Enhancing the science return of Mars missions via sample preparation, robotic surface exploration and in orbit fuel production

by

Julien-Alexandre Lamamy

Submitted to the Department of Aeronautics and Astronautics
in partial fulfillment of the requirements for the degree of

Master of Science, Aeronautics and Astronautics

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2004

Author .....
Department of Aeronautics and Astronautics
May 21, 2004

Certified by .....
David W. Miller
Associate Professor of Aeronautics and Astronautics
Thesis Supervisor

Accepted by .....
Edward M. Greitzer
H.N. Slater Professor of Aeronautics and Astronautics
Chairman, Department Committee on Graduate Students

Enhancing the science return of Mars missions via sample preparation, robotic surface exploration and in orbit fuel production

by

Julien-Alexandre Lamamy

Submitted to the Department of Aeronautics and Astronautics
on May 21, 2004, in partial fulfillment of the
requirements for the degree of
Master of Science, Aeronautics and Astronautics

Abstract

The future of Mars exploration is challenging from multiple points of view. To enhance their science return, future surface probes will most likely be equipped with complex Sample Preparation And Transfer (SPAT) facilities. Future rovers will need to be able to perform longer traverses and delicate sample acquisition operations. Mars return missions would benefit from a new propulsion system, with better fuel and travel time efficiencies than chemical and electric propulsions, respectively.

A model was developed that optimizes SPAT facilities in terms of productivity and system mass. The SPAT model especially investigates two trade-offs: shared versus specific preparation, and warm versus cold redundancy for SPAT elements. A Mars Surface Exploration (MSE) framework was created to help designers perform preliminary studies on rover missions. MSE applies multidisciplinary design optimization techniques for the analysis of design trade-offs relevant to the rover design community. The Propellant Production In Mars Orbit (PPIMO) is presented as a promising solution for performing return travels to Mars. PPIMO uses the concept of regenerative aerobraking to produce fuel in-situ.

The SPAT model shows that warm redundancy improves productivity by both reducing risk and removing sample throughput bottlenecks. A method is presented for determining the economy of scale the shared preparation architecture must exhibit for it to be competitive in comparison to the distributed architecture. MSE is used to budget the future development costs of rover autonomy, in addition to assessing: the benefits of oversized suspensions, the practicality of solar versus nuclear power for future missions, and the advantages of multi-rover missions. When compared to chemical and electric propulsions, PPIMO propulsion shows a better performance in terms of transportation ratio for payloads larger than 1000 kilograms.

Thesis Supervisor: David W. Miller
Title: Associate Professor of Aeronautics and Astronautics

Acknowledgments

I would like to thank my advisor David W. Miller for his guidance and support throughout my experience at MIT. My family must be acknowledged for their constant support and love. I would like to particularly acknowledge Alice Hunter and all the Mellens for their unwavering support and sacrifices which made this thesis possible. Members of the Space Systems Laboratory need to be thanked for their suggestions, in particular Dr. Raymond Sedwick, Prof Manuel Martinez-Sanchez, Prof. Olivier L. de Weck, Mathieu Chaize, and Dr Cyrus Jilla. The members of the Space Systems Engineering 2003 class, and especially my teammate Mark Hilstad, need to be acknowledged for their contribution to this thesis. The following individuals at the Jet Propulsion Laboratory (JPL) were helpful in answering questions regarding rover design and systems engineering: Charles Whetsel, Allen Chen, Jaret Matthews, Alberto Elfes, and Jim Chase. This research has been funded in part by NASA JPL.

Contents

List of Figures

List of Tables

Nomenclature

Abbreviations

APXS	Alpha Proton X-Ray Spectrometer
CNES	Centre National d'Etudes Spatiales, French space agency
DLR	Deutsches Zentrum für Luft- und Raumfahrt (German aerospace center)
DSN	Deep Space Network
DS1	Deep Space 1
DTE	Direct to Earth
ECS	Engineering for Complex Systems
ESA	European Space Agency
EVA	Evolved Gas Analysis
FIDO	Field Integrated Design and Operations
GCM	Global Circulation Model
GCMS	Gas Chromatograph Mass Spectrometer
HGA	High Gain Antenna
HMO	High Mars Orbit
IR	Infra-Red
ISAS	Institute of Space and Astronautical Science
JPL	Jet Propulsion Laboratory
LEO	Low Earth Orbit
LMO	Low Mars Orbit
MAE	Material Adherence Experiment on Sojourner
MAUA	Multi Attribute Utility Analysis
MEP	American Mars Exploration Program

MER	Mars Exploration Rovers (2003 JPL)
MMRTG	Multi-Mission Radioisotope Thermoelectric Generator
MSDO	Multidisciplinary System Design Optimization
MSL	Mars Science Laboratory (2009 JPL)
MSR	Mars Sample Return
MSSL-UCL	Mullard Space Science Laboratory - University College London
MUM	Mars Underground Mole
NASA	National Aeronautics and Space Administration
PAW	Position Adjustable Workbench (on board Beagle 2 mission)
PPIMO	Propellant Production In Mars Orbit
PSIG	Project Science Integration Group
RAT	Rock Abrasion Tool (on board MER mission)
RTG	Radioisotope Thermoelectric Generator
SD2	Sample Drilling and Distribution
SMART	Small Missions for Advanced Research in Technology
SPAD	Sample Preparation And Distribution
SPAT	Sample Preparation And Transfer
SRG	Stirling Radioisotope Generator
TRL	Technology Readiness Level
UHF	Ultra High Frequency
USA	United States of America
USSR	Union of Soviet Socialist Republics
WAE	Wheel Abrasion Experiment on Sojourner
WEB	Warm Electronics Box

Units

a.u.	astronomical unit
cm	centimeter
kbps	kilo bits per second
kg	kilogram

kJ	kilo Joule
km	kilometer
m	meter
m/s	meters per second
mol	mole
s	second
sol	A sol is the name of a day on Mars, its duration is 24 hours 39 minutes.

Symbols

$E_{s/c}$	Energy of the spacecraft
Isp	Engine's specific impulse
M_{Dry}	Spacecraft's dry mass
M_{Wet}	Spacecraft's wet mass
$M_{s/c}$	Mass of the spacecraft
$\dot{M}_{s/c}$	Mass flow of the spacecraft's propulsion system
$P_{s/c}$	Power of the spacecraft
$T_{s/c}$	Spacecraft's thrust
V_r	Radial velocity of the spacecraft
V_θ	Orthoradial velocity of the spacecraft
f_s	Factor of safety
g	Gravitational acceleration on Earth
g_{Mars}	Gravitational acceleration on Mars
h	Angular momentum of the spacecraft
s/c	Spacecraft
$\vec{r}$	Position vector of the spacecraft
β	Transformation facctor
η	Efficiency
μ	Gravitational parameter

Chapter 1

Introduction

1.1 Mars exploration

Since the end of January 2004, Mars has been under the attention of four major international missions, which include three orbiters[1] and a pair of rovers[2]. Never before has an extra-terrestrial planet been the focus of such scrutiny or approached in so many different ways at the same time. However, January 2004 was also marked by consecutive failures of the Japanese Nozomi and British Beagle 2 missions. Mars is, indeed, a terrain of scientific excitement, but it is also a graveyard to many missions.

1.1.1 Objectives of the Mars Exploration Program

The worldwide excitement about Mars has its source in the evidence of past water on its surface. Indeed, this fact suggests that Mars could have been the cradle for early life forms such as those found on the ALH84001[3] meteorite. Water is also the gist of NASA's Mars Exploration Program (MEP) strategy. The exploration of Mars is guided by the adage *Follow the water* to discover life, to understand Mars' genesis and to eventually send humans for exploration. The objectives of the MEP are specifically

[1] Mars Express (ESA), Odyssey (NASA) and Global Surveyor (NASA)

[2] The Mars Exploration Rovers twins (NASA)

[3] ALH84001 was discovered in 1984 by Dr. David McKay and his co-workers. They claimed in 1996 to have found evidence of dead, fossil bacteria and chemical traces that might have come from bacteria. The meteorite is believed to be of Martian origins.

to [34]:

1. Determine if *life* ever arose on Mars

2. Determine the *climate* on Mars

3. Characterize the *geology* of Mars: evolution of the surface and interior of Mars.

4. Prepare for *human exploration* of Mars

These goals pertain to different kinds of exploration and require adapted approaches. While Mars' geology and climate can first be investigated by tele-observation from orbiters, the search for life and the preparation for human exploration require landed and return missions. The future of Mars' exploration is, therefore, a balance of remote and contact science that eventually should lead to the analysis of Mars samples on Earth thanks to a Mars Sample Return mission (MSR) or human exploration of Mars (Figure 1-1).

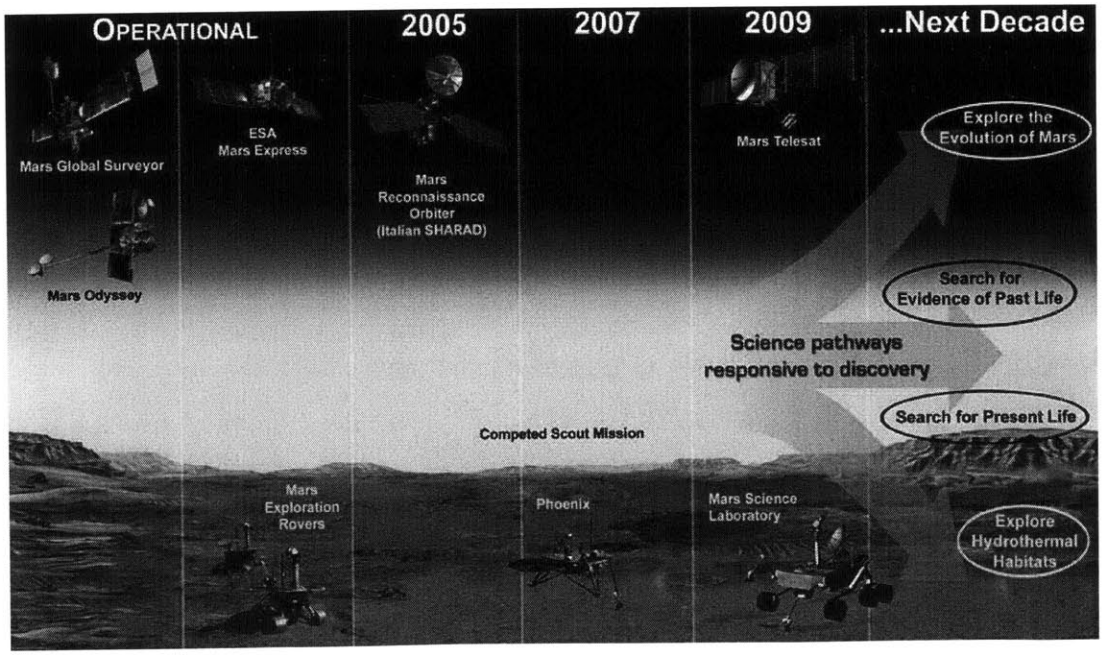


Figure 1-1: The planning of Mars Exploration [29]

1.1.2 Engineering challenges

The MSR mission is actually scheduled for no sooner than 2014. This mission is relevant of the technological challenges that will need to be surpassed to ensure the success of future planetary missions. The challenges concern multiple disciplines. The sample collection part of the mission requires a surface probe, most likely a rover, able to explore the surface of Mars, to collect samples, and to assess their science relevance. Getting the samples back to Earth represents an even harder engineering challenge. To return to Earth, a spacecraft must escape the gravitational attraction of Mars; the amount of fuel for that operation is proportional to the mass of the vehicle. One solution is to load the spacecraft on Earth with the amount of fuel it will need to come back. The payload mass that must be initially lifted-off the Earth is, as a consequence, the mass of the probe plus that of the fuel. This solution is too expensive with the current traditional chemical systems. An alternative solution is to send a probe to Mars that would produce the fuel it needs to come back from local resources (in-situ).

1.2 Motivation

This thesis presents three studies that address the disciplines of in-situ preparation and analysis of samples, robotic surface exploration, and regenerative aerobraking with the common goal of enhancing future Mars missions. The motivations for each study are presented in the subsequent paragraphs.

Sample preparation (Chapter 3)
David Beaty et al. have identified the in-situ sample preparation as a key technology for the success of missions to come [17]:

> Our future exploration of Mars will involve a mixture of orbiters, landers, and sample return missions. For the landers and sample return missions, it has been argued that one of the most important factors limiting the relative

effectiveness of in situ investigations (compared to returning samples to Earth) is the level of capability for in situ sample preparation.

This thesis presents a systems engineering study of in situ sample preparation tools, such as the Sample Preparation And Distribution (SPAD) device that will be used on the Mars Science Laboratory rover in 2009. The study uses multidisciplinary optimization techniques along with Markov reliability theory to optimize sample preparation systems with respect to mass, to sample throughput, and to operational risk.

Robotic surface exploration (Chapter 4 and 5)

In 1997, the NASA Mars Pathfinder mission successfully delivered a lander and a rover on Mars' surface. The mission was mainly driven by engineering motives; it demonstrated that the use of airbags is an appropriate landing method, and that rovers are suited for the exploration of Mars' surface. The Mars Exploration Rovers arrived on the planet in January 2004 with more scientific ambition. The next generation of rover is the Mars Science Laboratory which will be launched in 2009. For the design of each new mission, new trade-offs arise. The fundamental engineering design questions are: which capabilities of the system should be improved and how these improvements should be practically realized? For example, the Mars Science Laboratory is the first Mars rover mission to consider the use of nuclear power as opposed to the traditional use of solar power.

Several key aspects of the design and performance of Mars rover missions are actually driven by the relative positions between the Sun, Earth and Mars. Three of these aspects are introduced in this paragraph to illustrate the challenges of rover mission design. First, the celestial positions of the bodies impacts the schedule of missions to Mars. Indeed, celestial mechanics dictate the traveling time of a journey to Mars. Following a Hohmann transfer, a spacecraft cruises for at least six months. Additionally, launch opportunities happen every twenty six months with a launch window of about ten days. In comparison, optimum launch opportunities to the Moon happen every lunar day, namely every thirty Earth days. Second, because Mars is further from the Sun than the Earth, the solar power available on Mars is

less than that on the Earth. On Mars, the solar flux is in the order of 609 watts per meter square, compared to 1370 watts per meter square on Earth. Consequently, solar power may not be a viable solution for the operation of large vehicles such as MSL, which is likely to be as large as a minivan. The use of nuclear power is in that case an advantageous alternative (Section 5.4). Third, the relative distance between the Earth and Mars affects communication effectiveness. For radio frequencies, the round trip delay between the two bodies is about ten to forty minutes, whereas between the Earth and the Moon it is on the order of three seconds. On top of that, the availability of the Deep Space Network (DSN), which listens to messages sent by missions exploring the solar system, is limited to at most a couple of hours per day for each mission[4]. Hence, it is not possible to command or navigate Mars rovers in real time from a ground station on Earth. Therefore, Martian probes must be equipped with at least a minimum level of autonomous navigational and maintenance capabilities. A balance still needs to be struck between the operations that are managed by the ground station and those that are managed by the on-board autonomy (Section 5.6).

The challenges of missions to Mars are obvious when compared to lunar missions. The contrast between the design features of Lunokhod (first rover on the Moon and any extra-terrestrial body) and Sojourner (first rover on Mars) is especially striking (Table 1.1). Notice how the payload capacity of missions to Mars is far less than that of missions to the Moon. The payload mass is a major performance metric for planetary missions (Section 5.2.2).

The choice of power source and the cost-benefits of autonomy are examples of major design trade-offs that are handled by the Mars Surface Exploration (MSE) framework presented in this thesis. MSE is a system engineering tool that uses multidisciplinary system design optimization techniques for the optimization of Mars rover missions. MSE helps designers identify the rover architectures that best meet their objectives, such as maximizing science return, or minimizing mass and cost.

[4]The situation is actually critical starting on summer 2004 because Mars with its four operational missions and Saturn with its major Cassini mission are in the same vicinity in the Earth sky

Table 1.1: Comparison of Lunokhod and Sojouner missions ([24],[44])

		Lunokhod 1	Sojouner
Year & Country	Units	1970, USSR	1996, USA
Environment		Moon	Mars
Mission duration	Earth days	330	83
Mass	kg	756	10.5
Wheel size	m	0.51	0.13
Maximum velocity	m/s	0.56	0.01
Total traverse	km	10.5	0.1
Science return		200 TV panoramas	120 MAE, 9 WAE
		20000 TV pictures	500 images captured
		500 soil tests	9 rocks, 7 soil analysis by APXS

Regenerative aerobraking (Chapter 6)

In 1970, six weeks prior to the Lunokhod success, the Union of Soviet Socialist Republics (USSR) Space Program manages the first sample return mission. While the Luna 16 mission managed to bring back one hundred grams of lunar soil, the launch date of the first sample return mission to Mars remains up to now elusive. One of the chief engineering challenges of a return mission to Mars is to find an adequate propulsion system. Chemical propulsion is fast but not fuel efficient. Electric propulsion is slow but very fuel efficient. This thesis presents an innovative type of propulsion that uses the concept of regenerative aerobraking. The Propellant Production In Mars Orbit (PPIMO) system uses the heat generated during the aerocapture and aerobraking phases of a spacecraft around Mars to help the chemical process of fuel production from the carbon dioxide present in the Martian atmosphere. The PPIMO solution is an advantageous compromise since it exhibits better fuel efficiency than traditional chemical propulsion and better time efficiency than electric propulsion.

1.3 Research Context

This thesis gathers the work achieved in three research areas related to the exploration of Mars: sample preparation, robotic surface exploration (MSE), and regenerative aerobraking. Most of the research effort has been engaged in the development of the MSE framework. The other two research studies are preliminary analyses that were not given the time to come to full maturity. Still, they are presented in this thesis because the author believes they provide interesting contributions. They are sound foundations for possible future work.

1.4 Thesis Overview

This thesis consists of seven chapters. Chapter 2 introduces the mathematical notions that are used throughout this thesis. These notions pertain to two major fields: the Multidisciplinary System Design Optimization; and the reliability analysis based on Markov state models.

Chapter 3 presents the work achieved on the optimization of in situ sample preparation tools. This chapter chiefly develops the methods and rationale used to model and optimize sample preparation systems with respect to mass, sample throughput and operational risk.

Chapter 4 is the first of two chapters related to the Mars Surface Exploration (MSE) framework. The purpose of this first chapter is to present the architecture of the tool, as well as the approaches it uses to model rover missions. Specific sections focus on the description of the modeling assumptions and design for each rover subsystem.

Chapter 5 demonstrates the analysis capabilities of MSE. It shows how MSE can be used to examine four trade-offs relevant to rover mission design. These trade-offs are:

1. What are the benefits of oversizing a rover's suspension?

2. Is solar power a viable option for the MSL?

3. How big should the MSL rover be and how long should the mission last?

4. What are the cost-benefits of autonomy?

Chapter 6 is related to the study of propulsion methods for round-trip journeys to Mars. It introduces the innovative concept of Propellant Production In Mars Orbit (PPIMO) by spacecraft orbiting around Mars. The PPIMO solution is then compared to the traditional chemical propulsion system and various electric propulsion technologies.

Chapter 7 summarizes the findings, identifies the contributions and limits of the analysis, and sets recommendations for future studies in the three research areas addressed in this thesis.

Chapter 2

Optimization and reliability analysis fundamentals

This chapter introduces the fundamental design theories that are repeatedly used throughout this thesis. First, it presents the notions of design vector, full-factorial exploration, and Pareto front within the context of the Multidisciplinary System Design Optimization; these notions are applied in both Chapters 3 and 4. It then provides an overview of the Markov state models used for reliability analysis in Chapter 3.

2.1 Multidisciplinary System Design Optimization (MSDO)

2.1.1 Definition

Multidisciplinary System Design Optimization (MSDO) is a methodology for the design of complex engineering systems requiring analysis that accounts for interactions among various disciplines [23].

The Mars Surface Exploration study performed in Chapter 4 is an example of the MSDO methodology applied to the design of planetary rovers. This study is *multidisciplinary* because it addresses all a rover's subsystems, each of which represents at least one discipline in the sense that it uses specific governing equations. The study

models a *system* (the rover) which has a function (the exploration of Mars) whose performance depends on the interactions of its subsystems with each other and with the environment. It is a *design* study in the sense that it conceives a system that is subsequently implemented and operated for beneficial purposes. It *optimizes* the system by uncovering a broad spectrum of alternative solutions and assessing which one best achieves the system's objective function. The objective function comprises measures such as system behavior, resource utilization, and risk.

2.1.2 The need for MSDO in systems engineering

Within NASA, the management of major systems is performed by following the successive stages of the *project life cycle* [48]:

1. Pre-phase A – Advanced studies: find a suitable project
2. Phase A – Preliminary analysis: make sure the project is worthwhile
3. Phase B – Definition: define the project and establish a preliminary design
4. Phase C – Design: complete the system design
5. Phase D – Development: build, integrate, and verify the system, and prepare for operations
6. Phase E – Operations: operate the system and dispose of it properly

Regarding the pre-phase A in particular, the suitable project selected at the end of that phase is the result of the exploration of a large trade space (defined in Section 2.1.3). The purpose of that initial phase is to devise a broad spectrum of ideas and alternative for missions from which new projects can be selected [48]. The MSDO methodology is quite well adapted to this task because it permits the generation, analysis and comparison of a vast array of architectures that are possible solutions to a given system. Because MSDO is meant to cover broad trade spaces, it ensures that the true optimal design is found at the conclusion of the process. In contrast, a point design method, which relies on the modification of existing designs, is a localized approach that achieves feasibility rather than optimality.

2.1.3 Formulation of the optimization problem

All the examples of this section are based on the Mars rover mission study presented in Chapter 4. Furthermore, the Multidisciplinary System Design Optimization class, taught at the Massachusetts Institute of Technology by Olivier de Weck and Karen Wilcox, is the reference for most of the notions introduced in this section. Given a mission, the set of all architectures that are possible candidates for that mission is called the *trade space.* A system is defined as a set of interrelated components which interact with one another in an organized fashion and toward a common purpose (to achieve the mission) [48]. The first important aspect of this definition is that a mission is conceived with explicit goals specified prior to the design process. These are the objectives upon which the optimization step of the MSDO is based. For example, two goals of a rover mission are to maximize science return and minimize cost. The system's goals are gathered in the objective vector J:

$$\text{J}=\begin{bmatrix} \text{Science return} \\ \text{Cost} \end{bmatrix} \begin{matrix} \text{maximize} \\ \text{minimize} \end{matrix} \tag{2.1}$$

Often to simplify the selection of architectures, the objectives are combined into one *utility function.* A multi-objective optimization problem is, thereby, transformed into a single-objective problem, the maximization of the utility. The utility is a weighting of all the objectives based on the mission designers' preferences. Furthermore, these weights can be changed to explore the sensitivity of the design to the designers' preferences. For example, the above two-dimensional objective vector J, can be combined into the maximization of a function-per-cost utility function, U [37]:

$$U = \frac{R}{C} \tag{2.2}$$

where R is the science return and C is the cost. This utility assumes equal weighting for both objectives. Additionally, it is consistent with Equation 2.1 since the maximization of the science return and the minimization of the cost implies the maximization of the function-per-cost utility.

The second important aspect of the system's definition is that a system is an association of components, which encompasses hardware and software, but also people and organizations. Mobility, and power, as well as the operating environment are examples of a rover's components. The noteworthy point is that the trade space of architectures is generated by modifying the characteristics of these components. These characteristics are represented by what are called *design variables*, which are selected by designers to examine the trade-offs that concerned them. Design variables are also gathered under a vectorial form called the *design vector*. For example, the Mars Surface Exploration tool represents a rover system with eight design variables. The design vector illustrating the Mars Exploration Rover (MER) is:

$$V_{\text{Design}}^{\text{MER}} = \begin{bmatrix} 90 \text{ sols} & \text{lifetime} \\ 0.25 \text{ meters} & \text{wheel size} \\ 1 & \text{number of computers} \\ \text{solar} & \text{power source} \\ \text{DTE and X} - \text{UHF} & \text{communication type} \\ \text{A1} & \text{long distance autonomy} \\ \text{A1} & \text{short distance autonomy} \\ \text{A1} & \text{acquisition autonomy} \end{bmatrix} \tag{2.3}$$

One particular value of a design variable is called a *level*, for example, 0.25 meters is one level of the wheel size variable. The vector V_{Design}^{MER} is a particular set of design variable levels. Another set would define another architecture. The number of possible architectures, namely the size of the trade space, is:

$$S_{trade} = \prod_i l_i \tag{2.4}$$

where l_i is the number of levels of the i^{th} design variable. It is then obvious that the number of design variables, and their number of levels are critical decisions systems engineer must make. The larger the trade space, the longer it takes to find the optimal architectures.

The range of values of a design variable is also limited by the constraints inherent to the system. These can be equality or inequality constraints. To summarize, an optimization problem has the following form:

$$\begin{aligned} &\text{optimize } \mathrm{J}\left(\mathrm{V}_{\mathrm{Design}}\right) && (2.5)\\ &\text{while satisfying } \mathrm{g}\left(\mathrm{V}_{\mathrm{Design}}\right) \leq 0 \\ &\text{and } \mathrm{h}\left(\mathrm{V}_{\mathrm{Design}}\right) = 0 \end{aligned}$$

MSDO mathematically traces a path in the design space from some initial design V_0 toward optimal designs with respect to the objective J. The next section presents several optimization schemes which differ in their ability to find the quickest path.

2.1.4 Trade space exploration methods

The difficulty of the optimization process is to go through the assessment of architectures as efficiently as possible, in other words to find the quickest path from one initial design to the optimal designs. The subsequent paragraphs present some of the ways to explore a trade space. For each of these methods, there is a trade-off between the confidence that the global optimum is found and the time spent uncovering it. A global optimum is an optimum of the whole trade space, as opposed to a local optimum which is an optimum for a subset of that space (Figure 2-1).

Full factorial This method simply goes through all the possible combinations of design variable levels and, thereby, generates all possible architectures. Once all the architectures are assessed, they are ranked based on their performance and the optimal designs are identified. It is an exhaustive search for which the computational time requirement is proportional to the trade space size (Equation 2.4). However, the method covers the whole trade space and, therefore, there is no doubt that the optimum it provides is the true optimum.

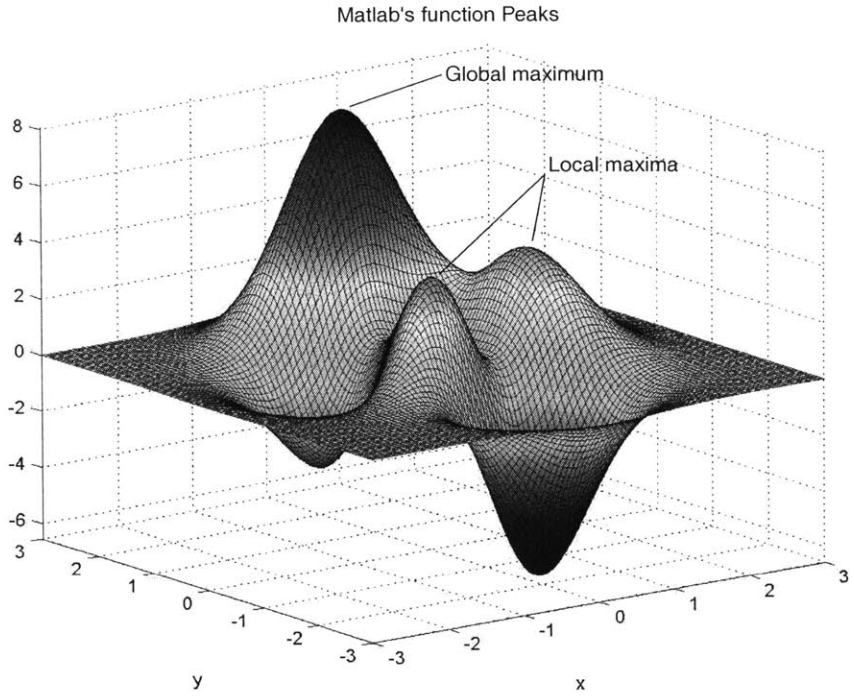


Figure 2-1: Illustration of the global maximum as opposed to local maxima

Partial search Because, for large trade spaces of complex systems, full-factorial is too computationally expensive, other methods have been conceived that explore the trade space *intelligently*. Each of these methods uses a principle that orients the search toward designs with better performance [23]. Gradient search methods calculate the rate of increase in the objective function around a reference point design in order to choose in which direction to move. Heuristic methods such as simulated annealing and genetic algorithms have a probabilistic nature and are based on analogies with the cooling of a material to a state of minimum energy and the mutations of a population, respectively. Heuristic methods are preferred for the exploration of complex shaped trade spaces for they tend not to get trapped in local optima, as opposed to gradient search methods.

