PROBLEM SET #6

TO: PROF. DAVID MILLER, PROF. JOHN KEESEE, AND MS. MARILYN GOOD

FROM: NAMES WITHELD

SUBJECT: PROBLEM SET #6 (LIFE SUPPORT, PROPULSION, AND POWER FOR AN EARTH-TO-MARS HUMAN TRANSPORTATION VEHICLE)

DATE: 12/3/2003

MOTIVATION

Mars is of great scientific interest given the potential evidence of past or present life. Recent evidence indicating the past existence of water deposits underscore its scientific value. Other motivations to go to Mars include studying its climate history through exploration of the polar layers. This information could be correlated with similar data from Antarctica to characterize the evolution of the Solar System and its geological history. Long-term goals might include the colonization of Mars.

As the closest planet with a relatively mild environment, there exists a unique opportunity to explore Mars with humans. Although we have used robotic spacecraft successfully in the past to study Mars, humans offer a more efficient and robust exploration capability. However, human spaceflight adds both complexity and mass to the space vehicle and has a significant impact on the mission design. Humans require an advanced environmental control and life support system, and this subsystem has high power requirements thus directly affecting the power subsystem design.

A Mars mission capability is likely to be a factor in NASA's new launch architecture design since the resulting launch architecture will need take into account the estimated spacecraft mass required for such a mission. To design a Mars mission, various propulsion system options must be evaluated and compared for their efficiency and adaptability to the required mission duration. A tool that evaluates the design trades between the human life support, power, and propulsion systems in terms of mass and viability for various possible mission durations would be fundamental to the design and evaluation of a potential mission to Mars.

PROBLEM STATEMENT

Design a software tool that performs a trade of the life support, propulsion, and power subsystems for a human transportation vehicle from earth to Mars. Based on the orbital transfer between the two planets, mission duration and ΔV will vary. These subsystems are then determined by the mission duration and ΔV for the human transportation vehicle. The objective is to determine the optimal design between system mass and mission duration for a variety of scenarios.

APPROACH

Orbits will be used as the independent variable for our three subsystems and total mass and power will be used as the analysis metric. This analysis assumes that the earth return vehicle is already present on Mars. Therefore, this transportation vehicle will only address the earth to Mars segment of the journey. First, mission duration and ΔV will be determined for a range of feasible transfer orbits (ie. Hohmann and high energy one-tangent-burns). For each specific transfer orbit, the life support and vehicle propulsion will be determined. The life support system will be designed based on the mission duration and crew size. For the propulsion system, various types of propulsion will be considered. These will in turn drive the power system design which encompasses both power generation and energy storage. Potential power sources include solar, radioisotope, nuclear, and fuel cell.

EARTH-MARS TRANSFER ORBITS

In order to understand the trades between the life support, propulsion, and power subsystems of an Earth-Mars human transport vehicle, these systems are sized for a particular transfer orbit. The orbit-specific parameters that drive these systems are the ΔV requirement, transfer time of flight, and the sun radial distance profile. The orbits examined range from a minimum energy Hohmann transfer up to high energy one-tangent transfers. Since the vehicle under examination is for human transport, these direct Earth-Mars transfers are most appropriate. Low energy chemical or electric propulsion systems would result in slow spiraling transfers and would leave humans in the harsh, weightless space environment for excessive amounts of time.

SUN-FRAME TRANSFER

The first necessary calculation is for the sun-frame earth-to-mars transfer. Since this analysis is intended to be a first-order approximation, earth and mars orbits are approximated as circular. First, the initial ΔV_I to begin the planetary transfer is calculated. This is the difference between the orbital circular velocity on the earth's solar orbit and the desired velocity on the earth's orbit to initiate the transfer orbit (Equation 1), where μ_S is the Sun gravity constant, $r_{E\text{-}S}$ is the Earth to sun distance, and a is the transfer orbit semi-major axis.

$$\Delta V_I = \sqrt{\mu_S\left(\frac{2}{r_{E-S}} - \frac{1}{a}\right)} - \sqrt{\frac{\mu_S}{r_{E-S}}} \tag{1}$$

To determine the final ΔV_F to circularize the vehicle trajectory upon reaching Mars, first the true anomaly, f, upon reaching Mars must be determined (Equation 3). This requires the transfer orbit eccentricity (Equation 2), e, transfer orbit semi-major axis, a, and the Mars to sun distance, $r_{E\text{-}S}$.

$$e = 1 - \frac{r_{E-S}}{a} \tag{2}$$

$$\cos f = \frac{1}{e}\left(\frac{a\left(1-e^2\right)}{r_{M-S}} - 1\right) \tag{3}$$

Next, the flight path angle at this final point (transfer orbit – mars orbit intersection) is determined (Equation 4).

$$\phi = \frac{a\tan(e\sin f)}{1+e\cos f} \tag{4}$$

Finally ΔV_F is found by Equation 5.

$$\Delta V_F = \sqrt{\frac{\mu_S}{r_{M-S}} + \mu_S\left(\frac{2}{r_{M-S}} - \frac{1}{a}\right) - 2\sqrt{\frac{\mu_S}{r_{M-S}}}\sqrt{\mu_S\left(\frac{2}{r_{M-S}} - \frac{1}{a}\right)}\cos\phi} \tag{5}$$

The initial and final ΔV can be determined for any direct transfer by performing these calculations for a value of the semi-major axis ranging from the minimum Hohmann value (Equation 6) to infinity.

$$a = \frac{r_{E-S} + r_{M-S}}{2} \tag{6}$$

EARTH-FRAME TRANSFER

Before the earth-to-mars transfer can begin, the vehicle must first escape the earth's gravity well at a parking orbit and reach the sphere of influence (SOI) with some specific velocity. This velocity at the SOI (v_{inf}) is equal to the initial sun-frame ΔV_I (Equation 1). For the analysis, it is assumed that the vehicle begins in a 200 km earth parking orbit. The ΔV_{SOI} to escape the earth and begin the earth-to-mars transfer orbit is determined by Equation 7.

$$\Delta V_{SOI} = \sqrt{v_\infty^2 + \frac{2\mu_E}{r_{park}}} - \sqrt{\frac{\mu_E}{r_{park}}} \tag{7}$$

TRANSFER TIME OF FLIGHT

For both transfers, knowing the time of flight can be important. First the eccentric anomaly, E, is calculated in Equation 8, where e is the transfer orbit eccentricity and f is the true anomaly at the final point on the transfer orbit.

$$E = \cos^{-1}\left(\frac{e + \cos f}{1 + e\cos f}\right) \tag{8}$$

Using this quantity, the time of flight (*tof*) can be found in Equation 9, where *a* is the transfer orbit semi-major axis and μ is the central body gravity constant (earth or sun)

$$tof = \sqrt{\frac{a^3}{\mu}}\left(E - e\sin E\right) \quad (9)$$

ENVIRONMENT CONTROL AND LIFE SUPPORT SYSTEM (ECLSS)

The ECLSS is a complex subsystem composed of various components. Two important design metrics are power and system mass, but reliability, and safety are also vital factors that are difficult to enumerate. The ECLSS also has interdependencies on the power, thermal and structure subsystems.

Two conflicting design principles exist. One approach is to minimize power use and technology development costs by using pre-existing flight-tested technologies, for example, technologies currently in use on the ISS, and the second approach is to implement what is sometimes referred to as an ALS, or advanced life support system. The idea behind using an ALS is to close the mass loop through the use of regenerative systems and hence to recover as much mass as possible over the mission duration. This can help to minimize initial launch mass and put less stress on the propulsion system. The development of ALS systems is especially important if current propulsion technology – i.e. Chemical rockets – are to be used for a Mars mission. For example, laundering clothing during the mission as opposed to launching enough clothing required for the entire mission would help to reduce launch mass.

In this problem set we look at possible ECLSS designs for a Mars-Transit vehicle. From an ECLSS design point of view, and indeed from a human mission point of view, the optimal solution is to minimize transfer time from Earth to Mars as much as possible. This would help not only from a mass point of view, but also to limit the adverse affects of radiation (Mars transit involves possibly high levels of galactic cosmic rays and solar protons) and microgravity on the crew members. However, decreasing transfer time is only feasible up to a certain point after which the propellant mass becomes prohibitive.

The power and ECLSS subsystems are computed via an Excel spreadsheet, with the power data being incorporated into a Matlab file for processing in conjunction with the power module.

LIFE SUPPORT SUBSYSTEMS

Figure 1 below shows a schematic ECLSS that can be broken down into various subsystems including those described below:

The *Air Revitalization Subsystem* maintains the atmospheric environment, including pressure control, composition maintenance, and trace elements. In addition to interfacing with all of the other life support systems, it also is responsible for detecting and responding to fires.

The *Biomass Subsystem* produces and stores agricultural products for the Food Subsystem. The Biomass Subsystem also serves to regenerate air and water.

The *Food Subsystem* stores ready-to-eat, prepackaged foods, beverages, and ingredients included on the spacecraft at launch. It also receives agricultural products from the Biomass Subsystem and stores them for consumption as necessary.

The *Thermal Subsystem* maintains temperature and humidity ranges for the crew and rejects waste heat to the space environment as needed. It is assumed that the rejection of heat to the environment provides an adequate amount of air circulation.

The *Waste Subsystem* collects and processes solid waste material, including human waste, inedible biomass, and food packaging materials.

The *Water Subsystem* stores and distributes water as necessary for consumption, hygiene, and other purposes. The Water Subsystem is also responsible for collecting, transporting, and processing wastewater.

