earth and the arrival date at Mars. With these known, the problem is simply the two-radius-vectors-and-time problem discussed in Chapter II. All of the calculations are based on a departure date of 2,444,200.5 Julian Days (October, 1980) and the trip lengths are from 40 days up to 250 days. Six orbital elements are computed for each orbit and the velocity differential (ΔV) is computed so as to determine the energy necessary for a given flight. The velocity differential is determined as outlined in Chapter II and assumes departure from Earth, and arrival at Mars, from, and to, a 500 kilometer circular orbit above the planet's surface; see Figure 3.1. The computer program used for these calculations is listed in the appendices.

3. Early Manned Missions

It is not worthwhile to spend a great deal of effort on the Early Manned Mission cases. Although it is possible that space stations of some sort will exist when the first manned missions depart for Mars, it is most likely that they

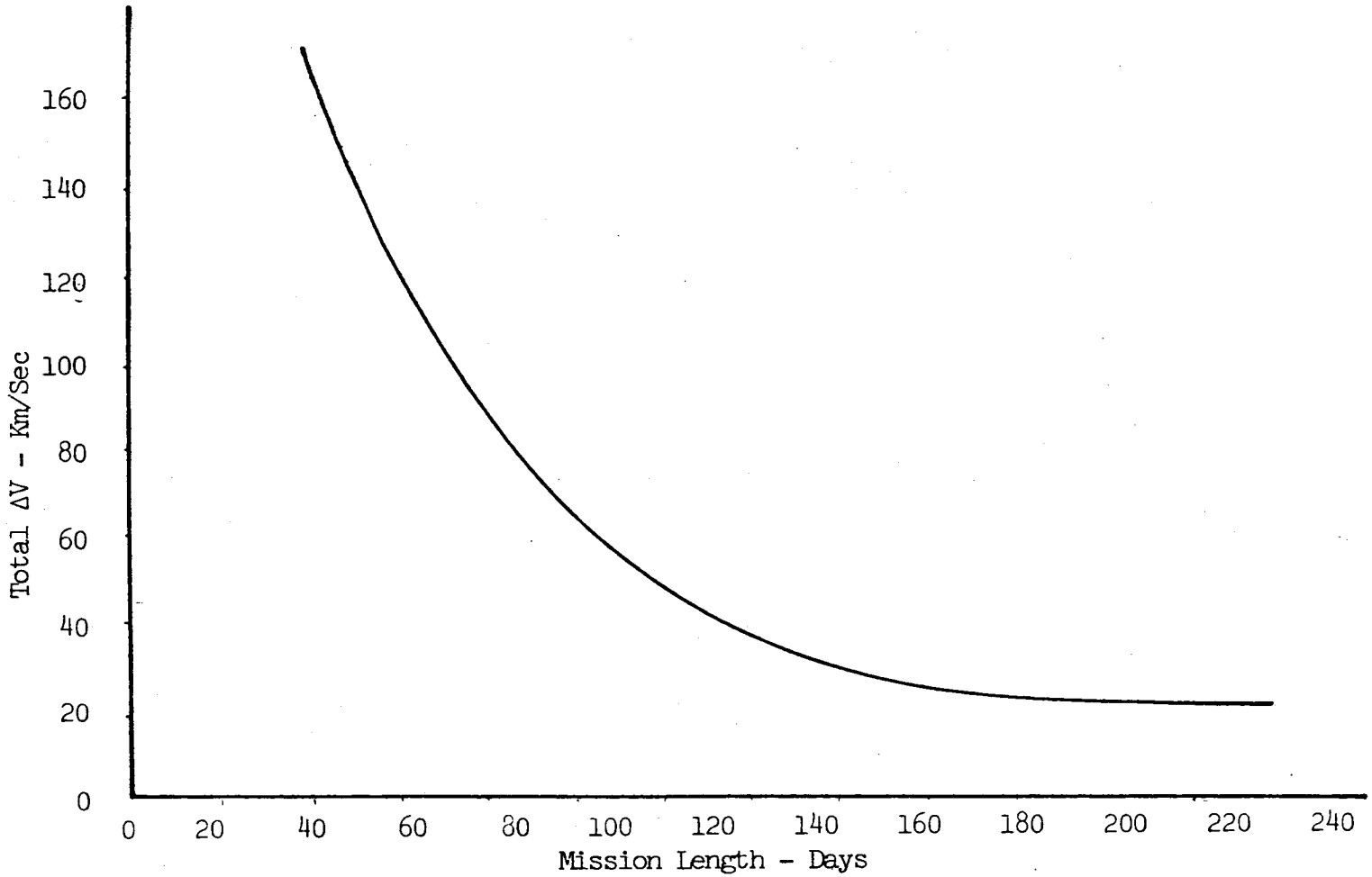


Figure 3.1. Earth-Mars Transfer.
Mission Length vs. ΔV

will be in geocentric orbits and will have little application after the initial portion of the flight. It is very likely that the first missions will be done in groups of two or more spacecraft for safety reasons. If an en route emergency does occur the spacecraft will have reduced ΔV potential and an unaffected life support system or, just the opposite, reduced life support capability and unaffected ΔV. One comforting characteristic of this type of mission is that it will, very likely, carry enough ΔV for the entire, round trip, mission. This is particularly fortunate for the first half of the mission since it expands the number of available options, if an emergency should arise.

Results for two abort strategies follow. Both are for an emergency 50 days into the nominal 150-day mission. The first plot shows the optimum turn-back orbit (see Figure 3.2), and the second shows the best orbit to continue on (see Figure 3.3).

Different emergency situations may have completely different constraints. It may be that no transfer orbit can help the endangered spacecraft. It is also possible that the mission can be completed, but with a few sacrifices, such as time on Mars. Many other possibilities exist and the only purpose of the following examples is to help demonstrate the applicability of the procedure to what may, someday, be actual situations.

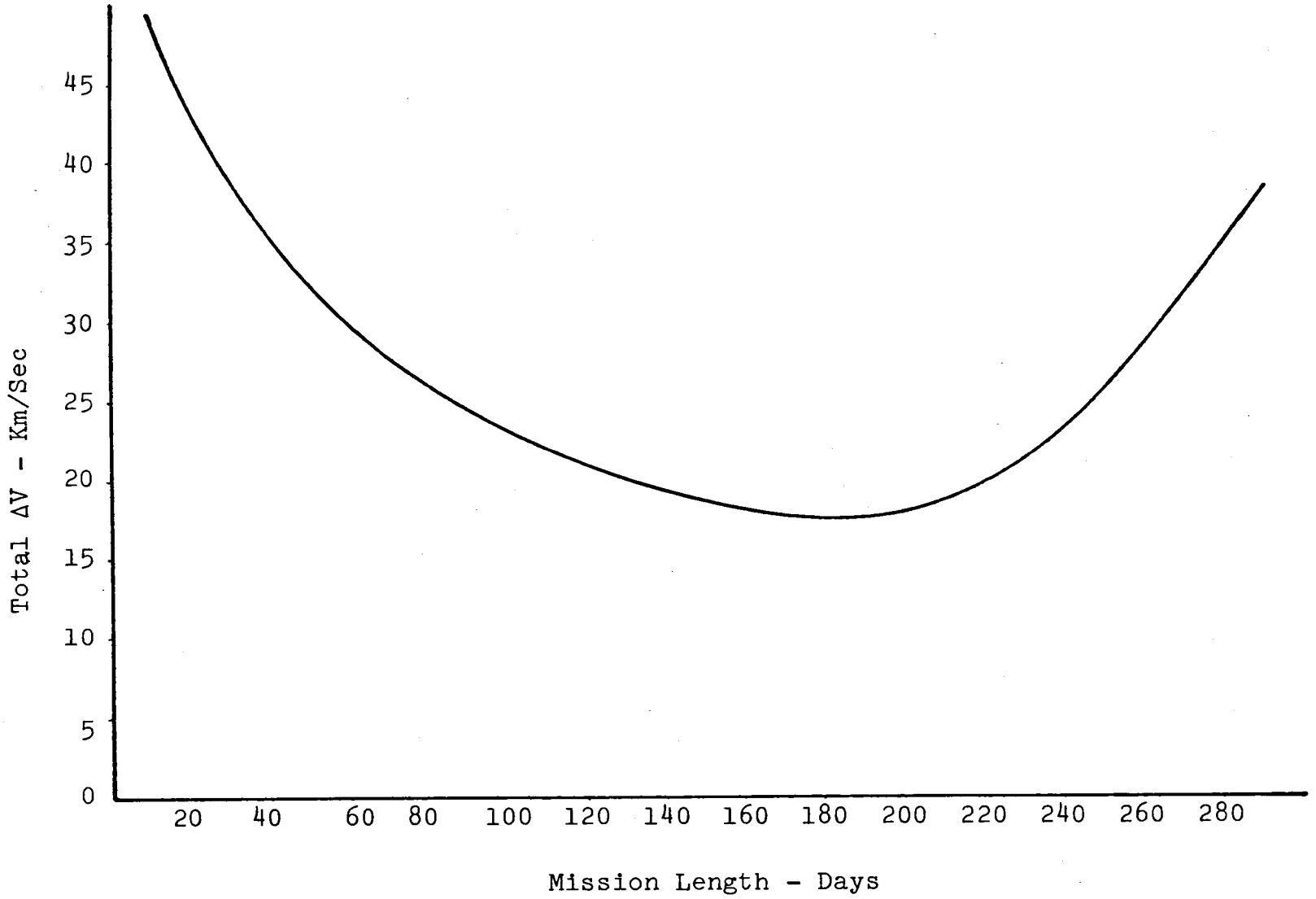


Figure 3.2. Early Manned Mission, Mid-Transfer Turn Back. Mission Length vs. ΔV.