The fundamental point is that such partial search methods are not guaranteed to find the global optimum for non-convex trade spaces (Figure 2-1). A more detailed comparison of these methods is presented in Cyrus Jilla's thesis [37]. Notice also that

partial search methods require an explicit utility function prior to the optimization process. Now, when a system involves several stakeholders with conflicting interests, it is often difficult to find a consensus on the weighting of the optimization objectives. For example, the goal of the science community is to maximize the science return of a mission, whereas the tax-payers are first concerned with its cost. In such cases, the full-factorial method is appropriate because it does not require a utility function to explore the trade space. The designers can explore the trade space, save the resulting database of architectures, and apply to that database as many utility functions as wished to select their optimal designs. Partial search methods would require one exploration for each utility function tried.

2.1.5 Analysis of the results

As mentioned above, the analysis of a trade space along several objectives is delicate. Multi-objective trade spaces have more than one optimal design. In this situation the notion of *Pareto front* is enlightening. A Pareto front is defined as the boundary connecting those architectures for which there is no other architecture that is more optimal with respect to *all* objectives. Points that do not belong to the Pareto front are called *dominated* architectures. Figure 2-2 is an example where the optimization problem has two objectives: minimizing cost and maximizing science return. The figure shows all the architectures in a plot with science return and cost axes; the *Utopia point* is then in the upper-left of the plot, region of highest return for the smallest cost. The Pareto front is represented by a dashed line. For any point that belongs to that front there is no other design that has a smaller cost and a larger return. All the points that are below the line are *dominated* designs. The point Design 2 in the figure, for example, is dominated by the Design 1 since Design 1 returns more science for less cost than design 2.

The knowledge of the Pareto front's shape can facilitate multi-criteria decision making by allowing the designer to put off assigning weights and preferences to the individual objectives. Decision analysis can therefore begin before preferences are known.

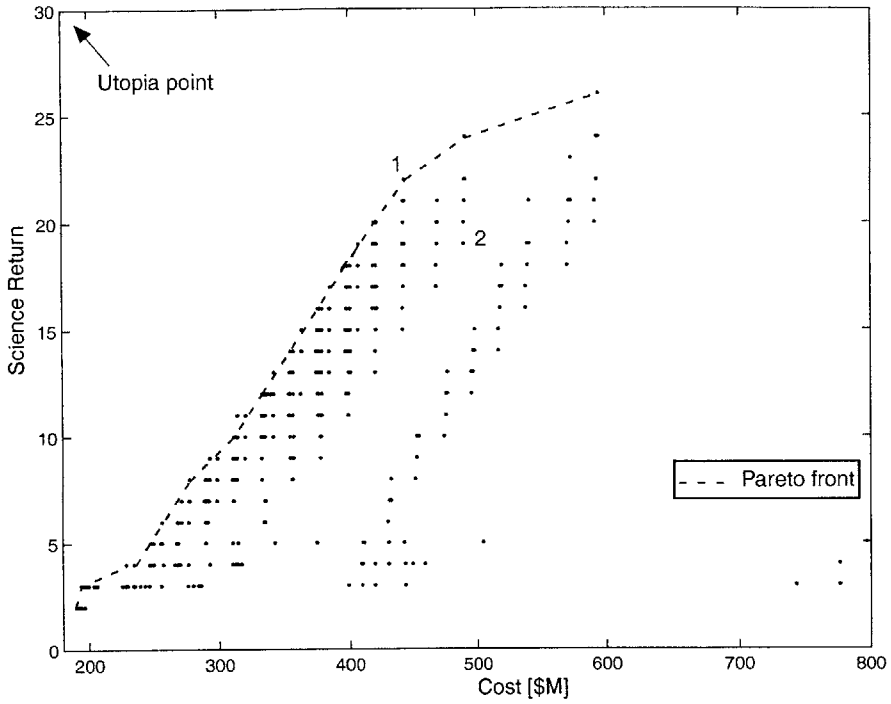


Figure 2-2: Illustration of the Pareto front

2.1.6 Applications

Besides the applications of this thesis, the MSDO methodology is widely used within and outside the aerospace community. In particular the work of Cyrus Jilla [37] is a pertinent application example of MSDO to the field of distributed satellites.

MSDO is used in Chapters 3 and 4 of this thesis to model Mars sample preparation facilities and Mars exploration rovers, respectively. Both studies use the full-factorial method for two reasons. First, their models are efficient enough that the complete search of sufficiently large trade spaces takes less than an hour. Second, the full-factorial search permits a better understanding of the shape of the trade space, and provides more insights into why certain designs are optimal.

2.2 Markov probability state models

Markov state models theory is used to calculate the overall performance of a system throughout its lifetime. As a system undergoes failures, its instantaneous performance decreases with time. The Markov reliability model computes lifecycle productivity of a system based on the probability that it is in different partially-degraded states at a given time. The references for this section are Julie Wertz' master thesis [57] and notes from the Probabilistic Systems Analysis class taught by Dimitri Bertsekas and John Tsitsiklis at the Massachusetts Institute of Technology [18].

2.2.1 A matrix

In the context of Markov's theory, every system starts at the initial all-working state and, with time, ends at the total-system-failure state after going through a chain of degraded states. A system with n possible states has n^2 possible transitions between states, each with a certain probability. The transition probability p_{ij} is the probability that the system currently at state i will transition to state j. The key assumptions underlying Markov chains are, first, that the transition probabilities, p_{ij}, are independent of the past history of the process and of the time the next transition takes effect. Second, the time until the next transition from the state i to the state j is exponentially distributed with a given parameter λ_{ij}. Additionally, the time until the next transition out of state i is exponentially distributed with a given parameter ν_i. This parameter is called the transition rate out of state i. The following relation exist between the transition rate out of state i, and the transition rates to other states:

$$\nu_i = \sum_j \lambda_{ij} \tag{2.6}$$

The transition rates are gathered in the transition probability matrix, also called the *A matrix* in this thesis. For a system with n possible states the A matrix is defined

as:

$$A = \begin{bmatrix} -\nu_1 & \lambda_{12} & \dots & \lambda_{1n} \\ \lambda_{21} & -\nu_{22} & \dots & \lambda_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \lambda_{n1} & \lambda_{n2} & \dots & -\nu_{nn} \end{bmatrix} \tag{2.7}$$

The transition rates out of each state are the diagonal elements of the matrix, while the transition rates from one state to another fill the remaining entries.

2.2.2 Probability state vector

The state probability vector, $\Pi(t)$, is defined as the time-dependent vector such that $\pi_i(t)$ is the probability of being in the state i at time t.

$$\Pi(t) = \begin{bmatrix} \pi_1(t) \\ \pi_2(t) \\ \vdots \\ \pi_n(t) \end{bmatrix} \tag{2.8}$$

In particular, the expression of the probability state vector at the beginning of the system's lifetime is:

$$\Pi(0) = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{2.9}$$

During the system's lifetime, the probability state vector behaves according to the following equation that links the derivative of the vector to the vector itself via the A matrix:

$$\frac{\partial \Pi(t)}{\partial t} = A\ \Pi(t) \tag{2.10}$$

2.2.3 Application example

Figure 2-3 presents an example of a system's degradation chain taken from Chapter 3. This system consists of seven elements, numbered 1 to 7, and has seven working states,

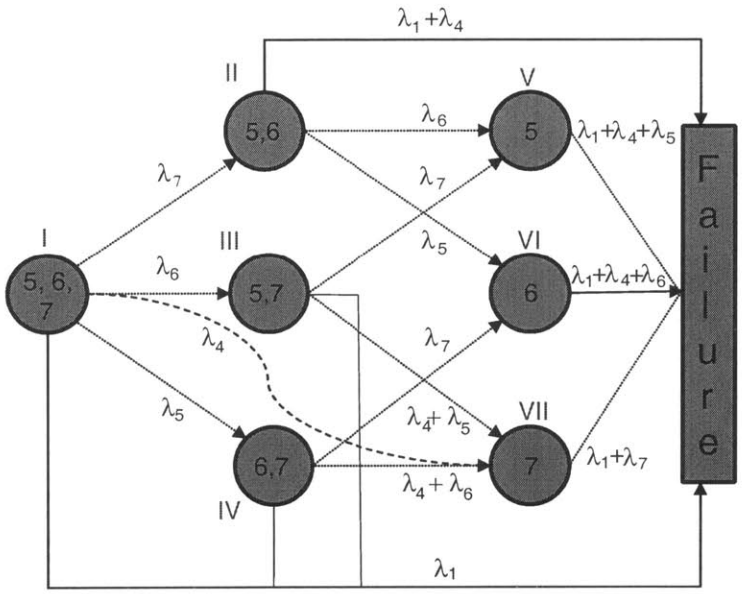


Figure 2-3: Example of transitions between degraded states

numbered I to VII. The variable λ_i is the failure rate of the element i; for example the transition from State I to State II is due to a failure from element 7. The figure shows that there are five paths that depart from the initial state; they involve the failures of any one of elements 1, 4, 5, 6, and 7. Using Equation 2.10 at $t = 0$ gives:

$$\frac{\partial \pi_1}{\partial t} = -(\lambda_1 + \lambda_4 + \lambda_5 + \lambda_6 + \lambda_7)\, \pi_1 \tag{2.11}$$

The solution for that differential equation is a simple exponential form:

$$\pi_1(t) = \pi_1(0)\, e^{-(\lambda_1+\lambda_4+\lambda_5+\lambda_6+\lambda_7)\, t}\ , \text{ where } \pi_1(0) = 1 \tag{2.12}$$

This expression is consistent with the initial value of the state probability vector (Equation 2.9). Similarly, the behavior equation of State II is:

$$\frac{\partial \pi_2}{\partial t} = \lambda_7 \, \Pi_1 \; - \; (\lambda_1 + \lambda_4 + \lambda_5 + \lambda_6) \, \Pi_2 \tag{2.13}$$

And the solution to this differential equation is:

$$\pi_2(t) = e^{-(\lambda_1+\lambda_4+\lambda_5+\lambda_6)\, t} \; \left(1 - \; e^{-\lambda_7 \, t}\right) \tag{2.14}$$

The time dependencies of the rest of Π's entries are calculated in the same fashion and illustrated in Figure 2-4.

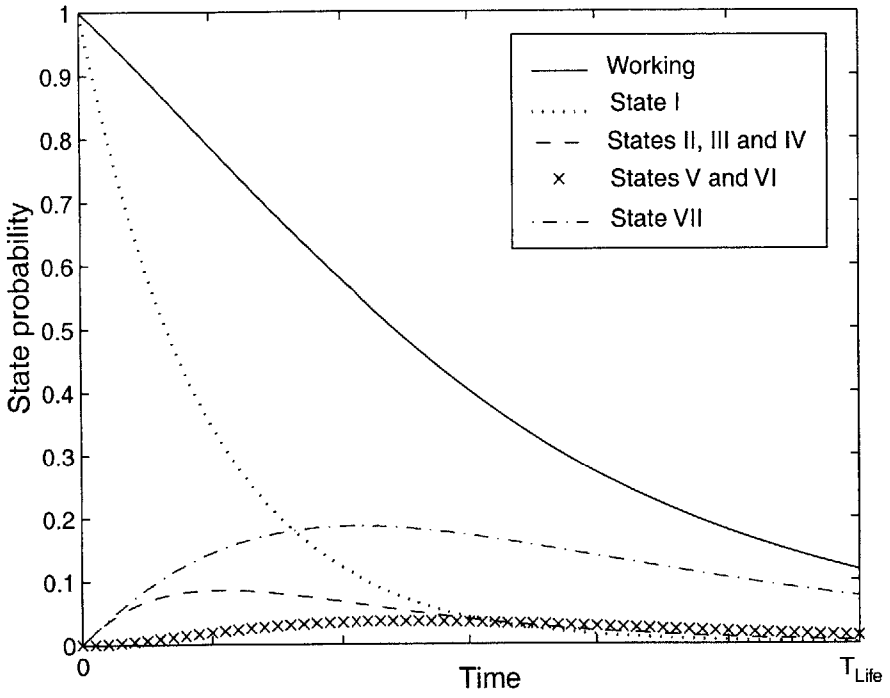


Figure 2-4: Degraded states' probabilities

2.2.4 Lifetime productivity

The productivity of a system in State i is noted C_i. The maximum productivity is C_1, the productivity of the system in its initial state. The system is in that state

at $t = 0$; then the instantaneous productivity of a system at a given time t is the expected productivity of the system over its possible states. In other words, the instantaneous productivity, $C(t)$, is the sum of the products of the productivity of a state with the probability that the system is in that state at t:

$$C(t) = \sum_{States} \pi_i(t)\ C_i \tag{2.15}$$

The lifetime productivity, P_{Life}, is then the integral of the instantaneous productivity over the lifetime of the system:

$$P_{life} = \int_{Life} \sum_{States} \pi_i(t)\ C_i\ dt \tag{2.16}$$

Very often the lifetime productivity value is subsequently handed to a MSDO engine that tries to maximize that objective. For example, the study presented in Chapter 3 optimizes sample preparation facilities; it uses a full-factorial search to find the architectures that maximize lifetime productivity and minimize mass. The lifetime productivity of each architecture is assessed by using a reliability model similar to that presented in this section.

Chapter 3

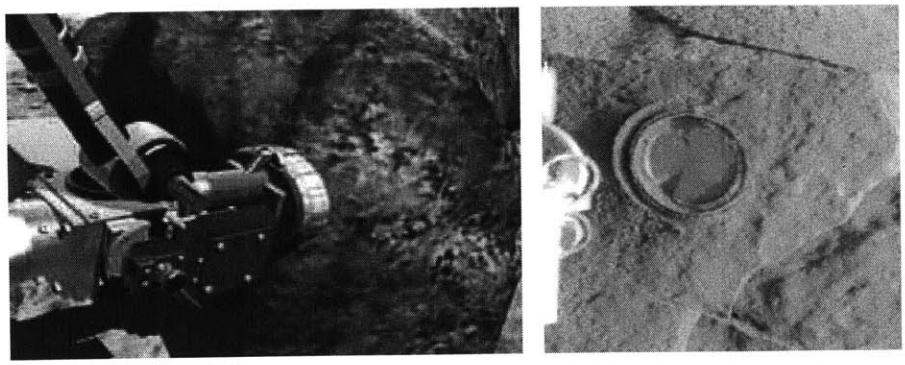
Optimization of Sample Acquisition and Transfer systems

3.1 Introduction and motivation

This section introduces the field of sample acquisition and more specifically the approaches to sample and instrument interaction. It then defends the need for the optimization of the Sample Preparation And Transfer (SPAT) system.

3.1.1 History and incentive for Sample Preparation And Transfer tools

Landed missions to date have used one of two strategies for samples to interact with instruments: bringing the instruments to the rocks or the rocks to the instruments. The first has been employed on most of the Martian surface missions; it involves bringing short-range, non-destructive instruments into the proximity of a sample [17], generally by means of an arm. For instance, on the Mars Exploration Rover (MER) the arm, also called Instrument Deployment Device (IDD), supports a microscopic imager and two spectrometers, along with a powerful grinder called the Rock Abrasion Tool (RAT) illustrated in Figure 3-1. The RAT is the first tool on Mars with the capability of creating a hole (45 millimeters in diameter and 5 millimeters deep) into



(a) Simulation (b) RAT grinding on Mars

Figure 3-1: The RAT grinding of a Martian rock

the surface of Martian rocks [21]. The tool was first used on the 34^{th} sol[1] of Spirit's operations. It was used on a rock called Adirondack (Figure 3-1(b)). This new level of rock processing is scientifically essential because it enables MER instruments to make measurements on fresh rock surfaces unaltered by the Mars environment. Overall many instruments can be improved either by this sort of simple sample processing or sample manipulation and orientation [16]. However, adding extra processing tools to the contact instrument suite significantly complicates the arm technology and design.

Taking large suites of instruments to rocks is quite an engineering challenge as illustrated by the complexity of the Beagle 2 lander's Position Adjustable Workbench (PAW) in Figure 3-2. The PAW is carried by the lander's arm and is composed of a stereo pair of cameras, a microscope, a gamma-ray Mossbauer spectrometer, a X-ray spectrometer, a rock corer and grinder, and a mole. On the one hand, this design maximizes science productivity; instruments can quickly work in sequence on a specific rock without any need for sample transfer between them. On the other hand, this efficiency is traded for complexity which entails risk, and high development and validation costs.

The alternative strategy is to bring unprocessed rocks to the instruments located on the probe's body like was first done on Mars with the Viking laboratory[2]. On

[1] A sol is the name of a day on Mars, its duration is 24 hours 39 minutes.

[2] Viking is the first mission that successfully landed on Mars. The Viking 1 Lander touched down at Chryse Planitia on July 20, 1976 and ended communications on November 13, 1982. The Viking

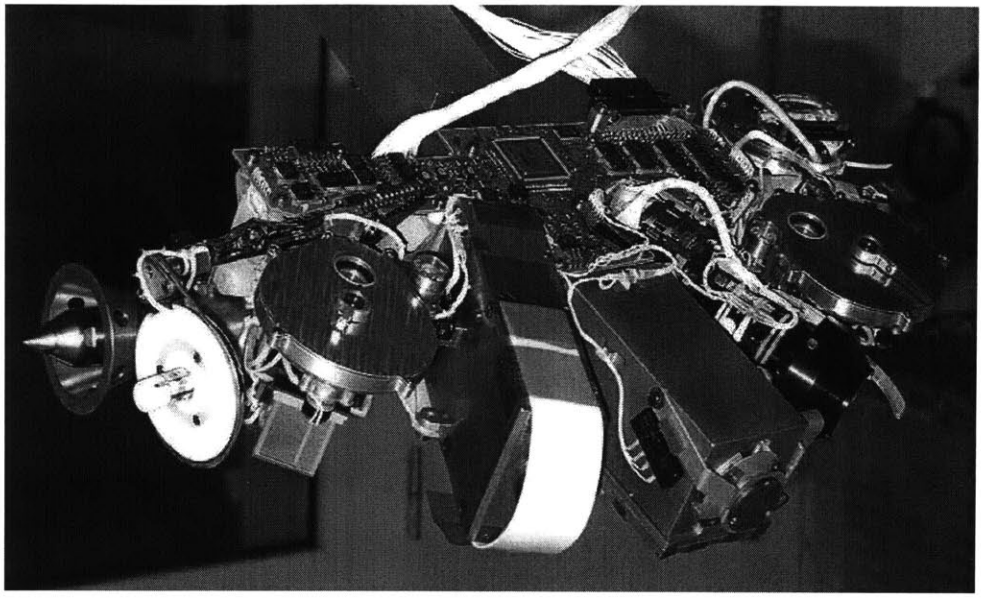


Figure 3-2: Close-up view of Beagle 2's PAW, courtesy of Mullard Space Science Laboratory (MSSL-UCL)

Viking, the arm is free of instruments and is used to scoop soil samples and feed them into the instruments fixed on the lander. The benefit of this arrangement is that the instrumentation design is less constrained; instruments do not need to be miniaturized or shrunk to fit on the arm's tip. The acquisition and transfer of samples to the instruments is the main engineering challenge regarding this approach.

The Viking laboratory is also distinctive because the innovative design of its biology experiment led the way to *shared sample processing.* There are three main experiments on Viking which are the gas chromatograph mass spectrometer (GCMS), the X-ray fluorescence spectrometer and the biology experiment [4]. The GCMS and the X-Ray instruments receive samples by separate inlets, whereas the biology experiment inlet is actually shared by three instruments. After the arm empties a soil sample into the biology inlet, it passes through a 1 millimeter screen. Then, a mechanical soil distribution assembly distributes measured portions of the sample to each of the pyrolytic release, labeled release and gas exchange instruments. This common screen

2 Lander touched down at Utopia Planitia on September 3, 1976 and ended communications on April 11, 1980.

is the first use of a *centralized processor* as opposed to *distributed processors* where each of the above instruments would have its own specific screen. The advantage of the centralized method is that it significantly reduces the hardware's mass.

In summary, the design of instrument suites manages the weighing of three major metrics, which are mass, productivity, and risk, and it also comprises interesting trade-offs such as the distribution as opposed to centralization of sample processing equipment.

3.1.2 Future shared-processing facilities

For intelligent strategic planning of the future exploration of Mars, significant improvements are needed in the fields of sample selection, acquisition, and preparation. Measurements of increasing sophistication are required to answer more refined scientific questions [17]. In the near future, three missions will be using innovative processing facilities in order to improve their science return. The Rosetta lander[3] is equipped with the Sample Drilling and Distribution (SD2) tool, and the Mars Science Laboratory[4](MSL) and ExoMars rovers are equipped with the Sample Preparation And Distribution (SPAD) and the Sample Preparation and Handling System (SPHS) tools, respectively. In the context of this study such tools are from now on referred to as Sample Preparation And Transfer (SPAT) systems. A SPAT system is a multipurpose facility capable of assessing the scientific worthiness of collected samples and of preparing them for analysis by science instruments. Such a facility is required when acquisition tools, such as a drill, provide more samples than can be analyzed by instruments, or when instruments require samples in a special format. The selection and preparation process is detailed for the SPAD tool used by MSL in Figure 3-3 and below:

1. Primary analysis and analysis strategy: a sample is selected for study after its quality and preparation requirements are assessed.

[3]The Rosetta spacecraft designed by ESA was launched on March 2, 2004 and will reach the comet Churyumov-Gerasimenko by 2014.

[4]Designed for Mars' surface exploration, the MSL (NASA) and Exomars (ESA) rovers are scheduled for launch in 2009.

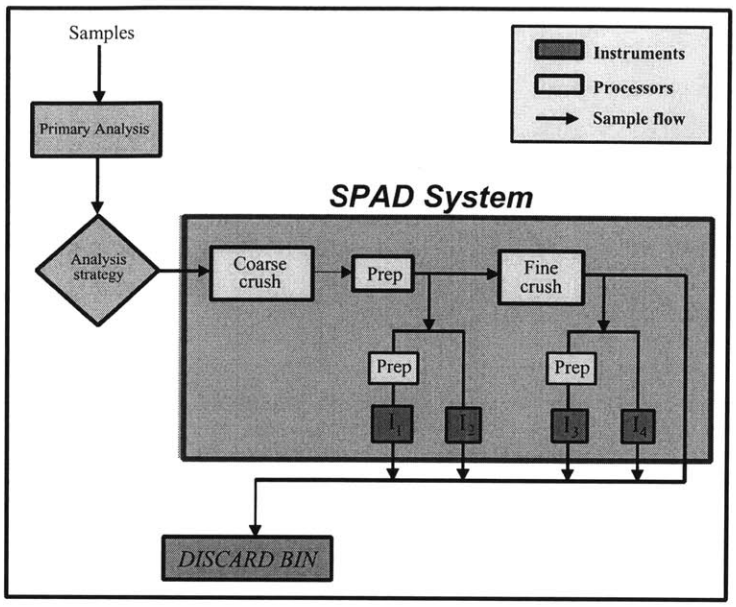


Figure 3-3: Representation of the sample flow through MSL's laboratory (adapted from [15]).

2. Preparation and distribution: the selected sample is prepared and distributed to instruments by devices called processors. All the processors together with the instruments form the SPAD tool.

3. Measurements: instruments analyze the sample.

4. Cleaning: processors and instruments are cleaned to prevent contamination of subsequent samples.

The tiered carousel example The tiered carousel is a possible architecture of the SPAD system (Figure 3-4). It is a design option investigated in detail by the Jet Propulsion Laboratory for its implementation on the MSL rover in 2009. The tiered carousel is a shared facility in the sense that preparation steps that are common to several instruments of the suite are shared by these instruments. For example, if two instruments analyze crushed samples, the carousel has a single crusher that delivers samples to both instruments. The carousel would also perform other types of preparations such as splitting, sieving, advance surfacing breaking and cutting of

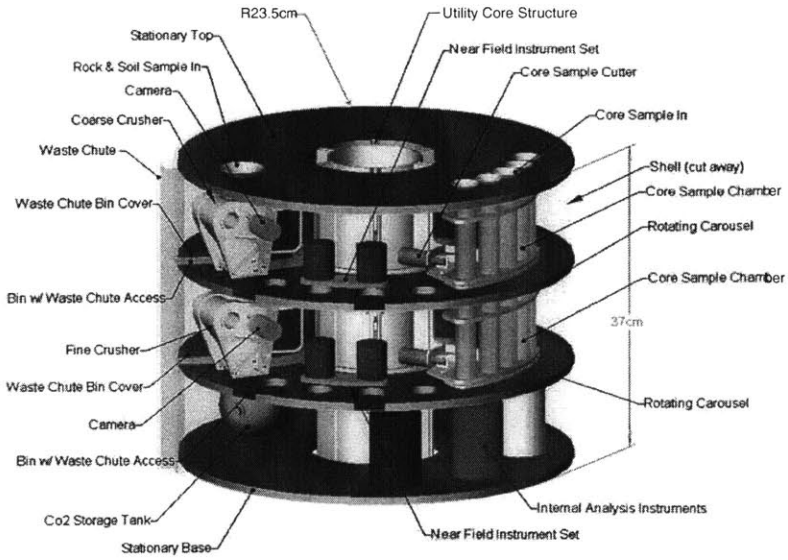


Figure 3-4: Tiered carousel SPAD for the MSL mission([15])

uncrushed rocks, precision loading and sizing [15]. Its potential mass is estimated to range from 25 to 30 kilograms without instruments and its power to be 25 watts [17].

The major rationale for designing SPAD with a shared architecture is to achieve synergy between instruments [17]. A critical strategy in sample science is to analyze the same sample by more than one instrument. Still, there are also arguments against a shared sample preparation and distribution system [17]. First, it is important to balance the amount of mass that is allocated to the shared processing facility as opposed to the instruments. Second, the shared sample preparation may slow down some sample analyses. Third, a shared system adds complexity and risk. These points serve as metrics for the following study which compares the performance of shared architectures to alternative designs.

3.1.3 Study objectives and approach

Objectives The shared-processing architecture is an appealing solution to the design of SPAT systems. However, for this solution to be completely justifiable, it must be compared to alternative architectures especially in terms of the arguments that

play against it. This study presents a process to optimize architectures for SPAT systems with respect to mass, sample throughput, and risk. By using the proposed method, designers can generate and assess a vast array of architectures, and defend the decision of whether or not a shared-processing facility is the optimal solution.

The goals of the SPAT model are specifically to:

1. Generate all possible SPAT architectures for a given instrument baseline

2. Rank architectures with respect to their mass, sample throughput, and operational risk

3. Identify the mass, productivity and risk drivers of SPAT

Approach Although this problem shares some similarities with queue problems, the queuing theories do not apply to it. Indeed, queues are probabilistic by nature whereas the flow of samples through a SPAT system is a deterministic process. The flow involves samples entering the system and then going through a sequence of processing steps which lead to a final analysis by an instrument. It is assumed that a new sample enters the SPAT as soon as the SPAT is ready to accept it. Moreover, the processing and analysis durations are assumed known and constant. A new approach was, therefore, created to model systems with behaviors similar to the SPAT (Figure 3-5).

First, an instrument baseline is defined based on the recommendations from the science community. This baseline includes the list of instruments that constitute the SPAT and also the processing sequences that each of these instruments requires. Then, this baseline is transformed into mathematical terms using matrices. These matrices contain information about the processors' and instruments' masses and processing times, as well as the connections between these, called *sample paths*. Once this is done, the trade space of all architectures is generated by arranging these sample paths in every possible combination. Finally utility functions are applied to these architectures in order to assess their cost-benefits.