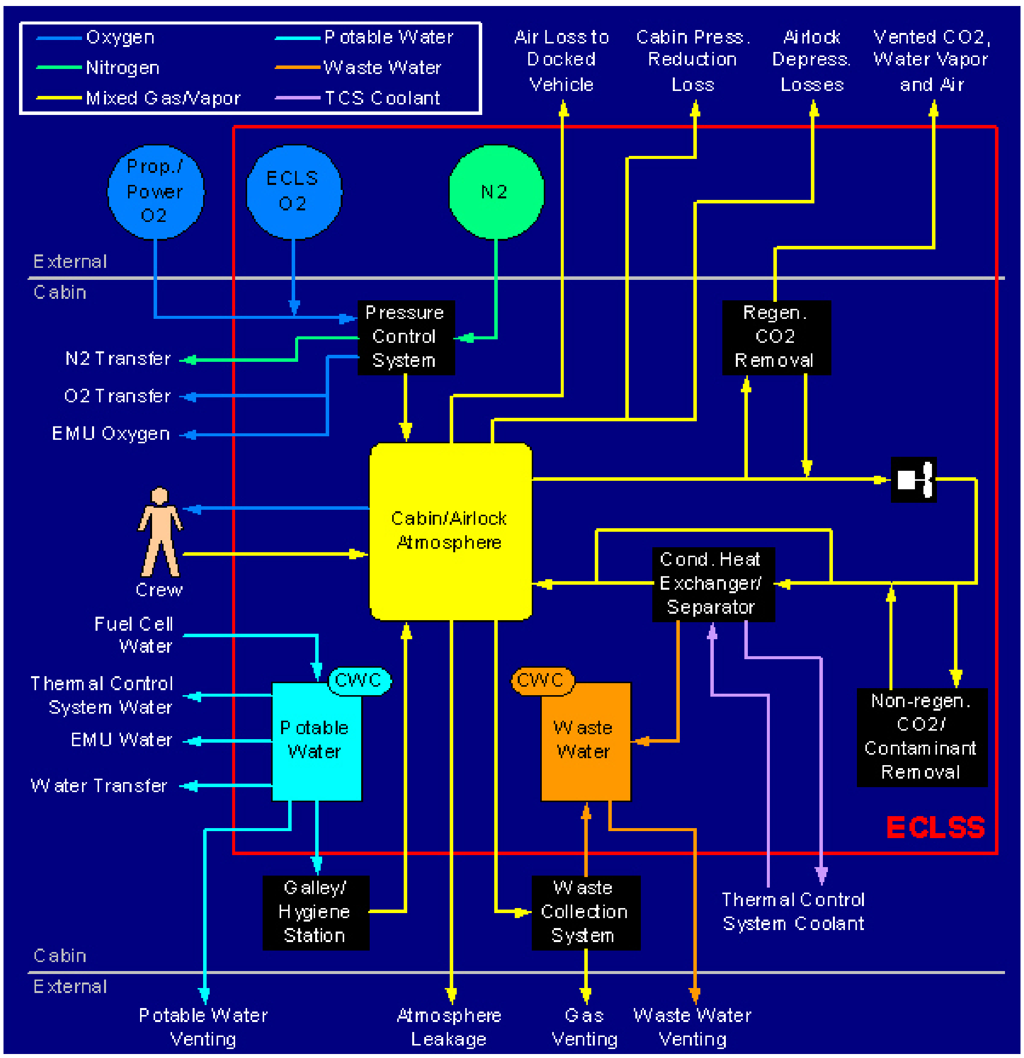

Figure 1. Schematic diagram of Environment Control and Life Support System [Lawson, 2003]

Various methods can be used to compare technologies when designing an ECLSS system. The NASA/MSFC and McDonnell Douglas Method uses eight trade study parameters to compare the available systems: mass, power consumption, volume, resupply (for shorter missions), development potential, emergency operation, reliability and safety. The method implemented here is based on NASA's Equivalent System Mass (ESM) metric that expresses mass, volume, power consumption, cooling requirement and crew-time requirement in terms of a so-called "equivalent" mass. Development potential, emergency operation, reliability and safety are considered in selecting the technologies to compare via the ESM metric and are not expressed quantitatively. ESM is often used as a transportation cost measure in studies comparing Advanced Life Support Systems (ALS) since the cost to transport a payload is proportional to its mass [Levri, 2003].

In this analysis the power subsystem is being studied in detail, so the ESM equation has been modified to remove the power equivalency factor. Instead, estimated power consumption is determined for each subsystem and used as an input into the power module. Further analysis follows the discussion of ESM, although power values are included in description of ECLSS options for comparison purposes. ESM* refers to the version of ESM without power equivalency (Equation 10).

The equation used for the ESM is that developed in [Levri, 2003]. It is a sum of the ESM values over the ECLSS subsystems being considered. The subscript *i* indicates the index of the subsystem.

$$ESM^* = \sum_{i=1}^{n} \left[\begin{array}{l} \left(M_i \cdot SF_i\right) + \left(V_i \cdot V_{eqi}\right) + \left(C_i \cdot C_{eqi}\right) + \\ \quad \left(CT_i \cdot D \cdot CT_{eqi}\right) + \left(M_{TDi} \cdot D \cdot SF_{TDi}\right) + \left(V_{TDi} \cdot D \cdot V_{eqi}\right) \end{array} \right] \tag{10}$$

Mass equivalency factors are used to express the non-mass parameters volume, cooling requirement and crewtime in terms of mass units. For example, the volume equivalency is computed as the total empty mass of the spacecraft divided by the total volume capacity. Mass is considered to be a resource and each unit of volume (power, cooling, crewtime) is expressed in terms of the amount of mass it uses.

The rationale for using crewtime as an equivalency factor is explained in detail in [Levri, 2003]. Briefly, crewtime used in maintenance or operation of the life support system takes away from crewtime available to carry out mission objectives. If the life support system requires too much crewtime, the crew size would have to be increased to achieve scientific objectives and hence the life support system would be incrementally larger.

The following Table 1 describes the parameters involved in the ESM calculation for each subsystem of the ECLSS onboard the spacecraft.

Table 1. Parameters involved in ESM* equation.

Variable	Description	Units	Comments
M_{Ii}	Initial mass	Kg	
SF_{Ii}	Initial mass stowage factor	kg/kg	Accounts for additional hardware required to fasten and contain equipment
V_{Ii}	Initial volume	m^3	Any pressurized volume necessary for life support hardware
V_{eqi}	Mass equivalency factor for the pressurized volume support infrastructure	kg/m^3	
C_i	Cooling requirement	kW	Typically equivalent to subsystem power needs

C_{eqi}	Mass equivalency factor for the cooling infrastructure	kg/kW	
CT_i	Crewtime requirement	CM-h/y	Time spent by the crew in operation/maintenance
CT_{eqi}	Mass equivalency factor for the crewtime	kg/CM-h	
D	Duration of the mission segment of interest	y	Calculated in trajectory generation module (dependent on orbit)
M_{TDi}	Time/Event-dependent mass	kg/y	Neglected in this analysis
SF_{TDi}	Time/Event-dependent stowage factor	kg/kg	Neglected in this analysis
V_{TDi}	Time/Event-dependent volume	m^3	Neglected in this analysis

<u>General notes and assumptions</u>:

- Subsystem-specific equivalencies are not used [Levri, 2003]. The equivalency factors are instead constant over all subsystems for the mission segment under consideration. Since the volume equivalency factor depends on the amount of radiation shielding provided, it may vary according to subsystem if for example the plant-growth area has less shielding. A more detailed model of the desired vehicle design would be required to approximate this parameter. Similarly, if different power systems are used to power the various subsystems, the power equivalency should be variable.

- Time dependent mass, stowage factor and volume values are not used in the code. Incorporating these variables could be an extension to the current work.

- A crew size of 6 is assumed for equipment sizing.

- Assumed food consumption is 1.82 kg/CM-d plus 0.23 kg/CM-d of disposable packaging for a total packaged food mass of 2.05 kg/CM-d.

- The expected oxygen generation and carbon dioxide removal from the biomass chamber are not currently incorporated into the ALS air revitalization design.

- Plants grown in the biomass chamber could also be used for water regeneration through a transpiration process however this is not taken into consideration in this study.

Two types of systems are compared in the following analysis intended to develop a life support system for a Mars transit vehicle. The first is model of the ISS-Baseline approach to life support, and the second is a model of an Advanced Life Support system, which makes use of as many regenerative technologies as possible. This section of the report builds on previous work and thus detailed descriptions of the function of subsystem components will not be developed in detail here. The following two figures (Figure 2 and Figure 3) show schematics of the ISS-Baseline and ALS Mars Transfer vehicle ECLSS designs respectively.

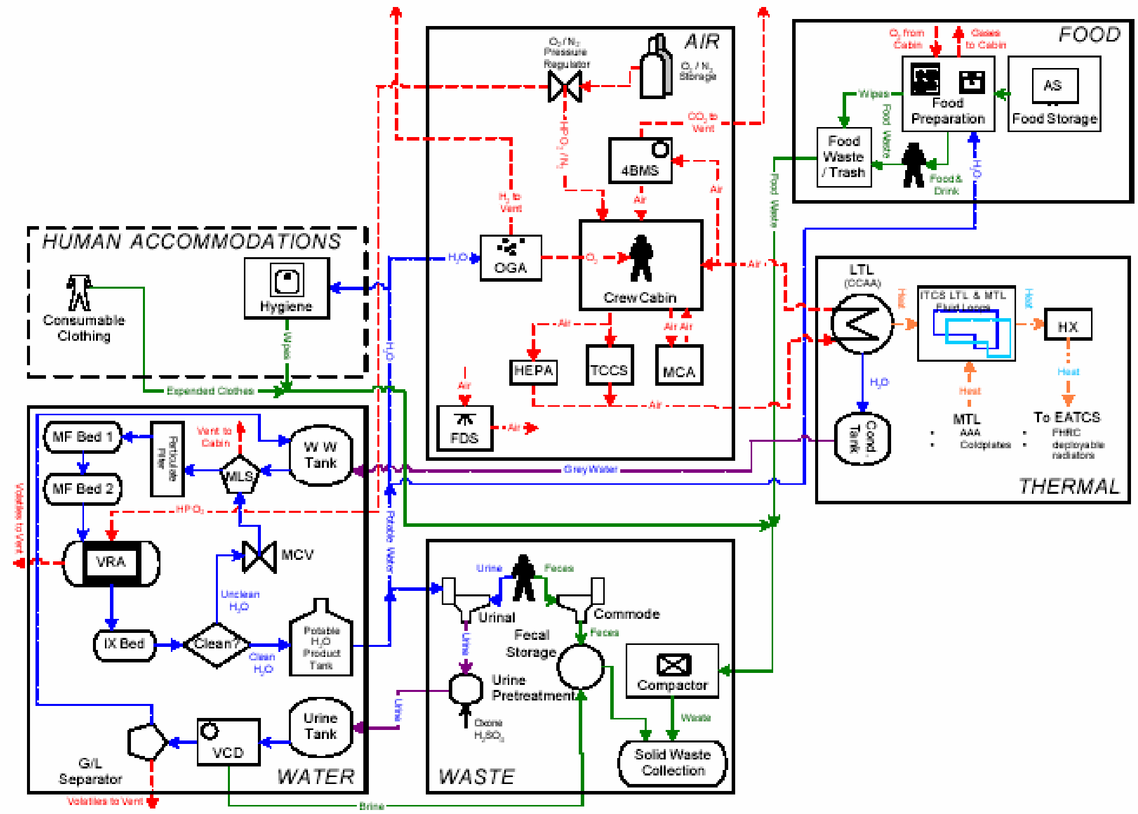

Figure 2. Schematic diagram of Environment Control and Life Support System for a Mars Transfer Vehicle using ISS-Baseline Technology [Stafford, 2001]

