A SYSTEMS ANALYSIS

OF SPACE INDUSTRIALIZATION

by

David L. Akin

S. B., Massachusetts Institute of Technology (1974)
S. M., Massachusetts Institute of Technology (1975)

Submitted in Partial Fulfillment of
the Requirements for the Degree of

Doctor of Science
in Aerospace Systems

at the
Massachusetts Institute of Technology
August, 1981

Signature of Author ______________________
Department of Aeronautics and Astronautics
July 31, 1981

Certified by ______________________
Rene H. Miller
Thesis Supervisor

Certified by ______________________
Richard Battin
Thesis Supervisor

Certified by ______________________
Jack L. Kerrebrock
Thesis Supervisor

Certified by ______________________
James W. Mar
Thesis Supervisor

Accepted by ______________________
Harold Y. Wachman
Chairman, Department Doctoral Committee

A SYSTEMS ANALYSIS

OF SPACE INDUSTRIALIZATION

by

David L. Akin

Submitted to the Department of Aeronautics and Astronautics on July 31, 1981 in partial fulfillment of the requirements for the degree of Doctor of Science in the field of Aerospace Systems

ABSTRACT

Space industrialization is defined as the use of nonterrestrial resources to support industrial activities which produce a net return on investment. The particular area of interest in this study is the use of the moon as a source of raw materials for production processes in space. Systems analysis is defined as the group of analysis techniques used to calculate both the technical feasibility and the economic viability of a candidate system or process. Following a review of past research in space industrialization, a discussion of likely products for near-term space industrial capabilities is presented. Candidate locations in the Earth-Moon systems are presented, and nine possible locations are selected for inclusion in this study. Velocity increments between orbital locations are found, including the estimation of plane change requirements, and an analysis of the effect of nonimpulsive and continual thrusting trajectories on ΔV requirements is made. A multiconic technique is used for accurate trajectory analysis between the earth and the moon, and for flights to libration points.

With all locations specified, individual system analyses are performed to optimally size transportation vehicles. Earth launch vehicles are parametrically analyzed for the cases of one and two stages to orbit, with internal reusable and external expendable propellant tanks. The production system is defined, and broken down into component processes of mining, refining, manufacturing, and assembly. Estimates of significant parameters are made for each step.

Costing algorithms are derived, and costing performed on integrated production systems. Candidate production scenarios, comprising the most cost-effective alternatives for a total program systems identified in earlier sections, are collected for later optimization.

One of the primary contributions of this work is the development of a new operations research technique, which can be used to solve a variant of the "knapsack" problem without resorting to classical integer programming. The problem addressed is one of optimal investment: with limited resources, optimize the distribution of resources among a set of competing systems over a length of time so as to maximize the net return over the lifetime of the program. The new solution technique, diagonal ascent linear programming (DALP), is derived and shown to produce a heuristic optimum for a sample problem of solar power satellite production. The technique is applicable to problems of competing systems which can be categorized by recurring and nonrecurring costs and returns: the returns may be either lump sum or recurring. The product of the analysis is a heuristic optimum solution for best investment of a resource in a year-by-year manner over recurring and nonrecurring costs of each system, in order to maximize the net return of the entire production scenario over its operational lifetime. Systems will be funded only if they contribute favorably to the return on investment, and multiple systems will be phased by finding the initial operational dates for each system which will maximize the value of the objective function. Examples use cost criteria as the objective function: net present value functions are included in the analysis procedure for these cases. The DALP analysis technique is used to find near-optimal stategies for competing solar power satellite production schemes; to find the best choice of vehicle configuration for a space shuttle based on the information available to NASA when the decision was made in 1973; and to optimize the selection of upper stages to use with the current Space Transportation System. In addition, further refinements of the solution algorithm are described which take into account such higher-order effects as commonality between competing systems and learning curve effects on recurring costs. Conclusions are drawn on the original question of nonterrestrial materials utilization, and on the use and applications of the diagonal ascent linear programming technique derived in order to analyze the original problem.

Thesis Supervisor: Rene H. Miller
Slater Professor of Flight Transportation

Thesis Supervisor: Richard H. Battin
Adjunct Professor of Aeronautics and Astronautics

Thesis Supervisor: Jack L. Kerrebrock
MacLaurin Professor in Aeronautics and Astronautics

Thesis Supervisor: James W. Mar
Hunsaker Professor of Aerospace Education

ACKNOWLEDGEMENTS

It is not possible to spend ten years at one place, six of them working on (or at) a doctoral thesis, without being indebted to a large number of people. Were I to recount all the friendships along the way, I would have to have to have a separate volume for the acknowledgements: so, I would like to thank all my friends *en masse,* apologize to those I overlook here, and specifically acknowledge those without whom the past years would not have been possible.

First of all, I would like to thank all those who have served on my thesis committee since its inception. If any one single person could be said to be responsible (in the best possible sense) for this thesis, it would have to be Prof. Rene H. Miller, who was willing to extend a research assistantship to a new doctoral student interested in designing space stations instead of MX guidance systems. His support, counseling, guidance, and friendship over the years has made this work possible. Prof. Jack L. Kerrebrock always made sure my academic progress was adequate, and deserves credit as well for being the only other member of my committee (besides Prof. Miller) who endured from the beginning to the end of this effort. I would like to thank Prof. John McCarthy, who saw to it that I eventually learned the first law of thermodynamics, Prof. Richard Battin, orbital mechanics mentor *extraordinaire,* and Prof. James Mar, who was nice enough not to ask a question at my general exams. I am also in debt to Dr. Phil Chapman, for several interesting conversations which helped to frame the initial direction of this thesis. I would especially like to thank Prof. Amedeo Odoni, who provided invaluable assistance on the operations research chapter.

I would like to acknowledge the memory of two friends who did not live to see this thesis completed. Dr. Ted Edelbaum was one of the original members of my thesis committee, and largely responsible for the orbital mechanics section of this work. And Fred Merlis, technical instructor, whose highest title was that of friend.

There have been many people along the way who, while not directly involved with this thesis, have meant a great deal to my ability to stay sane and finish this work, and I would like to thank them all briefly. If the first place on a list is a place of honor, then that position would have to go to Mr. Al Shaw, who kept me from slicing my fingers off on the milling machine as a freshman, and who has been instructor, confidant, and friend ever since. I would like to thank David B. S. Smith, who bet me that I could not work the word "brontosaurus" into my thesis (you just lost, Dave!), for the use of his Apple computer, on which a large portion of the orbital mechanics was programmed. I want to acknowledge and thank all of the members, past and present, of the Lab of Orbital Productivity, (better known as the Loonie Loopies), without whom this thesis would have been completed three years ago. I am especially indebted to Dave Mohr, who helped me through the ordeal of a doctoral language exam. I apologize to all of you for being such a grouch over the last six months while finishing this thesis.

Acknowledgements are due to the authors from whom I borrowed the quotes at the beginning of each chapter: Konstantin Tsiolkowsky, George Lucas, Arthur C. Clarke, Walter Lippmann, and especially Carey Rockwell, whose *Tom Corbett, Space Cadet* books were responsible for awakening my interest in space at the age of five.

I would especially like to thank my adopted second family, the Bowdens, for their support over the last few years. Stella, Becky, and, of course, my clone Marc (Margaret) kept me supplied with cookies and good cheer over the course of writing this thesis, and are the best sisters that anyone never had.

Finally, there are three people for whom no verbal or written words are sufficient to express my thanks and admiration. They have made me who and what I am, and in a small measure of repayment, I would like to dedicate this thesis to my Mother and Father, June and Thomas Earl Akin, and to Mary Bowden, for their help, support, and love.

CONTENTS

LIST OF ILLUSTRATIONS

1.0 INTRODUCTION

> *"There is one important lack in the space between Earth and Moon which we have opened up for colonization: the absence of sufficient quantities of materials for building and other social purposes."*
>
> *"Getting materials from Earth is costing far too much", Laplace agreed.*
>
> *"They could get it from the Moon," observed Franklin. "That would cost 22 times less. But the Moon is awkward to live and work on, as Ivanov and Nordenskjold explained after they'd been there...."*
>
> *- Konstantin Tsiolkowsky*

Although written 65 years ago, the situation in space industrialization is still summed up well by the dialogue between the characters in Tsiolkowsky's novel, Beyond the Planet Earth[1]. In order to develop and expand in the region of space between the earth and the moon, resources are required with which to expand. The free-space environment offers a number of advantages, such as unlimited solar energy, high vacuum, and weightlessness. Balanced against these advantages are the problems of materials supply and transportation. The techniques for mining ores, refining pure metals and alloys, manufacturing parts, and assembling machines and structures are all well known and understood, in so far as they apply to the surface of the earth. What is not understood is how these processes apply in space: to what extent Earth technology is applicable, what new techniques must be found, what innovative new technologies are required. And even before the production techniques, the question arises as to materials sources available. As Tsiolkowsky pointed out, it is difficult to transport materials off of the surface of the earth; getting mass into space from the moon is 22 times cheaper, if the actual energy change of the material were the only criteria.

With the increase in spaceflight capabilities represented by the first flight of the space shuttle Columbia, it is important to consider the goals and possibilities of the space program. It seems clear that the pursuit of technology for its own sake is not a viable option, at least not at this time in this country. Politically motivated goals, such as the decision to send men to the moon, succeeded in rapidly developing the capability for space flight; but these capabilities were allowed to deteriorate after the goal was reached. Lacking the mechanism for assuring long-term governmental support for a vital space program, the private sector is the logical place to look for long-term applications of current capabilites in space.

It is interesting to look at the current state of space flight in the historical perspective of air transportation. Due to the scale of effort required, space flight has lacked the "barnstorming" days of private individuals who flew around the country, making a meager living and introducing the populace to the concept of flight first-hand. Instead, space flight was introduced through the medium of television; interest lasted about as long as that for any new situation comedy or detective show. Both fields were encouraged by governmental support: much of the early aircraft development, such as the cantilever monoplane, was encouraged by the incentive offered by government mail routes.[2] However, when the government decided that it should not directly be involved with transporting mail, further aircraft development was dependent on two sources: military and commercial. With the development of passenger traffic on a scheduled basis, both domestic and international, air transport reached a level of "respectability" which encouraged private investment. It might be argued that the space program is about to enter the same stage of development: if not, then nothing of consequence will happen for some time in space, at least for the U.S.

In a study of space industrialization, that is, the development of industrial capability for both the government and private sectors in space, some care should be given to defining the terms and limitations of the items under investigation. It is not sufficient to say that expansion in space "should" take place, or even to show that the technical capability exists: the engineering capability is a necessary criterion, but not a sufficient one. In addition to the purely technical analysis, an economic analysis must be done. It will not be enough to show that something can be done in space; it must also be shown that it can be done better, more cheaply, more efficiently in space, or it will not be done at all. Based on these arguments, it is possible to specify the definitions of the two major themes of this work:

Systems Analysis The use of quantitative analysis techniques to determine both the technical feasibility and the economic viability of a process, system, or program.

Industrialization (Economic) Industrial activities in space which have return on investment as the primary motivation. Of particular interest in this study is the use of nonterrestrial materials to support such industrial processes.

(Functional) The use of devices which must be processed in space before becoming operative or valuable.

1.1 A BRIEF OVERVIEW OF SPACE INDUSTRIALIZATION

As demonstrated by the quote which introduced this chapter, much of the early work in this field was either science fiction, or technical work disguised as science fiction. One significant milestone in the pre-Sputnik era was The Man Who Sold the Moon, by Robert A Heinlein.[3] This novel described a first flight to the moon, financed by private corporations which bankrolled the research and development as an business investment. Much of the action of the novel involved the activities of the protagonist, who supported the moon flight on philosophical grounds and yet had to sell it to businessmen as a sound financial investment.

Much of the detailed work on actual engineering fundamentals was performed during the Apollo program. Most of the hardware needed for transportation in an industrial scenario in space went through one or more design iterations in the 1960's. Although the technical aspect was well addressed, almost all of these studies assumed that philosophical (i.e., exploration) or political (national prestige) issues formed the rationale for further space flight development, and little information is available on industrial potentials of programs such as the space station designs of this decade.

With the demise of much of the space program in the early 1970's, attention became focused on the near-term (and near to hand) applications of space technology. The Skylab program provided demonstrations of many of the technologies necessary for space industrialization, including materials processing, zero-g welding, and extravehicular capabilities.[4] In addition to the technical aspects, the three Skylab missions also demonstrated the capability of people to live and work productively in weightlessness for prolonged periods of time without ill effects.

The first examples of what might be legitimately called space industrialization dealt with the manipulation of physical materials. Suggested areas of interest included such fields as pharmaceuticals and semiconductors: products of this sort were typically of high enough intrinsic value that the added burden of launch and retrieval costs did not significantly affect the net cost of the finished product. The first proposal of large-scale services from space, rather than materials, was the idea of the satellite solar power station (later renamed the solar power satellite or satellite power station, abbreviated SPS)[5]. Dr. Peter Glaser proposed as an answer to the as yet unrecognized energy crisis that large arrays could be placed in orbit around the earth, converting sunlight into electricity and sending it via a microwave link to the surface of the earth. This was probably the first service satellite which might be considered as a space industrial product: unlike communications satellites, which are launched in operational configuration, the SPS would have to be assembled on-orbit on a scale that would require a substantial operations base and logistics support. In addition to satisfying the functional definition of space industrialization, it also satisfies the economic one: it is definitely a system with the objective of making a profit. Due both to its history as one of the first space industrial products proposed and the scale of operations required, it is fitting that the SPS should be one of the projects considered in this work.

Development of the SPS concept occured during the early 1970's. Independent study efforts by Boeing (sponsored by NASA Johnson Space Center) and Rockwell International (sponsored by NASA Marshall Space Flight Center) produced point designs for the SPS itself, as well as production system and transport system designs. This work was combined as part of a three-year study by the Department of Energy[6] into an SPS baseline design[7], which will also be used as the baseline for this study.

One peculiarity of the DOE baseline design is that it is in reality two designs: both the Boeing and Rockwell concepts were incorporated into the final baseline. The Rockwell design uses gallium aluminum arsenide (GaAlAs) solar cells with concentrators to double the energy flux at the cells. The Boeing design uses silicon cells without concentration. Since one of the objectives of this study is to look at the use of nonterrestrial materials for SPS, only the Boeing design will be considered here further, as it is more amenable to substitution of materials commonly found on the moon.

The critical SPS parameters used in this study are mass and power output. For the baseline DOE SPS, the mass is 40,000 metric tons, to supply 5 GW of electrical power at the ground.[7] An M.I.T. study of a space manufacturing system investigated the possible uses and substitutions of lunar material into the baseline SPS design, and concluded that a system which delivered 10 GW of power would have a mass of 100,000 tons if lunar materials were used.[8] Since the size of the SPS differs, a more direct comparison shows that an SPS built of lunar materials would have a specific mass of 10 kg/kw, 20% greater than the specific mass of the earth baseline.

While expansion into space may be philosophically attractive, the economic justification for such a step is not apparent. Therefore, it is important to compare not only terrestrial and nonterrestrial sources for SPS, but also to compare these systems to conventional systems, such as fossil fuel and nuclear generating plants. A 1974 Ford Foundation report[9] identified the capital costs in 1970 for fossil and nuclear plants as $185/kw and $325/kw, respectively. Applying an average inflation factor of 10% gives the 1981 equivalent costs as $528/kw and $927/kw. To this must be included the fuel costs, which are difficult to find accurate numbers for. The rate structure of Cambridge Electric Light Company as of April 2, 1981 has a median cost of $0.05/kwhr for residential use[10]. This corresponds to a rate of $438.30/kwyr. Assuming a 30-year life for conventional power sources and that the fuel costs are 10% of the retail costs, the equivalent capital cost for fuel over the lifetime of the power plant is $1315/kw. Since the SPS operates without fuel costs (and, ideally, with minimal maintenance costs), it must be compared to the cost of a conventional power plant with life-cycle costs for fuel, which would be equal to $1843/kw for a fossil-fuel plant.

When discussing a project of the scope of the solar power satellite, the possible use of nonterrestrial materials becomes of primary importance. Although the SPS is the first system to be considered as an industrialization product in this work, the first proposed use of nonterrestrial materials in space was for a much larger scale of project: the space colony. This was (like much else) first proposed by Tsiolkowsky in [1]. An independent rediscovery of this idea was made by O'Neill in the context of teaching a college class[11], and was originally proposed as a solution to overpopulation. Only in his second publication[12] did O'Neill link space colonies to energy production, as a space base for the personnel required to produce the SPS. Follow-on work at NASA Ames Research Center[13] and an independent assessment at M.I.T.[14] studied the details of habitat design in greater detail; later work at Ames[15] placed greater emphasis on the problems and techniques for lunar mining and refining. It is in this context that space colonies enter into this effort: it is interesting to note that large permanent habitats in space do not meet the economic definition of space industrialization, as there is no real market for them. Studies at M.I.T.[16] have shown that production of a closed cycle artificial gravity habitat is not cost effective in support of an SPS program for any reasonable SPS production rate.

In order to model the processes of nonterrestrial materials usage, it is necessary to define the component steps in the program:

Mining	The collection of raw ores on the lunar surface
Refining	Processing the ores to form feedstocks
Manufacturing	Using the feedstocks to produce component parts
Assembly	Putting the components together to form the finished product

Each of these steps will be discussed and applicable parameters found, for use in later sections of this study.

Mining on the moon consists largely of using lunar equivalents of a bulldozer and a dump truck. Apollo results indicate the moon is largely homogeneous, and average lunar soil seems to contain most of the elements necessary for industrial processes. Although this assumption would change if detailed lunar geological surveys revealed ore concentration areas, the detailed study of lunar mining operations in [15] will be used as representative of the class of lunar "strip mining" operations. Much of the system is designed for the high production rates of intensive habitat construction operations, and is therefore scaled for an order of magnitude more throughput per year than is necessary for SPS construction. For this reason, the smaller mining system proposed for the first year of colony production will be assumed as the baseline throughout SPS production. This system ships 30,000 metric tons of unbeneficiated lunar ore per year, with a mass of 12 tons for mining equipment, which also consumes 102 kw during operation. The 30,000 tons per year is based on a collection rate of 8 tons/hr, and a work year of 3700 hours. The crew size for operation is estimated in this reference to be 10 people/shift, or 30 people overall. These values will be scaled linearly in this study with the required mining output.

Refining is also addressed in reference [15]. Preliminary beneficiation is assumed to be done on the moon: this would convert the materials from native soil to concentrated plagioclase and ilmenite. One beneficiation module, based on 190 tons/hr and a usage rate of 3700 hours/yr, would mass 77 tons and use 272 kw of electrical power. The processing plant in the same study has an estimated mass of 230 ton, while producing 60,000 tons of feedstock material per year. A powerplant mass for this unit is listed as 415 tons; using the assumption listed elsewhere in this reference that powerplant specific mass is 8 kg/kw, this would give a power requirement of 52 kw. The throughput fraction, or mass of output material divided by mass of inputs, is .2 for the beneficiation step and .6 for refining beneficiated ores. Since these steps are sequential, the net throughput for refining from ores to feedstock is 12%. Productivity at this step is estimated in the reference to be 100 kg/crew-hr.

The topic of space manufacturing was addressed in detail in an M.I.T. study for NASA Marshall Space Flight Center[8]. The design reference mission is directly applicable to this analysis: the space manufacturing facility (SMF) had to produce the components for 1 SPS per year. This study performed a detailed part-by-part breakdown of the SPS, and concluded that 95% of the SPS was replaceable with lunar parts. The other 5% had to come from earth, and included both alloying elements not found on the moon and components needed in quantities too small to justify manufacturing them in space. For one 10,000 ton SPS produced per year, the total production machinery mass was 9448 tons, power requirement was 232 mw, and direct production crew was 216 people, assuming 8000 hrs/year of activities at the SMF. This report is also the source of the cost estimation factors used in this study, as summarized in Table 1-1. Values in this table have been corrected with a 12% inflation factor to convert them from 1979 to 1981 dollars. In an effort to more accurately estimate the costs of a system, it may be broken down into components, and the components costed on the basis of their estimated technology level. "Low" refers to static structures, such as propellant tanks. "Medium" refers to flight-critical structures, such as wings. "High" technology would be such things as crew cabin furnishings and power units. "Ultrahigh" technology devices are those with a substantial amount of electronics, such as flight control systems.

Assembly of large structural components in space has been studied in the M.I.T. Space Systems Laboratory for some time. Results from neutral buoyancy tests indicate that productivity can be as high as 800 kg/crew hour for short periods of neutral buoyancy testing[17]; and for prolonged assembly runs in A7LB pressure suits underwater, demonstrated productivity was still above 500 kg/crew hour[18]. In addition, evidence indicates that there is an instinctive adaptation to the weightless environment after 10 to 15 hours of assembly experience, and that after this length of time the assembly worker actually uses the zero-g environment to speed up the assembly procedure. Preliminary results with machine augmentation[18] indicates that very little mass of assembly aids need be flown for assembly: total mass of assembly aids might be 3-5 times the mass of the assembly worker.

Technology Level	R&D Cost	First Unit Procurement Cost
Low	625	63
Medium	6,250	625
High	25,000	2,500
Ultrahigh	125,000	12,500

All costs in $/kg

Table 1-1: Cost Estimation Parameters

Table 1-2 summarizes all of the data presented above, arranged in terms of the independent parameters used in the analysis algorithms of a later chapter. These parameters are estimates only, and can be varied to find the sensitivity of the solution to the accuracy of the estimates.

Of special note is a study done by General Dynamics - Convair Division for NASA Johnson Space Center, which is an overall systems analysis of space industrialization. One of the first conclusions of this study was that no program short of SPS is a viable candidate for use of nonterrestrial materials. The General Dynamics study identified a set of scenarios for solar power satellite production, differing primarily in transport options for getting off of the lunar surface. The primary result of this study was a point design for a lunar resources utilization system, and a set of costing curves similar to those of reference [19]. Little was done on the analysis of

Parameter	Mining	Refining	Manufacturing	Assembly
Crew Productivity (kg/crew-hr)	800	100	175	500
Machine Productivity (kg/kg-hr)	0.67	0.0165	0.00132	1
Power Requirement (kg/kw-hr)	80	120	0.05	1
Throughput Fraction	1	0.12	0.8	1
Earth Source Fraction	0	0	0.05	0.05

Table 1-2: Parameter Estimates for the Production System

---

earth launch or orbit-to-orbit transportation systems, and no rigorous analysis technique was developed for deciding between scenarios chosen. The assumption that only the SPS is suitable for lunar materials also merits further investigation.

1.2 THESIS PREVIEW

In the following chapter, the various possibilities of locations in the earth-moon system are discussed. Non-ideal effects on two-body motion from such sources as noncoplanar transfer, nonimpulsive thrust, and continuous thrust are analyzed, and their impact on system performance found. Patched conic techniques are used as initial estimates of spacecraft insertion maneuvers into trajectories between the earth and the moon: multiconics are used with a universal variable formulation of the two-body problem to find accurate three-body trajectories. The final result of this chapter is the selection of 9 candidate locations for industrial processes, along with accurate ΔV requirements for transport between them.

Chapter 3 deals with "classical" systems analysis: for a single system such as an earth launch system or orbital transfer system, independent parameters are chosen so as to optimize an objective function, such as system cost. A detailed parametric model of earth launch systems is presented, and details of vehicle configuration (one or two stages, reuseable or expendable propellant tanks) are considered. Following this, the various options for propulsion systems for interorbital transportation and lunar launch are similarly analyzed.

Chapter 4 represents the drawing together of the individual systems of Chapter 3 (which were sized by the requirements of Chapter 2) into an overall production scenario. Using the program estimation algorithms derived in this section, choices of transportation systems or production parameters may be varied to find the effect on the final cost of whatever product is specified. The output of this chapter is a set of "most attractive" production scenarios, which will be used as input to the next chapter.

Chapter 5 presents one of the most important contributions of this study, which is the development of a new technique in operations research. One of the biggest problems in overall program optimization is that of competing systems. Each system has an initial, nonrecurring cost which represents research and development and initial procurement. Each system also has a production cost for the units which it is producing. Finally, there is the expected return of the produced units, which may be either lump sum ("turnkey") or recurring (such as yearly sales of services). In a situation where investment capital is limited (such as the real world), the problem to be addressed is how to spread the limited resources over the systems in order to maximize the return. Provision must also be made for totally discarding a system if it is not effective in the objective function: this introduces an "existence" cost, where R&D must be paid if (and before) a system is used, and not paid otherwise. In addition, if returns on early units may be reinvested in the program, the possibility arises of optimally phased multiple systems. For example, a system with low nonrecurring costs may be used initially to gather returns, which are used for the R&D of a second-generation system with lower recurring costs or higher revenues. All of these factors are addressed in the derived technique, which has been called diagonal ascent linear programming (DALP).

Finally, Chapter 6 is a summary of the entire work, and is followed by appendices on two-body orbital mechanics, market research into the current shuttle mission model (a sidelight of Chapter 3), and listings and sample outputs of computer programs used in this study.

2.0 FLIGHT MECHANICS

"Travelling through hyperspace ain't like dusting crops, boy! Without precise calculations we'd fly right through a star or bounce too close to a supernova, and that'd end your trip real quick, wouldn't it?"

- Han Solo

2.1 INTRODUCTION

In analyzing the feasibility of space industrialization, it should come as no surprise that transportation issues tend to dominate the criteria for overall system viability. Until recently, in fact, transportation set real, physical limitations on activities in space, due to a limit in total kinetic energy which could be imparted to a payload. With the advent of the space shuttle, multipayload, multistage vehicles become technically possible, which would increase velocity change capabilities ("ΔV") to a range suitable for extensive cislunar operations. Whether these operations are economically viable or not must await a detailed analysis of the energy requirements for transfers within the system of interest.

As discussed earlier, the primary thrust of this work is to quantify the effect of nonterrestrial materials usage on the technical and economic feasibility of space industrialization. Although much of the solar system appears attractive from the materials resources point of view, this thesis will restrict its focus to materials available in the earth-moon system. Within cislunar space, nine generic volumes of space appear to be of primary interest:

- Earth surface (ES)
- Low Earth Orbit (LEO)
- Intermediate Earth Orbit (IEO)
- Geosynchronous Orbit (GEO)
- High Earth Orbit (HEO)
- Lunar surface (LS)
- Lunar Orbit (LO)
- Unstable Lagrange points (L1,L2,L3)
- Leading and Trailing Lagrange points (L4,L5)

In any reasonable scenario involving lunar materials, there will always be some materials requirements which can only be met with terrestrial materials. In addition, there will always be the requirement for crew rotation and logistics support, involving earth launch and landing. The earth's surface, therefore, will remain an important location in space for some time to come. Since earth launch will be shown to be the most energy intensive of all transfer maneuvers, it will be important to have a transfer station in LEO, at which point crews and cargos can be offloaded onto transport vehicles more suited to the space environment than launch vehicles are. The ideal location for this facility would be in an orbit with the lowest possible altitude at which the station can remain without excessive orbit make-up fuel requirements due to atmospheric drag. Recent work by Boeing on the Space Operations Center [20] indicates that an orbital altitude of 370-400 km would be preferable. Although this figure is based in part on the payload capability of the present shuttle, it is probably reasonable to assume that it would be advantageous to maintain commonality between present launch systems and those dedicated to advanced industrialization projects. For this reason, the radius of LEO used throughout this report will be 6750 km, which corresponds to a circular orbit of 372 km altitude. Due to the additional velocity reserves needed for injection into an equatorial orbit from a nonequatorial launch site, the orbital inclination of LEO will be assumed to be the same as that of Cape Canaveral, 28.5°. This will allow maximum payload capability to LEO, and conversion to desired final inclinations at lower ΔV surcharges.

Geosynchronous orbit (GEO) refers to that class of earth orbits which have an orbital period identical to the rotational period of the earth, and thus repeat their apparent groundtrack at

24-hour intervals. Although several different groundtrack patterns are possible [21], of primary interest is the special case of a circular zero-inclination geosynchronous orbit, which remains continually over a single point on earth. This is the geostationary orbit, and this report will assume that GEO refers to geostationary, unless otherwise noted. Intermediate orbit (IEO), then, refers to an orbit with radius between those of LEO and GEO, and high orbit (HEO) refers to an orbital radius greater than that of GEO. In general, all four of these appellations (LEO, IEO, GEO, and HEO) refer to circular orbits in the corresponding altitude range. In addition, it should be noted that "radius" will refer throughout this report to the distance of the orbit from the center of the planet, and "altitude" to the distance from the planet's surface. Thus, GEO has an orbital radius of 42164 km, and an orbital altitude of 35786 km; the two figures differ only, as one might expect, by the earth's radius of 6378 km.

The lunar surface is the origin for all nonterrestrial materials used in the scenarios assumed for this thesis. As such, it is similar to the earth surface as a terminal for large mass flows. For impulsive thrusting (standard rockets), the initial destination for cargoes of lunar origin is lunar orbit. Again, it is advantageous to transfer payloads to vehicles designed for interorbital transportation, so that no penalty is incurred from carrying between orbits equipment which is peculiar to the lunar landing and launch mission. Unlike earth orbit, no difference is drawn between low, intermediate, stationary, and high altitudes. Since the moon lacks an atmosphere, the minimum orbital radius is one which would insure that the vehicle does not intercept a mountain top: this is approximately 10 km. Selenostationary radius is 300,000 km, and at this distance would clearly be an unstable orbit due to perturbations from the earth and the sun. In fact, the maximum altitude which can be referred to as lunar orbit is dependent on the allowable earth perturbations. Due to these criteria, a single lunar orbital altitude will be considered; the exact value for this variable will be picked on the basis of a parametric analysis.

The locations of the Lagrange points are shown in Figure 1 on page 32. These are the five equilibrium points in the restricted three-body problem, as applied to the earth-moon system. It should be noted that the notation in Figure 1 on page 32 follows that in NASA SP-413 [13]; Kaplan [22] and others reverse the definitions of L1 and L2. At any rate, points L1, L2, and L3 represent points of unstable equilibrium, and continual station-keeping must be performed to maintain a vehicle at these locations. L3 will not be considered as a potential worksite in this study, as it is a long way from everywhere of interest. L1 has the advantage that it is in continual contact with both the moon and earth, but launches to

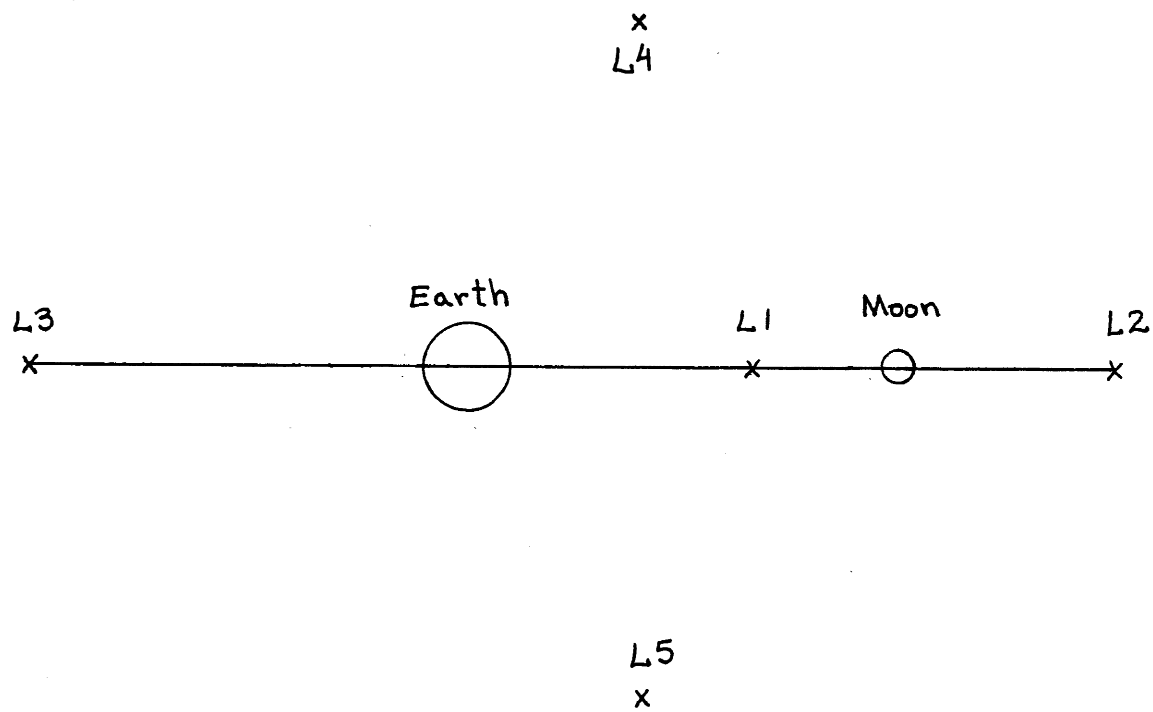

Figure 1. Lagrange Points in the Earth-Moon System

L1 must take place from the back side of the moon. If a mass driver system is used [23], the lunar mining base would have to be located out of direct communication with earth. L2 can be targeted from a launch site on the front side of the moon, but L2 itself can only see the limb of earth around the moon, and must depend on lunar polar or other relays for reliable communications. Lagrange points L4 and L5, leading and trailing the moon in its orbit by angles of 60 degrees, are the two stable points in the earth-moon system. These are the locations most often suggested for refining and manufacturing processes using lunar ores.

2.2 TWO-BODY ANALYSIS

Well within the sphere of influence of the earth, spacecraft maneuvers can be calculated without reference to the perturbing effects of lunar gravitation. As will be shown later, two-body techniques can be used to estimate velocity changes required for earth-moon transit trajectories as well.

2.2.1 HOHMANN TRANSFERS

The mechanics of Hohmann transfers are quite well known, and are formally laid out in reference [54]. The equations used in this report for Hohmann co-planar trajectories are described in "Hohmann Transfers" on page 191. For the purposes of comparison, it would be interesting to find the variation in velocity increment required with changing orbital altitude of the target orbit. From this, the effect of such considerations as noncoplanar and nonimpulsive transfers may be clearly seen. Choosing to nondimensionalize the transfers by the use of parameters ρ (radius of initial orbit/radius of final orbit) and ν (ΔV of transfer/circular velocity in initial orbit), the effect of this variation may be seen in Figure 2 on page 35. Of interest in this figure is that, for values of ρ below 0.3, the ΔV requirement for orbital transfer exceeds that of simple escape. The values of ρ for transfer to GEO and lunar altitudes from LEO are marked on the figure; both require velocity changes greater than escape velocity from LEO.

2.2.2 NONCOPLANAR TRANSFERS

As noted earlier, not all of the possible base locations are in coplanar orbits. For example, LEO is assumed to be at an orbital inclination of 28.5° in order to maximize launch capabilities, while GEO is of necessity at 0° inclination for geostationary orbit. Thus, additional ΔV is required to perform a plane change maneuver. Since a plane change is equivalent to rotating the velocity vector at the apsis of the orbit through the required angle, it seems apparent that the minimum fuel penalty is incurred if the plane change takes place at minimum velocity: that is, at the apoapsis. However,

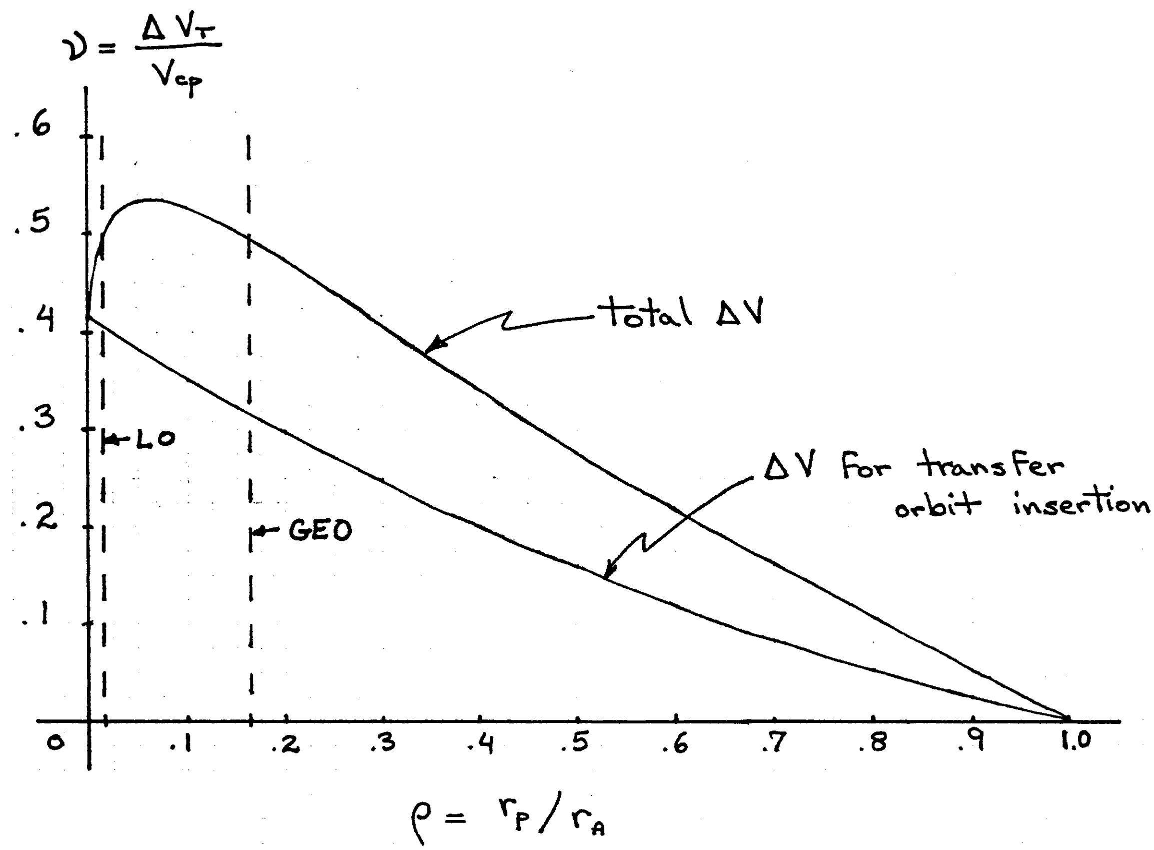

Figure 2. Characteristics of Hohmann Transfer Orbits

---

small plane changes at periapsis injection incur small penalties, and lead to more favorable velocity requirements at the circularization burn. It would therefore be desirable to distribute the plane change in an optimum manner between the two burns in the Hohmann transfer.

In order to perform this optimization, it is desirable to have an analytical expression for total ΔV in terms of the plane change angles in the two maneuvers. While such an expression is computationally messy, some simplifying assumptions lead to a straightforward formulation of approximate ΔV, which will allow the estimation of optimal plane change strategies. This approach is outlined below.

Figure 3 illustrates the geometry of the velocity vectors at the initial maneuver point. Since the assumption has been made, based on heuristic analyses with the exact equations, that the initial plane change angle is small, the line forming the base of the isoceles triangle is $v_{c1}\ \delta$. The angle opposite Δv_1 in the remaining triangle is $90+\ \delta/2$. Using the law of cosines, Δv_1 can be found to be

$$(2.1) \qquad \Delta v_1 = \sqrt{(v_1 - v_{c1})^2 + v_{c1}^2\,\delta^2 - 2\,v_{c1}\,\delta\,(v_1 - v_{c1})\cos\left(90 + \frac{\delta}{2}\right)}$$

If δ is assumed to be small, $\cos(90+\ \delta/2)$ can be neglected, and Δv_1 can be estimated using Pythagoras' theorem:

$$(2.2) \qquad \Delta v_1 = \sqrt{(v_1 - v_{c1})^2 + v_{c1}^2\,\delta^2}$$

Figure 4 on page 37 shows the geometry at the second, circularization burn. Since the angle θ, representing total plane change angle, is not in general small, the simplification of the analysis deals with estimating the difference in Δv_2 between the pseudo-optimal case and the reference case where all plane change is incorporated in the second maneuver. Two

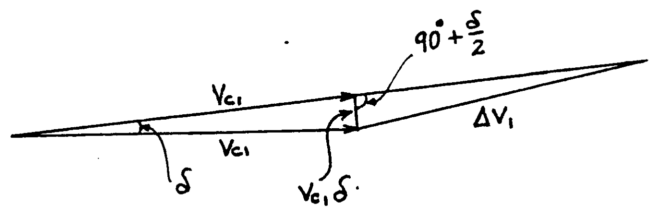

Figure 3. Velocity Vector Geometry for Initial Maneuver

isoceles triangles may be drawn in this case: one with sides v_2, the other with Δv_2. The overall length of the right side of the triangle is the reference velocity change for the second maneuver, Δv_{2r}:

$$(2.3) \qquad \Delta V_{2r} = \sqrt{V_{c2}^2 + V_2^2 - 2\, V_{c2}\, V_2 \cos\Theta}$$

The base angle Ω can be defined using the law of sines:

$$(2.4) \qquad \Omega = \sin^{-1}\left(\frac{V_{c2}}{\Delta V_{2r}} \sin\Theta\right)$$

It is desired to find v_ϵ, which is the difference between Δv_{2r} and Δv_2. The minor included angle ω is equal to Ω-90+ $\delta/2$, but using the approximation $\delta/2=0$ again yields $\omega=\Omega-90$, and

$$(2.5) \qquad V_\epsilon = V_2\, \delta$$

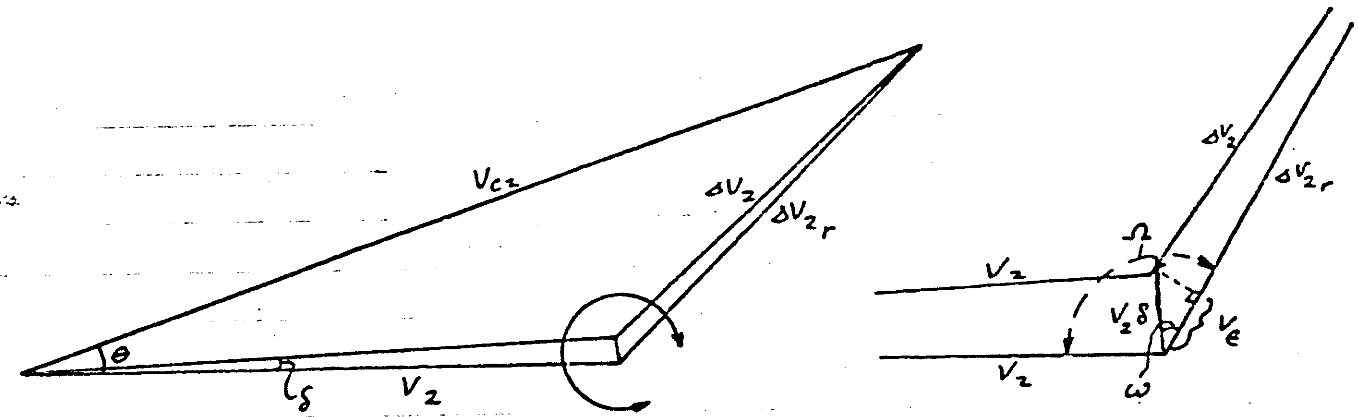

Figure 4. Velocity Vectors for Circularization Maneuver

Using (2.4) and (2.5), the estimate of Δv_2 can now be found:

$$(2.6) \qquad \Delta V_2 = \Delta V_{2r} - V_2 \, \delta \, \frac{V_{c2}}{\Delta V_{2r}} \sin \theta$$

The total velocity change required assuming a small plane change angle in the initial maneuver is:

$$(2.7) \qquad \Delta V_T = \sqrt{(V_1 - V_{c1})^2 + V_{c1}^2 \, \delta^2} + \Delta V_{2r} - \delta \sin\theta \, \frac{V_2 V_{c2}}{\Delta V_{2r}}$$

The desired result is the value of δ which will result in minimum total ΔV for the transfer. Since the approximations used resulted in a single expression linking δ and Δv_T, the simplest way to optimize for δ is to take $\partial \Delta v_T / \partial \delta$, set the derivative equal to zero, and solve for δ in terms of the remaining variables. Doing so, it can be found that the optimum initial plane change angle, δ_{opt}, is

$$(2.8) \qquad \delta = \frac{(V_1 - V_{c1})\, V_K}{V_{c1} \sqrt{V_{c1}^2 - V_K^2}}$$

where

$$(2.9) \qquad V_K = \sin\Theta \; \frac{V_2 \; V_{c2}}{\Delta V_{2r}}$$

Given the estimate of optimal initial plane change, equation (2.1) and (2.3) can be used to find the actual velocity changes for the transfer maneuvers. Figure 5 on page 40 shows the comparison between estimated and actual velocities as a function of δ for transfer between LEO and GEO, and Figure 6 on page 41 illustrates the actual velocity requirements for transfer from LEO to a range of circular orbits, using an estimated optimal δ found from equation (2.8).

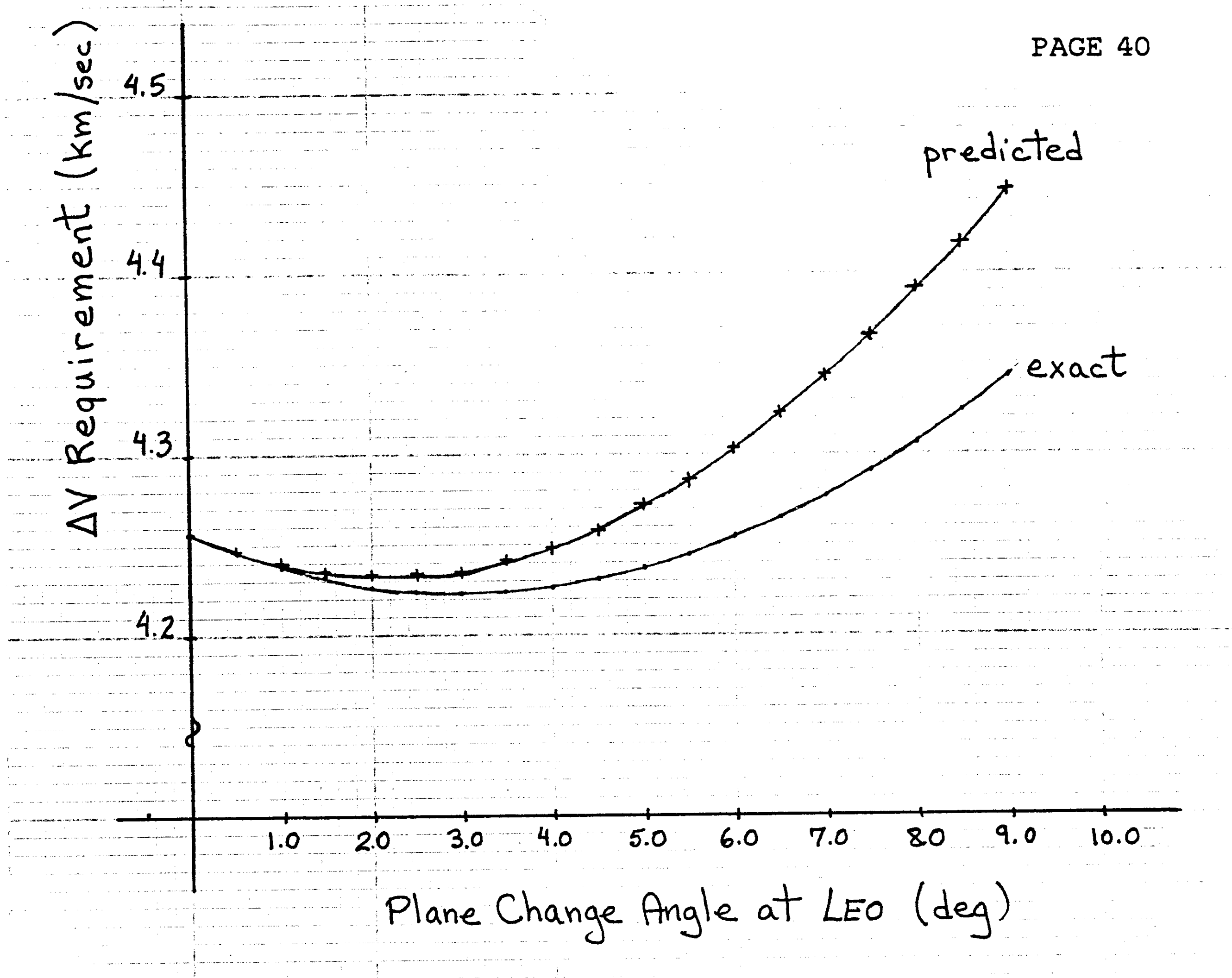

Figure 5. ΔV Requirements for GEO Transfer with Plane Change

2.2.3 NONIMPULSIVE THRUST TRANSFERS

A further source of possible inaccuracies in the classical two-body analysis is the assumption of impulsive thrust: that is, that all velocity changes are made instantly, with a corresponding infinite thrust of zero duration. This influences two separate analyses:

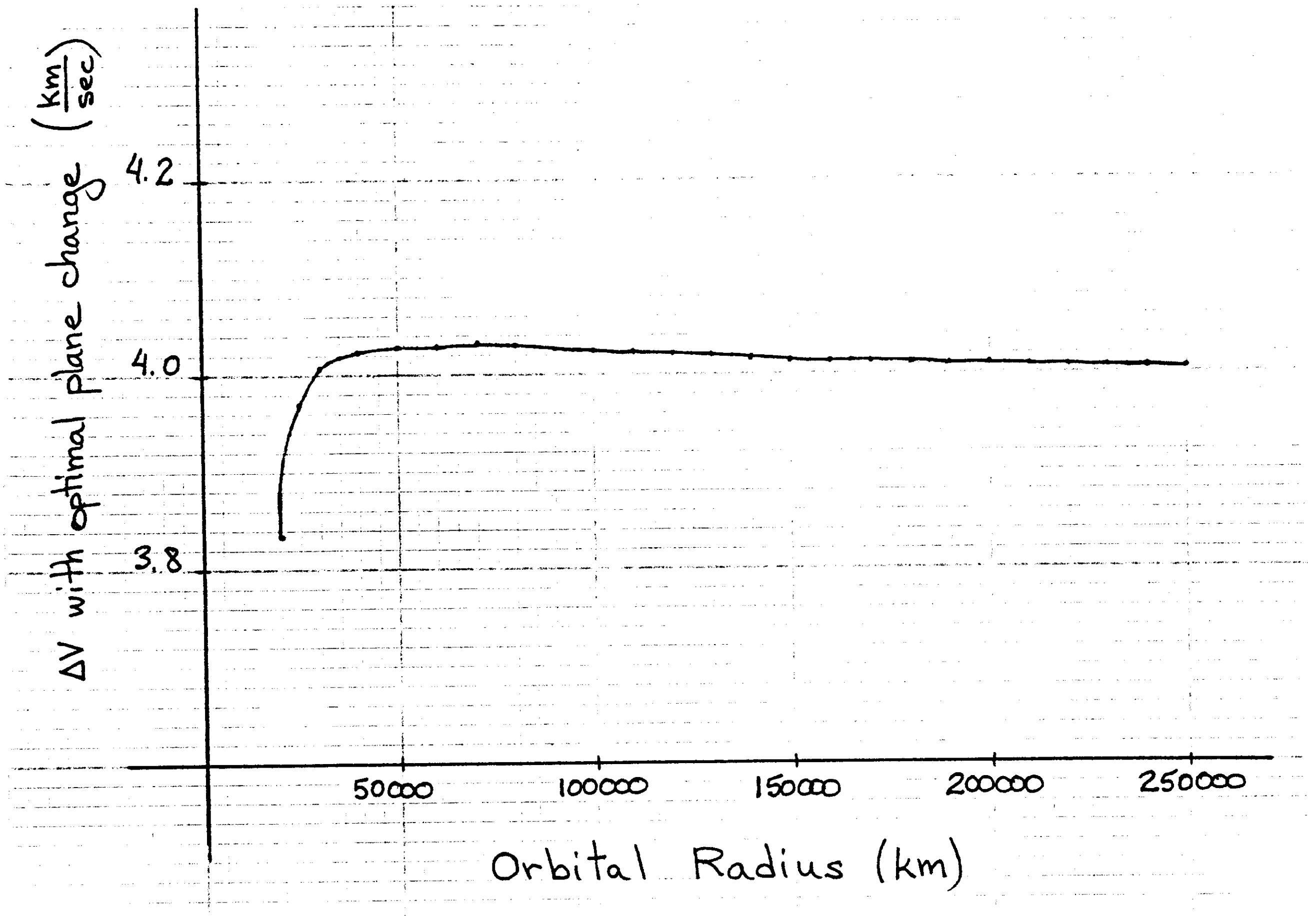

Figure 6. ΔV for Orbital Transfer with Optimal Plane Change: Initial orbital radius = 6750 km

---

- the high-thrust transfers (assuming chemical or nuclear engines) do not take place instantaneously, and it is therefore necessary to correct the ΔV estimates for these missions
- the low-thrust trajectories do not follow Hohmann ellipses at all, but instead tend to spiral outward over long periods of time.