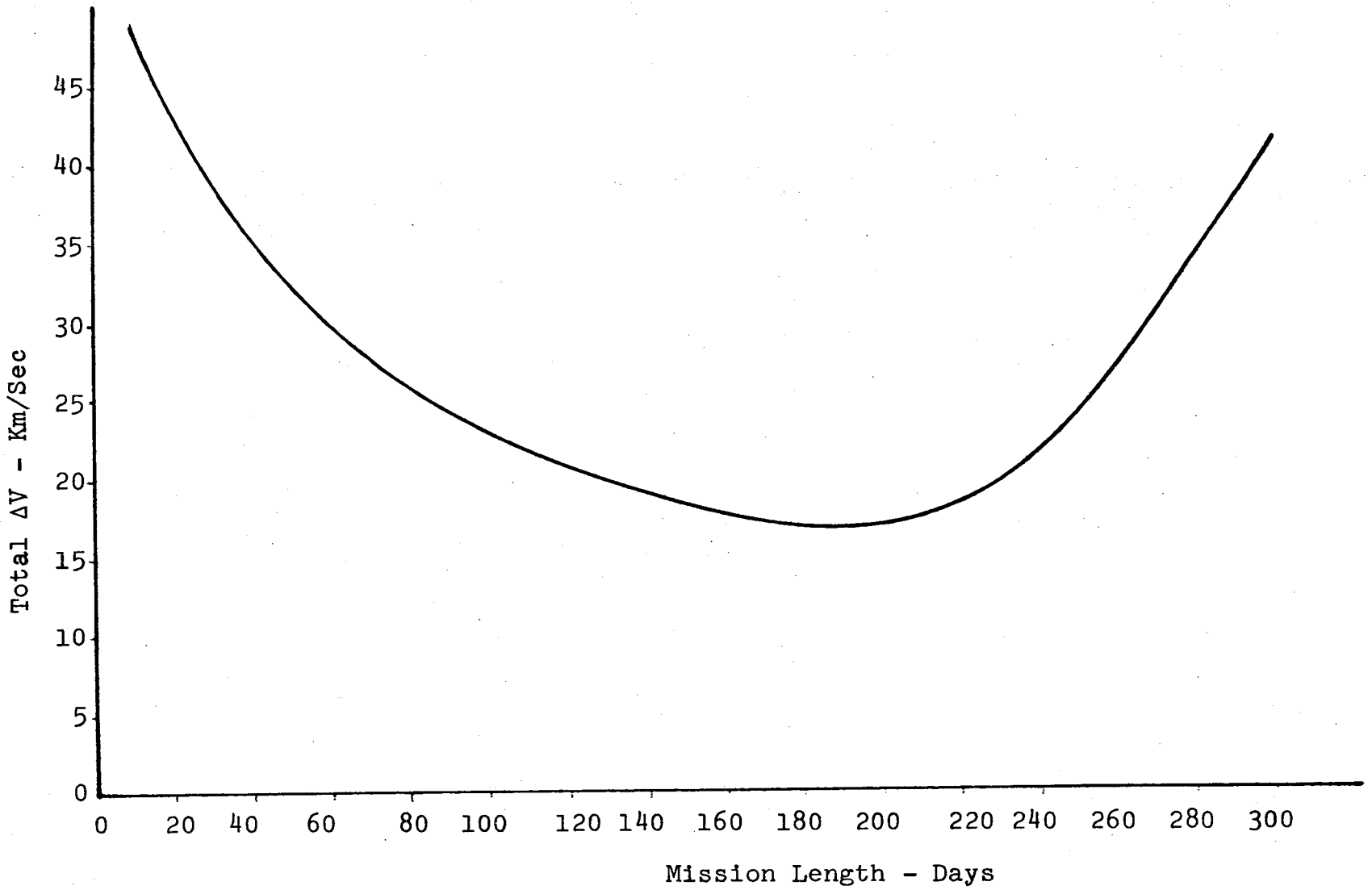


Figure 3.3. Early Manned Mission, Mid-Transfer Abort to Mars. Mission Length vs. ΔV.

Figure 3.2 shows the total velocity change (ΔV) needed, for a given time, to turn back to Earth. Figure 3.3 shows the ΔV required to Mars and, interestingly enough, shows that nearly the same amount of ΔV is necessary to continue, only to Mars, as is to return to Earth. It should be remembered that these examples assume an abort 50 days into the mission.

4. Advanced Manned Missions

A mission time of around 50 days is assumed for the advanced manned mission. It will also be assumed that pre-positioned space stations will exist, in heliocentric orbits, at that time. The nature of such stations will not be detailed; it is assumed only that a successful rendezvous with a space station insures a rescue of the crew or cargo involved.

Making use of the same algorithm as used in computing the nominal Earth-Mars trajectories, and adding the Newton-Raphson Extremization Subroutine, the optimum transfer, for minimum ΔV expenditure, can be determined.

The stations, it is assumed, are positioned midway between Earth and Mars, and are separated by 90° on the same orbit. The ΔV-versus-time plots are similar for all of the stations so results have been shown for only stations numbers 1 (Figure 3.4) and 2 (Figure 3.5) of the 4.

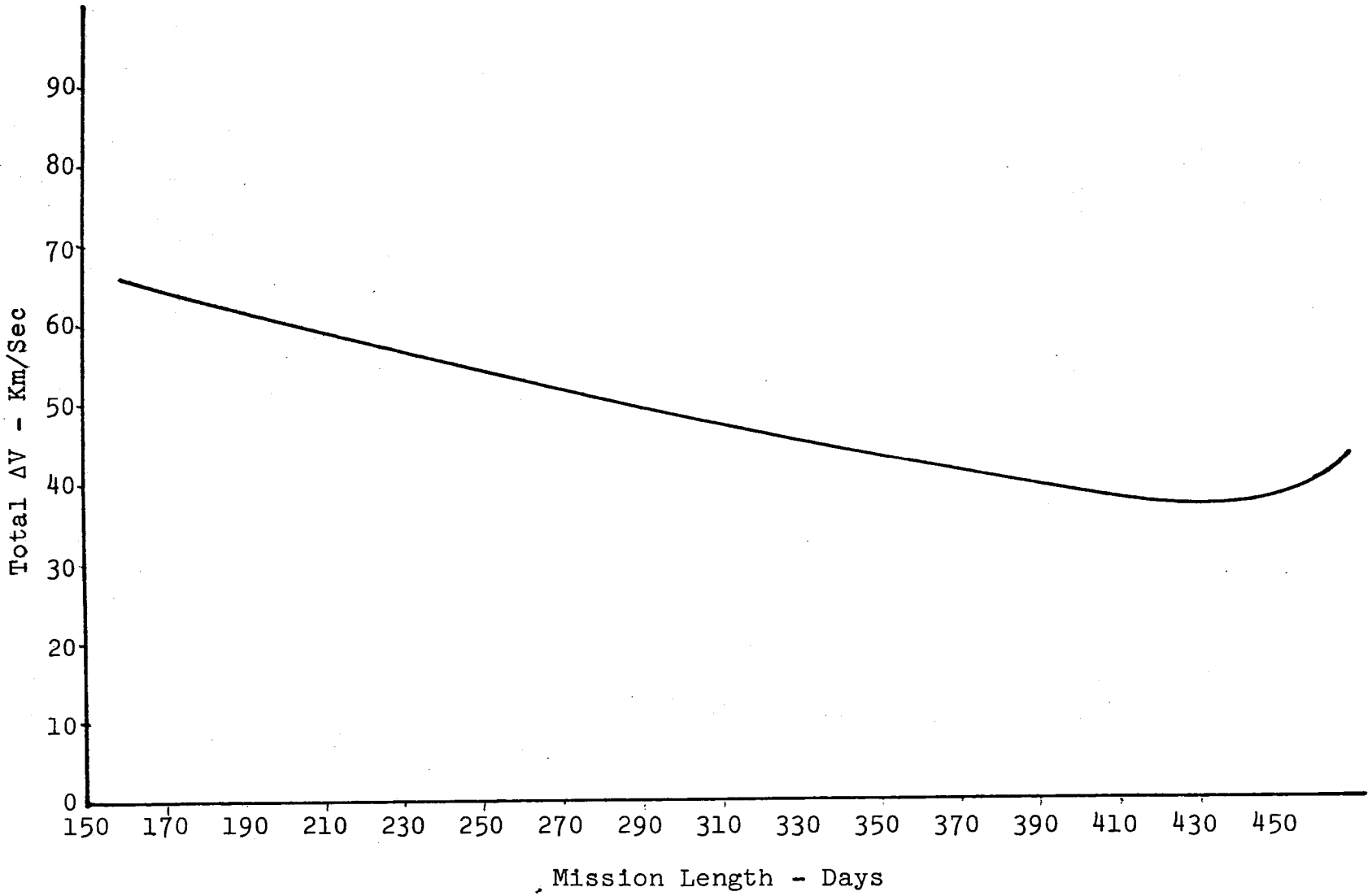


Figure 3.4. Advanced Manned Mission Mid-Transfer Abort to Station #1. Mission Length vs. ΔV.