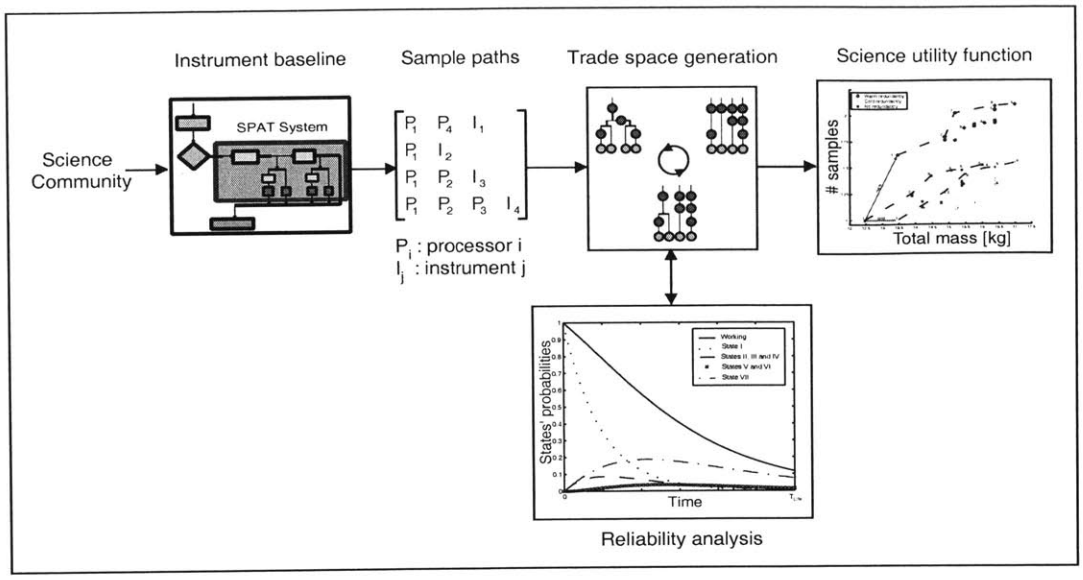


Figure 3-5: Study methodology

Full-factorial search is used for three reasons. First, the trade space of architectures is small enough to be explored exhaustively. Indeed, the current designs for SPAT do not have more than 10 instruments, therefore, the trade space size is limited to 1024 architectures. Second, partial search methods are not appropriate for this study, because a SPAT architecture cannot be defined by means of design variables. As explained in the following section, each architecture is a particular combination of sample paths which cannot be represented by a set of variables. Finally, the current model does not have an indisputable utility function to weigh the science return against mass and risk metrics. In a full-factorial method the utility function is conveniently the last stage of the design flow (Figure 3-5). It is therefore possible to run a single trade space exploration and subsequently try multiple utility functions on that same trade space (Section 2.1.4).

3.2 Architecture modeling

This section first defines the concepts used in the SPAT modeling such as sample paths, instruments, and processors and then presents their mathematical implemen-

tation.

3.2.1 Definitions

Sample paths As shown in the introduction of this section (Figure 3-3), after going through the primary analysis and analysis strategy, a sample undergoes a series of preparations performed by processors which lead to its eventual analysis by an instrument. The sample processing sequence is called a *sample path.* By definition, there are as many sample paths as there are instruments; in Figure 3-6, for example, all architectures have four sample paths, and $[P_1,\ P_4,\ I_1]$ is one of them. Since,

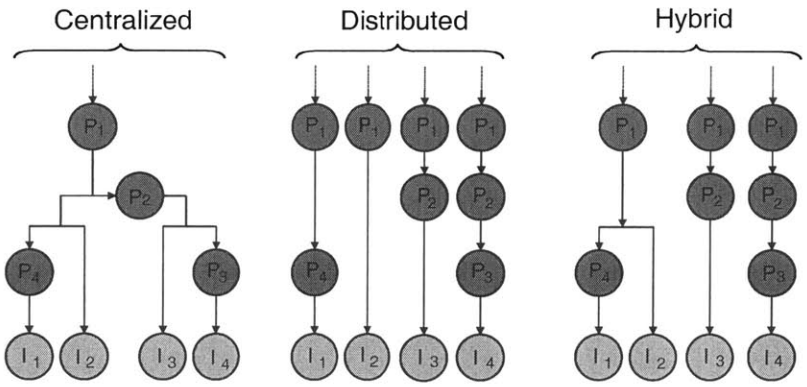


Figure 3-6: Examples of three possible SPAT architectures

within a trade space all architectures have the same instrument baseline, the number of sample paths does not change from one architecture to another. The sequence of processors in a path does not change either because the instrument in that path dictates what preparations samples must undergo. For example, the path leading to instrument I_3 is made of the processor P_1 followed by P_2 for any of the three architectures shown in Figure 3-6.

What changes along a sample path among the architectures is the size of its processors. A processor that belongs to several paths, meaning it is shared by multiple instruments, handles a larger number of samples than a processor which is dedicated to a single instrument. Therefore, the shared processor needs to be larger than the specific processor. In the centralized architecture (Figure 3-6), for example, the

processor P_2 of the sample path $[P_1,\ P_2,\ I_3]$ processes a sample unit (defined in Section 3.2.2) for Instrument I_3, and another one for Instrument I_4. In the distributed architecture, processor P_2 of the same path processes only a single sample unit at a time, dedicated to Instrument I_3. A replicate of this processor handles another sample unit for the other instrument I_4. Processor P_2 of the centralized architecture is, as a consequence, larger than the one for the distributed architecture.

In summary, the commonalities between architectures are the instrument baseline, the number of paths, and the sequences of processors on each path. The architectures differ by their arrangement of paths which entail different processor sizes depending on whether these are centralized or distributed.

The trade space of all possible architectures is generated by creating all the combinations of these sample paths, that is to say all the different ways of centralizing processors. Two or more processors can be centralized if they perform the same function on samples; a collection of paths that share processors is then called a *set of paths.* In Figure 3-6, for example, the hybrid architecture is obtained from the distributed one by combining sample paths of I_1 and I_2. This combination is possible because both paths use the same processor P_1 that can be centralized in a bigger processor for both paths, which consequently forms a set of paths.

Instruments and processors From a sample flow point of view, the triage system, processors, and instruments have a similar action on a sample. They all represent a step in the path where the sample is being acted on for different periods of time. Hence, in order to quantify the performance of an architecture in terms of mass and productivity, three parameters are defined that characterize all of the triage system, processors and instruments (as they are mathematically represented by the same variables, they are from now on just referred to as *elements*). These three parameters are:

1. The element's mass m.
2. The element's processing time τ. It is the time required for the element to act on a sample including the subsequent cleaning operations.

3. The element's failure rate λ. Its inverse is the expected working duration.

Moreover, as for family trees, one element is said to be the *parent* of all the elements that are directly following it in the sample flow, and these elements are then referred to as its *children*. In the example of Figure 3-7, which shows another hybrid architecture, E_{11} is the parent of E_4 and E_7, whereas E_{12} is the parent of E_{22} only. Regarding the elements' notation, the reference numbers are given arbitrarily with the norm that E_{ij} and E_{ik} represent the same sort of element E_i that is duplicated and used separately by the sets of paths j and k. Note that as the set of paths may serve different numbers of instruments, E_{ij} and E_{ik} may not be exact replicas in terms of size. For example, E_{11} and E_{12} are both elements of type E_1, but E_{11} is bigger than E_{12} because it serves three instruments instead of one.

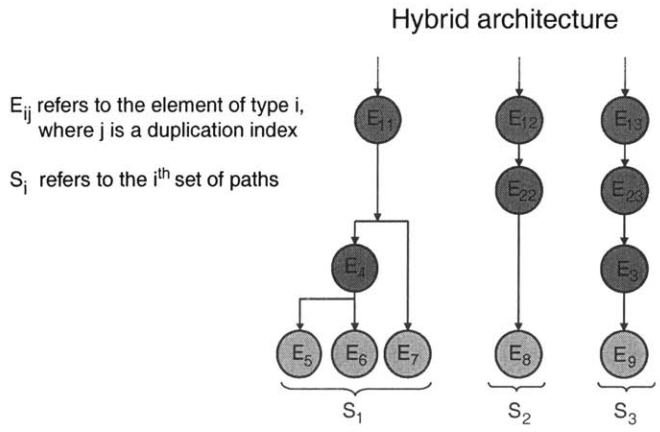


Figure 3-7: Parent and child relationship example on a hybrid architecture

3.2.2 Assumptions

The present model works under the following assumptions presented in two categories, the first dealing with sample path layout, and the second with sample flow rules.

Assumptions about sample path layout To simplify the model it is assumed that each sample path contains only one instrument. This implies that there cannot be an instrument in the middle of a path; instruments are the terminating elements

of each path. Additionally, the first preparation step is assumed to be common to all instruments. Hence, all sample paths of a system begin with the same element E_1. In the SPAD system example (Figure 3-3), the coarse crusher is the first processor for every instrument preparation sequence. The element E_1 is also the only one without a parent; it is assumed that it can be fed with a sample anytime, from an unlimited source of samples. For example, for the architecture presented in Figure 3-7 the first elements of the three sets of paths are E_{11}, E_{12} and E_{13} – all of them of type E_1. Additionally, any other element must have one and only one parent. There cannot be a configuration where two elements, except redundant ones, are feeding in the same child element. Finally, there cannot be any connections from one instrument to another [38]. These assumptions require that every set of paths have a tree shape with E_1 being the trunk, the processors being the branches and the instruments being the leaves.

Assumptions about sample flow All samples are first prepared by the same element E_1 before being transfered to subsequent elements. Regarding the sample flow, there can be multiple samples along a path as long as they are in separate elements and flow one after the other. One parent distributes samples instantaneously to all its children at the same time. In order to do so, one condition is that all the children must have empty bellies: one element cannot process two different samples together. Therefore, the parent has to wait for all of its children to digest their samples before giving them new ones. One consequence, which is explained in further detail below, is that, if children have different digestion times, the fast ones have to wait for the slow ones before getting a new sample to digest.

Furthermore, all instruments are fed the same amount of material which is called a *sample unit.* As a consequence, if a processor belongs to a set of paths of N instruments, it must deal with N sample units of material that are then distributed equally among these instruments. As a first order approximation, the time required to process N sample units of material is set equal to the time required for processing a single one. Finally, as no science utility function is easily available, instruments are

assumed to produce equivalent science. In other words, the science return on each sample analysis is assumed to be the same for every instrument. Fortunately, full-factorial analysis allows this assumption to be revised without rerunning the analysis.

3.2.3 Matricidal representation of SPAT architectures

The tool uses a general matricidal formulation to model architectures and their properties.

Design cell A SPAT architecture is represented mathematically by a matrix cell, which is simply a set of matrices. In the present model, each matrix of the cell represents one of the architecture's independent set of paths. An architecture with three independent sets of paths has a design cell of three matrices. For example, the architecture in Figure 3-7 is represented by the cell, C_{Hybrid}, defined as follows:

$$
\begin{aligned}
C_{Hybrid} &= \left\{M^1, M^2, M^3\right\} \qquad (3.1)\\
M^1 &= \begin{bmatrix} 1 & 4 & 5 \\ 1 & 4 & 6 \\ 1 & 7 & 0 \end{bmatrix}\\
M^2 &= \begin{bmatrix} 1 & 2 & 8 \end{bmatrix}\\
M^3 &= \begin{bmatrix} 1 & 2 & 3 & 9 \end{bmatrix}
\end{aligned}
$$

Each independent set of paths, S_k, in the original architecture is represented by a matrix M^k (Equation 3.1). The architecture in this example has three sets of independent paths and, therefore, is represented by three matrices M^1, M^2 and M^3. Each row inside a matrix represents one path within that set. The component M^k_{ij} is the reference number of the j^{th} element of the i^{th} path of the set of paths S_k. For example, the first set of paths S_1 is made up of three paths, therefore, M^1 has three rows. The first row of M^1, $[1, 4, 5]$, represents the first path $[E_{11}, E_4, E_5]$ of S_1.

Matrices of the elements' properties The mass, processing time, and failure rate properties of an architecture's elements are simply stored into three matrices M, T and Λ, respectively (Equation 3.2). The mass, processing time, and failure rate of the element E_i are m_i, t_i, and λ_i, respectively.

$$\begin{aligned} M_{Hybrid} &= [m_1\ m_2 \ldots m_9] \\ T_{Hybrid} &= [\tau_1\ \tau_2 \ldots \tau_9] \\ \Lambda_{Hybrid} &= [\lambda_1\ \lambda_2 \ldots \lambda_9] \end{aligned} \tag{3.2}$$

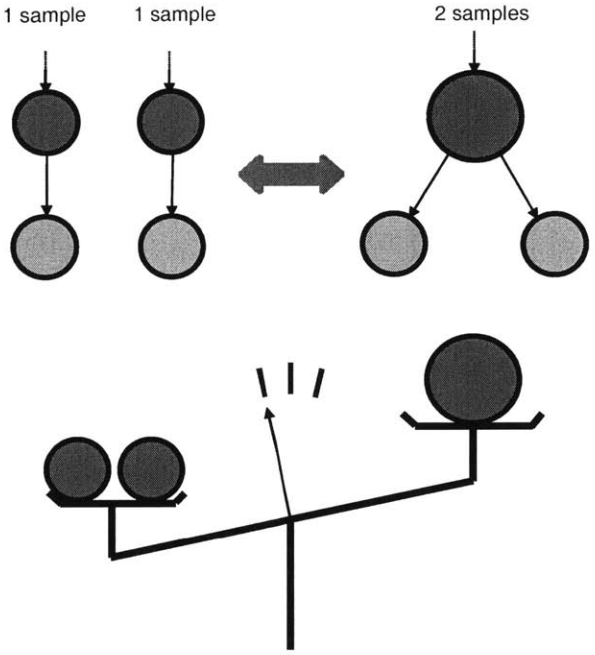
3.3 Trades modeling

This section explains in detail the fourth step of Figure 3-5 which applies the utility function to all the generated architectures. It goes through modeling notions like economy of scale for mass, bottleneck and reliability for productivity that also encompass operational risk. Finally, it highlights the trade-off between warm and cold redundancy.

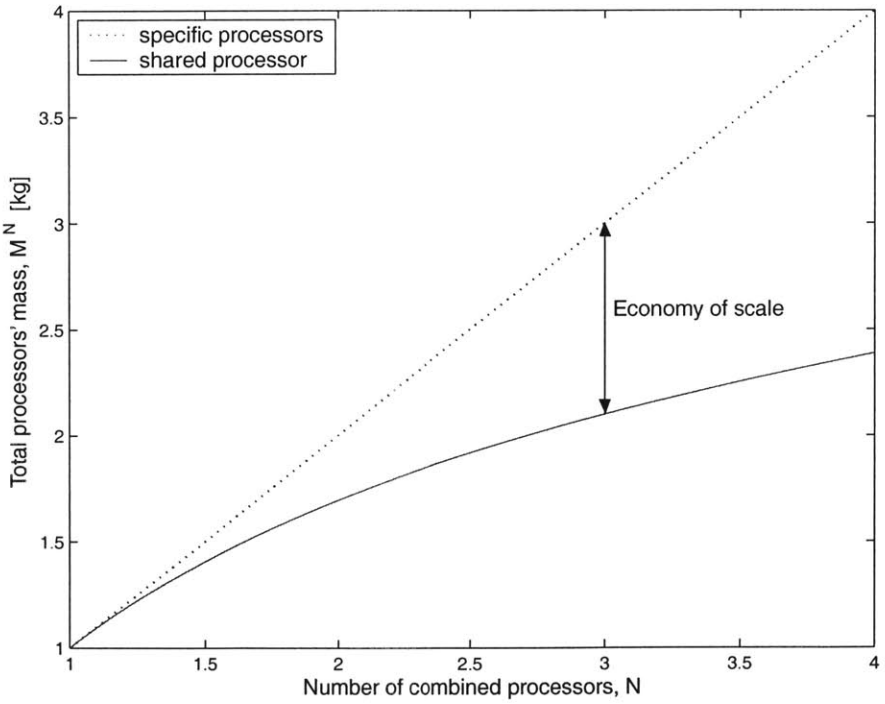
3.3.1 Mass

This part details the consequences on the overall mass of the architecture of merging two processors together. According to the assumptions detailed in the above section, only processors of a same type can be combined. The resulting processor then has to handle twice as many sample units. However, that does not necessarily imply that it needs to be twice as big or massive. Indeed, by taking into account some economy of scale, one processor dealing with N sample units is less massive than N individual processors dealing with one sample unit, as illustrated in Figure 3-8(a). The same notion is present in commercial packaging where a bottle of two liters is less massive and expensive than two of one liter each.

The economy of scale function used in this study is an educated guess; it is illus-



(a) Illustration of mass savings by centralization



(b) Quantification of the economy of scale

Figure 3-8: Economy of scale and processor centralization

trated in Figure 3-8(b), and its formulation is the following:

$$M^N = M^1 \ (ln\,(N) + 1) \tag{3.3}$$

$$M_{Savings} = N \times M^1 - M^N \tag{3.4}$$

In this equation M^N is the mass of the processor which is equivalent to the combination of N processors, each of mass M^1. The mass savings made through the economy of scale are expressed in Equation 3.4.

3.3.2 Productivity

The consequences of processor centralization on the overall architecture productivity are twofold: first, it decreases the instantaneous productivity by creating bottlenecks; second, it decreases lifetime productivity by adding single point failures.

Instantaneous productivity Centralized architectures are slower than distributed ones because before feeding in a new sample a parent element has to wait for all its children to have digested their former sample. The pace at which a set of paths works is dictated by its slowest element, called the *bottleneck*. No matter how fast the other elements of the set are, they still have to wait for the bottleneck (Figure 3-9). In the case of the distributed architecture, the sample path of the slowest element is isolated from the others. In that configuration, every sample path works at its own pace. Therefore, the distributed architecture is optimal in terms of productivity. For example, in the centralized configuration of Figure 3-9, if I_1 is the slowest element, since I_1 shares P_1 with I_2, this one must work at the pace of I_1. In the distributed architecture I_1 and I_2 work at their own pace; as a consequence, the overall productivity of the distributed configuration is higher than the centralized one. The mathematical expression of the bottleneck is that the productivity, C, of a set of paths is equal to the number of paths times the inverse of the longest processing time; in the case of

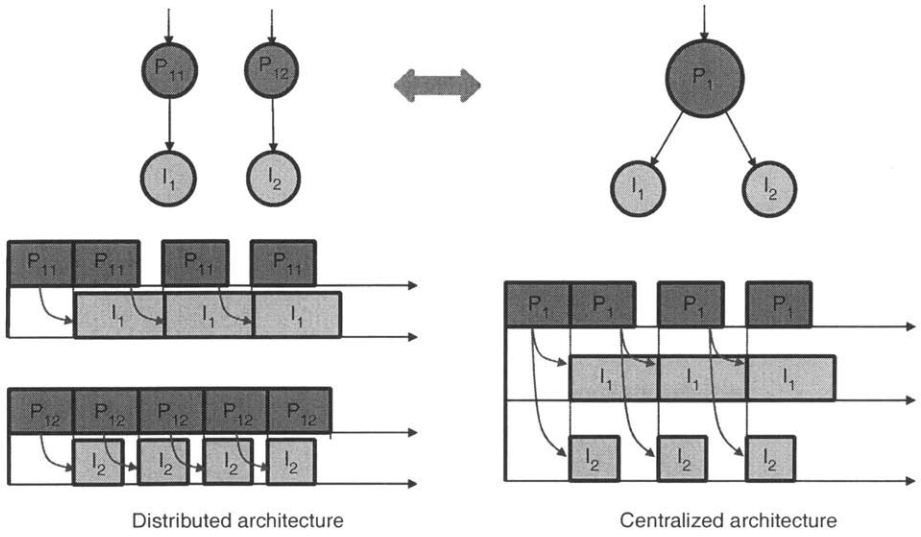


Figure 3-9: Illustration of the bottleneck effect in centralized architectures

the example in Figure 3-9:

$$\begin{aligned} C_{Centralized} &= \frac{2}{\tau_{I1}} \; [samples/s] \\ C_{Distributed} &= \frac{1}{\tau_{I1}} + \frac{1}{\tau_{P1}} \; [samples/s] \end{aligned} \tag{3.5}$$

The general formula for the instantaneous productivity of a hybrid architecture is the sum of the productivities of all of its sets of paths:

$$C_{Instantaneous} = \sum_{paths} \frac{N_{paths}}{max\,(\tau_i)} \; [samples/s] \tag{3.6}$$

The only case when centralized and distributed architectures have the same instantaneous productivity is when the first element, E_1, is the bottleneck. Indeed, whatever the architecture configuration, E_1 is common to every path, therefore, all paths work at the same pace whether they are centralized or not.

Lifetime performance Along its lifetime, an architecture is degrading and goes through a series of states with decreasing productivity. A degradation, namely the failure of an element, is a probabilistic event; consequently, the lifetime performance is

the expected total productivity of an architecture over its lifetime. Since degradations affect sets of paths independently, productivities are calculated for each set separately and then added together to result in the architecture's total performance.

For each set, a succession of degradations leads from the initial state where all elements work to the final failure state. From a performance point of view, the degradation states in between are defined in terms of the number of paths still functioning. For example in Figure 3-7, S_2 has one working state and six degraded states which consist of all the combinations of possible element failures. However, all of the degraded states are actually failure states, that is, non-productive states. Hence, only two states are differentiated (working and failure) in lieu of eight.

Degraded States This paragraph goes into the details of the degraded capability calculations using the set S_1 (Figure 3-7) as an application example. For a system with n connected paths there are 2^n possible states. The set S_1 is made of three paths and, therefore, has one nominal state and seven degraded states. The detail of all the possible states is given below (Equation 3.7) with the numbers 5, 6 and 7 referring to the paths of E_5, E_6 and E_7, respectively. The matrix M_{States} is constructed so that each row is a state with a non zero productivity, and the numbers of each row refer to the working paths for that state.

$$M_{States} = \begin{bmatrix} 5 & 6 & 7 \\ 5 & 6 & \\ 5 & 7 & \\ 6 & 7 & \\ 5 & & \\ 6 & & \\ 7 & & \end{bmatrix} \quad \underbrace{\begin{pmatrix} I \\ II \\ III \\ IV \\ V \\ VI \\ VII \end{pmatrix}}_{States'\ reference\ numbers} \tag{3.7}$$

The next step is to map all the possible transitions from the initial all working state to the final failure state as illustrated in Figure 3-10. The mathematical representation

of this figure is the A-matrix as defined in Chapter 2, whose automatic construction is explained in the next paragraph.

Transition matrix Consistent with Markov's theory, it is assumed that two elements cannot fail simultaneously. A transition from one state to another is then due either to an instrument failure or to a processor failure. In the former case the transition leads to a state amputated from exactly one path, whereas in the latter case it can lead to a state amputated from more than one path if the failed processor is shared by multiple paths. For instance, from the initial state (I) with three working paths, a failure of E_4 leads to state (VII) which has only one working path remaining (Figure 3-10). The transitions due to instrument and processor failures are mapped by means of the matrices INS and PRO, respectively, which detail the instruments and processors working in each state. As for M_{States}, each row of both matrices is a state and each column is now specific to one instrument in INS or processor in PRO.

$$INS = \begin{bmatrix} 5 & 6 & 7 \\ 5 & 6 & 0 \\ 5 & 0 & 7 \\ 0 & 6 & 7 \\ 5 & 0 & 0 \\ 0 & 6 & 0 \\ 0 & 0 & 7 \end{bmatrix}, \; PRO = \begin{bmatrix} 1 & 4 \\ 1 & 4 \\ 1 & 4 \\ 1 & 4 \\ 1 & 4 \\ 1 & 0 \\ 1 & 4 \end{bmatrix} \tag{3.8}$$

If INS_{ij} has a 0 value, it means the instrument in the column j is not working for the state i. For example, INS_{23} equal to 0 means that instrument 7 is not working in the degraded state II. Following on this idea, there exists a transition between state i and j if one of the two following conditions is satisfied:

1. **Instrument Failure** There is transition from state i to state j if the i^{th} and j^{th} rows of INS differ by one and only one digit. The state j is the state i minus one instrument, therefore, the j^{th} row of INS is similar to the i^{th} one

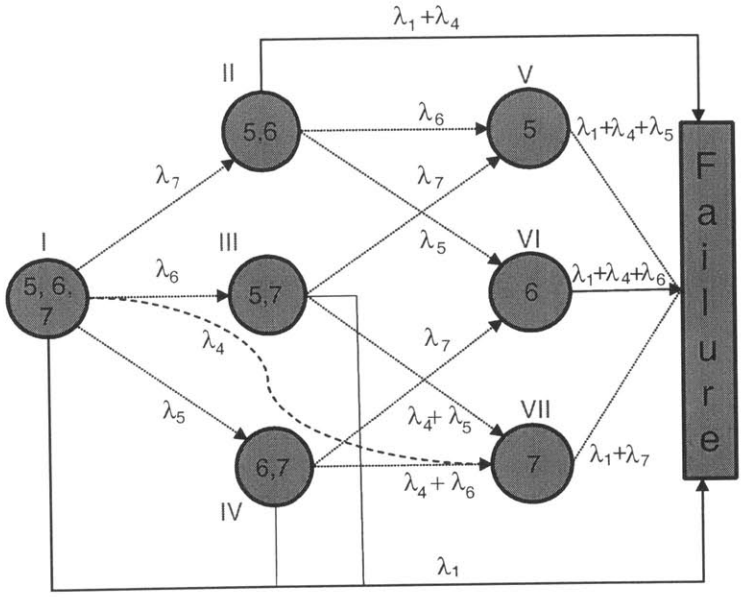


Figure 3-10: Example of transitions between degraded states

except for the 0 in the column of its failed instrument.

2. **Processor Failure** There is a transition from state i to state j if two conditions are met. First, the i^{th} and j^{th} rows of PRO must differ by one digit. Second, the working instruments of the original state i must include those of state j.

By inspection of the INS and PRO matrices, these state transitions are found and the A-matrix is updated with the failure rate of the related instrument or processor.

$$A = \begin{bmatrix} -\Sigma_{i\in\Phi}\lambda i & 0 & 0 & 0 & 0 & 0 & 0 \\ \lambda 7 & -\Sigma_{i\in\Gamma}\lambda i & 0 & 0 & 0 & 0 & 0 \\ \lambda 6 & 0 & -\Sigma_{i\in\Theta}\lambda i & 0 & 0 & 0 & 0 \\ \lambda 5 & 0 & 0 & -\Sigma_{i\in\Omega}\lambda i & 0 & 0 & 0 \\ 0 & \lambda 6 & \lambda 7 & 0 & -\lambda 1-\lambda 4-\lambda 5 & 0 & 0 \\ 0 & \lambda 5 & 0 & \lambda 7 & 0 & -\lambda 1-\lambda 4-\lambda 6 & 0 \\ \lambda 4 & 0 & \lambda 4+\lambda 5 & \lambda 4+\lambda 6 & 0 & 0 & -\lambda 1-\lambda 7 \end{bmatrix} \tag{3.9}$$

The sets Φ, Γ, Θ, and Ω are defined as follows:

$$\Phi = \{1,4,5,6,7\} \tag{3.10}$$

$$\Gamma = \{1,4,5,6\} \tag{3.11}$$

$$\Theta = \{1,4,5,7\} \tag{3.12}$$

$$\Omega = \{1,4,6,7\} \tag{3.13}$$

Lifetime productivity Once the A-matrix has been filled automatically, the probability π_i of being in the state i at time t is given by Equation 3.14 [57].

$$\pi_i(t) = (I + \Delta t\ A) \times \pi_i(t-1) \tag{3.14}$$

This equation is an approximation for small Δt. Finally, the total productivity of a set of paths over its lifetime is given by (Section 2.2.1):

$$P_{Life} = \int_{Life} \sum_{States} \pi_i(t)\ C_i\ dt \tag{3.15}$$

where C_i is defined in Equation 3.6 and π_i in Equation 3.14.