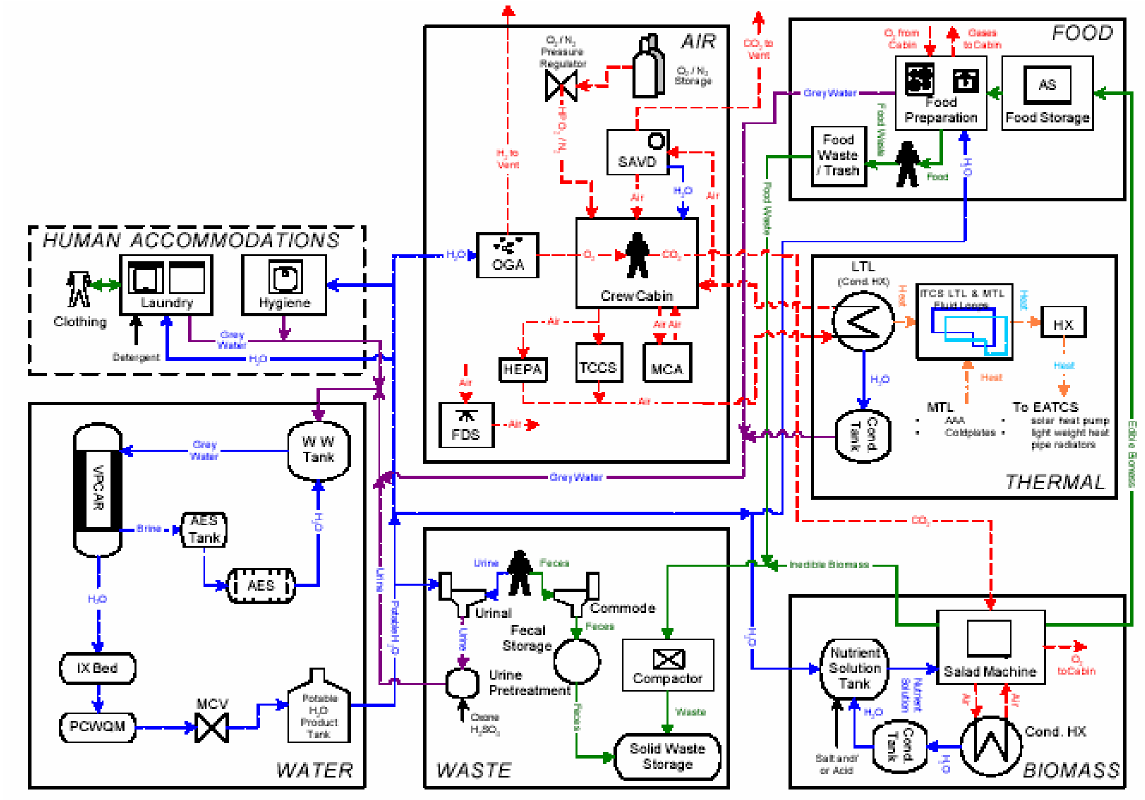

Figure 3. Schematic diagram of Environment Control and Life Support System for a Mars Transfer Vehicle using ALS Technology [Stafford, 2001]

Air Revitalization Subsystem –ISS

The ISS air revitalization subsystem can be broken down into three main components: a CO_2 removal system, an O_2 generation system and a trace contaminant control system (TCCS). The function and operation of various implementations of these subsystems are discussed in detail in previous work as well as in [Lawson, 2003] and will not be explained in detail here. $C0_2$ removal on the ISS is accomplished via a 4-Bed Molecular Sieve (4BMS), electrolysis is used to generate molecular oxygen from oxygen-containing compounds available in the spacecraft (Solid Polymer Water Electrolysis SPWE), and the ISS Baseline TCCS system uses activated carbon to remove non-combustible gases and bacteria filters to remove particulate [Hanford, 2003].

Table 2. ESM value for ISS Air Regeneration System

Option	Mass [kg]	Volume [m³]	Power [kW]	Crew Time [ch/y]	ESM* [kg]
ISS	715	3.63	2.35	8.1	1643.65

Air Revitalization Subsystem –ALS

ALS options for an air revitalization subsystem were introduced in previous work and are as follows [Drysdale, 1999]:

1. 4-Bed Molecular Sieve (4BMS) + Sabatier Carbon Dioxide Removal System (CRS) + Solid Polymer Water Electrolysis (SWPE) Oxygen Generation System (OGS) + Node 3 Advanced TCCS (regenerable sorbent bed)
2. Sabatier CGS + SWPE OGS + Node 3 TCCS

3. Improved Carbon Dioxide Removal Assembly (CDRA) + Sabatier CGS + SPWE OGS + Node 3 TCCS
4. 4BMS + Bosch CRS + SPWE OGS + ISS Baseline TCSS

Table 3. Comparison of ESM values for Air Regeneration System Options
Mission Duration 0.49y

Option	Mass [kg]	Volume [m3]	Power [kW]	Crew Time [ch/y]	ESM* [kg]
1	891	3.00	3.17	8.1	1732.2
2	893	1.60	2.39	8.1	1385.7
3	812	1.90	2.53	8.1	1377.8
4	783	2.99	3.33	8.1	1631.7

Mission Duration 0.73y (8.67months – Hohmann Transfer)

Option	Mass [kg]	Volume [m3]	Power [kW]	Crew Time [ch/y]	ESM* [kg]
1	891	3.00	3.17	8.1	1734.4

Using the ESM metric, various air-regeneration systems can be constructed and compared. With increased mission duration, there is minimal increase in equivalent mass since the only part of the ESM equation where this comes into play is the crewtime component. To minimize mass and power required, Option 2 with a reduced mission duration of 0.49y is the optimal choice.

Food and Biomass Subsystem-ISS
The food subsystem calculations assume that the majority of food is provided from Earth in the form of prepackaged dehydrated entrees. The food storage unit is approximated using values from an ISS freezer/refrigerator component [Hanford, 2002]. The ESM for food is proportional to mission duration since it is a consumable resource.

Table 4 – Comparison of ESM Values for Food System

System Component	ESM* [kg]
Refrigerator/Freezer	905
Prepackaged Food from Earth (D=0.49)	3719.2
Prepackaged Food from Earth (D=0.73)	5540.8

Food and Biomass Subsystem-ALS
For a long duration mission, salad greens and possibly other fresh vegetables such as potatoes could supplement packaged food if a small biomass chamber was included on board the transit vehicle. The equivalent system mass for this component takes into account grow-lamps and ballasts as well as the crops and the required growth space. Since this technology is still under development, a small unit of approximately $5m^2$ is included in the ALS system. This growth space factor multiplies the ESM* baseline value for $1m^2$ of space and also multiplies the power required. The addition of fresh produce to the astronauts' diet would probably be more of a psychological boost than a packaged meal replacement. However, with further study and testing it is possible that a more advanced biomass chamber could be added to a transit vehicle to increase the variety of foods available to the crew.

Table 5 –ESM Values for Biomass System

System Component	ESM* [kg]
Biochamber ($5m^2$ potato, salad greens)	2433.9

Temperature and Humidity Control (THC) System-ISS/ALS
Temperature and humidity must be kept within a nominal range to ensure comfortable working conditions for the crew. Human metabolism as well as cabin equipment must be taken into account when calculating the amount of heat produced and hence the amount of cooling required. Temperature control is usually accomplished by removing heat from the atmosphere using a heat exchange with the excess heat eventually vented to space. Humidity control can be accomplished using a desiccant or phase change process. Ventilation and air circulation are also important concerns when designing the THC System.

The ISS baseline thermal control system details were difficult to find at the component level, so the entire THC system is considered as a unit. This system includes avionic air assemblies to cool equipment, cabin air assemblies for cooling and dehumidification of the crew quarters, and condensate storage/water flow loops for heat transport. Heat exchangers are considered to be part of the external Thermal Control System as opposed to part of the ECLSS.

Table 6. ESM for Thermal Control System

Subsystem	Mass [kg]	Volume [m3]	Power [kW]	Crew Time [ch/y]	ESM* [kg]
Thermal	793	2.31	3.02	0	1734.4

Water Recovery System-ISS
The water recovery system must achieve loop closure approaching 100% to alleviate the need for resupply. No single system for water recovery has been designed to remove all contaminants or treat all types of wastewater that need to be processed. A trade study must therefore compare combinations of technologies that serve to fulfill all treatment requirements including storage, filtration or phase change processes, urine processing, water quality monitoring and disinfect ion.

The ISS Baseline system uses Vapor Compression Distillation in combination with Multifiltration and Volatile Removal Assembly.