For these reasons, it is desirable to find the sensitivity of the ΔV analysis to a nonimpulsive thrust situation.

The equations of motion in polar coordinates for a body undergoing external forces are

$$(2.10) \qquad \ddot{r} = a_r + r\dot{\theta}^2$$

$$(2.11) \qquad r\ddot{\theta} = a_\theta - \frac{2}{r}\dot{r}\dot{\theta}$$

Gravity exerts a force inward in the radial direction, of magnitude μ/r^2. The assumption made in this analysis is that the powered flight occurs at a constant thrust: this is generally advantageous, if non-optimal, for both chemical and ion thrust systems. With this simplifying assumption, the acceleration of the vehicle at any time is the ratio between thrust and mass, or

$$(2.12) \qquad \frac{T}{m} = \left[\frac{m_0}{T} - \frac{\dot{m}t}{T}\right]^{-1}$$

However, thrust can also be rewritten as

$$(2.13) \qquad T = \dot{m}c$$

where c is the engine exhaust velocity. Introducing the parameter τ, which is the initial vehicle acceleration, equation (2.12) can be rewritten as

$$\frac{T}{m} = \frac{\tau c}{c - \tau t} \tag{2.14}$$

Figure 7 shows that the current velocity vector, $\overline{V}$, is composed of the radial velocity vector $\overline{v}$ and the tangential velocity vector $\overline{r\omega}$. Flight path angle ϕ can be found by

$$\sin \phi = \frac{v}{V} \tag{2.15}$$

$$\cos \phi = \frac{r\omega}{V} \tag{2.16}$$

In addition, the radial and tangential components of acceleration can be specified by

$$a_r = \frac{T}{m} \sin \phi \tag{2.17}$$

$$a_\theta = \frac{T}{m} \cos \phi \tag{2.18}$$

Selecting radial distance r and downrange angle θ as the variables of integration, along with their derivatives v and ω, the equations of state during powered flight can be written as

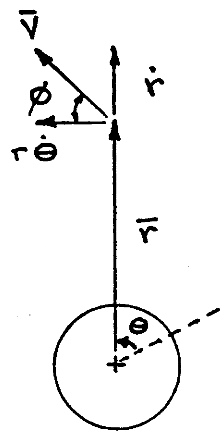

Figure 7. Powered Orbital Transfer Geometry

(2.19) $\dot{r} = v$

(2.20) $\dot{\theta} = \omega$

(2.21) $\dot{v} = \dfrac{\tau c}{c - \tau t} \dfrac{v}{V} + r\omega^2 - \dfrac{\mu}{r^2}$

(2.22) $$\dot{\omega} = \frac{\tau c}{c - \tau t} \frac{\omega}{V} - 2 \frac{v \omega}{r}$$

where

(2.23) $$V = \sqrt{v^2 + r^2 \omega^2}$$

Since the objective of this maneuver is to transfer between orbits, the approach to the desired final orbit can be monitored by continually updating the orbital parameters:

(2.24) $$a = \left(\frac{2}{r} - \frac{V^2}{\mu} \right)^{-1}$$

(2.25) $$e = \sqrt{1 - \frac{r^2 V^2}{\mu a} \cos^2 \phi} = \sqrt{1 - \frac{r^4 \omega^2}{\mu a}}$$

For example, to find the velocity requirements for transfer between LEO and GEO with a finite τ, equations (2.19) through (2.22) would be numerically integrated until such time as the apogee of the instantaneous orbit, $r_a = a(1+e)$, is equal to geostationary altitude. At that point, powered flight would be terminated, and coasting would occur until the second powered maneuver at apogee is required to circularize at GEO. Assuming that this maneuver is performed impulsively, due to the lower total Δv requirement for this maneuver, Figure 8 on page 46 shows the effect of differing thrust levels on the velocity change requirements for a LEO-GEO transfer. It can be seen that as the initial acceleration decreases, velocity requirements increase for the first burn; although Δv requirements are decreased for the circularization burn, overall total Δv increases with decreasing acceleration levels, and approach a finite level at the lower accelerations where thrusting occurs almost continually throughout the transfer. It would be desirable to find an analytical expression for this maximum Δv in the case of infinitesimal thrust.

From Battin [24], the variational equation for orbital semi-major axis is

$$\text{(2.26)} \qquad \frac{da}{dt} = \frac{2\,a^2\,v}{\mu}\,a_\theta$$

The acceleration can be found from equation (2.14). Figure 9 on page 47 shows the variation in maximum transfer eccentricity with variation in the initial acceleration, and demonstrates that the transfer orbit stays nearly circular as the acceleration levels decrease. From this, it seems reasonable to assume that the transfer orbit remains nearly circular throughout the transfer, so that

$$\text{(2.27)} \qquad v = \sqrt{\frac{\mu}{a}}$$

Using this result, equation (2.25) can be rewritten in the form

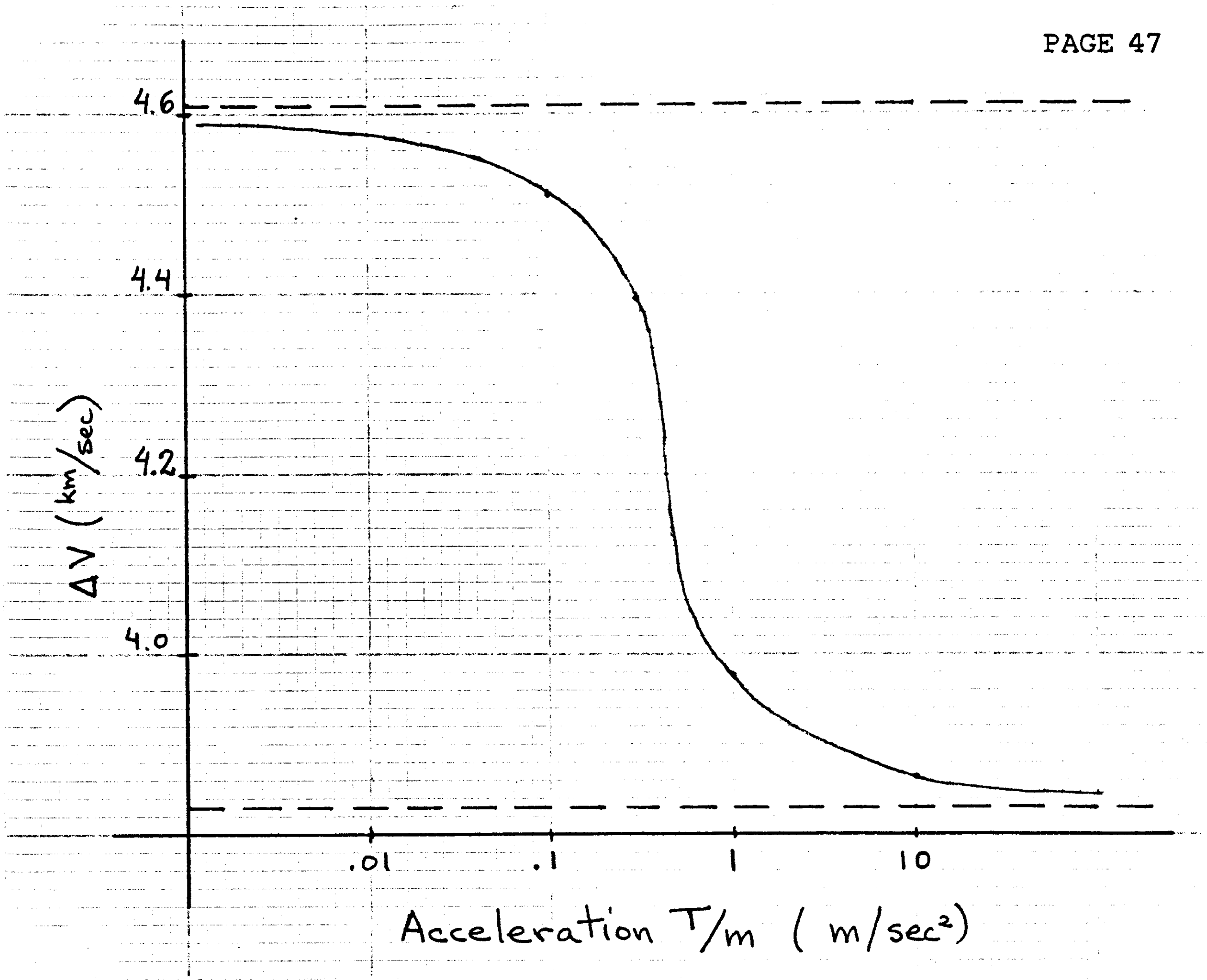

Figure 8. Velocity Requirements for GEO Transfer as a Function of Acceleration

---

(2.28) $$\frac{da}{dt} = \frac{2\tau}{\sqrt{\mu}} a^{3/2} \frac{1}{1 - \frac{\tau}{c} t}$$

This is a separable differential equation, and can be rewritten again as

(2.29) $$a^{-3/2}\, da = \frac{2\tau}{\sqrt{\mu}} \frac{dt}{1 - \frac{\tau}{c} t}$$

Integrating this results in

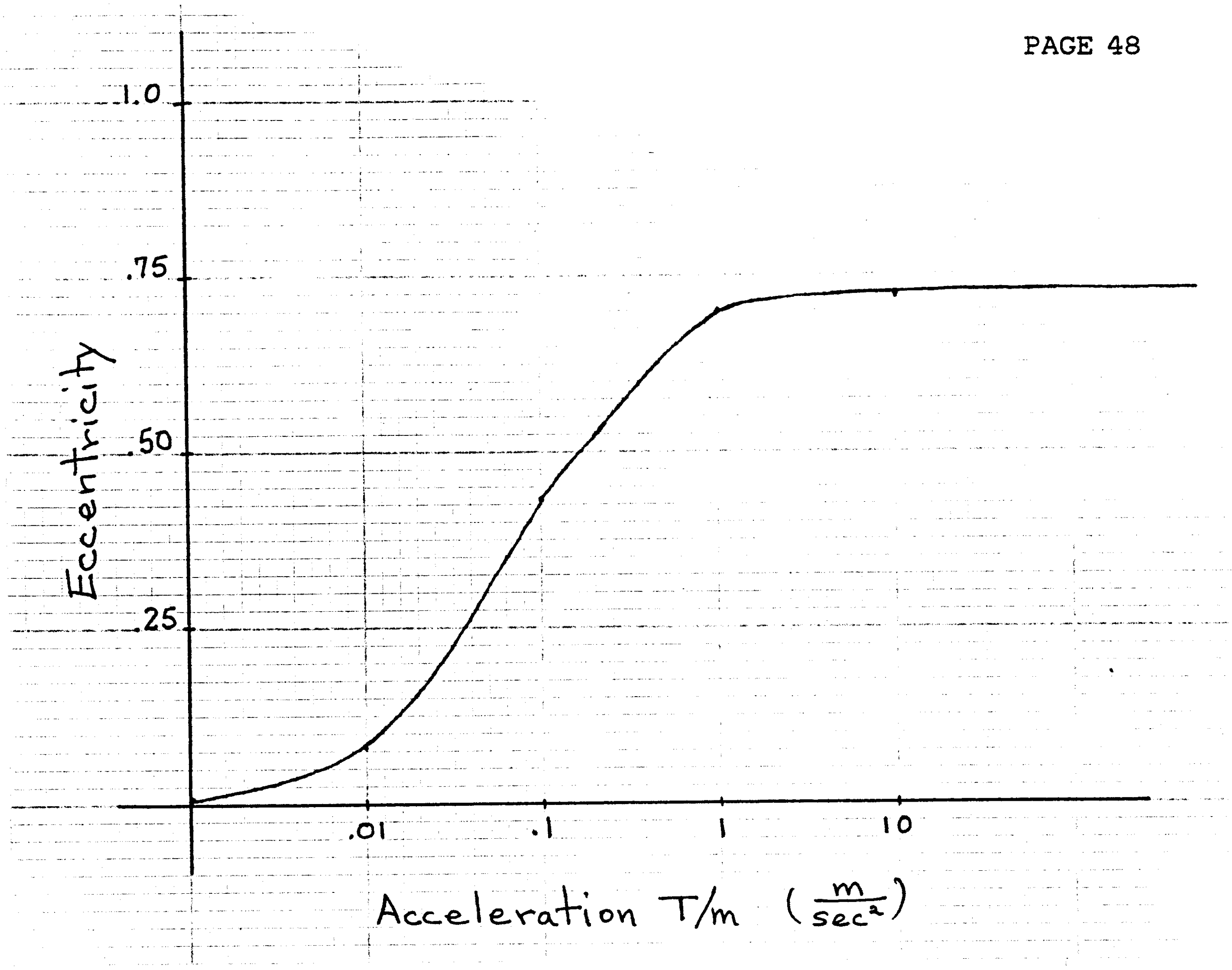

Figure 9. Eccentricity Variation with Vehicle Acceleration

(2.30) $$-2\,a^{-1/2} = \frac{2}{\sqrt{\mu}}\,(-c)\,\ln\left(1-\frac{\tau}{c}t\right) + K$$

where K is the constant of integration. From the basic rocket equation,

(2.31) $$\frac{\Delta v}{c} = \ln\left(1-\frac{\tau}{c}t\right)$$

so $\Delta v/c$ can be directly substituted into (2.30).

In order to find the constant of integration, the semimajor axis at transfer initiation t=0 can be defined as a_o . Substituting this into (2.30) gives

(2.32) $$K = \frac{1}{\sqrt{a_o}}$$

Combining (2.30), (2.31), and (2.32) results in the expression

(2.33) $$\sqrt{\frac{\mu}{c}} \left(\frac{1}{\sqrt{a}} - \frac{1}{\sqrt{a_o}} \right) = - \frac{\Delta v}{c}$$

Noting from (2.27) that $\sqrt{\mu/a}$ is equivalent to the circular orbital velocity at that altitude, the final relation for total Δv between two circular orbits with infinitesimal thrust is simply

(2.34) $$\Delta v = V_{c1} - V_{c2}$$

where V_{c1} and V_{c2} are the circular orbital velocities of the initial and final orbits, respectively.

2.2.4 TRANSPLANETARY TRAJECTORIES BY THE PATCHED CONIC METHOD

While previous sections have examined the energy requirements for transfers between circular orbits of a single body, the use of nonterrestrial materials demands that at least some transport take place between bodies: from the earth to the moon, and into the region where gravitational attractions of the bodies are comparable, such as the Lagrange points. Although analyzed later in greater depth, the current analysis will use the two-body techniques derived earlier to estimate velocity requirements for a spacecraft maneuvering between the two gravitating bodies.

The steps inherent in this approach are:

- Perform an initial maneuver to transfer from the starting orbital radius to the orbital radius of the second body (in this case, the earth-lunar radius.)
- Consider the difference between the spacecraft apogee velocity (in the two-body case) and the lunar orbital velocity to be the hyperbolic excess velocity of the spacecraft, as if it were approaching the moon from an infinite distance.

- Calculate the velocity change required to brake the spacecraft from its hyperbolic orbit of the moon into the desired circular orbit (again, a two-body analysis).

An excellent example of the use of this version of the patched conic technique can be seen in [25].

The initial velocity requirement can be found from equation (A.4), where r_2 is the earth-moon distance. After this maneuver, the spacecraft is on an elliptical orbit to the vicinity of the moon. In order to calculate the velocity requirement for the second maneuver, lunar orbit insertion, the point of view of the calculations must be changed from geocentric to selenocentric. As the spacecraft falls into the lunar field of influence, it carries with it a hyperbolic excess velocity of

$$V_h = V_{c2} - V_a \tag{2.35}$$

where v_{c2} is the circular orbital velocity of the moon around the earth, and v_a is the apogee velocity of the spacecraft in its transfer orbit, neglecting the influence of lunar gravitation.

Selecting the lunar orbital radius r_L, the launch timing is arranged so as to make the spacecraft fly by the moon tangent to the desired final orbit. By conservation of energy, the kinetic energy of the spacecraft at that point is the sum of the kinetic energy due to the hyperbolic excess velocity and that created by the spacecraft falling within the lunar gravitational field, which is equal to the parabolic escape energy at that point. Summing the two, the spacecraft velocity at perilune in the hyperbola is

$$V_p = \sqrt{\frac{2\mu_L}{r_L} + V_h^2} \tag{2.36}$$

In order to achieve circular lunar orbit, the velocity must be decreased to circular orbital velocity, or

$$\Delta V_2 = \sqrt{\frac{2\mu_L}{r_L} + V_h^2} - \sqrt{\frac{\mu_L}{r_L}} \tag{2.37}$$

Equations (2.4) and (2.37) thus describe the magnitudes of the two velocity changes which must be made in order to transfer from a circular orbit about one body to a circular orbit about a second. The accuracy of the patched conic solution technique will be investigated in the following section of the report.

2.3 THREE-BODY ANALYSIS

In order to check the accuracy of the patched conic solution to the three-body problem proposed earlier, some accurate method must be available for numerically evaluating the true trajectory of a spacecraft under the influence of two gravitating bodies. The technique chosen, multiconics, in turn relies on the capability of estimating the position and velocity vectors of a spacecraft at some given time after a specified position and velocity, while under the influence of a single body. This is known as "the Kepler problem", and can be solved by application of classical two-body analysis.

The angular momentum vector for an elliptical orbit is

$$\bar{h} = \bar{r} \times \bar{v} \tag{2.38}$$

The eccentricity vector points in the direction of the orbital periapsis, and has a magnitude equal to the orbital eccentricity. This vector can be found by

$$\bar{e} = \frac{1}{\mu}\left(\bar{v} \times \bar{h}\right) - \frac{\bar{r}}{r} \tag{2.39}$$

The true anomaly is the angle between the current radius vector and the periapsis of the orbit, and is

$$(2.40) \qquad \theta = \cos^{-1}\left(\frac{\bar{r}\cdot\bar{e}}{re}\right)$$

The eccentric anomaly, on the other hand, is the angle between the periapsis of the orbit and the line drawn from the center of the elliptical orbit to the projection of the current position onto the circumscribed circle (see Figure 10 on page 55), or

$$(2.41) \qquad E = \cos^{-1}\left(\frac{e + \cos\theta}{1 + e\cos\theta}\right)$$

There is an ambiguity as to the proper quadrant for θ and E. If $r \cdot v<0$, the spacecraft has already passed periapsis, and values for θ and E should be replaced by $2\pi-\theta$ and $2\pi-E$.

The semimajor axis of the ellipse is

$$(2.42) \qquad a = \left(\frac{2}{r} - \frac{v^2}{\mu}\right)^{-1}$$

The transit time from periapsis passage to the current location in the orbit can be found by

$$(2.43) \qquad t_1 = \left(E - e\sin E\right)\sqrt{\frac{a^3}{\mu}}$$

It should be noted that the total orbital period is

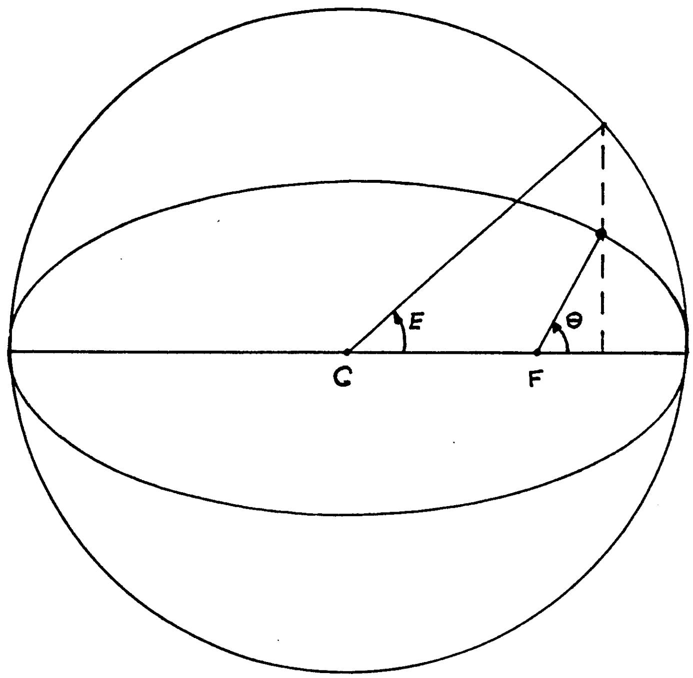

Figure 10. Geometry of Elliptical Orbit

(2.44) $$P = 2\pi\sqrt{\frac{a^3}{\mu}}$$

Since the time step Δt is known, the time of the desired position and velocity estimate is

(2.45) $$t_2 = t_1 + \Delta t$$

Equation (2.43) may be manipulated to find the eccentric anomaly at t_2:

$$(2.46)\qquad E_2 = \sqrt{\frac{\mu}{a^3}}\; t_2 + e \sin E_2$$

An initial estimate is made for E_2, and equation (2.46) solved repeatedly to iterate on the correct value for E_2. Once this is arrived at, the true anomaly at t_2 is found to be

$$(2.47)\qquad \theta_2 = \cos^{-1}\left(\frac{\cos E_2 - e}{1 - e\cos E_2} \right)$$

The parameter of the ellipse, which is the length of the semilatus rectum, is

$$(2.48)\qquad p = \frac{h^2}{\mu}$$

The magnitude of the new radius vector is then

$$(2.49)\qquad r_2 = \frac{p}{1 + e\cos\theta_2}$$

The new position and velocity vectors can now be found:

$$(2.50)\qquad \bar{r}_2 = \frac{r_1 r_2 \sin\Delta\theta}{h}\,\bar{V} + \left(1 - \frac{2r_2}{p}\sin^2\frac{\Delta\theta}{2}\right)\bar{r}$$

$$(2.51)\qquad \bar{V}_2 = \frac{h}{r_1 r_2 \sin\Delta\theta}\left[\left(1 - \frac{2r_1}{p}\sin^2\frac{\Delta\theta}{2}\right)\bar{r}_2 - \bar{r}_1\right]$$

where $\Delta\theta$ is the difference in the true anomalies, or $\theta_2 - \theta_1$.

This approach to solving the Kepler problem is straightforward, yet is not sufficient for the general case of solving orbital parameters given arbitrary initial position and velocity vectors. As mentioned earlier, the spacecraft approaching the moon is generally on a lunar hyperbolic trajectory; yet the previous approach assumes that the orbit is elliptical. In fact, this algorithm is inefficient, as the convergence for (2.46) is slow as orbital eccentricity approaches one. For this reason, a universal variable formulation for Kepler's problem will be used. The detailed background for this formulation can be found in [24] and [26]; only the solution algorithm will be presented here.

Rather than extrapolate forward in the orbit using the eccentric anomaly E, the universal variable formulation uses a independent variable x defined by the differential equation

$$\text{(2.52)} \qquad \dot{x} = \frac{\sqrt{\mu}}{r}$$

Since this is a differential equation for x, we can (with the proper constant of integration) define an arbitrary value for x at time t=0: for convenience, x=0 is usually assumed. Since any conic section may describe the orbit, a new orbital parameter α is defined as 1/a. In this manner, if the resultant orbit is a parabola, α will go to zero, instead of having 'a' approach infinity which would abort a computer run. For the sake of convenience, another variable z can be defined:

$$(2.53) \qquad Z = \alpha x^2$$

This variable can be used to derive two functions C and S, which are

$$(2.54) \qquad C = \frac{1 - \cos\sqrt{Z}}{Z}$$

$$(2.55) \qquad S = \frac{\sqrt{Z} - \sin\sqrt{Z}}{\sqrt{Z^3}}$$

if $\alpha>0$ (elliptical trajectory), or

$$(2.56) \qquad C = \frac{1 - \cosh\sqrt{-Z}}{Z}$$

$$(2.57) \qquad S = \frac{\sinh\sqrt{-Z} - \sqrt{-Z}}{\sqrt{-Z^3}}$$

for $\alpha<0$ (hyperbolic trajectory). If $\alpha=0$, the trajectory is a parabola: from (2.53), z is then identically zero, and then $C=1/2$, while $S=1/3$.

Given an estimate for x after the desired time interval, the values of z, C, and S can be evaluated.

In order to find the actual value of x at time Δt, it is necessary to perform an interation. One effective way to do this is via a Newtonian iteration

$$(2.58) \qquad x_{n+1} = x_n + \frac{t - t_n}{\left.\frac{dt}{dx}\right|_{x=x_n}}$$

From [26], the values of the terms in (2.58) are

$$(2.59) \qquad t_n = \frac{\bar{r}_1 \cdot \bar{V}_1}{\mu} x_n^2 C_n + \frac{1}{\sqrt{\mu}} (1 - r_1 \alpha) x_n^2 S_n + r_1 x_n$$

$$(2.60) \qquad \left.\frac{dt}{dx}\right|_{t=t_n} = \frac{x_n^2}{\sqrt{\mu}} C_n + \frac{\bar{r}_1 \cdot \bar{V}_1}{\mu} x_n (1 - z_n S_n) + r_1 (1 - z_n C_n)$$

Using equations (2.53) through (2.60), the iteration may be continued until the value of x is within a selected limit of the exact solution. The iteration convergence is dependent on the accuracy of the initial estimate of x. For elliptical orbits ($\alpha>0$), an effective starting estimate for x is

$$(2.61) \qquad x \simeq \frac{\sqrt{\mu}\,(t - t_1)}{a}$$

For hyperbolic trajectories ($\alpha<0$), the initial estimate is somewhat more complicated:

$$(2.62) \qquad x \simeq \frac{t - t_1}{|t - t_1|} \sqrt{-\frac{1}{\alpha}} \; \ln \left[\frac{-2\mu (t - t_1)}{\frac{1}{\alpha}\left[\bar{r}_1 \cdot \bar{V}_1 + \frac{t - t_1}{|t - t_1|} \sqrt{\frac{-\mu}{\alpha}} (1 - r_1 \alpha) \right]} \right]$$

Use of these initial estimates in general allows rapid convergence of the iteration for x. In the case of hyperbolic orbits, care must be taken to prevent range errors in the exponential functions used to derive the hyperbolic functions in C_h and S_h.

Since the final vectors $\bar{r}_2$ and $\bar{v}_2$ lie in the plane formed by the initial vectors $\bar{r}_1$ and $\bar{v}_1$, $\bar{r}_2$ can be expressed as a linear combination of $\bar{r}_1$ and $\bar{v}_1$:

(2.63) $$\bar{r}_2 = f\,\bar{r}_1 + g\,\bar{v}_1$$

Differentiating (2.63) gives the corresponding relation for v_2 :

(2.64) $$\bar{v}_2 = \dot{f}\,\bar{r}_1 + \dot{g}\,\bar{v}_1$$

With some derivation, expressions can be found for f, g, $\dot{f}$, and $\dot{g}$ in terms of the universal variable x:

(2.65) $$f = 1 - \frac{x^2 C}{r_1}$$

(2.66) $$g = t - \frac{x^3 S}{\sqrt{\mu}}$$

(2.67) $$\dot{f} = \frac{\sqrt{\mu}}{r_1 r_2}$$

(2.68) $$\dot{g} = 1 - \frac{x^2}{r_2} C$$

where r_2, the magnitude of the new radius vector used in (2.67) and (2.68), is found by substituting the results of (2.65) and (2.66) into (2.63).

In summary, then, the universal variable solution algorithm consists of

1. Given $\bar{r}_1$ and $\bar{v}_1$, find r_1 and α (2.42)
2. Given Δt, find x using equations (2.53) through (2.60)
3. Find f and g from (2.65) and (2.66), then find $\bar{r}_2$ and r_2 from (2.63)
4. Find $\dot{f}$ and $\dot{g}$ from (2.67) and (2.68), then find $\bar{v}_2$ from (2.64)

With the universal variable formulation, the position and velocity vectors for a spacecraft around a single gravitating body may be established for any arbitrary time, as long as one set of position and velocity vectors, and the time which corresponds to them, are known. Using this, then, a multiconic analysis can be used for estimating spacecraft motion between two gravitating bodies.

The multiconic formulation used in this study is the Stumpff-Weiss formulation [27], which relates a future position of the spacecraft to the positions it would have if it were in orbit about each of the gravitating bodies individually. Although this formulation is not as accurate as successive propagation algorithms in certain portions of the earth-moon system, it is still accurate to within a kilometer for time steps on the order of 2-3 hours [28].

The situation is as depicted in Figure 11 on page 63. The Stumpff-Weiss formulation states that the final geocentric position and velocity vectors are

$$(2.69) \qquad \overline{R}_f = \overline{R}_{iEf} + \overline{r}_{imf} - \overline{r}_i - \Delta t\,\dot{\overline{r}}_i + \mu\left(\overline{\rho}_f - \overline{\rho}_i - \Delta t\,\dot{\overline{\rho}}_i\right)$$

$$(2.70) \qquad \dot{\overline{R}}_f = \dot{\overline{R}}_{iEf} + \dot{\overline{r}}_{imf} - \dot{\overline{r}}_i + \mu\left(\dot{\overline{\rho}}_f - \dot{\overline{\rho}}_i\right)$$

where

$\overline{R}_f$ final earth radius vector to spacecraft

$\dot{\overline{R}}_f$ final spacecraft velocity vector relative to earth

$\overline{R}_{iEf}$ radius vector from earth to final point of geocentric conic propagated from point i

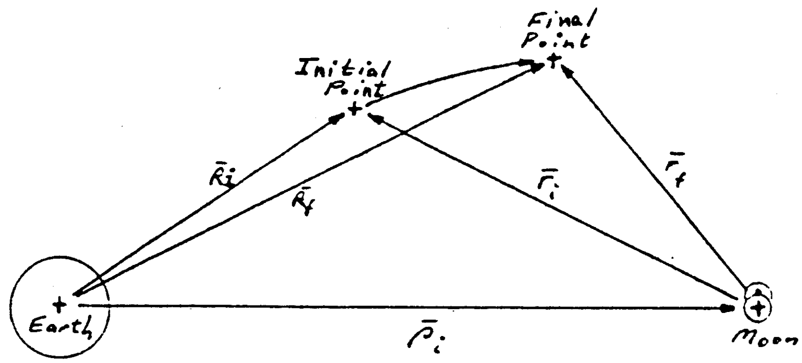

Figure 11. Spacecraft Motion in Cislunar Space

$\dot{\bar{R}}_{iEF}$ velocity vector relative to earth in geocentric conic at R_{iEF}

$\bar{r}_{imF}$ radius vector from the moon to final point of selenocentric conic propagated from point i

$\dot{\bar{r}}_{imF}$ velocity vector relative to moon in selenocentric conic at r_{imF}

$\bar{R}_i$ initial earth radius vector to spacecraft

$\dot{\bar{R}}_i$ initial velocity vector relative to earth

$\bar{r}_i$ initial lunar radius vector to spacecraft

$\dot{\bar{r}}_i$ initial velocity vector relative to moon

$\bar{\rho}_i$ initial earth-moon radius vector

$\bar{\rho}_f$ final earth-moon radius vector

$\dot{\bar{\rho}}_i$ initial velocity vector of moon in earth orbit

$\dot{\bar{\rho}}_f$ final velocity vector of moon in earth orbit

μ ratio between lunar and joint gravitating masses = $\mu_L/(\mu_L + \mu_E)$

For each step estimate in the multiconic analysis, two two-body problems must be solved, as well as the correction terms (terms involving μ) which relate the effect of the moon moving around the earth.

2.4 APPLICATIONS TO EARTH-MOON SYSTEM

Although most of the orbital transfer calculations to this point have been done for general cases, the intent of course has been to apply them to the sitation of interest. In the first part of this chapter, 11 different locations of interest were identified: four earth orbits, one lunar orbit, four Lagrange points of equilibrium in the Earth-moon system, and the surfaces of both the Earth and the moon. Five generic classes of transfers can be identified:

- Planetary surface to orbit
- Orbit to orbit around a single body
- Orbit to orbit transfers between bodies
- Orbit to Lagrange point
- Lagrange point to Lagrange point

The application task will consist of using the derived relations to locate those orbits not yet chosen (radii of IEO, GEO, and LO), and to find the actual ΔV requirements for all transfers of interest. Once this is done, the systems analysis portion of this work can progress with reasonable estimates of transportation requirements. Figure 12 on page 67 shows the effect of orbital altitude of an intermediate processing facility on the total velocity change requirements. The assumptions inherent in this figure are

- Hohmann transfer from LEO to the orbit of the intermediate processing facility (value printed on the abcissa).
- Hohmann transfer from the intermediate stop to GEO

Therefore, this figure shows the impact of the orbital radius of IEO or HEO on the mission requirement of bringing parts from LEO to an intermediate processing site, performing some activity at that site, then transferring the completed goods to GEO. This, for example, is the mission required for completing solar power satellites from parts prefabricated on Earth. Superimposed on the figure is radiation data for a range of altitudes as gathered by an Explorer satellite [29]. This data indicates that practically all of the region between low earth orbit and geostationary has a radiation flux sufficient to make long-term human habitation difficult. In fact, a GEO base would also require some radiation shielding, and has a background radiation high enough that routine EVA would not be possible within allowable cumulative radiation doses. Based on the radiation information and transfer delta-vee requirements, intermediate earth orbits (IEO) will not be further considered in this study, and high earth orbit (HEO) will be at a chosen orbital radius of 70,000 km. This value is chosen somewhat arbitrarily: high enough to be in the free-space radiation

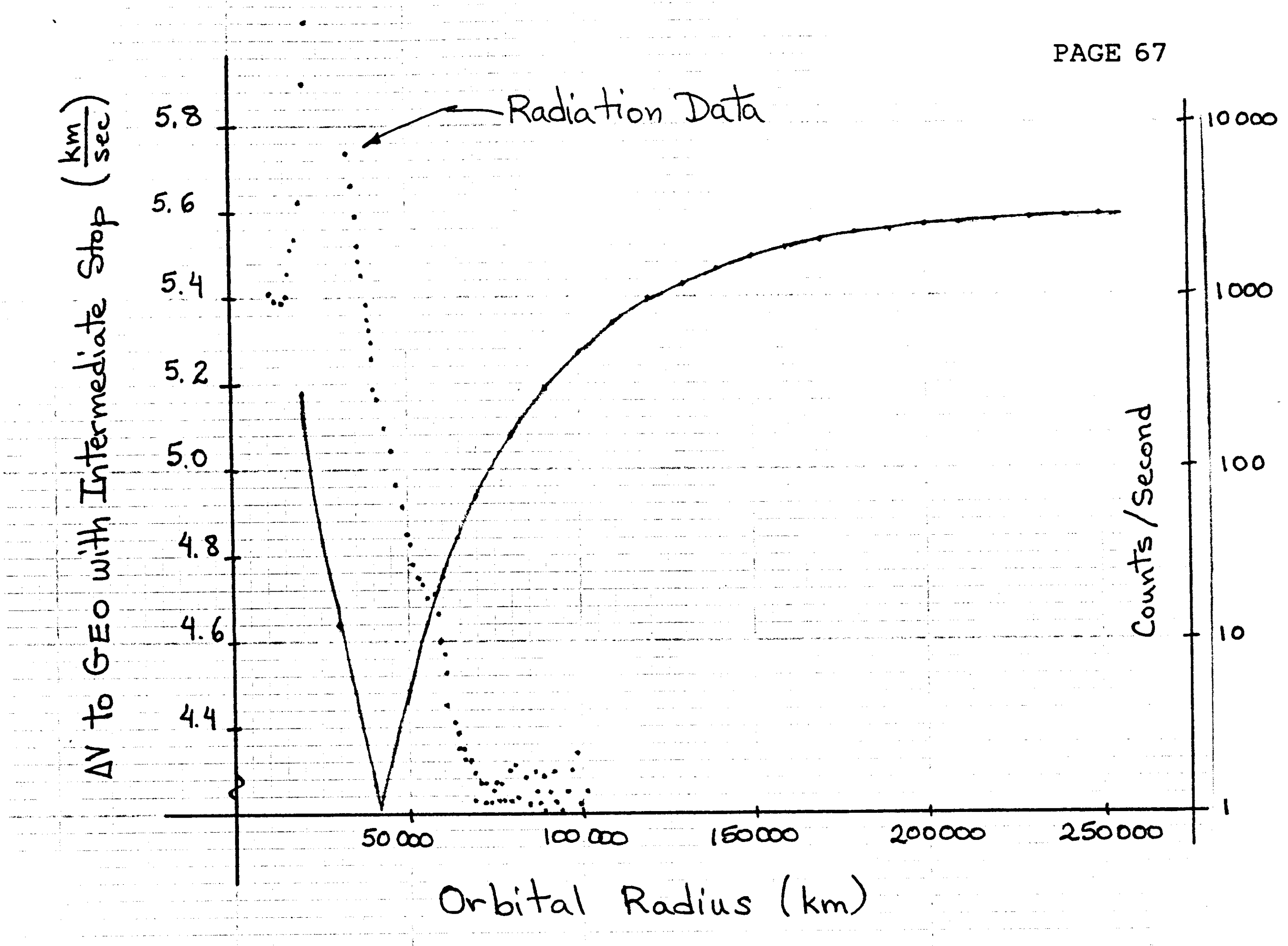

Figure 12. V from LEO to GEO with Intermediate Stop

---

environment, but as low as possible to minimize delta-vee penalty and lunar perturbations. There could be some advantage into selecting an orbital radius of 67,054 km: although this is in a marginally higher radiation environment, the orbital period is exactly 48 hours, rather than the 51 hour 12 minute period at 70000 km. Since no clear mission requirement exists for the 48-hour period, the 70,000 km orbital radius will be assumed for convenience.

A similar analysis can be performed to identify a favorable radius for lunar orbit (LO). Since the moon has no atmosphere, the minimum orbital altitude is one which will reliably allow the spacecraft to clear the lunar mountain ranges. Velocity requirements for launch into lunar orbit (or, since gravitation is a conservative field, for descent from lunar orbit to landing) can be estimated from equations (A.4) and (A.5), except that a spacecraft sitting on the surface is not in orbit, so (A.4) becomes

$$(2.71) \qquad \Delta v_1 = v_{cL} \sqrt{\frac{2r}{r + r_L}}$$

where r_L is the radius of the lunar surface (1738 km), and r is the radius of the lunar parking orbit. Normalizing in the manner of Figure 12 on page 67 ($\rho = r_L/r$, $\nu = v/v_{cL}$, where v_{cL} is circular velocity at lunar surface radius), landing ν can be found as a function of ρ. The same parametric relations can be used for circularization in lunar orbit from a translunar flight from low earth orbit. Using (2.71) and (A.5), the descent velocity requirement from lunar orbit to the surface is

$$(2.72) \qquad \nu = \sqrt{\frac{2}{1+\rho}} + \sqrt{\rho}\left(1 - \sqrt{\frac{2\rho}{1+\rho}}\right)$$

From the patched conic analysis and equation (2.57), the braking velocity into lunar orbit as a function of orbital altitude is

(2.73) $$\nu = \sqrt{\left(\frac{V_h}{V_{CL}}\right)^2 + 2\rho} - \sqrt{\rho}$$

Figure 13 on page 70 demonstrates the variation in braking and landing velocity requirements as a function of the parking orbit radius. In addition, since it might be assumed that a large proportion of material to or from the moon either originated or is destined beyond lunar orbit, the joint velocity of braking and descent is also shown on the graph. Although braking velocity is minimized at high orbital radii, the total velocity for lunar approach and landing is minimized with decreasing orbital radius. For this reason, a lunar orbital radius of 2000 km has been chosen, which corresponds to a lunar altitude of 262 km, which is more than sufficient for avoiding lunar surface features.

In addition, it is possible to eliminate one of the stable Lagrange points from consideration, although they are generally equivalent in terms of location and velocity. Since transfers to these points from the moon use an epoch change maneuver, a more rapid transfer is available to L4 than to L5. Therefore, L4 will be the only Trojan point considered in this study.

The nine locations of interest for this study have now been fully selected:

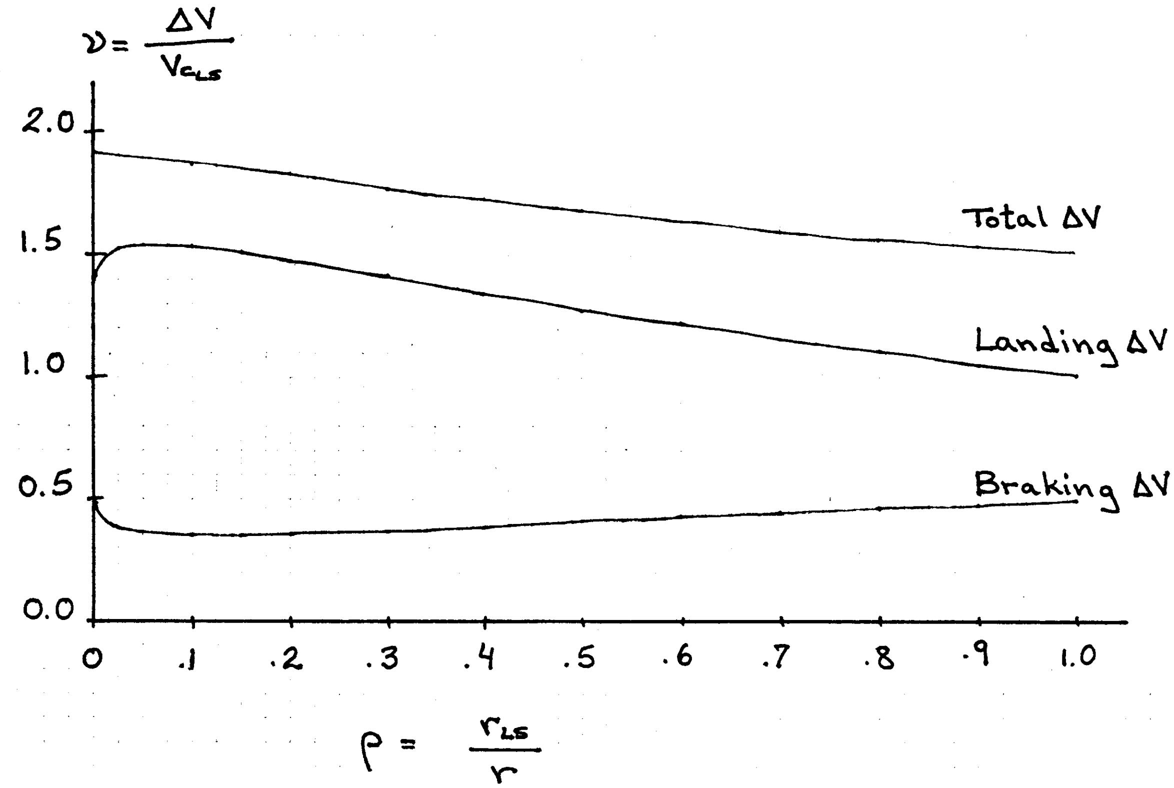

Figure 13. Braking and Landing Velocities as a Function of Lunar Orbital Altitude

---

Earth Surface (ES)	Launch site located at 28.5° latitude
Low Earth Orbit (LEO)	6750 km radius; 372 km altitude
Geostationary (GEO)	42164 km radius; 35786 km altitude
High Earth Orbit (HEO)	70000 km radius; 63622 km altitude
Lunar surface (LS)	Launch site located on the equator
Lunar Orbit (LO)	2000 km radius; 262 km altitude
Lagrange-1 (L1)	Between earth and moon; 326,554 km to Earth, 57,846 km to moon

Lagrange-2 (L2)	Beyond moon on Earth-moon line; 453,475 km to Earth, 69,075 km to moon
Lagrange-4 (L4)	Leading Trojan point; 384,400 km from both Earth and moon

Note: Distance specifications for Lagrange points are distances to the center of the Earth or the moon, not to their surfaces.

The velocity change requirements developed using the techniques derived in this chapter are shown in Table 2-1. Several explanatory notes should be made about the derivation of the numerical values in this table:

- No attempt was made to calculate the ΔV required for launch to LEO. This is a major undertaking, and depends on the particular aerodynamics of the particular launch vehicle design chosen. Use of Hohmann transfer equations, such as for lunar launch, indicate that the minimum ΔV requirement for earth launch would be 7940 m/sec for a launch due east from Kennedy Space Center, including the beneficial effect of the earth's rotation velocity, which is 408 m/sec at that point. To this must be added the effects of gravity loss during the vertical portions of the trajectory, and drag loss while in the atmosphere. For a typical vertical take-

	ES	LEO	GEO	HEO	L4	L2	L1	LO	LS
ES		9000	—	—	—	—	—	—	—
LEO	—		4198	4262	3922	4216	3853	4017	—
GEO	—	4610		678	1646	2084	1514	1812	—
HEO	—	5299	689		1206	2103	1082	1432	—
L4	—	8708	4098	3409					—
L2	—	8891	4281	3592	2229				*
L1	—	8554	3944	3255	1892	2075			*
LO	—	9248	4638	3949	2586	2769	2432		1733
LS	—	—	—	—	—	*	*	—	

Upper diagonal numbers refer to impulsive maneuver
Lower diagonal values refer to continual thrust
All velocity values in m/sec

* - accessible only via mass driver trajectory (one way)

Table 2-1: V Requirements Between Selected System Locations

---

off (VTO) booster, the total velocity increment to a 250 km altitude circular orbit is 9000 m/sec.[30]

- Velocity increments between Earth orbits (LEO, GEO, and HEO) were found by two-body Hohmann transfer requirements. The assumption made is that LEO is at an inclination of 28.5°, while GEO and HEO are equatorial orbits. Plane change penalties were therefore applied to the LEO-GEO and LEO-HEO transfers, using the pseudo-optimal plane-change distributions of this chapter.

- Earth-moon and Lagrange point high-thrust transfers were found by multiconic analyses, using a time step of 10,000 seconds during the coast phase. At 20,000 km from target encounter, the time step was decreased to 1000 seconds for greater accuracy of approach monitoring. Initial estimates of injection velocity vectors were made by patched conic analyses, then heuristically adjusted to accurately target the desired final orbit.

- Finding optimal (or even feasible) low-thrust trajectories in the three-body case is beyond the scope of this effort. All of the basic assumptions of continual thrust transfers (such as orbits remaining circular) are invalid when long powered trajectories pass into and out of the sphere of influence of the moon. While trajectories could be analyzed using Encke's method or Cowell's method[26], targeting would have to be done heuristically, and would be a strong function of vehicle parameters. The low-thrust estimates in the three-body cases were made by assuming that the spacecraft spirals out to infinity in the initial gravity field, then spirals down from infinity around the target body. Therefore, instead of subtracting orbital velocities (from LEO to GEO, for example), the ΔV estimates were found by adding orbital velocities. For example, in transfering from GEO to LO, the ΔV requirement was found by adding circular orbital velocities at GEO and LO (Earth escape velocity plus lunar deceleration from escape to orbital velocity).