Shared processors in centralized architectures are single point failures that prevent these designs from having graceful degradations and consequently reducing their lifetime productivity. One answer to single point failures is to introduce redundancy in the system, as explained in the next section.

3.3.3 Redundancy analysis

As seen in the former section and in Figure 3-10, the failure of the element E_1 is critical because it leads directly to the system failure state. To reduce the number of such single point failures, systems are built with redundant elements. This study examines two strategies, *cold* and *warm* redundancies, that can be used to improve SPAT systems. The two kinds of redundancies differ in the way the duplicated element is operated.

Cold redundancy It is the type of redundancy usually implemented on spacecraft: while the original element is working, the redundant one is on stand-by. As soon as the former fails, the latter takes over. Figure 3-11 illustrates the duplication of the element E_4 in the set of path S_1 introduced in Figure 3-7. The redundant elements E_{4a} and E_{4b} are identical to the original element E_4 . In a cold redundancy scenario, E_{4b} does not operate, and therefore does not start to wear out, until E_{4a} breaks. Thus, the introduction of the redundant element E_{4b} increases the number of possible degraded states of the architecture. For each state of Equation 3.8 with processors (E_1, E_4), there are two corresponding states in the redundant architecture:

$$\begin{pmatrix} E_1 & E_4 \end{pmatrix} \Rightarrow \begin{pmatrix} E_1 & E_{4a} & E_{4b} \\ E_1 & 0 & E_{4b} \end{pmatrix} \tag{3.16}$$

For example, the state II of the original architecture becomes two states in the redundant architecture:

$$INS = \begin{bmatrix} 5 & 6 & 0 \end{bmatrix} \Rightarrow INS = \begin{bmatrix} 5 & 6 & 0 \\ 5 & 6 & 0 \end{bmatrix} \tag{3.17}$$

$$PRO = \begin{bmatrix} 1 & 4 \end{bmatrix} \Rightarrow PRO = \begin{bmatrix} 1 & 4a & 4b \\ 1 & 0 & 4b \end{bmatrix} \tag{3.18}$$

The second rows of INS and PRO correspond to the state where E_{4a} fails and E_{4b} takes over. Notice that the failure of the element E_{4b} is not allowed before that of E_{4a}. In a mathematical terms, the state [1 4a 0] is not possible for the cold redundancy case.

Warm redundancy In the case of warm redundancy, both duplicates work simultaneously until they fail. In the cold redundancy case, the redundant elements work sequentially, whereas in the warm redundancy case both elements work in parallel. Therefore, in the warm redundancy case, E_{4b} can fail before E_{4a}. This implies that the warm redundancy solution provides even more degraded states than the cold one. For each state with processors (E_1, E_4) of the original architecture, there are three

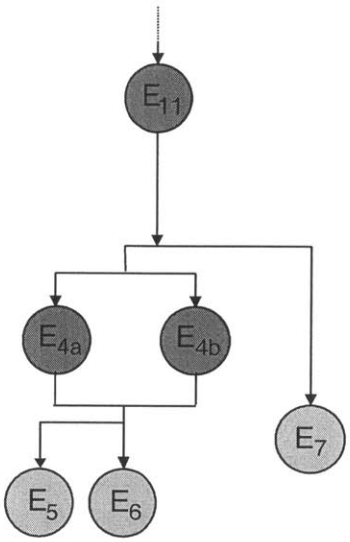


Figure 3-11: Example of element redundancy

corresponding states in the redundant architecture:

$$\begin{pmatrix} E_1 & E_4 \end{pmatrix} \Rightarrow \begin{pmatrix} E_1 & E_{4a} & E_{4b} \\ E_1 & 0 & E_{4b} \\ E_1 & E_{4a} & 0 \end{pmatrix} \tag{3.19}$$

For example, the State II of the original architecture becomes three states in the redundant architecture:

$$INS = \begin{bmatrix} 5 & 6 & 0 \end{bmatrix} \Rightarrow INS = \begin{bmatrix} 5 & 6 & 0 \\ 5 & 6 & 0 \\ 5 & 6 & 0 \end{bmatrix} \tag{3.20}$$

$$PRO = \begin{bmatrix} 1 & 4 \end{bmatrix} \Rightarrow PRO = \begin{bmatrix} 1 & 4a & 4b \\ 1 & 0 & 4b \\ 1 & 4a & 0 \end{bmatrix} \tag{3.21}$$

Notice that from a sample throughput point of view, two working duplicates are equivalent to a single element with half their processing time. Cold redundancy improves lifetime performance by lengthening the expected lifetime, while warm redundancy

increases lifetime performance by increasing the instantaneous productivity.

The current SPAT model permits the redundancy of only one element for the whole system. Identifying and generating all the possible combinations of redundancy is a delicate task and part of the future work.

3.4 Verification

Small automatically generated trade spaces were compared with manually generated ones to check if the SPAT model created the proper architectures. The processing times and masses of several architectures were also validated. Furthermore, the following discussion demonstrates that the model of SPAT manages to capture important trades such as the presence of bottlenecks in the architecture. The SPAT model is applied to the particular system shown in Figure 3-3; the centralized architecture for that system is shown in Figure 3-12. The trade space of architectures is analyzed

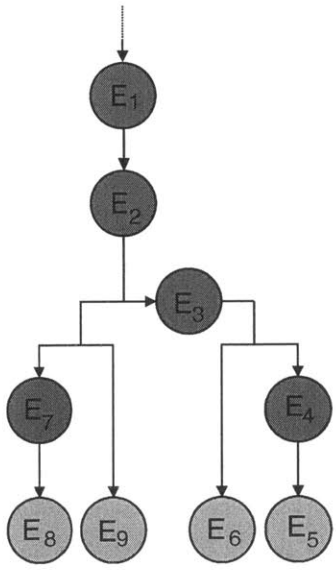


Figure 3-12: Centralized architecture of the SPAT system

for two scenarios: in one case, the element processing times are such that there are several bottlenecks in the system (Equation 3.22); in the other case, there is only one

bottleneck (Equation 3.23).

$$T_1 = [4\ 4\ 4\ 4\ 20\ 15\ 8\ 4\ 9]\ hours \tag{3.22}$$

$$T_2 = [20\ 4\ 4\ 4\ 4\ 15\ 8\ 4\ 9]\ hours \tag{3.23}$$

As explained in Section 3.2.3, $T(i)$ is the processing time for element E_i (Figure 3-12). In the first case (T_1), the children E_5, E_6, E_7 and E_9 take more time to digest samples than their parents; these four elements are potential bottlenecks. In the other case (T_2), E_1 is the bottleneck and, therefore, all sample paths work at the same pace whatever the architecture (Section 3.3.2).

Both trade space cases are plotted in Figure 3-13 as a function of the number of samples analyzed and mass. The number of samples is normalized with respect to

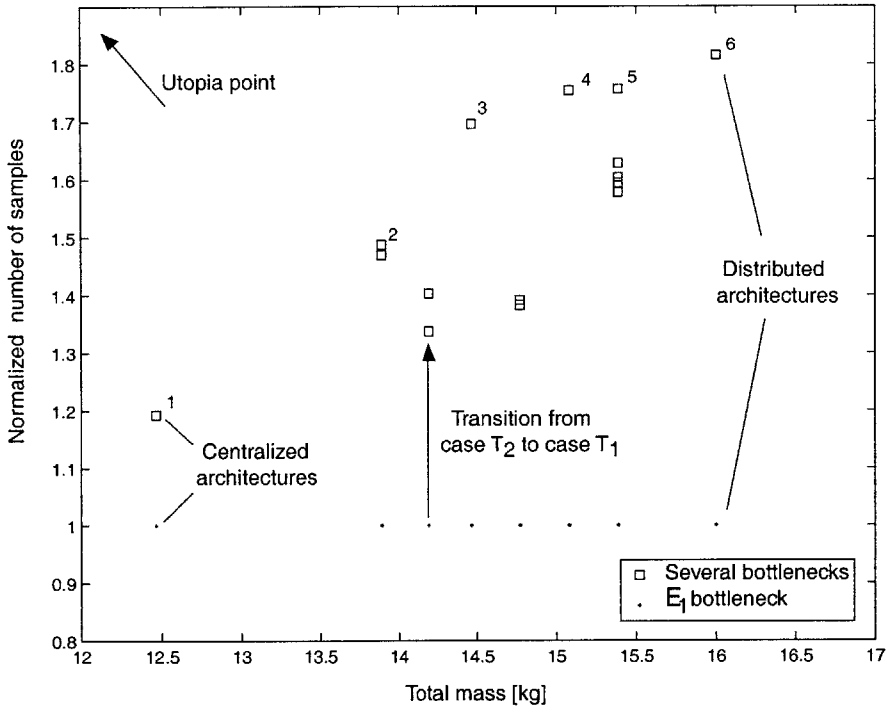


Figure 3-13: Trade space of SPAT architectures

that of the centralized architecture for the case of Equation 3.23. Several observations can be made about this plot. First, as mentioned earlier in this chapter, centralized architectures minimize the mass of the system and have the smallest productivity;

on the other hand, distributed architectures maximize the productivity but are the heaviest solutions. Second, the case with E_1 as the sole bottleneck is the special case where all architectures have the same productivity. In the other case, the architectures keep the same mass (they remain on the same vertical line) but have higher productivities.

The Pareto front (defined in Section 2.1.5) is composed of six architectures spanning from the centralized architecture, light but not very productive, to the distributed architecture, productive but heavy (Figure 3-13 and 3-14). In addition to the centralized (C_1) and distributed (C_6) architectures, four hybrid architectures constitute the Pareto front. These optimal hybrid architectures provide a continuous evolution of the degree of distribution from the centralized architecture to the distributed architecture. The lightest of these hybrid architectures, labeled C_2, shows a first degree of distribution. This architecture centralizes most of its sample paths but for the path of Element 8. Equation 3.22 shows that elements in this path have the shortest processing times. Among all the four sample paths that can be isolated, the sample path of Element 8 is, therefore, the best candidate. Architecture C_2 is heavier than the centralized architecture C_1 but has a higher productivity because it isolates the quickest path from the other slower paths. Reciprocally, Architecture C_5 represents a first degree of shared preparation. It groups the sample paths of Elements 8 and 9 which have a similar processing time, and which are the most productive paths. The results of this section prove that the SPAT model correctly handles the interesting trade-offs regarding bottlenecks and economy of scale.

3.5 Application example

This section compares the effects of warm and cold redundancy on the same trade space of architectures used in the validation (Equation 3.22). This short study focuses on the analysis of the cost benefits of a unique element redundancy. The element chosen for redundancy is the element E_5 of Figure 3-12. All the architectures with no redundancy, cold redundancy and warm redundancy of that element are gathered

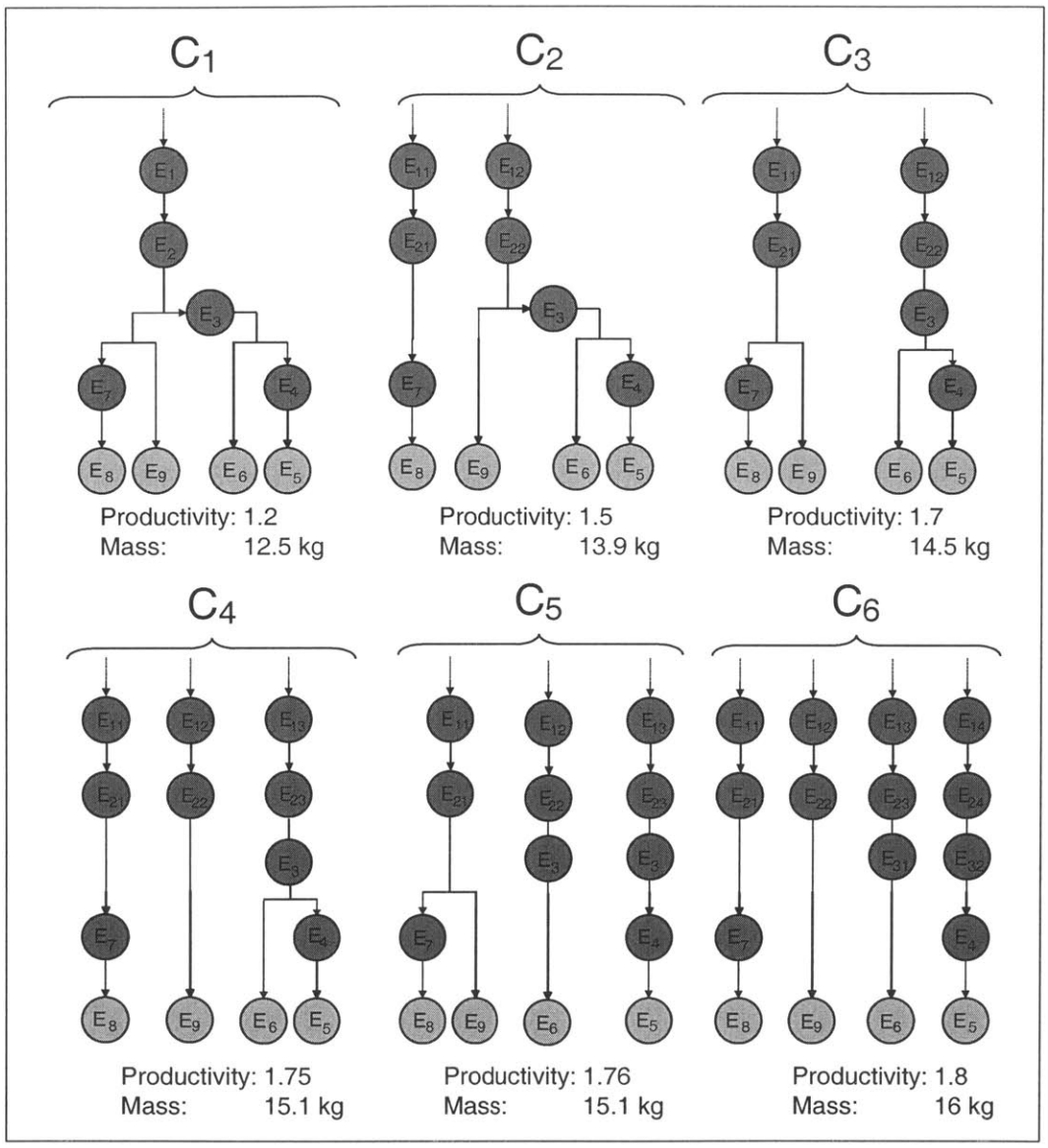
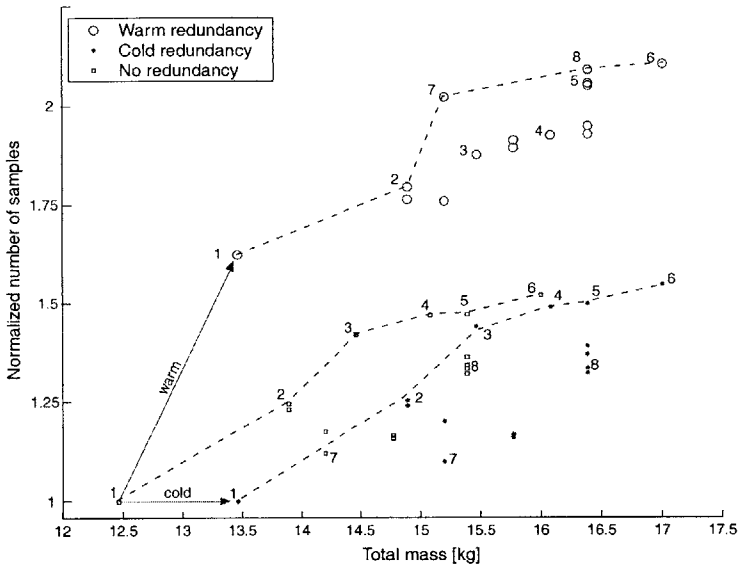


Figure 3-14: Architectures of the Pareto front

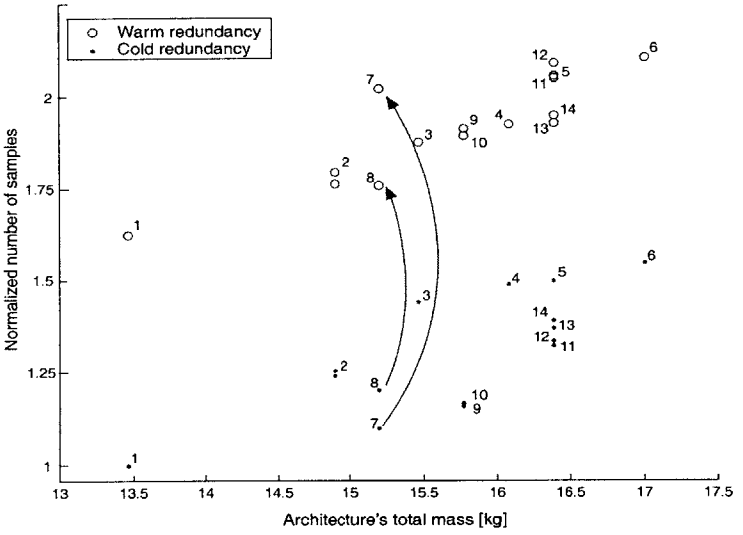
in Figure 3-15(a). The three dashed lines represent the Pareto fronts for each of the three cases.

A simple observation of Figure 3-15(a) shows that adding a cold redundant element does not entail major transformations of the original trade space without redundancies. In this section's example, the effect of the cold redundancy is a translation of all the designs along the mass axis (x-axis). This means the cost of adding a redundant element returns hardly any benefits in terms of added science. The cold redundancy of a single element does not lengthen the lifetime of the SPAT system enough to significantly improve the productivity. In particular, cold redundancy has an homogeneous effect on the Pareto front. The optimal architectures are identical for the no-redundancy and cold-redundancy cases (Figure 3-15(a)).

Interesting phenomena happen, however, when warm redundancy is introduced. First, architectures with warm-redundancy show better performance than the rest of the designs. The Pareto front for warm-redundancy dominates that of no- and cold-redundancy, because the warm redundancy canceled the bottleneck effect on the element E_5. Equation 3.22 shows that, without redundancy, E_5 is the bottleneck. When warm redundancy is introduced, the effective processing time of that element is reduced by half and it is no longer the bottleneck of the system. Second, the introduction of warm redundancy reorders the architectures. Warm redundancy has an heterogeneous effect on the Pareto front. Some optimal warm-redundancy architectures are different from the optimal no- and cold-redundancy architectures. Figure 3-15(b) shows the same trade space as Figure 3-15(a), but it illustrates how the ranking of architectures 7 and 8 switches when warm redundancy is used instead



(a) Architectures with and without redundancy



(b) Warm redundancy reorders the architectures

Figure 3-15: Trade space for architectures with warm and cold redundancies

of cold redundancy. The detail of the two architectures is given below:

$$C_7 = \left\{ \begin{bmatrix} E_1 & E_2 & E_3 & E_4 & E_5 \\ E_1 & E_2 & E_7 & E_8 & \\ E_1 & E_2 & E_9 & & \end{bmatrix}, \begin{bmatrix} E_1 & E_2 & E_3 & E_6 \end{bmatrix} \right\} \tag{3.24}$$

$$C_8 = \left\{ \begin{bmatrix} E_1 & E_2 & E_3 & E_4 & E_5 \end{bmatrix}, \begin{bmatrix} E_1 & E_2 & E_3 & E_6 \\ E_1 & E_2 & E_7 & E_8 \\ E_1 & E_2 & E_9 & \end{bmatrix} \right\} \tag{3.25}$$

Both architectures are constituted of two sets of path. Architecture C_7 isolates the path of instrument E_6, whereas architecture C_8 isolates the path of instrument E_5. In the cases of no- and cold-redundancy, the instrument E_5 has the longest processing time (Equation 3.23). Architecture C_8, which isolates this bottleneck, shows a better performance than architecture C_7. However, when warm redundancy is applied to the instrument E_5, its effective processing time is reduced by half. In this situation the instrument E_6 becomes the bottleneck. Since C_7 isolates the path of the instrument E_6, it performs better than architecture C_8. The example shows how warm redundancy reorders the ranking of the architectures by removing bottlenecks.

3.6 Conclusions

This study is motivated by the need for a mathematical rationale for the development of shared-preparation facilities. This work provides several contributions to the design of SPAT systems. First, it identifies the sample path as a key notion to depict SPAT systems and generate all their possible architectural solutions (Section 3.2.1). Second, it confirms the argument that when there are differences among element processing times, shared-preparation facilities slow down some sample analyses. The more productive architectures are those which isolate the paths with bottlenecks. Regarding the mass metric, centralized facilities are always the most mass efficient and for that reason belong to the Pareto front, whatever the SPAT system modeled. This study uses an arbitrary economy of scale because not enough information was

found to generate one with sufficient accuracy.

Yet, this model can be used to determine what level of economy of scale must be achieved for the centralized architecture to be a better option than the distributed one. For the case of the trade space shown in Figure 3-13, the minimum economy of scale that must be achieved is presented in Figure 3-16. The data point numbered 1

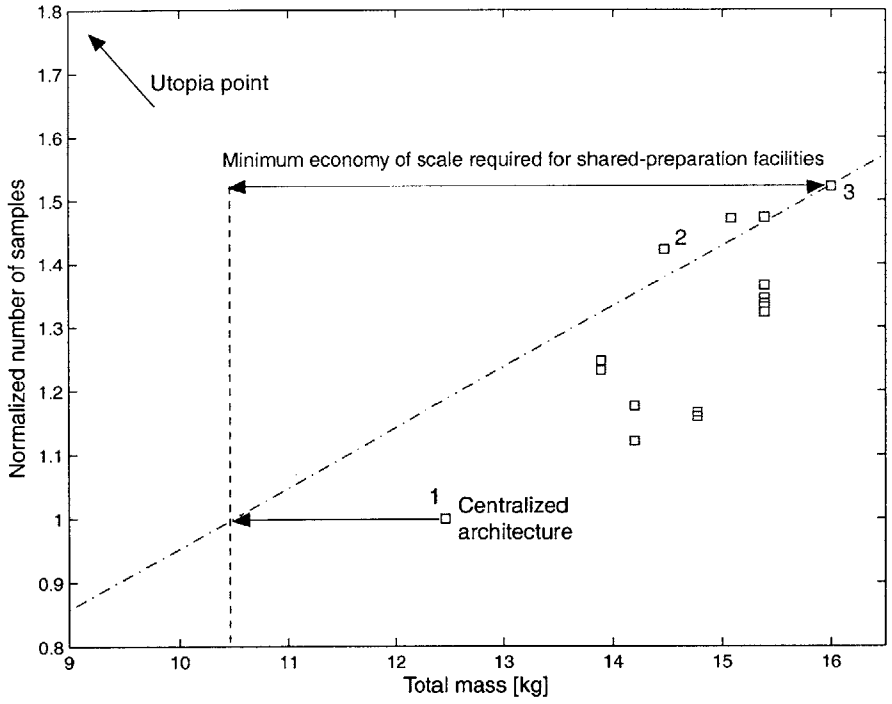


Figure 3-16: Economy of scale

represents the centralized architecture. The one numbered 2 is the architecture that shows the best ratio of number of samples per kilogram of the system. The data point 3 is the distributed architecture. The dash-dot line has a slope equal to the number of samples per kilogram ratio of the distributed architecture. All the data points that are below that line are dominated by the distributed architecture. Hence, the centralized architecture is dominated by the distributed architecture; the economy of scale as currently modeled for that example is insufficient to make the centralized architecture preferable to the distributed one. For the centralized-preparation design to be non-dominated, the economy of scale must lead the data point in the direction of

decreasing mass (parallel to the x-axis) until it is above the dash-dot line. According to that argument, for a 16 kilogram distributed architecture, the centralized architecture must weigh no more than 10.5 kilograms. Thus, the amount of mass saved by sharing the preparation facilities must be at least 34% of the mass of a distributed system.

Chapter 4

Mars Surface Exploration tool description

4.1 Introduction and Motivation

Planetary rovers are a relatively recent technology that has been used only four times so far, twice for Moon exploration with the 1970 and 1971 Russian Lunokhod rovers, and twice for Mars exploration with the NASA 1996 Sojourner and 2003 Mars Exploration Rovers. Still, these few missions were very successful, and consequently promoted rovers as a promising medium suited for further exploration of Mars and other bodies. This is why more of them are scheduled to follow the tracks of Sojourner and MER on Mars. The National Aeronautics and Space Administration (NASA) has plans for two rovers, the 2009 Mars Science Laboratory (MSL) and the Mars Sample Return[1], and the European Space Agency (ESA) intends to launch its ExoMars rover in 2011.

Our urge for a better understanding of the Martian environment drives the need for more missions to the planet. Hence, after each mission's success, scientific objectives are more specific as well as more ambitious for the following one. The engineering challenge is then to translate these ever-demanding objectives into improved rover capabilities. Designers must understand what technologies need to be developed

[1]Current plans call for the first sample return mission to be launched no earlier than 2014.

today to ensure the success of tomorrow's rover missions.

The increasing site-to-site separation requirement illustrates this statement. To improve the variety of samples collected at each site on Mars, the site-to-site separation will be increased from 100 meters, the current requirement for MER, to 3 kilometers for MSL. The engineering problem is to identify and develop the right technological solutions that will upgrade the rover's traverse to a 3 kilometer capability. One solution is to increase the size of the rover so that its ability to traverse rocks is improved. For example, the rover diameter was increased from 30 centimeters for Sojourner to 120 centimeters for MER (Figure 4-1). Another solution is to upgrade

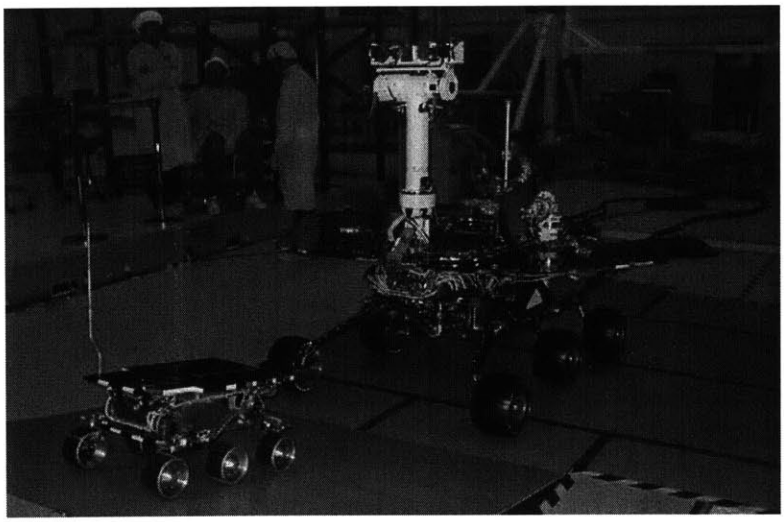


Figure 4-1: Comparison of the Sojourner and MER architectures, courtesy of JPL

the autonomy so that the rover's ability to navigate itself is improved. There is an alternative approach to find the right balance between these two solutions. Either the mission team starts the process of developing both technologies to, in the end, select the best from empirical evidence, or the team uses simulation tools to model the cost-benefits of building a bigger rover, as opposed to making it smarter, and then develops the better technology of the two. The latter approach, if less rigorous, allows analysis of competing alternatives by performing integrated calculations of system performance, mission cost, risk, and science metrics, and is also cheaper and quicker than the former method. These assets are most valuable for a mission's

early-design phase (pre-phase A, defined in Section 2.1.2) where budget and schedule are tight. In summary, rovers are complex systems, with many inherent trade-offs, that would benefit from an adequate trade space exploration model.

4.2 The Mars Surface Exploration tool

4.2.1 Objectives of the study

Among other capability shortfalls, NASA's Engineering for Complex Systems program identified the agency's limited system and trade space analysis capabilities. Its System Reasoning and Risk Management division has recently called for tools that better support risk analysis, design robustness, mishap modeling, and system trade-offs throughout the entire life cycle of the program [27]. This present work, resulting in the Mars Surface Exploration (MSE) framework, represents a major step in the realization of such tools.