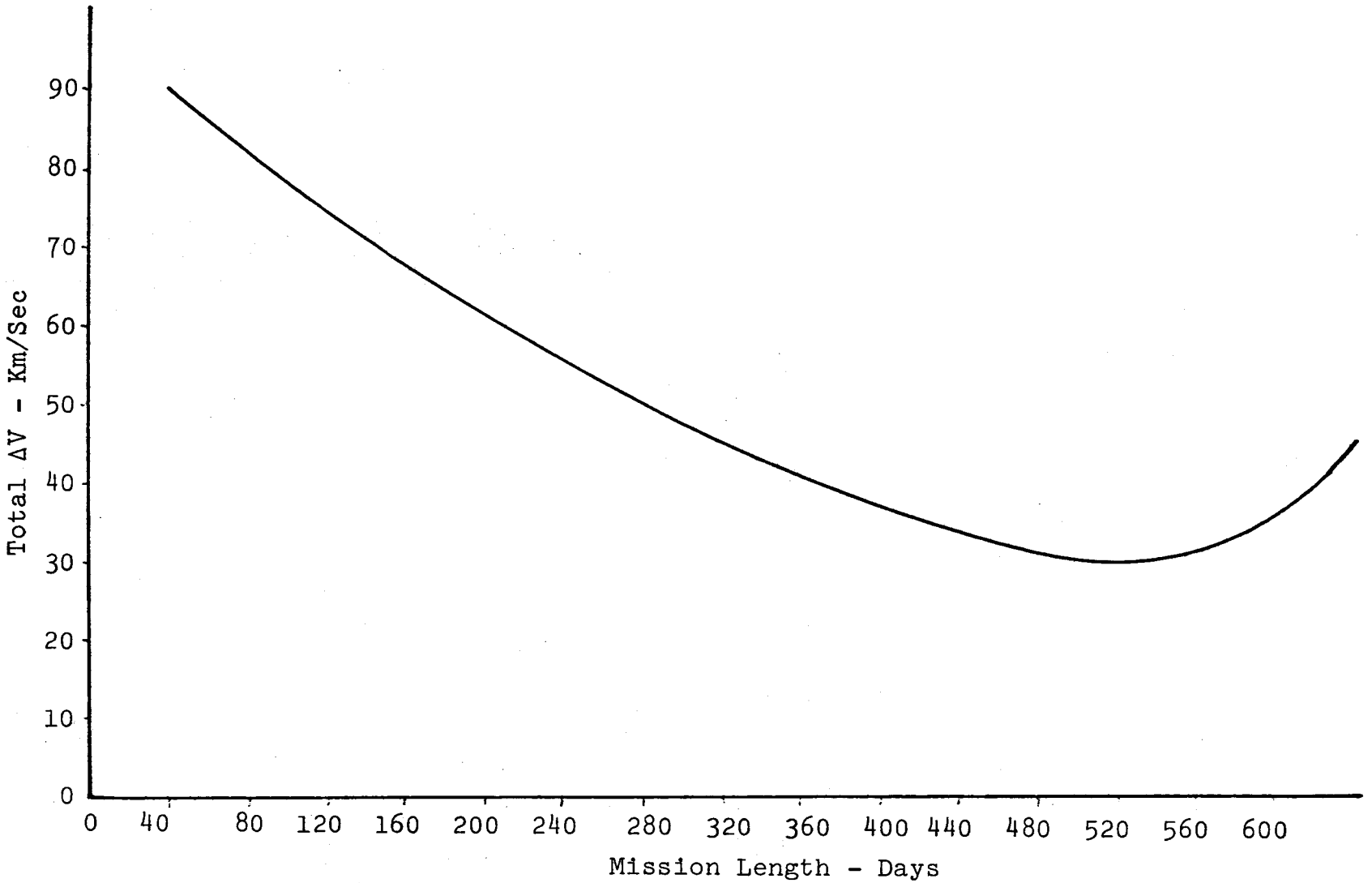


Figure 3.5. Advanced Manned Mission Mid-Transfer Abort to Station #2. Mission Length vs. ΔV.

It should be noted that the rendezvous times are extremely long, in the minimum-ΔV cases, and that, most likely, no manned mission would have life support reserves adequate to complete such long trips.

Since the advanced missions will require ΔV potentials of over 100 Km/Sec it is very likely that the minimum fuel orbit would not be used. It is much more likely that the optimum orbit, with constraints on the time en route, will be of greater significance.

5. Freighter Missions

Of the three mission types considered, the freighter mission has the least number of options. The freighter missions will be assumed unmanned and given only enough ΔV for a slow, around 200 days long, one-way mission. In the event of an en route loss of ΔV potential, it will be difficult to use the remaining ΔV effectively but no time constraints will exist either.

The results are based on a nominal 200 day mission with an en route emergency on the 100^{th} day. Rendezvous with all of the stations is considered and is shown in Figure 3.6. Assuming a need for rendezvous with a station, the minimum energy transfer can be easily found via the Newton-Raphson method. However, most situations may be concluded by continuing to Mars. If some method for entering a circular Mars orbit is available, upon arrival at Mars, then it would seem logical to continue. By the time

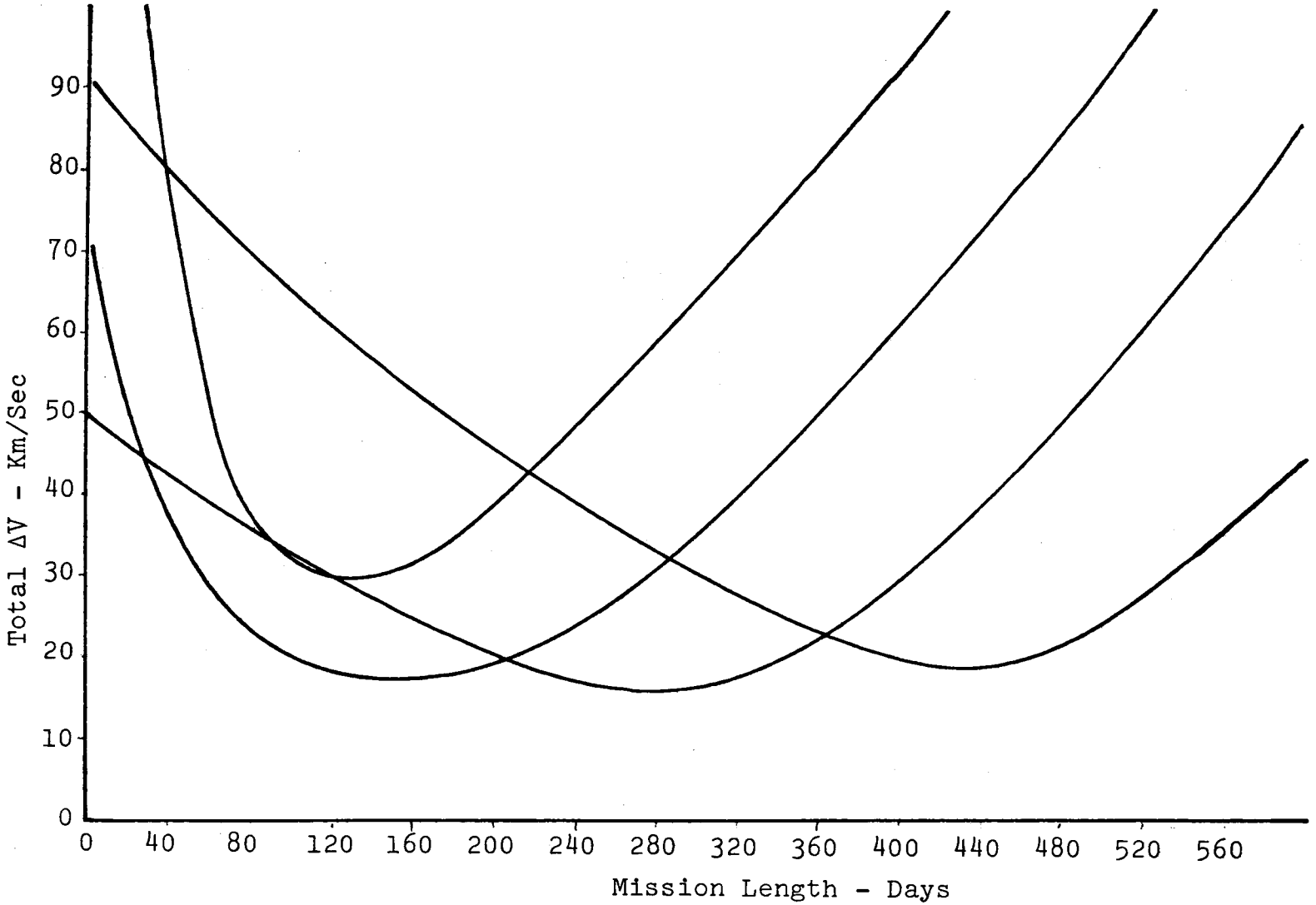


Figure 3.6. Freighter Mission Mid-Transfer Abort to Station Mission Length vs. ΔV.

such missions actually exist, it is very likely that these methods will exist. At that time a more complex optimization problem may consider the minimum energy rendezvous between freighter and de-orbit vehicle.

BIBLIOGRAPHY

BIBLIOGRAPHY

1. Bolger & Childs, Long Range Planning for the Development of Space Flight Emergency Systems, International Academy of Astronautics, International Astronautical Congress, 22nd, Brussels, Belgium, Sept. 20-25, 1971.

2. Bates, Mueller, & White, Fundamentals of Astrodynamics, Dover Publications, Inc., New York, 1971.

3. Escobal, P. R., Methods of Orbit Determination, John Wiley & Sons, Inc., New York, London, Sydney, 1963.

4. Planetary Flight Handbook, Volume 3, National Aeronautics and Space Administration, Washington, D. C., 1963.

5. The American Ephemeris & Nautical Almanac, 1976, U. S. Government Printing Office, Washington, D. C., 1974.

6. Nelson & Loft, Space Mechanics, Prentice-Hall, Inc., Englewood Cliffs, N. J., 1962.

7. Bates, D. R., Space Research and Exploration, William Sloane Associates, New York, 1961.

APPENDICES

APPENDIX A

SUBROUTINE OD(R,V,A,XMA,E,XI,XLN,AP,TA)

This subroutine computes the orbital elements, given the orbit's radius and velocity vector, for a given time.

The input data are:

R 3X1 array containing the components of the radius vector

V 3X1 array containing the components of the velocity vector

The output data are:

A semi-major axis

XMA mean anomaly at t_0

E eccentricity

XI inclination

XLN longitude of the ascending node

AP argument of pericenter

TA true anomaly

R and V must be dimensioned in the calling program. No additional subroutines or functions are required, besides the calling program. A common statement must be included for the constants shown in the program.

```
      SUBROUTINE OD(R,V,A,XMA,E,XI,XLN,AP,TA)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/C1/XK,DE,DM,XK1,XK2,PI,CF2
      DIMENSION R(3),V(3)
      DATANH(X)=DLOG((1.+X)/(1.-X))/2.
      XR-DSQRT(R(1)**2+R(2)**2+R(3)**2)
      XV=DSQRT(V(1)**2+V(2)**2+V(3)**2)
      HI=R(2)*V(3)-V(2)*R(3)
      HJ=R(1)*V(3)-V(1)*R(3)
      HK=R(1)*V(2)-R(2)*V(1)
      H=DSQRT(HI**2+HJ**2+HK**2)
      P=H*H/XK
      RDV=R(1)*V(1)+R(2)*V(2)+R(3)*V(3)
      XVR=RDV/XR
      XVF=DSQRT(XV*XV-XVR*XVR)
      E=DSQRT(1.+(XR*XVF*XVF/XK)*(XR*XV*XV/XK-2.))
      A=P/DABS(1.-E*E)
      W=XV*XV-XK/XR
      EI=(W*R(1)-RDV*V(1))/XK
      EJ=(W*R(2)-RDV*V(2))/XK
      EK=(W*R(3)-RDV*V(3))/XK
      IF(DABS(HK/K).LT.0.99)GO TO 1
      XI=0.0
      IF(HK.LT.0.0)XI=PI
      GO TO 2
    1 XI=DACOS(HK/H)
```

```
2 XNI=-HJ
  XNJ=-HI
  XNK=0.0
  XN=DSQRT(XNI*XNI+XNJ*XNJ)
  XLN=DACOS(XHI/XN)
  IF(XNJ.LT.0.0)XLN=2.*PI-XLN
  XNDE=XNI*EI+XNH*EJ+XNK*EK
  AP=XNDE/(XN*E)
  IF(EK.LT.0.0)AP=2.*PI-AP
  EDR=EI*R(1)+EJ*R(2)+EK*R(3)
  TA=DACOS(EDR/(E*XR))
  IF(RDV.LT.0.0)TA=2.*PI-TA
  IF(E.GT.1.0)GO TO 3
  EA=2.*DATAN(DSQRT((1.-E)/(1.+E)*DTAN (TA/2.))
  XMA=EA-E*DSIN(EA)
  GO TO 4
3 FA=2.*DATANH(DSQRT((E-1.)/(E+1.))*DTAN(TA/2.))
  XMA=E*DSINH(FA)-FA
4 RETURN
  END
```

APPENDIX B

SUBROUTINE PITR(R1,R2,T,A,E,XI,XLN,AP,XMA,TA)

This subroutine computes the orbital elements, given two radius vectors and the time of transfer between them.