3.0 SYSTEMS DEFINITION

Mouths open, Tom, Roger, and Astro stood gaping in fascination at the mighty spaceship resting on the concrete ramp. Her long two-hundred-foot polished beryllium hull mirrored the spaceport scene around them... They eyed the sleek ship from the needlelike nose of her bow to the stubby opening of her rocket exhausts. Not a seam or rivet could be seen in her hull. At the top of the ship, near her nose, a large blister made of six-inch clear crystal indicated the radar bridge. Twelve feet below it, six round window ports showed the position of the control deck. Surrounding the base of the ship was an aluminum scaffold with a ladder over a hundred feet high attached to it. The top rung of the ladder just reached the power-deck emergency hatch, which was swung open, like a giant plug, revealing the thickness of the hull, nearly a foot.

- Carey Rockwell

Within the scope of a space industrialization scenario, a wide variety of systems must be developed. Typical tasks which must be accomplished include production, transportation, support, and operations. A detailed knowledge of the capabilities and requirements of each of these systems is necessary in order to specify an overall industrialization scenario, and vital to the desired end result of identifying quantitatively the most favorable development path toward widespread space industrialization.

3.1 TRANSPORTATION

As developed in the previous chapter, there are three basic classes of transportation in the earth-moon area: Earth launch, lunar launch, and orbit-to-orbit transport. Each system has its own requirements, and must be specified in terms of its own critical parameters. It is these parameters, critical to the quantitative estimate of system capability, which must be identified and estimated.

3.1.1 EARTH LAUNCH

As mentioned in Chapter 3, a typical value for velocity change from Earth surface to low earth orbit is 9000 m/sec. This is for a typical vertical take-off vehicle, injecting into a 150 n.mi. (250 km) circular orbit. Like all rocket-propelled vehicles, a launch vehicle is governed by the "rocket equation":

$$\frac{m_{final}}{m_{initial}} = e^{-\frac{\Delta v}{c}} \tag{3.1}$$

where c is the effective exhaust velocity for the rocket engines. It is this equation which indicates if the propellant reserves on board are sufficient to reach orbital injection.

For each stage of the launch vehicle, the components can be identified in generic terms - that is, wing, fuselage, propulsion, etc. Through the use of linear curvefitting and a review of previous similar designs, the mass of these components can be identified on the basis of the other components of the system:

- Parameters

Δ_W	1 for winged vehicle (reuseable), 0 for ballistic (expendable)
ϵ	mass of fuel tanks/mass of propellants contained
δ	empty mass/total stage mass
δ_F	fuselage mass/total mass contained
δ_{PR}	propulsion system mass/mass carried
λ	stage payload mass/total stage mass
r	final mass/initial mass
R	(1-r)/r = propellant mass/inert mass
c	effective exhaust velocity = g*I
δ_{TP}	mass of thermal protection system/protected mass
Δ_{PL}	1 for internally carried payload, 0 for external

- Mass Elements

M_F	Fuselage

M_T	Tankage
M_P	Propellants
M_{FE}	Fixed Equipment
M_W	Aerodynamic Surfaces
M_{PR}	Propulsion System
M_{TP}	Thermal Protection System
M_{PL}	Payload

Table 3-1 shows the empirical relationships between the parameters listed above. The values of the scaling parameters are taken from [31], and represent general linear relationships between the various mass elements. The use of linear relations is preferable for initial estimation, in that it allows quick solution for vehicle mass properties. Reference [32] lists a detailed set of power-law curve fits for individual system mass properties, which is suitable for preliminary point-design concepts. Based on these values, it is now possible to establish a set of equations describing the mass properties based on the type of launch vehicle (number of stages, carriage of payload, etc.), and solve for the estimates of component system masses.

Since the primary interest of this work lies in the industrial applications of space, the objective function for optimization of the transportation system must be minimum cost. As shown in [33], minimizing cost to orbit in any realistic model (i.e.,

$$\epsilon = .2 \; m_P^{-.1}$$
$$\delta_F = .5$$
$$\delta_W = .125$$
$$\delta_{PR} = .04$$
$$\delta_{TP} = .25$$

Table 3-1: Mass Parameters Estimation Values

---

limited demand) produces an optimum vehicle size based on the vehicle parameters. Since the vehicle estimation algorithm developed here is different than that used in [33], it is necessary to find the specific cost factors affecting vehicle optimal sizing. Since this is an initial estimate, the costing will be done on the basis of vehicle mass. Using this technique, the following costing parameters can be found:

C_0	Nonrecurring cost (research and development), $/kg
C_1	Recurring cost (first unit production), $/kg
M_0	Total mass launched in mission model (kg)
M_E	Empty mass of vehicle (kg)
n	Number of flights/vehicle
n_v	Number of vehicles
r_v	Fraction of vehicle refurbished per flight
y	Number of years in the program
r_i	Interest rate for cost discounting purposes
p	Learning curve exponent

Based on these parameters, it is possible to identify the various cost components that sum to form the overall system cost.

- Research and development costs = $C_0 M_E$
- Number of vehicles required in program = $M_0/(nM_E)$
- Number of vehicle equivalents required to maintain inventory of parts and spares = $M_0/(nM_E) * nr_v = M_0 r_v/M_E$
- Total number of vehicles produced = $(1/n + r_v) M_0/M_E = n_v$
- First unit production cost = $C_1 M_E$
- k^{th} unit production cost = $C_1 M_E (k)^{-p}$
- Assuming that production is spread evenly over the entire program, the time of k^{th} unit production is = ky/n_v
- Net present value of k^{th} unit production cost = $C_1 M_E k^{-p} (1+r_i)^{-ky/n_v} = C_T$

Using these relations, the net present value (NPV) cost of the entire launch program is

$$(3.2) \quad C_T = C_0 M_E + \sum_{k=1}^{n_v} C_1 M_E k^{-p} (1+r_i)^{-ky/n_v}$$

The payload cost is then the total program cost divided by the total mass launched to LEO, or

$$(3.3) \quad C_{PL} = \frac{C_T}{M_0} = \frac{M_E}{M_0} \left[C_0 + C_1 \sum_{k=1}^{n_v} k^{-p} (1+r_i)^{-ky/n_v} \right]$$

3.1.1.1 Single Stage to Orbit (SSTO)

Based on the scaling relations of Table 3-1, six component masses (fuselage, tanks, propellants, wings, propulsion, and thermal protection system) can be specified in terms of the mass ratio r and the masses of the known systems (fixed equipment, payload). Only the specification of the tank mass in terms of propellant mass carried is a nonlinear equation. By finding the five linear equations, a system may be set up to allow a simple iteration for the tank mass.

As specified earlier, the fuselage mass is δ times the mass contained within it. For an internally fueled SSTO vehicle, this consists of fixed equipment, payload (if carried internally), and propellant tank masses. Although the propellants are also carried internally, the structure necessary to carry their mass is inherent in the tank mass, and propellants do not therefore enter directly into the fuselage mass equation, which is

$$(3.4) \qquad M_F = \delta_F \left(M_T + \Delta_W M_{FE} + \Delta_{PL} M_{PL} + M_{PR} \right)$$

Similarly, the wing mass is δ times the mass supported, or

$$(3.5) \qquad M_W = \Delta_W \delta_W \left(M_F + M_T + M_{FE} + M_{PR} + M_{TP} + \Delta_{PL} M_{PL} + M_W \right.$$

This assumes that a payload carried internally into orbit results in an equivalent capability for returning payloads to earth. It should be noted that this is not the case with the space shuttle orbiter, which has a maximum launch payload of 65000 pounds, but a maximum nominal landing payload of 32000 pounds.

In a similar manner, the masses of the propulsion system and thermal protection system may be estimated based on the masses with which they are associated:

$$(3.6)\quad M_{PR} = \delta_P \left[M_F + M_T + M_P + M_{TR} + M_{PL} + \Delta_W \left(M_{FE} + M_W + M_{TP} \right) \right]$$

$$(3.7)\quad M_{TP} = \Delta_W \, \delta_{TP} \left[M_F + M_T + M_{PR} + M_{FE} + M_W + \Delta_{PL} M_{PL} \right]$$

The propellant mass is a function of the mass ratio (dependent on the specific impulse of the engines and the velocity increment) and the total launch mass of the vehicle (empty mass plus payload). The propellant mass can therefore be written as

$$(3.8)\quad M_P = \frac{1-r}{r} \left[M_F + M_T + M_{PR} + M_{PL} + \Delta_W \left(M_{FE} + M_{TP} + M_W \right) \right]$$

Equations (3.4) through (3.8) form a set of five linear equations which define the component masses for a launch vehicle, based on the values of payload, fixed equipment, and tank masses. This linear system is presented in matrix form in Table 3-2. The mass of the propellant tanks is

$$(3.9) \quad M_T = .2\, M_P^{\,.9}$$

which is unfortunately not linear. Rather than attempt to integrate this exponential function into the preceeding equations, the following solution algorithm will be used:

1. Select values for engine specific impulse, payload and fixed equipment masses, parametric factors, and configuration factors
2. Set initial propellant tank mass estimate to zero
3. Solve the set of linear equations for component mass factors
4. Using the derived propellant mass, update the estimate for tank mass
5. Repeat steps 3 and 4 until the values of tank mass converge

This is in general a rapid convergence, requiring on the average only 3-5 iterations to converge within 100 kg.

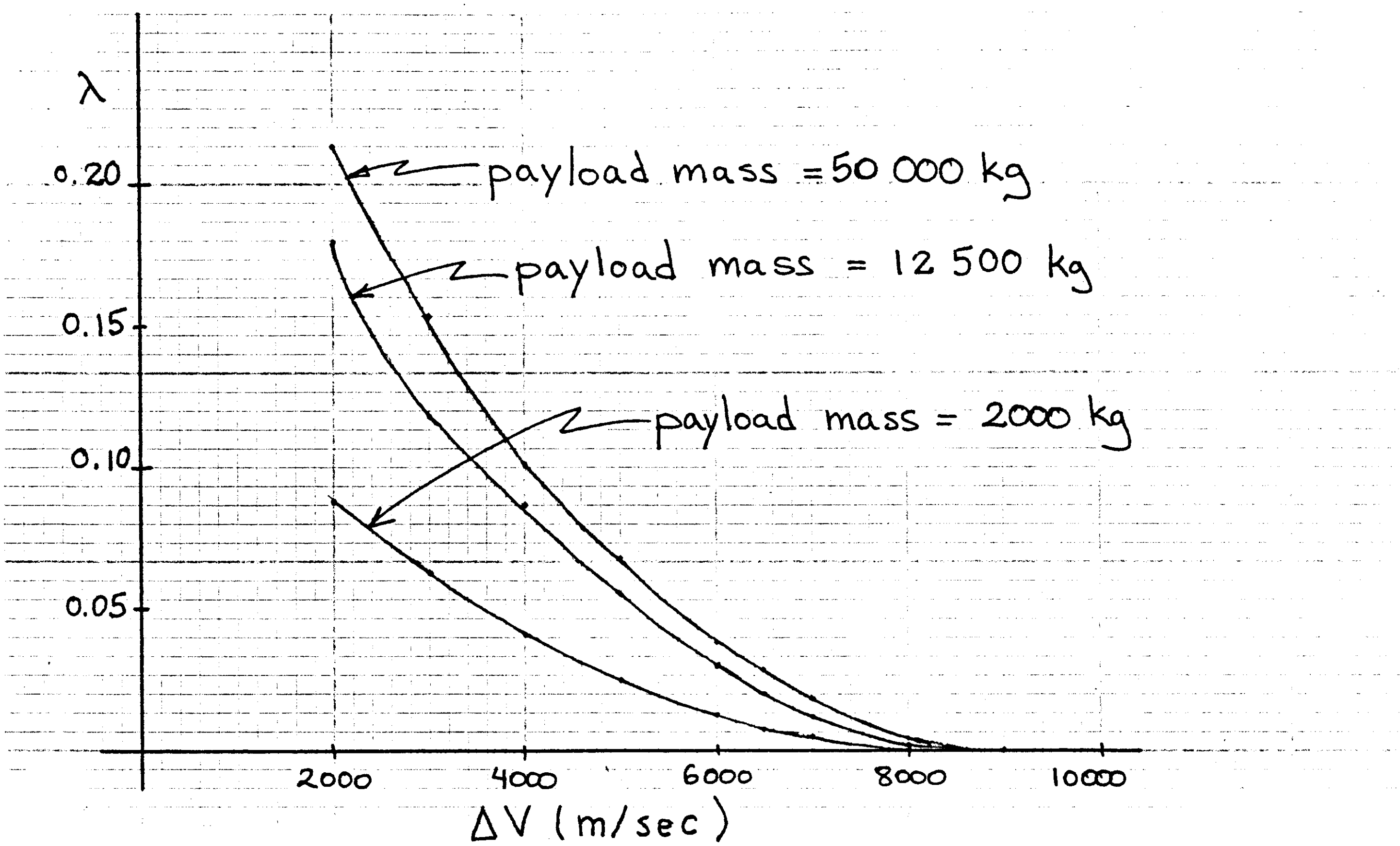

Figure 14. Effect of V on SSTO Payload Fraction

$$
\begin{bmatrix}
1 & 0 & 0 & -\delta_F & 0 \\
-R & 1 & -R & -R & -R \\
-\delta_W & 0 & 1-\delta_W & -\delta_W & -\delta_W \\
-\delta_P & -\delta_P & -\delta_P & 1-\delta_P & -\delta_P \\
-\delta_{TP} & 0 & -\delta_{TP} & -\delta_{TP} & 1
\end{bmatrix}
\begin{bmatrix}
m_F \\ m_P \\ m_W \\ m_{PR} \\ m_{TP}
\end{bmatrix}
=
\begin{bmatrix}
\delta_F \left(m_T + m_{FE} + \Delta_{PL} m_{PL}\right) \\
R \left(m_T + m_{FE} + m_{PL}\right) \\
\delta_W \left(m_T + m_{FE} + \Delta_{PL} m_{PL}\right) \\
\delta_P \left(m_T + m_{FE} + m_{PL}\right) \\
\delta_{TP} \left(m_T + m_{FE} + \Delta_{PL} m_{PL}\right)
\end{bmatrix}
$$

Table 3-2: Component Mass Equations for SSTO Vehicles

The use of a single-stage to orbit vehicle is shown in Figure 14 on page 84. As the velocity increment increases, the vehicle becomes larger and larger to enclose the necessary propellants. As an indication of this, this figure shows the payload fraction λ as a function of ΔV. As the ΔV increases, λ decreases, until at 8000 m/sec the payload fraction has dropped below 0.01 for all three of the payload masses plotted. As the relative position shows, payload fraction increases with increasing payload size. This might lead one to assume that larger vehicles are optimal. However, Figure 15 on page 86 shows that the smaller vehicles tend to cost less per kilogram carried, despite the small payload fractions. This is due to the larger production runs required to launch a set mass with smaller vehicles of a given lifetime, which in turn allows learning and mass production techniques to reduce the cost per vehicle significantly. As Figure 14 on page 84 shows, however, single stage to orbit with fully internal tanks becomes impractical above 8000 m/sec: this is why most designs for this class of vehicle include air launch, in order to cut down on drag and gravity losses in the launch trajectory. For a truly vertical takeoff single stage to orbit, the penalty for carrying all the tankage internally is prohibitive, and drop tanks present a possible alternative.

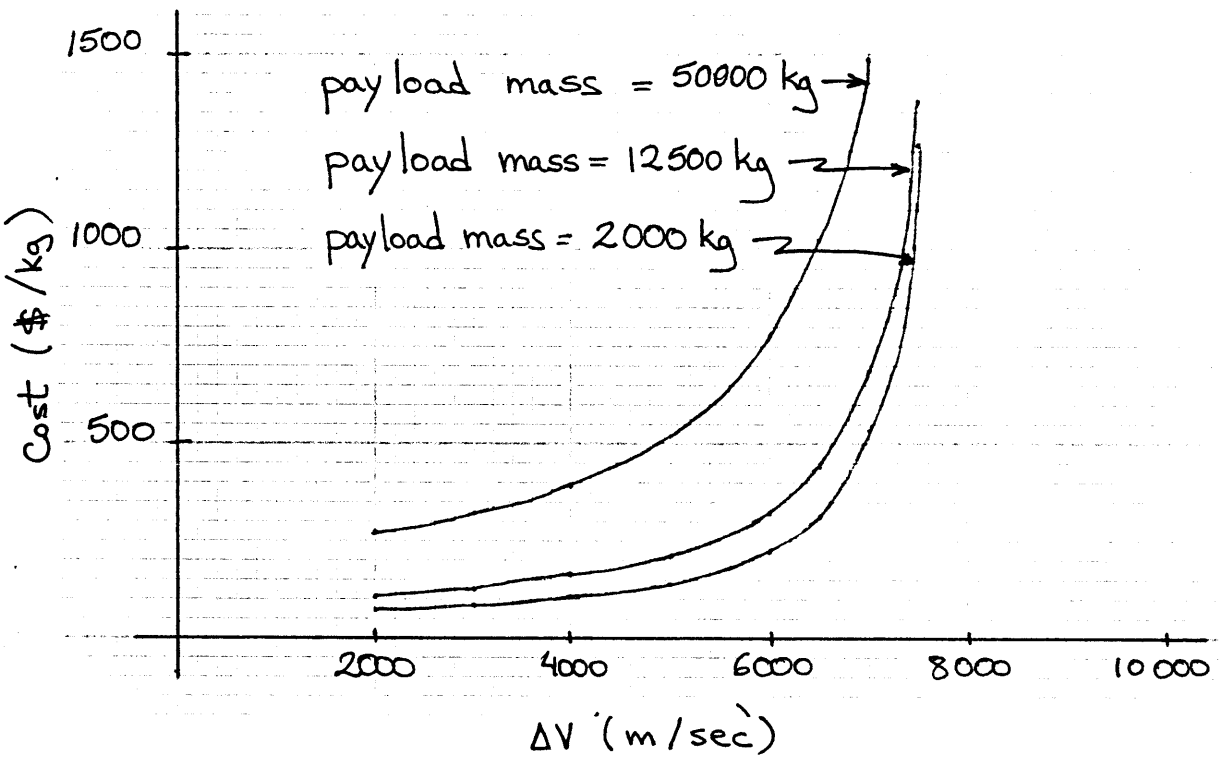

Figure 15. SSTO Cost as a Function of ΔV

3.1.1.2 Single Stage, External Tanks (SSET)

The derivation of the component masses for the SSET option is similar to that of the preceding section, with three of the linear equations modified by dropping the M terms. Since the propellant tanks are carried externally, there is no need to include them in calculating the fuselage, wing, or thermal protection system masses. However, the propellant and propulsion system masses are functions of gross lift-off mass, and there-

$$\begin{bmatrix} 1 & 0 & 0 & -\delta_F & 0 \\ -R & 1 & -R & -R & -R \\ -\delta_W & 0 & 1-\delta_W & -\delta_W & -\delta_W \\ -\delta_P & -\delta_P & -\delta_P & 1-\delta_P & -\delta_P \\ -\delta_{TP} & 0 & -\delta_{TP} & -\delta_{TP} & 1 \end{bmatrix} \begin{bmatrix} m_F \\ m_P \\ m_W \\ m_{PR} \\ m_{TP} \end{bmatrix} = \begin{bmatrix} \delta_F \left(m_{FE} + \Delta_{PL} m_{PL} \right) \\ R \left(m_T + m_{FE} + m_{PL} \right) \\ \delta_W \left(m_{FE} \quad \Delta_{PL} m_{PL} \right) \\ \delta_P \left(m_T + m_{FE} + m_{PL} \right) \\ \delta_{TP} \left(m_{FE} + \Delta_{PL} m_{PL} \right) \end{bmatrix}$$

Table 3-3: Component Mass Equations for SSET Vehicles

---

fore still include tank mass in their defining equations. The modified linear equation set is shown in Table 3-3. In addition, the equations for payload launch costs must be modified, in that the tanks are considered expendable. Therefore, tank costs must be accounted separately, with their own learning curve for the greater number of units produced. The modified form of cost equation (3.3) is

$$(3.10) \quad C_{PL} = \frac{M_E}{M_0}\left[C_0 + C_1 \sum_{k=1}^{n_v} k^{-p} \left(1 + r_i\right)^{-ky/n_v}\right] + \frac{M_T}{M_0}\left[C_0 + C_1 \frac{\left(M_0/M_E\right)^{1-p}}{(1-p)\,y} \; \frac{\left[1-(1+r_i)^{-y}\right]}{r_i}\right]$$

where the second term describes the cost of producing tanks for each of the flights in the program. In this formulation, M_E is assumed to be the empty mass of the orbiter alone, with the tankage empty mass accounted separately. Since the number of tanks built could easily run into the thousands for a moderate program, the cost per tank is done on the basis of average costs over the whole program, rather than performing the summation on a tank-by-tank basis to get exact expenditures as a function of time. While this introduces some error, the cost of summing over the number of tanks required for a large program and a small vehicle size quickly becomes computationally expensive. These modification produce the results shown in Figure 16 on page 89. The use of expendable external tanks results in a marginal increase in payload fraction λ at low ΔV's, and substantial increases in λ as the velocity increment approaches orbital insertion values.

Comparison of the relative costs of internal and external tankage is shown in Figure 17 on page 90, which plots cost per kilogram of payload against the velocity increment of the vehicle. It can be seen from this figure that, for low velocities, the reusable aspect of internal tankage results in lower overall payload costs. However, for larger payloads or larger velocity capabilities, the added complexity of carrying the tanks internally causes the external tank to become the cheaper option. At velocities approaching orbital, only the external tank option is economically viable. At the specified orbital requirement of 9000 m/sec, though, even the expendable tank option is economically disadvantaged, requiring in excess of \$1000/kg to orbit. This is due in large part to the assumption made in this analysis that all propellants were carried in a single external tank. Although a single large tank is struc-

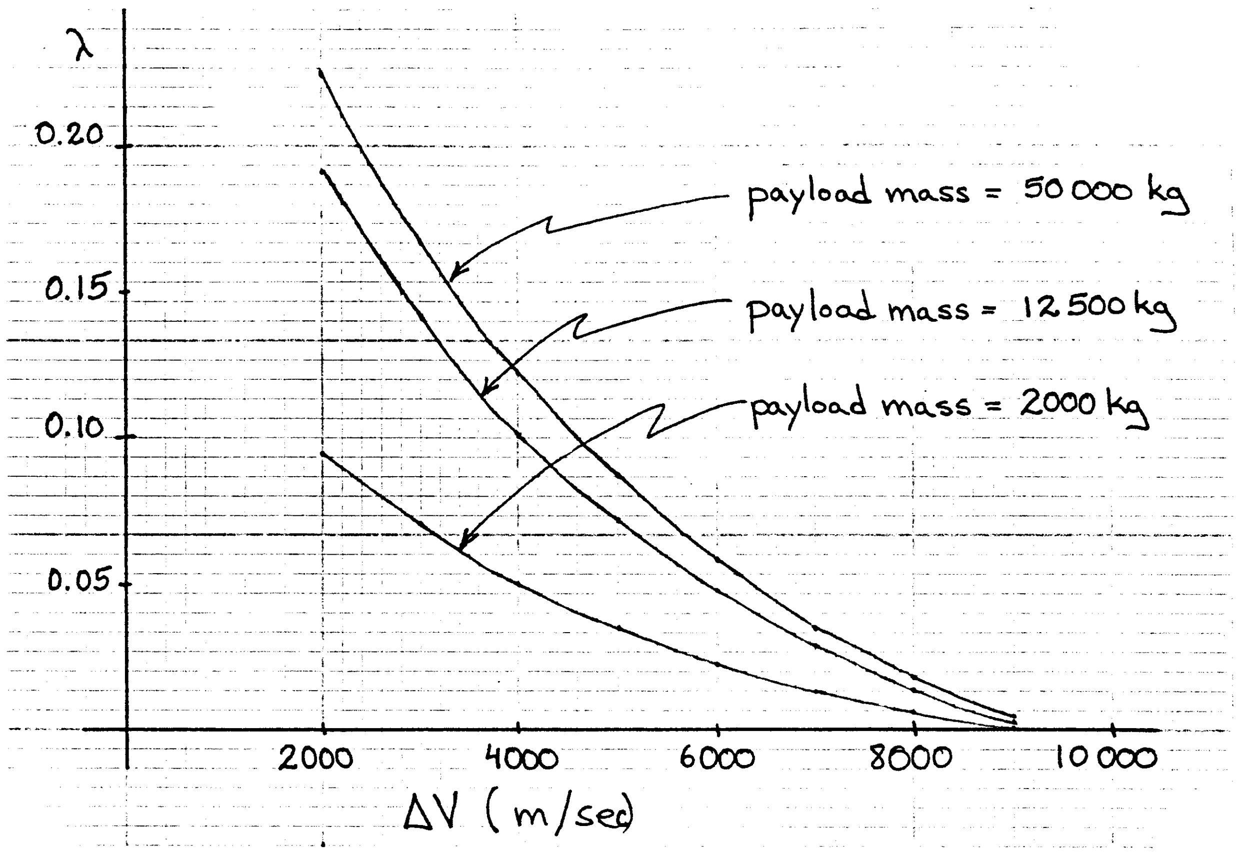

Figure 16. SSET Payload Fraction as a Function of ΔV

turally more efficient than a number of smaller ones, the use of several external tanks would allow tanks to be dropped as they are depleted, thereby reducing the amount of excess mass carried to orbit.

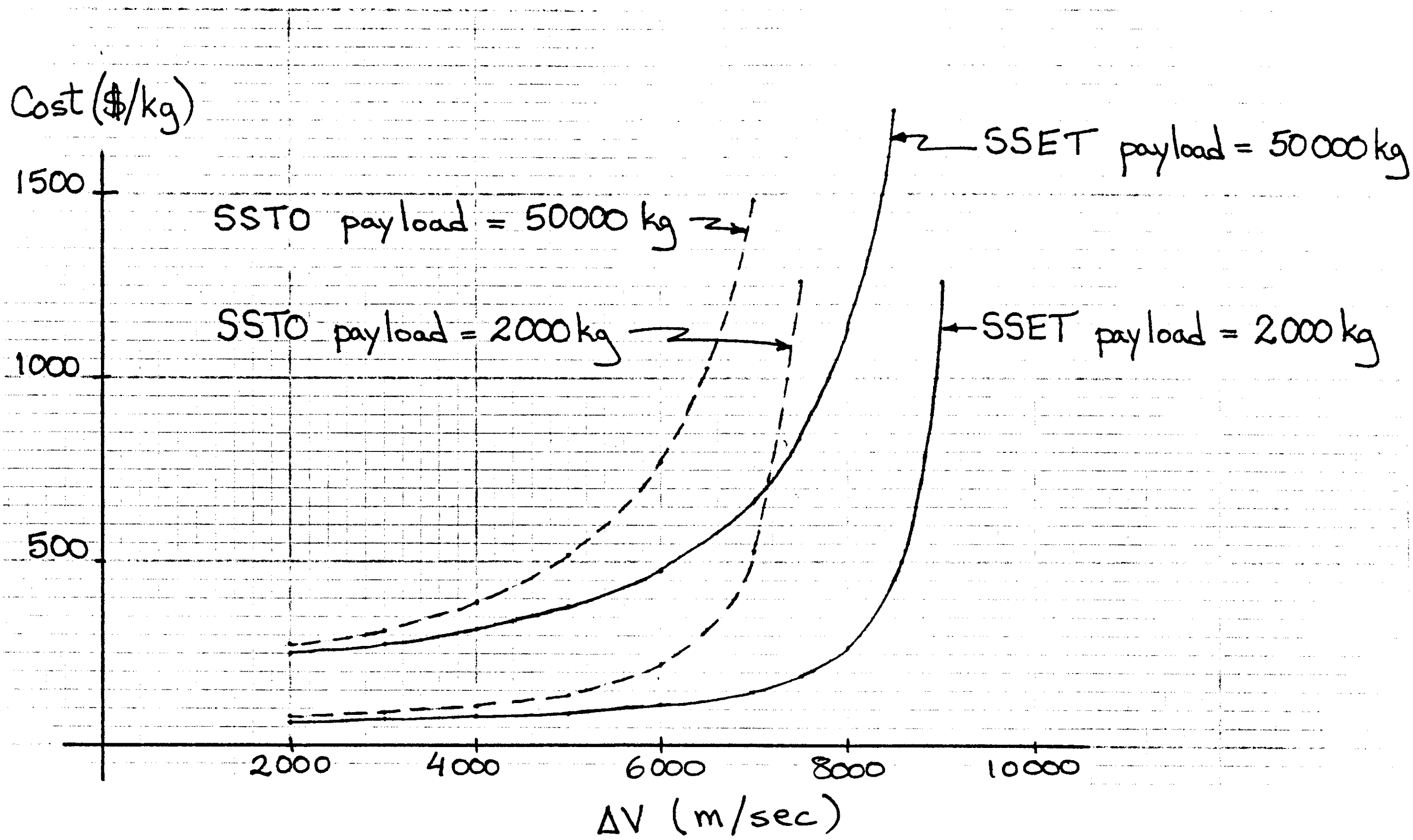

Figure 17. Cost vs. ΔV for SSTO and SSET Vehicles

3.1.1.3 Two Stage to Orbit (TSTO)

The previous sections found severe disadvantages for single stage vehicles when trying to reach earth orbit. In fact, it was found that internally tanked single-stage vehicles were totally impractical without some initial boost, while external tanked vehicles were possible, although economically unattractive. For this reason, launch vehicles in the past have invariably been multi-stage: even hydrogen-oxygen engines such as the Space Shuttle Main Engine (assumed in the preceding sections) are only marginally capable of single stage to orbit operation. Earlier engines using storable propellants were not physically capable of propelling a single stage vehicle into orbit. It seems reasonable that a multi-stage approach might offer physical and economic advantages when designing an earth launch system.

The physical equations governing the stage component masses for a two stage to orbit (TSTO) vehicle are the same as those for an SSTO vehicle, as presented in Table 2. The only difference is that one further parameter is added: the velocity at which staging takes place. The solution algorithm now becomes:

1. Select values for engine specific impulse, payload and fixed equipment masses, parametric δ factors, and configuration Δ factors for each stage
2. Choose a velocity increment for the second stage
3. Set the initial propellant tank mass estimate to zero
4. Solve the set of linear equations for component mass factors for the second stage
5. Using the derived propellant mass, update the estimate for second stage tank mass
6. Repeat steps 4 and 5 until the values of tank mass converge
7. Find the total initial mass for the second stage, and apply this as the payload mass for the first stage

8. Find the velocity increment for the first stage by subtracing the second stage ΔV from total ΔV required for orbital insertion
9. Set initial propellant first stage tank mass estimate to zero
10. Solve the set of linear equations for first stage component mass estimates
11. Using the derived propellant mass, update the estimate for first stage tank mass
12. Repeat steps 10 and 11 until the values for tank mass converge
13. With the total vehicle specified, find the payload cost for the specified mission model
14. Repeat steps 2 through 13 to find the staging velocity which minimizes payload launch costs

As can be seen by comparing this algorithm to the SSTO algorithm, it is a good deal more complex, involving nested iterations to optimize staging velocity. It has been found in this study that a good method for iterating the values of staging velocity is by parabolic extrapolation. From the initial choice of staging velocity, a vehicle analysis is performed at nominal staging, and at two off-design points spaced equidistant from the nominal case. Each of the three designs produces an estimate for payload cost to LEO.

In order to perform a parabolic extrapolation of the optimum, assume that the three points are $(x+\Delta)$, x, and $(x-\Delta)$. This would correspond to $\Delta V+\delta\,\Delta V$, nominal, and $-\,\delta\,\Delta V$ in the specific application of interest. Each of these would have with it an objective value function of y_1, y_2, and y_3, respectively. For the optimization of staging velocity, each y corresponds to the cost/kg of payload to LEO. It is desired to find the three coefficients of the parabola passing through the three sample points, which would satisfy the relation

$$(3.11)\qquad y_i = A\,x_i^2 + B\,x_i + C$$

By differentiating the basic equation of a parabola and setting it equal to zero, the extremum point is found to be at (-b/2a). Using Kramer's rule and equation (3.11), this is equivalent to

$$(3.12)\qquad -\frac{b}{2a} = -\frac{1}{2}\;\frac{\begin{vmatrix} (x+\Delta)^2 & y_1 & 1 \\ x^2 & y_2 & 1 \\ (x-\Delta)^2 & y_3 & 1 \end{vmatrix}}{\begin{vmatrix} y_1 & x+\Delta & 1 \\ y_2 & x & 1 \\ y_3 & x-\Delta & 1 \end{vmatrix}}$$

Solving the determinants in the numerator and denominator, the parabolic estimate for the optimum x is

$$(3.13) \quad X_{opt} = \frac{1}{2} \; \frac{2x(2y_2 - y_1 - y_3) - \Delta^2 (y_1 - y_3)}{\Delta (y_1 + y_3 - 2y_2)}$$

Using this estimate for optimum staging velocity, vehicle mass and cost estimates are made at this new point and at two off-nominal points, and the process repeated until the value of ΔV staging converges. At that point, the current nominal design case represents a vehicle with optimum staging characteristics. Typically, this convergence occurs within three iterations in this particular application.

Application of this algorithm is demonstrated by Figure 18 on page 95. The graphs show the cost per kilogram delivered to LEO as a function of the payload size of the vehicle, for launch vehicles capable of 100 and 500 flights. The mission model used is the current shuttle model of 350 flights with 29.5 metric tons each, for a total launch mass of approximately 10,000 tons. It is apparent from this figure that an optimum vehicle size exists for each of the reflight options. This optimum size is the result of balancing the economy of scale for larger vehicles with the increased economies of a greater production run for a larger number of smaller vehicles. Thus, although the larger vehicles have a larger payload fraction, the optimum payload size for a spacecraft capable of 100 flights is approximately 2000 kg, which would result in a fleet size of 50 vehicles. As the launch vehicle lifetime is extended from 100

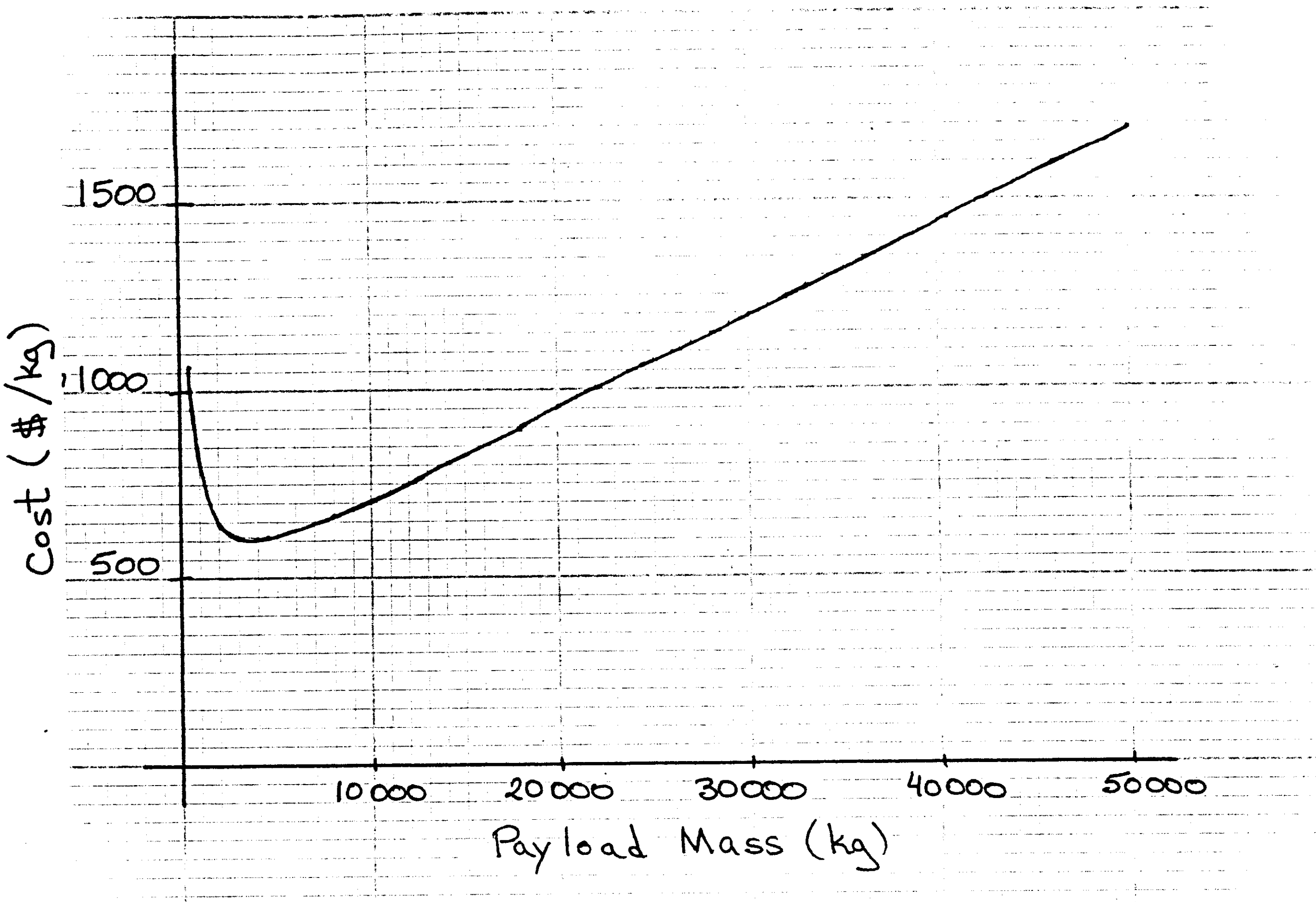

Figure 18. TSTO Cost as a Function of Payload Mass

to 500 flights, the costs go down due to higher vehicle capabilities, but at the same time production runs are limited by the longer-lived boosters. This trend pushes the optimum payload size down still further: the new optimum is on the order of 500 kg, which would correspond to a 40-ship fleet. This is clearly too small for most purposes in space, and launch requirements would force the spacecraft design off of the cost-optimum solution in favor of fulfilling spacecraft design requirements. This effect can be clearly seen in the current Space Shuttle: although it is similar in most respects to the conditions which generated the 2000 kg optimum payload of the 100-flight case, it carries 29.5 tons, which corresponds to the largest payload envisioned at the time of its design. Use of a single-design fleet is generally not commensurate with cost-optimal designs. A discussion of the effect of payload sizes in the planned mission model is presented in "Traffic Model to Low Earth Orbit" on page 194.

3.1.1.4 Two Stage with External Tanks (TSET)

As was the case with single stage vehicles, the two-stage design can incorporate external, expendable propellant tanks rather than internally mounted ones. This should decrease the size of both the first and second stage core vehicles, while increasing the recurring costs due to large numbers of expended tanks over the program lifetime.

The governing equations of component masses are as shown in Table 3-3, with modifications for external tanks as shown in Table 3-2. Three choices are available in specifying the design: external tank on the first stage, on the second stage, or on both stages. For simplicity, only the third choice is considered further in this report. The results of this analysis are shown in Figure 19 on page 98. Of interest in this figure is the fact that optimum payload sizes have increased from those for TSTO vehicles. This is due to the fact that large production runs are available in any case due to the number of tanks required, and payload sides can therefore increase to take advantage of the greater efficiency of a larger boost system.

3.1.1.5 Summary of Vehicle Applications to LEO Launch

The four launch system configurations discussed previously (single and two stage, internal and external tanks) are of interest for their ability to deliver payload into LEO in support of a space industrialization program. For that reason, it is proper to compare the systems and identify those most promising for further study.

Figure 20 on page 99 shows the relative merits of the SSET, TSTO, and TSET systems for the current shuttle program launch mass of 10,000 metric tons, assuming the vehicles are capable of 100 reflights. The SSTO configuration (single stage with internal tanks) is severely disadvantaged at orbital insertion velocities, and is totally uncompetitive with the other three

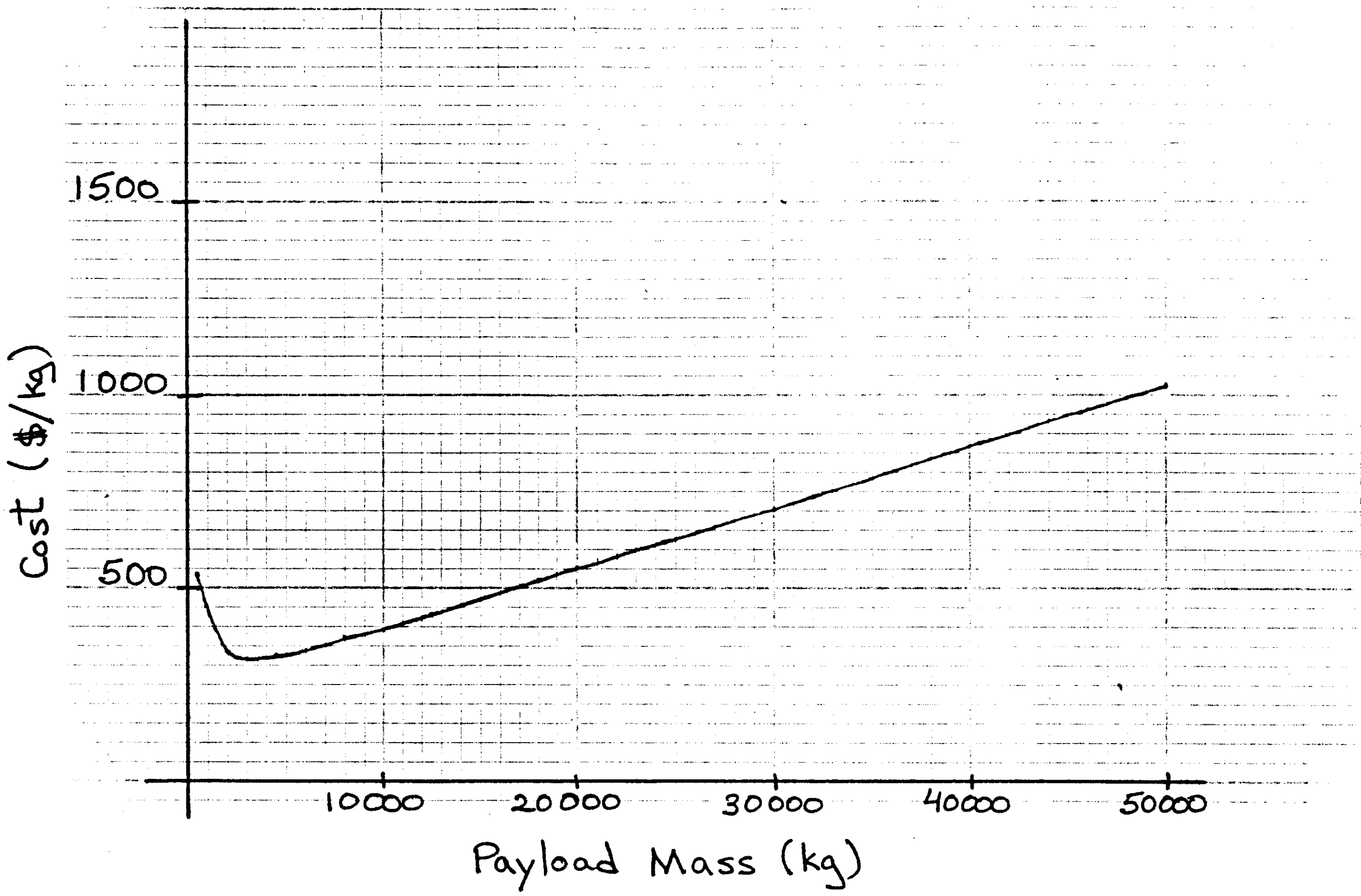

Figure 19.   Launch Costs for TSET Vehicles

systems. The values plotted for the SSET configuration are for a velocity increment of 8000 m/sec: the costs incurred with the initial 1000 m/sec increment (air launch) are not included in the values plotted.

Optimum payload size is on the order of 2000 kg, where minimum launch cost is achieved by using a two stage vehicle with internal tanks. If required payload size is greater than this, the single stage configuration with external tank has the lowest cost in the region from 4000 to 10000 kg payload size. If

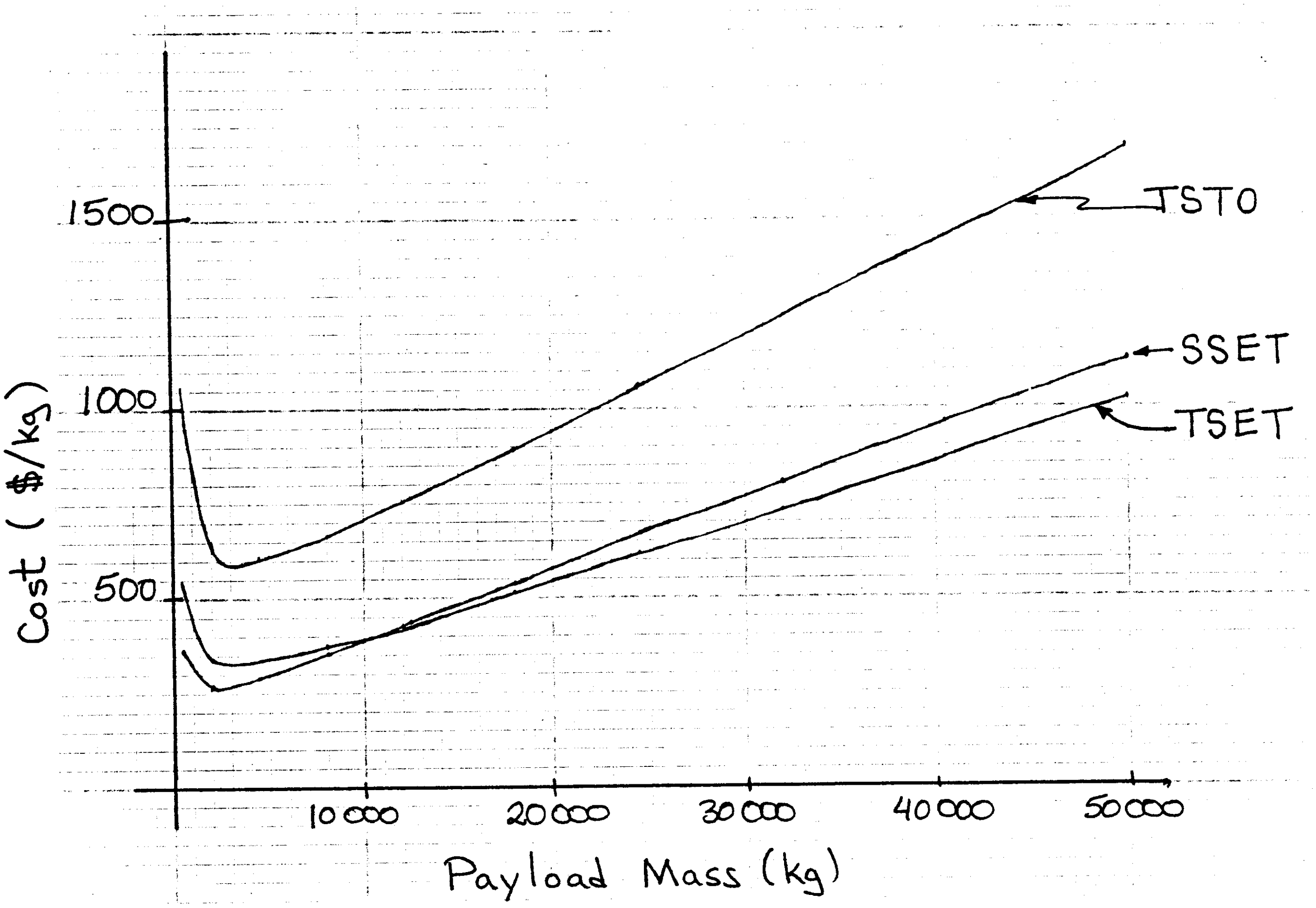

Figure 20. Launch Costs at 100 Flights/Vehicle: (Current Shuttle Mission Model)

---

the payload is forced to exceed this range, the two stage vehicle with external tanks becomes the most cost effective, due to the disadvantage of either single stage or internal tank configurations in a larger overall vehicle.