This study considers one particular example of complex systems, the Mars rover missions, for which it builds a systems engineering design tool, MSE. As explained in the former section, there is a clear need to conduct architectural trades early in the lifetime of a mission to identify those designs that best meet the needs of the stakeholders, who are in this case the science community and the tax-payers. MSE is intended to answer that need by providing mission designers with a full and astute picture of the trades for future Mars rover missions in order to enhance their system-level decision making. Other rover models already exist that take the approach of interconnecting sophisticated software design environments to conduct detailed analyses of a particular architecture. What these techniques gain in fidelity, they lose in breadth and agility and MSE's approach is a good complement to these techniques. In return, these other techniques allow validation of MSE's models at various points of the design space.

The system-level accuracy of MSE is commensurate with the needs of the early-design phases for which it is targeted. The tool's capabilities enable the analysis of

intricate trade-offs such as the cost-benefits of autonomy, and the need for nuclear power systems as opposed to solar panels. Although MSE does not capture risk aspects yet, its architecture is conceived to facilitate future inclusion of risk analysis. It has a modular architecture articulated around a sequence of three blocks representing mission requirements, rover subsystems and design, and trade analysis tools (Section 4.3.1). Therefore, risk can be added later, at the system and subsystem levels. Another interesting aspect of MSE's modularity is that other kinds of explorers, such as blimps and airplanes, can also be broken down into the same generic subsystems, such as mobility or communications. Therefore, more than a rover-specific tool, MSE is a framework that can inspire the creation of similar design tools for application to other complex systems.

MSE has a science-driven architecture that is divided into three segments. First,

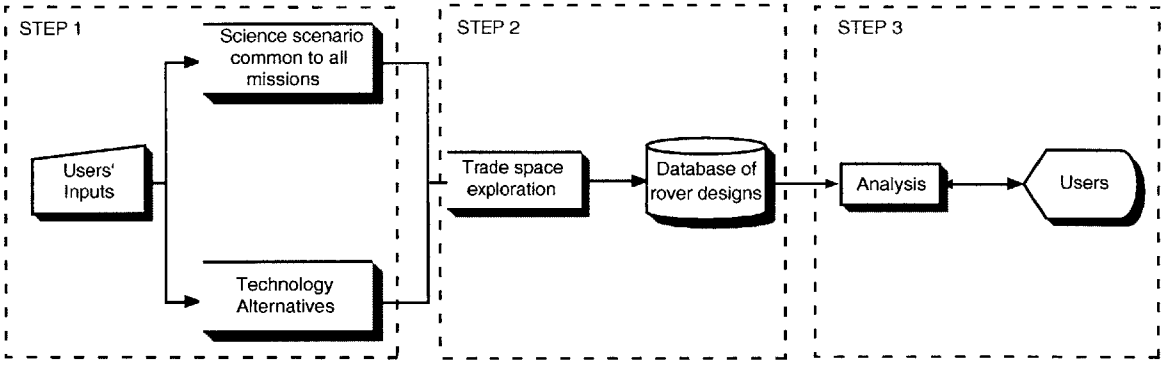


Figure 4-2: MSE's framework

MSE captures the user's inputs regarding the science scenario and the technology alternatives for designing the rover. The latter defines the types of technology alternatives, and thereby, the system-level trade-offs that are explored in the second step. The iteration of the design process for all the combinations of technological options generates a vast array of architectures which are saved in a database. Finally, MSE helps users to analyze this database by ranking mission architectures according to their preferred metrics (see Chapter 5 for detailed MSE analysis features).

In summary, the rover design is a science driven, systems engineering design tool targeted for use by mission designers in pre-phase A. The tool supplies them with a

multidimensional view of the trade space for future missions, upon which they are able to make grounded and traceable decisions on system-level design and development of new technologies. Furthermore, as missions are completed, data can be folded back into MSE to make it stronger.

4.2.2 Tool's characteristics

The tool exhibits six major features:

1. *Validation*: MSE's fidelity is adequate for its use during early-mission design phases. The tool's models are validated at system and subsystem levels against existing data and mission point designs. The models are open-source, so that they can be checked and updated by users.

2. *Usefulness*: The trade space available for investigation is large and provides insight to mission designers. The tool captures all subsystems and their interactions to bring out relevant trade-offs. Notably, the design vector provides ranges for mission lifetime, solar versus nuclear power sources, rover wheel size, and levels of autonomy for various mobility activities. MSE's applications can also be extended to the study of multi-rover missions such as MER. In addition, it provides a way to quantify the monetary value of developing higher levels of autonomy for future rover missions.

3. *Rapidity*: The model's code, written in MATLAB, is computationally expeditious in order to allow for efficient exploration of large trade spaces.

4. *Usability*: A graphical user interface enables those without expert assistance or practiced knowledge of MATLAB to use MSE. In particular, launching a customized trade space exploration and analyzing its results is straightforward.

5. *Modularity*: MSE architecture reflects the subsystems configuration of a rover. The subsystem modularity allows the user to easily update models as higher fidelity ones become available.

6. *Versatility*: The tool is not restricted to Mars mission applications. The tool enables the analysis of any planet or moon exploration mission by simply updating environmental parameters. For example, MSE could be applied to lunar exploration and resource utilization missions.

The following sections will demonstrate in detail how MSE is conceived to fulfill these requirements.

4.2.3 Context of the study

The work presented in this chapter was completed as part of the Spring 2003 Space Systems Engineering course in the the Department of Aeronautics and Astronautics at the Massachusetts Institute of Technology. MSE was first created by the fourteen students of the class. The rover system was broken down into seven subsystems, each to be modeled by a pair of students. The author is greatly indebted to the members of the class for the completion of the MSE tool, Christopher Roberts and Julie Wertz (Instruments), Ian Garrick-Bethell and Erisa Hines (Acquisition), Stephanie Chiesi and Jessica Marquez (Environment), Kalina Galabova and Roshanak Nilchiani (Power), Babak Cohanim and Tsoline Mikaelian (Communications), Edward Fong and Barry White (Autonomy), and finally my teammate Mark Hilstad (Rover).

The class benefited greatly from cooperation with outside industries. MSE architecture is based on rover design guidelines provided by Charles Whetsel from the Jet Propulsion Laboratory (JPL) [58]. Furthermore, a large part of the engineering database used for modeling and validation was provided by Allen Chen and Jaret Matthews, also from JPL, and Joe Parrish, president of Payloads Systems Inc. The author wishes to thank them for their valuable help.

The following sections of this chapter describe the methods used by each student group to model their respective subsystems. These sections are in large part based on a report, Rapid Modeling of Mars Robotic Explorers [28], that the class wrote at the end of the project.

The academic context in which MSE was created drove some of its features. First,

the rover system was divided into seven submodules, each one to be managed by a pair of students. Second, the scope of the project was initially bounded so that it could be completed in a semester, the duration of the class. These restrictions include limiting the overall modeling detail to system-level accuracy, which is consistent with the end use of MSE. This tool is to be used in early conceptual design to examine system-level trades, not to design rover hardware. Additionally, the tool focuses on surface operations, and prior mission phases (launch, cruise, entry, descent and landing) are not captured. Some design options are not investigated, such as the presence of an active lander. MSE primarily models missions, such as the 2009 Mars Science Laboratory, which involve a single rover traversing large distances to explore the Martian surface. However, MSE's framework is such that the single-rover trade space can be extended to analyze multi-rover missions, such as the Mars Exploration Rovers, which involve multiple rovers working independently on the surface of Mars (Section 5.5). The modeling approach and tool architecture are presented in the next section.

4.3 MSE's architecture

4.3.1 Approach

The MSE tool is organized into three segments as shown in Figure 4-3 which is an expansion of Figure 4-2. The first segment, called Inputs, is a front-end interface for the users. There, the users define mission requirements, as well as trade space exploration features (Section 4.3.2). These inputs are then carried to the Rover Modeling segment, which performs an automatic exploration of the trade space defined in the previous step. This segment is an integration of modules that model complementary parts of the rover design (Sections 4.4 to 4.10). Eventually, the exploration results in a rover design database that is handed to the *Analysis* segment. The *Analysis* is the back-end interface with the users who specify on which utility function to project the trade space (MSE's analysis capabilities are explained in Chapter 5).

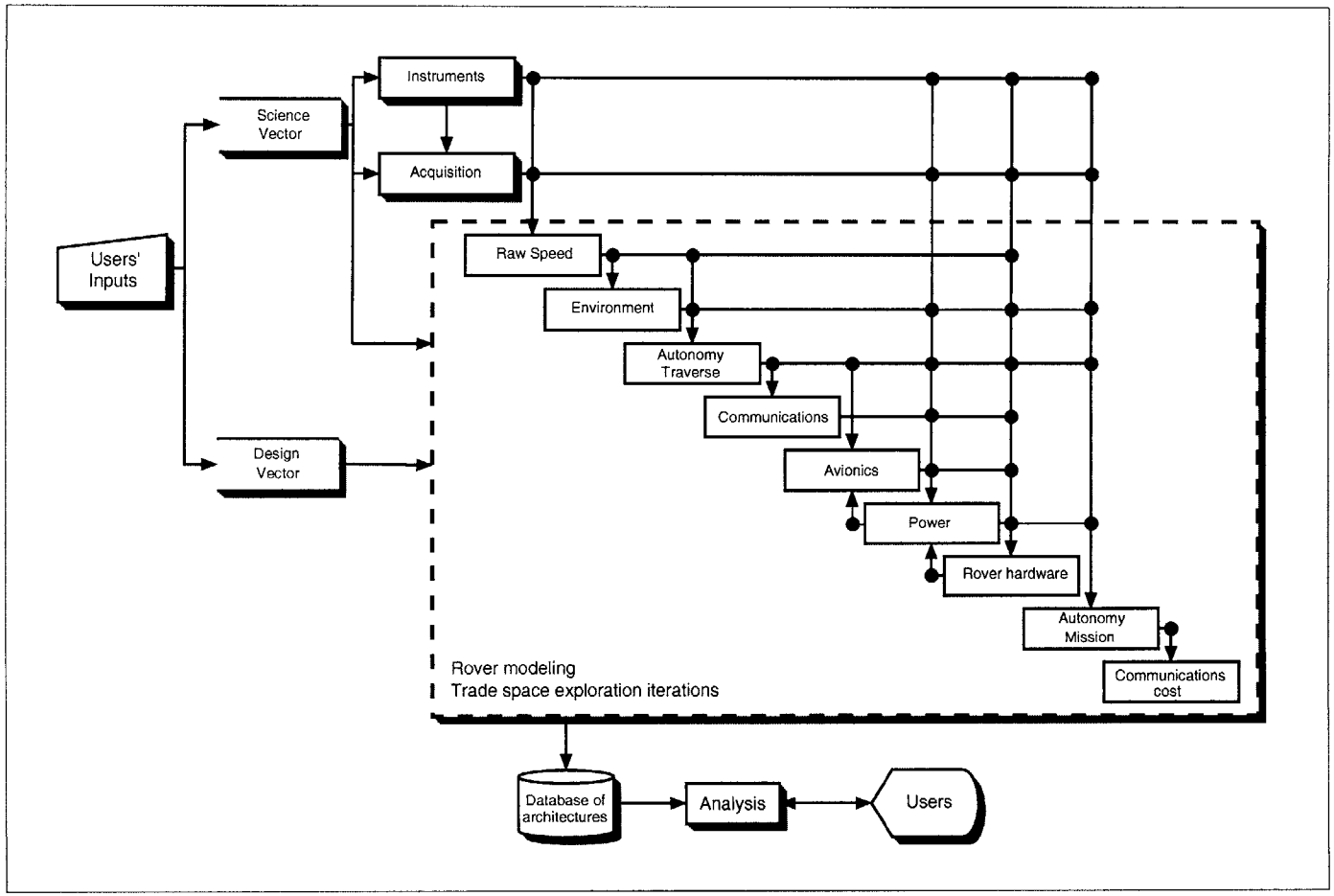


Figure 4-3: MSE's N^2 diagram

4.3.2 MSE inputs

The objective of the tool is to explore rover technology alternatives, given a science scenario defined in terms of payload, geography, and landing date. Therefore, there are two aspects to the exploration, the science requirements that are common to every design, and the technology solutions that differ from design to design. And so the tool is articulated around the definition of two vectors, the *science vector* and the *design vector*. The former provides the science requirements information, the latter contains the allowable values for all design variables. These two vectors then feed into the seven subsystems that model the rover design (Figure 4-3).

Science Vector

The science vector stores the information about what is constant across all the trade space, i.e. science and navigation payloads, and all site-specific information. The

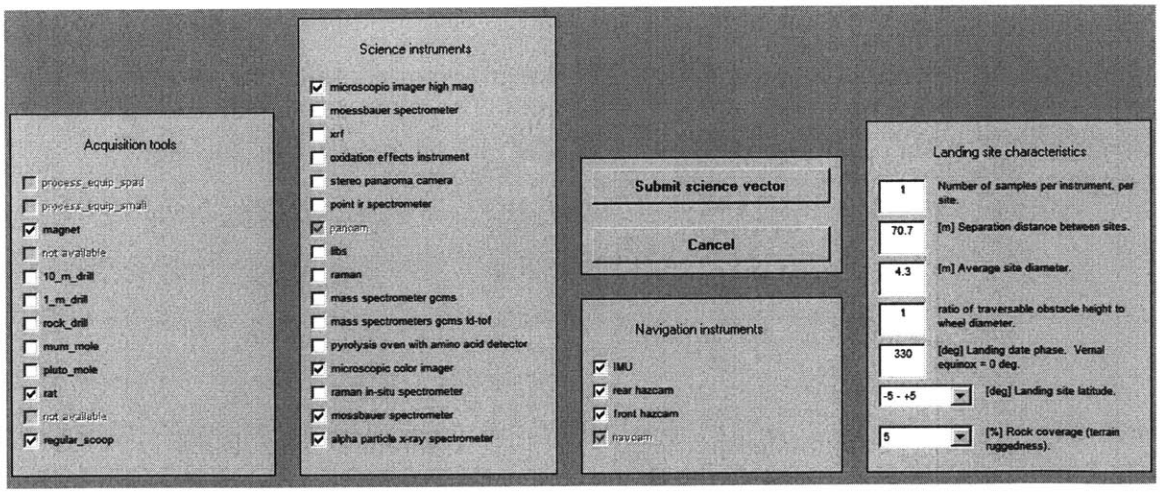


Figure 4-4: Graphical interface for the science vector

entries of the science vector are detailed below. The science and acquisition entries are addressed subsequently in their respective subsystems. The environmental entries are addressed in the Environment section. Figure 4-4 shows the graphical user interface

associated with the science vector's parameters.

$$V_{Science}= \begin{bmatrix} \text{science intruments} & [-] \\ \text{acquisition instruments} & [-] \\ \text{navigation instruments} & [-] \\ \text{samples per site} & [-] \\ \text{site to site separation} & [\text{m}] \\ \text{site diameter} & [\text{m}] \\ \text{rock coverage} & [-] \\ \text{landing site latitude} & [-] \\ \text{landing site hemisphere} & [-] \\ \text{areocentric longitude} & [^0] \end{bmatrix} \tag{4.1}$$

Design Vector

A rover design is uniquely defined by its particular set of design variable values. There are as many design points in the trade space as there are combinations of design-variable values (Chapter 2.1.3). The design vector contains the set of allowable values for all of the design variables listed below.

$$V_{Design}= \begin{bmatrix} \text{lifetime} & [\text{sol}] & \text{continuous} \\ \text{wheel diameter} & [\text{m}] & \text{continuous} \\ \text{number of computers} & [-] & \text{continuous} \\ \text{power source} & [-] & [\text{solar, RTG}] \\ \text{communication type} & [-] & \text{combinations of DTE} \\ & & \text{LMO, and HMO} \\ \text{long distance autonomy} & [-] & [\text{A1, A3}] \\ \text{short distance autonomy} & [-] & [\text{A1, A3}] \\ \text{acquisition autonomy} & [-] & [\text{A1, A3}] \end{bmatrix} \tag{4.2}$$

Lifetime and wheel diameter are continuous variables. The number of computers is also handled as a continuous variable reflecting the notion of computational efficiency

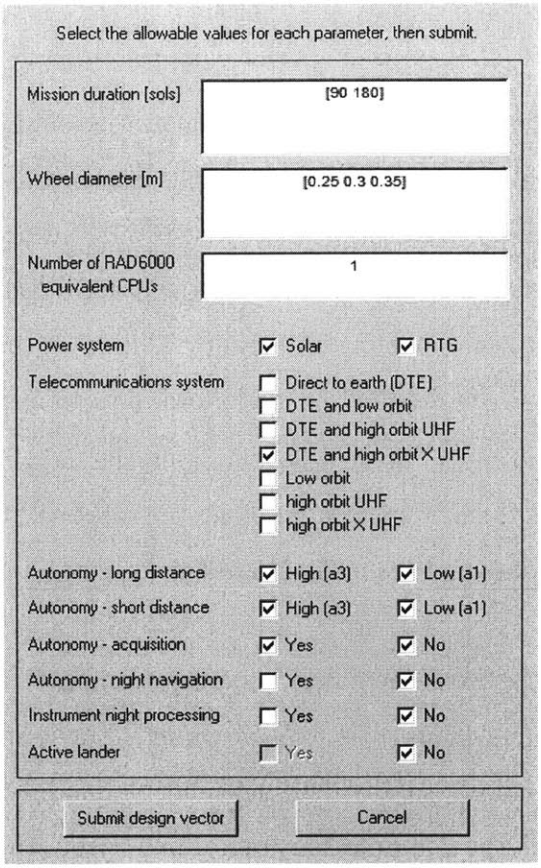


Figure 4-5: Graphical interface for the design vector

rather than the actual number of computers on board the rover. The other design variables have inherently discrete levels. The power system variable has two levels corresponding to solar and nuclear (RTG) systems. All autonomy variables have two levels, high and low autonomous capabilities, A3 and A1, respectively. The communication-type variable has seven levels, which are combinations of direct to Earth (DTE) with low and high altitude satellite relays (LMO and HMO, respectively) using X-band and UHF. In the current version of MSE, autonomous navigation, night processing of instruments, and active lander options are not yet modeled. The user defines the design vector entries via the graphical interface shown in Figure 4-5.

Rover Modeling

The complex rover system is subdivided into smaller disciplinary models called subsystems. There is one subsystem for each of the following areas: surface environment, science instruments, sample acquisition methods, rovers, autonomy, communication and power (Figure 4-3). These subsystems interface with each other and are integrated into an overall rover modeling framework. Modeling methods, assumptions and validation techniques are detailed in the following sections (Sections 4.4 to 4.10)

The N^2 diagram, shown in Figure 4-3, represents the data flow between the nine modules shown. The modules are executed along the diagonal, a connection between two modules above the diagonal is a feedforward flow, a connection below the diagonal is a feedback flow. The execution order of the modules is arranged to minimize the number of feedback loops and, therefore, maximize computational efficiency. Feedback loops still remain unavoidable between *Avionics*, *Power* and *Rover* subsystems. Some subsystems are actually subdivided into multiple modules along the execution line in order to minimize the computations performed within the feedback loops. For example, the modules *Raw Speed* and *Rover Hardware* are both modeled by the *Rover* subsystem.

A full-factorial method is used to search the whole trade space. This means design and performance characteristics of every architecture within the trade space are assessed. The method is computationally expensive. However, full-factorial search has the advantage that the trade space only needs to be explored once. After the search is performed, the users have an available design database on which they can apply as many utility functions as wished, using the *Analysis* segment.

Analysis

The trade space exploration results in a database of rover designs which contains the rover's hardware characteristics as well performance figures. These hardware characteristics include total mass, power requirements, and dimensions from every subsystem; additionally, subsystems provide the database with other relevant features

that are more specific. The performance figures include among others the number of samples collected, distance traversed, and amount of telemetry. From these features, the analysis module constructs multiple utility functions that are used to compare and rank designs. As noted above, the full-factorial method permits the examination of a mission trade space with a multitude of utility functions after only a single run of the Rover Modeling calculations. A thorough demonstration of the analysis capabilities of MSE is provided in the next chapter (Chapter 5).

In summary, through its trade space customization and thorough analysis capabilities, MSE allows mission designers to identify the designs that best meet their requirements.

4.4 Instruments

4.4.1 Responsibilities

The instruments on board the rover are selected by users during the creation of the science vector. The user must choose the mission's instruments among the ones presented in Table 4.1. The responsibility of this subsystem, vis a vis the users, is to offer a large pool of instruments so that they can customize pertinent science payloads. Vis a vis the rest of the subsystems, *Instruments* must provide all relevant characteristics of their instruments.

4.4.2 Instrument database

The instrument database is built according to NASA's objectives for searching ancient or extant exo-biology, and studying climate and geology. The list of the fifteen available instruments (Table 4.1) includes stand-alone instruments, while others are combination packages of several instruments. The instrument database is built primarily on information coming from the MSL science team, complemented with manufacturers' data on the Moessbauer spectrometer and the oxidation effects instruments. The database can easily be expanded to include new information. For each instrument,

Table 4.1: List of instruments currently included in the *Instruments* database

Remote Sensing	Contact Suite	Analytic Laboratory	
Point IR spectrometer with scene-rastering capability	Alpha particle X-ray spectrometer	Microscopic imager VIS, high magnification-1μm pixel, 6 color	X-ray fluorescence
Stereo panorama camera 4 color + calibration target	Mössbauer spectrometer	Moessbauer spectrometer	Mass spectrometers: GCMS + LD-TOF integrated instrument package
Laser induced breakdown spectrometer (LIBS)	Raman spectrometer, in-situ remote sensing	Oxidation effects instrument	Mass spectrometer: GCMS + evolved gas analysis
	Microscopic color imager, 30μm resolution	Pyrolysis oven integrated with GCMS, amino acid detector	Raman - analytical laboratory

The chief entries of the database are mass, power requirement, dimensions, cost and operating temperatures.

4.4.3 Assumptions

The *Instruments* model makes three main assumptions. First, it does not limit the number of samples that an instrument can process or the size of the sample obtained. Second, based on several MSL meetings attended by the instrument team, the default cost of an instrument is set to $10 million without any distinction. Finally, the mass of each instrument is doubled for the following two reasons. First, extra mass needs to be added to the mass of an instrument to account for the thermal, structure, and cabling hardware required to protect and support the instrument. Second, mass estimates are in several cases obtained from the instrument developers, hence they are optimistic.

4.4.4 Validation

MSE's instrument database is compared to instrument mass rules of thumb provided by Charles Whetsel [58] in Table 4.2. Results do not match perfectly, one by one,

Table 4.2: Comparison between database mass values and C.Whetsel's estimates

	Remote Sensing		Contact payload arm + instruments	Analytic Laboratory		
	Stereo Panoramic Imager	IR Spectrometer		Pyrolisis oven, GCMS, XRD, XRF	Raman Mossbauer	Oxidation Effects Instrs
C.Whetsel estimate [kg]	5	10	10	10	5	5
MSE estimate [kg]	0.7	6.4	18.4	14.2	3.4	3

but are consistent on average. Based on Charles Whetsel's data, the average mass of the total instrument payload is estimated at 7.5kg, which is very close to MSE's estimation of 7.7kg. If more accurate information becomes available, the instrument database can easily be updated.

4.5 Acquisition

4.5.1 Responsibilities

Acquisition tools are, like instruments, selected by users via the science vector. Therefore, the *Acquisition* module's responsibilities, as to the users and the rest of the subsystems, are the same as the ones mentioned for the *Instruments* module.

4.5.2 Acquisition database

Acquisition tools are defined as any hardware that collects samples from the environment, or prepares them to be analyzed by instruments. Therefore, the acquisition database is built consistently with the instrument database. There is no exclusive

link, however, between one instrument and one acquisition tool. Some instruments accept a plurality of sample types, and some acquisition tools serve multiple instruments. Automatic selection of acquisition devices based on an instrument payload is, as a consequence, difficult to implement. It was decided instead to rely on the user's expertise. Users are assumed to be knowledgeable about functions and characteristics of science instruments and their associated acquisition tools. The selection of acquisition hardware is totally free, even if it does not suit the chosen instruments and vice versa.

The tool database was compiled from modern planetary sample acquisition device that are flying on missions, or are in advanced development. Future technologies for sample collection, such as deep drilling machines, are not modeled because few complete designs exist. The eight tools available for selection are shown in Table 4.3 with their mass and power features, along with manufacturer, development status, and source information. As new acquisition tools gain maturity, they can be added to the database.

4.5.3 Validation

Since the values obtained from each of the tools are usually from existing designs, the data is mostly self-validated and has no margin. When no data could be found on a particular characteristic, conservative estimates were made instead.

4.6 Environment

4.6.1 Responsibilities

The *Environment* module provides other subsystems with information regarding the operating environment where the rover is located. Environmental data required by other subsystems includes solar irradiance, hours of sunlight per day, surface temperature ranges, and obstacle occurrence. Methods to calculate and validate each of these features are given in detail in the subsequent paragraphs. The *Environment*

Table 4.3: Database of acquisition tools

Tool	Mass [kg]	Power [W]	Manufacturer / Developer	Status	Source
Rock Abrasion Tool	0.77	10	Honeybee Robotics	Flying on MER	[9]
Rock Corer	2.7	10	Honeybee Robotics	Tested on FIDO rover	[35]
Pluto Mole	0.86	3	DLR	Flying on Beagle 2	L.Richter (DLR)
MUM mole	3.5	10	NASA Ames	In development	[51]
1meter drill	15	100	Honeybee Robotics	Tested in lab	[10]
10meter drill	99	100	Honeybee Robotics	Tested in lab	[10]
Magnets	0.06	N/A	-	Flying on MER	MER webpage
Soil Scoop	0.5	20	-	-	A.Chen (JPL)
Sample Processing Hardware	10	25	JPL-SPAD study results	Concept design for MSL	A.Chen (JPL)

module works from indications by the users about the landing site geography. These indications include landing date, approximate latitude range and rock coverage of the landing site (all of which are entries in the science vector).

4.6.2 Modeling Assumptions

The map of Mars is cut in broad latitude bands in order to keep the tool as generic as possible (Table 4.4). Landing site characteristics, like temperature, are not modeled as longitude-dependent. Consequently, their values are greatly diluted. Limited availability of data forces three other assumptions to be made. First, polar regions are not considered among the possible landing regions. Second, rock coverage is estimated based on data from previous missions and reporting only. It is important to note that most of these missions have occurred in low rock and crater density areas; rock densities in high rock coverage zones are, therefore, calculated by extrapolation.

Table 4.4: Representative latitudes and albedo for each latitude band

Latitude Band (degrees)		Representative latitude (degrees)	Average Albedo Used
Equatorial	[-5 , 5]	0	0.2277
Low South	[-5 , -40]	-20	0.2277
Low North	[5 , 40]	20	0.2277
High South	[-40 , -80]	-60	0.2167
High North	[40 , 80]	60	0.2534

Third, no data is found for incorporating local slope estimation or crater avoidance modeling in the module.

Other approximations are made in calculating solar irradiance on the surface of Mars. It is calculated only for horizontal surfaces, and an average albedo is used for each latitude band of the science vector (Table 4.4). The optical depth of the atmosphere used in the calculations is set to 0.5, which corresponds to a typical clear day [11]. Additionally, the highest solar irradiance for a particular sol is assumed to happen at high noon.