The input data are:

R1 3X1 array containing the components of the first radius vector

R2 3X1 array containing the components of the second radius vector

T the time of transfer between the radius vectors

The output data are:

A semi-major axis

E eccentricity

XI inclination

XLN longitude of the ascending node

AP argument of pericenter

XMA mean anomaly at t_0

TA true anomaly

R1 and R2 must be dimensioned in the calling program. A common statement must be included for the constants shown in the program.

```
      SUBROUTINE PITR(R1,R2,T,A,E,XI,XLN,AP,XMA,TA)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/C1/XK,DE,DM,XK1,XK2
      DIMENSION R1(3),R2(3),U1(3),U2(3),W(3)
      RA=DSQRT(R1(1)**2+R1(2)**2+R1(3)**2)
      RB=DSQRT(R2(1)**2+R2(2)**2+R2(3)**2)
      ZNU=ACOS((R1(1)*R2(1)+R1(2)*R2(2)+R1(3)*R2(3))/(RA*RB))
      XK5=RA*RB*(1.-DCOS(ZNU))
      XL1=RA+RB
      XM1=RA*RB*(1.+DCOS(ZNU))
      P1-XK5/(XL1+DSQRT(2.*XM1))
      P2=XK5/(XL1-DSQRT(2.*XM1))
      P=(P1+P2)/2.
    1 A=P*XM1*XK5
      B=(2.*XM1-XL1*XL1)*P*P
      Q=2.*XK5*XL1*P-XK5*XK5
      XA=A/(B+Q)
      F=1.(RB/P)*(1.-DCOS(ZNU))
      G=(RA+RB*DSIN(ZNU))/DSQRT(XK*P)
      A=DSQRT(XK/P)*DTAN(ZNU/2.)
      B=(1.-DCOS(ZNU))/P-(1./RA)-(1./RB)
      FD=A*B
      IF(XA.LT.0.0)GO TO 2
      DE=DCOS(1.-(RA/XA)*(1.-F))
      TN=G+DSQRT(XA**3/XK)*(DE-DSIN(DE))
```

```
      A=(XK5**2+(2.*XM1-XL1**2)*P*P)/(XM1*XK5*p*p)
      B=(DSQRT(XA**3/XK))*2.*XK5*DSIN(DE)/P*(XK5-XL1*P)
      DT=(-G/2.*P)-1.5*XA*(T-G)*A+B
      E=DSQRT(1.-P/XA)
      IF(DT.LT.1.D-12)GO TO 3
      P=P+(TN-T)/DT
      GO TO 1
    2 DF=ACOSH(1.-RA/XA)*(1.-F))
      TN=G+DSQRT((DABS(XA))**3)/XK)*(DSINH(DF)-DF)
      A=(XK5*XK5+(2.*XM1-XL1)*P*P)/(XM1*XK5*P*P)
      B=(DSQRT((DABS(XA)**3)/XK))*2.*XK5*DSIN(DE)/P*(XK5-XL1*P)
      DT=(-G/2.*P(-1.5*XA*(T-G)*A-B
      E=DSQRT(1.+DABS(P/XA))
      SALPHA=DSIN(ZNU)
      DO 10 I=1,3
      U1(I)=R1(I)/RA
   10 U2(I)=R2(I)/RB
      W(1)=(U1(2)*U2(3)-U1(3)*U2(2))/SALPHA
      W(2)=(U1(3)*U2(1)-U1(1)*U2(3))/SALPHA
      W(3)=(XN3/SALPHA
      XI=ACOS(W(3))
      Z=DATAN(W(1)/(DSIN(XI)-W(2)))
      IF(Z.LT.0.0)Z=Z+3.14159
      XLN=2.*Z
      SLN=DSIN(XLN)
      CLN=DCOS(XLN)
```

```
      Z+DATAN((-U1(1)*SLN+U1(2)*CLN)/((1.+U1(1)*CLN+U1(2)*SLN)*
     1DCOS(XI)))
      IF(Z.LT.0.)Z=Z+3.14159
      TA=ACOS((P/RA-1.)/E)
      AP=2.*Z-TA
      IF(DT.LT.1.D-12)GO TO 3
      P=P+(T-TN)/DT
      GO TO 1
    3 RETURN
      END
```

APPENDIX C

SUBROUTINE TAN(R1,R2,T1,T2,I,XØ,XK,E,P,AP,XLN,XI,XMA)

This subroutine computes the orbital elements from two radius vectors at two given times.

The input data are:

R1 3X1 array containing the components of the radius vector at time t_1

R2 3X1 array containing the components of the radius vector at time t_2

T1 time t_1

T2 time t_2

I 1 for a direct orbit, -1 for a retrograde orbit

XØ estimate for the true anomaly at time t_1

XK gravitational parameter k

The output data are:

E eccentricity

P semi-latus rectum

AP argument of pericenter

XLN longitude of the ascending node

XI inclination

XMA mean anomaly at time t_1

R1 and R2 must be dimensioned in the calling program. Subroutine TAN requires functions FODl and X.

```
      SUBROUTINE TAN(R1,R2,T1,T2,I,X0,E,P,AP,XLN,XI,XMA)
      IMPLICIT REAL*8(A-H,0-Z)
      DIMENSION R1(3),R2(3),XN(3)
      COMMON/COD1/ALPHA,XMR1,XMR2
      EXTERNAL FOD1
      DATANH(X)=DLOB((1.+X)/1.-X))/2.
      XMR1=DSQRT(R1(1)**2+R1(2)**2+R1(3)**2)
      XMR2-DSQRT(R2(1)**2+R2(2)**2+R2(3)**2)
      ALPHA=DACOS((R1(1)*R2(1)+R1(2)+R1(3)*R2(3))/(XMR1*X
     1MR2))
      XN(1)+R1(2)*R2(3)-R1(3)*R2(2)
      XN(2)=R1(3)*R2(1)-R1(1)*R2(3)
      XN(3)=R1(1)*R2(2)-R1(2)*R2(1)
      IF(I*XN(3).LT.0.0)ALPHA=6.283185307-ALPHA
      XMN=XMR1*XMR2*DSIN(ALPHA)
      DO 1 J=1,3
    1 XN(J)=XN(J)/XMN
      F1=X(FOD1,X0)
      F2=F1+ALPHA
      CF1=DCOS(F1)
      CF2=DCOS(F2)
      P=(CF1-CF2)*XMR1*XMR2/(XMR1*CF1-XMR2*CF2)
      E=(XMR2-XMR1)/(XMR1*CF1-XMR2*CF2)
      IF(E.GT.1.)GO TO 2
      CE1-2.*DATAN(DSQRT((1.-E)/(1.+E))*DTAN(F1/2.))
      XMA=CE1-E*DSIN(CE1)
```

```
      GO TO 3
    2 CF1=2.*DATANH(DSQRT((E-1.)/E+1.))*DTAN(F1/2.))
      XM1=E*DSINH(CF1)-CF1
    3 XI=DACOS(XN(3))
      XLN=2.*DATAN(XN(1)/(DSIN(XI)-XN 2)))
      SLN=DSIN(XLN)
      CLN=DCOS(XLN)
      AP=2.*DATAN((-R1(1)*SLN+R1(2)*CLN)/((XMR1+R1(1)*CLN+R1(2)*
     1SLN)*DCOS(XI)))-F1
      RETURN
      END
```

```
      FUNCTION FOD1(F1)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/COD1/ALPHA,T1,T2,XK,XMR1,XMR2
      DATANH(X)=DLOG((1.+X)/(1.-X))/2.
      F2=F1+ALPHA
      CF1=DCOS(F1)
      CF2=DCOS(F2)
      P=(CF1-CF2)*XMR1*XMR2/(XMR1*CF1-XMR2*CF2)
      E=(XMR2-XMR1)/XMR1*CF1-XMR2*CF2)
      IF (E.GT.1.) GO TO 1
      A=DSQRT((1.-E)/(1.+E))
      CE1=2.*DATAN(A*DTAN(F1/2.))
      CE2=2.*DATAN(A*DTAN(F2/2.))
      XM1=CE1-E*DSIN(CE1)
      XM2-CE2-E*DSIN(CE2)
      FOD1=T2-T1-(XM2-XM1)*DSQRT((P/(1.-E**2))**3/XK)
      GO TO 2
    1 A=DSQRT((E-1.)/(E+1.))
      CF1=2.*DATANH(A*DTAN(F1/2.))
      CF2=2.*DATANH(A*DTAN(F2/2.))
      XM1=E*DSINH(CF1)-CF1
      XM2=E*DSINH(CF2)-CF2
      FOD1=T2-T1-(XM2-XM1)*DSQRT((P/(E**2-1.))**3/XK)
    2 RETURN
      END
```

FUNCTION XE(F,XO)

This function returns the value of x that extremizes the function f(x).

XO is an estimate for the extremal x. F is a dummy name for a user-supplied function. This function has a single argument, X, and returns the value f(X). The function's actual name must appear in an EXTERNAL statement in the calling program.

```
      FUNCTION XE(F,XO)
      IMPLICIT REAL*8 (A-H,O-Z)
      XE=XO
    1 F1=F(0.99*XE)
      F2=F(1.01*XE)
      DXE=0.005*XE*(F2-F1)/F1-2.*F(XE)+F2)
      XE=XE-DXE
      IF (DABS(DXE/XE).GT.1.D-12) GO TO 1
      RETURN
      END
```

APPENDIX D

SUBROUTINE FGS(R1,R2,V1,T1,T2,E,A,AP,XLN,XI,XMA)

This subroutine computes the orbital elements, given two radius vectors and the time of transfer between them.