Table 7. ESM for ISS Water Recovery System

Option	Mass [kg]	Volume [m3]	Power [kW]	Crew Time [ch/y]	ESM* [kg]
ISS	638	0.5	0.99	8.0	811.8

Water Recovery System-ALS
The ALS water treatment schemes offer a significant improvement in recovery over the ISS baseline model. For the systems outlined below, loop closure approaches 100% and hence the amount of water launched can be significantly reduced.
The five combinations available for comparison according to ESM are as follows:

1. Vapor Compression Distillation (VCD) + Ultrafiltration/Reverse Osmosis (UF/RO) + Aqueous Phase Catalytic Oxidation Subsystem (APCOS)
2. VCD + UF/RO + Milli-Q Post Processor (MilliQ)
3. Biological Water Processor (BWP) + RO + MilliQ
4. BWP + RO + Air Evaporation Subsystem (AES) + MilliQ

Table 8. Comparison of ESM values for ALS Water Recovery System Options

Option	Mass [kg]	Volume [m3]	Power [kW]	Crew Time [ch/y]	ESM* [kg]
1	263	0.5	1.56	8.0	471.0
2	229	0.5	0.72	8.0	386.6
3	141	0.5	0.68	8.0	296.2
4	186	0.5	1.68	8.0	401.2

All simulations are for nominal mission duration of 0.49 years. Varying the mission duration will not have a strong effect on the water recovery system since most of the associated costs are infrastructure related rather than consumable. From the above chart we can see that option 3 is currently the best from an ESM perspective. These values are approximate however since CrewTime and Volume data are not yet available for the technologies under study and therefore are set to be constant across all options.

Waste Processing System-ISS
The waste processing system involves the collection and storage of waste material. For shorter duration missions, storage is usually preferred to in-situ treatment, and hence the ISS waste processing system consists of toilet facilities and a storage tank. Again, since a detailed breakdown of power and mass values for individual components was unavailable, the system is treated and analyzed as a whole.

Table 9. ESM for ISS-Baseline Waste Processing System

Subsystem	Mass [kg]	Volume [m3]	Power [kW]	Crew Time [ch/y]	ESM* [kg]
Waste	149	0.09	0.17	90	228.9

Waste Processing System-ALS
For a Mars transit mission, waste treatment may become an issue due to the long mission duration and the desire to recover as much material as possible from waste. Techniques to consider include incineration, dessication, freezing, heat sterilization and chemical treatment. The ALS system can either use the ISS-Baseline components for storage alone or add to this a treatment component. The only option considered for treatment is super-critical wet oxidization (SCWO), which is a physiochemical process used to treat waste. The process is able to breakdown waste products quite completely, with the resulting products being water vapor, carbon dioxide, nitrogen and salts. Other processes are likely to produce toxins and hence are not as viable for a Mars transit mission where safety and reliability are critical.

Table 10. ESM for ALS Waste Processing System

Subsystem	Mass [kg]	Volume [m3]	Power [kW]	Crew Time [ch/y]	ESM* [kg]
Waste-ISS	149	0.09	0.17	90	228.9
SCWO	200	104	0.5	0.2	820.7

Human Accomodations-ISS
All clothing is assumed to be brought from Earth. The relationship used to calculate clothing mass is 1.5 * CrewSize * Mission Duration (days) [Hanford, 2003].

Human Accomodations-ALS
A smaller amount of clothing may be brought from Earth if there is an aqueous laundry available to the astronauts. Since water recovery is increasingly possible with an ALS system, it can be used to wash clothing and thus reduce the vehicle launch mass. With a laundry system available, clothing mass at launch can be estimated as 0.267 * CrewSize * Mission Duration (days) [Hanford, 2003]. Assume the crew do a load of laundry once a week, so the Crew Time value is multiplied by a factor of 52.

Table 11. ESM for ALS Waste Processing System

Subsystem	Mass [kg]	Volume [m3]	Power [kW]	Crew Time [ch/y]	ESM* [kg]
Laundry	118	0.66	0.31	0.33*52	288.4

Logistics ISS/ALS
Because the ALS systems tend to have greater recycling capabilities, the amount of air, water and clothing that must be launched is lower for the ALS systems. This comparison is shown in Table 12 [Hanford, 2003] with values calculated for a mission duration of 0.49y.

Table 12. Comparison of Logistics for ISS and ALS Systems

ECLSS	O_2 and N_2 [kg]	Water [kg]	Clothing [kg]
ISS-Baseline	294.9	863.87	1609.7
ALS	99.9	11.76	286.5

Summary and Comparison
The ISS-Baseline approach to ECLSS design has the advantage of being technologically proven over a long-term mission duration, however, the mass of this system tends to be higher than the mass of an ECLSS that incorporates advanced regenerative technologies (see Table 13).

The biomass chamber is the single highest contributor to mass for the ALS system, and so the ALS is examined with and without a biomass chamber to show that in general, the ALS mass is less than that of the ISS (less-regenerative) system.

Table 13. Comparison of Logistics for ISS and ALS Systems

ECLSS	ESM*[kg] (min)	ESM*[kg] (nominal)	ESM*[kg] (max)
ISS-Baseline		10339.66	
ALS	8609.258		9138.508
ALS+Biomass	11043.17		11572.42

Comparison cannot be done on the basis of mass alone however, as the ESM in this case does not include power. Power use of the different approaches must also be compared, as must technological development cost and reliability/safety. For a Mars mission scheduled to leave in the next two or three years, the ISS technologies would probably be preferable for the added safety relative to the minor mass improvement in using ALS.

Figure 4 below shows a plot of the ISS and ALS ESM* as mission duration varies. The four ALS plots correspond to minimum ALS value with and without the biomass chamber and maximum ALS value with and without the biochamber respectively. As expected, the tendency to higher masses with longer mission durations is close to linear, and the ECLSS system design favours low mission duration in terms of mass.

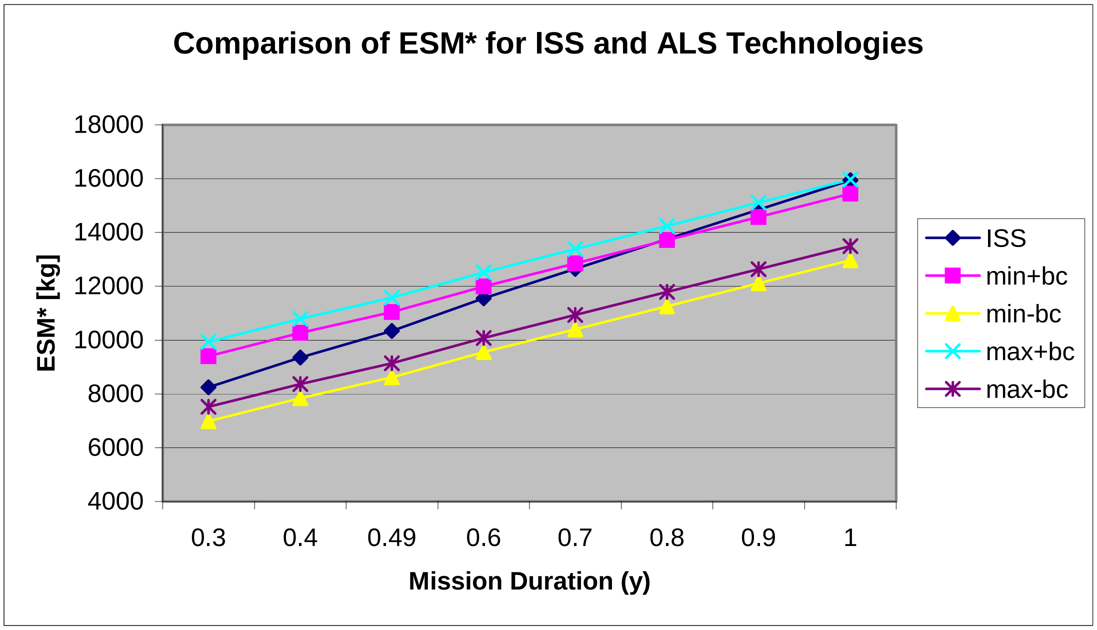

Figure 4. Comparison of ESM* for ISS and ALS Systems

CHALLENGES FOR A MANNED MARS MISSION

A manned flight to Mars also embodies a host of new safety challenges. For one, given the long duration of a mission to Mars, there is an increased chance of failure of dynamic systems such as pumps, valves, and electronics. These failures may be caused by the increases in operating times, on/off cycles, structural flexing, corrosion, and abrasion. Secondly, there is no opportunity to receive equipment or supplies from Earth, underscoring the need for reusable, robust systems. A third challenge is the increasing communications lag between the transportation vehicle and Mission Control as the vehicle approaches Mars. Once in a Mars orbit, this lag will vary from eight to forty minutes depending upon orbital positions. Data rates will also decrease dramatically, on the order of 10^{-6} compared to Earth orbiters. One obvious challenge facing the power system is the reduction in available solar energy as the vehicle approaches Mars. In fact, the power available to solar array will decrease by 50% from the start of the mission.

CREW SIZE

Given the length of the mission and probability that injury or illness will be suffered by crew members at some point in time, it is required for at least two crew members to be trained in performing each mission critical task. Given a review of the literature on this subject, we assume that six astronauts will be adequate to meet the mission requirements.

POWER

For long-duration manned missions it is critical to provide continuous high power while minimizing cost. In this problem set, we considered two energy production devices, solar cells and radioisotope thermal generation (RTG). Solar arrays can provide power at low specific mass and cost, but are subject to eclipse periods and day-night cycles (insignificant for Earth to Mars transfer). Solar arrays also require an energy storage device to meet the continuous power requirement. Radioisotope thermal generation is a compact system that can provide continuous power for a long duration. However, additional mass is required with the RTG system to provide crew safety from nuclear radiation.

Energy storage devices are necessary for non-continuous power generation and to provide an emergency source of power in continuous systems. For human space flight, it necessary to have a low specific energy density while minimizing cost. Chemical batteries and flywheels were deemed to meet cost, reliability, and technology-readiness requirements.

To simply the analysis, a modified version of an existing tool, `PowerDesignResult.m`, developed by Chung, Hilstad, and Kwon of 16.851 is used to design the power subsystem (see Chung, Hilstad, and Kwon problem set #1 for details). The inputs and output to this tool is as follows:

Inputs

- Load power as a function of time in Watts, sampled at constant time steps.
- Source power as a function of time in Watts, sample at constant time steps. This can be supplied either directly as a power profile, or indirectly as constituent data such as the time histories of incident sunlight intensity and angle of solar array with respect to the sun.
- The length of the time step in seconds.
- The initial life fraction of the energy storage device.
- The energy initially stored in the energy storage device, in Joules.