4.6.3 Environment Models

Solar Irradiance

The flow of calculations for the determination of the solar irradiance is represented in Figure 4-6. This paragraph summarizes the equations used for these calculations. Equations for solar irradiance on horizontal surfaces were taken from Appelbaum et al. papers, [11] and [12]. Global irradiance, G_h, is determined in Equation 4.3, where G_{ob} is the instantaneous beam irradiance of Mars and $f(z, \tau, al)$ is the normalized net solar flux function. Values of this function, which take into consideration the general circulation model (GCM) for Mars, are presented in tabular form in Appelbaum et al. [11]. The function is dependent on three parameters, optical depth, τ, solar zenith angle, z, and surface albedo, al.

$$G_h = G_{ob} \cos z \frac{f(z, \tau, al)}{1 - al} \tag{4.3}$$

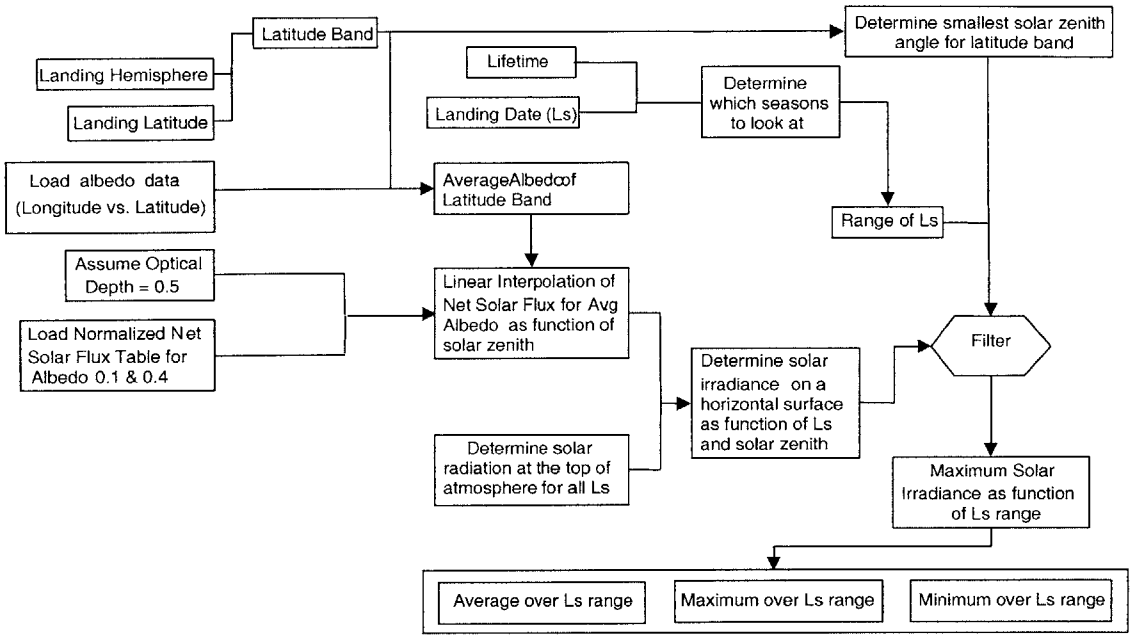


Figure 4-6: Solar irradiance flow diagram

The instantaneous beam irradiance of Mars, G_{ob}, which is the solar radiation on top of the atmosphere, is itself governed by the following equation [11].

$$G_{ob} = 590 \frac{(1 + e \cos(L_s - 248))^2}{(1 - e^2)^2} \tag{4.4}$$

The solar radiation G_{ob} is expressed as function of two celestial parameters. The first one is Mars' orbital eccentricity, e, whose value is 0.093377. The second is the areocentric longitude, L_s, which is a measure of Mars' position on its orbit around the Sun. The orbit's perihelion corresponds to an areocentric longitude of $L_s = 248^o$. At this position, Mars is closest to the Sun and the irradiance is at its maximum, in agreement with Equation 4.4.

The normalized net solar flux is a three parameter function. The first one is the optical depth, τ, whose assumed value is 0.5 (section 4.6.2). The second is the albedo, al, whose values are averages from the tables in Appelbaum et al. [11] as shown in Table 4.4. The third is the solar zenith angle, z, whose value depends on the latitude, ϕ, and the areocentric longitude, L_s, according to Equations 4.5 and 4.6. With the

assumption that solar irradiance is at a maximum at high noon, ω is set to 0. The declination number δ, that appears in the expression of the zenith angle, is directly given as a function of the areocentric longitude in Equation 4.6.

$$\cos(z) = \sin(\phi)\ \sin(\delta) + \cos(\phi)\ \cos(\delta)\ \cos(\omega) \tag{4.5}$$

$$\sin(\delta) = \sin(24.936^o)\ \sin(L_s) \tag{4.6}$$

Hours of sunlight per sol

The number of hours of sunlight per sol on the surface of Mars, $H_{Sunlight}$, is calculated using the following formula [12].

$$H_{Sunlight} = \frac{2}{15}\ \cos^{-1}(-\tan(\delta)\ \tan(\phi)) \tag{4.7}$$

The notations are consistent with irradiance calculation notations. The declination, δ, is given by Equation 4.6, and the latitude, ϕ, is a user input via the science vector.

Temperature

Temperature data utilized in the *Environment* module comes from the Mars Climate Database, made available to the public by the Laboratoire de Météorologie Dynamique of the Centre National de la Recherche Scientifique in Paris [1].

Rock density characterization

Rock density characterization serves the *Autonomy* module to compute the rover's mean free path. These calculations require the knowledge of the function $N(D)$, which provides the cumulative number of rocks expected to be greater than some size D. For mean free path calculations, D is the size of the largest rock that the rover can drive over. It is expressed as a multiple of the wheel size. For instance, a rover equipped with a rocker-bogie suspension is able to handle rock sizes from one to one and a half times its wheel size.

The relation between N and D is

$$N(D) = L \ \exp^{-sD} \tag{4.8}$$

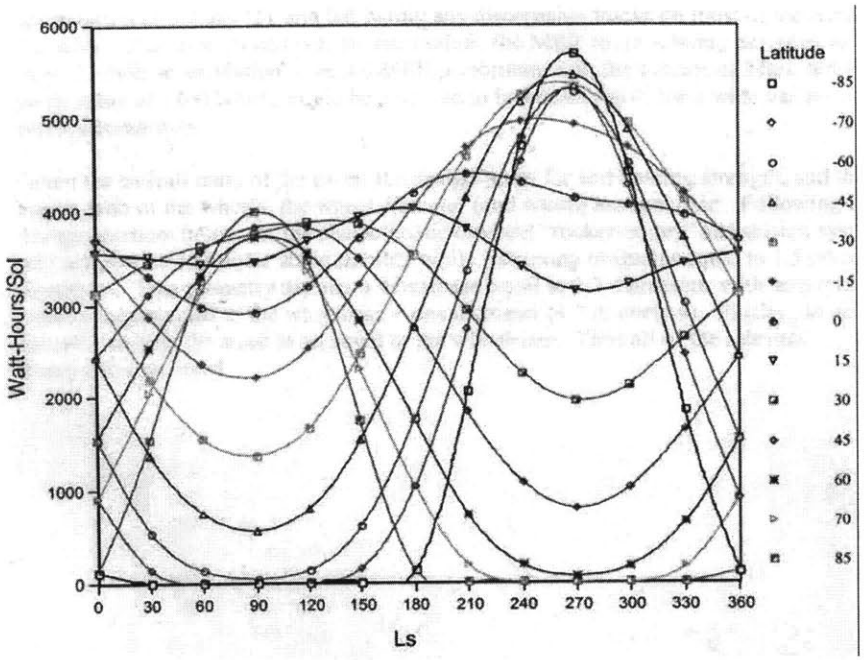
Where L is the total number of rocks of all sizes per square meter and s an exponent based on L. Numerical data collected from the previous Martian landing sites, as well as sample rock fields on Earth, are used to determine equations for L and s based on the total rock coverage k (Equation 4.9) [32, 33]. The landing site's total rock coverage is provided by the user during the science-vector acquisition.

$$\begin{aligned} s &= 2.28 + \frac{0.055}{k} \\ L &= -4.28\ k^2 + 11.54\ k + 1.36 \end{aligned} \tag{4.9}$$

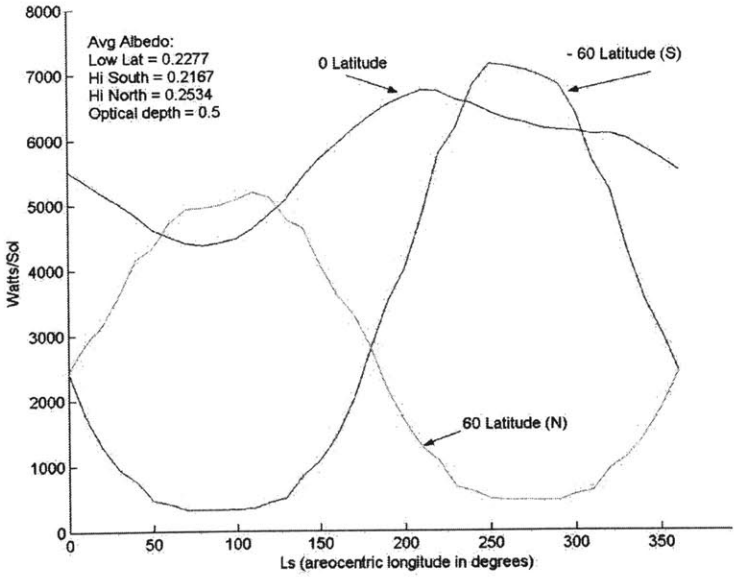
Equation 4.8 is then numerically integrated in small diameter bins to determine the number of rocks greater than the largest traversable rock that the rover would encounter per square meter.

4.6.4 Validation

The hours of sunlight and temperature models do not need specific validation as they come from reliable sources. The irradiance model is compared to a graph provided by Charles Whetsel [58] in Figure 4-7(a). This graph plots the areocentric longitude against the daily total solar energy flux impinging on a horizontal surface at an optical depth of 0.5. In comparison to that graph, the irradiance model overestimates the total amount of watt-hour per sol (Figure 4-7(b). It seems, though, to be a systematic inaccuracy because the functions' wave shapes are replicated closely. The first explanation for the discrepancy of the results is that the MSE model does not capture the change in solar irradiance during a sol. The Sun is assumed to be always at high noon. Additionally, Charles Whetsel's document does not comment on which albedos were used for creating the graph. Another explication for the results' discrepancy is that the document is based on a global circulation model antecedent to the one used



(a) Charles Whetsel's model



(b) MSE Environment's model

Figure 4-7: Irradiance model validation

in Appelbaum's calculations.

Regarding the rock density model, a plot of the cumulative fractional area covered with rocks larger than a given rock size, plotted versus that same rock size, is used for validation. The left panel of Figure 4-8, shows this plot as generated by the *Environment* model for rock coverage percentages ranging from 5% to 50%, the right panel is published data from JPL. Both plots match with satisfactory fidelity.

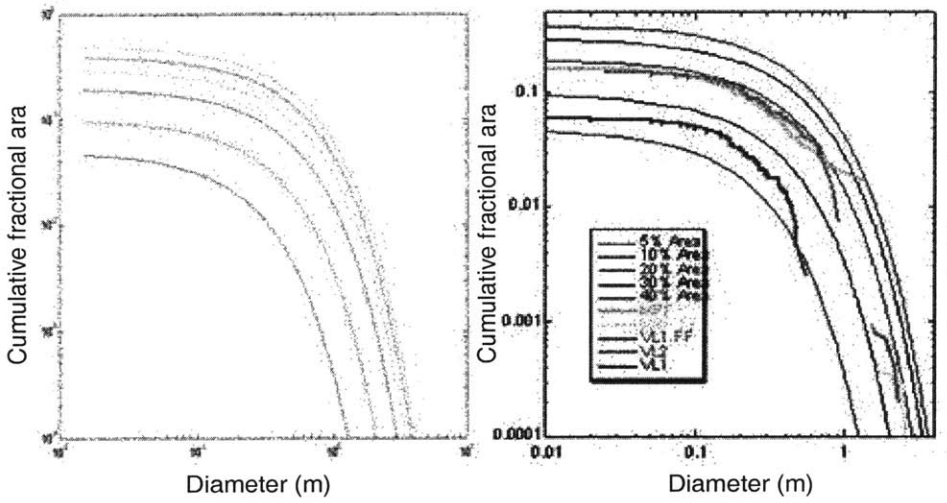


Figure 4-8: Rock density model validation

Thanks to its modular architecture with an independent *Environment* module, MSE enables the study of rover missions in other planetary environments, provided that the other planetary models supply all environmental outputs, described in this section.

4.7 Rover

4.7.1 Responsibilities

The primary purpose of the *Rover* module is to design the structure, mobility, and thermal components of the rover (Figure 4-9). Important outputs of this module are hardware size and mass, as well as raw speed and power requirements of the

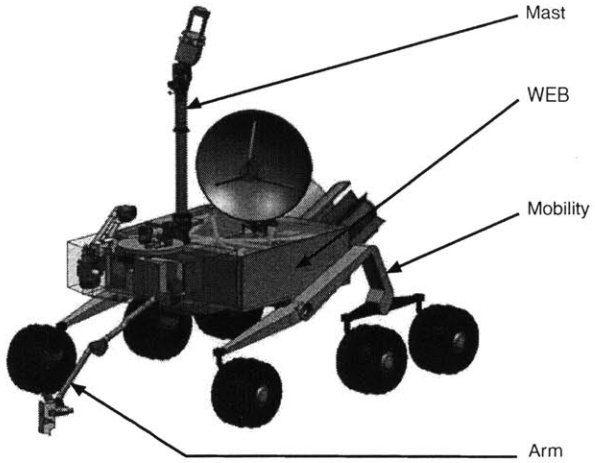


Figure 4-9: Example of rover design [26]

rover. The *Rover* module is subdivided into three submodules (Figure 4-10), namely *Structure*, *Mobility* and *Thermal*, whose functionalities are described below.

Structure

The structure model is in charge of sizing three components of the rover hardware, the arm, the mast and the Warm Electronics Box (WEB).

Arm and mast The arm is a jointed appendage mounted in front of the WEB, onto which acquisition tools and science instruments are attached. The mast is a vertical appendage mounted on top of the WEB, onto which navigation and science instruments are attached. The algorithm currently allows for only one mast and one arm on a rover. Both arm and mast models are very low fidelity; they are essentially placeholders in the current version of MSE.

Warm electronics box The main body of the rover is called warm electronics box (WEB). It is more than the rover's chassis, it houses and protects on-board electronics and batteries from the temperature extremes of the Martian surface. The WEB is designed to meet packaging and structural requirements. Its dimensions are driven

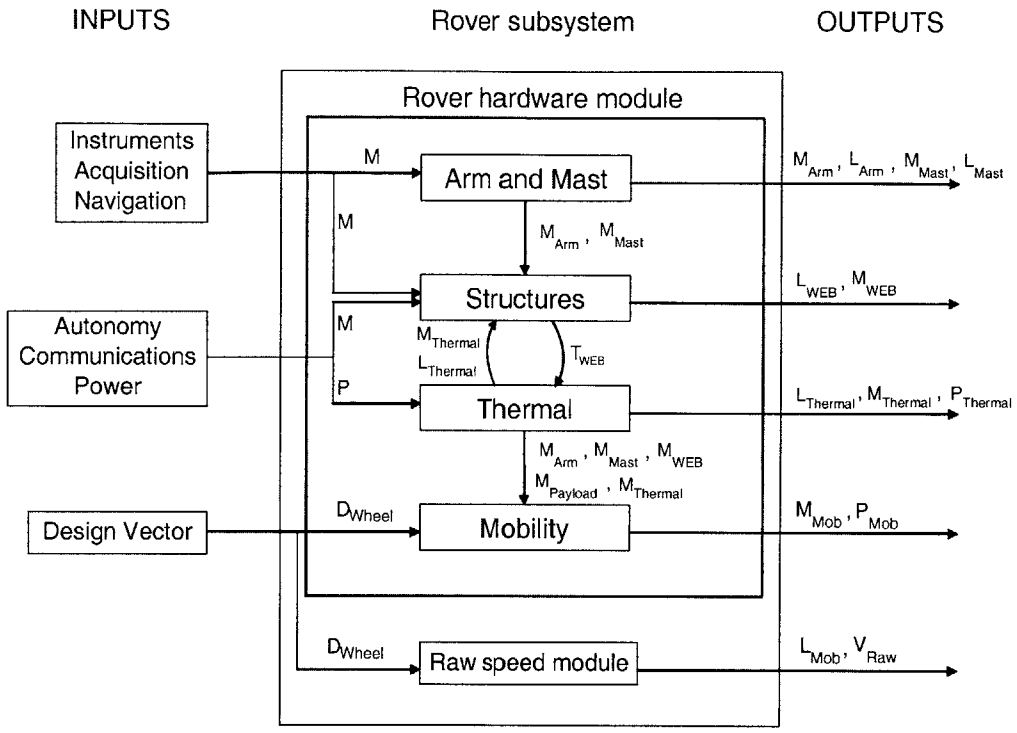


Figure 4-10: *Rover* module program flow

by the payload volume that it must accommodate. Its wall' thicknesses are sized to withstand bending and buckling under the most critical loads of launch and reentry.

Mobility

The mobility module sizes the rocker-bogie suspension system, including wheels, motors and linkages. The total mass of the rover directly affects the design of the mobility system through structural considerations (bending loads) and through actuator sizing. Overall dimensions of the suspension, as well as the rover's raw speed are, however, directly dependent on the wheel size.

Thermal

The thermal module determines the mass and power required to provide heating and cooling to electronics and other hardware inside the WEB. Each piece of payload has individual thermal requirements, generally characterized by maximum temperature,

minimum operating temperature, and minimum survival temperature. In order to size the components of the thermal subsystem, the temperature limits are considered with respect to ambient average and ambient extreme temperatures. The heating and cooling components of the rover thermal system must be able to maintain each instrument within its allowable temperature range, at all times. Primary outputs of the module include mass, size, and power requirements of thermal control devices, including heaters, heat pipes, and radioisotope heater units.

The fifth module shown in Figure 4-10, called *Raw Speed*, is actually the algorithm of *Mobility* which calculates the raw speed. It is independent from the main body of *Mobility* so that the rover's raw speed is determined very early in the Rover Modeling calculations (Figure 4-3). The modeling work on each of the *Rover*'s submodules is presented in the subsequent sections.

4.7.2 Rover structure design

Mast and arm

The models of the mast and arm are currently rudimentary, they provide other modules with estimates of the hardware mass and dimensions. The dimensions of the mast and arm are assumed to be simply proportional to the wheel diameter; the next two paragraphs present the methods used to compute their masses.

The mast is designed purely structurally to meet a maximum deflection requirement. It is modeled as a simple beam of square cross-section undergoing bending due to the weight of a tip mass representing the navigation and instrument equipment's mass. Buckling and vibration considerations are not yet taken into account. Assuming a maximum allowable deflection, $\delta = 0.5mm$, the mast's area moment of inertia, I_{mast}, is first calculated from the Bernoulli-Euler relation

$$I_{mast} = f_s \frac{N\ L^3}{E\ \delta} \tag{4.10}$$

On the right hand side, the load, N, is equal to the carried equipment mass times the

gravitational acceleration on Mars. The length, L, of the mast is scaled with respect to the overall size of the rover. The Young modulus, E, is 68 giga Pascal for a beam made of aluminum 6061-T6 (Table 11-53 of [56]). The factor of safety, f_s, is set to 5.

The moment of inertia is also expressed directly from the geometry of the mast in Equation 4.11, where t is the beam's thickness and b its width.

$$I_{mast} = \frac{1}{12}\, b^4 - \frac{1}{12}(b - 2t)^4 \tag{4.11}$$

Assuming a thickness to width ratio, $t/b = 0.05$, Equations 4.10 and 4.11 are solved for both variables. The mass of the mast is then easily derived from the mast's dimensions and material density.

The arm is a segmented appendage that accommodates, and gives mobility to, the contact-sensing science payload. It is a complex system, in terms of moving parts and degrees of freedom, that must stand many loading configurations. To model all these features would be painstaking, as only the mass of the arm is a required output for the rest of the model. To keep calculations efficient the mass of the arm is simply scaled with respect to the moment of its payload weight, which is one of the loads acting on it. The mass of the arm, M_{arm}, is assumed to be simply proportional to the arm's tip mass, M_{tip}, times its length, L_{arm}.

$$M_{arm} = \frac{L_{arm}}{C}\, M_{tip} \tag{4.12}$$

In this equation, C is a constant set to 1 meter according to the FIDO[2] and MER rovers arm characteristics.

Warm electronics box

The design process of the warm electronics box (WEB) has two steps. First, its geometry is drawn so that it can accommodate all necessary elements. Then, the thicknesses of its walls are structurally designed to support them.

[2]Fido (Field Integrated and Design Rover) is a conceptual vehicle used for technology definition and field tests by the Jet Propulsion Laboratory.

WEB geometry Similar to Sojourner's design, the shape of the WEB is a simple parallelepiped. The geometry of the WEB must handle two opposed constraints. On the one hand, it must be large enough to house all required equipment. On the other hand, it must be small enough to fit in the footprint of the rover delimited by its six wheels (Figure 4-11). The dimensions of this footprint are fixed because they are

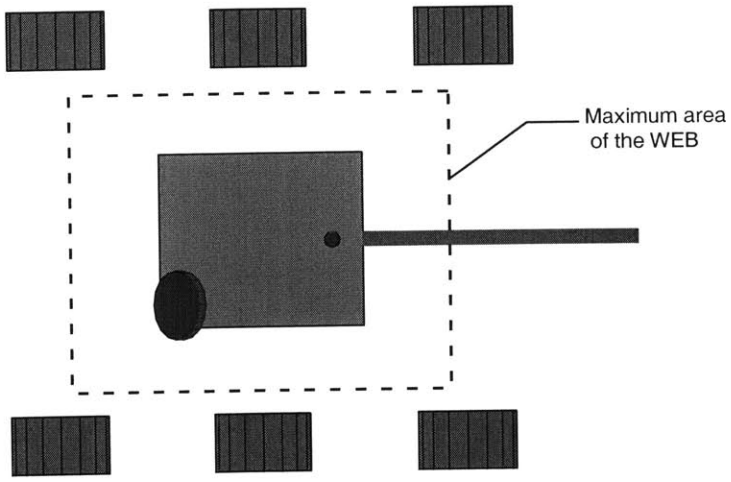


Figure 4-11: View from top of the footprint of the rover and the WEB

directly dictated by the wheel size, which is a design variable (Section 4.7.3, Equation 4.21, page 108). The WEB's geometry is, therefore, subjected to the footprint dimensions. The WEB is contrived in two steps in the following manner: its dimensions are first calculated to satisfy the housing requirement and then checked against the rover's footprint dimensions. If the WEB is too large, the design is rejected.

WEB structural design The WEB is made of two horizontal plates, top and bottom, and four vertical walls (Figure 4-12). The design procedures for the plates and walls are explained separately, as they undergo different loading conditions.

Top and bottom plates support the mast and the equipment inside the WEB, respectively. They are loaded normally and, therefore, their thicknesses are designed for bending. Both plates have a sandwich structure composed of a lightweight core between two thin metallic skins (Figure 4-13). The core's thickness t_{core} is determined

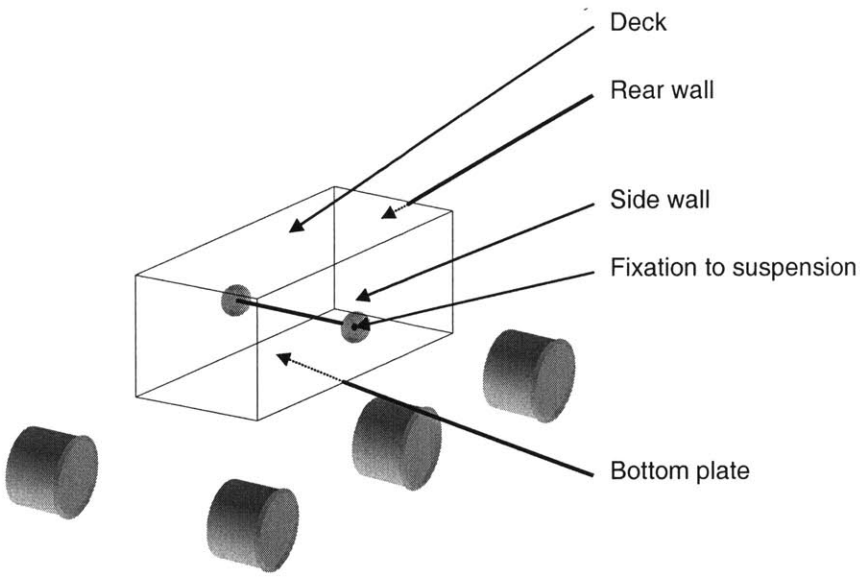


Figure 4-12: Nomenclature for the walls constituting the WEB

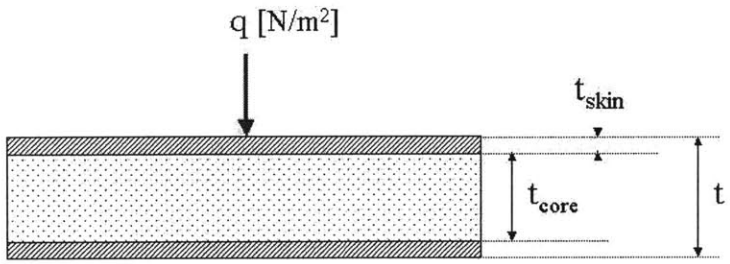


Figure 4-13: Plate's sandwich structure

by the following weight-minimization expression ([3]):

$$t_{core} = \sqrt{\frac{w_{core}\ \beta\ q\ a^2}{2\ w_{skin}\ F_{skin}}} \tag{4.13}$$

The variables w_{core} and w_{skin} refer to the densities of the core and skin, respectively. The variable q is the unit load per area and a is the plate's width. The allowable facing stress, F_{skin}, is equal to 420 mega Pascal for the aluminum 2219-T851 constituting the skins (Table 11-52 of [56]). The non-dimensional variable β depends on the plate aspect ratio and is conservatively set to 0.12 [3].

The skins' thicknesses are driven by a given maximum deflection requirement

($\delta \leq \delta_{max}$). The maximum deflection, δ_{max}, dictates the necessary flexural rigidity, D (Equation 4.14), which in turn is a function of the plate's thickness, t (Equation 4.15).

$$\delta_{max} = \frac{16\ q\ a^4}{\pi^6\ D}\ C_1 \tag{4.14}$$

$$D = \frac{E_{skin}}{12\ (1-\nu_{skin}^2)}\ (t^3\ -\ t_{core}^3(1\ -\ \frac{E_{core}}{E_{skin}})) \tag{4.15}$$

$$t_{skin} = \frac{t\ -\ t_{core}}{2} \tag{4.16}$$

In Equation 4.14, C_1 is another constant used in [3] and conservatively set to 2. Finally, the skin thickness is simply half the difference between the total and core thicknesses (Equation 4.16).

Regarding the walls, their design must satisfy two functions. First, they are the support for instruments, navigation elements, and arm. Second, they connect the WEB to the suspension (Figures 4-12 and 4-14). This connection is assumed to be located at the center of the WEB's side walls. It is at this location that the weight of the WEB and its attached equipment is transferred to the ground via the suspension system. As a first approximation, the upper-halves of the side walls work under compression, whereas the lower-halves work under tension. The upper-half is compressed between the load of the deck plus the attached hardware and the ground reaction. Therefore, it is designed for buckling. The lower half is stretched between the ground reaction and the weight of the equipment attached to the bottom plate. Hence, it is designed to resist tension. The other two walls, front and rear, are designed with the same thickness as the side walls.

Similar to Sojourner's design, the walls are made of three components: a structural z-spar element to carry the loads, layers of aerogel for insulation, and two fiberglass sheets for containment [52] (Figures 4-14).