The input data are:

R1 3X1 array containing the components of the first radius vector

R2 3X1 array containing the components of the second radius vector

T1 time at t_0

T2 time at t_1

V1 3X1 array containing the components of the initial estimate for the velocity at t_0

The output data are:

A semi-major axis

E eccentricity

XI inclination

XLN longitude of the ascending node

AP argument of pericenter

XMA mean anomaly at t_o

R1, R2, and V1 must be dimensioned in the calling program. A common statement must be included for the constant, XK (gravitational constant). Subroutine FGS requires the inclusion of Subroutine OD.

```
      SUBROUTINE FGS(R1,R2,V1,T1,T2,E,A,AP,XLN,XI,XMA)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION R1(3),R2(3),V1(3)
      COMMON/C1/XK
      RA=DSQRT(R1(1)**2+R1(2)**2+R1(3)**2)
    1 VA=DSQRT(V1(1)**2+V1(2)**2+V1(3)**2)
      RDV=(R1(1)*V1(1)+R1(2)*V1(2)+R1(3)*V1(3))/(RA*VA)
      U=XK/(RA**3)
      P=RDV/(RA**2)
      Q=(VA**2)/(RA**2)-U
      T=T2-T1
      F=1-.5*U*T*T+.5*U*P*(T**3)+(1/24)*U*(U-15*P*P+3*Q)*(T**4)+
        (.125)*U
     1*XP*(7*P*P-U-3*Q)*(T**5)
      G=T-(1/6)*U*(T**3)+.25*U*P*(T**4)+(1/120)*U*(U-45*P*P+9*Q)*
        (T**5)
      DO 2 I=1,3
    2 V1(I)=(R2(I)-F*R1(I))/G
      VC=DSQRT(V1(1)**2+V1(2)**2+V1(3)**2)
      TEST=DABS(VC-VA)
      IF(TEST.GT.1.D-12)GO TO 1
      CALL OD(R1,V1,A,XMA,E,XI,XLN,AP,TA)
      RETURN
      END
```

APPENDIX E

SUBROUTINE UV(R1,R2,T1,T2,A,XMA,E,XI,XLN,AP)

This subroutine computes the orbital elements, given two radius vectors and the time of transfer between them.

The input data are:

R1 3X1 array containing the components of the first radius vector

R2 3X1 array containing the components of the second radius vector

T1 time at t_0

T2 time at t_1

The output data are:

A semi-major axis

E eccentricity

XI inclination

XLN longitude of the ascending node

AP argument of pericenter

XMA mean anomaly at t_0

R1 and R2 must be dimensioned in the calling program. A common statement must be included for the constants shown in the program. Subroutine UV requires the inclusion of Subroutine OD.

```
      SUBROUTINE UV(R1,R2,T1,T2,A,XMA,E,XI,XLN,AP)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/C1/XK,DE,DM,XK1,SK2,PI,CF2
      DIMENSION VS1(3),VS2(3),U1(3),U2(3),R1(3),R2(3),R0(3)
      T=T2-T1
      RA=DSQRT(R1(1)**2+R1(2)**2+R1(3)**2)
      RB=DSQRT(R2(1)**2+R2(2)**2+R2(3)**2)
      DO 1 I=1,3
      U1(I)=R1(I)/RA
    1 U2(I)=R2(I)/RB
      ZNU=DACOS(U1(1)*U2(1)+U1(2)*U2(2)+U1(3)*U2(3))
      UC1=(U1(1)*U2(2)-U2(1)*U1(2))
      IF(UC1.LT.0.0)ZNU=2.*PI-ZNU
      ZNUS=ZNU
      AS=DSQRT(R1*R2)*DSIN(ZNUS)/DSQRT(1.-DCOS(ZNUS))
      Z=0.0
      S=1/6
      C=0.5
      YS=R1+R2-AS*(1.-Z*S)/DSQRT(C)
      IF(YS.LT.0.0)GO TO 4
      XS=DSQRT(YS/C)
      TS=(XS**3)*S+AS*DSQRT(YS))/DSQRT(XK)
      SP=(1./(2.*Z))*(C-3.*S)
      CP=(1./(2.*Z))*(1.-Z*S-2.*C)
      P=SP-(3.*S*CP/(2.*C))
      QS=3*S*DSQRT(YS)/C+AS/XS
```

```
      DTS=((XS**3)*P+(AS/8.)*QS)/DSQRT(XK)
      ZS=Z+(T-TS)/DTS
    8 IF(ZS.LT.0.0)GO TO 16
      SZ=DSQRT(ZS)
      S=(SZ-DSIN(SZ))/DSQRT(ZS**3)
      C=(1.-DCOS(SZ))/ZS
      GO TO 18
   16 SZ=DSQRT(DABS(ZS))
      S=(DSINH(SZ)/DSQRT(DABS(ZS**3))
      C=(1.-DCOSH(SZ))/ZS
   18 YS=R1+R2-AS*(1.-ZS*S)/DSQRT(C)
      IF(YS.LT.0.0)GO TO 4
      XS=DSQRT(YS/C)
      TS=((XS**3)*S+AS*DSQRT(YS))/DSQRT(XK)
      SP=(1./(2.*ZS))*(C-3.*S)
      CP=(1./(2.*ZS))*(1.-ZS*S-2.*C)
      P=SP-(3.*S*CP/(2.*C))
      QS=3*S*DSQRT(YS)/C+AS/XS
      DTS=((XS**3)*P+(AS/8.)*QS)/DSQRT(XK)
      ZS=ZS+(T-TS)/DTS
      U=(T-TS)/DTS
      U=DABS(U)
      IF(U.LT.1.D-12)GO TO 10
      GO TO 8
   10 FS=1.-YS/R1
      GS=AS*DSQRT(YS/XK)
```

```
      GDS=1.-YS/R2
      F=FS
      G=GS
      GD=GDS
      DO 12 I=1,3
      VS1(I)=(U2(I)*RB-F*U1(I)*RA)/G
      VS2(I)=(GD*U2(I)*RB-RA*U1(I))/G
   12 RO(I)=RA*U1(I)
      CALL OD(RO,VS1,A,XMA,E,XI,XLN,AP,TA)
      RETURN
      END
```

APPENDIX F

SUBROUTINE RV(A,E,XI,XLN,AP,XMO,T,R,TA,U)

This subroutine computes the radius vector for an orbit given the orbital elements and the time.

The input data are:

A	semi-major axis
E	eccentricity
XI	inclination
XLN	longitude of the ascending node
AP	argument of pericenter
XMO	mean anomaly at t_0
T	time at t_1

The output data are:

R	radius magnitude
TA	true anomaly
U	3X1 array of the components of the radius vector

Subroutine RV requires Subroutine ET.

```
      SUBROUTINE RV(A,E,XI,XLN,AP,XM0,T,R,TA,U)
      IMPLICIT REAL*8(A-H,0-Z)
      DIMENSION U(3)
      COMMON/C1/XK
      COMMON/C2/PI
      XM=XM0+T*DSQRT(XK/A**3)
      IF(E.GT.1.)GO TO 2
      EA=XM
    1 DEA=-(EA-E*DSIN(EA)-XM)/(1.-E*DCOS(EA))
      EA=EA+DEA
      IF(DABS(DEA).GE.1.D-12)GO TO 1
      TA=2.*DATAN(DSQRT((1.+E)/(1.-E))*DTAN(EA/2.))
      GO TO 4
    2 EA=DLOG(XM/E+DSQRT(1.+XM/E)**2))
    3 DEA=(E*DSINH(EA)-EA-XM)/(E*DCOSH(EA)-1.)
      EA=EA-DEA
      IF(DABS(DEA/EA).GT.1.D-12)GO TO 3
      TA=2.*DATAN(DSQRT((E+1.)/E-1.))*DTANH(EA/2.))
    4 CONTINUE
      CTA=DCOS(TA)
      R=A*DABS(1.-E**2)/(1.+E*CTA)
      U(1)=CTA
      U(2)=DSIN(TA)
      U(3)=0.
      CALL ET(XLN,XI,AP,U)
      RETURN
      END
```

APPENDIX G

SUBROUTINE VV(A,E,XI,XLN,AP,TA,V)

This subroutine computes the velocity vector for an orbit given the orbital elements and the time.

The input data are:

A	semi-major axis
E	eccentricity
XI	inclination
XLN	longitude of the ascending node
AP	argument of pericenter
TA	true anomaly

The output data is:

V	3X1 array containing the components of the velocity vector

Subroutine VV requires Subroutine ET and also a common statement for the gravitational constant, XK.

```
SUBROUTINE VV(A,E,XI,XLN,AP,TA,V)
IMPLICIT REAL*8(A-H,O-Z)
DIMENSION V(3)
COMMON/C1/XK
STA=DSIN(TA)
CTA=DCOS(TA)
C=DSQRT(XK/(A*DABS(1.-E**2)))
VR=C*E*STA
VT=C*(1.+E*CTA)
V(1)=VR*CTA-VT*STA
V(2)=VR*STA+VT*CTA
V(3)=0.
CALL ET(XLN,XI,AP,V)
RETURN
END
```

APPENDIX H

SUBROUTINE ET(PSI,THETA,PHI,A)

This subroutine transforms from orbital plane coordinates to ecliptic coordinates, given the necessary orientation elements.

The input data are:

PSI longitude

XI inclination

AP argument of pericenter

A 3X1 array containing the orbital plane vectors

The output data is:

A the transformed vectors in a 3X1 array now in ecliptic, rectangular coordinates

Subroutine ET requires only that the 3X1 vector A be dimensioned in the calling program.

```
      SUBROUTINE ET(PSI,THETA,PHI,A)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION A(3),B(3),T(3,3)
      SPSI=DSIN(PSI)
      CPSI=DCOS(PSI)
      STHETA=DSIN(THETA)
      CTHETA=DCOS(THETA)
      SPHI=DSIN(PHI)
      CPHI=DCOS(PHI)
      T(1,1)=CPHI*CPSI-SPHI*CTHETA*SPSI
      T(2,1)=CPHI*SPSI+SPHI*CTHETA*CPSI
      T(3,1)=STHETA*SPHI
      T(1,2)=-SPHI*CPSI-CPHI(CTHETA*SPSI
      T(2,2)=-SPHI*SPSI+CPHI*CTHETA*CPSI
      T(3,2)=STHETA*CPHI
      T(1,3)=STHETA*SPSI
      T(2,3)=-STHETA*CPSI
      T(3,3)=CTHETA
      DO 1, I=1,3
    1 B(I)=T(I,1)*A(1)+T(I,2)*A(2)+T(I,3)*A(3)
      DO 2 I=1,3
    2 A(I)=B(I)
      RETURN
      END
```

APPENDIX I

NOMINAL EARTH-MARS TRAJECTORIES

This program computes the orbital elements for trajectories from Earth to Mars.