If the vehicle lifetime is extended to 500 flights, these trends change as shown in Figure 21 on page 100. In this case, the SSET and TSET cost curves cross in the neighborhood of 8000 kg payload mass. Below this mass, the SSET is favored: above

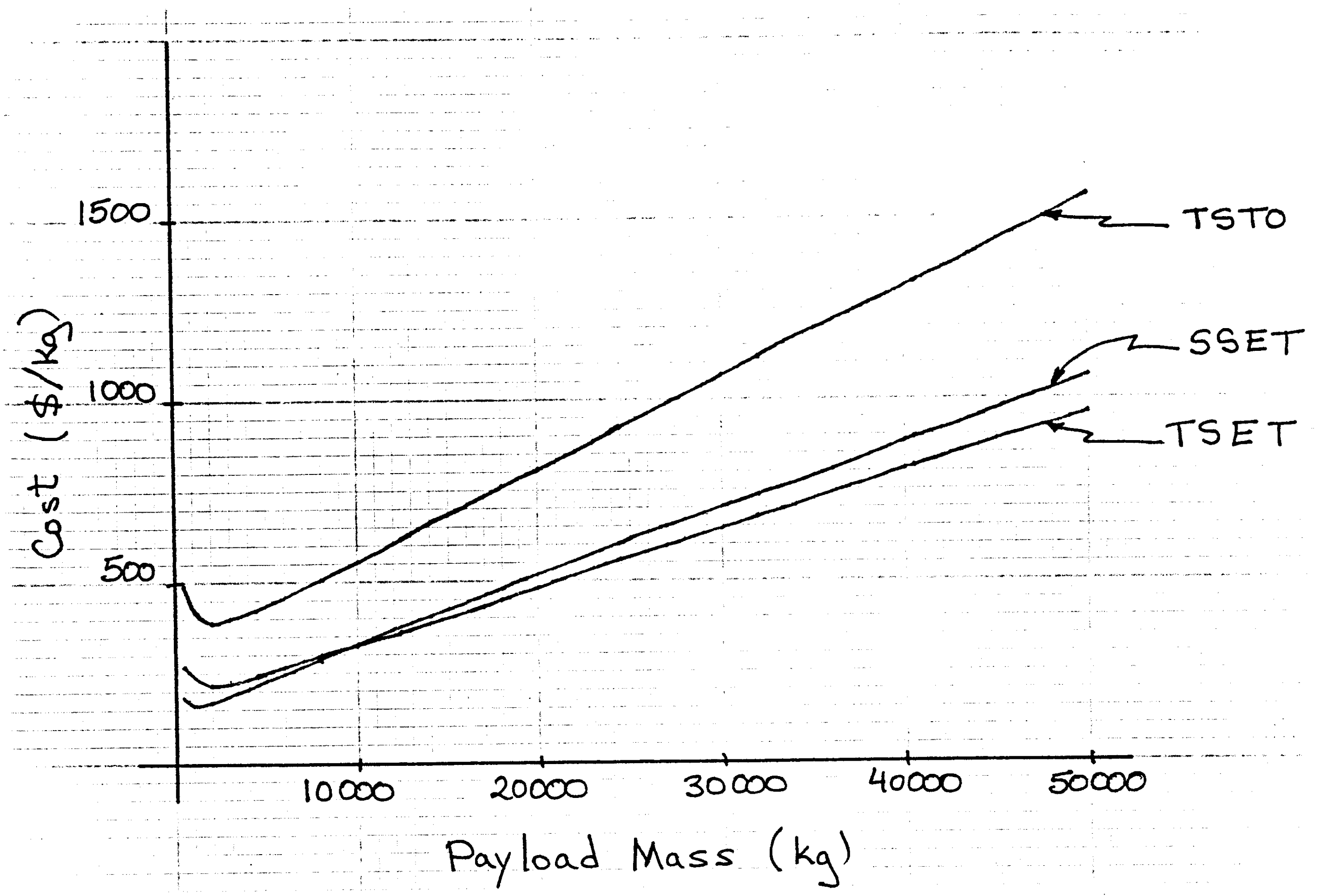

Figure 21. Launch Costs at 500 Flights/Vehicle: (Current Shuttle Mission Model)

---

it, the externally fueled two stage vehicle has the lowest cost. The two stage vehicle with internal tanks has no region of cost effectiveness in this scenario. The optimum payload size is again approximately 2000 kg.

Figure 22 on page 101 shows the minimum cost to low earth orbit for each of the systems considered, as a function of total mass launched to LEO. At each value of total launch mass, it is assumed that the vehicle is sized optimally to minimize launch

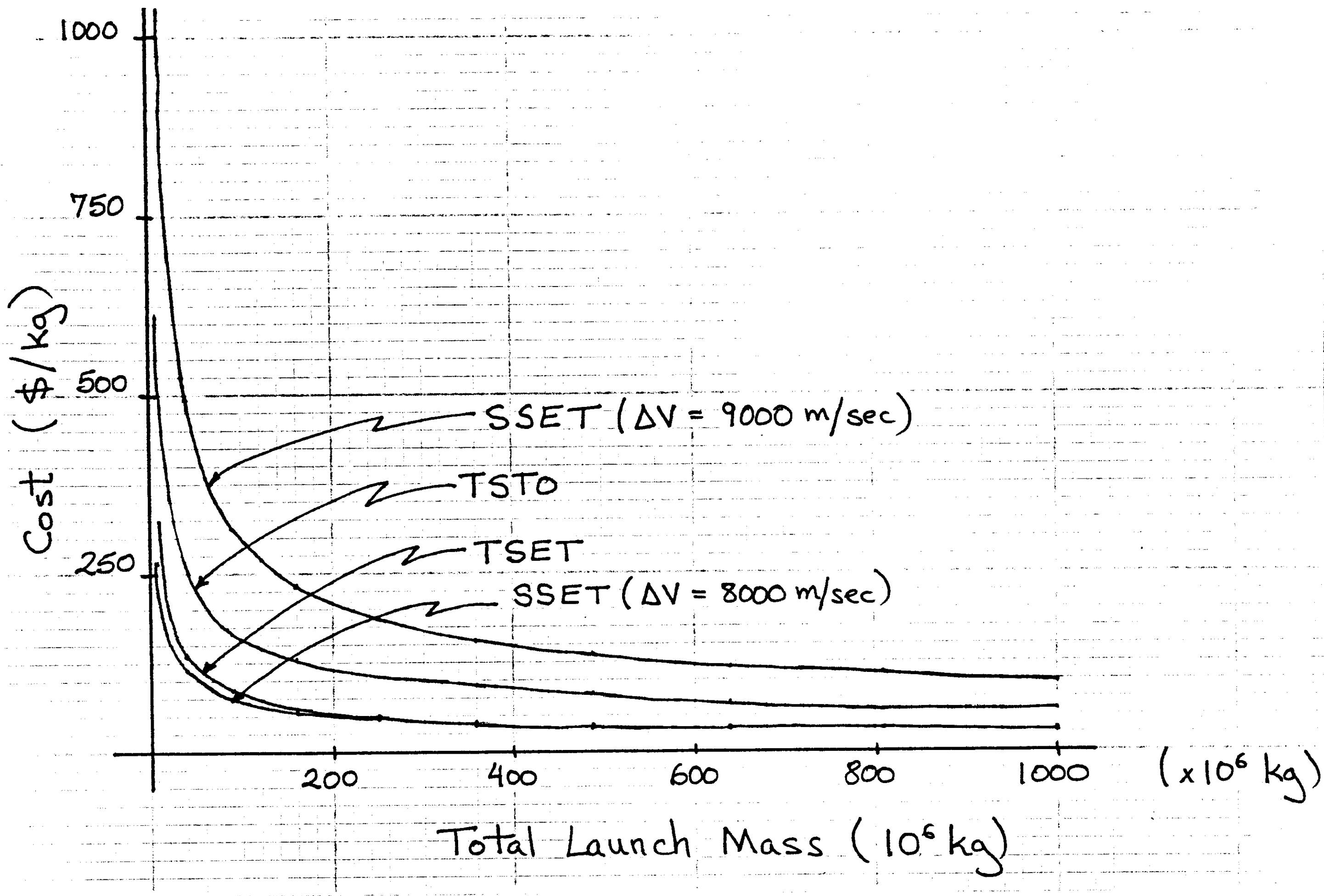

Figure 22. Costs with Optimum Payload Size as a Function of Total Launch Mass

---

costs over the entire program. As can be seen, the optimum costs go down with increasing mass to LEO. In the companion plot, Figure 23 on page 102 shows that the optimum payload size also increases as a function of total launch mass: however, the optimum vehicle size is still quite small. As an aside, current information on long-term goals of the Soviet [34],

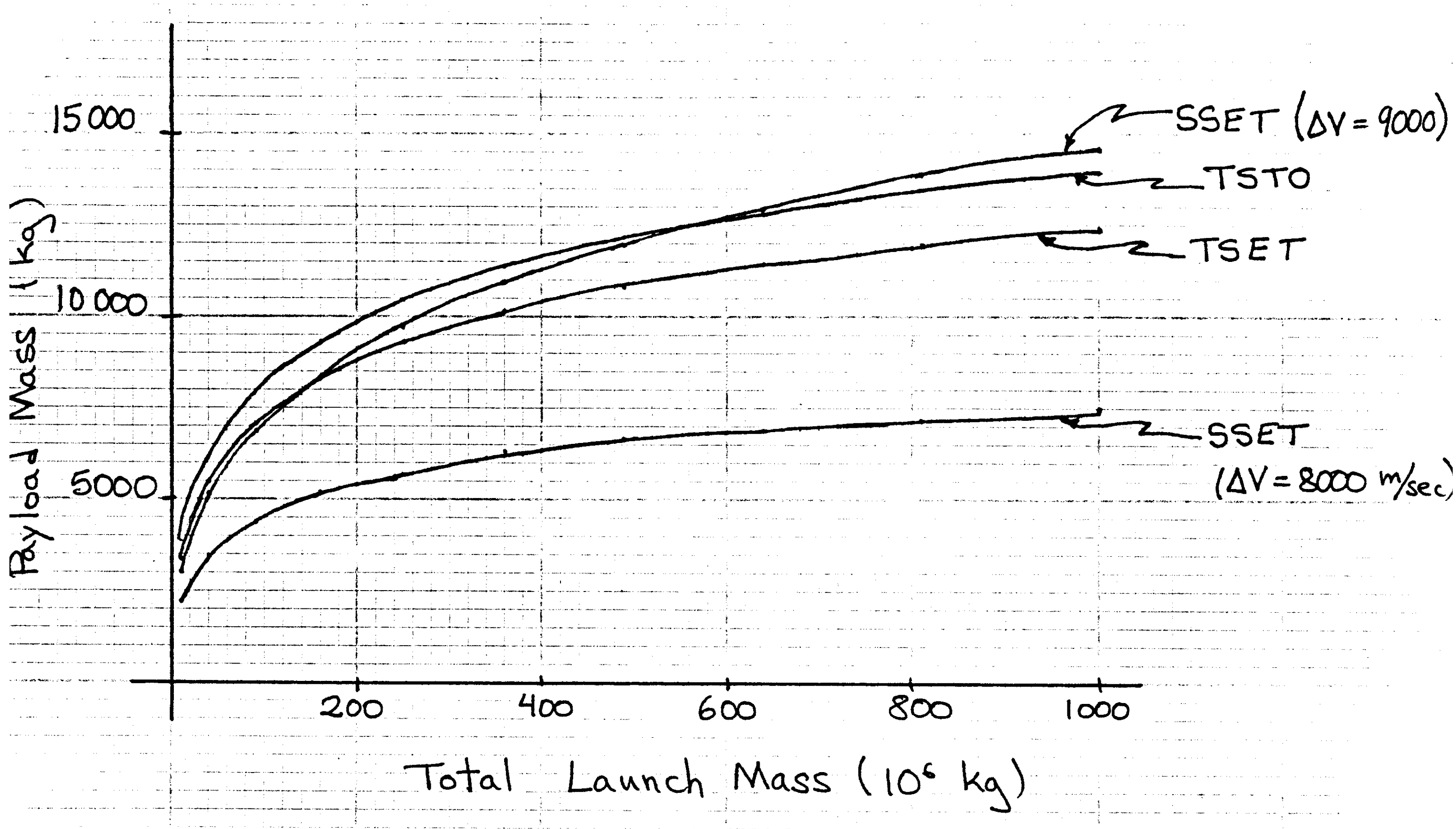

Figure 23. Optimum Payload Size vs. Total Launch Mass

Japanese [35], and European [36] space programs all include designs for reuseable manned spacecraft of the class discussed here.

3.1.2 ORBIT TO ORBIT

The requirements for orbit-to-orbit (OTO) transports are at once both simpler and more critical than those of an earth launch system. The design requirements are much less constraining, due to low acceleration levels and the lack of gravity or atmosphere. On the other hand, the vehicle must be reliable, and must fly multiple missions without the large crews available for checking out and turning around a launch vehicle after each flight. If the OTO transport is large, provision must be made for assembling it and performing initial checkout on-orbit.

Due to the design environment, it can be assumed that an OTO vehicle consists of only four components:

- Propulsion
- Propellants
- Structure
- Payload

The structure need consist of little more than the tanks for holding the propellants. The propellants are sized in accordance with the velocity increments required, as calculated in the preceding chapter. The payload is the independent parameter, and may be different on each flight of an orbital transfer vehicle (OTV). It is therefore the propulsion system where the important parameters and choices are to be found. For this reason, a closer look should be taken at the possible propulsion systems for the OTO application.

The critical parameters to be estimated for each propulsion system are its performance (specific impulse, or I_{SP}), mass, and scaling functions. As will be seen, choice of acceleration levels or exhaust velocities can strongly influence transport costs, and the effects of variations in these parameters should be known prior to system optimization.

3.1.2.1 High Thrust Systems

High thrust OTO vehicles are those with thrust/weight ratios of the order of 1, and which can therefore use the impulsive ΔV's derived previously. This is the class where there are existing examples of orbital transfer vehicles: the payload assist modules and inertial upper stage are both examples of high-thrust OTV's. However, they will not be examined in depth in this analysis, as neither of these systems is reuseable. The first such vehicle, and probably the first to actually merit the name of OTV, is the modified Centaur upper stage, currently under development. Original versions of the Centaur will also be expendable, but the system should be adaptable to being reused.

Chemical Systems: Chemical propulsion systems are the simplest to use for orbital transfer vehicles, as almost all experience with propulsion systems to date has been with this type of propulsion system. Chemical systems could be further broken down into liquid and solid propellant systems, but solids seem to be at a severe disadvantage in a space industrialization system. None of the available lunar materials seems to be suited to use in a solid propellant grain, and the procedure of refurbishing and refilling a used solid rocket motor in weightlessness would be difficult at best. For this reason, only liquid propellants will be considered here.

Chemical propellants could be further broken down into storable and cryogenic propellants. The storables, such as unsymmetrical dimethylhydrazine/monomethyhydrazine ("Aerozine 50") and nitrogen tetroxide, offer the advantages of high density, easy long-term storage, and hypergolic reaction (no need for an engine igniter); they are unfortunately difficult to manufacture, consist largely of nitrogen, which is unavailable on the moon, and do not have the performance of liquid hydrogen/liquid oxygen engines.

The Centaur stage used RL-10 engines, which are the first generation of LH /LO engines. Designs for OTV engines based on space shuttle main engine technology have the potential for substantial performance increase over the Centaur RL-10's. Based on Rocketdyne designs for the advanced space engine (ASE), an LH /LO engine powering an OTV would have a specific impulse of 473 seconds, and a thrust/weight ratio of 52:1.[37] The respective densities of liquid oxygen and liquid hydrogen are 1140 and 64 kg/m^3 at a nominal mixture ratio of 6:1, this corresponds to an average propellant density of 218 kg/m^3.[38] From [32], the mass of a liquid oxygen tank is 1.5% of the mass of propellant it contains. Since hydrogen is much less dense, the tank must be correspondingly larger to contain an equal amount, and a hydrogen tank therefore has a mass equal to 12.1% of the internal fuel. Again assuming a 6:1 oxidizer-fuel mass ratio, the effective propellant tank mass is 3% of the propellant mass. This data is based on the Saturn series of launch vehicles. Although the tanks could certainly be built lighter to accomodate the space environment, the full value of 0.03 will be used to compensate for the mass of insulation, and any non-tank mass required on the OTV. This parameter of empty mass over full mass of the tank is the tank mass fraction, or ε.

Nuclear: With the expanding space program of the middle 1960's, advanced planning for manned interplanetary exploration included the development of a capability for nuclear propulsion in orbit. With the KIWI and NERVA series of reactors, NASA developed the technology to replace the hydrogen/oxygen J2 engine on the S4B stage of the Saturn 5 with a nuclear engine. Unfortunately, the funding for this program was cut before a flight article could be built. Based on the NERVA experience, an operational nuclear engine for OTO applications would have a specific impulse of 825 seconds, and a thrust/weight ratio of approximately 1. [39] This baseline data is for liquid hydrogen as a propellant, and tank mass is derived from the discussion above.

Since propellants brought from earth incur the same launch costs as the rest of the materials, there is a strong incentive to find nonterrestrial sources for propellants. Much of the lunar samples from Apollo was composed of oxygen: typical ore might run as high as 40% oxygen by weight, bound in metallic oxides.[15] Calculations will show that life support requirements would only take a few percent of the oxygen extracted in the course of refining the lunar materials into metals, so that oxygen is a waste product of the refining procedure. It might be argued that, with some effort, a nuclear engine could be modified to use oxygen as a propellant rather than hydrogen. Since this would involve heating the oxygen to a high temperature, care would have to be taken to prevent oxidation of the engine surfaces. Since exhaust velocity (and therefore specific impulse) scales as the inverse square root of molecular weight [40], the effect of going from hydrogen (M=2) to oxygen (M=32) would be to cut the specific impulse down to 200 seconds. On the other hand, the greater density of oxygen would allow a lighter propellant tank per unit mass of propellants. Design mass of the NERVA flight engine was 36 metric tons. Based on the estimates of [41], the research and development costs leading up to the first flight unit of a NERVA engine would be approximately $1.25 billion. Due to the problems of high-temperature oxygen around the uranium fuel rods, it is assumed that the development cost of a NERVA-class engine using oxygen propellant would double that of the baseline, for a total of $2.5 billion.

Advanced Nuclear: For a thermal rocket engine, the performance (as measured by specific impulse) is limited by the available temperature in the engine. The performance increase of nuclear engines over chemical engines is that greater energy is available from the nuclear reaction than is available from chemical bonding energy. The propellant can therefore be heated to a higher temperature, which results in a higher exhaust velocity and specific impulse.

Current limitations on propellant temperature are set by allowable temperatures of the fuel rods and moderator material. If these were allowed to get hotter, the propellant could be heated more, and higher performance would result. This is the reason behind the concept of the gas core nuclear rocket (GCNR).

In the original concept, the fuel elements were allowed to heat until they reached the gaseous state, after which they would be contained in a bound vortex by the flow of the propellant through the reactor vessel and out the nozzle. However, nuclear fuel is expensive enough that even small loss rates were prohibitively expensive, and other techniques were suggested. In one of the most advanced ideas, the uranium would be held in a bound vortex inside a "light bulb": a cylinder of fused silica. This glass window would be cooled by neon flow (also used to keep the uranium vapor in a vortex), and the propellant flowing past on the outside of the tube would be heated by radiation. [42] Using a technique such as this, an engine using liquid hydrogen as a propellant would have a specific impulse of about 2000 seconds, at a thrust/weight ratio of 0.2. Engine performance must again be degraded by a factor of 4 if liquid oxygen is used. Costs are estimated from [41] to be $2.5 billion for the liquid hydrogen option and $4 billion for liquid oxygen as a propellant.

One concept which may be even more advanced than that of the GCNR is the fusion rocket engine. Although most fusion concepts are too bulky for use in a spacecraft, one concept under study

[43] uses aerospace techniques for high-temperature design, and is aimed at producing a small, expendable fusion reactor without the need for cryogenics and superconducting magnets. This device, called the riggatron, is currently estimated at 3600 kg (8000 pounds) for a fusion power output of 1200-1300 MW. Assuming this energy can be transferred to the propellant at an efficiency of 70%, calculations indicate a thrust/weight ratio of 1.5 at a specific impulse of 3000 seconds.

Table 3-4 summarizes the parameters for the proposed impulsive thrust systems. In this table, f refers to the mass fraction of propellants which are available only from earth. While the numerical values in this table are based on published data, they must still be considered speculative for the cases of the advanced fission and fusion engines. Program implementation should therefore allow for a variation of engine parameters to find the sensitivity of the solution to engine parameter estimates.

3.1.2.2 Low Thrust Systems

The low thrust propulsion systems are characterized by operating continually during the transfer mission, as opposed to the

TYPE	I_{sp}	T/W	ϵ	F_E	R+D Cost ($M)
Chemical	473	52	.030	.14	10
Nuclear (LH_2)	825	.94	.121	1	1250
Nuclear (LO_2)	200	.94	.015	0	2500
GCNR (LH_2)	2000	.2	.121	1	2500
GCNR (LO_2)	500	.2	.015	0	3500
Fusion	3000	1.5	.121	1	2500

Table 3-4: Impulsive Thrust Propulsion System Summary

intermittant use which is made of a high-thrust system. Most of these propulsion systems depend on an outside power source, rather than powered stored internal to the propellant (chemical) or engine (nuclear). Therefore, the study of potential types of low-thrust propulsion can be broken down into two separate systems: propulsion and power.

Propulsion: A continual-thrust propulsion system generally works by accelerating a very small mass to a very high velocity, and exhausting it from the vehicle to provide thrust. This is done either on a microscopic level, with plasma or charged particles as the exhaust, or on a macroscopic level, as is the case with mass drivers or rail guns.

A number of different designs for particle accelerator thrusters have been studied over the past twenty years. An overview of the literature as of several years back is available in reference [44]. For the purpose of this study, this variety of types will be represented by three systems: ion using noble gases, ion using oxygen, and magnetoplasmadynamic.

Ion engines using earth-based propellants are among the most well developed of low thrust systems, and have been baselined for use on the Solar-Electric Propulsion Stage (SEPS) of the Space Transportation System. Most of the initial research in this field concentrated on the use of cesium or mercury as a propellant, due to the ease of ionization and acceleration of the ionized particles. However, this would be impractical in a large-scale industrial process, as both propellants are rare and expensive: mercury is also hazardous to humans in small concetrations, and exhaust impinging on the earth's atmosphere could create a long-term health hazard. For this reason, recent efforts have concentrated on the use of alternate propellants such as xenon and argon. [45] While these propellants would solve the polution problem of mercury, they are still not optimal, since they are only available from the earth. For large-scale transport in space, it seems reasonable to maximize the use of nonterrestrial resources, especially for expendables such as propellants, in order to reduce the burden of launch costs from the earth. About the only gas available in large scale from lunar resources is oxygen, which is a waste product from the refining procedures. If oxygen could be used as an ion engine propellant, significant reductions in orbit-to-orbit transport costs could be realized. For this reason, both of these propellant options for ion engines will be used as available propulsion systems for the sake of this analysis: the defining parameters of the two systems are taken from [46]. The other particle system in competion is the magnetoplasmadynamic (MPD) thruster. Defining parameters for this system are taken from [47]. This system, while better suited to the high power levels of OTO transport applications than ion engines, are not suitable for use with oxygen as a propellant. Argon will therefore be assumed for this system.

The last propulsion system under consideration for use in the continual-thrust orbital transfer vehicles is the mass driver reaction engine (MDRE). This system, as described in [48], is a long track which accelerates superconducting "buckets" containing the reaction mass. The track itself is a linear synchronous motor, capable in theory of accelerating the bucket to any desired velocity. Upon reaching the desired exhaust velocity, the payload is allowed to leave the MDRE, while the buckets are retained, decelerated (restoring most of the acceleration energy via regenerative braking), and recycled for further use. The advantages of this system are that the exhaust velocity can be varied as necessary for specific impulse matching to the system, and that any unwanted mass can be used as propellant mass. The disadvantages of the system are that, for a fixed acceleration, the length of the system varies as the square of the exhaust velocity (length = velocity $^2/2 \times$ acceleration). Furthermore, use of solid matter in the exhaust can create a navigation hazard, as particles can become trapped in the earth's gravitational field and form a stream of massive meteoroids. It is therefore much better to use a propellant which will dissipate, such as liquid oxygen.

For a continual-thrust propulsion system, the engine mass is assumed to be a linear function of both the mass flow rate and the square of the exhaust velocity:

(3.14) $$M_{eng} = \phi \dot{m} + K c^2$$

The $\dot{m}$ term reflects the possibility of clustering the ion or MPD engines if greater thrust is needed. The c^2 term represents the incremental acceleration needed for additional exhaust velocity from the mass driver. Further associated with each engine type is the mass fraction of the tank for its required propellant. As in the case of high-thrust systems, this propellant tank mass fraction will be referred to as ε. Sizing parameters for each of the propulsion system candidates can be seen in Table 3-5.

Power: Possible power generating sources include solar and nuclear. Most spacecraft to date in the earth-moon system have

Engine Type	Efficiency	$\dot{m}$ scaling Factor $(kg/\frac{kg}{sec})$	mass flow rate (kg/sec)	exhaust velocity scaling factor $(kg/\frac{m^2}{sec^2})$	Specific impulse (sec)	propellant tankage Fraction	earth propellant Fraction
Ion/Argon	.65	1.67×10^6	3×10^{-5}	0	8000	.09	1
Ion/Oxygen	.65	1.67×10^6	3×10^{-5}	0	5000	.015	0
MPD	.57	4.1×10^3	6×10^{-3}	0	5000	.09	1
MDRE	.95	0	.07	.0014	500	.015	0

Table 3-5: Low-Thrust Propulsion System Parameters

---

used solar photovoltaic systems, which are reliable and long-lived, if not exactly inexpensive. Solar thermal systems are also a possibility, but in general require high pointing accuracies, and mechanical parts such as pumps. Details on solar cell performance, which will be used as the assumed preferred method of solar power collection in this report, are detailed in [41].

Nuclear energy has been used in spacecraft intended for outer-planet exploration, where the inverse-square law of radiation would severely limit available power from solar energy. In most applications to date, the decay of radioactive material has been used (through the heat generated in the decay process) to drive solid-state thermal generators. These devices, known as radiothermal isotopic generators (RTG's), have been used on Mariner, Voyager, Viking, and the Apollo

Lunar Scientific Experiment Package (ALSEP). While long-lived, these units generally do not provide sufficient power for a propulsion system. Recent work [49] has resulted in the design of a nuclear reactor capable of supplying 1200 kw (thermal) with an operational lifetime of 7-10 years.

The power needed for the propulsion system is a function of the kinetic energy of the exhaust, and the efficiency of the engine:

$$(3.15) \quad P = \frac{\dot{m} \; c^2}{2 \eta}$$

Characteristics of the power generating systems under consideration in this study are presented in Table 3-6.

Vehicle Sizing: With the selection of both the propulsion and the power generating systems, the orbital transfer vehicle can be sized for later cost analysis. The vehicle mass ratio is defined as

$$(3.16) \quad r = e^{-\frac{\Delta v}{c}}$$

and is equal to

$$(3.17) \quad r = \frac{M_o - M_{prop}}{M_o}$$

System	Specific Power (kg/kW)	Cost ($/kW)
Solar Photovoltaic	10	50 000
Solar Thermionic	40	4000
Nuclear	15	7500

Table 3-6: Power Generation System Parameters

As mentioned earlier, the component masses of the OTV are engine, power, tank, and payload masses. Equation (3.17) can be rewritten as

$$(3.18) \quad r = \frac{M_{eng} + M_{str} + M_{power} + M_{PL}}{M_o}$$

Some algebra will show that the initial mass of the vehicle is

$$(3.19) \quad M_o = \frac{1}{r}\left[M_{PL} + \alpha P + \phi \dot{m} + \kappa c^2 + M_o \gamma (1-r) \right]$$

or, solving explicitly for initial mass,

$$(3.20) \quad M_o = \left(M_{PL} + \alpha P + \phi \dot{m} + \kappa c^2 \right) \left[r - \gamma (1-r) \right]^{-1}$$

Since the burn time is equal to the trip time for a low-thrust trajectory, the overall travel time τ is equal to

$$(3.21) \quad \tau = \frac{M_o(1-r)}{\dot{m}}$$

It should be mentioned that this analysis procedure follows that of reference [16].

3.1.2.3 *Propellant Fraction*

Whether an impulsive or continual thrust system is used, it is necessary to know the relation between payload and propellant required to transport it. This will then determine the propellant that will be transported as payload throughout the transportation system, to be used as propellant in a later leg of the journey. The relevant parameters can be defined as

ι	Inert mass (propulsion, power systems) / gross mass
δ_T	Tank mass / gross mass
λ	Payload mass / gross mass
ζ	Inert mass / payload mass
ξ	Propellant mass / payload mass

Thus, ξ is the parameter of interest, as it relates the mass of propellant needed to the mass of payload carried. The total system can be specified by

(3.22) $$r = \tau + \delta_T + \lambda$$

This equation can be rewritten in terms of ξ:

(3.23) $$\xi = \frac{1}{\lambda} - \frac{\tau}{\lambda} - \frac{\delta_T}{\lambda} - 1$$

Although the tank mass is present in the form of δ_T, it would be preferable to express it in the form of tank fraction ε. Rewritting the definition of δ_T as

(3.24) $$\delta_T = \frac{m_T}{m_T + m_P} \quad \frac{m_T + m_P}{m_G}$$

the replacement value for δ_T can be found to be

(3.25) $$\delta_T = \frac{\epsilon(1-r)}{1-\epsilon}$$

In a similar manner, chain-rule multiplication can be used to find that

(3.26) $$\frac{1}{\lambda} = \frac{1}{1-r}\,\xi$$

Substituting (3.25) and (3.26) into (3.23), and utilizing the definition of ζ, the propellant/payload fraction ξ can be specified as

$$(3.27)\quad \xi = \frac{1}{1-r}\,\xi - \zeta - \frac{\epsilon}{1-\epsilon}\,\xi - 1$$

which can be simplified to find an explicit value for ξ:

$$(3.28)\quad \xi = \frac{(1+\zeta)(1-r)(1-\epsilon)}{(r-\epsilon)}$$

In this formulation, r is fixed by the velocity interval, ε is fixed by the type of propellant, and ζ is the ratio of ship inert mass to payload mass. With these parameters, the ratio of propellant to payload mass can be determined, and the actual payload mass ratio for the transfer can then be calculated using equation (3.26).

3.2 HABITATION AND LOGISTICS

Although not as intricate as the transportation problems studied earlier, the parameters describing crew logistics and habitation requirements do impact the overall system cost. Reference [50] represents the state-of-the-art in partially closed cycle life support systems, and is fairly typical of what might be available for use in the time period of interest. This system, designed for the Space Operations Center (SOC), recycles water from wastes and atmosphere, and scrubs CO from the air. All life support make-up is done with hydrazine (which supplies both nitrogen, and hydrogen for water make-up). Basic consumables stock in the operating system is 690 kg/person, with a refurbishment rate of 2 kg/person-day. Of this mass, 1.1 kg/person-day consists of food, with the rest being environmental life control system make-up consumables. The power supply on board is 4.4 kw/person, which supplies both life support and some experiment power.

Habitat module parameters are also based on the SOC estimates. Basic volume requirements are 40 m^3 /person, including the volume occupied by equipment and furnishings. Of perhaps more interest due to the preliminary nature of this analysis procedure is the unit mass of the habitat and associated support and logistics modules. Based on SOC working papers [51], the appropriate mass is 8000 kg/person, including all necessary equipment such as docking systems, transfer tunnels, reaction control, and instrumentation.

4.0 LINE ITEM COSTING

"I'm not in this for you, sister, and I'm not in it for your precious revolution. I expect to be well paid."

"You needn't worry about that. If money is all you love, than that's just what you will receive."

- Han Solo and Princess Leia Organa

In the last chapter, systems were defined which would play a role in the industrialization of space. Given the parameters which define a transportation system, production technique, or support facility, all of the components must be brought together into the overall system. Possible choices between competing systems must be made in a rational and quantitative manner, so as to maximize the viability of the total enterprise.

If industry is to be brought into space (as is implied in the term industrialization), then the techniques and interests of industry must be addressed in the program planning. Therefore, the objective function is usually to maximize profits or cost-benefit ratio in a realistic estimate of the business climate. This would include factors such as the cost of capital, which is reflected in the cost discounting techniques introduced earlier.

The primary thrust of this effort is to identify a method by which choices between competing systems can be made logically, in order to find an optimal or advantageous course to follow over the long term committtments of a space program. This course should lead the user into the correct investments in the competing options for industrial growth, and result in both maximized profits and expanding industrial capabilities in space.

In order to identify the most attractive program options for optimization, a line item costing program is used to find the total cost for a system under a set of assumptions. By varying the assumptions (type of launch vehicle, placement of facilities, etc.), the effect of the system parameters on the overall program cost can be estimated. The result of this exercise will be a set of possible programs which seem to offer significant advantages over competing programs. It is at this point that the programs can be compared using the operations research technique developed in the following chapter.

4.1 INDUSTRIAL MODEL DEFINITION

In order to quantify the decision process, it is necessary to develop a costing model which includes all of the significant factors affecting the system. By setting up a parametric model, the effect of changes in a parameter may be studied in detail, in order to provide information on the effect of uncertainties in the estimation process. In order to be as useful as possible, the model must remain accurate while being generally applicable. For example, the solar power satellite (SPS), as discussed in the introduction, presents at once both the largest and the most lucrative of the possibilities for space industrialization. Some of the choices which may be used for the production of SPS units are

- Assembly in LEO from terrestrial parts, and transport of the completed units to GEO
- Transfer of terrestrial parts from LEO to HEO, assembly in HEO, and transport of the completed units from HEO to GEO
- Use of lunar materials
  - Location options
    - Lunar refining and manufacturing, with assembly in LO or HEO
    - Lunar refining, with manufacturing and assembly in LO, HEO, L1, L2, or L4

  - Lunar mining, with refining, manufacturing, and assembly in orbit
- Placement options
  - LO
  - L1
  - L2
  - L4
  - HEO
  - GEO
- Transport options
  - Chemical proulsion with all propellants brought from earth
  - Chemical propulsion with earth hydrogen and lunar oxygen.
  - Chemical as above with noble-gas electric propulsion for cargo
  - Chemical as above with lunar oxygen electric propulsion for cargo
  - Use of electromagnetic launchers (massdriver, rail gun) for launch of raw lunar ores
  - Use of electromagnetic launchers as reaction engines
  - Advanced concepts (nuclear, transmitted electric, light sails)

As can be seen, a large number of options are available in the design of a space-based production system.

Assessing the costs of logistics support for a program such as the SPS ranks among the most complex problems of systems analysis. With the available combinations of transport methods, production tooling, crew habitat designs, equipment procurements, and various other parameters, the production scenario under consideration is difficult to completely specify, much less analyze. There is obviously a tradeoff in the systems analysis: whether to model the system with more parameters, generally producing higher accuracy, or fewer, resulting in quicker model verification and more manageable algorithms. The approach taken here is an attempt to compromise between the two extremes.

As described in the introductory chapter, there are nine locations of interest in the earth-moon system:

ES	Earth Surface
LEO	Low Earth Orbit
GEO	Geostationary Orbit
HEO	High Earth Orbit
L4	Leading stable Lagrange point
L2	Lunar farside Lagrange point
L1	Inline Lagrange point between the Earth and the moon
LO	Lunar Orbit
LS	Lunar Surface

The transportation network possible for such a system is diagrammed in Figure 24 on page 128. Limitations on the system are associated with launch conditions: earth surface is accessible only through low earth orbit, and the lunar surface is similarly accessible only through lunar orbit. If a massdriver is used for lunar materials launch, then the nearby Lagrange points L1 and L2 may be reached directly from the lunar surface, but this is a one-way connection (dotted lines). The seven in-space locations are all equally interaccessible, with the ΔV's derived in Chapter 2.

For transport between the various nodes of the system, either direct or indirect paths may be followed. For a fully interconnected system of n nodes, there is obviously 1 direct path between any two points, and (n-2) paths which have one stopover. Assuming that there is no return to original points, at each of the (n-2) intermediate stops there are (n-3) further stopping points between the first intermediate point and the final one. Therefore, there are (n-2)(n-3) connecting paths between two points which feature two intermediate stopping points. Assuming that no point may be revisited, this recursive analysis will show that the total number of possible connection paths between any two points in a system of n total points is

$$\text{Paths} = 1 + \sum_{k=2}^{n-1} \prod_{i=2}^{k} (n-i) \tag{4.1}$$

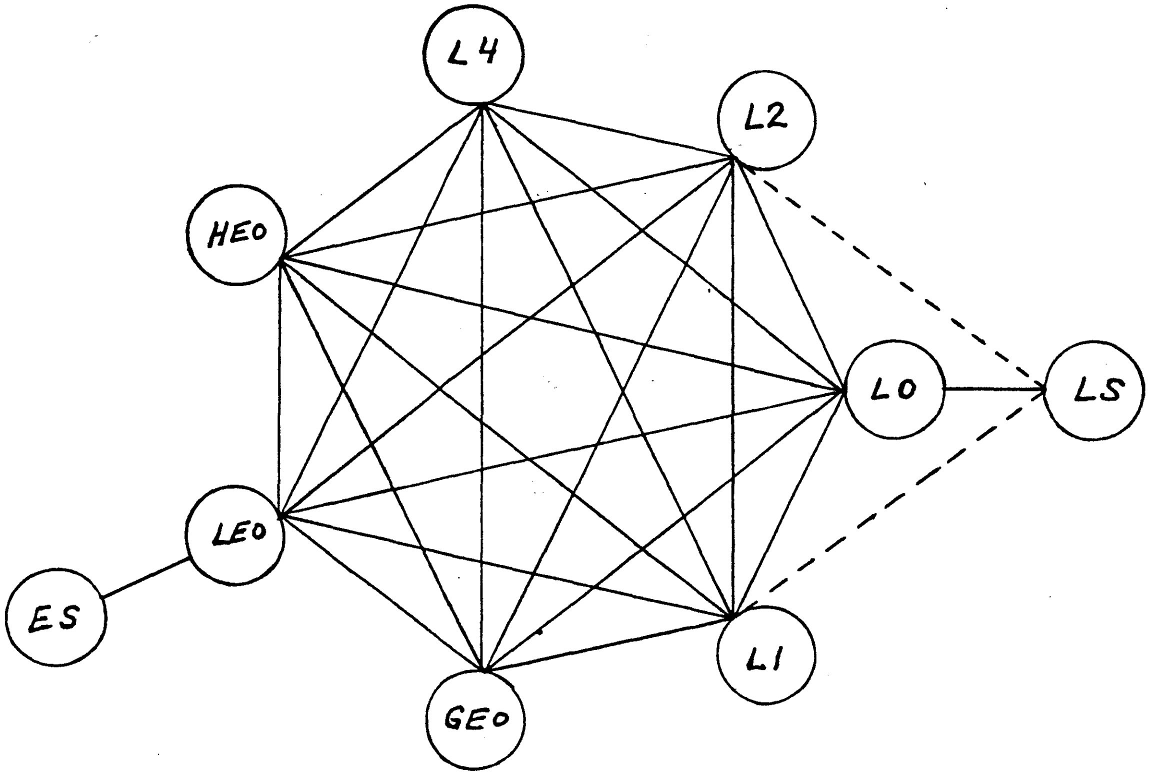

Figure 24. General Transportation Network

If revisits to intermediate points are allowed, the number of possible connecting paths becomes

$$\text{Paths} = \sum_{i=0}^{k} (n-2)^{k} \tag{4.2}$$

The parameter k in (4.2) is an independent variable: since there is no restriction on the path returning to a previously visited intermediate point, there is also no restriction on the number of stops made between the initial and final nodes. In the more restrictive case of (4.1), there are still 326 possible paths between any two points.

In order to simplify the analysis procedure, the transportation model used in this analysis is shown in Figure 25 on page 130. In this model, the production procedure is broken down into four processes, as described in Chapter 1:

- Mining
- Refining
- Manufacturing
- Assembly

The presence of the fifth site is to account for instances where assembly does not take place at the same location as the eventual destination of the product. Between each location, there is a specified round-trip path over which the production goods flow. This is in general assumed to be the low-thrust path without intermediate stops: if, however, the lunar surface is the mining site, a high-thrust transport path must be substituted instead. The round-trip paths between low earth orbit and each site are the flow paths for crew, logistics, and

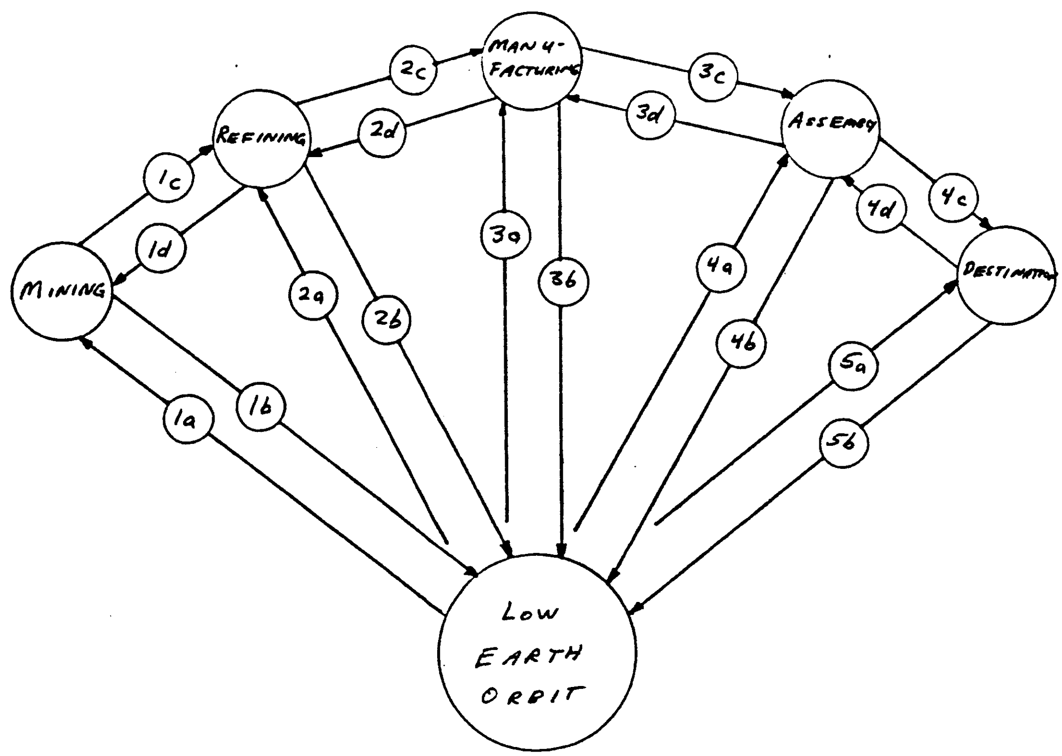

Figure 25. Production System Transport Model

manufacturing materials from earth. Each of these is actually two round-trip paths: a low-thrust transport is used for bulk goods, and a high-thrust system transports crew and consumables. Since only the low-thrust system is assumed between sites, no personnel in general travel between the production locations: again, the only exception to this is for a lunar mining base, where a refining base in lunar orbit, for example, might act as a way station for crew bound for the lunar surface.

With this system, therefore, it is possible to identify three types of parameters in the system:

Site A variable associated with the location in space

Process A variable associated with a particular production step

Global A variable which applies throughout the system

The site variables generally are based upon the physical environment of the location itself, rather than any dependence on the processes performed. The site variables identified in the current model are:

R_{c_j} Number of times/year that the crew is rotated back to earth (generally based on the radiation environment)

L_{c_j} Effective productive lifetime of a crew at the site (based on cumulative radiation dose)

f_{L_j} Fraction of time the site is in sunlight (used for solar-powered systems)

These three parameters are applied to each of the 9 possible sites, resulting in 27 variables. The values used for this study are listed in Table 4-1.

The process-specific independent parameters are:

Site	Crew Rotation	Daylight Fraction	Working Lifetime
ES	0	.5	50
LEO	2	.7	20
GEO	6	1	5
HEO	4	1	20
L4	4	1	20
L2	4	1	20
L1	4	1	20
LO	4	.7	20
LS	2	.5	30

Table 4-1: Site Parameter Values

P_{ci}: Crew productivity (kg/person-hr)

P_{mi}: Machine productivity (kg output/hr-kg machine)

E_{mi}: Machine power (kWhr/kg output)

f_{Ti}: Throughput fraction (kg out/kg in)

f_{Ei}: Fraction of output from Earth (kg Earth/kg output)

M_{PLNi}: Payload mass of vehicle to next site

O_i: Output mass (kg)

I_{Li}: Lunar input mass (kg)

I_{Ei}: Earth input mass (kg)

T_{Wi}: Working year (person-hr/year)

C_{Pi}: Production crew (people)

C_{Ti}: Total crew

M_{CNi}: Yearly consumable mass at a site (kg)

M_{Ri}: Yearly mass for crew rotation at a site (kg)

E_{Pi} Site production power (kW)

E_{Ci} Site crew support power (kW)

E_{Ti} Total power (kW)

M_{Pi} Mass of production equipment (kg)

M_{Si} Mass of support equipment (kg)

M_{Ti} Total site mass (kg)

ΔV_{Ni} ΔV to the next site (m/sec)

ΔV_{Hi} High-thrust ΔV from LEO

ΔV_{Li} Low-thrust ΔV from LEO

r_{Ni} Mass fraction to the next site

r_{Hi} High-thrust mass fraction to LEO

r_{Li} Low-thrust mass fraction to LEO

M_{HIi} Yearly payload mass on high-thrust transports to/from LEO

M_{LOi} Yearly payload mass on low-thrust transports from LEO

M_{LIi} Yearly payload mass on low-thrust transports to LEO

M_{Ni} Yearly payload mass to the next site

M_{Bi} Yearly payload mass back from the next site

M_{FHi} Propellant mass for high-thrust transport - one trip to/from LEO

M_{FLOi} Propellant mass for low-thrust transport from LEO

M_{FLIi} Propellant mass for low-thrust transport to LEO

M_{FNi} Propellant mass for transport to next site

M_{FBi} Propellant mass for transport back from next site

T_{tai} Trip time from LEO with low thrust

T_{tbi} Trip time to LEO with low thrust

T_{tci}	Trip time to next site
T_{tdi}	Trip time back from next site

This set of 40 process variables translates into 160 parameters. However, since many of these values can be found from calculations with the other variables, the first five parameters listed are the primary independent variables. The nominal values of these parameters are listed in Table 1-2.

The global variables required are:

N_u	Number of units produced per year
M_u	Mass of an individual unit
Y_u	Yearly return on a unit (\$/yr)
L_u	Lump sum return on a unit (\$)
T_S	Setup time for the production system (yrs)
T_P	Production lifetime for the system (yrs)
T_R	Runout time for units with yearly return (yrs)
r_i	Discount rate on investment capital
M_{CR}	Mass of an average crewperson (kg)
M_{HU}	Housing unit mass per crewman
S	Multiplicative factor for support personnel
W_E	Yearly wage on Earth (\$/person)
W_S	Yearly wage in space (\$/person)
C_c	Consumables usage (kg/person-day)
C_{TR}	Crew training cost (\$/person)

p	Exponent of learning curve
C_0	R&D cost factor ($/kg)
C_1	Production cost factor($/kg-unit)
M_{PLH}	Payload mass of high-thrust vehicle
M_{PLL}	Payload mass of low-thrust vehicle
I_{SP_H}	Specific impulse of high-thrust vehicle
I_{SP_L}	Specific impulse of low-thrust vehicle
f_{E_H}	Fraction of high-thrust propellant of earth origin
f_{E_L}	Fraction of low-thrust propellant of earth origin
ϵ_H	Propellant tank mass fraction for high-thrust vehicle
ϵ_L	Propellant tank mass fraction for low-thrust vehicle
η	Low-thrust engine efficiency
κ	Low-thrust engine length scaling factor
ϕ	Low-thrust engine mass flow scaling factor
m	Low-thrust propellant mass flow rate (kg/sec)
α	Specific powerplant mass (kg/kW)
ψ	Specific powerplant cost ($/kW)
L	Cost from ES to LEO ($/kg)

This set of 34 parameters completes the system specification: in all, 81 variables are required for each scenario. It should be pointed out that not all the parameters are used in any particular estimate: for example, the site variables specify 9 locations, but no more than 5 sites are used in any single scenario.

4.2 COST ESTIMATION RELATIONS

With the specification of the independent parameters, the values of interest (such as the 35 dependent process parameters) can be derived from the definitions. For example, the input mass at each site i from lunar origin is

$$(4.3) \quad I_{L_i} = O_i \, \frac{1 - f_{E_i}}{f_{T_i}}$$

where the output is defined as

$$(4.4) \quad O_i = I_{L_{i+1}}$$

for the first three sites, and as

$$(4.5) \quad O_4 = M_u \, N_u$$

at the assembly site. The size of the production crew at each site is

$$(4.6) \quad C_{P_i} = \frac{I_{L_i} + I_{E_i}}{P_{c_i} \, T_{w_i}}$$

This is the number of people actively producing the end product of the system. To this number must be added the people at the site associated with support function such as maintenance, logistics, and sanitation, in order to arrive at the total crew complement at each site:

(4.7) $$C_{Ti} = S\, C_{Pi}$$

In order to survive in space, the crew must be provided with consumables from Earth. The total yearly mass of these consumables is

(4.8) $$M_{CNi} = 365\, C_c\, C_{Ti}$$

The factor of 365 is, of course, required to convert from the kg/day units of C_c to a yearly mass.

Due to the cumulative effects of radiation and weightlessness, the crew must be periodically rotated back to earth. The total payload mass associated with this rotation is

(4.9) $$M_{HTi} = M_{CRi}\, R_{cj}\, C_{Ti}$$

As throughout this section, a sub-subscript of i refers to process i, and a sub-subscript j refers to a particular location j. As (4.9) shows, the only payload carried by the high-thrust transports is personnel and their effects. All of the remaining payloads are transported on the low-thrust transports, which are generally more economical in fuel usage.

However, this assumption should be checked by costing out scenarios where all goods travel by high-thrust transport, particularly those systems (such as NERVA/LOX) which operate entirely with lunar propellants.