The thickness of the z-spar element is calculated to satisfy both tensile and buckling requirements, mentioned in the previous paragraph. The design of the walls for buckling is similar in process to the design of the plates for bending. The buckling sets a condition on the value of the flexural rigidity, D_{spar}, which is expressed as

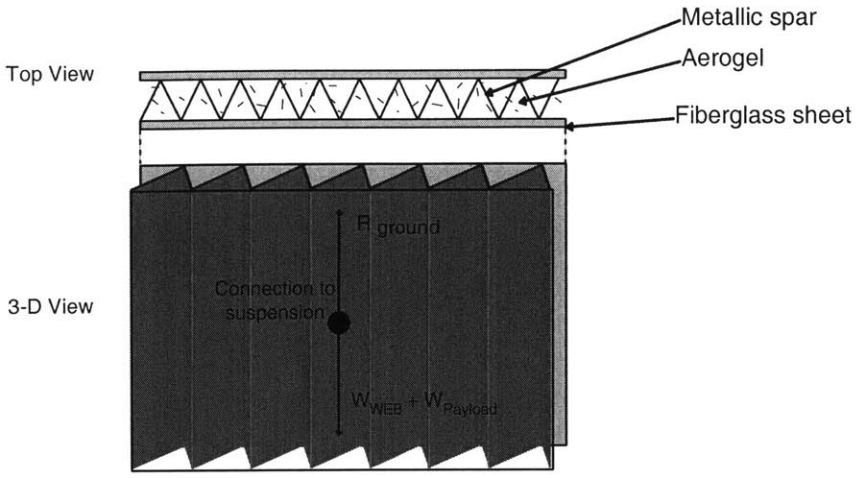


Figure 4-14: Structure of a wall

a function of thickness, $t_{buckling}$. The buckling condition is that the axial load on the wall , N_{spar}, must be less than a critical load, N_{crit}, as defined by Euler in the following equation

$$\begin{aligned} N_{crit} &= \frac{4\,\pi^2}{3}\,D_{spar}\,a^2\left(\frac{3}{a^4}+\frac{3}{b^4}+\frac{2}{a^2\,b^2}\right) \\ N_{crit} &\geq f_s N_{spar} \end{aligned} \tag{4.17}$$

where a is the plate dimension in the loading direction and b in the orthogonal direction. The relation between the flexural rigidity, D_{spar}, and the thickness, $t_{buckling}$, involves structural properties of the metal that constitutes the spar. For aluminum 2219-T851, the Young modulus E is 72 giga Pascal and the Poisson coefficient, ν is set to 0.3 [56].

$$D_{spar} = \frac{E\,t_{buckling}}{12\,(1-\nu^2)} \tag{4.18}$$

The design for tension is more straightforward. It simply stipulates that the spar's thickness, $t_{tension}$, must be large enough so that T_{spar}, the tensile strength acting on it, is less than T_{ult}, its ultimate tensile strength. Keeping the same notations, the designing equation is

$$t_{tension} = \frac{f_s\,T_{spar}}{b\,T_{ult}} \tag{4.19}$$

The thickness of the spar, t_{spar}, is then set to satisfy both requirements.

$$t_{spar} = \max(t_{buckling}, t_{tension}) \tag{4.20}$$

Finally, the total mass of the WEB is easily derived from the geometry of all its components.

4.7.3 Mobility

The purpose of the *Mobility* module is to calculate the dimensions, mass and power requirements for the rocker-bogie suspension system. This system includes wheels, motors and linkages. It is worth noting that, as discussed in subsequent paragraphs, the mobility system features are primarily driven by the size of the wheel.

Mobility geometry

The geometry of the mobility system is driven by the choice of its suspension type. The decision has been made to model only the rocker-bogie suspension system, patented by the Jet Propulsion Laboratory. This six-wheel suspension was chosen because it is the one used on Sojourner, MER, and most likely will be used on MSL. Moreover, the suspension used on the ESA's ExoMars rover, developed by the Rover Science and Technology Company, has a design similar enough to be modeled like the rocker-bogie [30].

For a rocker-bogie suspension to be stable, its wheelbase, $L_{wheelbase}$, track, L_{track}, and rover length, L_{rover}, must be direct functions of the wheel diameter, D_{wheel}, as illustrated in Figure 4-15. These relations are given below [58]

$$\begin{aligned} L_{wheelbase} &= C_{base}\, D_{wheel} \\ L_{track} &= C_{track}\, L_{wheelbase} \\ L_{rover} &= L_{wheelbase} + D_{wheel} \end{aligned} \tag{4.21}$$

where C_{base} and C_{track} are constants set to 4.2 and 1, respectively, based on averages

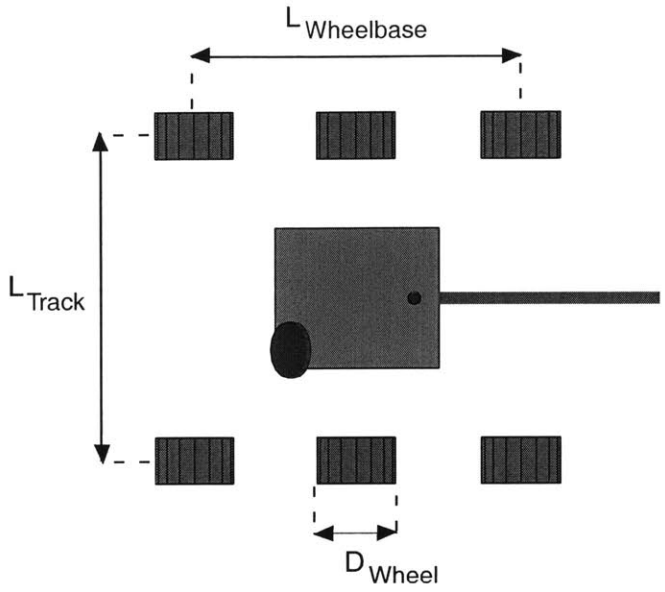


Figure 4-15: Rover geometry

from Sojourner, MER and MSL designs.

Mobility structural design

The suspension system is modeled as an association of simple beams. As for the design of the mast explained above, each beam is then designed to bend within a maximum deflection range. The mass of the differential situated inside the WEB is crudely set to 4.4 kilograms, the value corresponding to the ExoMars differential [30].

The mass of each wheel is scaled with respect to its size. The scaling relationship is created by a curve fit to both a database of sport-car wheel masses [2] and the wheel of the ExoMars rover (Figure 4-16). The mass of the wheel scales simply to the cube of its size.

Mobility performance

The notion of rover performance is composed of two aspects, its ability to drive over rocks and its raw speed, namely the maximum speed at which it can drive at on flat terrain. For a rocker-bogie suspension, rock management ability is directly related

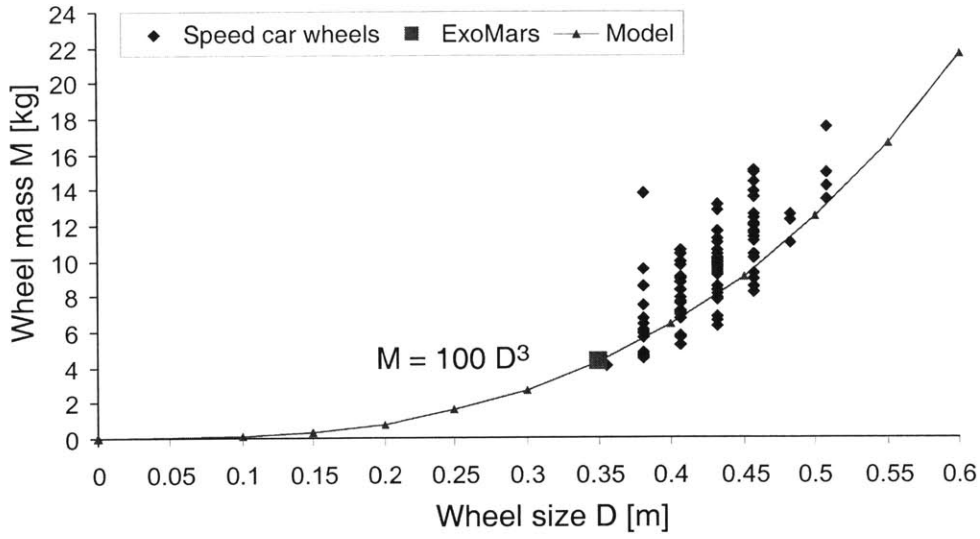


Figure 4-16: Wheel mass as a function of the its diameter

to the wheel size. The maximum traversable obstacle height is one to one and a half times the wheel diameter. The rover's maximum raw speed, however, is a function of many parameters. It depends on the power available, the rover's mass and its structural design. Still, the decision was made to model the raw speed as a simple function of the wheel diameter for two main reasons. First, the wheel diameter is unquestionably the key variable of the whole mobility system. As the geometry of the rover scales with the wheel size, it makes sense that speed behaves in a similar manner. Second, this assumption allows the velocity to be determined independently from the rover's other properties. And as the N^2 diagram demonstrates (Figure 4-3), the disconnection between raw velocity, calculated in the *Raw Velocity* module, and hardware design, calculated in the *Rover Hardware* module, greatly simplifies the program flow.

The function relating raw speed to wheel size was generated based on a curve fit to data points from Sojourner, MER and the current available values for MSL (Figure 4-17). The curve fit implies that for large wheel sizes, the velocity is an affine function[3] of the wheel diameter that increases with a small slope. In this range of

[3]Affine functions are of the type $f : x \mapsto a\ x + b$. Linear functions are a particular case of affine functions for which $b = 0$.

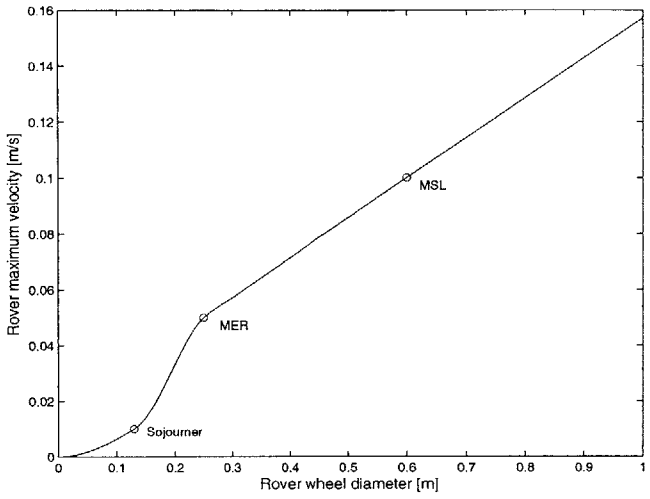


Figure 4-17: Rover raw velocity model

wheel diameters, the rover's structural integrity and navigation ability are the limiting factors to speed increase. The behavior of the velocity law reproduces conservatively the impact of these limitations. The validity of the curve fit is discussed in more detail in the validation section (Section 4.7.5).

Based on Charles Whetsel's guidelines [58], the driving power scales with the maximum velocity and the rover total mass. Each of the six wheels is equipped with a motor sized to provide a thrust equal to half the weight of the rover on Mars. The driving power, P_{drive}, is then a function of the number of wheels, N_{wheels}, rover mass, M_{rover}, and raw speed, V_{Max}:

$$P_{drive} = \frac{M_{rover}\ g_{Mars}}{2}\ V_{Max} \tag{4.22}$$

4.7.4 Thermal model

As stated in the Responsibilities paragraph, the *Thermal* module designs heating and cooling components to maintain the temperature inside the WEB within the allowable temperatures of its payload. The thermal requirements of each element of the rover payload, mainly batteries and electronic components, are generally characterized by the following three quantities:

1. The *maximum temperature* is the highest temperature that the element may reach without sustaining damage

2. The *minimum operating temperature* is the lowest temperature that the component can reach while in a powered-on, operational state, without sustaining damage

3. The *survival temperature* is the absolute minimum temperature at which the component can be maintained without sustaining damage.

The scope of the thermal module does not include temperature regulation for instruments or other equipment located outside of the WEB; thermal controls of components outside the WEB are handled by their respective subsystems. Heat generated from the dissipation of unused solar or RTG power is currently not included in the model. The thermal model is shown in Figure 4-18, where T1, T2, T3, and T4 are the payload temperature, interior wall temperature, exterior wall temperature, and ambient temperature, respectively. The heat exchange between the payload and the environment is represented by Q. Thermal transfer occurs by three primary methods:

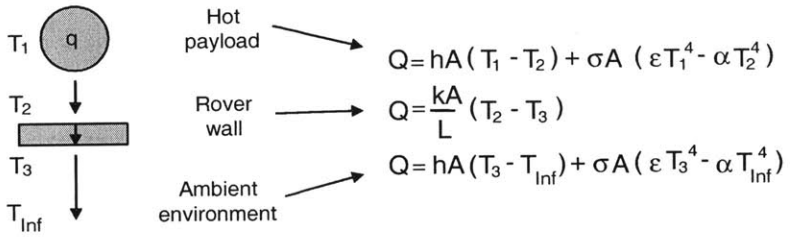


Figure 4-18: Scheme of the *Thermal* model

conduction, convection, and radiation. All three of these heat transfer methods are included in the WEB thermal model.

Heat transfer through a flat object is proportional to the difference between the surface temperatures on each side of the object, and can be modeled by Equation 4.23. In this calculation Q is the heat transfer rate, L is the thickness of the material, A is the area of the material, and the temperature-independent conduction coefficient, k, depends on the properties of the material. The convection coefficient used in

the module is $k = 0.5W/(m^2K)$. This is an estimate based on the low end of the convection coefficient scale reported by Incropera and DeWitt [36].

$$Q = \frac{k\ A}{L}\ (T - T_0) \tag{4.23}$$

Heat transfer by convection is proportional to the difference between the surface temperature of the object and the ambient temperature. Convective heat transfer can be modeled by the following equation, where A is the area of the material, and the convection coefficient, h, depends on environmental factors such as wind speed and the density and chemical composition of the convecting medium.

$$Q = h\ A(T - T_0) \tag{4.24}$$

Heat transfer by radiation depends on the difference between the fourth powers of the surface temperature of the object and the ambient temperature. The material may have different emission and absorption properties, which are characterized by the emissivity coefficient, ϵ, and the absorptivity coefficient, α. Radiative heat transfer can be modeled by the following equation, with σ as the Stephan-Boltzmann constant

$$Q = \sigma\ A(\epsilon\ T^4 - \alpha\ T_0^4) \tag{4.25}$$

Three heating and cooling systems are considered. These include two passive systems, Radioisotope Heater Units (RHU) and heat pipes, and one active heater system. To size these elements, the *Thermal* module algorithm considers four extreme temperature pair cases

1. Maximum daytime ambient temperature and maximum payload temperature limit. This case decides whether to use RHU's or heat pipes.

2. Average daytime ambient temperature and minimum operating temperature limit. This case reckons daytime average heater power.

3. Average nighttime ambient temperature and minimum survival temperature

limit. This case reckons nighttime average heater power.

4. Minimum nighttime ambient temperature and minimum survival temperature limit. This case reckons nighttime maximum heater power.

For each pair of environment and payload temperatures, there is one steady state heat exchange value, q_{steady}. In each of the four cases the thermal power requirements are sized so that the heat coming from the payload and the heaters equal, q_{steady}, using Equations 4.23, 4.24, and 4.25.

4.7.5 Validation

The critical assumptions that need to be validated concern the speed and power models of the mobility system. A validation of the WEB and suspension masses is presented in the system validation section (Table 4.9, page 140).

Raw speed model

The maximum speed of the rover is defined as a function of the wheel diameter by a curve fit to existing data points, extrapolated by an asymptotic behavior (Figure 4-17). Because very few data points exist, the curve fit is somewhat arbitrary; however, no better modeling approach has been contrived. Given this uncertainty, it is of interest to asses how robust the overall rover model is to uncertainties specific to maximum speed calculations.

The raw speed is used by the *Autonomy Traverse* module to calculate the average velocity of the rover during a traverse (Figure 4-3). The average speed is defined over a driving cycle which breaks down into two phases (Figure 4-19). For each cycle the rover first stands still and examines the terrain in the area just in front of it. Then, if the navigation algorithms decide the terrain is safe, the rover drives forward half a rover length with a speed equal to the raw velocity. The first phase requires a computing time, T_{Comp}, which depends on the area to examine and the maximum traversable rock size. On the one hand, when a rover gets larger it needs to examine a more terrain before moving to next step. On the other hand, it has better ground

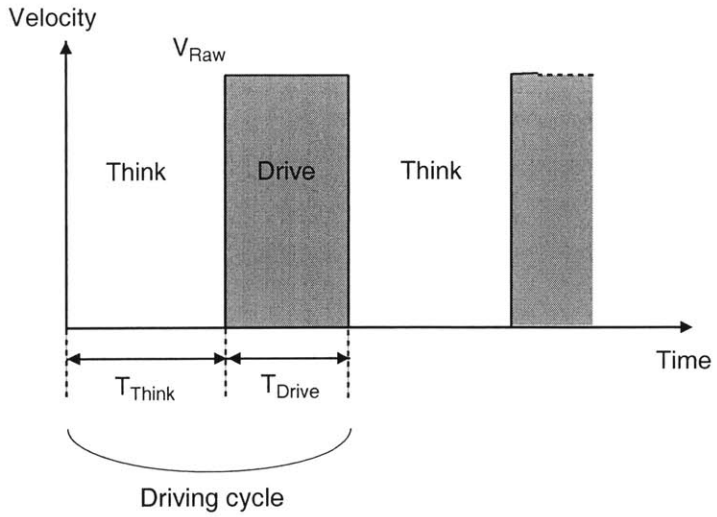


Figure 4-19: Driving cycle

clearance and fewer rocks to be concerned about. For these reasons, as a first order approximation, T_{Comp} is assumed to be independent of the rover size, and thus of the wheel size, D_{wheel}. With a computer similar to that of MER (RAD6000), the computing time is estimated to be 45 seconds; it decreases if more computing power, $N_{Computers}$, is used:

$$T_{Comp} = \frac{45}{N_{Computers}} \tag{4.26}$$

The time required for the second phase, T_{Drive}, is simply given by:

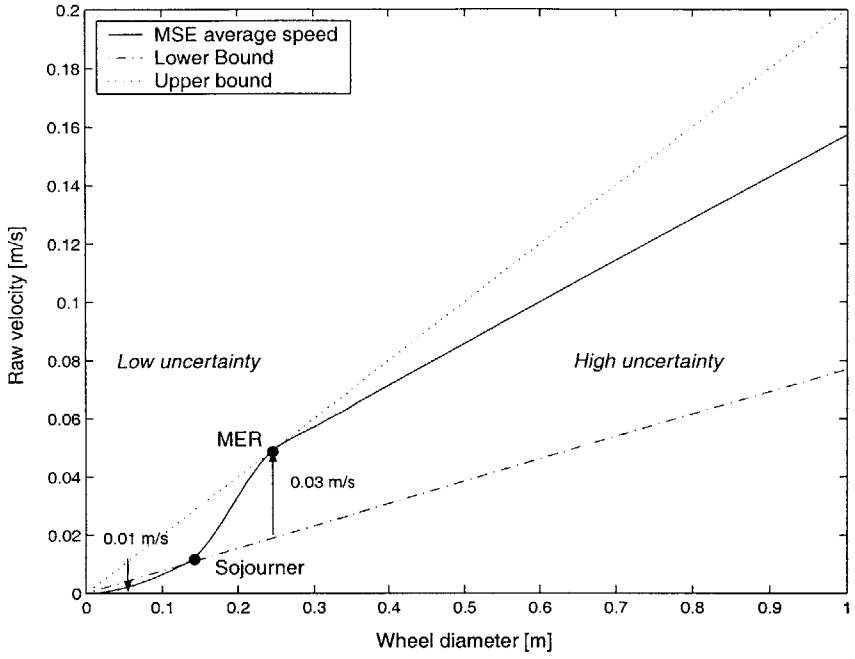
$$T_{Drive} = \frac{1}{2}\frac{L_{Rover}}{V_{Raw}} \tag{4.27}$$

Thus, the expression for the average velocity is:

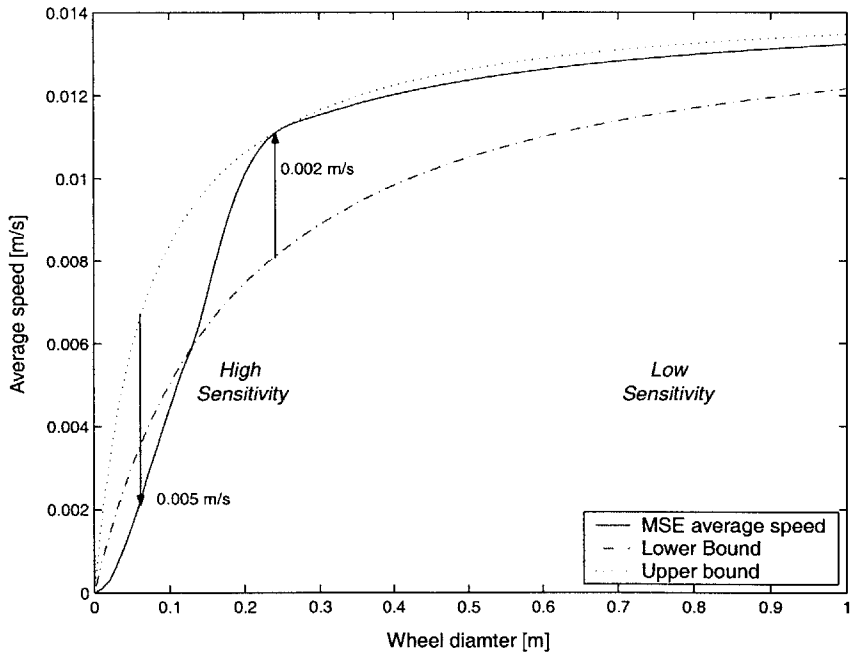
$$V_{Avg} = \frac{V_{Raw}\ L_{Rover}}{L_{Rover} + 2\ V_{Raw}\ T_{Comp}} \tag{4.28}$$

$$V_{Avg}^{Lim} = \frac{L_{Rover}}{2\ T_{Comp}} \tag{4.29}$$

This paragraph examines how sensitive V_{Avg} is to the uncertainties in V_{Raw}. Figure 4-20(a) relates the raw speed on the y-axis to the wheel diameter on the x-axis for three raw speed laws. Represented by a solid line is MSE's raw speed model, roughly



(a) Raw velocities



(b) Average velocities

Figure 4-20: Sensitivity of the average speed to the raw speed model

bound by two arbitrary linear raw speed laws. The upper-bound uses the MER design point as a reference, the lower-bound uses Sojourner; both can be seen as examples of uncertainty boundaries for MSE's model. Using Equation 4.28, these three raw speed laws result into three average speed laws plotted on Figure 4-20(b). The largest difference between MSE's model and the MER arbitrary law is for $D_{Wheel} = 0.06$ meters, and the largest difference between MSE's model and the Sojourner arbitrary law happens for $D_{Wheel} = 0.25$ meters. The average velocity is most sensitive to uncertainties in the raw speed model in the region of wheel diameters smaller than 0.25 meters. However, this is the region where MSE's raw speed model is most reliable because it uses the origin, Sojourner and MER as references. The average velocity is less sensitive in the region of large wheel sizes, where the asymptotic behavior of MSE's raw speed is quite uncertain.

In conclusion, the average velocity is robust to raw speed modeling uncertainties. Where uncertainties in raw speed are high, the sensitivity of the average velocity is low and vice versa.

Driving power model

The Equation 4.22 is validated against driving power values for the four rovers shown in Table 4.5. The actual and modeled values are consistent except for the case of

Table 4.5: Validation table of the power model

Rover	Actual P_{Drive} in [W]	Modeled P_{Drive} in [W]	Sources
Marsokhod 75	70	76	[24]
MER	17	17	[21]
ExoMars	12.5	14	[30]
Sojourner	**1.7**	**0.2**	[44]

Sojourner for which there is a strong discrepancy. MSE's power equation can therefore be used with confidence for rovers whose wheel sizes are larger than 0.20 meters.

4.8 Power

4.8.1 Responsibilities

The *Power* module sizes the rover's power generator and batteries. These two hardware pieces are tailored to satisfy, if possible, the power usages of the other subsystems. In particular, power is necessary to operate instruments, acquisition tools, and computers, and also to communicate, drive and keep the rover at nominal temperatures. Because the power plant and batteries are sized based on other subsystems' requirements, the *Power* module appears downstream of the N^2 diagram, at the center of a loop with the *Avionics* module and one with the *Rover* module (Figure 4-3, page 84).

One critical design trade deals with the use of Radioisotope Thermoelectric Generators (RTGs), instead of traditional solar panels, as a solution to provide power to future rovers which will operate for over 500 sols. On the one hand, solar panels provide a lot of energy at the beginning of a mission, but on Mars they suffer from dust deposition. This phenomenon diminishes the panels' available power by 0.28% every sol and is a limiting factor for the rover mission's lifetime [40]. On the other hand, one RTG provides initially less power but at a constant rate day and night, for an almost unlimited lifetime (more than 20 years). Missions using RTGs must, however, go through a special qualifications process which adds expenses to the already high cost of RTGs. The power plant system is a design variable, therefore, this trade can be analyzed in detail with the MSE tool (Section 5.4).

4.8.2 Main Assumptions

The users are given the choice among many technologies to customize the power system. The technology defaults for batteries, panels and RTGs are the following:

- Regarding the batteries, the users can choose traditional solutions like nickel-cadmium and nickel-hydrogen varieties or pick the lithium-ion technology. The latter is preferred as it is used on MER [21]. Whatever their variety, batteries

are allowed a depth of discharge of 25%.

- Three main types of solar panels are considered: the crystalline silicon, multi-junction and gallium arsenide. The last kind is set as the default because a similar technology (GaInP/GaAs/Ge) is used on MER [21]. All cell characteristics are taken from [56].

- If MSL is to be powered by a nuclear plant, it will use either the Multi-Mission Radioisotope Thermoelectric Generator (MMRTG), or the Stirling Radioisotope Generator (SRG). The MMRTG is chosen as the default RTG because it has a better technology readiness level, even if a lower efficiency, than the SRG.

4.8.3 Design Flow

This section describes the design flow for both solar and nuclear kinds of power plants. In both cases the power plant is sized for a sol during which the rover is traversing. Not enough information about the science payload's power profile is available to size the power system for a sol of scientific experiments.

Solar-powered plant design flow

The design of the solar panels must satisfy two conflicting constraints. On the one hand, it must capture enough solar energy during a sol to fulfill the rover's power thirst. To do so, the solar panels must be large in order to capture as much solar flux as possible. On the other hand, the dimensions of the solar panels must be scaled with those of the WEB on which they are attached. Following the program flow shown in Figure 4-21, the power available and the power requirements are first calculated independently and then compared to each other.