The input data are:

J	number of consecutive days computation of orbits
I	initial length mission wanted computed
N	departure day from J.D. 2442800.5

The output data are:

TA	departure day
TB	arrival day
E	eccentricity
A	semi-major axis
XI	inclination
XLN	longitude of the ascending node
AP	argument of pericenter
TA1	true anomaly
DV1	delta velocity necessary for establishing the transfer orbit
DV2	delta velocity necessary for establishing a circular orbit around Mars
DVTOT	DV1+DV2

```
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/C1/XK
      COMMON/C2/PI
      COMMON/C3/A1,A2,E1,E2,XI1,XI2,XLN2,AP1,AP2,XM01,XM02
      COMMON/C4/T1,T2
      COMMON/C9/DE,DM,XK1,XK2
      DIMENSION VA1(3),VA2(3)
      OPEN(UNIT=3,NAME='ORBIT.PRT',DISPOSE='PRINT')
      PI=3.141592654
      TA1=PI
C     TA1 IS AN INITIAL ESTIMATE FOR THE TRUE ANOMALY
      TYPE *,'INPUT TA1'
      ACCEPT 1000,TA1
 1000 FORMAT(F20,10)
      N=1400
C     N IS THE NUMBER OF DAYS FROM THE BASE JULIAN DAY FOR
        THE FIRST ORBIT
      XK1=500000.
C     XK1 IS THE DEPARTURE ALTITUDE ABOVE THE INITIAL PLANET,
        METERS
      XK2=500000.
C     XK2 IS THE ARRIVAL ALTITUDE ABOVE THE SECOND PLANET,
        METERS
      XK3=6378165.0
C     XK3 IS THE EQUATORIAL RADIUS OF THE FIRST PLANET, IN
        METERS
```

```
      XK4=3380000.0
C     XK4 IS THE EQUATORIAL RADIUS OF THE SECOND PLANET,
        IN METERS
      XK1=XK1+XK3
      XK2=XK2+XK4
      TO=2442800.
C     PLANETARY MOTION VALUES FOR PLANETS TAKEN ON THIS
        JULIAN DAY
      A1=1.0
C     SEMI-MAJOR AXIS, PLANET 1
      E1=.016719
C     ECCENTRICITY, PLANET 1
      XI1=0.0
C     INCLINATION, PLANET 1
      XLN1=0.0
C     LONGITUDE OF THE ASCENDING NODE, PLANET 1
      XMO1=18.944
C     MEAN ANOMALY, PLANET 1
      AP1=102.52868
C     ARGUMENT OF PERIHELION, PLANET 1
      A2=1.523691
      E2=0.093383
      XI2=1.84983
      XLN2=49.37285
      XMO2=117.0884
      AP2=286.24549
```

```
      XK=1.32718D20
      DE=3.986D14
C     GRAVITATIONAL CONSTANT, PLANET 1
      DM=4.273D13
C     GRAVITATIONAL CONSTANT, PLANET 2
      I=1
      PI=3.141592654
      CF1=0.01745329252
      CF2=1.4960D11
      CF3=86400.
      XI1=CF1*XI1
      XI2=CF1*XI2
      XLN1=CF1*XLN1
      XLN2=CF1*XLN2
      AP1=CF1*AP1
      AP2=CF1*AP2
      XMO1=CF1*XMO1
      XMO2=CF1*XMO2
      A1=CF2*A1
      A2=CF2*A2
      N=N-1
  112 N=N+1
      M=0
      TYPE *, 'INPUT J,# OF ORBITS WANTED L DAYS > L+J DAYS'
      ACCEPT 80,J
   80 FORMAT(I3)
```

```
      TYPE *, 'INPUT L, INITIAL MISSION LENGTH'
      accept 79,L
   79 FORMAT(I5)
      P=XK1-XK-3
      Q=XK2-XK4
      WRITE(6,250)P,Q
  250 FORMAT('1',5X,'CUT OFF ALT.(MTRS), FROM EARTH=',F10.1,
        5X,'CUT OFF ALT.(M
     XALT.(MTRS),FROM MARS=',F10.1).
      T1=T0+N
      T2=T1+L-1
      WRITE(6,119)
  119 FORMAT(1X,5X,'THE FOLLOWING DATA COMES FROM CALCULATIONS
        FOR AN EARTH XRTH TO MARS ORBIT')
   11 WRITE(6,110)
  110 FORMAT(7X,'DEPART',5X,'ARRIVE',6X,'ECCEN',6X, 'SMA',
        8X,'INCL',3X,'L XONG ASC NODE',2X,'ARG PER',4X,'TRUE
        ANO')
      WRITE(6,111)
  111 FORMAT(7X,'J.DAYS',5X,'J.DAYS',17X,'MTRS',7X,'RADS',9
     1X,'RADS',7X,'RADS')
  222 T2=T2+1
      T1=T0+N
      M=M+1
      T2=CF3*(T2-T0)
      T1=CF3*(T1-T0)
```

```
      CALL HT(T1,T2,TA1,VA1,VA2)
  221 T2=T2/86400.+T0
      IF(M.LT.J)GO TO 222
      IF(N.LT.I)GO TO 112
      STOP
      END
      SUBROUTINE HT(T1,T2,TA1,VA1,VA2)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION U1(3),U2(3),VA1(3),VA2(3),VP1(3),VP2(3),
        VS1(3),VS2(3),
     1W(3)
      COMMON/C2/PI
      COMMON/C3/A1,A2,E1,E2,XI1,XI2,XLN1,XLN2,AP1,AP2,XM01,
        XM02
      COMMON/C5/ALPHA,R1,R2
      COMMON/C9/DE,DM,XK1,XK2
      CALL RV(A1,E1,XI1,XLN1,AP1,XM01,T1,R1,TAP1,U1)
      CALL VV(A1,E1,XI1,XLN1,AP1,TAP1,VP1)
      CALL RV(A2,E2,XI2,XLN2,AP2,XM02,T2,R2,TAP2,U2)
      CALL VV(A2,E2,XI2,XLN2,AP2,TAP2,VP2)
      XN2=U1(1)*U2(2)-U1(2)*U2(1)
      ALPHA=ACOS(U1(1)*U2(1)*U1(2)*U2(2)+01(3)*U2(3)
      IF (XN3.LT.0.) ALPHA=2.*PI-ALPHA
      SALPHA=DSIN(ALPHA)
    1 G1=F(TA1-0.001)
      G2=F(TA1+0.001)
```

```
      DTA1=-0.001*(G1+G2)/(G2-G1)
      TA1=TA1+DTA1
      IF (DABS(DTA1).GE.1.D-12)GO TO 1
      TA2=TA1+ALPHA
      CTA1=DCOS(TA1)
      CTA2=DCOS(TA2)
      E=(R2 -R1)/R1*CTA1-R2*CTA2)
      A=RM*(1.+E*CTA1)/DABS(1.-E**2)
      W(1)=(U1(2)*U2(3)-U1(3)*U2(2))/SALPHA
      W(2)=(U1(3)*U2(1)-U1(1)*U2(3))/SALPHA
      W(3)=XN3/SALPHA
      XI=ACOS(W(3))
      X=DATAN(W(1)/(DSIN(XI)-W(2)))
      IF (X.LT.0.) X=X+PI
      XLN=2.*X
      SLN=DSIN(XLN)
      CLN=DCOS(XLN)
      X=DATAN(-U1(1)*SLN+U1(2)*CLN/((1.+U1(1)*CLN+U1(2)
        *SLN)*
     1DCOS(XI)))
      IF (X.LT.0.) X=X+PI
      AP=2.*X-TA1
      CALL VV(A,E,XI,XLN,AP,TA1,VS1)
      CALL VV(A,E,XI,XLN,AP,TA2,VS2)
      DO 2 I=1,3
      VA1(I)=VS1(I)-VP1(I)
```

```
  2 VA2(I)=VS2(I)-VP2(I)
    VA0=DSQRT(VA1(1)*VA1(1)+VA1(2)*VA1(2)+VA1(3)*VA1(3))
    VAI=DSQRT(VA2(1)*VA2(1)+VA2(2)*VA2(2)+VA2(3)*VA2(3))
    VE1=DSQRT(2*XK1/DE)
    VC1=DSQRT(VAU*VA0+VE1*VE1)
    V01=DSQRT(XK1/DE)
    DV1=VC1-V01
    VE2=DSQRT(2*XK2/DM)
    VC2=DSQRT(VAT*VAT+VE2*VE2)
    V02=DSQRT(XK2/DM)
    DV2=VC2-V02
    DVTOT=DV1+DV2
    TA=T1/86400.+2442800.
    TB=T2/86400.+2442800.
    E=DSQRT(E*E)
    WRITE(6,120) TA,TB,E,A,XI,XLN,AP,TA1
120 FORMAT(6X,F9.1,2X,F9.1,4(1X,D10.3),3X,D10.3,1X,D10.3)
    WRITE(6,130)DV1,DV2,DVTOT
130 FORMAT(6X,'DV1=',F10.3,1X,'M/S',4X,'DV2-',F10.3,1X,
      'M/S',4X,'DVTOT
   X=',F10.3,1X,'M/S')
    RETURN
    END
    FUNCTION F(TA1)
    IMPLICIT REAL*8 (A-H,O-Z)
    COMMON/C1/XK
```

```
      COMMON/C2/PI
      COMMON/C4/T1,T2
      COMMON/C5/ALPHA,R1,R2
      ACOSH(X)=DLOG(DABS(X+DSQRT(DABS(X**2-1.))))
1     TA2=TA1+ALPHA
      CTA1=DCOS(TA1)
      CTA2=DCOS(TA2)
      E=(R2-R1)/(R1*CTA1-R2*CTA2)
      A=R1*(1.+E*CTA1)/DABS(1.-E**2)
      XN=DSQRT(XK/A**3)
      IF (E.GT.1.)GO TO 4
      V=DABS((E+CTA1)/(1.+E*CTA1))
      IF (V.GT.1.)GO TO 4
      EA1=ACOS(E+CTA1)/(1.+E*CTA1))
      IF (DSIN(TA1).LT.0.) EA1=2.*PI-EA1
      EA2=ACOS((E+CTA2)/(1.+E*CTA2))
      IF (DSIN(TA2).LT.0.) EA2-2.*PI=EA2
      XM1=EA1-E*DSIN(EA1)
      SM2=EA2-E*DSIN(EA2)
      IF (TA1.GE.2.*PI) XM1=XM1+2.*P1
      IF (TA2.GE.2.*PI) XM2=XM2+2.*PI
      GO TO 5
    4 EA1=ACOSH((E+CTA1)/(1.+E*CTA1))
      IF (DSIN(TA1).LT.0.) EA1=-EA1
      EA2=ACOSH(E+CTA2)/(1.+E*CTA2))
      IF (DSIN(TA2).LT.0.) EA2=-EA2
```

```
      XM1=E*DSINH(EA1)-EA1
      XM2=E*DSINH(EA2)-EA2
    5 F=T2-T1-(XM2-XM1)/XN
      RETURN
      END
      SUBROUTINE RV(A,E,XI,XLN,AP,XMO,T,R,TA,U)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION U(3)
      COMMON/C1/XK
      COMMON/C2/PI
      XM=SMO+T*DSQRT(XK/A**3)
      EA=XM
1     DEA=-(EA-E*DSIN(EA)-XM)/(1.-E*DCOS(EA))
      EA=EA+DEA
      IF (DABS(DEA).GE.1.D-12) GO TO 1
      CEA=DCOS(EA)
      TA=ACOS((CEA-E)/(1.-E*CEA))
      IF(DSIN(EA).LT.0.)TA=2.*PI-TA
      R=A*(1.-E*CEA)
      U(1)=DCOS(TA)
      U(2)=DSIN(TA)
      U(3)=0.
      CALL ET(XLN,XI,AP,U)
      RETURN
      END
```

```
      SUBROUTINE VV(A,E,XI,SLN,AP,TA,V)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION V(3)
      COMMON/C1.XK
      STA=DSIN(TA)
      CTA=DCOS(TA)
      C=DSQRT(XK/(A*DABS(1.-E**2)))
      VR=C*E*STA
      VT=C*(1.+E*CTA)
      V(1)=VR*CTA-VT*STA
      V(2)=VR*STA+VT*CTA
      V(3)=0.
      CALL ET(XLN,XI,AP,V)
      RETURN
      END
      SUBROUTINE ET(PSI,THETA,PHI,A)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION A(3),B(3),T(3,3)
      SPSI=DSIN(PSI)
      CPSI=DCOS(PSI)
      STHETA=DSIN(THETA)
      CTHETA=DCOS(THETA)
      SPHI=DSIN(PHI)
      CPHI=DCOS(PHI)
      T(1,1)=CPHI*CPSI-SPHI*CTHETA*SPSI
      T(2,1)=CPHI*SPSI+SPHI*CTHETA*CPSI
```

```
      T(3,1)=STHETA*SPHI
      T(1,2)=-SPHI*CPSI-CPHI*CTHETA*SPSI
      T(2,2)=-SPHI*SPSI+CPHI*CTHETA*CPSI
      T(3,2)=STHETA*CPHI
      T(1,3)=STHETA*SPSI
      T(2,3)=-STHETA*CPSI
      T(3,3)=CTHETA
      DO 1 I=1,3
1     B(I)=T(I,1)*A(1)+T(I,2)*A(2)+T(I,3)*A(3)
      DO 2 I=1,3
      A(I)=B(I)
      RETURN
      END
```