The payload carried on the cargo vehicles outbound from Earth consists of earth inputs for the production process and consumables for the crew:

(4.10) $M_{LOi} = I_{Ei} + M_{CNi}$

The production power requirement sizes the power generating equipment which is dedicated to the production process:

(4.11) $E_{Pi} = O_i \, E_{Mi} \, T_{Wi}$

The power plant must be augmented to supply the power needed for crew life support:

(4.12) $E_{Ci} = E_S \, C_{Ti}$

Equations (4.11) and (4.12) can be summed to find the total required power:

(4.13) $E_{Ti} = E_{Pi} + E_{Ci}$

In a manner similar to the power requirement, the mass of the production site can be dissociated into production mass

$$(4.14) \quad M_{Pi} = \frac{O_i}{P_{Mi} \; T_{Wi}}$$

and support mass (composed primarily of crew housing)

$$(4.15) \quad M_{Si} = M_{HU} \; C_{Ti}$$

in order to find the total mass of the production facility (excluding the mass of the power plant):

$$(4.16) \quad M_{Ti} = M_{Pi} + M_{Si}$$

The initial cost of this facility is composed of the R&D and procurement costs and the transport costs, which are all estimated on a mass basis, and the costs associated with the powerplant:

$$(4.17) \quad C_{Ii} = (C_0 + C_1 + C_L) \, M_{Ti} + \Psi \, E_{Ti}$$

In a similar manner, the yearly operating cost for a production site is the cost of transporting people, propellants, and goods; crew wages and training costs are the other contributing factors. The number of new people who require training are a function of the length of time a person can hold a job at a particular site without exceeding cumulative limits for radiation. The yearly training cost for a process location is therefore

$$(4.18) \qquad C_{TRi} = \frac{C_{Ti}}{L_{Tj}} C_{TR}$$

The total yearly mass launched from Earth bound for site i is

$$(4.19) \qquad M_{LOi} = O_i F_{Ei} + M_{CNi}$$

One other cost element is the amortization of the initial cost of the low-thrust vehicles. Since each flight takes a substantial amount of time, the payload should have to bear the interest on the capital investment while it is using the vehicle. This cost is

$$(4.20) \qquad C_{LO} = \left(\frac{M_{LOi}}{M_{PLL}} T_{ta} + \frac{M_{LTi}}{M_{PLL}} T_{tb} + \frac{M_{Ni}}{M_{PLL}} T_{tc} + \frac{M_{Bi}}{M_{PLL}} T_{td} \right) \overbrace{\left(KC^2 + \phi \dot{m} \right)}^{C_I r_i}$$

The total cost for a year of operations at site i is therefore

$$(4.21) \qquad C_{OPi} = \left(\frac{C_{TR}}{L_{Tj}} + W_s \right) C_{Ti} + C_{LO} + C_L \left(M_{HIi} + M_{LOi} \right)$$

4.3 PROPELLANT TRANSPORT ADJUSTMENTS

With the relations specified in the previous section, all of the parameters which directly relate to site or process costs have been identified. However, the effect of propellant transportation has yet to be addressed. The propellant needed for one path in the system may not be available at the outset of that path; at some previous time, it will have had to be brought from the source locations. For earth-derived propellants, that source is the earth's surface via low earth orbit. For lunar propellants, the materials source is the scrap from the refining procedure in the second process step. The remaining question is how to distribute the propellants throughout the system. In [31], this was done by a complicated formula, sizing the propellant tanks by the largest propellant mass expected if sufficient propellants were carried for both legs of a round trip, modified by the possibility of partial or total refueling at one or both ends. This technique was difficult to understand intuitively, and could be implemented only with "transport map overlays", which had to be derived by hand. It was desired that this costing implementation should have a straightforward method for correcting for propellant movement throughout the transport system, which could be easily implemented in computer code.

From (3.22), it is possible to find that

$$(4.22)\quad M_F = M_i + M_G \frac{\epsilon(1-r)}{1-\epsilon} + M_{PL}$$

Using the definition of the mass ratio to substitute in for M_G, and then separating terms, the mass balance of the vehicle can be found using the final mass M_F:

$$(4.23)\quad M_F \left[\frac{r(1-\epsilon) - \epsilon(1-r)}{r(1-\epsilon)} \right] = M_i + M_{PL}$$

or, solving explicitly for M_F,

$$(4.24)\quad M_F = \frac{(M_i + M_{PL})\, r(1-\epsilon)}{r(1-\epsilon) - \epsilon(1-r)}$$

Again using the definition of the mass ratio r, and substituting in for M_F, the desired mass M_{PR} can be found to be

$$(4.25)\quad M_{PR} = \frac{(M_i + M_{PL})(1-r)(1-\epsilon)}{r(1-\epsilon) - \epsilon(1-r)}$$

Thus, the propellant mass for a vehicle can be found by a simple function of the inert mass M_i (propulsion system mass), propellant mass M_{PL}, and dimensionless parameters ε(tank mass fraction) and r (mass ratio). This applies to a vehicle making a single trip, with fully loaded tanks at the beginning of the mission. If a round trip is to be performed, the vehicle must be refueled at the far end for the return flight. Therefore, all flights which leave a node of the transportation system must be refueled at that node. That fuel must be carried to the node from the two source nodes (LEO and the refining site), along the most direct path.

This system has several advantages over the previous map overlay system. The high-thrust systems, which are often chemical propulsion systems and therefore high in fuel usage, are overly penalized if they must also carry fuel for the return mission. In this system, fuel for a future transport leg may be carried via the most economical system to a future refuelling site, where it can be used to refuel both cargo and personnel vehicles.

Referring to Figure 25 on page 130, the adjustments in payload to account for the transport of propellants around the system can be specified on a path-by-path basis:

1a	Not used (no direct path from LEO to LS)
1b	Not used (no direct path from LEO to LS)
1c	No change [0]
1d	Add lunar propellant for 1c; earth propellant for 1c [1]
2a	Add earth propellant for 1c,1d,2b,2c [5]
2b	Add lunar propellant for 1a,2a,3a,4a,5a [5]
2c	Add lunar propellant for 2d,3b,3c,3d,4b,4c,4d,5b [4]
2d	No change [0]

3a Add earth propellant for 2d,3b,3c ^{3}

3b No change ^{0}

3c Add lunar propellant for 3d,4b,4c,4d,5b ^{2}

3d No change ^{0}

4a Add earth propellant for 3d,4b,4c ^{2}

4b No change ^{0}

4c Add lunar propellant for 4d,5b ^{1}

4d No change ^{0}

5a Add earth propellant for 4d,5b ^{1}

5b No change ^{0}

Note: Implementation of the propellant mass adjustments will cause changes in payloads, and therefore changes in the proper adjustments, elsewhere in the system. The proper adjustment technique is to calculate the payloads for all the unchanged systems (superscript ^{0}), then the first order paths (superscript ^{1}). Use the revised first order paths payloads to find propellant requirements for second order paths(^{2}), and so on. Only the two fifth order paths are recursive: that is, propellant for 2a is carried as payload on 2b, and vice versa. As this is a linear system with two unknowns, the proper payload adjustments can be easily calculated, without the need for iterative solutions which may not converge.

With the technique for line item costing derived, the algorithm may be used for

- Finding system costs based on estimated parameters
- Finding cost sensitivity from a variation of parameters analysis
- Finding attractive program scenarios for further study
- Deriving input values for the optimization technique of the following chapter

5.0 OPERATIONS OPTIMIZATION

"Look, Dave, I can see you're really upset about this...I know I've made some rather poor decisions lately..."

- HAL 9000

5.1 INTRODUCTION

In the preceding chapters, techniques have been shown to provide optimum choices in the design of individual systems, such as launch systems, and to select the best mix of systems for an overall program scenario. However, the most interesting question to be answered involves the selection of program options from among a set of possible scenarios. For example, assume that three energy production options appear most promising: coal-powered conventional plants, solar power satellites assembled in low earth orbit from components manufactured on earth, and solar power satellites manufactured in high earth orbit from lunar materials. Each of these systems has its own recurring and nonrecurring costs. In addition, there are constraints on the overall system, such as yearly budget allocations or maximum demand for new power generating capacity.

In this situation, it is quite straightforward to calculate the total cost for each system, assuming only one type of power generation is used, and that the demand is satisfied. However, the optimum solution for competitive systems will (in general) not be found by this method. Instead, the optimum will generally consist of a mix of plant types, with parallel and/or sequential use of competing options. The desired result from such an optimization algorithm would be the year-by-year allocation of investment funds (constrained below set levels) for each of the candidate systems, resulting in the optimum distribution of investments in order to maximize a desired objective function (generally net discounted profits). To perform such an optimization, operations research techniques must be used.

5.2 OPTIMIZATION ALGORITHM AND IMPLEMENTATION

The problem at hand is the distribution of a limited set of resources among a group of candidate options, with the intention of maximizing the return of the system as a whole. This problem is often addressed through the use of linear programming. If the problem can be specified within the limits of a set of linear equalities and inequalities, use of the simplex method provides a computationally convenient optimal solution.[52],[53]

Assume that the problem to be solved is the optimal investment strategy for funding a variety of different systems. Each system can be categorized by an initial nonrecurring cost, a recurring cost per unit, and a yearly benefit from each unit. Clearly, this system will describe the sample problem discussed in the introduction to this chapter. Limitations on this formulation include the lack of any direct representation of unit cost reduction due to learning effects, which is a power law relationship. However, as will be shown, the model can be adjusted to include the effects of cost discounting, which also is a power law effect, but which is not applied to a decision variable, as is the case with the learning curve.

By examining the set of "most favorable" scenarios, it is possible to limit the optimization to a set of m candidates. Each system i will have a nonrecurring cost of A_i ($B), a production cost of C_i ($B/unit), and a benefit of B_i ($B/unit-year). All of these parameters are assumed to be independent of the year, j. The objective of the optimization is to invest money yearly over n years as determined by two decision variables for each system: the fraction of nonrecurring cost A_i paid in year j, F_{ij} (1/year), and units of system i built in year j, U_{ij} (units/year). U_{ij} is not constrained to be integer: fractional units may be built in a single year, just as in the real world a single unit may take longer than one year to produce.

Although the yearly benefit from each unit produced is expressed as B_i, it would be desirable to find the total return from a system related to the production date, rather than to have to sum up yearly returns over the lifetime of the production unit (typically 30 years). Since the benefit is invariant over the course of the program for each type of unit, the total net return for each unit is the sum of the net value of the yearly returns over the lifetime of the unit, minus the initial production cost. This factor, R_{ij}, represents the net value in year j of a unit of type i produced in that year. The equation relating the input parameters B_i and C_i to R_{ij} is

$$(5.1) \qquad R_{ij} = B_i \frac{1-(1+r)^{-n}}{r} - C_i$$

where the fraction represents the total discounted value of a constant payment made over a period of time.

External constraints to the system arise from the limitations on investment capital (L_j, the limit of new funds available in year j), and on demand for new units (U_{ij}). The constraint equations can be expressed as:

- Payment of nonrecurring costs -

$$(5.2) \quad \sum_j F_{ij} = 1 \qquad \text{For all } i$$

- Spending limited to new funds and returns -

$$(5.3) \quad \sum_i U_{ij} C_i \leqslant L_j + \sum_i R_{i(j-1)} \qquad \text{For all } j$$

- Total demand constrained -

$$(5.4) \quad \sum_i \sum_j U_{ij} \leqslant U_{lim}$$

Equation (5.2) is an equality constraint, which assumes that the optimal solution includes the use of system i. If this is not the case, (5.2) equals zero rather than one: this must be accounted for in the simplex representation. Since this constraint holds for each system, m constraint equations arise from this condition. Inequality (5.3) says that investment credits may come only from the new money allocated within the spending limit for year j (L_j), and from returns from the previous year. This assumes that returns not immediately reinvested become profits, and are therefore no longer accessible. It should also be noted that (5.3) assumes interaccessibility of assets: that is, profits from one system may be used as investment capital in a different system. This would be the case for a single coordinating body (such as the government) which is choosing between a set of program alternatives. If it is assumed that each system is self-contained in terms of investments and returns, (5.3) no longer applies. However, it is reasonable to assume in such a case that the option of choosing an optimal phased development between systems is not possible (i.e., there is no way to develop independent capital to initiate a second system while operating the first, if profits may not be moved between systems). While independence of competing programs simplifies the solution by eliminating phased transitions between production methods, it is more interesting to investigate the general case which analyzes the interplay between systems. Equation (5.3) applies to each year, creating n constraint equations. Since (5.4) defines global demand, one constraint equation is generated, for a total of m+n+1 constraints for this formulation.

The objective function for this system is:

$$(5.5) \quad \mathrm{Max} \sum_j D_j \left[\sum_i \left(R_{ij} - C_i U_{ij} - A_i F_{ij} \right) \right]$$

Maximizing this function results in maximizing net present value profits. The interior of this function merely represents the returns minus the production investment and the research and development costs summed over all of the candidate systems at a given year j. This is multiplied by a factor which converts the current year costs into the net present value of the costs for year j. This factor, D_j, is found by

$$(5.6) \qquad D_j = (1+r)^{-j}$$

where r is the yearly interest rate assumed for the cost of investment capital.

At this point, enough data exists to run an LP optimization of the system, based on sample values of the variables. However, the answer would not be meaningful, due to a problem with the payment of nonrecurring costs. Since the nonrecurring cost has no positive effect on the objective function, the optimal solution always chooses (when possible) to pay no nonrecurring costs at all. By expressing (5.2) as an equality constraint, this problem may be dealt with; however, two associated problems then arise. The nonrecurring cost is an "existence" cost: it has a zero value if a candidate system is not used, and its full value if the system is used. This noncontinuous behavior cannot be modelled in a straight-forward manner in a linear program. The second problem is that recurring and nonrecurring costs are time-critical: it is not possible to build units of a system and receive a yearly return, then do the initial research and development. However, this is exactly what the optimum tries to do. Since the cost discounting procedure favors getting returns early and putting off expenses until later, sample optimization runs resulted in paying the nonrecurring costs in the final year of the program for each system. In an attempt to force the payment of nonrecurring costs before system production begins, another set of constraints were added:

$$(5.7) \qquad U_{ij} \leq \sum_{k=1}^{j-1} (F_{ik} + U_{ik}) \qquad \text{for all } i$$

This constraint limits the production rate to no more than the fraction of R&D paid: for example, one full unit could be produced only after the year in which the nonrecurring costs were paid in full. In order to allow for production expansion, constraint (5.7) provides that the yearly constraint on production is increased by the total number of units produced to date. The maximum production expansion allowed thereby is geometric: for example, after paying the nonrecurring cost in full, one unit could be built in the following year, two units in the next, three units in the next, six units in the next, and so on. It was hoped that this would allow adequate expansion capability for the candidate systems. If this proved to be a binding constraint, the effect of changing it would have been investigated. Since this condition applies to each system in each year, m*n new constraints are generated, greatly increasing the size of the tableau to be solved. This increases the total number of constraints to m*n+m+n+1.

Implementation of the system showed the drawback: it is not possible to require that all R&D be paid before production begins. When (5.2) was an inequality constraint (≤ 1), typically 10% of the nonrecurring cost would be paid in the first year, followed by 0.1 units built the next year, 0.2 units the next, and so on. Changing (5.2) to an equality constraint resulted in the same solution, except that the final 90% of the R&D cost was paid in the final program year (when discounting effects are minimized). With (5.2) as an inequality constraint, the nonrecurring costs of a single system were varied, inhopes of finding a case where the fraction of nonrecurring costs paid before production times the specified R&D costs would equal the original R&D cost. The test case was terrestrial-origin SPS, with an actual (assumed) nonrecurring cost of $50 billion. It was found that as the specified R&D cost increased, the initial fraction paid decreased; even if the final specified R&D cost was $500,000 billion, the actual amount of R&D paid in the optimal solution was substantially less than $50 billion, and the SPS production remained bounded at the upper growth rate. This exercise indicates the difficulty of forcing a strict linear programming solution of the time-critical R&D cost problem: there is no direct monetary return from paying nonrecurring costs, especially early in the time frame where cost discounting applies the largest surcharges. It is only in the real world that R&D must precede production.

At this point, it became obvious that simple linear programming would not be sufficient to adequately constrain the problem. Two possible solutions were investigated. The first consisted of formulating an integer programming problem, forcing the fraction of R&D paid to be either 0 or 1. This was not chosen, since it was felt to be unrealistic: research and development is never paid in a single year, and one of the objects of the study was to investigate the optimal distribution of returns from ongoing programs to pay for R&D of more sophisticated systems. The second possibility was to perform an integer programming solution holding the number of units produced in each year to integer values. While this results in a realistic problem statement, it results in an extremely complex IP formulation, since the number of units produced in each year of each type must be held constant: this could mean as many as 150 integer decision variables. This approach was rejected due to numerical complexity.

The approach taken to solve this problem involved separating it into two parts: solve the LP solution for the optimum distribution of resources without optimizing the choice of initial production time, then optimize the production startup time.

The first task was completed by introducing a variable y_i, which represents the year of initial operational capability for system i. The variable F_{ij} exists only for values of $1<j<y_i-1$, and the variable U_{ij} is valid only for values of $j>y_i$. In effect, where there were before two sets of decision variables for each system of interest (F_{ij} and U_{ij}, $1<j<m$), there is now only one (F_{ij}, $j<y_i$; and U_{ij}, $j \geqslant y_i$). Whereas before payments on the R&D could take place concurrently with unit production, the new formulation sets an initial production year. All R&D must be paid before that year (F_{ij}); all production must occur during or after it (U_{ij}). This decrease in the size of the tableau shortens solution time. The constraint equations now become:

- Payment of nonrecurring costs prior to initial production of system i in year y_i (m equations)-

$$(5.8) \qquad \sum_{j=1}^{y_i-1} F_{ij} = 1 \quad \text{for all } i$$

- Spending limited to new funds and surplus of the previous year (n equations) -

$$(5.9) \qquad \sum_{i} u_{ij} C_i \leq L_j + \sum_{i} R_{i(j-1)}$$

- Bounded maximum demand -

(5.10) $$\sum_{i} \sum_{j=Y_i}^{m} U_{ij} \leq U_{lim}$$

The objective function is not changed from (5.5), except for the caveat that the variables F_{ij} and U_{ij} do not exist for certain values of j. The total set of constraints consists of m+n+1 equations.

This formulation permits the solution of the LP problem, resulting in the optimum payment history for nonrecurring and recurring costs over a set of competing systems for a multi-year program, given the external specification of the earliest initial production year for each system. It should be noted that (5.8) is an equality constraint: the R&D costs on a system are paid, whether that system is used or not. A separate procedure, detailed later in this chapter, tests the solutions where single or multiple systems are entirely unused, and no R&D costs are charged for these systems.

Due to the elimination of equation (5.7), the tableau matrix to be solved is considerably reduced in size. For a typical program with 5 candidate systems over 30 years, the number of constraints is reduced from 186 in the original case to 36 for the second formulation. Similarly, the number of decision variables has been reduced from 300 to 150 (5 systems over 30 years, where each variable represents investment in one system in a single year, applied to either R&D or procurement).

Given the current formulation of the constraints and objective function, it is possible to use simplex methods to arrive at an optimal distribution of resources between competing programs which will maximize the profits returned from the units produced. However, it should be emphasized that this is a sub-optimal solution, in that the initial operational year for each program is specified externally to the linear programming problem. Thus, no information results from the LP solution which yields any information on the correct choices of initial operational years y_i which will arrive at a truly optimal solution for the overall scenario.

The solution adopted combines two optimization techniques: using the standard linear programming techniques to find local optima given values for y_i, and a steepest ascent algorithm to find the optimum choices for y_i. Due to the integer nature of y_i, it is not possible to perform a classical steepest ascent optimization. Instead, at each step a linear program optimization is performed for the current best estimate for y_i for each of the m systems. This set of m values of y_i represents a "state vector" of the LP solution, Y. After this, an LP optimization is done for each of the m systems, with y_i for that system varied by 1. The difference in values of the objective function between the base case and the variational case for system i demonstrates the local value of ∂(objective function)/∂y, which indicates the optimal direction for the next trial value. The set of m values by which the state vector Y will be modified for the next step of the optimization is called the ascent vector, V. Since y_i is constrained to be an integer, the values of the elements of V are constrained to be 1,0, or -1. Since a value of 0 can occur only if the value of the objective function is unchanged with a change in the decision variable, it is generally true that all elements of V are either 1 or -1, and the change in Y from one step to the next is along a diagonal in Y-space. This technique, developed for this study, will be referred to as "diagonal ascent linear programming optimization", or DALP optimization.

The iteration algorithm is as follows: at Y, an LP solution is found. Each value of Y (y_i) is varied in turn by the corresponding value of V (v_i). For example, The next value of v_2 (the second variable in array V) is found by

$$(5.11) \qquad V_{2_{k+1}} = V_{2_k} * \text{sign}\left[opt\,(Y + V_{2_k}) - opt\,(Y) \right]$$

If v_2 is unchanged from the previous step, y_2 is replaced by y_2+v_2, and the next step proceeds in the same manner. If, however, v_2 changes during the current step, y_2 is held at its current value for the following step. For example, the iteration may reach the value of y_2 which produces the maximum value of the objective function. Since the other variables have not yet reached their optimum values, the algorithm holds the value of y_2 at its determined value, yet searches the neighboring points in system 2 to insure that y_2 is still the optimum. If the interdependence of systems results in a change of y_2(optimum) while varying the other values of y_i, the iteration scheme for y_2 again starts scanning neighboring values of v_2 during the iteration cycle to check for new optima. Convergence of the solution occurs when all neighboring points result in objective function values less than those at the base point for this step. This is the case when all elements of Y are unchanged between successive steps.

This diagonal ascent iteration is started by taking assumed values for the elements of Y and V. Since (in general) it is equally likely for the optimal value of y_i for a system to be found at either the beginning or the end of the overall program, a value in the middle is chosen to minimize the distance to the optimum. For a system covering n program years, n/2 iteration steps are a conservative estimate for the number of steps to reach the optimum, assuming the initial values of y_i are set to n/2. Lacking initial information on the variational effects of V, the values of v_i are set initially to arbitrary values, generally 1. Since each iterative step requires (m+1) LP solutions, (n/2)(m+1) LP solutions are required to find the optimum solution.

Thus, the two-step procedure of diagonal ascent and linear programming results in the optimum choice of initial years of production for each of the candidate systems, as well as the optimal distribution of limited resources in research and procurement over the possible systems. It should be noted that the analytical proof of optimality has not been done: indeed, since this formulation is strongly dependent on the trends of the problem, any such proof may have to be repeated for each class of application. The use of the word "optimum" should be taken to read "the heuristic approximation of optimum", in the absence of a rigorous proof. It should be noted, however, that in exhaustive searches of lower order systems (i.e., trying every possible solution for examples of single and two-set problems), the predicted answer did indeed turn out to be the global optimum.

Constraint (5.8) in the LP optimization insured that full research and development costs were paid on all of the candidate systems, regardless of whether or not production units of that system were included in the optimum basis. Therefore, it is necessary to include a third step, which searches systematically through the possible combinations of systems to be used, and arrives at the optimum choice of systems, along with the optimum initial operational years and the optimum distribution of resources.

For example, assume that the program under study includes three candidate systems. The goal of the optimization is to identify which of the systems should be used, in which year each system should reach production capability, and what the optimum distribution of money is so as to maximize net discounted returns. The DALP optimization procedure will give the optimum result, assuming that all three systems are funded at least through the R&D phase. Likewise, three more DALP runs will identify the optima, assuming that only two systems are used (1&2, 2&3, 3&4). Three more DALP runs will identify the optima for single systems (1, 2, and 3). In all, there will now be 7 possible optimum solutions. That system which maximizes the objective function from among the seven possible solutions will finally be the overall optimum.

In general, the combinations of possible systems to be tested goes according to the binomial theorem:

$$\sum_{i=1}^{m} \binom{m}{i} = \sum_{i=1}^{m} \frac{m!}{(m-i)!\, i!} = 2^{m} \tag{5.12}$$

There is in addition the degenerate case of no systems selected; this is the global optimum only when all of the other possible solutions result in negative values of the objective function.

Figure 26 on page 164 summarizes the overall optimization algorithm used for this analysis technique. The entry point to the algorithm depends on the order of the system to be optimized. After the initial analysis is performed assuming that all of the candidate systems are funded, all combinations of systems are tested in order to find the choice of systems to be funded which will maximize the objective function. The number of DALP optimizations which must be performed to produce this result is the same as the number of combinations from equation (5.12). The number of LP solutions necessary, which is a measure of the numerical complexity of the solution, must include the fact that, for each DALP optimization of order i, (i+1)(n/2) LP solutions must, in general, be found. The total number of LP solutions in the overall system of order n is therefore

$$\text{number of solutions} = (m+1)\, n\, 2^{m-1} \tag{5.13}$$

This equation does not include the fact that, as the number of candidate systems decreases, the size of the tableau decreases. Due to the formulation of the LP problem discussed earlier, the tableau for i systems over n years is $(i \cdot n) x (i+n+1)$. The size of the tableau does not, of course,

```
 ---------------------------------------------
|Initialize values of                         |
| m:     number of candidate systems          |
| A(i): R&D cost for system i                 |
| B(i): yearly benefit from system i          |
| C(i): unit cost of system i                 |
| L(j): spending limit for year j             |
| Umax: limit of demand for new units         |
| (example assumes a 30-year program)         |
 ---------------------------------------------
                      |
                      V
 ---------------------------------------------------
| Enter program based on number of systems:         |
|  m=    1      2      3      4      5              |
 ---------------------------------------------------
         |      |      |      |      |
         |      |      |      |      V
         |      |      |      |  --------------------------------       -------------------------
         |      |      |      | | Find optimum y(i) assuming     |     | Find individual LP      |
         |      |      |      | | all 5 systems used             |-->  | solutions given y(i)    |
         |      |      |      | |  90 LP solutions               |<--  |  150x36 tableau         |
         |      |      |      |  --------------------------------       -------------------------
         |      |      |      |<-----'
         |      |      |      V
         |      |      |  ----------------------------      ------------------------------      -------------------------
         |      |      | | Test all combinations of   |    | Find optimum y(i) assuming   |    | Find individual LP      |
         |      |      | | 4 systems out of m used    |--> | 4 systems used               |--->| solutions given y(i)    |
         |      |      | |  m!/(m-4)!4! optimums      |<-- | 75 LP solutions              |<---|  120x35 tableau         |
         |      |      |  ----------------------------      ------------------------------      -------------------------
         |      |      |<------'
         |      |      V
         |      |  ----------------------------      ------------------------------      -------------------------
         |      | | Test all combinations of   |    | Find optimum y(i) assuming   |    | Find individual LP      |
         |      | | 3 systems out of m used    |--> | set of 3 systems used        |--> | solutions given y(i)    |
         |      | |  m!/(m-3)!3! optimums      |<-- |  60 LP solutions             |<-- |  90x34 tableau          |
         |      |  ----------------------------      ------------------------------      -------------------------
         |      |<-----'
         |      V
         |  ----------------------------      ------------------------------      -------------------------
         | | Test all combinations of   |    | Find optimum y(i) assuming   |    | Find individual LP      |
         | | 2 systems out of m used    |--> | set of 2 systems used        |--> | solutions given y(i)    |
         | |  m!/(m-2)!2! optimums      |<-- |  45 LP solutions             |<-- |  60x33 tableau          |
         |  ----------------------------      ------------------------------      -------------------------
         |<-----'
         V
  ------------------------         ------------------------------      -------------------------
 | Test all individual    |       | Find optimum y(i) assuming   |    | Find individual LP      |
 | systems singly         |------>| a single system used         |--> | solution given y(i)     |
 |  m optimums            |<------|  30 LP solutions             |<-- |  30x32 tableau          |
  ------------------------         ------------------------------      -------------------------
         |
         V
 -----------------------------------------------
| Given optimum set of systems                  |
| and initial operational years y(i),           |
| find optimum distribution of                  |
| resources and production schedule             |
|  1 LP solution                                |
 -----------------------------------------------
```

Figure 26. Flow Chart of DALP Optimization

affect the number of LP solutions, but it does affect the speed of the individual LP analyses. Table 5-1 summarizes the number of LP solutions to be performed based on the number of systems originally under consideration, for a 30-year production program.

The convergence of the diagonal ascent portion of this optimization technique is dependent on two conditions: first, the "state space" of the LP solutions for various values of initial years must be well behaved, without discontinuities of slope between adjacent values of y_i. The second assumption implicit in this approach is that the system has a single global optimum for each set of candidate systems analyzed, without local minima or maxima. It might be expected that, since the values of intermediate steps in the diagonal ascent algorithm are independent LP solutions, the value of the objective functions from step to step could vary widely as systems are moved into or out of the basis. However, in practice this is not the case. The key to the monotonic behavior of the LP solutions is constraint (5.8), which requires that nonrecurring costs be paid on all candidate systems, regardless of whether or not that system is cost-effective. As results indicate, a system which is profitable is used to the fullest extent possible, bounded only by budget constraints, other systems, and initial operational years specified externally. If a competing system is more advantageous, it is used to the exclusion of the first

TABLE 5-1:
LP Solutions Required
for DALP Optimizations

Systems	LP Solutions
1	60
2	180
3	480
4	1200
5	2880

Table 5-1: LP Solutions Required for DALP Optimizations

---

system, except in those times in which the first system is the only feasible one of the pair (i.e., before the initial operational year of the second system). If the second system is not as cost-effective as the first, the first is used exclusively, and (due to the nature of the LP solution) to the greatest possible extent of the externally spaecified LP constraints. Since these constraints are not changed during the diagonal ascent portion of the optimization, the solution space is indeed monotonic and possesses a single global optimum.

5.3 EXAMPLE DALP OPTIMIZATION

In order to more fully demonstrate the methods and capabilities of the diagonal ascent linear programming optimization technique, a sample case representative of solar power satellites was used as an example. The parameters chosen are shown in Table 5-2. The three systems used range from low initial expense with high recurring costs (system 1), representing a fossil fuel generating plant on earth; through an intermediate case (system 2), similar to an SPS from earth materials; to a high nonrecurring cost which results in low unit production costs (lunar SPS, system 3).

Note: While these numbers are representative of solar power satellite production systems, they should not be taken as definitive values for any of the systems under consideration. In the current usage, they are primarily intended to illustrate the concepts of the DALP optimization technique.

Further constraints placed on the overall system were: new funding limited to $5B per year for each program year; and total demand for new production limited to 125 units.

Based on the yearly funding limits and the nonrecurring costs, the earliest possible starting year was estimated for each system, and the initial year chosen to begin the iteration was halfway between the minimum for that system and the end of the program in year 30. Table 5-3 details the solution procedure for the initial iteration, which assumes that all three systems are funded through the nonrecurring phase. It should be remembered that, for each of the estimated values shown in the

TABLE 5-2:
Parameters Used In
Sample DALP Solution

System:	1	2	3
R&D Cost:	5	25	50
Yearly Benefit:	3.5	3.5	3.5
Unit Production Cost:	30	20	15

Table 5-2: Parameters Used in Sample DALP Solution

---

preceding table, 4 LP solutions are found: this allows the estimation of the objective function at the present state, and the approximation of the partial derivatives of the objective function with a change in each of the initial production years for the three systems. The initial ascent vector, which is arbitrarily chosen in this case to be (1,1,1), results in lower objective function values for each of the three trial cases run after the first state estimation for years (15,17,20). For this reason, the state estimation was not revised for the second iteration, but the ascent vector was changed to (-1,-1,-1). All three trials resulted in increased values of the objective function, so the state vector was revised to (14,16,19), and the ascent vector kept as (-1,-1,-1). At step 4, it is found that further decreases in the initial year of system 3 results in reduced returns, so the state value for system 3 is unchanged between steps 4 and 5, while the ascent vector is changed from (-1,-1,-1) to (-1,-1,1). The optimization proceeds in the same manner throughout the iterative process,

TABLE 5-3:
Iteration Results for DALP
Solution with Three Systems

Systems:	Initial Years 1	2	3	Optimum Return
	15	17	20	-5.83
	15	17	20	-5.83
	14	16	19	-2.05
	13	15	18	0.44
	12	14	18	3.29
	11	13	19	16.20
	10	12	20	32.90
	9	11	21	50.76
	8	10	22	70.84
	7	9	22	92.82
	6	8	21	105.48
	5	8	21	108.49
	4	8	21	112.41
	3	8	21	117.74
	3	9	21	118.97
	3	9	20	119.25

Table 5-3: Iteration Results for DALP Solution with Three Systems

---

resulting in the eventual optimum solution of years (3,9,20) to begin production of units of systems 1,2, and 3, respectively. The total discounted net return on all three systems (which represents the value of the objective function) is 119.25 B.

Following the optimization with all three candidate systems used, three iterations are run for the three combinations of two units (1,2), (1,3), and (2,3). The iteration traces for these runs are shown graphically in Figure 27 on page 170.
For systems (1,2) and (1,3), the iteration proceeds along the diagonal until system 2 reaches its initial year constraint, which corresponds to the earliest year which system 2 can be brought on line with the funding limitations. The iteration

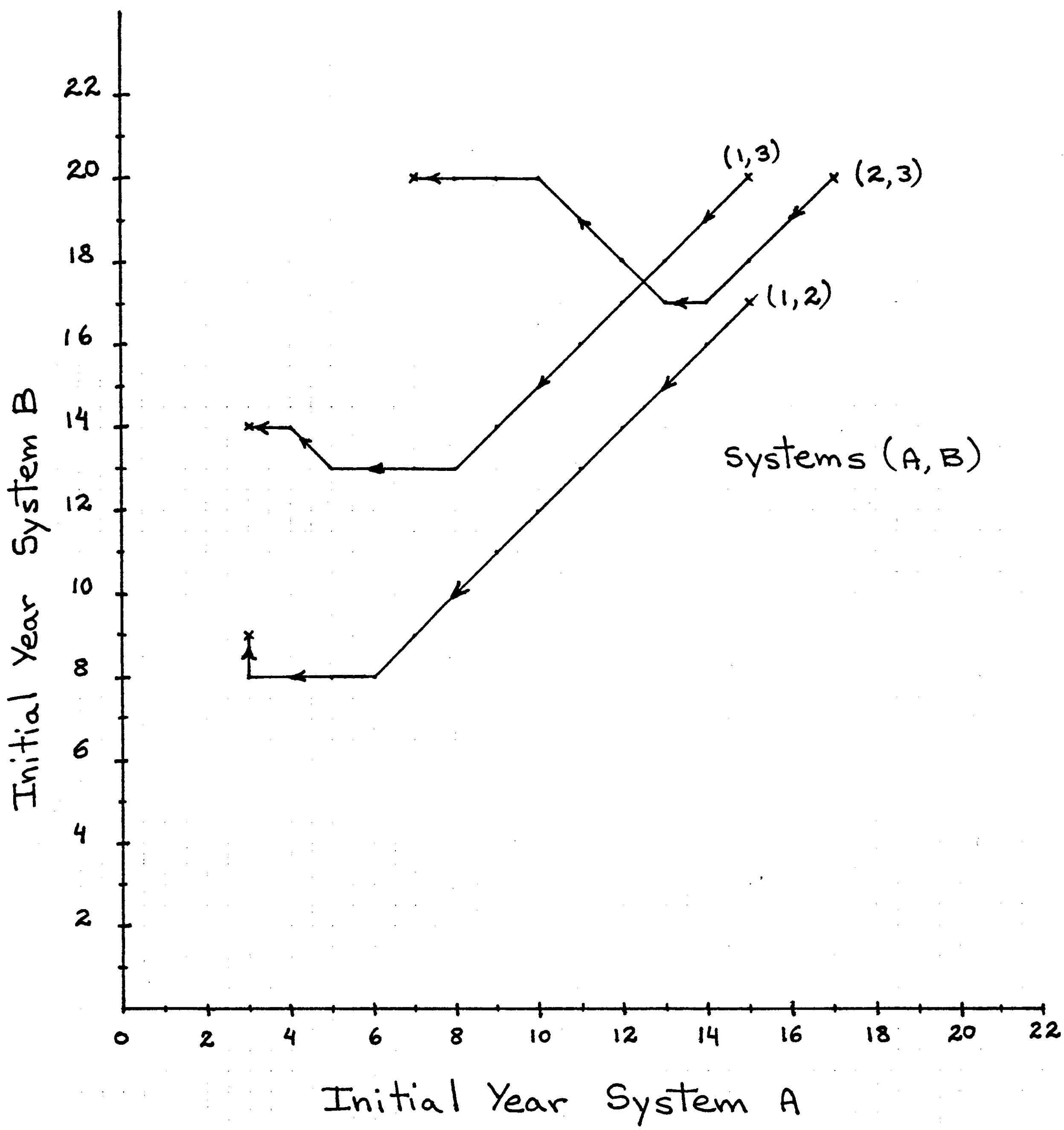

Figure 27. Two System Iteration Paths

then proceeds until system 1 hits its initial constraint. The optimums for systems (1,2) and (1,3) are therefore constrained at the earliest possible years for both candidate systems. On the other hand, the iteration for systems (2,3) reaches an optimum where system 2 is set as early as possible, but system 3 is at an unconstrained optimum value. The returns from these systems are: (1,2) - 76.02, (1,3) - 141.81, (2,3) - 118.92.

From the four possible solutions at this point, the best return is from using only systems 1 and 3, which results in the return of 141.8. Usng all three systems would result in an optimal return of 119.25, only marginally better than using systems 2 and 3. The worst optimal return is from using only systems 1 and 2. In order to insure that the overall best optimal solution is indeed from the use of systems 1 and 3, it is necessary to check for the best returns available from using the candidate systems individually. The results from these LP iterations are shown in figures Figure 28 on page 172, Figure 29 on page 173, and Figure 30 on page 174. As can be seen from these bar graphs, the best returns from using systems 1, 2, and 3 are 8.82, 77.38, and 94.92, respectively.

Since all the possible combinations of systems to be used have been investigated, it is evident that the most favorable return results from using only systems 1 and 3, which avoids paying the nonrecurring costs for the intermediate system, 2. At this point, one final LP solution is found, using the optimum values for the initial availability of systems 1 and 3. If system 1 (the low initial, high recurring cost system) is available for production in year 3, and system 3 (high nonrecurring cost, low

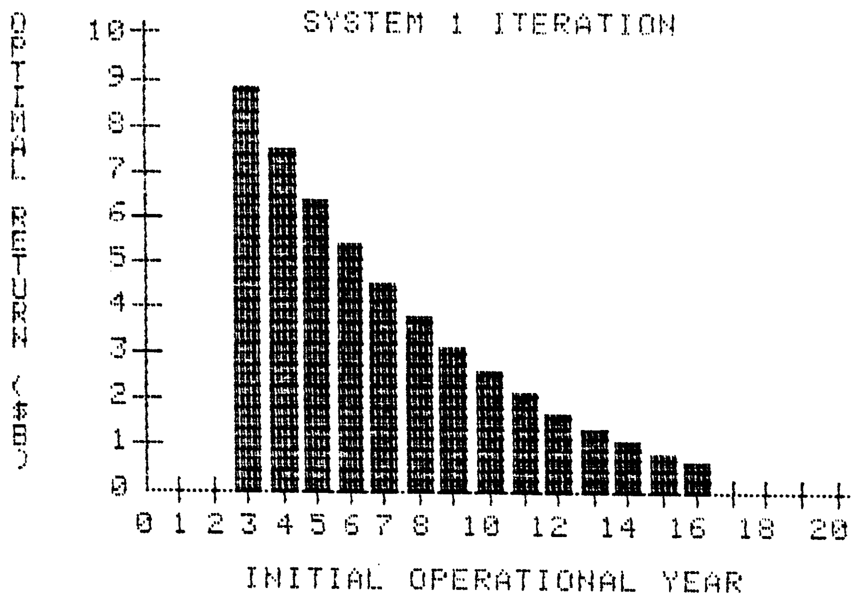

Figure 28. System 1 Iteration

cost per operational unit) starts producing units in year 14, the optimum distribution of resources is detailed in Table 5-4. Of particular interest is the fact that only 1.57 units of system 1 are produced, although this scenario proved to be significantly more cost-effective than using only system 3 alone. This is due to the impact of early monetary returns in the net present value of the total system, and illustrates that getting some returns early is more important than bringing the less expensive production system on line earlier. The small number of units produced by system 1 demonstrates that the funding

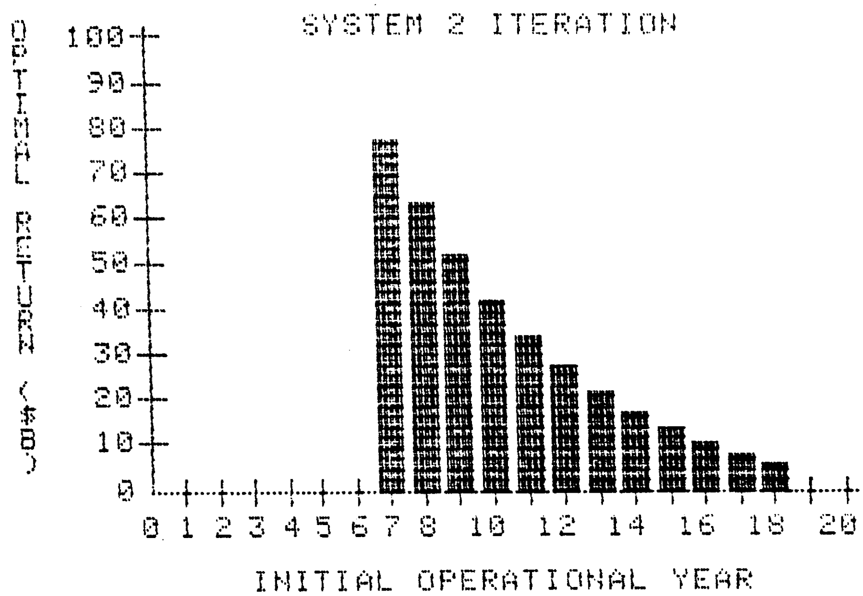

Figure 29. System 2 Iteration

limitation ($5B/year) is a significant constraint when each unit has a production cost of $30B. One possible inaccuracy is induced thereby, as returns from the .17 units produced in year 3 are used to increase production capacity to .19 units in year 4, and so on. Obviously, there are no returns from a partially completed unit. However, since each unit in this example was sized on a 10 MW solar power satellite, a production unit represents a large capability: it is equivalent to producing a larger number of smaller units in the same time period. Thus, the fact that .17 or .19 units are produced indicates that the

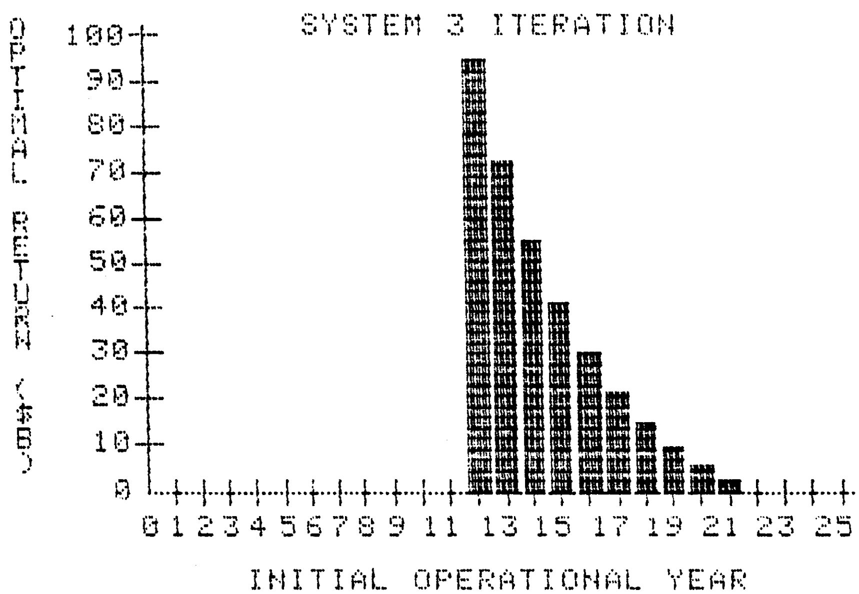

Figure 30. System 3 Iteration

---

demand for early returns on investment is great enough to warrant producing finished small units early in the program, and going to the larger units when system 3 comes on line in year 14.

5.4 ALGORITHM REFINEMENTS

TABLE 5-4:
Optimum Distribution of Resources
for Sample DALP Problem

Year	System 1 R&D	System 1 Units	System 3 R&D	System 3 Units
1			5.0	
2	5.0			
3		0.17		
4		0.19		
5		0.21		
6		0.23		
7		0.26		
8		0.29		
9		0.22	3.0	
10			10.5	
11			10.5	
12			10.5	
13			10.5	
14				0.70
15				0.86
16				1.06
17				1.31
18				1.61
19				1.99
20				2.45
21				3.03
22				3.73
23				4.60
24				5.68
25				7.00
26				8.63
27				10.65
28				13.13
29				16.19
30				19.97
Totals:	5.0	1.57	50.0	102.59

Table 5-4: Optimum Distribution of Resources for Sample DALP Problem

---

With the basic DALP optimization technique developed, it is possible to modify the algorithms to address some further refinements which would be desirable to include in the model. Since the optimization technique is built around the linear programming optimization of a representative scenario, it is important to emphasize that it is not possible at this point to redefine parameters which are internal to a single LP run, such as including learning curve reductions on the cost of production units within a system. However, external variables may be modified in the course of the optimization routine so as to increase the fidelity of the model.

For example, one feature which is of importance in competing production systems is the issue of commonality. For example, two competing systems may use the same transportation system: it is not accurate to charge the R&D cost of the transports twice. If both systems are used, the cost of transportation development should be shared between them. If, on the other hand, only one of the two systems is chosen, it must pay the full cost for developing the transport system. This can be included in the DALP algorithms in the following manner.

Each system i has a nonrecurring cost A_i, representing the total nonrecurring cost for that system. The issue is the amount of commonality between two systems, i and j. It is possible to introduce a commonality matrix M, which is of order (mxm), where m is the total number of systems under consideration. Each element of M, m_{ij}, represents the total R&D cost common to both systems i and j. Therefore, if i and j are both included in the list of systems to be examined at that step of the DALP iteration, the nonrecuring costs A and A are each reduced by $0.5m_{ij}$.

The problem with this system arises when more than two systems are used in the LP analysis. If three systems are present, for example, system 1 may have commonality with both systems 2 and 3; system 2 may or may not have some commonality with system 3, and commonality that does exist may or may not correspond to the same elements of commonality as between 1 and 2, and 1 and 3. This problem could be addressed with a three dimensional matrix, where m_{ijk} represents the cost of common elements in systems i, j, and k. However, for m systems, a matrix of order m^m is required in the general case: for 5 systems, 3125 commonality factors are required. It would be a major task just to accurately estimate this many parameters, much less include them in the optimization algorithm. For this reason, the approach taken herein is to use an (mxm) commonality matrix, and for each system i investigate the degree of commonality with all other competing systems. For m systems in the analysis, each will have m-1 commonality factors. Somewhat arbitrarily, the largest single commonality factor will be used in revising the nonrecurring cost estimate. Since three-way or greater commonality is not considered, the final estimates for nonrecurring cost for each system should be conservative. Similar commonality factors can be derived for recurring costs.

Although learning curve effects are both interior to the LP solution and nonlinear, it should be possible to include learning curve effects by the use of a piecewise linear approximation. Rather than specify the cost of a unit in any given year as a flat rate, the cost could be specified as being greater than or equal to a value proportional to the number of units built in the preceeding years. Several such functions for each year of the program will allow the use of linear approximations to the learning curve over several different parts of the program. However, if for example five linear regions are selected to approximate the learning curve, five new inequality constraints must be added for each of the decision variables. This greatly increases the size of the tableau matrices to be solved. While the learning curve could be modelled by this approach, it could be computationally tedious, and would probably be prohibitive in terms of computer time for use in all but the final decision runs.

5.5 OPTIMIZATION OF SPACE SHUTTLE CONFIGURATION

As part of its planning in the Apollo era, NASA attempted to draft a set of goals for hardware development which would befit an active, growing space program. Since it appeared at an early point in the planning process that manned planetary missions were not likely, NASA identified the development of a manned space station in low earth orbit as one of its top goals. A space shuttle was also planned, as a necessary vehicle for station resupply and crew rotation. When the budgetary "squeeze" started, NASA forsook the space station in favor of a fully reusable space shuttle, which was now justified on the basis of the economics of reducing the cost to low earth orbit for all payloads. However, as the budget kept shrinking, NASA was faced with yet another choice: how to get the most from the least amount of money, when designing a new launch vehicle.

The possible configurations as of early 1972 are presented in Table 5-5 [59]. The Office of Management and Budget had informed NASA that the peak funding for the shuttle could not exceed $1 billion per year, and NASA had a mission model of about 500 payloads for the vehicle. What would the best choice be?

	Configuration	R&D ($B)	Cost/Flt ($m)
A	Two Stage, Fully Reusable	15	4.5
B	Fly-Back Booster	8.2	6.3
C	Series Burn Pressure-Fed Booster	6.7	7.3
D	Current Shuttle	5.2	10.3

Table 5-5: Space Shuttle Configuration Options

The information on shuttle configurations and projected launch costs, budget constraints, and mission model were encoded as input to the DALP optimization routine. In order to force the system to include flights in the basis, a flat rate profit per flight was assumed to be paid to NASA per launch, independent of the type of booster. This allowed the routine to return a non-negative optimum: however, the feature described earlier whereby past year returns are applied to new capital was disabled, so that only the budget allocation could be used to pay for vehicle R&D or for flights.

Over a 30-year program, $1 billion per year results in a total budget of $30 billion. By looking at Table 5-5, it can be seen that this sum would be insufficient to fund all four candidate systems: therefore a set of four systems was not analyzed. Neither would the available funds pay for a three system solution with types A, B, and C. The first analysis run consisted of comparing systems A, C, and D. After comparing all three systems together, all combinations of two systems together, and all systems individually, the DALP routine identified option D alone as the most favored choice. The prioritized ranking of options was

1. Option D (current space shuttle)
2. Option C (partly reusable, pressure fed boosters)
3. Option C and D
4. Option A (fully reusable), C, and D
5. Option A
6. Option A and D
7. Option A and C

Since the high initial cost of option A was a disadvantage due to the cost discounting function of DALP, another optimization was made between systems B (flyback booster), C, and D. Again, option D (the current space shuttle design) was the most cost effective:

1. Option D
2. Option B, C, and D
3. Option C
4. Option B
5. Option C and D
6. Option B and D

7. Option B and C

It can therefore be seen that, whether knowingly or not, NASA took the proper choice in selecting the current shuttle configuration from among the available options. It should be emphasized that these results are biased by the cost discounting factors towards long-term expenditures (money spent a long ways in the future), but since the cost discounting is also applied in congressional budget estimates for NASA, the use of the technique in this application does not seem unreasonable. Total computer time for this study was 451 cpu-seconds on an IBM 370/168.