Outputs

- The mass of the power system in kilograms, including the mass of the energy storage system (batteries or flywheel) and power generation system (solar array or RTGs). This mass does not include other components of the system such as power conditioning electronics.
- The cost of the energy storage and power generation systems in millions of dollars.
- The time history of the state of charge of the energy storage system, in Joules.
- The time indices at which the energy storage capacity was insufficient to meet demand, if this has occurred.
- The time history of excess thermal energy that must be dissipated, in Joules.
- The remaining life in the storage system as a fraction of the original lifespan.

The two major modifications made are as follows:

1. Allow power load profile with varying time steps.

2. Add a factor of 10 to the solar array mass.

The second modification accounts for the mass of the solar array structure. Thought this is a very crude estimate, it was necessary so that the comparison between RTG and solar array as power supply options is faire.

PROPULSION SUBSYSTEM SIZING

The propulsion system is divided into two sections: the propellant mass and the system hardware including the engines and tanks. Liquid bipropellant chemical engines are the only type of propulsion system considered in this analysis since the other basic options have either too little thrust or are not fully developed technology. High thrust is important for the direct Earth-Mars transfers (ie. Hohmann and one-tangent transfers) where large initial and final burns provide the ΔV.

Several issues are involved in sizing the propulsion subsystem. For a specific transfer from Earth to Mars, a unique ΔV is required. The propulsion system must therefore be sized to provide enough propellant to enact the ΔV. The propellant mass is calculated using the rocket equation (11) [Wertz, 1999] where ΔV is the combined velocity change required for the initial and final transfer burns, m_p is the propellant mass required, m_{dry} is the unfueled mass of the spacecraft, I_{sp} is the specific impulse of the particular engine and g is the earth gravitational acceleration constant. An additional 25% is added to this value as a margin since this is a first order estimation [Wertz, 1999].

$$m_p = m_{dry}\left\lfloor e^{\Delta V / I_{sp} \cdot g} - 1 \right\rfloor \tag{11}$$

The propellant tank mass is estimated by scaling the propellant mass by the storage tank factor (here using a factor of 10%). Line and pressure regulating equipment mass is considered to be negligible for a first order assumption.

Another issue for the propulsion subsystem is engine thrust. Chemical engines generally have high thrust and relatively short operational lives (100's – 10,000's of seconds). In this operation time the engine must be capable of providing sufficient thrust to carry the vehicle mass to mars. Equation 12 shows the engine acceleration required, a, to provide the necessary ΔV for the engine operation time, t_{eng}. For a conservative margin of safety, the engine operation time, t_{eng}, is reduced by a factor of 4. This ensures that the engine will be able to make the required ΔV burn, but will only use ¼ of the engine life.

$$a = \frac{\Delta V}{t_{eng}} \tag{12}$$

Using Newton's law, the required engine thrust is determined (Equation 13), where m_{wet} is the total vehicle mass including the fuel.

$$T_{eng} = m_{wet} \cdot a \tag{13}$$

For any particular chemical engine, the thrust might be insufficient for the time limited burn, so multiple engines would be required on board. The mass from the engine (or engines) is added into the previous wet mass that includes the other vehicle systems, propellant, and propellant tank mass. This entire process is iterated until the initial wet mass and final wet mass converge to a consistent solution.

During engine burns, an additional ΔV is required to compensate for gravity loss. Since chemical engine burn times are relatively small as compared to the total trajectory time of flight, the burns can be approximated as "instantaneous". This allows the gravity loss ΔV to be neglected.

The propulsion subsystem software module takes in the vehicle dry mass (excluding the propulsion system hardware) and the required ΔV. These propulsion subsystem calculations are performed as described above over a number of liquid bipropellant engines from different manufacturers with various I_{sp}'s and various engine masses (Table 14). The resulting complete vehicle wet mass is output from the module for each type of engine.

Table 14: Liquid Bipropellant Rocket Engines

Engine Name	Propellant Type	Thrust (N)	Isp (s)	Life (s)	Engine Mass (kg)
RL10-A	LO_2/LH_2	73400	446	400	138.35
R4-D	N_2O_4/MMH	489	310	20000	3.76
RS-41	N_2O_4/MMH	11100	312	2000	113.4
R-40A	N_2O_4/MMH	4000	309	25000	7.26

ANALYSIS

The transfer orbits considered ranged from Hohmann transfer to a high energy transfer with a semi-major axis of approximately 2.5 AU. As shown in Figure 5, total ΔV required ranges from 6 km/s to 20 km/s. Note that the final burn required for Mars orbit insertion (MOI) is greater relative to the initial burn required for escaping from Earth orbit (see Figure 6 for the illustration of the trajectories). This suggests that using a small capsule for MOI and jettisoning majority of the transfer vehicle may be desirable. Thus, for the remaining analysis, only the initial burn is considered for propellant calculation.

Also, note that the highest energy transfer requires only about a half of the transfer time compared to the Hohmann transfer. This is an important factor for the consideration of the health of the crew.

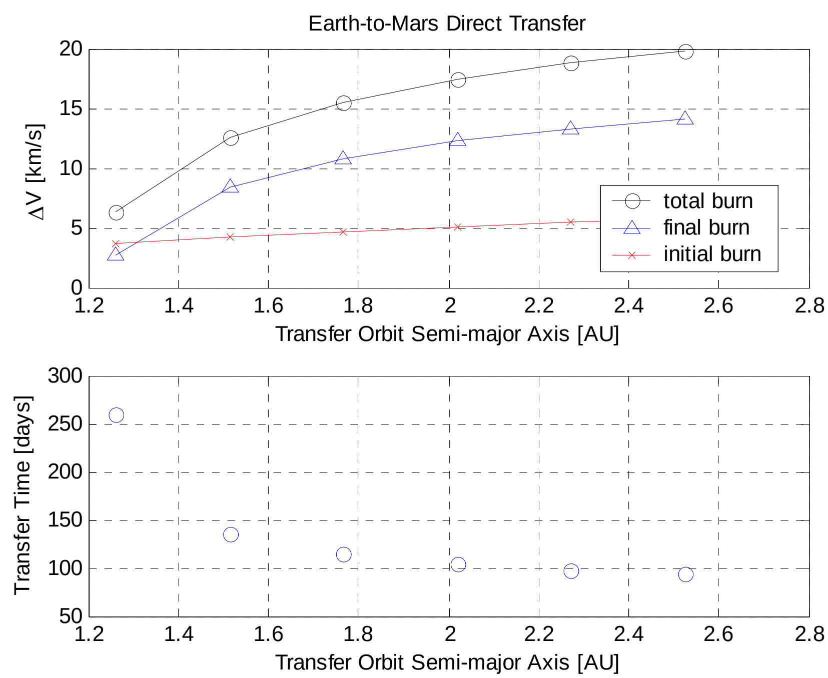

Figure 5. Transfer ΔV and transfer time as a function of transfer orbit semi-major axis.

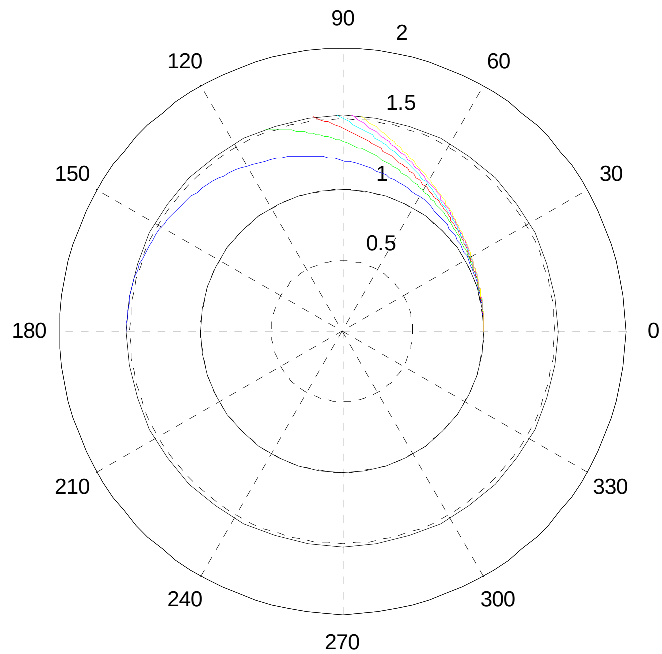

Figure 6. Transfer orbit trajectories ranging from Hohmann transfer to a high energy transfer with 2.5 AU semi-major axis. Note that the radius is in AU.

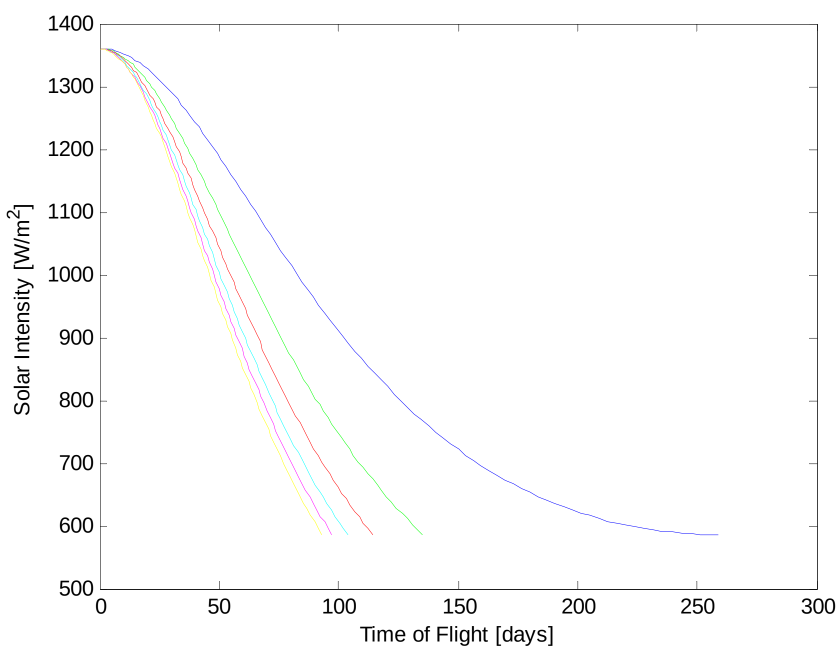

Figure 7. Solar power intensity for transfer orbits ranging from Hohmann transfer to a high energy transfer with 2.5 AU semi-major axis.