APPENDIX J

MINIMIZATION PROGRAM

This program computes the minimum ΔV transfer between two orbits. Time is used as the variable for finding the optimum values.

The input data are:

RR initial estimate for time of transfer

M station for rendezvous

The output data are:

1. the elements of the nominal mission
2. the elements of the transfer mission (optimum)
3. ΔV necessary for both the nominal and transfer missions

```
IMPLICIT REAL*8(A-H,O-Z)
COMMON/C1/XK,DE,DM,XK1,XK2,PI,CF2
COMMON/C2/A3/A4,E3,E4,XI3,XI4,XLN3,XLN4,AP3,AP4,XMO3,
  XMO4
COMMON/C4/JJ,CF3
DIMENSION X(1),R1(3),VP1(3),U1(3)
EXTERNAL F
CALL ASSIGN(1,'SDS.OUT')
CALL ERRSET(73,.TRUE.,.FALSE.,.FALSE.,.FALSE.,31)
CALL ERRSET(84,.TRUE.,.FALSE.,.FALSE.,.FALSE.,31)
CALL ERRSET(81,.TRUE.,.FALSE.,.FALSE.,.FALSE.,31)
CALL ERRSET(72,.TRUE.,.FALSE.,.FALSE.,.FALSE.,31)
DATA PI,N,XK1,XK2/3.14159D00,1400,5.D05,5.D05/
DATA E4,A4,XI4,XLN4,AP4/5,5D-2,1.262D0,9.25D-1,0.D0,
  0.D0/
DATA XK,DE,DM,CF1/1.3271544D20,3.986D14,4.273D13,
  1.7453D-2/
DATA CF2, CF3/1.496D11,8.64D4/
DETERMINE NOMINAL ORBITAL ELEMENTS
DATA A1,E1,XI1,XLN1,XM01/1.D00,1.6719D-02,0.D00,
  0.D00,1.8944D01/
DATA A2,E2,XI2,XLN2,AP1/1.524D0,9.338D-2,1.85D0,
  4.937D1,1.0253D2/
DATA XM02,AP2/1.1709D2,2.8624529D2/
A1=A1*CF2
A2=A2*CF2
```

```
      XI1=XI1*CF1
      XI2=XI2*CF1
      XLN1=XLN1*CF1
      XLN2=XLN2-CF1
      XM01=XM01*CF1
      XM02=XM02*CF1
      AP1=AP1*CF1
      AP2=AP2*CF1
      T1=1400.*CF3
      T2=1600.*CF3
      CALL UV(A1,A2,E1,E2,XI1,XI2,XLN1,XLN2,AP1,AP2,XM01,
        XM02,T1,T2,SXA,
     1SMA1,SE,SXI,SXLN,SAP,DVS1,DVS2,DVSTOT,STA)
      T3=25.*CF3
      A3=SXA
      E3=SE
      XI3=SXI
      SLN3=SXLN
      AP3=SAP
      XM03=SMA1
      WRITE(1,3)
    3 FORMAT(1X,'NOMINAL MISSION ELEMENTS')
      WRITE(1,1)SXA,SE,SXI
    1 FORMAT(1X,'A=',E10.4,'MTRS',3X,'E='E10.4,3X,'1=',
        E10.4,'RD')
      WRITE(1,2)SXLN,SMA1,SAP
```

```
 2 FORMAT(1X,'L,A.N.=',E10.4,'RD',3X,'M.A.=',E10.4,'RD',
     3X,'A.P.=',E1
   10.4,'RD')
    A4=CF2*A4
    XI4=XI4*CF1
    XLN4-SLN4*CF1
    AP4=AP4*CF1
    TYPE *,'INPUT TIME TO RENDEZVOUS,RR (DAYS):'
    ACCEPT 10,RR
 10 FORMAT(F20.10)
    TYPE *,'STATION 1,s,3, OR 4'
    ACCEPT 300,M
300 FORMAT(I4)
    IF(M.EQ.1.)XM04=0.0
    IF(M.EQ.2.)XM04=1.5
    IF(M.EQ.3.)XM04=3.14
    IF(M.EQ.4.)XM04=4.73
    T=RR*CF3
    TOPT=XE(F,T)
    WRITE(1,23)TOPT
 23 FORMAT(1X,'TIME FOR OPT. TRANSFER=',1X,1PD12.5)
    T1=100.0*CF3
    T2=TOPT+T1
    CALL UV(A3,A4,E3,E4,XI3,XI4,XLN3,XLN4,AP3,AP4,XM03,
      XM04,TI,T2,SXA,
   1SMA1,SE,SXI,SXLN,SAP,DVS1,DVS2,DVSTOT,STA)
```

```
      WRITE(1,52)M
   52 FORMAT(1X,'ORBITAL ELEMENTS FOR OPTIMUM RENDEZVOUS
        WITH EARTH
     1 ',1X,I4)
      WRITE(1,53)SXA,SE,SXI
   53 FORMAT(1X,'A=',E10.4,'MTRS',3X,'E=',E10.4,3X,'I=",
        E10.4,'RD')
      WRITE(1,54)SXLN,SMA1,SAP
   54 FORMAT(1X,'L.A.N.=',E10.4,'RD',3X,'M.A.=',E10.4, 'RD',
        3X,'A.P.=',E1
     10.4,'RD')
      WRITE(1,60)DVS1,DVS2,DVSTOT
   60 FORMAT(1X,'DV1=',E10.4,3X,'DV2=',E10.4,3X,'DVTOT=',
        E10.4)
      STOP
      END
C
C
C
C
      FUNCTION F(X)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/C4/JJ,CF3
      COMMON/C2,A3,A4,E3,E4,XI3,XI4,XLN3,XLN4,AP3,AP4,XM03,
        XM04
      DIMENSION X(1)
      T1=100.*CF3
      T2=X(1)+T1
```

```
      CALL UV(A3,A4,E3,E4,XI3,XI4,XLN3,XLN4,AP3,AP4,XM03,
        XM04,TI,T2,SXA,
     1SMA1,SE,SXI,SXLN,SAP,DVS1,DVS2,DVSTOT,STA)
      F=DVSTOT
      RETURN
      END
C
C
C
C
      FUNCTION XE(F,X0)
      IMPLICIT REAL*8(A-H,0-Z)
      XE=X0
    1 F1=F(0.99*XE)
      F2=F(1.01*XE)
      DXE=0.005*XE*(F2-F1)/F1-2.*F(XE)+F2)
      XE=XE-DXE
      IF(DABS(F1-F2).GT.1.0)GO TO 1
      RETURN
      END
C
C
C
C
      SUBROUTINE RV(A,E,XI,XLN,AP,XM0,T,R,TA,U)
      IMPLICIT REAL*8 (A-H,0-Z)
      DIMENSION U(3)
      COMMON/C1/XK
      COMMON/C2/PI
```

```
      XM=SMO+T*DSQRT(XK/A**3)
      IF(E.GT.1.)GO TO 2
      EA=XM
    1 DEA=-(EA-E*DSIN(EA)-XM)/(1.-E*DCOS(EA)
      EA=EA+DEA
      IF (DABS(DEA).GE.1.D-12) GO TO 1
      GO TO 4
    2 EA=DLOG(XM/E+DSQRT(1.+(XM/E)**2))
    3 DEA=(E*DSINH(EA)-EA-XM)/(E*DCOSH(EA)-1.)
      EA=EA-DEA
      IF(DABS(DEA/EA).GT.1.D-12)GO TO 3
      TA=2.*DATAN(DSQRT((E+1.)/(E-1.))*DTANH(EA/2.))
    4 CONTINUE
      CTA=DCOS(TA)
      R=A*DABS(1.-E**2)/(1.+E*CTA)
      U(1)=CTA
      U(2)=DSIN(TA)
      U(3)=0.
      CALL ET(XLN,XI,AP,U)
      RETURN
      END
C
C
C
C
      SUBROUTINE VV(A,E,XI,XLN,AP,TA,V)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION V(3)
```

```
      COMMON/C1/XK
      STA=DSIN(TA)
      CTA=DCOS(TA)
      C=DSQRT(XK/(A*DABS(1.E**2)))
      VR=C*E*STA
      VT=C*(1.+E*CTA)
      V(1)=VR*CTA-VT*STA
      V(2)=VR*STA+VT*CTA
      V(3)=0.
      CALL ET(XLN,XI,AP,V)
      RETURN
      END
C
C
C
C
      SUBROUTINE ET(PSI,THETA,PHI,A)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION A(3),B(3),T(3,3)
      SPSI=DSIN(PSI)
      CPSI=DCOS(PSI)
      STHETA=DSIN(THETA)
      CTHETA=DCOS(THETA)
      SPHI-DSIN(PHI)