5.6 OPTIMIZATION OF STS UPPER STAGES

Another example of DALP applications offers the opportunity to analyze a smaller system than a solar power satellite or a launch system. The current Space Transportation System is limited in the orbit which it can reach to about 300-400 km with a full payload. Since many of the known payloads are intended for geostationary orbit, some method must be used of boosting these payloads into their intended orbits. Reference [55] presents three different options for this upper stage unit, summarized

	Upper Stage	R+D ($m)	Flight Cost ($m/ton)
A	Inertial Upper Stage	1	6.5
B	Modified Centaur	100	3.5
C	Orbital Transfer Vehicle	620	1.9

Table 5-6: Upper Stage Parameters

---

in Table 5-6. The same reference listed a possible market size of 800 tons to LEO during the course of the shuttle program.

As in the previous example, there is no direct monetary return from the use of an upper stage: by assuming a flat rate profit for each flight, the DALP algorithm will allow direct comparison of the candidate systems. The initial budget constraint was a limit of $100 million per year, which had to pay for R&D of the upper stages and direct upper stage flight costs (shuttle launch costs were not included, since they were the same for each of the options). The results indicated that the most favorable option was the procurement and use of both the interim upper stage and the modified Centaur.

Under the preferred case, the IUS would be procured initially, and used to launch 15 satellites in the initial two years of the STS program. After sufficient funds are found to bring the

Centaur stage to operational capability, all launch traffic would switch over to it. Nearly tied for second place in the options are developing either the IUS or the Centaur alone. All options including the OTV were well behind these three options, at the funding and flight levels assumed.

Another case was run, with the funding reduced to $50 million per year. In this case, the best solution found was to develop and use only the interim upper stage. Options following that were to use both the IUS and Centaur, followed by the option of developing the Centaur alone. Total computer time for this study was 732 cpu-seconds on an IBM 370/168.

6.0 CONCLUSIONS

The best things of mankind are as useless as Amelia Earhart's adventure. They are things that are undertaken, not for some definite, measurable result, but because someone, not counting the costs or calculating the consequences, is moved by curiosity, the love of excellence, a point of honor, the compulsion to invent, or to make or to understand. In such persons mankind overcomes the inertia which would keep it earthbound forever in its habitual ways. They have in them the free and useless energy with which men alone surpass themselves.

Such energy cannot be planned and managed and made purposeful or weighed by the standards of utility or judged by its social consequences. It is wild and free. But all the heroes, the saints and the seers, the explorers and the creators, partake of it. They can give no account in advance of where they are going, or explain completely where they have been. They are possessed for a time with the extraordinary passion which is unintelligible in ordinary terms.

No preconceived theory fits them. No material purpose actuates them. They do the useless, brave, noble, the divinely foolish and the very wisest things that are done by men. And what they may prove to themselves and to others is that man is no mere creature of his habits, no mere cog in the collective machine, but that in the dust of which he is made there is also fire, lighted now and again by great winds from the sky.

- Walter Lippmann

- Orbital Mechanics

  - Radiation considerations prevent the emplacement of a manned work site in earth orbit between low earth orbit and geostationary altitudes

- It is possible to construct a geometrical approximation which results in an analytical expression for the optimum plane change maneuver angles in a Hohmann transfer
- Impulsive trajectory approximations are adequate for accelerations above .2 g; continual thrust assumptions are adequate for thrust levels below .01 g
- The multiconic analysis technique with the universal variable formulation of the two-body problem forms a computationally effective method of analyzing restricted three-body motion
- The use of patched conic techniques for earth-moon trajectories provides a fair approximation of ΔV requirements, but a poor approximation of targeting parameters, resulting in the need for heuristic solutions for initial targeting angles.
- Use of a 70,000 km high earth orbit for processing, rather than geostationary, results in a drop in radiation dosage of two orders of magnitude, with the penalty of 600 m/sec extra ΔV required for total transfer from LEO to GEO with a stop at the processing facility at HEO
- The preferred altitude for a lunar orbit is as low as possible without danger of impacting the lunar highlands

- Three-body mechanics, particularly continual thrust trajectories in a three-body field, should be studied in greater depth

- Systems Definition

  - Minimum costs to low earth orbit do not necessarily correspond to maximum paylload fractions
  - Single stage to orbit vehicles with internal tanks are not cost effective at required ΔV to reach orbit
  - Use of parabolic extrapolation/interpolation allows quick and generally reliable optimization mathod for both single (i.e., optimized ΔV distribution) and double (ΔV and payload mass) sequential optimizations
  - If single stage vehicles with external, expendible tanks can be air launched without significant cost impact, they compare favorably with two stage to orbit vehicles with external tanks, and are generally less expensive in terms of $/kg to orbit than two stange totally reusable vehicles.
  - Optimum payload sizes are generally in the range of 5-10 thousand kilograms for earth launch systems
  - A wide variety of propulsion and power systems types are available for orbital transfer vehicles

- Line item costing

- It is possible to completely specify a space industrial scenario with 81 parameters
- Additional transport costs associated with carrying propellant as payload along certain legs of a trip can be calculated analytically, avoiding the problem with convergence of an iterative scheme

• Operations optimization

- The use of the diagonal ascent linear programming (DALP) method, derived in this study, allows the comparative assessment of competing systems, and will specify the most favorable program choices, including combinations and scheduling of multiple systems
- For the assumptions inherent in the sample case, the best method of producing power over a 30-year time span is to invest in ground-based power initially, and use the profits to develop nonterrestrial materials sources for solar power satellites starting in program year 14. The use of SPS's prefabricated on earth and assembled in low earth orbit does not appear economically attractive
- A review of program options available to NASA indicates that the choice of the current shuttle configuration with reusable solid rocket boosters and an expendable external tank was the best choice which could have been made based on available information.

- The best choice for space shuttle upper stages would be a mix of the interim upper stage (IUS) and modified Centaur stages. If cost constaints increase, the optimum will become the exclusive use of the interim upper stage.

A.0 HOHMANN TRANSFERS

Assume that the spacecraft is initially in a circular orbit. Orbital velocity can be found by

(A.1) $$V_c = \sqrt{\frac{\mu}{r_1}}$$

where μ is the gravitational constant for the central body, and r_1 is the orbital radius. It is desired to transfer to a circular orbit of radius r_2. The minimum energy transfer is a Hohmann orbit, which is an ellipse tangent to the inner circular orbit at periapsis and tangent to the outer orbit at apoapsis. The vis-viva equation relates velocity and position in an orbit:

(A.2) $$V = \sqrt{\mu\left(\frac{2}{r} - \frac{1}{a}\right)}$$

where a is the semi-major axis, which in this case is $.5(r_1+r_2)$. Assuming that the initial orbit is the inner one, equation (A.2) can be rewritten using (A.1) to find the velocity at periapsis of the transfer ellipse in terms of the initial and final radii and the circular orbit velocity at periapsis:

(A.3) $$V_{p1} = V_{c1}\sqrt{\frac{2r_2}{r_1+r_2}}$$

In this and following equations, subscript 1 refers to the initial orbit, and subscript 2 refers to the desired final orbit. The initial maneuver then involves accelerating from v_{c_1} to v_{p_1}, such that the vehicle is injected into the transfer orbit. Since the orbits are tangential at the injection point, the velocity change required for this maneuver is simply the difference between circular and periapsis velocities, or

$$\text{(A.4)} \qquad \Delta V_1 = V_{c_1}\left[\sqrt{\frac{2r_2}{r_1+r_2}} - 1\right]$$

It should be noted that this assumes that the spacecraft undergoes impulsive thrusting: that is, that the velocity change is made instantaneously. The impact of nonimpulsive thrusting will be analyzed later in this section. Similarly, the ΔV required for the second maneuver, which circularizes the orbit at the desired final radius, is found to be

$$\text{(A.5)} \qquad \Delta V_2 = V_{c_1}\sqrt{\frac{r_1}{r_2}}\left[1 - \sqrt{\frac{2r_1}{r_1+r_2}}\right]$$

Comparing (A.4) and (A.5), it should be noted that the two equations differ in two ways. The signs of the major terms are reversed, which indicates that apoapsis velocity is less than circular orbital velocity at that altitude; the spacecraft velocity must therefore be increased again in order to circularize its orbit. The presence of the r_1/r_2 term is necessary in order to keep the equation in terms of v_{c_1} , rather than introduce v_{c_2} , the circular velocity at the higher orbit.

The total velocity change required for this transfer can be found from adding (A.4) and (A.5). However, to examine the influence of the parameters of the transfer, it is desirable to introduce the nondimensional parameters $\rho=r_2/r_1$, and $\nu=v/v_{c_1}$. The velocity change equations can now be rewritten as

(A.6) $$\Delta \nu_1 = \sqrt{\frac{2}{1+\rho}} - 1$$

(A.7) $$\Delta \nu_2 = \sqrt{\rho}\left(1 - \sqrt{\frac{2\rho}{1+\rho}}\right)$$

In this nondimensional form, ρ varies from 1 (corresponding to no orbit change at all) to 0 (transfer to a new orbit at infinite radius, or parabolic escape). The eccentricity of the transfer ellipse, e, is also of interest as a parametric indicator. Since, at apoapsis,

(A.8) $$r_2 = a(1+e)$$

it will be left as an exercise to the reader to show that

(A.9) $$e = \frac{1-\rho}{1+\rho}$$

These relations provide a convenient set of equations which can be easily solved during preliminary mission planning, in order to estimate velocity change requirements for a variety of orbital transfer maneuvers.

B.0 TRAFFIC MODEL TO LOW EARTH ORBIT

The parametric analysis of earth launch systems in Chapter 3 found that the optimum payload size for a launch vehicle was substancially smaller than that of the current space shuttle orbiter. Depending on the mission model and capability of the vehicle to refly a number of time, the payload size which resulted in minimum launch costs was often 2000 kg or less. The assumption underlying this analysis, of course, is that the body of material needed in orbit can be subdivided into arbitrarily small bits for launch. In order to check this assumption, some "market research" was done on a NASA Space Transportation System traffic model.

The traffic model chosen [56] was never officially sanctioned by NASA, but was instead an attempt to quantify the type and volume of traffic which might be expected in a rapidly expanding space program. The total traffic to orbit in this model was 779 flights, far more than the current (July 1981) expected model of 350 flights throughout 1992. This mission model was chosen for two reasons:

- it represents an aggressive, expanding program in space engineering. While this has certainly not been the case to date, it might be argued that such an attitude towards space transportation is necessary for space industrialization to be a viable business alternative for the investment capital necessary

- it was available on punched cards, and did not have to be typed into the computer

A compilation of payload masses and V requirements is shown in Table B-1. In this instance, V is defined as additional velocity required from the shuttle initial circular orbit to the desired final orbit of the payload. The data contained in this table covers 375 payloads: Department of Defense payloads and shuttle upper stage uses are not included in this data. The payloads for the most part can be divided into five groupings:

- Planetary and Space Science (escape orbits)
- Communications (geostationary orbit)
- Scientific payloads (low masses, low orbits)
- Space Station and support (high masses, low orbit)
- Miscellaneous payloads (low orbits, assorted masses)

Of these categories, only the second currently retains some similarity to the current shuttle traffic model. However, the distribution distortion induced by the planetary missions is small (38 flights out of 375), while the spacecraft support

ΔV from LEO (km/sec)

Payload Mass (metric tons)

	.5	1.0	1.5	2.0	2.5	3.0	3.5	4.0	GEO	Escape	Totals
1	16	13	8			4		5	69	15	130
2	35								44	8	87
3	15								12	7	34
4										3	3
5	2									4	6
6	7									1	8
7	14										14
8	13										13
9	6										6
10	10										10
11	2										2
12	23										23
13											
14	1										1
15	35										35
16											
17	2										2
18	1										1
19											
Totals	182	13	8			4		5	125	38	375

Table B-1: Distribution of Payload Mass and V Requirements

---

missions represent a "worst case" for small launch vehicles, as they increase the number of payloads of larger sizes.

Figure 31 on page 198 shows the distribution of payload sizes within this traffic model. In this figure, 1000 kg is a 35

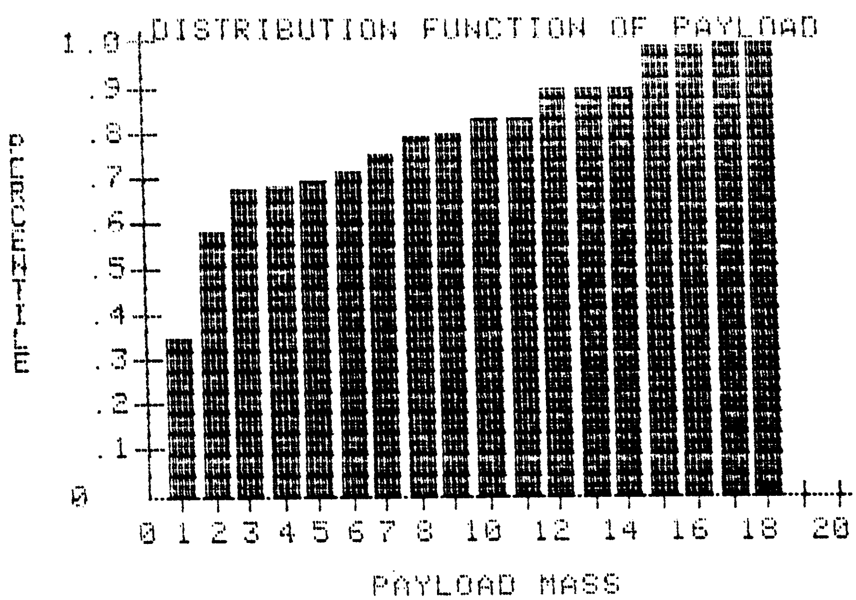

Figure 31. Distribution of Payload Sizes

---

percentile payload, and so on. As can be seen from this, 67% of all payloads in this data base are 3000 kg or less. This means that 251 payloads could be launched on a vehicle with a maximum payload of 3000 kg: in fact, doubling the payload capability to 6000 kg would only result in an additional market of 17 payloads. Therefore, a launch vehicle with a maximum payload capability of 3000 kg would be extremely effective in the market represented by this mission model.

However, this conclusion overlooks one factor: not all of the payloads are destined for low earth orbit. With the current space shuttle, payloads intended for high energy orbits are launched while attached to an upper stage. The entire assembly is jettisoned from the orbiter payload bay, then launched into the desired transfer orbit. This would not be possible with the small launch vehicle, as the propulsion system mass would reduce the usable payload to practically nil. As with the current mission model, the traffic to geostationary represents the largest single group of civil payloads for the small booster; some accomodation must be made for this class of spacecraft in order to remain competitive.

The suggested solution is to develop an operational transportation capability in low earth orbit for transferring and integrating payloads bound for outer orbits. The basic components of this system would be a checkout facility for upper stages, a space erectable reuseable upper stage, and a zero-gee refueling capability. Results from M.I.T. Space Systems Lab tests indicate that productivity of humans in weightless assembly is quite high ([17],[18]), so that assembling the propellant tanks and structure for a reusable propulsion stage should not present great difficulties. Refueling in weightlessness is now standard procedure for the Soviets ([58]), so orbital integration of payloads and upper stages represents the only real unknown.

There is another advantage to the orbital operations approach: since some flights of the "mini-shuttle" would be required to carry propellants for upper stages, additional propellant could be carried on all flights not fully loaded by the nominal payload. This would allow the load factor to be maintained near one over the lifetime of the system, thus reducing the incurred launch costs. Heinz [57] showed that such a scheme could save $6 billion over the course of the shuttle program; it would be even more beneficial to a small launch vehicle, where the severe size constraints would prevent most multi-payload missions, and almost all customers would be buying dedicated missions.

In addition to specifying the payload mass, it is important to determine the necessary physical size of the payload compartment. There is a tradeoff here between minimizing volume in order to keep structural weight down, and maximizing volume to decrease payload design constraints, and therefore payload costs. The current space shuttle has a payload bay which is 4.57m in diameter and 18.3m long, for a total volume of 300.5 m . With a maximum payload of 29,500 kg, this corresponds to an average payload density of 98 kg/m .

Applying the shuttle payload density to the small launch vehicle would result in a payload volume of 30.6 m , or (maintaining the same length/diameter ratio as the orbiter) a payload bay of 2.13m diameter by 8.54m long. Comparing this to the payloads in the mission model which fit within the 3000 kg payload limitation should give some indication of how well payload density scales with size. Taking the average values for length, diameter, and volume over the applicable payloads, weighted with the number of flights for payloads of each type, results in the following dimensions:

- Average length: 3.82 m (maximum = 7.62 m)
- Average diameter: 3.29 m (maximum = 4.57 m)
- Average volume: 39.75 m (maximum =125.1 m)

Thus, the payload sized from the mission model does not miss that scaled down from orbiter design constraints. This is due in large measure to the design of the current orbiter: payloads optimized for launch on the space shuttle must take maximum advantage of the diameter, and thus tend to be short and fat. Since it would not be possible to design the small launch vehicle to have the same payload diameter as the orbiter, there would be a lack of commonality in payload design between the two launch systems. This is similar to the current case of payloads designed to launch on either the space shuttle or an expendible booster: designed for both, they are optimized for neither. However, since the mini-shuttle would be an ongoing system, it might be assumed that payload designers could decide at the early design stages whether or not a mini-shuttle would be suitable for launching the payload, and design accordingly.

C.0 COMPUTER PROGRAMS

Room 14-0551
77 Massachusetts Avenue
Cambridge, MA 02139
Ph: 617.253.2800
Email: docs@mit.edu
http://libraries.mit.edu/docs

DISCLAIMER

MISSING PAGE(S)