Figure 7 illustrates the solar power available as a function of the transfer time. This is important in designing the solar arrays, as it defines the total power available throughout the transfer.

In computing the power subsystem mass, the battery, solar array, and RTG sizes were varied. The battery mass ranged from 100 kg to 500 kg. The solar array size was varied from 50 to 250 m^2. The RTG mass was varied from 56 to 3584 kg. As a reference Cassini spacecraft's RTG's mass was 56 kg. Of all combinations, the design with lower range RTG mass was found to produce insufficient power as required by the life support system. Similarly, lower range solar array size design were found to produce insufficient amount of power. Figure 8 shows a set of feasible designs and the associated design mass as a function of transfer orbit types, where the red points correspond to solar array design and blue corresponds to the RTG design. This design matrix concludes that the best option is the design with the solar array area of 100 m^2 and 100 kg battery. Note that the mass of the power subsystem does not vary with the duration of the mission. This may be due to the fact that all transfer orbits impose the same limiting restriction for solar array, i.e. solar intensity at Mars. Similarly, the RTG design does not vary due to the fact that an RTG life time is must longer than the mission life time.

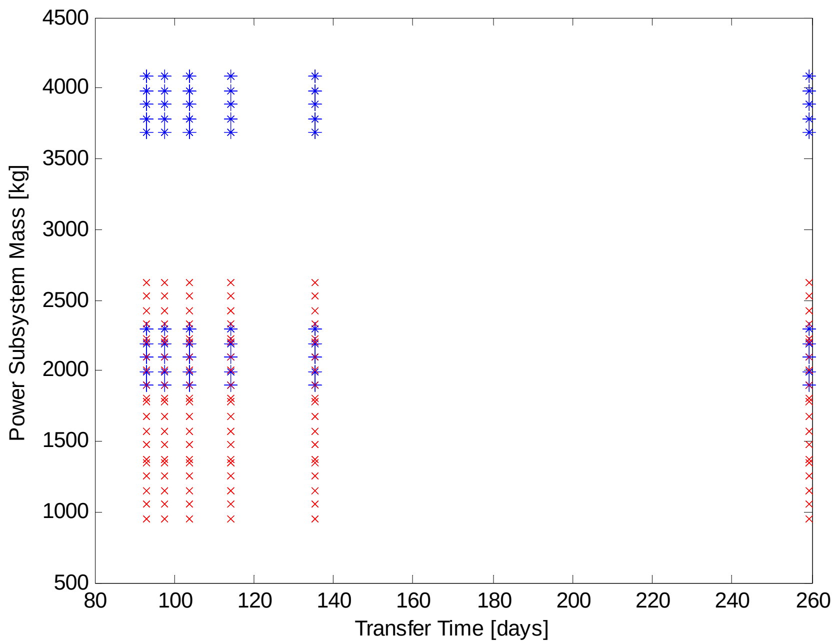

Figure 8. Power subsystem mass for transfer orbits ranging from Hohmann transfer to a high energy transfer with 2.5 AU semi-major axis. Blue signifies the use of RTG and red signifies the use of solar array as a power source.

Finally, Figure 9 illustrates the total vehicle mass that takes the propulsion system into consideration. The design choices within the propulsion system includes propellant type and engine type. In the case of RL10-A system, while the high ISP lowers the propellant mass, the mass of the engine itself increase the total mass close to that of R4-D and R-40A systems (NTO/MMH system). On the other hand, the RS-41 has an unusually high engine mass compared to other NTO/MMH systems; as a result, the total mass is high relative to other systems.

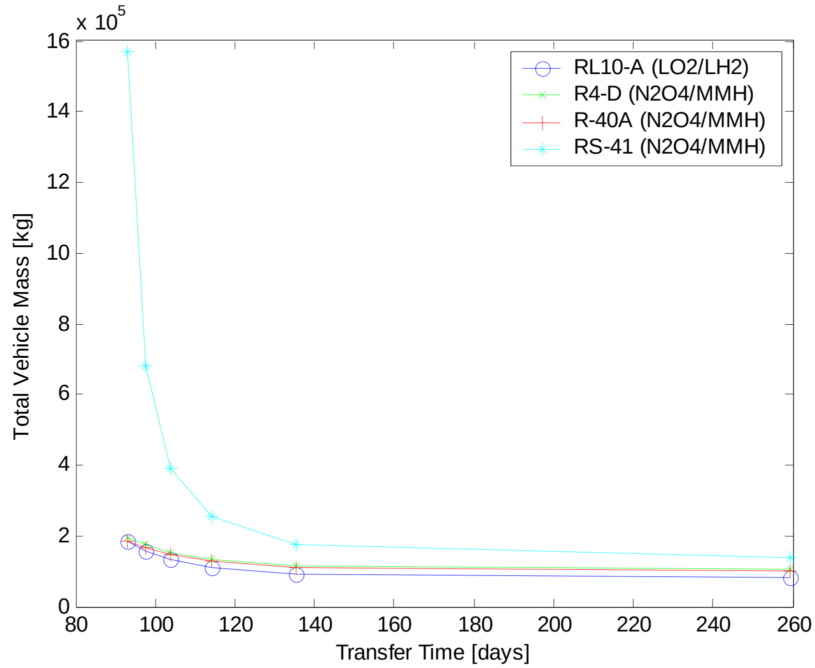

Figure 9. Total vehicle mass for transfer orbits ranging from Hohmann transfer to a high energy transfer with 2.5 AU semi-major axis with respect to various propulsions system.

According to the result, longer the transfer time, lower the total mass as expected. The design choice, however must also take the safety of the crew into consideration, as longer flight duration implies longer time the crew is exposed to hazardous space environment. Finally, note that this analysis can easily be extended to include larger design choices.

REFERENCES

Bate, R., Mueller, D., and White, J., *Fundamentals of Astrodynamics*, Dover Publications, Inc., New York, 1971.

Battin, Richard. *An Introduction to the Mathematics and Methods of Astrodynamics*, Revised Edition. AIAA, Reston, VA. 1999.

Drake, Bret G. *"Technologies for Human Space Exploration: Earth's Neighborhood and Beyond*." AIAA, 2001.

Drysdale, A. E. *Advanced Life Support Systems Modeling and Analysis Project Baseline Values and Assumptions Document*. NASA CTSD-ADV-371. Houston TX, 1999.

Figgess, Alexa., and Vaughan, Andrew., *Interactions between Mission Orbit, Space Environment, and Human Needs*, 16.851 Satellite Engineering Analysis. MIT, 2003.

Hanford, A. J. *Advanced Life Support Baseline Values and Assumptions Document*. NASA CTSD-ADV-484. Houston TX, 2002.

Hanford, A. J. *Advanced Life Support Research and Technology Development Metric – Fiscal Year 2003*. Lockheed Martin Space Operations. Houston TX, 2003.

Kennedy, Kriss J. "*The Vernacular of Space Architecture*." AIAA, 2002.

Lawson, Mike. *Advanced Life Support*. NASA Johnson Space Center. Updated August 2003. http://advlifesupport.jsc.nasa.gov

Levri, Julie. *Advanced Life Support Equivalent System Mass Guidelines Document*. NASA/TM-2003-2112278. Washington DC , 2003.

Oliver, Fernando Abilleira. "*Mars Generations Mission, a Manned Mission to Mars*." AIAA, 2002.

Peterson, Donald H. "*Safety Concepts for a Crewed Mars Mission*." AIAA, 2000.

Stafford, Kristin W. "*Advanced Life Support, Systems Integration, Modeling, and Analysis Reference Missions Document*." Lockheed Martin Space Operations—Crew and Thermal Systems Division. 5 November 2001. http://peer1.nasaprs.com/peer_review/prog/RMD.pdf (30 November 2003).

Wertz, James R. and Larson, Willey J., *Space Mission Analysis and Design*, Third Edition, Microcosm Press, 1999.

Zubrin, Robert. *The Case for Mars*. Touchstone, New York, 1997.