      CPHI-DCOS(PHI)
      T(1,1)=CPHI*CPSI-SPHI*CTHETA*SPSI
      T(2,1)=CPHI*SPSI+SPHI*CTHETA*CPSI
      T(3,1)=STHETA*SPHI
```

```
      T(1,2)=-SPHI*CPSI-CPHI*CTHETA*SPSI
      T(2,2)=-SPHI*SPSI+CPHI*CTHETA+CPSI
      T(3,2)=STHETA*CPHI
      T(1,3)=STHETA*SPSI
      T(2,3)=-STHETA*CPSI
      T(3,3)=CTHETA
      DO 1 I=1,3
1     B(I)=T(I,1)*A(1)+T(I,2)*A(2)+T(I,3)*A(3)
      DO 2 I=1,3
2     A(I)=B(I)
      RETURN
      END
C
C
C
      SUBROUTINE UV(A1,A2,E1,E2,XI1,XI2,XLN1,XLN2,AP1,AP2,
        XM01,XM02,T1,T
     12,SXA,SMA1,SE,SXI,SXLN,SAP,DVS1,DVS2,DVSTOT,STA)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/C1,XK,DE,DM,XK1,XK2,PI,CF2
      DIMENSION VS1(3),VS2(3),U1(3),U2(3),VSA1(3),VSA2(3),
        VP1(3),VP2(3),
     1W(3),RO(3)
      T=T2-T1
      CALL RV(A1,E1,XI1,XLN1,AP1,XM01,T1,R1,TAP1,U1)
      CALL VV(A1,E1,XI1,XLN1,AP1,TAP1,VP1)
      CALL RV(A2,E2,XI2,XLN2,AP2,SM02,T2,R2,TAP2,U2)
```

```
      CALL VV(A2,E2,XI2,XLN2,AP2,TAP2,VP2)
      ZNU=DACOS(U1(1)*U2(1)+U1(2)*U2(2)+U1(3)*U2(3))
      UC1=(U1(1)*U2(2)-U2(1)*U1(2))
      IF(UC1.LT.0.0)ZNU=2.*PI-ZNU
      ZNUS=ZNU
      TYPE *,ZNU
      AS=DSQRT(R1*R2)*DSIN(ZNUS)/DSQRT(1.-DCOS(ZNUS))
      Z=0.0
      S=1./6.
      C=0.5
      YS=R1+R2-AS*(1.-Z*S)/DSQRT(C)
      IF(YS.LT.0.0)GO TO 4
      SX=DSQRT(YS/C)
      TS=((XS**3)*S+AS*DSQRT(YS))/DSQRT(XK)
      SP=(1./(2.*Z))*(C-3.*S)
      CP=(1./(2.*Z))*(1.-Z*S-2.*C)
      P=SP-(3.*S*CP/(s.*C))
      QS=3*S*DSQRT(YS)/C+AS/XS
      DTA=((XS**3)*P+(AS/8.)*QS)/DSQRT(XK)
      ZS=Z+(T-TS)/DTS
      SHORT WAY COMPUTATIONS
    8 IF (ZS.LT.0.0)GO TO 16
      SZ=DSQRT(ZS)
      S=(SZ-DSIN(SZ))/DSQRT(ZS**3)
      C=(1.-DCOS(SZ))/ZS
      GO TO 18
```

```
16 SZ=DSQRT(DABS(ZS))
   S=(DSINH(SZ)-SZ)/DSQRT(DABS(ZS**3))
   C=(1.-DCOSH(SZ))/ZS
18 YS=R1+R2-AS*(1.-ZS*S)/DSQRT(C)
   IF(YS.LT.0.0)GO TO 4
   XS=DSQRT(YS/C)
   TS=((XS**3)*S+AS*DSQRT(YS))/DSQRT(XK)
   SP=(1./(2.*ZS))*C-3.*S)
   CP=(1./(2.*ZS))*(1.-ZS*S-2.*C)
   P=SP-(3.*S(CP/(2.*C))
   QS=3*S*DSQRT(YS)/C+AS/XS
   DTS=((XS**3)*P+(AS/8.)*QS)/DSQRT(XK)
   ZS=ZS+(T-TS)/DTS
   U=(T-TS)/DTS
   U=DABS(U)
   IF(U.LT.1.D-12)GO TO 10
   GO TO 8
10 FS=1.-YS/R1
   GS=AS*DSQRT(YS/XK)
   GDS=1.-YS/R2
   F=FS
   G=GS
   GD=GDS
   DO 12 I=1,3
   VS1(I)=(U2(I)*R2-F*U1(I)*R1)/G
12 VS2(I)=(GD*U2(I)*R2-R1*U1(I))/G
```

```
      RO(1)=R1*U1(1)
      RO(2)=R1*U1(2)
      RO(3)=R1*U1(3)
      DO 13 I=1,3
      VSA1(I)=VS1(I)-VP1(I)
   13 VSA2(I)=VS2(I)-VP2(I)
      VSA0=DSQRT(VSA1(1)**2+VSA1(2)**2+VSA1(3)**2)
      VSAT=DSQRT(VSA2(1)**2+VSA2(2)**2+VSA2(3)**2)
      VE1=DSQRT(2.*DE/XK1)
      VCS1=DSQRT(VSA0**2+VE1**2)
      V01=DSQRT(DE/XK1)
      DVS1-VCS1-V01
      VE2=DSQRT(2.*DM/XK2)
      VCS2=DSQRT(VSAT**2+VE2**2)
      V02=DSQRT(DM/XK2)
      DVS2=VCS2=V02
      DVSTOT=DVS1+DVS2
    4 RETURN
      END
C
C
C
C
      SUBROUTINE OD(R,V,A,XMA,E,XI,XLN,AP,TA)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/C1/XK/DE,DM,XK1,XK2,PI,CF2
      DIMENSION R(3),V(3)
      DATANH(X)=DLOG((1.+X)/(1.-X))/2.
```

```
      XR=DSQRT(R(1)*R(1)+R(2)*R(2)+R(3)*R(3))
      SV=DSQRT(V(1)*V(1)+V(2)*V(2)+V(3)*V(3))
      HI=R(2)*V(3)-V(2)*R(3)
      HJ=R(1)*V(3)-V(1)*R(3)
      HK=R(1)*V(2)-R(2)*V(1)
      H=DSQRT(HI*HI+HJ*HJ+HK*HK)
      P=H*H/XK
      RDV=R(1)*V(1)+R(2)*V(2)+R(3)*V(3)
      XVR=RDV/XR
      XVF=DSQRT(XV*XV-XVR*XVR)
      E=DSQRT(1.+(XR*XVF*XVF/XK)*(XR*XV*XV/XK-2.))
      A=P/DABS(1.-E*E)
      W=XV*XV-XK/XR
      EI=(W*R(1)-RDV*V(1))/XK
      EJ=(W*R(2)-RDV*V(2))/XK
      EK=(W*R(3)-RDV*V(3))/XK
      IF(DABS(HK/H).LT.0.99)GO TO 1
      XI=0.0
      IF(HK.LT.0.0)XI=PI
      GO TO 2
    1 XI=DACOS(HK/H)
    2 XNI=-HJ
      XNJ=-HI
      XNK=0.0
      XN=DSQRT(XNI+XNI+XNJ*XNJ)
      XLN=DACOS(XNI/XN)
```

```
      IF(XNJ.LT.0.0)XLN=2.*PI-XLN
      XNDE=XNI*EI+XNJ*EJ+XNK*EK
      AP=XNDE/(XN*E)
      IF(EK.LT.0.0)AP=2.*PI-AP
      EDR=EI*R(1)+EJ*R(2)+EK*R(3)
      TA=DACOS(EDR/(E*XR)
      IF(RDV.LT.0.0)TA=2.*PI-TA
      IF(E.GT.1.0)GO TO 3
      EA=2.*DATAN(DSQRT((1.-E)/(1.+E))*DTAN(TA/2.))
      XMA=EA=E*DSIN(EA)
      GO TO 4
    3 FA=2.*DATANH(DSQRT((E-1.)/(E+1.))*DTAN(TA/2.))
      XMA=E*DSINH(FA)-FA
      4 RETURN
      END
```

VITA

Robert Joseph Vosburgh was born in Dundee, Scotland, on March 15, 1954. He lived there 6 months before returning with his parents to the United States. He attended elementary schools in Iowa, and graduated from Clear Lake Community High School in May, 1972. The following September he entered Iowa State University, and in February, 1977, he received a Bachelor of Science degree in Aerospace Engineering and a commission in the United States Air Force. In March, 1977, he accepted a research assistantship at The University of Tennessee Space Institute and began study towards a Master's degree. This degree was awarded in December, 1977.

The author is a member of Sigma Phi Epsilon Fraternity and the American Institute of Aeronautics and Astronautics. Lieutenant Vosburgh will enter USAF pilot training after graduation.