Page 203 is missing from the original document.

```
C***************************************************************
C*   THIS PROGRAM CALCULATES ACTUAL AND APPROXIMATE DELTA-VEES
C*   BETWEEN LOW EARTH ORBIT (R=6675 KM) AND GEOSTATIONARY
C*   (R=42200 KM) FOR A TWO-BODY HOHMANN TRANSFER WITH
C*   INITIAL PLANE CHANGES BETWEEN 0 AND 10 DEGREES.
C*
C*   MARCH 5, 1981
C***************************************************************
      V1=10.15484
      VC1=7.72761
      V2=1.60625
      VC2=3.07337
      TH=28.5
      DTOR=.0174533
      THRAD=DTOR*TH
      DV1R=V1-VC1
      DV2R=SQRT(V2**2+VC2**2-2.*V2*VC2*COS(THRAD))
      VK=SIN(THRAD)*V2*VC2/DV2R
      WRITE(6,201)V1,VC1,V2,VC2,TH
      DO 1 I=1,41
      DEG=.25*FLOAT(I-1)
      DEL=DTOR*DEG
      DV1=SQRT(V1**2+VC1**2-2.*V1*VC1*COS(DEL))
      DV2=SQRT(V2**2+VC2**2-2.*V2*VC2*COS(THRAD-DEL))
      DV1A=SQRT(DV1R**2+(VC1*DEL)**2)
      DV2A=DV2R-DEL*VK
      DVT=DV1+DV2
      DVTA=DV1A+DV2A
 1    WRITE(6,202)DEG,DV1,DV1A,DV2,DV2A,DVT,DVTA
      THOPT=DV1R*VK/SQRT(VC1**4-VC1**2*VK**2)
      THOPTD=THOPT/DTOR
      WRITE(6,203)THOPTD
      STOP
 201  FORMAT(
     +20X,'PERIGEE VELOCITY   (KM/SEC):',F10.5/
     +20X,'LEO ORBIT VELOCITY (KM/SEC):',F10.5/
     +20X,'APOGEE VELOCITY    (KM/SEC):',F10.5/
     +20X,'GEO ORBIT VELOCITY (KM/SEC):',F10.5/
     +20X,'PLANE CHANGE ANGLE (DEG)   :',F10.5///)
 202  FORMAT(10X,F7.2,6F12.5)
 203  FORMAT(///20X,'PREDICTED OPTIMUM ANGLE (DEG):',F10.5)
      END
```

```
(*$S+*)

PROGRAM PWRFLITE;

  USES TRANSCEND;

  CONST MU=398604.0;
        G =9.8;

  TYPE STATEVEC=ARRAY[1..4] OF REAL;

  VAR STATE,ZERO,K1,K2,K3,K4:STATEVEC;
      R,V,T,TARGET,A,E,R0,TAU,C,DELTAT:REAL;
      I,INTERVAL:INTEGER;
      PRT:INTERACTIVE;
      HEIGHT:BOOLEAN;

  PROCEDURE DERIV(VAR KOUT:STATEVEC;STATE,KIN:STATEVEC;MULT:REAL);

    VAR R,V,TH,VR,OM,T1,T2,T3:REAL;

    BEGIN
      R:= STATE[1]+MULT*DELTAT*KIN[1];
      TH:=STATE[2]+MULT*DELTAT*KIN[2];
      VR:=STATE[3]+MULT*DELTAT*KIN[3];
      OM:=STATE[4]+MULT*DELTAT*KIN[4];
      V:=SQRT(VR*VR+R*R*OM*OM);

      KOUT[1]:=VR;
      KOUT[2]:=OM;
      T1:=TAU*C/(C-TAU*T)*VR/V;
      T2:=R*OM*OM;
      T3:=MU/(R*R);
      KOUT[3]:=T1+T2-T3;
      KOUT[4]:=TAU*C/(C-TAU*T)*OM/V-2*VR*OM/R;
    END;


  PROCEDURE UPDATE;

    CONST ONEOVER6=0.166667;

    VAR I:INTEGER;

    BEGIN
      FOR I:=1 TO 4 DO
        STATE[I]:=STATE[I]+DELTAT*(K1[I]+2*(K2[I]+K3[I])+K4[I])*ONEOVER6;
    END;
```

```
PROCEDURE PRTLINE;

  CONST RTOD=57.29578;

  VAR THDEG:REAL;

  BEGIN
    THDEG:=STATE[2]*RTOD;
    WRITELN(PRT,T:10:1,R:10:1,THDEG:10:4,V:10:4,A:10:1,E:10:4);
  END;
```

```
BEGIN
  T:=0;
  FOR I:=1 TO 4 DO
    ZERO[I]:=0;
  REWRITE(PRT,'PRINTER:');
  WRITE('ENTER INITIAL RADIUS (KM): ');
  READLN(R0);
  WRITE('ENTER TARGET RADIUS (KM): ');
  READLN(TARGET);
  WRITE('ENTER DELTA-T (SEC): ');
  READLN(DELTAT);
  WRITE('ENTER SPECIFIC IMPULSE (SEC): ');
  READLN(C);
  WRITE('ENTER INITIAL ACCELERATION (M/SEC): ');
  READLN(TAU);
  WRITE('ENTER PRINTING INTERVAL: ');
  READLN(INTERVAL);
  WRITELN(PRT);
  WRITELN(PRT,'ISP: ',C:10:2,'   T/M0: ',TAU:10:4);
  C:=C*G*0.001;
  TAU:=0.001*TAU;
  STATE:=ZERO;
  STATE[1]:=R0;
  R:=R0;
  V:=SQRT(MU/R0);
  A:=R0;
  E:=0;
  STATE[4]:=V/R0;
  WRITELN(PRT);
  I:=INTERVAL;
  REPEAT
    IF I=INTERVAL THEN BEGIN
      I:=0;
      PRTLINE;
    END;
    I:=I+1;
    DERIV(K1,STATE,ZERO,0);
    T:=T+0.5*DELTAT;
    DERIV(K2,STATE,K1,0.5);
    DERIV(K3,STATE,K2,0.5);
    T:=T+0.5*DELTAT;
    DERIV(K4,STATE,K3,1.0);
    UPDATE;
    R:=STATE[1];
    V:=SQRT(STATE[3]*STATE[3]+R*R*STATE[4]*STATE[4]);
    A:=1/(2/R-V*V/MU);
    E:=1-R*R*R*R*STATE[4]*STATE[4]/(MU*A);
    IF E<0 THEN E:=0;
    E:=SQRT(E);
    HEIGHT:=A*(1+E)>TARGET;
  UNTIL HEIGHT;
  PRTLINE;
END.
```

```
(*$S+*)

PROGRAM MULTICON;

  USES TRANSCEND,VECTORS,UNIVERSE,APPLESTUFF,TURTLEGRAPHICS,PRINTGRAPHICS;

  CONST MUEARTH=3.98604E5;
        MUMOON=4886.9;
        MURATIO=0.012112;

  VAR RBAR,VBAR,R2BODY,V2BODY,RHOI,RHOF,VRHOI,VRHOF,
      TEMPVEC,RMOON,VMOON,RMINIT,VMINIT,RTARG,VTARG,
      RENCOUNT,VENCOUNT: VECTOR;
      RM,VICINITY,T,ANGLE,EPOCH,DELTAT,TAREPOCH,TARGPER,TARGRAD:REAL;
      I,XPLT,YPLT:INTEGER;
      PRT:INTERACTIVE;
      BOTH,FIRST,ENCOUNTER:BOOLEAN;
      START,FINISH:STRING;

  PROCEDURE INPUT;

    BEGIN
      WRITE('ENTER INITIAL RADIUS VECTOR:    ');
      READLN(RBAR[1],RBAR[2],RBAR[3]);
      WRITE('ENTER INITIAL VELOCITY VECTOR: ');
      READLN(VBAR[1],VBAR[2],VBAR[3]);
      WRITE('ENTER INITIAL LUNAR ANGLE: ');
      READLN(ANGLE);
      EPOCH:=ANGLE/360;
      WRITE('ENTER TIMESTEP: ');
      READLN(DELTAT);
      T:=0;
      WRITE('ENTER TARGET RADIUS: ');
      READLN(TARGRAD);
      WRITE('ENTER TARGET PERIOD: ');
      READLN(TARGPER);
      WRITE('ENTER TARGET ANGLE: ');
      READLN(TAREPOCH);
      TAREPOCH:=TAREPOCH/360;
      WRITE('ENTER STARTING POINT: ');
      READLN(START);
      WRITE('ENTER FINAL POINT: ');
      READLN(FINISH);
      WRITE('ENTER ENCOUNTER RADIUS: ');
      READLN(VICINITY);
      REWRITE(PRT,'PRINTER:');
      FOR I:=1 TO 3 DO WRITELN(PRT);
      WRITELN(PRT,'              MULTICONIC ANALYSIS');
      WRITELN(PRT);
      WRITELN(PRT,'TRANSFER FROM',START,'TO',FINISH);
      WRITELN(PRT);
      WRITELN(PRT,'LUNAR POSITION (DEG): ',ANGLE:10:3);
      WRITELN(PRT,'INITIAL VELOCITY VECTOR: ',
               VBAR[1]:10:4,VBAR[2]:10:4,VBAR[3]:10:4);
      WRITELN(PRT);
      WRITELN(PRT,'   TIME        R(X)        R(Y)        R(Z)     ',
               '    V(X)        V(Y)        V(Z)');
      WRITELN(PRT);
    END;
```

```
PROCEDURE MOONTRAK(VAR RHO,VRHO:VECTOR;T:REAL);

  CONST PERIOD=2.36062E6;
        VELOCITY=1.022;
        DISTANCE=3.844E5;
        TWOPI=6.283185;

  VAR A:REAL;

  BEGIN
    A:=TWOPI*(T/PERIOD+EPOCH);
    RHO[1]:=DISTANCE*COS(A);
    RHO[2]:=DISTANCE*SIN(A);
    RHO[3]:=0;
    VRHO[1]:=-VELOCITY*SIN(A);
    VRHO[2]:=VELOCITY*COS(A);
    VRHO[3]:=0;
  END;

PROCEDURE TARGTRAK(VAR RTARG,VTARG:VECTOR;T:REAL);

  CONST TWOPI=6.283185;
  VAR A,VEL :REAL;

  BEGIN
    A:=TWOPI*(T/TARGPER+TAREPOCH);
    RTARG[1]:=TARGRAD*COS(A);
    RTARG[2]:=TARGRAD*SIN(A);
    RTARG[3]:=0;
    VEL:=TWOPI*TARGRAD/TARGPER;
    VTARG[1]:=-VEL*SIN(A);
    VTARG[2]:=VEL*COS(A);
    VTARG[3]:=0;
  END;

PROCEDURE INITPLOT;

  VAR I,J:INTEGER;

  BEGIN
    INITTURTLE;
    FOR I:=-4 TO 4 DO
      BEGIN
        J:=140+25*I;
        PENCOLOR(NONE);
        MOVETO(J,99);
        PENCOLOR(WHITE);
        MOVETO(J,94);
      END;
```

```
    FOR I:=-3 TO 3 DO
      BEGIN
        J:=96+25*I;
        PENCOLOR(NONE);
        MOVETO(137,J);
        PENCOLOR(WHITE);
        MOVETO(143,J);
      END;
      PENCOLOR(NONE);
  END;


PROCEDURE PRTLINE;

  BEGIN
    WRITELN(PRT,T:10:1,RBAR[1]:10:1,RBAR[2]:10:1,RBAR[3]:10:1,
                  VBAR[1]:10:4,VBAR[2]:10:4,VBAR[3]:10:4);
  END;


BEGIN

  INPUT;
  FIRST:=TRUE;
  MOONTRAK(RHOI,VRHOI,T);
  R2BODY:=RBAR;
  V2BODY:=VBAR;
  PRTLINE;

  REPEAT
    VECSCALE(TEMPVEC,-1,RHOI);
    VECADD(RMOON,RBAR,TEMPVEC);
    VECSCALE(TEMPVEC,-1,VRHOI);
    VECADD(VMOON,VBAR,TEMPVEC);
    RMINIT:=RMOON;
    VMINIT:=VMOON;
    T:=T+DELTAT;
    MOONTRAK(RHOF,VRHOF,T);
    TWOBODY(R2BODY,V2BODY,MUEARTH,DELTAT);
    TWOBODY(RBAR,VBAR,MUEARTH,DELTAT);
    TWOBODY(RMOON,VMOON,MUMOON,DELTAT);
    FOR I:=1 TO 3 DO BEGIN
      VBAR[I]:=VBAR[I]+VMOON[I]-VMINIT[I]+MURATIO*(VRHOF[I]-VRHOI[I]);
      RBAR[I]:=RBAR[I]+RMOON[I]-RMINIT[I]-DELTAT*VMINIT[I]+
             MURATIO*(RHOF[I]-RHOI[I]-DELTAT*VRHOI[I]);
      END;
```

```
    RHOI:=RHOF;
    VRHOI:=VRHOF;
    TARGTRAK(RTARG,VTARG,T);
    VECSCALE(TEMPVEC,-1,RTARG);
    VECADD(RENCOUNT,RBAR,TEMPVEC);
    VECSCALE(TEMPVEC,-1,VTARG);
    VECADD(VENCOUNT,VBAR,TEMPVEC);
    WRITELN('T: ',T:10:1,RENCOUNT[1]:10:1,RENCOUNT[2]:10:1,RENCOUNT[3]:10:1);
    PRTLINE;
    RM:=MAGNITUDE(RENCOUNT);
    IF RM<VICINITY THEN ENCOUNTER:=TRUE
                   ELSE ENCOUNTER:=FALSE;
    BOTH:=ENCOUNTER AND FIRST;
    IF BOTH THEN
      BEGIN
        FIRST:=FALSE;
        DELTAT:=0.1*DELTAT;
        INITPLOT;
        XPLT:=TRUNC(0.005*RENCOUNT[1])+140;
        YPLT:=TRUNC(0.005+RENCOUNT[2])+96;
        MOVETO(XPLT,YPLT);
        PENCOLOR(WHITE);
      END;
    IF ENCOUNTER THEN BEGIN
      WRITELN(PRT,'E: ',T:8:1,RM:8:1,
                 RENCOUNT[1]:10:1,RENCOUNT[2]:10:1,RENCOUNT[3]:10:1,
                 VENCOUNT[1]:10:4,VENCOUNT[2]:10:4,VENCOUNT[3]:10:4);
      XPLT:=TRUNC(0.005*RENCOUNT[1])+140;
      YPLT:=TRUNC(0.005*RENCOUNT[2])+96;
      MOVETO(XPLT,YPLT);
      END;
  UNTIL KEYPRESS;

  IF ENCOUNTER THEN BEGIN
    SETUNIDIRECT;
    SETNEGATIVE;
    SETDARK(7);
    PRINTPIC;
    RESTORE;
  END;

  TEXTMODE;

END.
```

```
(*$S++)

UNIT UNIVERSE; INTRINSIC CODE 28;

  INTERFACE

    USES TRANSCEND,VECTORS;
    PROCEDURE TWOBODY(VAR RBAR,VBAR:VECTOR;MU,DELTAT:REAL);

  IMPLEMENTATION

  PROCEDURE TWOBODY;

    CONST XLIMIT=0.0001;

    VAR R0,V0,ONEOVERA,X,Z,DOTFAC,RADFAC,SQRTMU,
        CN,SN,DTDX,TN,XERROR,F,G,R,FDOT,GDOT,A,SQRTA:REAL;

        TEMP1,TEMP2,RNEW:VECTOR;


    FUNCTION COSH(X:REAL):REAL;

      BEGIN
        COSH:=0.5*(EXP(X)+EXP(-X))
      END;


    FUNCTION SINH(X:REAL):REAL;

      BEGIN
        SINH:=0.5*(EXP(X)-EXP(-X))
      END;


    FUNCTION C(Z:REAL):REAL;

      BEGIN
        IF Z=0 THEN C:=0 ELSE
          IF Z>0 THEN C:=(1-COS(SQRT(Z)))/Z
                 ELSE C:=(1-COSH(SQRT(-Z)))/Z
      END;
```

```
FUNCTION S(Z:REAL):REAL;

  BEGIN
    IF Z=0 THEN S:=0.166667 ELSE
      IF Z>0 THEN S:=(SQRT(Z)-SIN(SQRT(Z)))/SQRT(Z*Z*Z)
             ELSE S:=(SINH(SQRT(-Z))-SQRT(-Z))/SQRT(-Z*Z*Z)
  END;


BEGIN
  R0:=MAGNITUDE(RBAR);
  V0:=MAGNITUDE(VBAR);
  ONEOVERA:=(2*MU/R0-V0*V0)/MU;
  SQRTMU:=SQRT(MU);
  DOTFAC:=DOT(RBAR,VBAR)/SQRTMU;
  RADFAC:=1-R0*ONEOVERA;
  IF ONEOVERA>0
    THEN X:=DELTAT*SQRTMU*ONEOVERA
    ELSE BEGIN
      A:=1/ONEOVERA;
      SQRTA:=SQRT(-A);
      X:=SQRTA*LN(-2*MU*DELTAT*ONEOVERA/
                 (DOT(RBAR,VBAR)+SQRTMU*SQRTA*RADFAC));
    END;

  REPEAT
    Z:=X*X*ONEOVERA;
    CN:=C(Z);
    SN:=S(Z);
    DTDX:=(X*X*CN+DOTFAC*X*(1-Z*SN)+R0*(1-Z*CN))/SQRTMU;
    TN:=(DOTFAC*X*X*CN+RADFAC*X*X*X*SN+R0*X)/SQRTMU;
    XERROR:=(DELTAT-TN)/DTDX;
    X:=X+XERROR;
  UNTIL ABS(XERROR)<XLIMIT;

  Z:=X*X*ONEOVERA;
  F:=1-X*X*C(Z)/R0;
  G:=DELTAT-X*X*X*S(Z)/SQRTMU;
  VECSCALE(TEMP1,F,RBAR);
  VECSCALE(TEMP2,G,VBAR);
  VECADD(RNEW,TEMP1,TEMP2);
  R:=MAGNITUDE(RNEW);
  GDOT:=1-X*X*C(Z)/R;
  FDOT:=SQRTMU*X*(Z*S(Z)-1)/(R0*R);
  VECSCALE(TEMP1,FDOT,RBAR);
  VECSCALE(TEMP2,GDOT,VBAR);
  VECADD(VBAR,TEMP1,TEMP2);
  RBAR:=RNEW;
 END;
END.
```

```
(*$S+*)
UNIT VECTORS; INTRINSIC CODE 25;

INTERFACE

  USES TRANSCEND;
  TYPE       VECTOR=ARRAY[1..3] OF REAL;
  PROCEDURE CROSS(VAR A:VECTOR;B,C:VECTOR);
  PROCEDURE VECSCALE(VAR A:VECTOR;X:REAL;B:VECTOR);
  PROCEDURE VECEQUAL(VAR A:VECTOR;B:VECTOR);
  PROCEDURE VECADD(VAR A:VECTOR;B,C:VECTOR);
  FUNCTION  DOT(A,B:VECTOR):REAL;
  FUNCTION  MAGNITUDE(A:VECTOR):REAL;

IMPLEMENTATION

  VAR I:INTEGER;

  PROCEDURE CROSS;

    BEGIN
      A[1]:=B[2]*C[3]-B[3]*C[2];
      A[2]:=B[3]*C[1]-B[1]*C[3];
      A[3]:=B[1]*C[2]-B[2]*C[1];
    END; (*CROSS*)

  PROCEDURE VECSCALE;

    BEGIN
      FOR I:=1 TO 3 DO
        A[I]:=X*B[I];
    END; (*VECSCALE*)

  PROCEDURE VECEQUAL;

    BEGIN
      FOR I:=1 TO 3 DO
        A[I]:=B[I];
    END; (*VECEQUAL*)
```

```
PROCEDURE VECADD;

   BEGIN
     FOR I:=1 TO 3 DO
       A[I]:=B[I]+C[I];
   END; (*VECADD*)

FUNCTION DOT;

   VAR TEMP:REAL;

   BEGIN
     TEMP:=0;
     FOR I:=1 TO 3 DO
       TEMP:=TEMP+(A[I]*B[I]);
       DOT:=TEMP;
   END; (*DOT*)

FUNCTION MAGNITUDE;

   VAR TEMP:REAL;

   BEGIN
     TEMP:=0;
     FOR I:=1 TO 3 DO
       TEMP:=TEMP+A[I]*A[I];
     MAGNITUDE:=SQRT(TEMP);
   END; (*MAGNITUDE*)

END.
```

```
C*************************************************************
C*  PROGRAM SSTO: CALCULATES COST/KG TO LEO FOR PAYLOAD CARRIED
C*  ON A SINGLE STAGE TO ORBIT (SSTO) VEHICLE
C*  D. L. AKIN                                     7/11/81
C************************************************************
      IMPLICIT REAL(M)
      DIMENSION DEL(5),B(3),X(3),COSTEC(10)
      COMMON/COSTS/MO,CO,C1,Y,FLTPV,RINT,EXPLC
      COMMON/VEHICL/DELPL,DVTOT,SPI,MFE
C************************************************************
C*  READ IN STAGE MASS FACTORS:
C*    DEL(1)=FUSELAGE MASS/MASS CONTAINED
C*    DEL(2)=PROPELLENT MASS/EMPTY MASS
C*    DEL(3)=MASS OF LIFT COMPONENTS/MASS CARRIED
C*    DEL(4)=PROPULSION SYSTEM MASS/MASS CARRIED
C*    DEL(5)=THERMAL PROTECTION SYSTEM MASS/MASS PROTECTED
C*
      READ(5,101)(DEL(I),I=1,5)
C************************************************************
C*  READ IN COSTING VALUES:
C*    MO=TOTAL LAUNCH MASS OF PROGRAM (MT)
C*    CO=$/KG R&D
C*    C1=$/KG INITIAL PRODUCTION
C*    Y=OPERATIONAL PROGRAM YEARS
C*    FLTPV=FLIGHTS PER VEHICLE
C*
      READ(5,101)MO,CO,C1,Y,FLTPV
      MO=MO*1000.
C*************************************************************
C*  READ IN COSTING AND VEHICLE PARAMETERS:
C*    DELPL=0 FOR EXTERNAL PAYLOAD, 1 FOR INTERNAL
C*    RINT=INTEREST RATE FOR COST DISCOUNTING
C*    DVTOT=TOTAL DELTA-V (M/SEC)
C*    SPI=SPECIFIC IMPULSE
C*    XNULL=NULL VARIABLE
C*
      READ(5,101)DELPL,RINT,DVTOT,SPI,XNULL
C*************************************************************
C*  READ IN COSTING AND VEHICLE PARAMETERS:
C*    MFE=STAGE FIXED EQUIPMENT MASS (KG)
C*    EXPLC=EXPONENT OF LEARNING CURVE
C*
      READ(5,102)MFE,EXPLC
C*
C*  STARTING POINT FOR CALCULATIONS
C*
      MPLLIM=50.
      MPLDEL=500.
      DO 3 I=1,10
      MO=1.E7 * FLOAT(I*I)
      MPL=5000.
 1    B(1)=MPL+MPLDEL
      B(2)=MPL
      B(3)=MPL-MPLDEL
      DO 2 I2=1,3
```

```
      CALL MASCAL(B(I2),DEL,X(I2),XLAMB)
 2    WRITE(6,201)MO,B(I2),XLAMB,X(I2)
      CALL PARAB(B,X,MPL,DIFF,MPLDEL)
      IF (DIFF.GE.MPLLIM) GO TO 1
      CALL MASCAL(MPL,DEL,COSTEO,XLAMB)
 3    WRITE(6,201)MO,MPL,XLAMB,COSTEO
      STOP
 101  FORMAT(5F8.3)
 102  FORMAT(2F8.3)
 201  FORMAT(' MO=',F15.0,' MPL=',F8.0,' LAMBDA=',F8.4,
     +       ' COST TO LEO=',F10.2)
      END
      SUBROUTINE PARAB(B,X,VAL,DIFF,DEL)
      DIMENSION B(3),X(3)
      XVAL=X(1)-2.*X(2)+X(3)
      VALNEW=-.5*((-2.*B(2)*XVAL+DEL*(X(1)-X(3)))/XVAL)
      IF (VALNEW.LE.0.) VALNEW=.5*VAL
      DIFF=ABS(VALNEW-VAL)
      VAL=VALNEW
      RETURN
      END
      SUBROUTINE MASCAL(MPL,DEL,CPERKG,XLAMB)
C**************************************************************
C*  BASED ON VEHICLE PARAMETERS, THIS SUBROUTINE CALCULATES
C*  THE VEHICLE COMPONENT MASSES, INCLUDING ITERATING FOR
C*  THE NONLINEAR PROPELLENT TANK MASS TERMS.
C**************************************************************
      IMPLICIT REAL(M)
      DIMENSION DEL(5),M(5),MX(3),A(5,5),B(5)
      COMMON/VEHICL/DELPLT,DVTOT,SPI,MFE
      DATA G/9.8/
      DELPL=DELPLT
      R=EXP(-DVTOT/(G*SPI))
      DEL(2)=(1.-R)/R
      MX(1)=MPL
      MX(2)=MFE
      MX(3)=MPL
 1    CALL ASETUP (A,DEL)
      CALL BSETUP(B,MX,DEL,DELPL)
      CALL MAXSOL(A,B,M)
C     WRITE(6,301)(M(I),I=1,5)
      CALL MTANK(M,MX,DIFF)
C     WRITE(6,302)MX(1)
      IF (ABS(DIFF).GT.10.) GO TO 1
      MEMPTY=0.
      DO 4 I=1,5
      IF (I.EQ.2) GO TO 4
      MEMPTY=MEMPTY+M(I)
 4    CONTINUE
      DO 5 I=1,2
 5    MEMPTY=MEMPTY+MX(I)
      CPERKG=COST(MEMPTY,MPL)
      XLAMB=MPL*R/(MPL+MEMPTY)
      RETURN
 201  FORMAT(' DELTA-V=',F7.0,' MPL=',F7.0,' MEMPTY=',F10.0,' C=',F9.2)
```

```
 301   FORMAT(' MASSES: ',5F9.0)
 302   FORMAT(' TANK MASS: ',F9.0)
       END
       SUBROUTINE ASETUP(A,DEL)
C************************************************************
C*  SETS UP A MATRIX AS PER NOTES 11/18/79
C************************************************************
       DIMENSION A(5,5),DEL(5)
       DO 2 I=1,5
       DO 1 J=1,5
 1     A(I,J)=-DEL(I)
 2     A(I,I)=1.
       A(1,3)=0.
       A(1,2)=0.
       A(1,5)=0.
       A(3,2)=0.
       A(3,3)=1.-DEL(3)
       A(4,4)=1.-DEL(4)
       A(5,2)=0.
       RETURN
       END
       SUBROUTINE BSETUP(B,MX,DEL,DELPL)
C************************************************************
C*  SETS UP B MATRIX, AS PER NOTES 11/18/79
C************************************************************
       IMPLICIT REAL (M)
       DIMENSION B(5),DEL(5),MX(3)
       DO 1 I=1,5
       FACT=DELPL
       IF (I.EQ.2.OR.I.EQ.4) FACT=1.
 1     B(I)=DEL(I)*(MX(1)+MX(2)+FACT*MX(3))
       RETURN
       END
       SUBROUTINE MTANK(M,MX,DIFF)
C************************************************************
C*  CALCULATES NEW VALUE OF TANK MASS BASED ON PROPELLANT
C*  MASS, THEN FINDS DIFFERENCE BETWEEN NEW AND OLD VALUES
C************************************************************
       IMPLICIT REAL(M)
       DIMENSION M(5),MX(3)
       MT=.2*M(2)**.9
       DIFF=MT-MX(1)
       MX(1)=MT
       RETURN
 301   FORMAT(' ENTERING MTANK:',G14.5)
       END
       FUNCTION COST(ME,MPL)
C************************************************************
C*  CALCULATES LAUNCH COSTS IN $/KG OF PAYLOAD, AS PER NOTES
C************************************************************
       IMPLICIT REAL(M)
       COMMON /COSTS/MO,CO,C1,Y,FLTPV,RINT,EXPLC
       V=MO/(MPL*FLTPV)
       NV=IFIX(V)
C      WRITE(6,301)NV,RINT,Y,EXPLC
```

```
      SUM=0.
      EXTRA=0.
      DO 1 I=1,NV
      X=FLOAT(I)
      EXTRA=EXTRA+X**EXPLC*(1.+RINT)**(-X*Y/FLOAT(NV+1))
 1    SUM=SUM+X**EXPLC*(1.+RINT)**(-X*Y/FLOAT(NV))
      X=X+1.
      EXTRA=EXTRA+X**EXPLC*(1.+RINT)**(-X*Y/FLOAT(NV+1))
      AVG=(EXTRA-SUM)*(V-FLOAT(NV))+SUM
      COST=ME/MO*(CO+C1*AVG)
      RETURN
 301  FORMAT(1X,I10,3F10.2)
      END
      SUBROUTINE MAXSOL(A,B,X)
C***************************************************************
C*  FINDS THE SOLUTION TO A SET OF 5 SIMULTANEOUS EQUATIONS
C*  OF THE FORM <A><X>=<B>.  SOLUTION IS BY PIVOTING.
C***************************************************************
      DIMENSION A(5,5),B(5),X(5)
      DO 3 I=1,4
      DIV=A(I,I)
      DO 1 J=1,5
 1    A(I,J)=A(I,J)/DIV
      B(I)=B(I)/DIV
      I1=I+1
      DO 4 K=I1,5
      DIV=A(K,I)
      IF (DIV.EQ.0.) GO TO 4
      B(K)=B(K)/DIV-B(I)
      DO 2 J=1,5
 2    A(K,J)=A(K,J)/DIV-A(I,J)
 4    CONTINUE
 3    CONTINUE
      B(5)=B(5)/A(5,5)
      A(5,5)=1.
      X(5)=B(5)
      X(4)=B(4)-A(4,5)*X(5)
      X(3)=B(3)-A(3,5)*X(5)-A(3,4)*X(4)
      X(2)=B(2)-A(2,5)*X(5)-A(2,4)*X(4)-A(2,3)*X(3)
      X(1)=B(1)-A(1,5)*X(5)-A(1,4)*X(4)-A(1,3)*X(3)-A(1,2)*X(2)
      RETURN
      END
      SUBROUTINE PRTARY(A)
C***************************************************************
C*  PRINTS OUT AN 5X5 ARRAY
C***************************************************************
      DIMENSION A(5,5)
      DO 1 J=1,5
 1    WRITE(6,101)(A(J,I),I=1,5)
      RETURN
 101  FORMAT(1X,6G10.4)
      END
```

```
C*************************************************************
C*  PROGRAM SSET: CALCULATES COST/KG TO LEO FOR PAYLOAD CARRIED
C*  ON A SINGLE STAGE VEHICLE WITH EXTERNAL TANKS (SSET)
C*  D. L. AKIN                                       7/11/81
C************************************************************
      IMPLICIT REAL(M)
      DIMENSION DEL(5),B(3),X(3),COSTEC(10)
      COMMON/COSTS/MO,CO,C1,Y,FLTPV,RINT,EXPLC
      COMMON/VEHICL/DELPL,DVTOT,SPI,MFE
C************************************************************
C*  READ IN STAGE MASS FACTORS:
C*    DEL(1)=FUSELAGE MASS/MASS CONTAINED
C*    DEL(2)=PROPELLENT MASS/EMPTY MASS
C*    DEL(3)=MASS OF LIFT COMPONENTS/MASS CARRIED
C*    DEL(4)=PROPULSION SYSTEM MASS/MASS CARRIED
C*    DEL(5)=THERMAL PROTECTION SYSTEM MASS/MASS PROTECTED
C*
      READ(5,101)(DEL(I),I=1,5)
C************************************************************
C*  READ IN COSTING VALUES:
C*    MO=TOTAL LAUNCH MASS OF PROGRAM (MT)
C*    CO=$/KG R&D
C*    C1=$/KG INITIAL PRODUCTION
C*    Y=OPERATIONAL PROGRAM YEARS
C*    FLTPV=FLIGHTS PER VEHICLE
C*
      READ(5,101)MO,CO,C1,Y,FLTPV
      MO=MO*1000.
C*************************************************************
C*  READ IN COSTING AND VEHICLE PARAMETERS:
C*    DELPL=0 FOR EXTERNAL PAYLOAD, 1 FOR INTERNAL
C*    RINT=INTEREST RATE FOR COST DISCOUNTING
C*    DVTOT=TOTAL DELTA-V (M/SEC)
C*    SPI=SPECIFIC IMPULSE
C*    XNULL=NULL VARIABLE
C*
      READ(5,101)DELPL,RINT,DVTOT,SPI,XNULL
C*************************************************************
C*  READ IN COSTING AND VEHICLE PARAMETERS:
C*    MFE=STAGE FIXED EQUIPMENT MASS (KG)
C*    EXPLC=EXPONENT OF LEARNING CURVE
C*
      READ(5,102)MFE,EXPLC
C*
C*  STARTING POINT FOR CALCULATIONS
C*
      MPLLIM=50.
      MPLDEL=500.
      DO 3 I=1,10
      MO=1.E7 * FLOAT(I*I)
      MPL=2500.
 1    B(1)=MPL+MPLDEL
      B(2)=MPL
      B(3)=MPL-MPLDEL
      DO 2 I2=1,3
```

```
      CALL MASCAL(B(I2),DEL,X(I2),XLAMB)
 2    WRITE(6,201)MO,B(I2),XLAMB,X(I2)
      CALL PARAB(B,X,MPL,DIFF,MPLDEL)
      IF (DIFF.GE.MPLLIM) GO TO 1
      CALL MASCAL(MPL,DEL,COSTEO,XLAMB)
 3    WRITE(6,201)MO,MPL,XLAMB,COSTEO
      STOP
 101  FORMAT(5F8.3)
 102  FORMAT(2F8.3)
 201  FORMAT(' MO=',F15.0,' MPL=',F8.0,' LAMBDA=',F8.4,
     +       ' COST TO LEO=',F10.2)
      END
      SUBROUTINE PARAB(B,X,VAL,DIFF,DEL)
      DIMENSION B(3),X(3)
      XVAL=X(1)-2.*X(2)+X(3)
      VALNEW=-.5*((-2.*B(2)*XVAL+DEL*(X(1)-X(3)))/XVAL)
      IF (VALNEW.LE.0.) VALNEW=.5*VAL
      DIFF=ABS(VALNEW-VAL)
      VAL=VALNEW
      RETURN
      END
      SUBROUTINE MASCAL(MPL,DEL,CPERKG)
C***********************************************************
C*   BASED ON VEHICLE PARAMETERS, THIS SUBROUTINE CALCULATES
C*   THE VEHICLE COMPONENT MASSES, INCLUDING ITERATING FOR
C*   THE NONLINEAR PROPELLENT TANK MASS TERMS.
C***********************************************************
      IMPLICIT REAL(M)
      DIMENSION DEL(5),M(5),MX(3),A(5,5),B(5)
      COMMON/VEHICL/DELPLT,DVTOT,SPI,MFE
      DATA G/9.8/
      DELPL=DELPLT
      R=EXP(-DVTOT/(G*SPI))
      DEL(2)=(1.-R)/R
      MX(1)=MPL
      MX(2)=MFE
      MX(3)=MPL
 1    CALL ASETUP (A,DEL)
      CALL BSETUP(B,MX,DEL,DELPL)
      CALL MAXSOL(A,B,M)
C*    WRITE(6,301)(M(I),I=1,5)
      CALL MTANK(M,MX,DIFF)
C*    WRITE(6,302)MX(1)
      IF (ABS(DIFF).GT.10.) GO TO 1
      MEMPTY=0.
      DO 4 I=1,5
      IF (I.EQ.2) GO TO 4
      MEMPTY=MEMPTY+M(I)
 4    CONTINUE
      MEMPTY=MEMPTY+MX(2)
      CPERKG=COST(MEMPTY,MPL,MX)
      WRITE(6,201)DVTOT,MPL,MEMPTY,MX(1),CPERKG
      RETURN
 201  FORMAT(' DEL-V,MPL,MORB,MTANK,C: ',4F10.0,F9.2)
 301  FORMAT(' MASSES: ',5F9.0)
```

```
 302  FORMAT(' TANK MASS: ',F9.0)
      END
      SUBROUTINE ASETUP(A,DEL)
C************************************************************
C*  SETS UP A MATRIX AS PER NOTES 11/18/79
C************************************************************
      DIMENSION A(5,5),DEL(5)
      DO 2 I=1,5
      DO 1 J=1,5
 1    A(I,J)=-DEL(I)
 2    A(I,I)=1.
      A(1,3)=0.
      A(1,2)=0.
      A(1,5)=0.
      A(3,2)=0.
      A(3,3)=1.-DEL(3)
      A(4,4)=1.-DEL(4)
      A(5,2)=0.
      RETURN
      END
      SUBROUTINE BSETUP(B,MX,DEL,DELPL)
C************************************************************
C*  SETS UP B MATRIX, AS PER NOTES 11/18/79
C************************************************************
      IMPLICIT REAL (M)
      DIMENSION B(5),DEL(5),MX(3)
      DO 1 I=1,5
      FACT=DELPL
      FACTOR=0.
      IF (I.EQ.2.OR.I.EQ.4) FACTOR=1.
      IF (I.EQ.2.OR.I.EQ.4) FACT=1.
 1    B(I)=DEL(I)*(FACTOR*MX(1)+MX(2)+FACT*MX(3))
      RETURN
      END
      SUBROUTINE MTANK(M,MX,DIFF)
C************************************************************
C*  CALCULATES NEW VALUE OF TANK MASS BASED ON PROPELLANT
C*  MASS, THEN FINDS DIFFERENCE BETWEEN NEW AND OLD VALUES
C************************************************************
      IMPLICIT REAL(M)
      DIMENSION M(5),MX(3)
      MT=.2*M(2)**.9
      DIFF=MT-MX(1)
      MX(1)=MT
      RETURN
 301  FORMAT(' ENTERING MTANK:',G14.5)
      END
      FUNCTION COST(ME,MPL,MX)
C************************************************************
C*  CALCULATES LAUNCH COSTS IN $/KG OF PAYLOAD, AS PER NOTES
C************************************************************
      IMPLICIT REAL(M)
      DIMENSION MX(3)
      COMMON /COSTS/MO,CO,C1,Y,FLTPV,RINT,EXPLC
      V=MO/(MPL*FLTPV)
```

```
      NV=IFIX(V)
      IF (NV.EQ.O) NV=1
      SUM=O.
      EXTRA=O.
      DO 1 I=1,NV
      X=FLOAT(I)
      EXTRA=EXTRA+X**EXPLC*(1.+RINT)**(-X*Y/FLOAT(NV+1))
 1    SUM=SUM+X**EXPLC*(1.+RINT)**(-X*Y/FLOAT(NV))
      X=X+1.
      EXTRA=EXTRA+X**EXPLC*(1.+RINT)**(-X*Y/FLOAT(NV+1))
      AVG=(EXTRA-SUM)*(V-FLOAT(NV))+SUM
      COST=ME/MO*(CO+C1*AVG)
      COST=COST+MX(1)/MO*(CO+C1*(MO/MPL)**EXPLC/(1.+EXPLC))
      RETURN
      END
      SUBROUTINE MAXSOL(A,B,X)
C***********************************************************
C*   FINDS THE SOLUTION TO A SET OF 5 SIMULTANEOUS EQUATIONS
C*   OF THE FORM <A><X>=<B>.  SOLUTION IS BY PIVOTING.
C***********************************************************
      DIMENSION A(5,5),B(5),X(5)
      DO 3 I=1,4
      DIV=A(I,I)
      DO 1 J=1,5
 1    A(I,J)=A(I,J)/DIV
      B(I)=B(I)/DIV
      I1=I+1
      DO 4 K=I1,5
      DIV=A(K,I)
      IF (DIV.EQ.O.) GO TO 4
      B(K)=B(K)/DIV-B(I)
      DO 2 J=1,5
 2    A(K,J)=A(K,J)/DIV-A(I,J)
 4    CONTINUE
 3    CONTINUE
      B(5)=B(5)/A(5,5)
      A(5,5)=1.
      X(5)=B(5)
      X(4)=B(4)-A(4,5)*X(5)
      X(3)=B(3)-A(3,5)*X(5)-A(3,4)*X(4)
      X(2)=B(2)-A(2,5)*X(5)-A(2,4)*X(4)-A(2,3)*X(3)
      X(1)=B(1)-A(1,5)*X(5)-A(1,4)*X(4)-A(1,3)*X(3)-A(1,2)*X(2)
      RETURN
      END
      SUBROUTINE PRTARY(A)
C***********************************************************
C*   PRINTS OUT AN 5X5 ARRAY
C***********************************************************
      DIMENSION A(5,5)
      DO 1 J=1,5
 1    WRITE(6,101)(A(J,I),I=1,5)
      RETURN
 101  FORMAT(1X,6G10.4)
      END
```

```
C***************************************************************
C*  THIS PROGRAM DESIGNS TWO-STAGE LAUNCH VEHICLES FOR
C*  A VARIETY OF PAYLOAD MASSES TO LEO, USING A PARABOLIC
C*  EXTRAPOLATION TO INTERATE FOR OPTIMUM STAGING VELOCITY
C*  D. L. AKIN                                 7/12/81
C***************************************************************
      IMPLICIT REAL(M)
      DIMENSION DEL1(5),DEL2(5),B(3),X(3)
      COMMON/COSTS/MO,CO,C1,Y,FLTPV,RINT,EXPLC
      COMMON/VEHICL/DELPL,DVTOT,SPI1,SPI2,MFE1,MFE2
C***************************************************************
C*  READ IN FIRST STAGE MASS FACTORS:
C*    DEL1(1)=FUSELAGE MASS/MASS CONTAINED
C*    DEL1(2)=PROPELLENT MASS/EMPTY MASS
C*    DEL1(3)=MASS OF LIFT COMPONENTS/MASS CARRIED
C*    DEL1(4)=PROPULSION SYSTEM MASS/MASS CARRIED
C*    DEL1(5)=THERMAL PROTECTION SYSTEM MASS/MASS PROTECTED
C*
      READ(5,101)(DEL1(I),I=1,5)
C***************************************************************
C*  READ IN SECOND STAGE MASS FACTORS:
C*    DEL2(I)=SAME AS DEL1(I)
C*
      READ(5,101)(DEL2(I),I=1,5)
C***************************************************************
C*  READ IN COSTING VALUES:
C*    MO=TOTAL LAUNCH MASS OF PROGRAM (MT)
C*    CO=$/KG R&D
C*    C1=$/KG INITIAL PRODUCTION
C*    Y=OPERATIONAL PROGRAM YEARS
C*    FLTPV=FLIGHTS PER VEHICLE
C*
      READ(5,101)MO,CO,C1,Y,FLTPV
      MO=MO*1000.
C***************************************************************
C*  READ IN COSTING AND VEHICLE PARAMETERS:
C*    DELPL=0 FOR EXTERNAL PAYLOAD, 1 FOR INTERNAL
C*    RINT=INTEREST RATE FOR COST DISCOUNTING
C*    DVTOT=TOTAL DELTA-V (M/SEC)
C*    SPI1=SPECIFIC IMPULSE, FIRST STAGE
C*    SPI2=SPECIFIC IMPULSE, SECOND STAGE
C*
      READ(5,101)DELPL,RINT,DVTOT,SPI1,SPI2
C***************************************************************
C*  READ IN COSTING AND VEHICLE PARAMETERS:
C*    MFE1=FIRST STAGE FIXED EQUIPMENT MASS (KG)
C*    MFE2=SECOND STAGE FIXED EQUIPMENT MASS (KG)
C*    EXPLC=EXPONENT OF LEARNING CURVE
C*
      READ(5,102)MFE1,MFE2,EXPLC
C***************************************************************
C*  STARTING ESTIMATIONS
C*
      D1LIM=30.
      D2LIM=100.
```

```
      DIFF2=D2LIM
      DO 5 ILOOP=1,10
      MPL=500*FLOAT(ILOOP*ILOOP)
      DV2=5000.
      DV2DEL=50.
C***************************************************************
C*  PARABOLIC ESTIMATION ITERATION
C*
 1    B(1)=DV2+DV2DEL
      B(2)=DV2
      B(3)=DV2-DV2DEL
      DO 2 I=1,3
      CALL MASCAL(B(I),MPL,DEL1,DEL2,X(I))
 2    WRITE(6,301)B(I),MPL,X(I)
      CALL PARAB(B,X,DV2,DIFF1,DV2DEL)
      IF (DIFF1.GE.D1LIM) GO TO 1
      CALL MASCAL(DV2,MPL,DEL1,DEL2,C)
      DV1=DVTOT-DV2
 5    WRITE(6,201)DV1,DV2,MPL,C,RINT
      STOP
 101  FORMAT(5F8.3)
 102  FORMAT(3F8.3)
 201  FORMAT(' DV1=',F6.0,' DV2=',F6.0,' MPL=',F7.0,' $/KG=',
     +        F6.2,' RINT=',F6.2)
 301  FORMAT(' DV2=',F7.0,' MPL=',F6.0,' $=',F7.2)
      END
      SUBROUTINE PARAB(B,X,VAL,DIFF,DEL)
      DIMENSION B(3),X(3)
      XVAL=X(1)-2.*X(2)+X(3)
      VALNEW=-.5*((-2.*B(2)*XVAL+DEL*(X(1)-X(3)))/XVAL)
      IF (VALNEW.LE.0.) VALNEW=.5*VAL
      DIFF=ABS(VALNEW-VAL)
      VAL=VALNEW
      RETURN
      END
      SUBROUTINE MASCAL(DV2,MPL,DEL1,DEL2,CPERKG)
C***************************************************************
C*  BASED ON VEHICLE PARAMETERS, THIS SUBROUTINE CALCULATES
C*  THE VEHICLE COMPONENT MASSES, INCLUDING ITERATING FOR
C*  THE NONLINEAR PROPELLENT TANK MASS TERMS.
C***************************************************************
      IMPLICIT REAL(M)
      DIMENSION DEL1(5),DEL2(5),M1(5),M2(5),MX1(5),MX2(5),A(5,5),B(5)
      COMMON/VEHICL/DELPLT,DVTOT,SPI1,SPI2,MFE1,MFE2
      DATA G/9.8/
      DELPL=DELPLT
      DV1=DVTOT-DV2
      R1=EXP(-DV1/(G*SPI1))
      R2=EXP(-DV2/(G*SPI2))
      DEL1(2)=(1.-R1)/R1
      DEL2(2)=(1.-R2)/R2
      MX1(1)=0.
      MX1(2)=MFE1
      MX1(3)=0.
      MX2(1)=0.
```

```
      MX2(2)=MFE2
      MX2(3)=MPL
 1    CALL ASETUP (A,DEL2)
      CALL BSETUP(B,MX2,DEL2,DELPL)
      CALL MAXSOL(A,B,M2)
      CALL MTANK(M2,MX2,DIFF)
      IF (ABS(DIFF).GT.10.) GO TO 1
      DO 2 I=1,3
 2    MX1(3)=MX1(3)+M2(I)+MX2(I)
      MX1(3)=MX1(3)+M2(4)+M2(5)
 3    CALL ASETUP(A,DEL1)
      CALL BSETUP(B,MX1,DEL1,0.)
      CALL MAXSOL(A,B,M1)
      CALL MTANK(M1,MX1,DIFF)
      IF (ABS(DIFF).GT.10.) GO TO 3
      MEMPTY=0.
      DO 4 I=1,5
      IF (I.EQ.2) GO TO 4
      MEMPTY=MEMPTY+M1(I)+M2(I)
 4    CONTINUE
      DO 5 I=1,2
 5    MEMPTY=MEMPTY+MX1(I)+MX2(I)
      CPERKG=COST(MEMPTY,MPL)
      RETURN
      END
      SUBROUTINE ASETUP(A,DEL)
C**************************************************************
C*   SETS UP A MATRIX AS PER NOTES 11/18/79
C**************************************************************
      DIMENSION A(5,5),DEL(5)
      DO 2 I=1,5
      DO 1 J=1,5
 1    A(I,J)=-DEL(I)
 2    A(I,I)=1.
      A(1,3)=0.
      A(1,2)=0.
      A(1,5)=0.
      A(3,2)=0.
      A(3,3)=1.-DEL(3)
      A(4,4)=1.-DEL(4)
      A(5,2)=0.
      RETURN
      END
      SUBROUTINE BSETUP(B,MX,DEL,DELPL)
C**************************************************************
C*   SETS UP B MATRIX, AS PER NOTES 11/18/79
C**************************************************************
      IMPLICIT REAL (M)
      DIMENSION B(5),DEL(5),MX(3)
      DO 1 I=1,5
      FACT=DELPL
      IF (I.EQ.2.OR.I.EQ.4) FACT=1.
 1    B(I)=DEL(I)*(MX(1)+MX(2)+FACT*MX(3))
      RETURN
      END
```

```
      SUBROUTINE MTANK(M,MX,DIFF)
C************************************************************
C*  CALCULATES NEW VALUE OF TANK MASS BASED ON PROPELLANT
C*  MASS, THEN FINDS DIFFERENCE BETWEEN NEW AND OLD VALUES
C************************************************************
      IMPLICIT REAL(M)
      DIMENSION M(5),MX(3)
      MT=.2*M(2)**.9
      DIFF=MT-MX(1)
      MX(1)=MT
      RETURN
 301  FORMAT(' ENTERING MTANK:',G14.5)
      END
      FUNCTION COST(ME,MPL)
C************************************************************
C*  CALCULATES LAUNCH COSTS IN $/KG OF PAYLOAD, AS PER NOTES
C************************************************************
      IMPLICIT REAL(M)
      COMMON /COSTS/MO,CO,C1,Y,FLTPV,RINT,EXPLC
      V=MO/(MPL*FLTPV)
      NV=IFIX(V)
      IF (NV.EQ.O) NV=1
      SUM=O.
      EXTRA=O.
      DO 1 I=1,NV
      X=FLOAT(I)
      EXTRA=EXTRA+X**EXPLC*(1.+RINT)**(-X*Y/FLOAT(NV+1))
 1    SUM=SUM+X**EXPLC*(1.+RINT)**(-X*Y/FLOAT(NV))
      X=X+1.
      EXTRA=EXTRA+X**EXPLC*(1.+RINT)**(-X*Y/FLOAT(NV+1))
      AVG=(EXTRA-SUM)*(V-FLOAT(NV))+SUM
      COST=ME/MO*(CO+C1*AVG)
      RETURN
      END
      SUBROUTINE MAXSOL(A,B,X)
C************************************************************
C*  FINDS THE SOLUTION TO A SET OF 5 SIMULTANEOUS EQUATIONS
C*  OF THE FORM <A><X>=<B>.  SOLUTION IS BY PIVOTING.
C************************************************************
      DIMENSION A(5,5),B(5),X(5)
      DO 3 I=1,4
      DIV=A(I,I)
      DO 1 J=1,5
 1    A(I,J)=A(I,J)/DIV
      B(I)=B(I)/DIV
      I1=I+1
      DO 4 K=I1,5
      DIV=A(K,I)
      IF (DIV.EQ.O.) GO TO 4
      B(K)=B(K)/DIV-B(I)
      DO 2 J=1,5
 2    A(K,J)=A(K,J)/DIV-A(I,J)
 4    CONTINUE
 3    CONTINUE
      B(5)=B(5)/A(5,5)
```

```
      A(5,5)=1.
      X(5)=B(5)
      X(4)=B(4)-A(4,5)*X(5)
      X(3)=B(3)-A(3,5)*X(5)-A(3,4)*X(4)
      X(2)=B(2)-A(2,5)*X(5)-A(2,4)*X(4)-A(2,3)*X(3)
      X(1)=B(1)-A(1,5)*X(5)-A(1,4)*X(4)-A(1,3)*X(3)-A(1,2)*X(2)
      RETURN
      END
      SUBROUTINE PRTARY(A)
C***********************************************************
C*  PRINTS OUT AN 5X5 ARRAY
C***********************************************************
      DIMENSION A(5,5)
      DO 1 J=1,5
 1    WRITE(6,101)(A(J,I),I=1,5)
      RETURN
 101  FORMAT(1X,6G10.4)
      END
```

```
C***********************************************************
C*  THIS PROGRAM DESIGNS TWO-STAGE LAUNCH VEHICLES FOR
C*  A VARIETY OF PAYLOAD MASSES TO LEO, USING A PARABOLIC
C*  EXTRAPOLATION TO INTERATE FOR OPTIMUM STAGING VELOCITY
C*  AND EXTERNAL EXPENDIBLE PROPELLANT TANKS FOR BOTH STAGES
C*  D. L. AKIN                                 7/13/81
C**********************************************************
      IMPLICIT REAL(M)
      DIMENSION DEL1(5),DEL2(5),B(3),X(3)
      COMMON/COSTS/MO,CO,C1,Y,FLTPV,RINT,EXPLC
      COMMON/VEHICL/DELPL,DVTOT,SPI1,SPI2,MFE1,MFE2
C**********************************************************
C*  READ IN FIRST STAGE MASS FACTORS:
C*    DEL1(1)=FUSELAGE MASS/MASS CONTAINED
C*    DEL1(2)=PROPELLENT MASS/EMPTY MASS
C*    DEL1(3)=MASS OF LIFT COMPONENTS/MASS CARRIED
C*    DEL1(4)=PROPULSION SYSTEM MASS/MASS CARRIED
C*    DEL1(5)=THERMAL PROTECTION SYSTEM MASS/MASS PROTECTED
C*
      READ(5,101)(DEL1(I),I=1,5)
C***********************************************************
C*  READ IN SECOND STAGE MASS FACTORS:
C*    DEL2(I)=SAME AS DEL1(I)
C*
      READ(5,101)(DEL2(I),I=1,5)
C**********************************************************
C*  READ IN COSTING VALUES:
C*    MO=TOTAL LAUNCH MASS OF PROGRAM (MT)
C*    CO=$/KG R&D
C*    C1=$/KG INITIAL PRODUCTION
C*    Y=OPERATIONAL PROGRAM YEARS
C*    FLTPV=FLIGHTS PER VEHICLE
C*
      READ(5,101)MO,CO,C1,Y,FLTPV
      MO=MO*1000.
C***********************************************************
C*  READ IN COSTING AND VEHICLE PARAMETERS:
C*    DELPL=O FOR EXTERNAL PAYLOAD, 1 FOR INTERNAL
C*    RINT=INTEREST RATE FOR COST DISCOUNTING
C*    DVTOT=TOTAL DELTA-V (M/SEC)
C*    SPI1=SPECIFIC IMPULSE, FIRST STAGE
C*    SPI2=SPECIFIC IMPULSE, SECOND STAGE
C*
      READ(5,101)DELPL,RINT,DVTOT,SPI1,SPI2
C***********************************************************
C*  READ IN COSTING AND VEHICLE PARAMETERS:
C*    MFE1=FIRST STAGE FIXED EQUIPMENT MASS (KG)
C*    MFE2=SECOND STAGE FIXED EQUIPMENT MASS (KG)
C*    EXPLC=EXPONENT OF LEARNING CURVE
C*
      READ(5,102)MFE1,MFE2,EXPLC
C***********************************************************
C*  STARTING ESTIMATIONS
C*
      D1LIM=30.
```

```
      D2LIM=100.
      DIFF2=D2LIM
      DO 5 ILOOP=1,10
      MPL=500*FLOAT(ILOOP*ILOOP)
      DV2=5000.
      DV2DEL=50.
C***********************************************************
C*   PARABOLIC ESTIMATION ITERATION
C*
 1    B(1)=DV2+DV2DEL
      B(2)=DV2
      B(3)=DV2-DV2DEL
      DO 2 I=1,3
      CALL MASCAL(B(I),MPL,DEL1,DEL2,X(I))
 2    WRITE(6,301)B(I),MPL,X(I)
      CALL PARAB(B,X,DV2,DIFF1,DV2DEL)
      IF (DIFF1.GE.D1LIM) GO TO 1
      CALL MASCAL(DV2,MPL,DEL1,DEL2,C)
      DV1=DVTOT-DV2
 5    WRITE(6,201)DV1,DV2,MPL,C,RINT
      STOP
 101  FORMAT(5F8.3)
 102  FORMAT(3F8.3)
 201  FORMAT(' DV1=',F6.0,' DV2=',F6.0,' MPL=',F7.0,' $/KG=',
     +        F6.2,' RINT=',F6.2)
 301  FORMAT(' DV2=',F7.0,' MPL=',F6.0,' $=',F7.2)
      END
      SUBROUTINE PARAB(B,X,VAL,DIFF,DEL)
      DIMENSION B(3),X(3)
      XVAL=X(1)-2.*X(2)+X(3)
      VALNEW=-.5*((-2.*B(2)*XVAL+DEL*(X(1)-X(3)))/XVAL)
      IF (VALNEW.LE.0.) VALNEW=.5*VAL
      DIFF=ABS(VALNEW-VAL)
      VAL=VALNEW
      RETURN
      END
      SUBROUTINE MASCAL(DV2,MPL,DEL1,DEL2,CPERKG)
C***********************************************************
C*   BASED ON VEHICLE PARAMETERS, THIS SUBROUTINE CALCULATES
C*   THE VEHICLE COMPONENT MASSES, INCLUDING ITERATING FOR
C*   THE NONLINEAR PROPELLENT TANK MASS TERMS.
C***********************************************************
      IMPLICIT REAL(M)
      DIMENSION DEL1(5),DEL2(5),M1(5),M2(5),MX1(5),MX2(5),A(5,5),B(5)
      COMMON/VEHICL/DELPLT,DVTOT,SPI1,SPI2,MFE1,MFE2
      DATA G/9.8/
      DELPL=DELPLT
      DV1=DVTOT-DV2
      R1=EXP(-DV1/(G*SPI1))
      R2=EXP(-DV2/(G*SPI2))
      DEL1(2)=(1.-R1)/R1
      DEL2(2)=(1.-R2)/R2
      MX1(1)=0.
      MX1(2)=MFE1
      MX1(3)=0.
```

```
      MX2(1)=0.
      MX2(2)=MFE2
      MX2(3)=MPL
 1    CALL ASETUP (A,DEL2)
      CALL BSETUP(B,MX2,DEL2,DELPL)
      CALL MAXSOL(A,B,M2)
      CALL MTANK(M2,MX2,DIFF)
      IF (ABS(DIFF).GT.10.) GO TO 1
      DO 2 I=1,3
 2    MX1(3)=MX1(3)+M2(I)+MX2(I)
      MX1(3)=MX1(3)+M2(4)+M2(5)
 3    CALL ASETUP(A,DEL1)
      CALL BSETUP(B,MX1,DEL1,0.)
      CALL MAXSOL(A,B,M1)
      CALL MTANK(M1,MX1,DIFF)
      IF (ABS(DIFF).GT.10.) GO TO 3
      MEMPTY=0.
      DO 4 I=1,5
      IF (I.EQ.2) GO TO 4
      MEMPTY=MEMPTY+M1(I)+M2(I)
 4    CONTINUE
      MEMPTY=MEMPTY+MX1(2)+MX2(2)
      MTANKS=MX1(1)+MX2(1)
      CPERKG=COST(MEMPTY,MPL,MTANKS)
      RETURN
      END
      SUBROUTINE ASETUP(A,DEL)
C************************************************************
C*  SETS UP A MATRIX AS PER NOTES 11/18/79
C************************************************************
      DIMENSION A(5,5),DEL(5)
      DO 2 I=1,5
      DO 1 J=1,5
 1    A(I,J)=-DEL(I)
 2    A(I,I)=1.
      A(1,3)=0.
      A(1,2)=0.
      A(1,5)=0.
      A(3,2)=0.
      A(3,3)=1.-DEL(3)
      A(4,4)=1.-DEL(4)
      A(5,2)=0.
      RETURN
      END
      SUBROUTINE BSETUP(B,MX,DEL,DELPL)
C************************************************************
C*  SETS UP B MATRIX, AS PER NOTES 11/18/79
C************************************************************
      IMPLICIT REAL (M)
      DIMENSION B(5),DEL(5),MX(3)
      DO 1 I=1,5
      FACT=DELPL
      FACTOR=0
      IF (I.EQ.2.OR.I.EQ.4) FACTOR=1
      IF (I.EQ.2.OR.I.EQ.4) FACT=1.
```

```
 1    B(I)=DEL(I)*(FACTOR*MX(1)+MX(2)+FACT*MX(3))
      RETURN
      END
      SUBROUTINE MTANK(M,MX,DIFF)
C*************************************************************
C*  CALCULATES NEW VALUE OF TANK MASS BASED ON PROPELLANT
C*  MASS, THEN FINDS DIFFERENCE BETWEEN NEW AND OLD VALUES
C*************************************************************
      IMPLICIT REAL(M)
      DIMENSION M(5),MX(3)
      MT=.2*M(2)**.9
      DIFF=MT-MX(1)
      MX(1)=MT
      RETURN
 301  FORMAT(' ENTERING MTANK:',G14.5)
      END
      FUNCTION COST(ME,MPL,MTANKS)
C*************************************************************
C*  CALCULATES LAUNCH COSTS IN $/KG OF PAYLOAD, AS PER NOTES
C*************************************************************
      IMPLICIT REAL(M)
      COMMON /COSTS/MO,CO,C1,Y,FLTPV,RINT,EXPLC
      V=MO/(MPL*FLTPV)
      NV=IFIX(V)
      IF (NV.EQ.O) NV=1
      IF (NV.EQ.O) NV=1
      SUM=O.
      EXTRA=O.
      DO 1 I=1,NV
      X=FLOAT(I)
      EXTRA=EXTRA+X**EXPLC*(1.+RINT)**(-X*Y/FLOAT(NV+1))
 1    SUM=SUM+X**EXPLC*(1.+RINT)**(-X*Y/FLOAT(NV))
      X=X+1.
      EXTRA=EXTRA+X**EXPLC*(1.+RINT)**(-X*Y/FLOAT(NV+1))
      AVG=(EXTRA-SUM)*(V-FLOAT(NV))+SUM
      COST=ME/MO*(CO+C1*AVG)
      COST=COST+MTANKS/MO*(CO+C1*(MO/MPL)**EXPLC/(1.+EXPLC))
      RETURN
      END
      SUBROUTINE MAXSOL(A,B,X)
C*************************************************************
C*  FINDS THE SOLUTION TO A SET OF 5 SIMULTANEOUS EQUATIONS
C*  OF THE FORM <A><X>=<B>.  SOLUTION IS BY PIVOTING.
C*************************************************************
      DIMENSION A(5,5),B(5),X(5)
      DO 3 I=1,4
      DIV=A(I,I)
      DO 1 J=1,5
 1    A(I,J)=A(I,J)/DIV
      B(I)=B(I)/DIV
      I1=I+1
      DO 4 K=I1,5
      DIV=A(K,I)
      IF (DIV.EQ.O.) GO TO 4
      B(K)=B(K)/DIV-B(I)
```

```
      DO 2 J=1,5                                                        TSE02210
 2    A(K,J)=A(K,J)/DIV-A(I,J)                                          TSE02220
 4    CONTINUE                                                          TSE02230
 3    CONTINUE                                                          TSE02240
      B(5)=B(5)/A(5,5)                                                  TSE02250
      A(5,5)=1.                                                         TSE02260
      X(5)=B(5)                                                         TSE02270
      X(4)=B(4)-A(4,5)*X(5)                                             TSE02280
      X(3)=B(3)-A(3,5)*X(5)-A(3,4)*X(4)                                 TSE02290
      X(2)=B(2)-A(2,5)*X(5)-A(2,4)*X(4)-A(2,3)*X(3)                     TSE02300
      X(1)=B(1)-A(1,5)*X(5)-A(1,4)*X(4)-A(1,3)*X(3)-A(1,2)*X(2)         TSE02310
      RETURN                                                            TSE02320
      END                                                               TSE02330
      SUBROUTINE PRTARY(A)                                              TSE02340
C****************************************************************       TSE02350
C*  PRINTS OUT AN 5X5 ARRAY                                             TSE02360
C****************************************************************       TSE02370
      DIMENSION A(5,5)                                                  TSE02380
      DO 1 J=1,5                                                        TSE02390
 1    WRITE(6,101)(A(J,I),I=1,5)                                        TSE02400
      RETURN                                                            TSE02410
 101  FORMAT(1X,6G10.4)                                                 TSE02420
      END                                                               TSE02430
```

```
PROCEDURE ENTRY;

BEGIN

  WRITELN ('EARTH-MOON LOCATION CODES:');
  WRITELN;
  WRITELN ('     1: EARTH SURFACE');
  WRITELN ('     2: LOW EARTH ORBIT');
  WRITELN ('     3: GEOSYNCHRONOUS ORBIT');
  WRITELN ('     4: HIGH EARTH ORBIT');
  WRITELN ('     5: L-4');
  WRITELN ('     6: L-2');
  WRITELN ('     7: L-1');
  WRITELN ('     8: LUNAR ORBIT');
  WRITELN ('     9: LUNAR SURFACE');
  WRITELN;

  WRITE (' ENTER ORIGIN:');
  READLN (SITE[1]);
  WRITE (' ENTER REFINING SITE:');
  READLN (SITE[2]);
  WRITE (' ENTER MANUFACTURING SITE:');
  READLN (SITE[3]);
  WRITE (' ENTER ASSEMBLY SITE:');
  READLN (SITE[4]);
  WRITE (' ENTER DESTINATION:');
  READLN (SITE[5]);
  WRITELN;

  WRITE (' ENTER DATA FILE NAME:');
  READLN (FILENAME);

END;


PROCEDURE SETARRAY;

VAR I:INTEGER;

BEGIN
  NAMES[1]:='ES';
  NAMES[2]:='LEO';
  NAMES[3]:='GEO';
  NAMES[4]:='HEO';
  NAMES[5]:='L4';
  NAMES[6]:='L2';
  NAMES[7]:='L1';
  NAMES[8]:='LO';
  NAMES[9]:='LS';
```

```
FOR I:= 1 TO 9 DO
  BEGIN
    DELTAVEE[1,I]:=1E6;
    DELTAVEE[9,I]:=1E6;
    DELTAVEE[I,1]:=1E6;
    DELTAVEE[I,9]:=1E6;
  END;

FOR I:= 1 TO 9 DO
  DELTAVEE[I,I]:=0;

DELTAVEE[1,2]:=9000;
DELTAVEE[2,3]:=4198;
DELTAVEE[2,4]:=4262;
DELTAVEE[2,5]:=3922;
DELTAVEE[2,6]:=4216;
DELTAVEE[2,7]:=3853;
DELTAVEE[2,8]:=4017;
DELTAVEE[3,2]:=4610;
DELTAVEE[3,4]:=678;
DELTAVEE[3,5]:=1646;
DELTAVEE[3,6]:=2048;
DELTAVEE[3,7]:=1514;
DELTAVEE[3,8]:=1812;
DELTAVEE[4,2]:=5299;
DELTAVEE[4,3]:=689;
DELTAVEE[4,5]:=1206;
DELTAVEE[4,6]:=2103;
DELTAVEE[4,7]:=1082;
DELTAVEE[4,8]:=1432;
DELTAVEE[5,2]:=8708;
DELTAVEE[5,3]:=4098;
DELTAVEE[5,4]:=3409;
DELTAVEE[5,6]:=1E6;
DELTAVEE[5,7]:=1E6;
DELTAVEE[5,8]:=1E6;
DELTAVEE[6,2]:=8891;
DELTAVEE[6,3]:=4281;
DELTAVEE[6,4]:=3592;
DELTAVEE[6,5]:=2229;
DELTAVEE[6,7]:=1E6;
DELTAVEE[6,8]:=1E6;
DELTAVEE[7,2]:=8554;
DELTAVEE[7,3]:=3944;
DELTAVEE[7,4]:=3255;
DELTAVEE[7,5]:=1892;
DELTAVEE[7,6]:=2075;
DELTAVEE[7,8]:=1E6;
DELTAVEE[8,2]:=9248;
DELTAVEE[8,3]:=4638;
DELTAVEE[8,4]:=3949;
DELTAVEE[8,5]:=2586;
DELTAVEE[8,6]:=2769;
DELTAVEE[8,7]:=2432;
DELTAVEE[8,9]:=1733;

END;
```

```
PROCEDURE READVEC (VAR X:ARRAY[1..4] OF REAL);

BEGIN
  READLN(DISK,X[1],X[2],X[3],X[4]);
END;



PROCEDURE READFILE;

VAR I:INTEGER;
    X:ARRAY[1..4] OF REAL;

BEGIN

   READVEC(X);
   CREWPROD:=X;
   READVEC(X);
   MACHPROD:=X;
   READVEC(X);
   MACHPOW:=X;
   READVEC(X);
   THRUPUT:=X;
   READVEC(X);
   EARTHORG:=X;

   READLN (DISK,NUNITS,MASSUNIT,YRLYRET,LUMPRET);
   READLN (DISK,TSETUP,TPRODUCE,TRUNOUT,DISCRATE);
   READLN (DISK,MCREWMAN,MHOUSE,MENPRHSE,SUPPORT);
   READLN (DISK,ERTHWAGE,SPACWAGE,ECONSUM,LCONSUM);
   READLN (DISK,CRWTRAIN,CREWLIFE,EXPONLC,RANDDCST);

 END;


 PROCEDURE HIENTRY;

 VAR TYPE:CHAR;

 BEGIN

   WRITELN;
   WRITELN (' ENTER HIGH-THRUST PROPULSION TYPE:');
   WRITELN ('      A: LO2/LH2 CHEMICAL');
   WRITELN ('      B: NERVA/LH2');
   WRITELN ('      C: NERVA/LOX');
   WRITELN ('      D: GAS CORE / LH2');
   WRITELN ('      E: GAS CORE / LOX');
   WRITELN ('      F: FUSION');
   WRITE (' CHOICE:');
   READLN (TYPE);
```

```
CASE TYPE OF

  'A':BEGIN
        ISPHITHR:=450;
        EFRACHI:=0.14;
        MPROPHI:=500;
        EPSHITHR:=0.03;
      END;

  'B':BEGIN
        ISPHITHR:=825;
        EFRACHI:=1;
        MPROPHI:=80000;
        EPSHITHR:=0.121;
      END;

  'C':BEGIN
        ISPHITHR:=200;
        EFRACHI:=0;
        MPROPHI:=8000;
        EPSHITHR:=0.015;
      END;

  'D':BEGIN
        ISPHITHR:=4000;
        EFRACHI:=1;
        MPROPHI:=250000;
        EPSHITHR:=0.121;
      END;

  'E':BEGIN
        ISPHITHR:=1000;
        EFRACHI:=0;
        MPROPHI:=250000;
        EPSHITHR:=0.015;
      END;

  'F':BEGIN
        ISPHITHR:=3000;
        EFRACHI:=1;
        MPROPHI:=8000;
        EPSHITHR:=0.121;
      END;

END;
```

```
PROCEDURE LOTHRUST;

BEGIN

  WRITELN ('ENTER LOW THRUST PROPULSION SYSTEM:');
  WRITELN ('     G: ION/ARGON');
  WRITELN ('     H: ION/LH2');
  WRITELN ('     I: MPD       ');
  WRITELN ('     J: MORE      ');
  WRITE ('CHOICE: ');
  READLN (LOTYPE);

  CASE LOTYPE OF

    'G': BEGIN
           ENGEFF:=0.65;
           LENGFACT:=0.0;
           MDOTFACT:=1.67E6;
           MDOTLO:=3E-5;
           ISPLO:=8000;
           EPSLO:=0.09;
           EFRACLO:=1;
         END;

    'H': BEGIN
           ENGEFF:=0.65;
           LENGFACT:=0;
           MDOTFACT:=1.67E6;
           MDOTLO:=3E-5;
           ISPLO:=5000;
           EPSLO:=.015;
           EFRACLO:=0;
         END;

    'I': BEGIN
           ENGEFF:=0.57;
           LENGFACT:=0;
           MDOTFACT:=4.1E3;
           MDOTLO:=0.006;
           ISPLO:=5000;
           EPSLO:=0.09;
           EFRACLO:=1;
         END;

    'J': BEGIN
           ENGEFF:=0.95;
           LENGFACT:=0.0014;
           MDOTFACT:=0.0;
           MDOTLO:=0.07;
           ISPLO:=300;
           EPSLO:=0.015;
           EFRACLO:=0.0;
         END;

  END;
```

```
WRITELN;
WRITELN ('ENTER LOW THRUST POWER SYSTEM:');
WRITELN ('     K: SOLAR PHOTOVOLTAIC');
WRITELN ('     L: SOLAR THERMIONIC');
WRITELN ('     M: NUCLEAR');
WRITE ('CHOICE: ');
READLN (POWTYPE);

CASE POWTYPE OF

   'K': BEGIN
          SPECPOW:=10;
          POWCOST:=50000;
        END;

   'L': BEGIN
          SPECPOW:=40;
          POWCOST:=4000;
        END;

   'M': BEGIN
          SPECPOW:=15;
          POWCOST:=7500;
        END;

 END;

 WRITELN;
 WRITE ('IS A LUNAR MASS DRIVER USED?');
 READLN (TYPE);
 IF TYPE='Y' THEN MASSDRIV:=TRUE
             ELSE MASSDRIV:=FALSE;

END;



PROCEDURE SITESET;

  VAR I:INTEGER;

  BEGIN

    CREWROT[1]:=0;
    CREWROT[2]:=2;
    CREWROT[3]:=6;
    CREWROT[4]:=4;
    CREWROT[5]:=4;
    CREWROT[6]:=4;
    CREWROT[7]:=4;
    CREWROT[8]:=4;
    CREWROT[9]:=2;
```

```
 LIGHTIME[1]:=0.5;
 LIGHTIME[2]:=0.7;
 FOR I:=3 TO 7 DO LIGHTIME[I]:=1;
 LIGHTIME[8]:=0.7;
 LIGHTIME[9]:=0.5;

 LIFETIME[1]:=50;
 LIFETIME[2]:=20;
 LIFETIME[3]:=5;
 FOR I:=4 TO 8 DO LIFETIME[I]:=20;
 LIFETIME[9]:=30;

END;



PROCEDURE PROCSET;

BEGIN

   CREWPROD[1]:=800;
   CREWPROD[2]:=100;
   CREWPROD[3]:=175;
   CREWPROD[4]:=500;

   MACHPROD[1]:=0.67;
   MACHPROD[2]:=0.0165;
   MACHPROD[3]:=1.32E-3;
   MACHPROD[4]:=1;

   MACHPOW[1]:=80;
   MACHPOW[2]:=120;
   MACHPOW[3]:=0.05;
   MACHPOW[4]:=1;

   THRUPUT[1]:=1;
   THRUPUT[2]:=0.12;
   THRUPUT[3]:=0.8;
   THRUPUT[4]:=1;

   EARTHORG[1]:=0;
   EARTHORG[2]:=0;
   EARTHORG[3]:=0.05;
   EARTHORG[4]:=0.05;

 END;
```

```
PROCEDURE SITECOST;

CONST HOURYEAR=8766;
      DAYYEAR=365.25;

VAR I:INTEGER;

BEGIN

  FOR I:=4 DOWNTO 1 DO

  BEGIN

  IF I:=4 THEN OUTPUT[4]:=NUNITS+MASSUNIT
          ELSE OUTPUT[I]:=LINPUT[I+1];

  LINPUT[I]:=OUTPUT[I]*(1-EARTHORG[I])/THRUPUT[I];

  EINPUT[I]:=OUTPUT[I]*EARTHORG[I];

  WORKYEAR[I]:=HOURYEAR*LIGHTIME[I];

  PRODCREW[I]:=(LINPUT[I]*EINPUT[I])/(CREWPROD[I]*WORKYEAR[I]);

  TOTCREW[I]:=SUPPORT*PRODCREW[I];

  MCONSUM[I]:=TOTCREW[I]*USAGE*DAYYEAR;

  MROTATE[I]:=TOTCREW[I]*CREWROT[SITE[I]]*MCREWMAN;

  MHI[I]:=MROTATE[I]+MCONSUM[I];

  MLOOUT[I]:=EINPUT[I];

  PRODPOW[I]:=OUTPUT[I]*MACHPOW[I]*WORKYEAR[I];

  CREWPOW[I]:=TOTCREW[I]*LIFEPOW;

  TOTPOW[I]:=PRODPOW[I]+CREWPOW[I];

  PRODMASS[I]:=MACHPROD[I]*OUTPUT[I]*WORKYEAR[I];

  SUPMASS[I]:=TOTCREW[I]*MPERCREW;

  SITEMASS[I]:=PRODMASS[I]+SUPMASS[I];

 END;

END;
```

```
PROCEDURE SETDVEES;

BEGIN

  FOR I:=1 TO 4 DO

    BEGIN

      IF SITE[I]>SITE[I+1] THEN
        DVNEXT[I]:=DELTAVEE[SITE[I],SITE[I+1]];
        ELSE DVNEXT:=DELTAVEE[SITE[I+1],SITE[I]];

      LAGRANGE:=SITE[I+1]=7 OR SITE[I+1]=8;

      IF SITE[I]<>1 THEN
        BEGIN
          DVHI[I]:=DELTAVEE[2,SITE[I]];
          DVLO[I]:=DELTAVEE[SITE[I],2];
        END;
        ELSE BEGIN
          DVHI[I]:=0;
          DVLO[I]:=0;
        END;

      RHI[I]:=EXP(-DVHI[I]/(G*ISPHI));
      RLO[I]:=EXP(-DVLO[I]/(G*ISPLO));
      RNEXT[I]:=EXP(-DVNEXT[I]/(G*ISPLO));

    END;

    DVNEXT[I]:=DELTAVEE[SITE[2],8]:
    RNEXT[I]:=EXP(-DVNEXT[I]/(G+ISPHI));

  END;
```

```
FUNCTION MPLFACT(R,E:REAL):REAL;
  BEGIN
    MPLFACT:=(1-R)*(1-E)/(R*(1-E)-E*(1-R));
  END;


PROCEDURE TRANSPRT;
  BEGIN
    FOR I:=1 TO 4 DO
      BEGIN
        MPRHI[I]:=(MPLHI+MPROPHI)*MPLFACT(RHI[I],EPSHI);
        MPRLOOUT[I]:=(MPLLO+MPROPLO)*MPLFACT(RLO[I],EPSLO);
        MPRLOBCK[I]:=(MPLLO*MBACKLO[I]/MOUTLO[I]+MPROPLO)
                     *MFACT(RLO[I],EPSLO);
        MPRNEXT[I]:=(MPLNEXT[I]+MPROPLO)*MPLFACT(RNEXT[I],EPSLO);
        MPRBACK[I]:=(MPLNEXT[I]*MNEXT[I]/MBACK[I]+MPROPLO)
                    *MPLFACT(RNEXT[I],EPSLO);
      END;
    END;
```

```
C************************************************************
C*  DIAGONAL ASCENT LINEAR PROGRAMMING OPTIMIZATION  (DALPO)
C*
C*  MAIN PROGRAM: OPTIMIZES UP TO FIVE SYSTEMS, WITH A 30 YEAR
C*                PROGRAM AND 30 YEAR RUNOUT ON UNITS PRODUCED
C*
C*  USES SUBROUTINES: ENTRY   EQUATE  COMPAR  REPLAC  ZERO
C*                    LP1     LP2     LP3     LP4     LP5
C*                    OPT1    OPT2    OPT3    OPT4    OPT5
C*                    SCAN1   SCAN2   SCAN3   SCAN4   FNDMIN
C*                    ISGN
C*
C*  D. L. AKIN                  4/3/81
C************************************************************
      DIMENSION IPTVEC(5),IPTTRY(5),M2(2),M3(3),M4(4),P(4,5),YLIM(30),
     +          DF(30)
      COMMON/COUNTS/IC
      COMMON/PARAMS/P,YLIM,LIMU,IFLAG,DF
      CALL ENTRY(N)
      CALL FNDMIN(N)
      IC=0
      OPT=0
      CALL ZERO (IPTVEC)
      GO TO (5,4,3,2,1),N
 1    CALL OPT5(OPTA,IPTTRY)
      CALL COMPAR(OPT,IPTVEC,OPTA,IPTTRY)
      WRITE(6,301)OPT,(IPTVEC(I),I=1,5)
 2    CALL SCAN4(N,OPTA,IPTTRY)
      CALL COMPAR(OPT,IPTVEC,OPTA,IPTTRY)
      WRITE(6,301)OPT,(IPTVEC(I),I=1,5)
 3    CALL SCAN3(N,OPTA,IPTTRY)
      CALL COMPAR(OPT,IPTVEC,OPTA,IPTTRY)
      WRITE(6,301)OPT,(IPTVEC(I),I=1,5)
 4    CALL SCAN2(N,OPTA,IPTTRY)
      CALL COMPAR(OPT,IPTVEC,OPTA,IPTTRY)
      WRITE(6,301)OPT,(IPTVEC(I),I=1,5)
 5    CALL SCAN1(N,OPTA,IPTTRY)
      CALL COMPAR(OPT,IPTVEC,OPTA,IPTTRY)
      WRITE(6,301)OPT,(IPTVEC(I),I=1,5)
C
C  OPTIMUM HAS BEEN FOUND: VALUE IS 'OPT', INITIAL YEARS IN 'IPTVEC'
C
      NSOL=5
      DO 6 I=1,5
 6    IF (IPTVEC(I).EQ.0) NSOL=NSOL-1
C
C  NSOL = NUMBER OF SYSTEMS IN OVERALL OPTIMUM
C
      IFLAG=2
      IF (NSOL.EQ.5) CALL LP5(IPTVEC,OPT)
      IF (NSOL.EQ.5) GO TO 16
      ISYS=0
      DO 11 I=1,5
      IF (IPTVEC(I).EQ.0) GO TO 11
      ISYS=ISYS+1
```

```
      IF (ISYS.EQ.1) IS=I
      IF (ISYS.EQ.2) JS=I
      IF (ISYS.EQ.3) KS=I
      IF (ISYS.EQ.4) LS=I
      GO TO (7,8,9,10),NSOL
 7    M1=IPTVEC(I)
      WRITE(6,304)M1
      GO TO 11
 8    M2(ISYS)=IPTVEC(I)
      GO TO 11
 9    M3(ISYS)=IPTVEC(I)
      GO TO 11
 10   M4(ISYS)=IPTVEC(I)
 11   CONTINUE
      GO TO (12,13,14,15),NSOL
 12   WRITE(6,302)IS,M1
      CALL LP1(IS,M1,OPT)
      GO TO 16
 13   WRITE(6,303)IS,JS,(M2(I),I=1,2)
      CALL LP2(IS,JS,M2,OPT)
      GO TO 16
 14   CALL LP3(IS,JS,KS,M3,OPT)
      GO TO 16
 15   CALL LP4(IS,JS,KS,LS,M4,OPT)
 16   WRITE(6,101)IPTVEC,OPT,IC
      STOP
 101  FORMAT(///20X,'OVERALL OPTIMUM SOLUTION:',
     +       //' INITIAL OPERATIONAL YEARS:',5I6,
     +       / ' MAXIMUM VALUE OF OBJECTIVE FUNCTION:',F10.3,
     +       / ' TOTAL LP SOLUTIONS PERFORMED:',I6)
 301  FORMAT(10X,'OPTIMUM:',F10.3,' VECTOR:',5I4)
 302  FORMAT(10X,'SYSTEM:',I2,' YEAR:',I3)
 303  FORMAT(10X,'SYSTEMS:',2I2,' YEARS:',2I3)
 304  FORMAT(10X,'M1=',I5)
      END
      SUBROUTINE ENTRY (NS)
C************************************************************
C*  THIS SUBROUTINE PROMPTS AT THE TERMINAL (LOGICAL UNIT
C*  NUMBER=5), AND READS IN THE INTERNAL PARAMETERS FOR
C*  DEFINING THE LP OPTIMIZATION.
C*
C*  14-MAR-81
C************************************************************
      DIMENSION P(4,5),DF(30),YLIM(30)
      COMMON/PARAMS/P,YLIM,LIMU,IFLAG,DF
      R=.1
      BR=(1.-(1.+R)**(-30))/R
      DO 10 I=1,3
      DO 10 J=1,5
 10   P(I,J)=0.
      IFLAG=0
      WRITE(5,101)
      READ(5,201)NS
      DO 1 I=1,NS
      WRITE(5,102)I
```

```
      READ(5,202)P(1,I)
      WRITE(5,103)I
      READ(5,202)P(2,I)
      WRITE(5,104)I
      READ(5,202)P(3,I)
 1    P(4,I)=BR*P(2,I)-P(3,I)
      DO 9 I=1,30
 9    DF(I)=(1.+R)**(-I)
      WRITE(5,105)
      READ(5,201)NY
      GOTO(2,4,6),NY
C
C  CONSTANT YEARLY FUNDING
C
 2    WRITE(5,106)
      READ(5,202)Y
      DO 3 I=1,30
 3    YLIM(I)=Y
      GOTO 8
C
C  LINEAR YEARLY FUNDING
C
 4    WRITE(5,107)
      READ(5,202)Y1
      WRITE(5,108)
      READ(5,202)Y30
      YDIFF=(Y30-Y1)/29.
      DO 5 I=1,30
      YLIM(I)=Y1+YDIFF*FLOAT(I-1)
 5    IF (YLIM(I).LT.0.) YLIM(I)=0.
      GOTO 8
C
C  ARBITRARY YEARLY FUNDING
C
 6    DO 7 I=1,30
      WRITE(5,109)I
 7    READ(5,202)YLIM(I)
 8    WRITE(5,110)
      READ(5,203)LIMU
      WRITE(5,111)
      READ(5,201)IFLAG
      RETURN
 101  FORMAT('                  30-YEAR DALP OPTIMIZATION'//
     +       ' ENTER NUMBER OF SYSTEMS: ')
 102  FORMAT(' ENTER R&D COST FOR SYSTEM ',I1,': ')
 103  FORMAT(' ENTER UNIT YEARLY BENEFIT FOR SYSTEM ',I1,': ')
 104  FORMAT(' ENTER UNIT COST FOR SYSTEM ',I1,': ')
 105  FORMAT(' ENTER FUNDING LIMIT:  1) CONSTANT',
     +                               ' 2) LINEAR',
     +                               ' 3) ARBITRARY  :')
 106  FORMAT(' ENTER YEARLY SPENDING RATE: ')
 107  FORMAT(' ENTER SPENDING RATE FOR YEAR 1: ')
 108  FORMAT(' ENTER SPENDING RATE FOR YEAR 30: ')
 109  FORMAT(' ENTER SPENDING RATE FOR YEAR ',I2,': ')
 110  FORMAT(' ENTER MAXIMUM NUMBER OF UNITS: ')
```

```
 111  FORMAT(' ENTER 1 FOR DIAGNOSTICS, O FOR NONE: ')
 201  FORMAT(I1)
 202  FORMAT(F10.0)
 203  FORMAT(I3)
      END
      SUBROUTINE SCAN1(N,OPT,IPTVEC)
C************************************************************
C*  THIS SUBROUTINE SEARCHES THROUGH A SCENARIO OF N POSSIBLE
C*  SYSTEMS TO FIND THE OPTIMUM CHOICE, ASSUMING THAT ONLY
C*  1 SYSTEM IS CHOSEN FOR USE. RETURNED VALUES ARE THE
C*  VALUE OF THE OBJECTIVE FUNCTION (OPT) AND OPTIMUM VALUES
C*  OF Y(I) (IPTVEC), WHERE AN UNCHOSEN SYSTEM HAS Y(I)=0
C*
C*   14-MAR-81
C************************************************************
      DIMENSION IPTVEC(5),IPTEMP(5)
      CALL ZERO(IPTVEC)
      OPT=0.
      DO 1 I=1,N
      CALL OPT1(I,OPTI,IPTEMP)
  1   CALL COMPAR(OPT,IPTVEC,OPTI,IPTEMP)
      WRITE(6,301)OPT,(IPTVEC(I),I=1,5)
      RETURN
 301  FORMAT(' SCAN1 - OPT:',F10.3,' VECTOR:',5I3)
      END
      SUBROUTINE SCAN2(N,OPT,IPTVEC)
C************************************************************
C*  THIS SUBROUTINE SEARCHES THROUGH A SCENARIO OF N POSSIBLE
C*  SYSTEMS TO FIND THE OPTIMUM CHOICE, ASSUMING THAT ONLY
C*  2 SYSTEMS ARE CHOSEN FOR USE. RETURNED VALUES ARE AS IN
C*  SUBROUTINE SCAN1.
C*
C*   14-MAR-81
C************************************************************
      DIMENSION IPTVEC(5),IPTEMP(5)
      CALL ZERO(IPTVEC)
      OPT=0.
      N1=N-1
      DO 1 I=1,N1
      I1=I+1
      DO 1 J=I1,N
      CALL OPT2(I,J,OPTI,IPTEMP)
  1   CALL COMPAR(OPT,IPTVEC,OPTI,IPTEMP)
      RETURN
      END
      SUBROUTINE SCAN3(N,OPT,IPTVEC)
C************************************************************
C*  THIS SUBROUTINE SEARCHES THROUGH A SCENARIO OF N POSSIBLE
C*  SYSTEMS TO FIND THE OPTIMUM CHOICE, ASSUMING THAT ONLY
C*  3 SYSTEMS ARE CHOSEN FOR USE. RETURNED VALUES ARE AS IN
C*  SCAN3.
C*
C*   14-MAR-81
C************************************************************
      DIMENSION IPTVEC(5),IPTEMP(5)
```

```
      CALL ZERO(IPTVEC)
      OPT=0.
      N2=N-2
      N1=N-1
      DO 1 I=1,N2
      I1=I+1
      DO 1 J=I1,N1
      J1=J+1
      DO 1 K=J1,N
      CALL OPT3(I,J,K,OPTI,IPTEMP)
 1    CALL COMPAR(OPT,IPTVEC,OPTI,IPTEMP)
      RETURN
      END
      SUBROUTINE SCAN4(N,OPT,IPTVEC)
C***********************************************************
C*  THIS SUBROUTINE SEARCHES THROUGH A SCENARIO OF N POSSIBLE
C*  SYSTEMS TO FIND THE OPTIMUM CHOICE, ASSUMING THAT ONLY
C*  4 SYSTEMS ARE CHOSEN FOR USE. RETURNED VALUES ARE
C* THE SAME AS THOSE IN SCAN1.
C*
C*  14-MAR-81
C***********************************************************
      DIMENSION IPTVEC(5),IPTEMP(5)
      CALL ZERO(IPTVEC)
      OPT=0.
      N3=N-3
      N2=N-2
      N1=N-1
      DO 1 I=1,N3
      I1=I+1
      DO 1 J=I1,N2
      J1=J+1
      DO 1 K=J1,N1
      K1=K+1
      DO 1 L=K1,N
      CALL OPT4(I,J,K,L,OPTI,IPTEMP)
 1    CALL COMPAR(OPT,IPTVEC,OPTI,IPTEMP)
      RETURN
      END
      SUBROUTINE OPT1(IS,OPT,IPTVEC)
      DIMENSION IPTVEC(5),P(4,5),DF(30),YLIM(30),MIN(5),MSTART(5)
      COMMON/PARAMS/P,YLIM,LIMU,IFLAG,DF
      COMMON/MINIMS/MIN,MSTART
      CALL ZERO(IPTVEC)
      M=MSTART(IS)
      CALL LP1(IS,M,OPTA)
      CALL LP1(IS,M+1,OPTB)
      MASC=ISGN(OPTB-OPTA)
 2    CALL LP1(IS,M+MASC,OPTB)
      IF (OPTB.LT.OPTA) GO TO 3
      OPTA=OPTB
      M=M+MASC
      GO TO 2
 3    OPT=OPTA
      IPTVEC(IS)=M
```

```
      WRITE(6,301)OPT,(IPTVEC(I),I=1,5)
      RETURN
 301  FORMAT(' OPT1   - OPT:',F10.3,' VECTOR:',5I3)
      END
      SUBROUTINE OPT2(IS,JS,OPT,IPTVEC)
      DIMENSION P(4,5),DF(30),YLIM(30),M(2),MASC(2),MIN(5),MWORK(2),
     +          MLAST(2),LMIN(2),MSTART(5),MASC2(2),IPTVEC(5)
      COMMON/PARAMS/P,YLIM,LIMU,IFLAG,DF
      COMMON/MINIMS/MIN,MSTART
      CALL ZERO(IPTVEC)
      IL=0
      M(1)=MSTART(IS)
      M(2)=MSTART(JS)
      LMIN(1)=MIN(IS)
      LMIN(2)=MIN(JS)
      DO 1 I=1,2
 1    MASC(I)=-1
 2    CALL LP2(IS,JS,M,OPTA)
      MCHECK=0
      DO 3 I=1,2
      CALL EQUATE(MWORK,M,MASC,2,I)
      CALL LP2(IS,JS,MWORK,OPTB)
 3    MASC2(I)=ISGN(OPTB-OPTA)*MASC(I)
      DO 5 I=1,2
      MLAST(I)=M(I)
      IF (MASC(I).EQ.MASC2(I)) M(I)=M(I)+MASC2(I)
      IF (M(I).LT.LMIN(I)) M(I)=LMIN(I)
      IF (M(I).GT.30) M(I)=30
      MCHECK=MCHECK+IABS(M(I)-MLAST(I))
 5    MASC(I)=MASC2(I)
      IL=IL+1
      IF (IL.LT.2) GO TO 2
      IF (MCHECK.NE.0) GO TO 2
 6    OPT=OPTA
      IPTVEC(IS)=M(1)
      IPTVEC(JS)=M(2)
      RETURN
      END
      SUBROUTINE OPT3(IS,JS,KS,OPT,IPTVEC)
      DIMENSION P(4,5),DF(30),YLIM(30),M(3),MASC(3),MIN(5),MWORK(3),
     +          MLAST(3),LMIN(3),MSTART(5),MASC2(3),IPTVEC(5)
      COMMON/PARAMS/P,YLIM,LIMU,IFLAG,DF
      COMMON/MINIMS/MIN,MSTART
      CALL ZERO(IPTVEC)
      IL=0
      M(1)=MSTART(IS)
      M(2)=MSTART(JS)
      M(3)=MSTART(KS)
      LMIN(1)=MIN(IS)
      LMIN(2)=MIN(JS)
      LMIN(3)=MIN(KS)
      DO 1 I=1,3
 1    MASC(I)=-1
 2    CALL LP3(IS,JS,KS,M,OPTA)
      IF (OPTA.NE.0.) GO TO 10
```

```
      WRITE(5,401)(M(I),II=1,3)
      READ(5,402)(M(II),II=1,3)
      GO TO 2
 10   MCHECK=0
      DO 3 I=1,3
      CALL EQUATE(MWORK,M,MASC,3,I)
      CALL LP3(IS,JS,KS,MWORK,OPTB)
 3    MASC2(I)=ISGN(OPTB-OPTA)*MASC(I)
      DO 5 I=1,3
      MLAST(I)=M(I)
      IF (MASC(I).EQ.MASC2(I)) M(I)=M(I)+MASC2(I)
      IF (M(I).LT.LMIN(I)) M(I)=LMIN(I)
      IF (M(I).GT.30) M(I)=30
      MCHECK=MCHECK+IABS(M(I)-MLAST(I))
 5    MASC(I)=MASC2(I)
      IL=IL+1
      IF (IL.LT.3) GO TO 2
      IF (MCHECK.NE.0) GO TO 2
 6    OPT=OPTA
      IPTVEC(IS)=M(1)
      IPTVEC(JS)=M(2)
      IPTVEC(KS)=M(3)
      RETURN
 401  FORMAT(' INFEASIBLE SOLUTION - LP3 : ',3I3)
 402  FORMAT(I2,1X,I2,1X,I2)
      END
      SUBROUTINE OPT4(IS,JS,KS,LS,OPT,IPTVEC)
      DIMENSION P(4,5),DF(30),YLIM(30),M(4),MASC(4),MIN(5),MWORK(4),
     +          MLAST(4),LMIN(4),MSTART(5),MASC2(4),IPTVEC(5)
      COMMON/PARAMS/P,YLIM,LIMU,IFLAG,DF
      COMMON/MINIMS/MIN,MSTART
      CALL ZERO(IPTVEC)
      IL=0
      M(1)=(MSTART(IS)+30)/2
      M(2)=(MSTART(JS)+30)/2
      M(3)=(MSTART(KS)+30)/2
      M(4)=(MSTART(LS)+30)/2
      LMIN(1)=MIN(IS)
      LMIN(2)=MIN(JS)
      LMIN(3)=MIN(KS)
      LMIN(4)=MIN(LS)
      DO 1 I=1,4
 1    MASC(I)=-1
 2    CALL LP4(IS,JS,KS,LS,M,OPTA)
      IF (OPTA.NE.0) GO TO 10
      WRITE(5,401)(M(II),II=1,4)
      READ(5,402)(M(II),II=1,4)
      GO TO 2
 10   MCHECK=0
      DO 3 I=1,4
      CALL EQUATE(MWORK,M,MASC,4,I)
      CALL LP4(IS,JS,KS,LS,MWORK,OPTB)
 3    MASC2(I)=ISGN(OPTB-OPTA)*MASC(I)
      DO 5 I=1,4
      MLAST(I)=M(I)
```

```
      IF (MASC(I).EQ.MASC2(I)) M(I)=M(I)+MASC2(I)
      IF (M(I).LT.LMIN(I)) M(I)=LMIN(I)
      IF (M(I).GT.30) M(I)=30
      MCHECK=MCHECK+IABS(M(I)-MLAST(I))
 5    MASC(I)=MASC2(I)
      IL=IL+1
      IF (IL.LT.3) GO TO 2
      IF (MCHECK.NE.0) GO TO 2
 6    OPT=OPTA
      IPTVEC(IS)=M(1)
      IPTVEC(JS)=M(2)
      IPTVEC(KS)=M(3)
      IPTVEC(LS)=M(4)
      RETURN
 401  FORMAT(' INFEASIBLE SOLUTION - LP4 :',4I4)
 402  FORMAT(I2,1X,I2,1X,I2)
      END
      SUBROUTINE OPT5(OPT,IPTVEC)
      DIMENSION P(4,5),DF(30),YLIM(30),M(5),MASC(5),MIN(5),MWORK(5),
     +          MLAST(5),MSTART(5),MASC2(5),IPTVEC(5)
      COMMON/PARAMS/P,YLIM,LIMU,IFLAG,DF
      COMMON/MINIMS/MIN,MSTART
      CALL ZERO(IPTVEC)
      IL=0
      DO 1 I=1,5
      M(I)=(MSTART(I)+30)/2
 1    MASC(I)=-1
 2    CALL LP5(M,OPTA)
      MCHECK=0
      DO 3 I=1,5
      CALL EQUATE(MWORK,M,MASC,5,I)
      CALL LP5(MWORK,OPTB)
 3    MASC2(I)=ISGN(OPTB-OPTA)*MASC(I)
      DO 5 I=1,5
      MLAST(I)=M(I)
      IF (MASC(I).EQ.MASC2(I)) M(I)=M(I)+MASC2(I)
      IF (M(I).LT.MIN(I)) M(I)=MIN(I)
      IF (M(I).GT.30) M(I)=30
      MCHECK=MCHECK+IABS(M(I)-MLAST(I))
 5    MASC(I)=MASC2(I)
      IL=IL+1
      IF (IL.LT.3) GO TO 2
      IF (MCHECK.NE.0) GO TO 2
 6    OPT=OPTA
      DO 7 I=1,5
 7    IPTVEC(I)=M(I)
      RETURN
      END
      SUBROUTINE EQUATE(A,B,C,I,J)
C***********************************************************
C*  THIS SUBROUTINE FORMS A MATRIX A FROM MATRICES B AND C
C*  (ALL SIZE=I). A IS B PLUS THE J(TH) ELEMENT OF C, UNLESS
C*  J IS 0, WHEN A=B+C.
C*
C*  16-MAR-81
```

```
C***********************************************************
      IMPLICIT INTEGER(A-C)
      DIMENSION A(I),B(I),C(I)
      DO 1 K=1,I
 1    A(K)=B(K)
      IF (J.EQ.O) GO TO 2
      A(J)=A(J)+C(J)
      RETURN
 2    DO 3 K=1,I
 3    A(K)=A(K)+C(K)
      RETURN
      END
      SUBROUTINE COMPAR(A,IVEC,B,JVEC)
C***********************************************************
C*  THIS SUBROUTINE COMPARES THE VALUES OF A AND B. IF A<B,
C*  A IS SET EQUAL TO B, AND IVEC IS SET EQUAL TO JVEC.
C*
C*  14-MAR-81
C***********************************************************
      DIMENSION IVEC(5), JVEC(5)
      IF (A.GE.B) RETURN
      A=B
      CALL REPLAC(IVEC,JVEC)
      RETURN
      END
      SUBROUTINE REPLAC(IVEC,JVEC)
C***********************************************************
C*  THIS SUBROUTINE SETS IVEC EQUAL TO JVEC, AND RETURNS
C*
C*  14-MAR-81
C***********************************************************
      DIMENSION IVEC(5),JVEC(5)
      DO 1 I=1,5
 1    IVEC(I)=JVEC(I)
      RETURN
      END
      SUBROUTINE ZERO(A)
C***********************************************************
C*  ZEROS OUT THE ELEMENTS OF ARRAY A (SIZE=5), SINCE THE
C*  STUPID IBM370 IS TOO DUMB TO UNDERSTAND A DATA STATEMENT
C*
C*  15-MAR-81
C***********************************************************
      DIMENSION A(5)
      DO 1 I=1,5
 1    A(I)=O.
      RETURN
      END
      SUBROUTINE FNDMIN(NS)
C***********************************************************
C*  THIS SUBROUTINE TAKES THE R&D COSTS FROM ARRAY P IN COMMON
C*  AREA PARAMS, AND CALCULATES MINIMUM VALUES OF Y(I) (MIN)
C*  AND AVERAGE VALUES OF Y(I) FOR INITIALIZATION (MSTART)
C*
C*  16-MAR-81
```

```
C*************************************************************
      DIMENSION MIN(5),MSTART(5),I(5),P(4,5),DF(30),YLIM(30)
      COMMON/PARAMS/P,YLIM,LIMU,IFLAG,DF
      COMMON/MINIMS/MIN,MSTART
      DO 1 J=1,5
 1    I(J)=0
      ICOUNT=0
      J=0
      X=0.
 2    J=J+1
      X=X+YLIM(J)
      DO 3 K=1,5
      IF (X.LT.P(1,K).OR.I(K).NE.0) GO TO 3
      ICOUNT=ICOUNT+1
      I(K)=1
      MSTART(K)=29
      MIN(K)=J
 3    CONTINUE
      IF (ICOUNT.LT.5) GO TO 2
      RETURN
      END
      FUNCTION ISGN(X)
      IF (X) 1,1,3
 1    ISGN=-1
      RETURN
 2    ISGN=0
      RETURN
 3    ISGN=1
      RETURN
      END
      SUBROUTINE LP1(IS,M,OPT)
      DIMENSION TAB(34,30),CON(34),OBJF(30),PSOL(32),DSOL(34),
     +          RW(1255),IW(99),P(4,5),YLIM(30),DF(30)
      COMMON/PARAMS/P,YLIM,LIMU,IFLAG,DF
      COMMON/COUNTS/IC
      IA=34
      N=30
      M1=31
      M2=1
      IC=IC+1
C
C  INITIALIZE TABLEAU
C
      DO 1 I=1,34
      DO 1 J=1,30
 1    TAB(I,J)=0.
C
C  BUDGETARY CONSTRAINTS
C
      DO 6 J=1,30
      CON(J)=YLIM(J)
      IF (J-M) 2,5,5
 2    TAB(J,J)=P(1,IS)
      GO TO 6
 3    K1=M
```

```
      K2=J-1
      DO 4 K=K1,K2
 4    TAB(J,K)=-P(2,IS)
 5    TAB(J,J)=P(3,IS)
 6    CONTINUE
C
C  DEMAND FOR NEW UNITS BOUNDED
C
      DO 7 J=1,30
 7    IF (J.GE.M) TAB(31,J)=1.
      CON(31)=FLOAT(LIMU)
C
C  CONSTRAINT TO PAY R&D
C
      CON(32)=1.
      DO 8 J=1,30
 8    IF (J.LT.M) TAB(32,J)=1.
C
C  OBJECTIVE FUNCTION
C
      DO 10 J=1,30
      IF (J.GE.M) GO TO 9
      OBJF(J)=-P(1,IS)*DF(J)
      GO TO 10
 9    OBJF(J)=P(4,IS)*DF(J)
 10   CONTINUE
C
C  CALL LP SOLUTION SUBROUTINE (IMSL LIBRARY)
C
      CALL ZX3LP(TAB,IA,CON,OBJF,N,M1,M2,OPT,PSOL,DSOL,RW,IW,IER)
      IF (IER.NE.0) OPT=0.
      IF (IFLAG.EQ.2) WRITE(6,201)
      IF (IFLAG.GE.1) WRITE(6,101)IS,M,OPT
      IF (IFLAG.NE.2) RETURN
      DO 12 I=1,2
      IF (I.EQ.2) WRITE(6,201)
      DO 11 J=1,32
 11   WRITE(6,102)(TAB(J,(I-1)*15+K),K=1,15)
      WRITE(6,202)
 12   WRITE(6,103)(OBJF((I-1)*15+K),K=1,15)
      WRITE(6,201)
      WRITE(6,104)(CON(I),I=1,16)
      WRITE(6,104)(CON(I),I=17,32)
      WRITE(6,202)
      WRITE(6,105)(PSOL(I),I=1,15)
      WRITE(6,105)(PSOL(I),I=16,30)
      WRITE(6,202)
      WRITE(6,106)(DSOL(I),I=1,16)
      WRITE(6,106)(DSOL(I),I=17,32)
      RETURN
 101  FORMAT(' SYSTEM:',I2,' YEAR:',I3,' OPTIMUM:',F8.2)
 102  FORMAT(' TAB:   ',15F8.2)
 103  FORMAT(' OBJF: ',15F8.2)
 104  FORMAT(' CONST:',16F7.2)
 105  FORMAT(' PRIME:',15F8.2)
```

```
 106   FORMAT(' DUAL: ',16F7.2)
 201   FORMAT(1H1)
 202   FORMAT(1X)
       END
       SUBROUTINE LP2(IS,JS,M,OPT)
       DIMENSION TAB(35,60),CON(35),OBJF(60),PSOL(60),DSOL(35),
      +          RW(1326),IW(101),P(4,5),YLIM(30),DF(30),M(2)
       COMMON/PARAMS/P,YLIM,LIMU,IFLAG,DF
       COMMON/COUNTS/IC
       IA=35
       N=60
       M1=31
       M2=2
       IC=IC+1
C
C  INITIALIZE TABLEAU
C
       DO 1 I=1,35
       DO 1 J=1,60
  1    TAB(I,J)=0.
C
C  BUDGETARY CONSTRAINTS
C
       DO 2 J=1,30
  2    CON(J)=YLIM(J)
       DO 7 I=1,2
       I1=30*(I-1)
       IF (I.EQ.1) IUNIT=IS
       IF (I.EQ.2) IUNIT=JS
       DO 7 J=1,30
       IF (J-M(I)) 3,6,6
  3    TAB(J,J+I1)=P(1,IUNIT)
       GO TO 7
  4    K1=M(I)
       K2=J-1
       DO 5 K=K1,K2
  5    TAB(J,K+I1)=-P(2,IUNIT)
  6    TAB(J,J+I1)=P(3,IUNIT)
  7    CONTINUE
C
C  DEMAND FOR NEW UNITS BOUNDED
C
       DO 8 J=1,30
       IF (J.GE.M(1)) TAB(31,J)=1.
  8    IF (J.GE.M(2)) TAB(31,J+30)=1.
       CON(31)=FLOAT(LIMU)
C
C  CONSTRAINT TO PAY R&D
C
       CON(32)=1.
       CON(33)=1.
       DO 9 J=1,30
       IF (J.LT.M(1)) TAB(32,J)=1.
  9    IF (J.LT.M(2)) TAB(33,J+30)=1.
C
```

```
C  OBJECTIVE FUNCTION
C
      DO 11 I=1,2
      I1=30*(I-1)
      IF (I.EQ.1) IUNIT=IS
      IF (I.EQ.2) IUNIT=JS
      DO 11 J=1,30
      IF (J.GE.M(I)) GO TO 10
      OBJF(J+I1)=-P(1,IUNIT)*DF(J)
      GO TO 11
 10   OBJF(J+I1)=P(4,IUNIT)*DF(J)
 11   CONTINUE
C
C  CALL LP SOLUTION SUBROUTINE (IMSL LIBRARY)
C
      CALL ZX3LP(TAB,IA,CON,OBJF,N,M1,M2,OPT,PSOL,DSOL,RW,IW,IER)
      IF (IER.NE.0) OPT=0.
      IF (IFLAG.EQ.2) WRITE(6,201)
      IF (IFLAG.GE.1) WRITE(6,101)IS,JS,(M(I),I=1,2),OPT
      IF (IFLAG.NE.2) RETURN
      DO 13 I=1,4
      IF (I.GT.1) WRITE(6,201)
      DO 12 J=1,33
 12   WRITE(6,102)(TAB(J,(I-1)*15+K),K=1,15)
      WRITE(6,202)
 13   WRITE(6,103)(OBJF((I-1)*15+K),K=1,15)
      WRITE(6,201)
      WRITE(6,104)(CON(I),I=1,17)
      WRITE(6,104)(CON(I),I=18,34)
      WRITE(6,202)
      DO 14 I=1,4
      I1=15*(I-1)
 14   WRITE(6,105)(PSOL(J+I1),J=1,15)
      WRITE(6,202)
      WRITE(6,106)(DSOL(I),I=1,17)
      WRITE(6,106)(DSOL(I),I=18,34)
      RETURN
 101  FORMAT(' SYSTEMS:',2I2,' YEARS:',2I3,' OPTIMUM:',F8.2)
 102  FORMAT(' TAB:  ',15F8.2)
 103  FORMAT(' OBJF: ',15F8.2)
 104  FORMAT(' CONST:',17F7.2)
 105  FORMAT(' PRIME:',15F8.2)
 106  FORMAT(' DUAL: ',17F7.2)
 201  FORMAT(1H1)
 202  FORMAT(1X)
      END
      SUBROUTINE LP3(IS,JS,KS,M,OPT)
      DIMENSION TAB(36,90),CON(36),OBJF(90),PSOL(90),DSOL(36),
     +          RW(1399),IW(103),P(4,5),YLIM(30),DF(30),M(3)
      COMMON/PARAMS/P,YLIM,LIMU,IFLAG,DF
      COMMON/COUNTS/IC
      IA=36
      N=90
      M1=31
      M2=3
```

```
      IC=IC+1
C
C  INITIALIZE TABLEAU
C
      DO 1 I=1,36
      DO 1 J=1,90
 1    TAB(I,J)=0.
C
C  BUDGETARY CONSTRAINTS
C
      DO 2 J=1,30
 2    CON(J)=YLIM(J)
      DO 7 I=1,3
      I1=30*(I-1)
      IF (I.EQ.1) IUNIT=IS
      IF (I.EQ.2) IUNIT=JS
      IF (I.EQ.3) IUNIT=KS
      DO 7 J=1,30
      IF (J-M(I)) 3,6,6
 3    TAB(J,J+I1)=P(1,IUNIT)
      GO TO 7
 4    K1=M(I)
      K2=J-1
      DO 5 K=K1,K2
 5    TAB(J,K+I1)=-P(2,IUNIT)
 6    TAB(J,J+I1)=P(3,IUNIT)
 7    CONTINUE
C
C  DEMAND FOR NEW UNITS BOUNDED
C
      DO 8 J=1,30
      IF (J.GE.M(1)) TAB(31,J)=1.
      IF (J.GE.M(2)) TAB(31,J+30)=1.
 8    IF (J.GE.M(3)) TAB(31,J+60)=1.
      CON(31)=FLOAT(LIMU)
C
C  CONSTRAINT TO PAY R&D
C
      CON(32)=1.
      CON(33)=1.
      CON(34)=1.
      DO 9 J=1,30
      IF (J.LT.M(1)) TAB(32,J)=1.
      IF (J.LT.M(2)) TAB(33,J+30)=1.
 9    IF (J.LT.M(3)) TAB(34,J+60)=1.
C
C  OBJECTIVE FUNCTION
C
      DO 11 I=1,3
      I1=30*(I-1)
      IF (I.EQ.1) IUNIT=IS
      IF (I.EQ.2) IUNIT=JS
      IF (I.EQ.3) IUNIT=KS
      DO 11 J=1,30
      IF (J.GE.M(I)) GO TO 10
```

```
      OBJF(J+I1)=-P(1,IUNIT)*DF(J)
      GO TO 11
 10   OBJF(J+I1)=P(4,IUNIT)*DF(J)
 11   CONTINUE
C
C  CALL LP SOLUTION SUBROUTINE (IMSL LIBRARY)
C
      CALL ZX3LP(TAB,IA,CON,OBJF,N,M1,M2,OPT,PSOL,DSOL,RW,IW,IER)
      IF (IER.NE.0) OPT=0.
      IF (IFLAG.EQ.2) WRITE(6,201)
      IF (IFLAG.GE.1) WRITE(6,101)IS,JS,KS,(M(I),I=1,3),OPT
      IF (IFLAG.NE.2) RETURN
      DO 13 I=1,6
      IF (I.GT.1) WRITE(6,201)
      DO 12 J=1,34
 12   WRITE(6,102)(TAB(J,(I-1)*15+K),K=1,15)
      WRITE(6,202)
 13   WRITE(6,103)(OBJF((I-1)*15+K),K=1,15)
      WRITE(6,201)
      WRITE(6,104)(CON(I),I=1,17)
      WRITE(6,104)(CON(I),I=18,34)
      WRITE(6,202)
      DO 14 I=1,6
      I1=15*(I-1)
 14   WRITE(6,105)(PSOL(J+I1),J=1,15)
      WRITE(6,202)
      WRITE(6,106)(DSOL(I),I=1,17)
      WRITE(6,106)(DSOL(I),I=18,34)
      RETURN
 101  FORMAT(' SYSTEMS:',3I2,' YEARS:',3I3,' OPTIMUM:',F8.2)
 102  FORMAT(' TAB:  ',15F8.2)
 103  FORMAT(' OBJF: ',15F8.2)
 104  FORMAT(' CONST:',17F7.2)
 105  FORMAT(' PRIME:',15F8.2)
 106  FORMAT(' DUAL: ',17F7.2)
 201  FORMAT(1H1)
 202  FORMAT(1X)
      END
      SUBROUTINE LP4(IS,JS,KS,LS,M,OPT)
      DIMENSION TAB(37,120),CON(37),OBJF(120),PSOL(120),DSOL(37),
     +          RW(1474),IW(105),P(4,5),YLIM(30),DF(30),M(4)
      COMMON/PARAMS/P,YLIM,LIMU,IFLAG,DF
      COMMON/COUNTS/IC
      IA=37
      N=120
      M1=31
      M2=4
      IC=IC+1
C
C  INITIALIZE TABLEAU
C
      DO 1 I=1,37
      DO 1 J=1,120
 1    TAB(I,J)=0.
C
```

```
C  BUDGETARY CONSTRAINTS                                                DAL08260
C                                                                       DAL08270
      DO 2 J=1,30                                                       DAL08280
 2    CON(J)=YLIM(J)                                                    DAL08290
      DO 7 I=1,4                                                        DAL08300
      I1=30*(I-1)                                                       DAL08310
      IF (I.EQ.1) IUNIT=IS                                              DAL08320
      IF (I.EQ.2) IUNIT=JS                                              DAL08330
      IF (I.EQ.3) IUNIT=KS                                              DAL08340
      IF (I.EQ.4) IUNIT=LS                                              DAL08350
      DO 7 J=1,30                                                       DAL08360
      IF (J-M(I)) 3,6,6                                                 DAL08370
 3    TAB(J,J+I1)=P(1,IUNIT)                                            DAL08380
      GO TO 7                                                           DAL08390
 4    K1=M(I)                                                           DAL08400
      K2=J-1                                                            DAL08410
      DO 5 K=K1,K2                                                      DAL08420
 5    TAB(J,K+I1)=-P(2,IUNIT)                                           DAL08430
 6    TAB(J,J+I1)=P(3,IUNIT)                                            DAL08440
 7    CONTINUE                                                          DAL08450
C                                                                       DAL08460
C  DEMAND FOR NEW UNITS BOUNDED                                         DAL08470
C                                                                       DAL08480
      DO 8 J=1,30                                                       DAL08490
      IF (J.GE.M(1)) TAB(31,J)=1.                                       DAL08500
      IF (J.GE.M(2)) TAB(31,J+30)=1.                                    DAL08510
      IF (J.GE.M(3)) TAB(31,J+60)=1.                                    DAL08520
 8    IF (J.GE.M(4)) TAB(31,J+90)=1.                                    DAL08530
      CON(31)=FLOAT(LIMU)                                               DAL08540
C                                                                       DAL08550
C  CONSTRAINT TO PAY R&D                                                DAL08560
C                                                                       DAL08570
      CON(32)=1.                                                        DAL08580
      CON(33)=1.                                                        DAL08590
      CON(34)=1.                                                        DAL08600
      CON(35)=1.                                                        DAL08610
      DO 9 J=1,30                                                       DAL08620
      IF (J.LT.M(1)) TAB(32,J)=1.                                       DAL08630
      IF (J.LT.M(2)) TAB(33,J+30)=1.                                    DAL08640
      IF (J.LT.M(3)) TAB(34,J+60)=1.                                    DAL08650
 9    IF (J.LT.M(4)) TAB(35,J+90)=1.                                    DAL08660
C                                                                       DAL08670
C  OBJECTIVE FUNCTION                                                   DAL08680
C                                                                       DAL08690
      DO 11 I=1,4                                                       DAL08700
      I1=30*(I-1)                                                       DAL08710
      IF (I.EQ.1) IUNIT=IS                                              DAL08720
      IF (I.EQ.2) IUNIT=JS                                              DAL08730
      IF (I.EQ.3) IUNIT=KS                                              DAL08740
      IF (I.EQ.4) IUNIT=LS                                              DAL08750
      DO 11 J=1,30                                                      DAL08760
      IF (J.GE.M(I)) GO TO 10                                           DAL08770
      OBJF(J+I1)=-P(1,IUNIT)*DF(J)                                      DAL08780
      GO TO 11                                                          DAL08790
 10   OBJF(J+I1)=P(4,IUNIT)*DF(J)                                       DAL08800
```

```
 11    CONTINUE
C
C   CALL LP SOLUTION SUBROUTINE (IMSL LIBRARY)
C
       CALL ZX3LP(TAB,IA,CON,OBJF,N,M1,M2,OPT,PSOL,DSOL,RW,IW,IER)
       IF (IER.NE.0) OPT=0.
       IF (IFLAG.EQ.2) WRITE(6,201)
       IF (IFLAG.GE.1) WRITE(6,101)IS,JS,KS,LS,(M(I),I=1,4),OPT
       IF (IFLAG.NE.2) RETURN
       DO 13 I=1,8
       I1=15*(I-1)
       IF (I.GT.1) WRITE(6,201)
       DO 12 J=1,35
 12    WRITE(6,102)(TAB(J,I1+K),K=1,15)
       WRITE(6,202)
 13    WRITE(6,103)(OBJF(I1+K),K=1,15)
       WRITE(6,201)
       WRITE(6,104)(CON(I),I=1,18)
       WRITE(6,104)(CON(I),I=19,36)
       WRITE(6,202)
       DO 14 I=1,8
       I1=15*(I-1)
 14    WRITE(6,105)(PSOL(J+I1),J=1,15)
       WRITE(6,202)
       WRITE(6,106)(DSOL(I),I=1,18)
       WRITE(6,106)(DSOL(I),I=19,36)
       RETURN
 101   FORMAT(' SYSTEMS:',4I2,' YEARS:',4I3,' OPTIMUM:',F8.2)
 102   FORMAT(' TAB:  ',15F8.2)
 103   FORMAT(' OBJF: ',15F8.2)
 104   FORMAT(' CON:',18F7.2)
 105   FORMAT(' PRIME:',15F8.2)
 106   FORMAT(' DUAL:',18F7.2)
 201   FORMAT(1H1)
 202   FORMAT(1X)
       END
       SUBROUTINE LP5(M,OPT)
       DIMENSION TAB(38,150),CON(38),OBJF(150),PSOL(150),DSOL(38),
     +           RW(1551),IW(107),P(4,5),YLIM(30),DF(30),M(5)
       COMMON/PARAMS/P,YLIM,LIMU,IFLAG,DF
       COMMON/COUNTS/IC
       IA=38
       N=150
       M1=31
       M2=5
       IC=IC+1
C
C   INITIALIZE TABLEAU
C
       DO 1 I=1,38
       DO 1 J=1,150
 1     TAB(I,J)=0.
C
C   BUDGETARY CONSTRAINTS
C
```

```
      DO 2 J=1,30
 2    CON(J)=YLIM(J)
      DO 7 I=1,5
      I1=30*(I-1)
      DO 7 J=1,30
      IF (J-M(I)) 3,6,6
 3    TAB(J,J+I1)=P(1,I)
      GO TO 7
 4    K1=M(I)
      K2=J-1
      DO 5 K=K1,K2
 5    TAB(J,K+I1)=-P(2,I)
 6    TAB(J,J+I1)=P(3,I)
 7    CONTINUE
C
C  DEMAND FOR NEW UNITS BOUNDED
C
      DO 8 J=1,30
      DO 8 I=1,5
 8    IF (J.GE.M(I)) TAB(31,J+30*(I-1))=1.
      CON(31)=FLOAT(LIMU)
C
C  CONSTRAINT TO PAY R&D
C
      DO 9 I=1,5
      I1=I-1
      CON(32+I1)=1.
      DO 9 J=1,30
 9    IF (J.LT.M(I)) TAB(32+I1,J+30*I1)=1.
C
C  OBJECTIVE FUNCTION
C
      DO 11 I=1,5
      I1=30*(I-1)
      DO 11 J=1,30
      IF (J.GE.M(I)) GO TO 10
      OBJF(J+I1)=-P(1,I)*DF(J)
      GO TO 11
 10   OBJF(J+I1)=P(4,I)*DF(J)
 11   CONTINUE
C
C  CALL LP SOLUTION SUBROUTINE (IMSL LIBRARY)
C
      CALL ZX3LP(TAB,IA,CON,OBJF,N,M1,M2,OPT,PSOL,DSOL,RW,IW,IER)
      IF (IER.NE.0) OPT=0.
      IF (IFLAG.EQ.2) WRITE(6,201)
      IF (IFLAG.GE.1) WRITE(6,101)(M(I),I=1,5),OPT
      IF (IFLAG.NE.2) RETURN
      DO 13 I=1,10
      I1=15*(I-1)
      IF (I.GT.1) WRITE(6,201)
      DO 12 J=1,36
 12   WRITE(6,102)(TAB(J,I1+K),K=1,15)
      WRITE(6,202)
 13   WRITE(6,103)(OBJF(I1+K),K=1,15)
```

```
      WRITE(6,201)                                                      DAL09910
      WRITE(6,104)(CON(I),I=1,18)                                       DAL09920
      WRITE(6,104)(CON(I),I=19,36)                                      DAL09930
      WRITE(6,202)                                                      DAL09940
      DO 14 I=1,10                                                      DAL09950
      I1=15*(I-1)                                                       DAL09960
 14   WRITE(6,105)(PSOL(J+I1),J=1,15)                                   DAL09970
      WRITE(6,202)                                                      DAL09980
      WRITE(6,106)(DSOL(I),I=1,18)                                      DAL09990
      WRITE(6,106)(DSOL(I),I=19,36)                                     DAL10000
      RETURN                                                            DAL10010
 101  FORMAT(' ALL SYSTEMS - YEARS:',5I3,' OPTIMUM:',F8.2)              DAL10020
 102  FORMAT(' TAB:  ',15F8.2)                                          DAL10030
 103  FORMAT(' OBJF: ',15F8.2)                                          DAL10040
 104  FORMAT(' CON:',18F7.2)                                            DAL10050
 105  FORMAT(' PRIME:',15F8.2)                                          DAL10060
 106  FORMAT(' DUAL:',18F7.2)                                           DAL10070
 201  FORMAT(1H1)                                                       DAL10080
 202  FORMAT(1X)                                                        DAL10090
      END                                                               DAL10100
```