SUBSYSTEM MODULES

```
%###############################################
%## Direct Transfers
%##   (One-Tangent burn ==> Hohmann transfer)
%##
%## Input:
%##  r1 - initial orbit radius [m]
%##  r2 - final orbit radius [m]
%##  a - transfer orbit semi-major axis [m]
%##  mu - central body gravity constant [m^3/s^2]
%##
%## Output:
%##  DeltaV_escape - initial Delta V burn [m/s]
%##  deltaV2 - final Delta V burn [m/s]
%##  T_flight - time of flight [sec]
%##  time - time of flight vector [s]
%##  r_mag - radial position vector over time [m]
%##  nu_time - true anomaly vector over time [rad]
%##
%## Assumes:
%##  - circular initial and final orbits
%##

function [DeltaV_escape, deltaV2, T_flight, time, r_mag, nu_time] = transfer_direct(r1,r2,a,mu)

    e = 1 - r1/a;  %transfer orbit eccentricity

               v1 = sqrt(mu/r1);  %[m/s], initial orbital velocity
               v2 = sqrt(mu/r2);  %[m/s], final orbital velocity

    v1_trans = sqrt(mu*(2/r1 - 1/a));  %[m/s], transfer velocity at initial point
    v2_trans = sqrt(mu*(2/r2 - 1/a));  %[m/s], transfer velocity at final point

    cos_nu = ((a*(1-e^2)/r2-1)/e);  %[rad], true anomaly at final orbit
    nu = acos(cos_nu);  %[rad]

    phi = atan(e*sin(nu))/(1+e*cos_nu);  %[rad], flight path angle at final orbit

    deltaV1 = abs(v1_trans - v1);  %[m/s], initial DeltaV from outside earth SOI to
                                          %begin transfer trajectory.

    deltaV2 = sqrt(v2^2 + v2_trans^2 - 2*v2*v2_trans*cos(phi));  %[m/s], final DeltaV to
                                                       %circularize the orbit once reaching Mars

    T_flight_sun = t_flight(a,e,cos_nu,mu);  %[s], time of flight for sun centered trajectory
phase


    %###############################################
    %## Calculates the orbital transfer trajectory

                      oe(1) = a; %[m], semi-major axis
                      oe(2) = e; %eccentricity
                      oe(3) = 0; %[rad], orbit inclination
```

```
                    oe(4) = 0; %[rad], argument of periapsis
                    oe(5) = 0; %[rad], longitude of ascending node
        i = 1;

        for nu_current=[0:nu/99:nu]
                                              oe(6) = nu_current; %[rad], true anomaly (at epoch)
                                              nu_time(i,1) = nu_current;


        %==========================================================
                                              %= Takes orbital elements of transfer orbit and
calculates
                                              %= the postion for the point current true anomaly.

                                              [r_mag(i,1),v_mag,gamma] = oe2rvg(oe,mu);  %[m],
radius

                                              time(i,1) = t_flight(a,e,cos(nu_current),mu);  %[s],
time from perigee to nu

                                              i=i+1;
        end


    %###############################################
    %## Calculates SOI transfer

        v_inf = deltaV1; %[m/s], v_inf at SOI equals the req DeltaV to initialize the earth/mars
transfer
        r_park = 200000+6378000;  %[m], inital earth parking orbit (200km)
        mu_planet = 3.986e14; %[m^3/s^2], earth mu
        rps = r1;  %[m], earth-to-sun distance

        [DeltaV_escape, T_flt_escape] = transfer_escape(v_inf, r_park, mu_planet, mu, rps);

        T_flight = T_flight_sun + T_flt_escape;  %[s], total time of flight


%###############################################################
%## Planetary Escape Transfer
%## outputs the deltaV for time from perigee to nu.

function [DeltaV_escape, T_flight] = transfer_escape(v_inf, r_park, mu_planet, mu_sun, rps);

  v_park = sqrt(mu_planet/r_park);  %[m/s], circular velocity at parking
  v1 = sqrt(v_inf^2 + 2*mu_planet/r_park);  %[m/s], desired velocity at parking
  DeltaV_escape = v1-v_park;  %[m/s], escape delta V

  E = (v_inf^2)/2;  %energy
  h = r_park*v1;  %angular momentum
  e_trans = sqrt(1 + 2*E*h^2/mu_planet^2);  %eccentricity of the transfer orbit
  p = 2*r_park;
  r_SOI = rps*(mu_planet/mu_sun)^(2/5);  %[m], sphere of influence radius
  nu = acos((p/r_SOI - 1)/e_trans);  %[rad], true anomaly at SOI

  a = 100000000000; %[m], ~inf for hyperbolic orbit
```

```
  e = 1 - r_park/a;  %transfer orbit eccentricity

  T_flight = t_flight(a,e,cos(nu),mu_planet);  %[s], time of flight for earth centered trajectory
phase

%##############################################################
%## time of flight calculation
%## outputs the delta time from perigee to nu.

function [dT] = t_flight(a,e,cos_nu,mu)

  EE = acos( (e+cos_nu) / (1+e*cos_nu) );  %Eccentric anomaly at nu
  dT = sqrt(a^3/mu) * (EE-e*sin(EE));

%## Propulsion mass

function [p_mass, propulsion_system_mass, m_wet, eng_name] = propulsion_module(dV,m_dry)

        %Load Engine Data
        engine_data = eng_read;

        tank_factor = 10/100; % 10 percent - SMAD)
        margin = 25/100; % 25% for a first order estimate - SMAD)

        for i = 1:size(engine_data,1)-1 %loop over all engines we want to look at
    T_actual_req = 1; %initialize
    T_actual = 0; %initialize
    m_dry_tmp = m_dry;

    T = engine_data{i+1,1};
    Isp = engine_data{i+1,2};
    life = engine_data{i+1,3};
    eng_mass = engine_data{i+1,4};
    eng_name{i,1} = [engine_data{i+1,5} ' (' engine_data{i+1,6} ')'];

    if (life/60/60/24 < 1) %day
      accel = dV/(life/2);  %[m/s^2], required engine acceleration
    else
      accel = dV/(24*60*60);  %[m/s^2], required engine acceleration
    end
  %##############################################################
    while (T_actual < T_actual_req) %thrusts are inconsistent, iterate

      T_req = m_dry_tmp*accel;  %[N], required engine thrust
      num_eng = ceil(T_req/T);  %number of engines required to create dV in alotted time.
      total_eng_mass = eng_mass*num_eng; %[kg], mass of engine group

      %# Propellant mass
            p_mass(i,1) = Propellant_Mass(Isp, m_dry+total_eng_mass, dV, margin);

      %# Propellant + Tank + Engine + {margin}
            propulsion_system_mass(i,1) = Propulsion_Syst(total_eng_mass, p_mass(i,1),
tank_factor);

      %# Total Vehicle Mass
      m_wet(i,1) = propulsion_system_mass(i,1)+m_dry;  %[kg], complete vehicle mass


      T_actual_req = m_wet(i,1)*accel;  %[N], actual required engine thrust
      T_actual = num_eng*T;  %[N], actual engine thrust

%       fprintf('Thrust Error = %g\n', (T_actual_req-T_actual))

      if (T_actual < T_actual_req) %thrusts are inconsistent
        m_dry_tmp = m_wet(i,1);
      end
```

```
    end %WHILE
end

%##############################################################

function p_mass = Propellant_Mass(Isp, dry_mass, deltav, margin)
  g = 9.80665;  %[m/s^2], earth gravity
  p_mass = dry_mass * (exp(deltav/(Isp*g)) - 1) * (1 + margin);

function propulsion_system_mass = Propulsion_Syst(engine_mass, prop_mass, tank_factor)
  propulsion_system_mass = prop_mass * (1 + tank_factor) + engine_mass;
       %%assume line/pressure regulating equipment mass is negligible for
       %%a first order assumption




function [r_mag, v_mag, gamma] = oe2rvg(oe, mu)
       [r,v] = oe2rv(oe,mu); %[m, m/s], position and velocity vector at entry interface
       r_mag = sqrt(r(1)^2 + r(2)^2 + r(3)^2); %[m], position vector magnitude
       v_mag = 0; %sqrt(v(1)^2 + v(2)^2 + v(3)^2); %[m/s], velocity vector magnitude
       gamma = 0; %vangle(r,v) - pi/2; %[rad], flight path angle at entry interface


%#####################################################
% CREDIT: Christopher D. Hall
% http://www.aoe.vt.edu/~cdhall/
%
% oe2rv.m Orbital Elements to r,v
%
% [r,v] = oe2rv(oe,mu)
% oe = [a e i Om om nu]
% r,v expressed in IJK frame
%
% a = semi-major axis
% e = eccentricity
% i = inclination
% Om = argument of periapsis
% om = right ascension of the ascending node (longitude of ascending node)
% nu = true anomaly (at epoch). ***(location on orbit)***


function [ri,vi] = oe2rv(oe,mu)
       a=oe(1); e=oe(2); i=oe(3); Om=oe(4); om=oe(5); nu=oe(6);
       p = a*(1-e*e);
       r = p/(1+e*cos(nu));
       rv = [r*cos(nu); r*sin(nu); 0]; % in PQW frame
       vv = sqrt(mu/p)*[-sin(nu); e+cos(nu); 0];
       %
       % now rotate
       %
       cO = cos(Om); sO = sin(Om);
       co = cos(om); so = sin(om);
       ci = cos(i); si = sin(i);
       R = [cO*co-sO*so*ci -cO*so-sO*co*ci sO*si;
       sO*co+cO*so*ci -sO*so+cO*co*ci -cO*si;
       so*si co*si ci];
       ri = (R*rv)';
       vi = (R*vv)';



% Life Support
% returns power in W and mass in kg
% duration is in mission seconds

function [power_day, power_night, life_mass] = Life_Support(duration, crew_size, support_type)
```

```
% kilowatts for nightime activities, ISS model
iss_refrig = .205;
water_recovery = .99;
thermal = 3.02;
waste_treat = .61;
air_revit = 2.35;

iss_power_night = iss_refrig + water_recovery + thermal + waste_treat + air_revit;

% kilowatts for daytime activities, ISS model
crew_comp = .2;
lighting = .1;
rec_equip = .3;
clean_equip = .2;
personal_com = .11;
laundry = .31;

iss_power_day = iss_power_night + crew_comp + lighting + rec_equip + clean_equip + personal_com;

% kilowatts for nightime activities, ALS model
als_refrig = .205;
als_water_recovery = .68;
als_thermal = 3.02;
als_waste_treat = .17;
als_air_revit = 2.53;
als_biomass = 13;

als_power_night = als_refrig + als_water_recovery + als_thermal + als_waste_treat + als_air_revit
+ als_biomass;

% kilowatts for daytime activities, ALS model
als_food_prep = .5;

als_power_day = als_power_night + als_food_prep + crew_comp + lighting + rec_equip + clean_equip
+ personal_com + laundry;

if support_type == 'iss'
    power_day = iss_power_day*1000;
    power_night = iss_power_night*1000;
else
   power_day = als_power_day*1000;
   power_night = als_power_night*1000;
end

% compute life support system mass, adjusted from 730 day mission
base_mass = 20000;
astronaut_mass = 74 * crew_size;
oxygen_mass = 1502 / 730 * crew_size * duration/3600/24;
water_mass = 296 / 730  * crew_size * duration/3600/24; % 90% of used water recycled
food_mass = 3.8 / 2.2 * crew_size * duration/3600/24;
life_mass = base_mass + astronaut_mass + oxygen_mass + water_mass + food_mass;



%## Propulsion mass

function [p_mass, propulsion_system_mass, m_wet, eng_name] = propulsion_module(dV,m_dry)

        %Load Engine Data
        engine_data = eng_read;

        tank_factor = 10/100; % 10 percent - SMAD)
        margin = 5/100; % 5% for a first order estimate)

        for i = 1:size(engine_data,1)-1 %loop over all engines we want to look at
    T_actual_req = 1; %initialize
    T_actual = 0; %initialize
    m_dry_tmp = m_dry;
```

```
    T = engine_data{i+1,1};
    Isp = engine_data{i+1,2};
    life = engine_data{i+1,3};
    eng_mass = engine_data{i+1,4};
    eng_name{i,1} = [engine_data{i+1,5} ' (' engine_data{i+1,6} ')'];

    if (life/60/60/24 < 1) %day
      accel = dV/(life/4);  %[m/s^2], required engine acceleration
    else
      accel = dV/(24*60*60);  %[m/s^2], required engine acceleration
    end

    %################################################################
    while (T_actual < T_actual_req) %thrusts are inconsistent, iterate

      T_req = m_dry_tmp*accel;  %[N], required engine thrust
      num_eng = ceil(T_req/T);  %number of engines required to create dV in alotted time.
      total_eng_mass = eng_mass*num_eng; %[kg], mass of engine group

      %# Propellant mass
           p_mass(i,1) = Propellant_Mass(Isp, m_dry+total_eng_mass, dV, margin);

      %# Propellant + Tank + Engine + {margin}
           propulsion_system_mass(i,1) = Propulsion_Syst(total_eng_mass, p_mass(i,1),
tank_factor);

      %# Total Vehicle Mass
      m_wet(i,1) = propulsion_system_mass(i,1)+m_dry;  %[kg], complete vehicle mass


      T_actual_req = m_wet(i,1)*accel;  %[N], actual required engine thrust
      T_actual = num_eng*T;  %[N], actual engine thrust

%       fprintf('Thrust Error = %g\n', (T_actual_req-T_actual))

      if (T_actual < T_actual_req) %thrusts are inconsistent
        m_dry_tmp = m_wet(i,1);
      end

    end %WHILE #######################################################

end



%#########################################################################
%#########################################################################
%#########################################################################

function p_mass = Propellant_Mass(Isp, dry_mass, deltav, margin)
  g = 9.80665;  %[m/s^2], earth gravity
  p_mass = dry_mass * (exp(deltav/(Isp*g)) - 1) * (1 + margin);

function propulsion_system_mass = Propulsion_Syst(engine_mass, prop_mass, tank_factor)
  propulsion_system_mass = prop_mass * (1 + tank_factor) + engine_mass;
       %%assume line/pressure regulating equipment mass is negligible for
       %%a first order assumption



close all;
clear;
clc;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Generate Transfer Orbits

gravitational_constant_sun =    1.327e20;   %[m^3/s^2], Sun's gravitational constant
distance_sun2earth =            1.49598e11; %[m], average distance from Sun to Earth
```

```
distance_sun2mars =            2.28e11;    %[m], average distance from Sun to Mars

% Types of transfer orbits in terms of the semimajor axis of eliptical
% transfer orbit.
semimajor_axis_hohmann =    (distance_sun2earth + distance_sun2mars)/2;
semimajor_axis_max =        2*semimajor_axis_hohmann;
semimajor_axis_increment =  (semimajor_axis_max - semimajor_axis_hohmann)/5;
semimajor_axis =            semimajor_axis_hohmann:semimajor_axis_increment:semimajor_axis_max;

% Compute delta v, total transfer time, and time profile of position and
% true anomaly of the vehicle for each transer orbit type.
for i = 1:length(semimajor_axis);
    [dv1, dv2, T_flight, t, r_mag, nu_time] = ...
        transfer_direct(distance_sun2earth,...
                        distance_sun2mars,...
                        semimajor_axis(i),...
                        gravitational_constant_sun);
    delta_v1(i) = dv1;
    delta_v2(i) = dv2;
    delta_v_total(i) = dv1 + dv2;
    time_total(i) = T_flight;
    time(i,:) = t';
    distance_sun2veh(i,:) = r_mag';
    true_anomaly(i,:) = nu_time';
end

%%%%%%%%%%
% Plot
figure
subplot(2,1,1), ...
       plot(semimajor_axis/distance_sun2earth, delta_v_total/1000, 'ok-', ...
       semimajor_axis/distance_sun2earth, delta_v2/1000, '^b-', ...
       semimajor_axis/distance_sun2earth, delta_v1/1000, 'xr-');
title('Earth-to-Mars Direct Transfer');
xlabel('Transfer Orbit Semi-major Axis [AU]');
ylabel('\DeltaV [km/s]');
legend('total burn', 'final burn', 'initial burn',0);
grid on;
subplot(2,1,2), plot(semimajor_axis/distance_sun2earth, time_total/60/60/24, 'o');
% title('Earth-to-Mars Direct Transfer');
xlabel('Transfer Orbit Semi-major Axis [AU]');
ylabel('Transfer Time [days]');
grid on;

%%%%%%%%%%
% Plot
figure;
color = ['b' 'g' 'r' 'c' 'm' 'y' ...
         'b' 'g' 'r' 'c' 'm' 'y' ...
         'b' 'g' 'r' 'c' 'm' 'y' ...
         'b' 'g' 'r' 'c' 'm' 'y' ...
         'b' 'g' 'r' 'c' 'm' 'y'];
polar([0:pi/20:2*pi],distance_sun2mars/distance_sun2earth*ones(1,41),'k');
hold on;
polar([0:pi/20:2*pi],ones(1,41),'k');
for i = 1:length(semimajor_axis)
    polar(true_anomaly(i,:),distance_sun2veh(i,:)./distance_sun2earth,color(i));
end
hold off;
axis([-2 2 -2 2]);
axis equal;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Life Support System
for i = 1:length(semimajor_axis)
    [power_life_support_day, power_life_support_night, m] = Life_Support(time_total(i), 6,
'iss');
    mass_life_support(i) = m;
end
```

```
power_life_support_avg = (16*power_life_support_day + 8*power_life_support_night)/24;

figure
plot(time_total/3600/24,mass_life_support,'o-');
xlabel('Transfer Time [days]');
ylabel('Life Support Subsystem Mass [kg]');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Power Subsystem

%take in the design vector
[storage, generation, solar] = power_read_xls('power_design_vector.xls');
storage.RemainingLife = 1.0;
solar.IncidentAngle = 0;

%solar array vector
solar_array_area=50:50:250; %m^2

%battery array vector
battery_mass = 100:100:500; %kg

solar_radiation = 3.826e26; % [W]

% Iterate over all possible power designs
for i = 1:length(semimajor_axis)
    power_load = power_life_support_avg*ones(size(time(i,:)));
    ENV(i).IlluminationIntensity = solar_radiation./(4*pi*distance_sun2veh(i,:).^2);
    for j=1:length(battery_mass)
        storage.Mass = battery_mass(j);
        for k=1:length(solar_array_area)
            solar.SurfaceArea =  solar_array_area(k);
            solar_array_results(i,j,k) = PowerDesignResult({solar}, storage, power_load, ENV(i),
time(i,:));
        end
        for k=1:length(generation);
            rtg_results(i,j,k) = PowerDesignResult({generation(k)}, storage, power_load, ENV(i),
time(i,:));
        end
    end
end

mass_power = inf*ones(size(semimajor_axis));
mass_battery = inf*ones(size(semimajor_axis));
area_solar_array = inf*ones(size(semimajor_axis));
mass_rtg = inf*ones(size(semimajor_axis));
[i_length,j_length,k_solar_length] = size(solar_array_results);
[i_length,j_length,k_rtg_length]   = size(rtg_results);
for i = 1:i_length
    for j = 1:j_length
        for k = 1:k_solar_length
            if (solar_array_results(i,j,k).Mass < mass_power(i))
                mass_power(i) = solar_array_results(i,j,k).Mass;
                power_subsystem_type{i} = 'solar array';
                area_solar_array(i) = solar_array_area(k);
                mass_rtg(i) = nan;
                mass_battery(i) = battery_mass(j);
            end
        end
        for k = 1:k_rtg_length
            if (rtg_results(i,j,k).Mass < mass_power(i))
                mass_power(i) = rtg_results(i,j,k).Mass;
                power_subsystem_type{i} = 'rtg';
                area_solar_array(i) = nan;
                mass_rtg(i) = generation(k).Mass;
                mass_battery(i) = battery_mass(j);
            end
        end
    end
end
```

```
figure
plot(time(1,:)/3600/24,ENV(1).IlluminationIntensity,color(1));
hold on;
for i = 2:length(semimajor_axis)
    plot(time(i,:)/3600/24,ENV(i).IlluminationIntensity,color(i));
end
xlabel('Time of Flight [days]');
ylabel('Solar Intensity [W/m^2]');

figure;
hold on;
for i = 1:i_length
    for j = 1:j_length
        for k = 1:k_solar_length
            plot(time_total/60/60/24,solar_array_results(i,j,k).Mass,'xr');
        end
        for k = 1:k_rtg_length
            plot(time_total/60/60/24,rtg_results(i,j,k).Mass,'*b');
        end
    end
end
xlabel('Transfer Time [days]');
ylabel('Power Subsystem Mass [kg]');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Propulsion Subsystem

% Dry mass is assumed to be structure + life support + power.
mass_dry = mass_life_support + mass_power;
for i = 1:length(semimajor_axis)
    [m_prop, m_sys, m_wet, name] = propulsion_module(delta_v1(i),mass_dry(i));
    mass_propellant{i} = m_prop;
    mass_propulsion_system{i} = m_sys;
    mass_wet{i} = m_wet;
    propulsion_system_type{i} = name;
end

for i = 1:length(time_total)
    for j = 1:length(mass_wet{i})
        mass_wet_new{j}(i) = mass_wet{i}(j);
    end
end

color = ['b' 'g' 'r' 'c' 'm' 'y' 'k'];
shape = ['o' 'x' '+' '*' 's' 'd' '^'];
% figure
% plot(time_total(1)/3600/24*ones(size(mass_wet{1})), mass_wet{1}, [color(1) shape(1) '-']);
% hold on;
% for i = 1:length(time_total)
%     plot(time_total(i)/3600/24*ones(size(mass_wet{i})), mass_wet{i}, [color(i) shape(i) '-']);
% end
% xlabel('Time of Flight [days]');
% ylabel('Total Vehicle Mass [kg]');

figure
plot(time_total/3600/24, mass_wet_new{1}, [color(1) shape(1) '-'],...
     time_total/3600/24, mass_wet_new{2}, [color(2) shape(2) '-'],...
     time_total/3600/24, mass_wet_new{3}, [color(3) shape(3) '-'],...
     time_total/3600/24, mass_wet_new{4}, [color(4) shape(4) '-']);
xlabel('Transfer Time [days]');
ylabel('Total Vehicle Mass [kg]');
legend([propulsion_system_type{1}(1),propulsion_system_type{1}(2),propulsion_system_type{1}(3),pr
opulsion_system_type{1}(4)]);
```