REFERENCES

[1] Konstantin Tsiolkowsky, Beyond the Planet Earth Translated by Kenneth Syers. Pergamon Press, Inc. 1960

[2] -, The Boeing Story

[3] -, Advancements in the Aerospace Sciences Volume 31: Skylab Results

[4] Robert A Heinlein, The Man Who Sold The Moon

[5] Peter Glaser, "Solar Power via Satellite" Astronautics & Aeronautics August, 1973

[6] -, Satellite Power System Program Summary DOE/ER-0022, December, 1978

[7] -, Satellite Power System Concept Development and Evaluation Program DOE/ER-0023, October, 1978

[8] M.I.T. Space Systems Laboratory, Extraterrestrial Processing and Manufacturing of Large Space Systems NASA CR-161293, September, 1979

[9] Hass, Mitchell, Stone, and Downes, Financing the Energy Industry Ballinger Publishing Company, 1975

[10] -, pamphlet in Cambridge Electric bill mailing, March, 1981

[11] Gerard O'Neill, "The Colonization of Space" Physics Today September 1974

[12] Gerard O'Neill, "Space Colonies and the Energy Supply to Earth" Physics Today

[13] Johnson and Holbrow, ed., Space Settlements: A Design Study NASA SP-413, 1977

[14] D.B.S. Smith, ed., A Systems Design for a Prototype Space Colony M.I.T. Department of Aeronautics and Astronautics, 1976

[15] Billingham, Gilbreath, and Gossett, ed., Space Resources and Space Settlements NASA SP-428, 1979

[16] David L. Akin, "Optimization of Space Manufacturing Systems" AIAA 79-1394, May, 1979

[17] Mary L. Bowden and David L. Akin, "Underwater Simulation of Human Dynamics and Productivities in Extra-Vehicular Assembly" IAF 79-109, Sept., 1979

[18] D. L. Akin and M. L. Bowden, "Assembly of Large Space Structures: Simulation in Neutral Buoyancy and Parabolic Flight" AIAA 81-0436, Feb., 1981

[19] General Dynamics Convair Division, Lunar Resources Utilization for Space Construction GDC-ASP79-001, April, 1979

[20] Gordon R. Woodcock, "Space Operations Center" AIAA 81-0440, Feb., 1981

[21] D. L. Akin, "Some Applications of Nonstationary Geosynchronous Orbits" AIAA 78-1407, Aug., 1978

[22] Marshall H. Kaplan, Modern Spacecraft Dynamics and Control John Wiley & Sons, 1976

[23] Gerard O'Neill, "The Low (Profile) Road to Space Industrialization" Astronautics & Aeronautics March, 1978

[24] Richard Battin, Astronautical Guidance

[25] Wernher von Braun, The Mars Project University of Illinois Press, 1962

[26] Bate, Mueller, and White, Fundamentals of Astrodynamics Dover Publications, 1971

[27] L. D'Amario and T. N. Edelbaum, "Minimum Impulse Three-Body Trajectories" AIAA 73-145, Jan., 1973

[28] L. A. D'Amario and T. N. Edelbaum, "Error Analysis of Multi-conic Techniques" AAS/AIAA Astrodynamics Conference, July, 1973

[29] -, Second Symposium on Radiation in Space, Gatlinburg, 1963

[30] J. B. Kendrick, ed., TRW Space Data TRW Systems Group, 1967

[31] R. H. Miller and D. L. Akin, "Logistics Costs of Solar Power Satellites" Space Solar Power Review, pp. 191-208, 1980

[32] C. R. Glatt, WAATS - A Computer Program for Weights Analysis of Advanced TRansportation Systems NASA CR-2420, 1974

[33] R. H. Miller, "The Effect of Demand on Optimum Launch Vehicle Size" Journal of Spacecraft and Rockets

[34] Aviation Week and Space Technology

[35] Masayoshi Kanabayashi, "Japan Sets Mini-Entry in Space Race" The Wall Street Journal, July 20, 1981

[36] "La Navette Hermes", periodical unknown

[37] -,Advanced Space Engine pamphlet, Rockwell International, 1977

[38] Boris Kit and Douglas S. Evered, Rocket Propellant Handbook Macmillan, 1960

[39] Kurt P. Johnson and Richard J. Holl, "Nuclear Shuttle for Interorbital and Transplanetary Applications" Journal of Spacecraft and Rockets, June, 1971

[40] Phillip G. Hill and Carl R. Peterson, Mechanics and Thermodynamics of Propulsion Addison-Wesley, 1970

[41] -, A Forecast of Space Technology 1980-2000 NASA SP-387, 1976

[42] G. H. McLafferty, "Survey of Advanced Concepts in Nuclear Propulsion" Journal of Spacecraft and Rockets, October, 1968

[43] Clarence A. Robinson, Jr., "Aerospace Aids Fusion Power Concept" Aviation Week and Space Technology, June 12, 1978

[44] Ernst Stuhlinger, Ion Propulsion for Space Flight McGraw-Hill, 1964

[45] B. Lenz, R. J. Walther, and H. Loeb, "Plasma Diagnostic Contributions to Advanced RIT-Engines" AIAA 76-1012, Nov., 1976

[46] David C. Byers, personal communication, Dec. 20, 1978

[47] Ross M. Jones, communication to David B. S. Smith, July 9, 1980

[48] Kolm, Fine, Mongeau, and Williams, "Electromagnetic Propulsion Alternatives" AIAA 79-1400, May, 1979

[49] J. Angelo, "Nuclear Power for Space Applications", AAS 80-220, October, 1980

[50] NASA, Space Operations Center: A Concept Analysis JSC-16277, Nov., 1979

[51] JSC Operations Analysis Branch, "Space Operations Center Costing Rationale" (unpublished) Nov., 1979

[52] Bradley, Hax, and Magnanti, Applied Mathematical Programming Addison-Wesley, 1977

[53] David G. Luenberger, Introduction to Linear and Nonlinear Programming Addison-Wesley, 1973

[54] Peter van de Kamp, Elements of Astromechanics W. H. Freeman, 1964

[55] Daniel A. Heald, "Economic Benefits of the OTV Program" 80-IAA-38, Sept., 1980

[56] MSFC Program Development Office, Traffic Model for the Space Shuttle NASA TM-X-64717, January, 1973

[57] Val Heinz, "Analysis of a Space-Based Fuel Station" AIAA National Student Conference, 1976

[58] Current Events session at the IAF Congress, Munich, Germany, Sept. 17, 1979

[59] Preston Layton, "Our Next Steps in Space" Astronautics & Aeronautics, May, 1972