Power available The dimensions of the WEB set an upper-bound on the area of the panels. The maximum ratio of the panel area, $A_{Panel_{max}}$, to the WEB area, A_{WEB}, is set to a default value of 6 (Equation 4.30). This default value accounts for

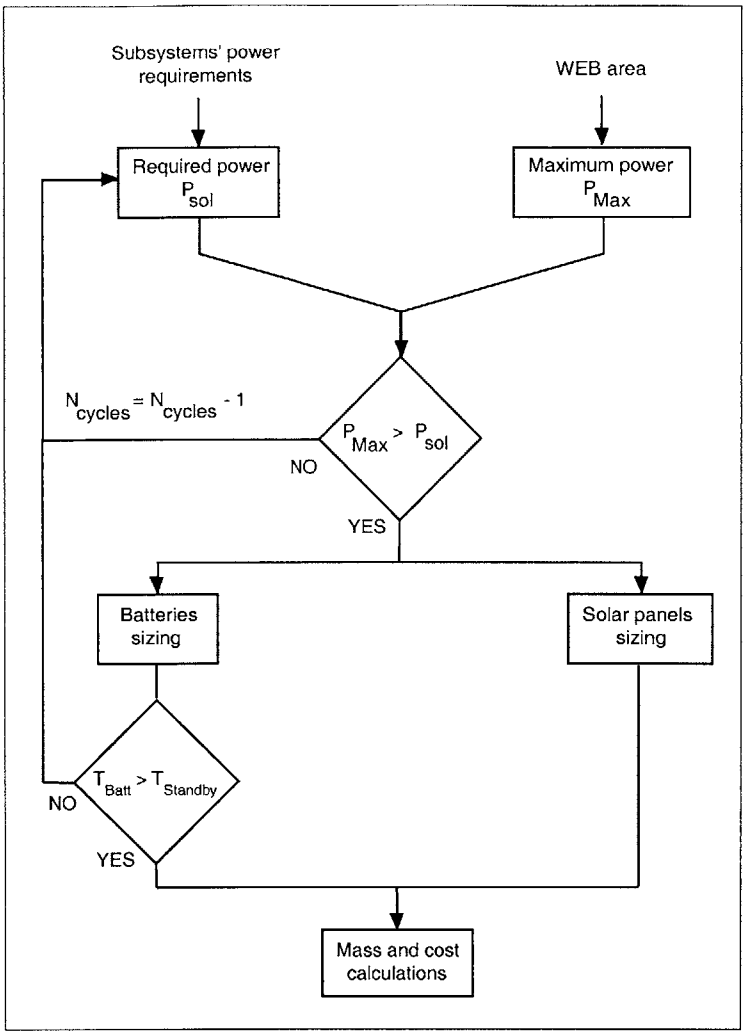


Figure 4-21: *Power* module program flow

a growth of the panel area compared to the MER design, whose panel to WEB ratio is 4.

$$A_{Panel_{max}} = 6 \times A_{WEB} \tag{4.30}$$

The maximum panel area determines P_{max}, the maximum power that the rover can generate at its end of life (EOL). P_{max} is simply the maximum panel area multiplied by the solar flux that reaches the panels, Φ_{panel} (Equation 4.31). The solar flux on the arrays is less than the nominal solar flux on the Mars surface, $\Phi_{surface}$, for several reasons. First, the rays of the Sun are inclined with respect to the panels'

normal vector by an angle, called the Sun incidence angle, θ. Second, the arrays have inherent manufacturing degradations and life degradations due to solar cells' deficiencies. Third, the Martian dust covers the panels over time. These last two kind of losses are referred to as $L_{Degradations}$ in Equation 4.32.

$$P_{max} = A_{Panel_{max}} \times \Phi_{panel} \tag{4.31}$$

$$\Phi_{panel} = \cos\theta \times L_{Degradations} \times \Phi_{surface} \tag{4.32}$$

Power requirements Operations during a sol are divided into five categories: communication, driving, thinking, stand-by, and night operations. The *Power* module is provided with a power and time of usage profile, $P_{operation}$ and $T_{operation}$, for each of the five operations mentioned. Power draws from instruments, as well as acquisition tools and thermal devices, are averaged for a sol and added to $P_{operation}$. Therefore, all subsystems' requirements are accounted for, and the overall power demand for a sol is ([56]):

$$P_{sol} = \frac{\frac{P_{night} \times T_{night}}{x_e} + \frac{P_{day} \times T_{day}}{x_d}}{T_{day}} \tag{4.33}$$

$$P_{day} = \frac{P_{stand} \times T_{stand} + P_{drive} \times T_{drive} + P_{think} \times T_{think} + P_{com} \times T_{com}}{T_{day}}$$

During stand-by it is assumed that only avionics and thermal components are working. Drive, think and communication power draws are provided by *Rover*, *Autonomy* and *Communications* modules, respectively. The drive and think energies for a sol depend on N_{cycles}, the number of driving cycles performed during a sol as defined in Section 4.7.5. This number is initially computed by the *Autonomy* Traverse module and it determines T_{drive} and T_{think} of the Equation 4.33. The drive time during a sol is N_{cycles} times the driving time for one moving step, $T_{drive_{step}}$; the same is true for the thinking time.

$$T_{Drive} = N_{cycles} \times T_{Drive_{step}} \tag{4.34}$$

$$T_{Think} = N_{cycles} \times T_{Think_{step}}$$

The number of cycles is the only variable at the disposal of the *Power* module to reduce the overall daily power demand.

Comparison of the required and maximum powers As shown in Figure 4-21, the next step is to check the daily power requirement against the power available. If the latter is larger, then the solar arrays are sized to provide the daily power demand, P_{sol}. Otherwise, the power demand, as it is, is not satisfied by the available range of power. In this case, the power requirements must be lowered until they are less than the maximum power available. Now, the only way for the *Power* module to lower the demand, without modifying the rover payload or hardware, is to play with the N_{cycles} parameter mentioned in the previous paragraph. Decreasing the number of Think-Drive cycles reduces the time, and consequently energy, required for the Think and Move operations (Equations 4.34 and 4.33). The *Power* module decreases N_{cycles} until the power condition is met (Figure 4-21). If by the process N_{cycles} reaches 0, meaning the rover does not move at all, the design is rejected.

Sizing of the panels and batteries From a relationship similar to Equation 4.31, solar panel area and power demand are related as follows

$$A_{panel} = \frac{P_{sol}}{\Phi_{panel}} \tag{4.35}$$

The design of the batteries is less straightforward; it depends on how the overall power available, P_{sol}, compares to individual operation's requirements, $P_{operation}$. For example, regarding the driving cycle, it is assumed that if at least one of the driving or thinking powers is larger than P_{sol}, the rover recharges its batteries between each cycle (Figure 4-22). The amount of energy stored inside the batteries and the recharge times are calculated by examination of such profiles for driving cycles, communications, and night operations. If there is not enough stand-by time, $T_{Standby}$, during a sol to recharge the batteries the number of driving-cycles must again be decreased (Figure 4-21).

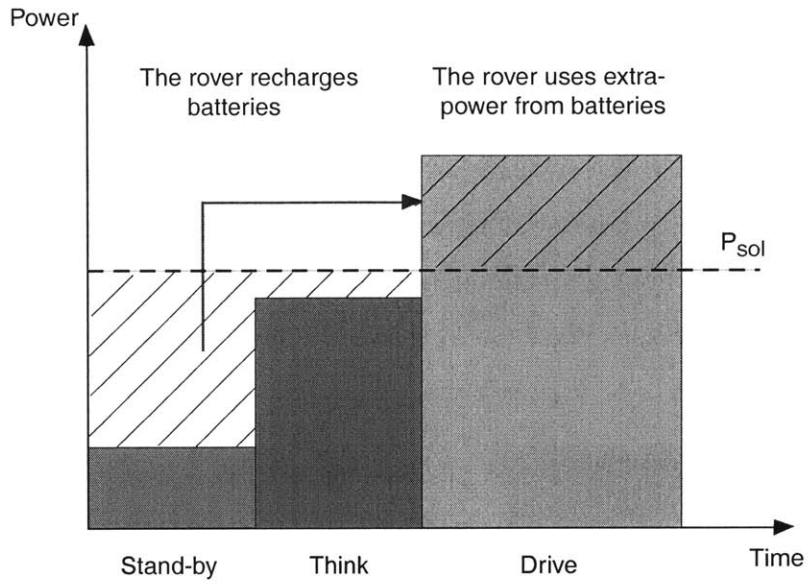


Figure 4-22: Example of a battery recharge cycle

Nuclear-powered plant design flow

The total power requirement for a sol, P_{sol}, is calculated in the same fashion as in the solar case (Equation 4.33), with the exception that an RTG produces power day and night:

$$P_{sol} = \frac{\frac{P_{night} \times T_{night}}{x_e} + \frac{P_{day} \times T_{day}}{x_d}}{T_{day} + T_{night}} \tag{4.36}$$

The *Rover Hardware* module does not explicitly impose an upper-bound on the number of RTGs that a rover can carry. To fulfill the power requirement, P_{sol}, the number of necessary RTGs, N_{RTG}, is given by:

$$N_{RTG} = E\left[\frac{P_{sol}}{P_{RTG}}\right] + 1 \tag{4.37}$$

where P_{RTG} is the power provided by each RTG and $E[\,]$ is the floor function. Then the sizing of the batteries is done in the same manner as in the solar case. The number of RTGs could be sized to meet the total peak power requirement, in which case there would be no need for batteries. However, due to the very high cost of RTGs this solution is no retained.

4.8.4 Validation

As the *Power* module depends so much on the entries from other subsystems, its validation is presented in the systems validation section. In this section, Table 4.9 presents the total power produced along with the battery masses for Sojourner, MER and MSL as modeled by MSE compared to their actual values. Additionally, as a complement to this table, Figure 4-23 shows the distribution of the total energy received during a sol to the various operations. The figure shows that the useful

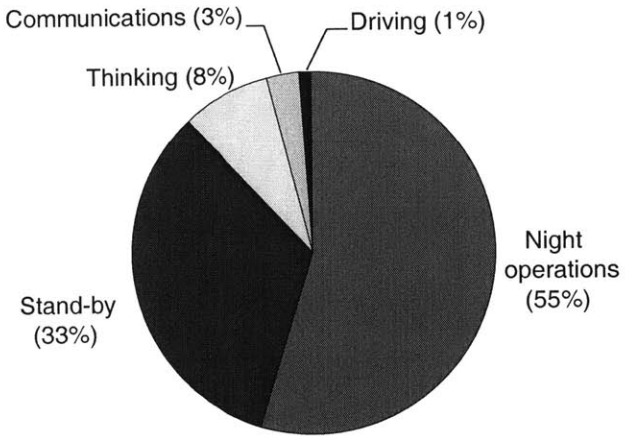


Figure 4-23: Distribution of the energy received during a sol for a MER-like rover

operations, involving traversing and communicating, only represents 12% of the total demand in energy during a sol.

4.9 Communications

4.9.1 Responsibilities

The *Communications* subsystem is for the most part embedded in the *Autonomy* subsystem. The latter calls the *Communications* module to schedule communications activities based on window opportunities. *Communications* provides *Autonomy* with the delay associated with communicating given data volumes and average communication duration per day and night. The *Communications* module is also responsible

for sizing the telecommunication hardware, consisting of the antennas and transponders, in terms of mass and power usage. Finally, it calculates the communication costs, including the Deep Space Network (DSN) usage and equipment costs.

4.9.2 Modeling Assumptions

The communication architecture is a design variable (Section 4.3.2, page 86). The user can choose among four single architecture types and three hybrid architecture types. The single architectures include communications directly to Earth (DTE), with a low altitude orbiter, with a high altitude satellite operating in ultra-high frequency (UHF), or with a high altitude dedicated tele-satellite operating in both UHF and X bands. The hybrid architectures are the combinations of DTE with the other three communication types. For hybrid architectures, DTE is assumed to be the primary communication method, while the other methods are used as backup. Moreover, as the exact launch date is not a tool variable, the DTE architecture is sized for the worst-case scenario, which means for the largest Earth to Mars distance. DTE power usages as well as communication delays are, therefore, modeled conservatively. Additionally, the total delay assumes a human response time to telemetry of two hours and a DSN availability of four hours per sol.

4.9.3 Design Flow

The *Communications* module is divided into three submodules that reflect the three functionalities presented in the introduction to this section (Figure 4-24). The execution sequence of the submodules is detailed in the following paragraphs.

Initially, the autonomy structure accesses the scheduling submodule with a typical data volume to be communicated, and specifies whether there is nighttime operation. The design vector specifies the communication architecture type, while the science vector defines the latitude range of the landing site used to estimate the communication windows' duration. The submodule calculates the delay and communication duration for the given data volume and outputs this information to *Autonomy*. The

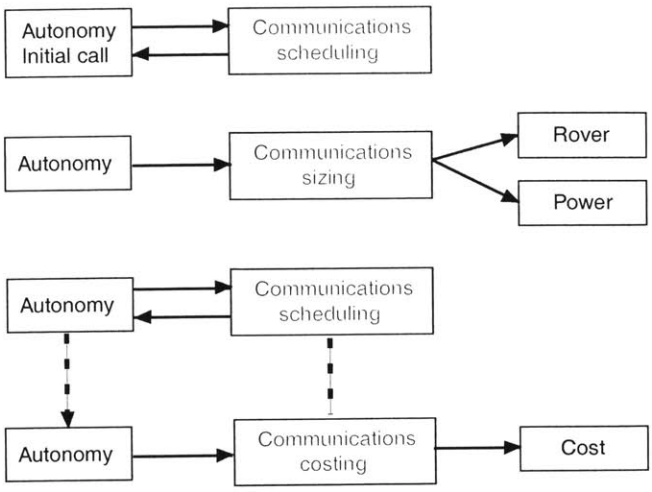


Figure 4-24: *Communications* module program flow

purpose of this initial execution is to initialize the *Communications* and *Autonomy* modules.

Following the initialization, the sizing submodule is executed once, just upstream of the loop between *Avionics*, *Power* and *Rover* in the N^2 diagram (Figure 4-3, page 84). In this execution, the module uses the communication duration and delay information previously determined to calculate the communication duration per day and per night. It also performs link budget calculations to estimate power requirements and size the communication subsystem.

In the third phase, *Autonomy* accesses the scheduling submodule multiple times, outputting a different set of data volumes to be communicated for each of the operations-intensive phases, such as traverse or sample acquisition. The submodule then calculates the total delay associated with communicating the data volumes, as well as associated values including the total duration of data volume communication. Following this sequence of execution, *Autonomy* determines the total number of communication cycles required during the mission lifetime.

In the last phase, the master program executes the costing submodule. Information regarding the total number of communication cycles is now available and can be used for estimating the communication system's cost, including equipment cost and

DSN usage cost.

4.9.4 Validation

Due to lack of rover communication data against which benchmarking analysis could be performed, the validation of the *Communications* model consists of results from the sizing and scheduling submodules.

Link budget

The link budget equations used are those presented in [56]. Table 4.6 shows the link budget inputs and outputs for the three basic communication architectures. Hybrid options are simply a combination of its columns. Regarding the sources used in this

Table 4.6: Link budget results

item	Units	DTE	LMO	HMO (UHF)	HMO (X)
Inputs					
Data rate	[bps]	8000	2.56e5	6.4e4	6.4e4
Frequency	[GHz]	7.145	0.4597	0.4597	7.145
Transmit antenna beamdwidth	[deg]	8.5	180	180	60
Transmit antenna pointing offset	[deg]	0.005	5	5	5
Propagation path length	[km]	4.01e8	1600	20000	20000
Receive antenna diameter	[m]	34	1.3	1.3	1.3
Receive antenna pointing error	[deg]	0.005	0.95	0.95	0.95
System noise temperature	[K]	30	200	200	200
Signal to noise	[dB]	2.7	2.7	2.7	2.7
Transmitter line loss	[dB]	-0.3	-0.3	-0.3	-0.3
Propagation and polarization loss	[dB]	-0.3	-0.3	-0.3	-0.3
Outputs					
Transmit antenna diameter	[m]	0.346	0.254	0.254	0.049
Transmitter Power	[W]	78.3	0.081	3.14	0.94

table, the data rates are those specified by Charles Whetsel in the document [58]. Furthermore, the use of the 34 meters HEF Deep Space Network (DSN) antenna is assumed when communication is via DTE. The Odyssey 1.3 meters diameter antenna is used as reference for LMO and HMO communications. Additionally, the system noise temperature is found to be a critical parameter in the transmitter power

calculation. Charles Whetsel's document suggests a power level of 50 watts. This level of power cannot be achieved unless the system noise temperature is as low as 20 kelvins, whereas receivers typically operate at around 30 kelvins. This indicates that the power must be around 78 watts, which remains within 2 decibels of Charles Whetsel's power requirement value. Finally, the antenna diameters given in Table 4.6 seem consistent with the dimensions of MER communication system.

Communication duration

Figure 4-25 shows the total window duration required for uplink from the rover and downlink to the rover. The plots of the communication durations are for one command cycle, consisting of a varying uplink volume and fixed downlink volume. The figure highlights the relative link capability of the three types of communication architectures. As expected, DTE communication duration is the longest at the lowest data rate of 8kbps, followed by communication via a high orbit relay satellite at 64kbps, and the fastest link is the low orbit relay at 256kbps. Note that the uplink and downlink data rates are equal for each type of communication architecture.

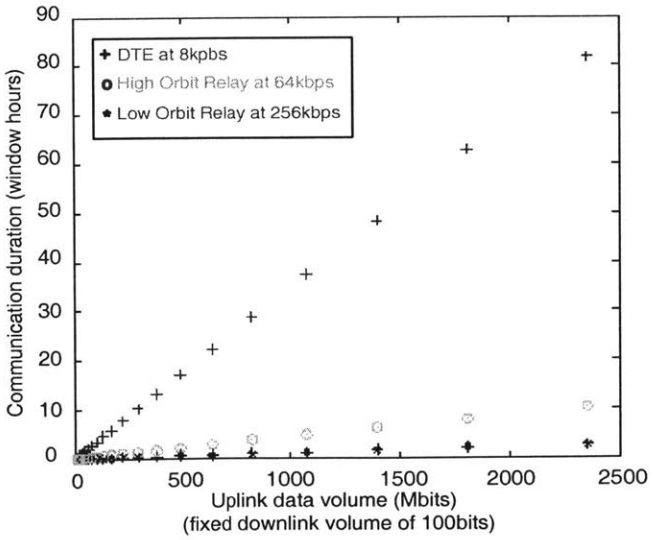


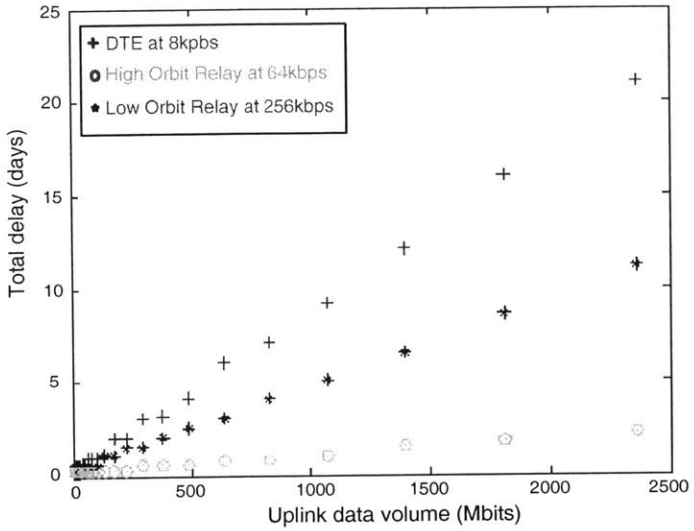
Figure 4-25: Communication durations

Communication delay

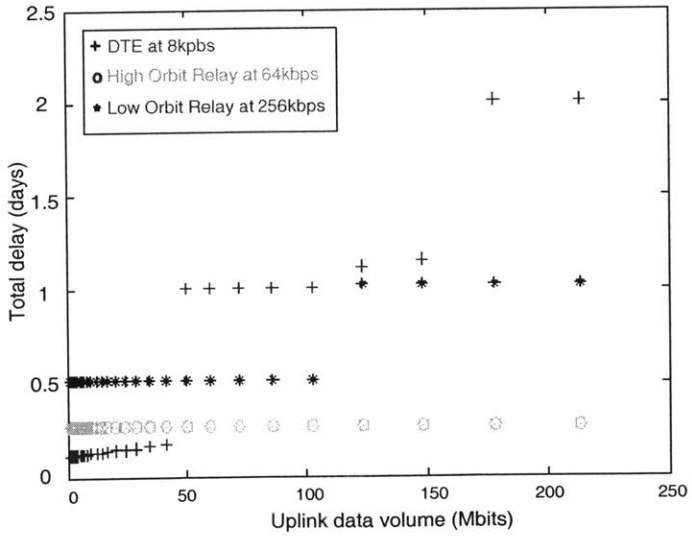
The communication delay is the total delay including communication window availability. It includes uplink duration from Mars, and, if a response is expected from Earth, it also includes round-trip propagation delays, downlink duration, human response time on Earth and command execution time by the rover before the next command cycle starts. Figure 4-26(a) shows the total communication delay for one command cycle with varying uplink data volume from Mars, and a fixed downlink volume of 100 bits expected from Earth.

DTE communication is available anytime during the day. However, it is assumed that the DSN is only available for four hours per day for this mission. Thus the window duration is limited to four hours. Low orbit relay satellites typically have overflights every twelve hours, with a duration of seven to twelve minutes per overflight, depending on the rover's latitude. High orbit relay satellites will be available more frequently than low orbit satellites, with overflights centered approximately six hours apart and with a window duration of seventy-two minutes per overflight. The longest Earth-Mars round-time propagation delay of approximately forty minutes is used. It is assumed that operators on Earth will only be making tactical choices with a response time of two hours. These tactical choices refer to strategic re-planning of a whole day's activities that may require the overnight command cycle to be neglected in delay calculations.

Figure 4-26(a) indicates that the longest delays are associated with DTE. Although DTE has the longest communication window opportunity, the relatively low data rate results in longer delays for large data volumes. Low orbit relay, which has the highest data rate, is only available for short window durations, and, although it represents an improvement over DTE, it does not provide the shortest delays. Better performance is achieved by communicating via a high altitude relay satellite. This method offers more availability than low orbit satellites and provides data rates much faster than DTE, thus achieving a relatively better performance than the other two architectures. This result is consistent with the future plans for high altitude telecommunication



(a)



(b) Zoom of the figure above

Figure 4-26: Communication delays for various architectures

satellites orbiting Mars.

Figure 4-26(b) is a finer scale of Figure 4-26(a) and shows variations of delay with smaller increases of data volume. There are discontinuous jumps in the delays due to finite communication windows. For instance, for data volumes around 20 Mb, DTE has the shortest delay because the command cycle delay is less than one window duration. However, for low orbit relay the command cycle cannot be accomplished within the duration of one communication window, which is approximately seven minutes long. It is completed on the next available communication window, which is half a day apart, thus the discontinuous jumps in communication delay by half a day. Similarly, high orbit relay has discontinuous jumps of approximately six hours. Notice that as the data volume increases, the crossover occurs and DTE becomes less efficient, since the relatively low DTE data rate essentially starts giving rise to longer delays.

4.10 Autonomy

4.10.1 Responsibilities

The autonomy subsystem is responsible for modeling the effect of implementing different levels of autonomy on the mission science return. Three application fields of autonomy are considered. For each of these applications, the autonomy's performance is assessed in terms of operation time, namely how long it takes the rover to complete various tasks. Autonomy applications include long-distance traverse (site-to-site), short-distance traverse (sample-to-sample within a site), and sample acquisition. In addition, the autonomy subsystem models the effects of whether or not the rover can process data during the night, and the effects of increasing computing power. The autonomy subsystem uses the combination of these capabilities to assess the performance of the rover in terms of the rate of samples analyzed, total number of samples obtained in the mission lifetime, and the time required to perform each of the rover's major tasks.

4.10.2 Modeling assumptions

Since the Mars Exploration Rovers (MER) possess the most recent Mars rover technology, several parts of the code are modeled using MER characteristics as a baseline. For instance, the navigational sensors modeled in the program are identical to those on MER. However, it is easy to update the list of navigation sensors as new ones become available. Similarly, the default flight computer modeled in the code is a RAD 6000. However, its computational power can be modified by playing on the *number of computers* design variable (Section 4.3.2, page 86). Several modeling assumptions are used in the autonomy algorithm. These assumptions fall into two main categories, navigation and sample acquisition and processing. The navigation assumptions are:

- Only one type of environment will be encountered during the mission.
- When the rover drives, it drives at a maximum speed provided by the *Rover* module.
- Driving over a rock does not slow the rover down.
- The time required to turn the rover 90^0 is approximated to one minute
- For low levels of autonomy, the rover will not drive itself farther than the terrain seen by the Navcam images processed on the ground.
- All samples in a site are assumed to be the same average distance apart.
- The rover will only perform one reconnaissance per site.

The sample acquisition and processing assumptions are:

- Only one acquisition tool is used per location at a site.
- There is no parallel processing. Instruments process samples sequentially.
- Multiple instruments can process one sample.
- The user determines the samples to be acquired once per site during reconnaissance.

- The *Instruments* and *Acquisition* subsystems provide their own power requirements to the power module.

- There is no remote analysis. All samples to be analyzed need to be acquired.

Additionally, the mission lifetime is reduced by a 33% margin at the front end of the calculations. For example, the science return of 90 sol mission is collected over 60 sols of actual surface operations. This margin is incorporated for two reasons. First, it makes the science return estimates more conservative. Second, a 33% margin is used in some of NASA calculations to evaluate MER and MSL performances. The user can easily modify this margin by changing the appropriate constants in the *Autonomy* module.

Furthermore, the decision was made to limit the number of autonomy levels to the two outlined in various NASA Science Definition Team (SDT) reports, namely A1 and A3. A1 is defined as the autonomous capabilities of the MER rovers, and A3 is defined as having fully autonomous navigational capabilities. The idea of adding an additional level, A2, into the modeling as somewhere between A1 and A3, was not implemented for several reasons. First, A1 and A3 are already defined in NASA literature, whereas a definition of A2 is ambiguous. Second, the trades between A1 and A3 would be sufficient since A2 would have performance levels somewhere between those of A1 and A3. Thus, the information that could be gleaned from calculating the effects of a new level of autonomy seems to be extraneous. If a level A3 of autonomy is to be achievable in the near future, an intermediate level for comparison is irrelevant.

Finally, the *Autonomy* does not include any cost model to evaluate the cost of future levels of autonomy. Not enough information was found to accurately determine how much it will cost to develop, test, validate, and employ a future autonomous capability. Therefore, the goal of this module is to focus on the value of autonomy in terms of performance. Still, Section 5.6 (page 170) demonstrates that this decision does not restrict MSE's applications, for it provides a method to quantify the maximum budget under which A3 autonomy should be developed.

4.10.3 Design Flow

The autonomy subsystem is made up of two separate modules on the N^2 diagram (Figure 4-3, page 84). The first module, *Autonomy Traverse*, determines the rover's theoretical average speed based on its actual driving speed and path planning time, as explained in Section 4.7.5. It then computes the number of driving cycles, N_{cycles}, that the rover performs per sol. This module also interfaces with *Communications* to determine the communication time for a driving sol (Figure 4-24). From those calculations, the power requirements for a traverse sol can be determined and subsequently passed to the power structure, along with the initial value for N_{cycles} (Figure 4-21). Then, once N_{cycles} is updated by the *Power* module, the *Autonomy Mission* module computes the site-to-site traversal and sample approach times. Next, it calculates the time it takes the rover to reconnoiter a site, process instrument data, and communicate data to Earth.

After the time required for all of the rover's tasks are calculated, a loop in the code is initiated that begins with the mission lifetime, less the aforementioned 33% margin and a rover deployment and egress time. Based on the execution order of the tasks the rover must perform, the time for each task is sequentially subtracted from the lifetime until the lifetime is depleted.

The on-site operations are modeled as follows (Figure 4-27):

1. Reconnaissance of a site is performed, information is sent back to the user on Earth, and then a reply is received, specifying which samples to analyze. The total time required for these events is stored as the reconnaissance time. The rover performs only one reconnaissance per site.

2. The rover then approaches a sample, acquires it, processes it with its instruments, and returns the data to Earth.

3. Step 2 is repeated until the specified number of samples for the site has been obtained (as specified in the science vector).

4. Once on-site operations are completed, the rover moves on to the next site by

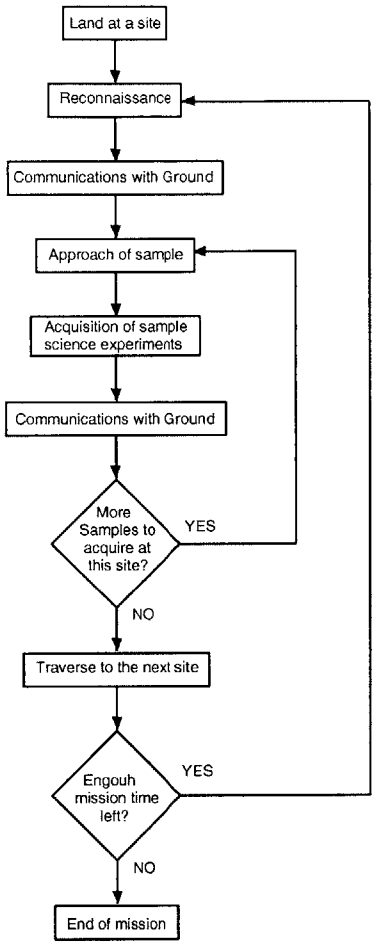


Figure 4-27: Autonomy module program flow

subtracting the previously calculated site-to-site traversal time from the mission lifetime, and then returns to step one, above. It is assumed that the rover lifetime begins at a site.

After each step 2 in this sequence is completed, a tracker updates the number of samples analyzed and adds the amount of data obtained from the science instruments to the total amount of data sent to Earth. When the mission lifetime has been exhausted, the total number of samples analyzed and data obtained will have been calculated. Notice also that the time for step 2 in the on-site operations is re-calculated for each sample to account for consumables in the acquisition tools and science instruments, which may become exhausted.

4.10.4 Validation

The *Autonomy* module was validated using one of the mission scenarios from the MER Mission Plan as shown in Figure 4-28. The operations of the MER rover as

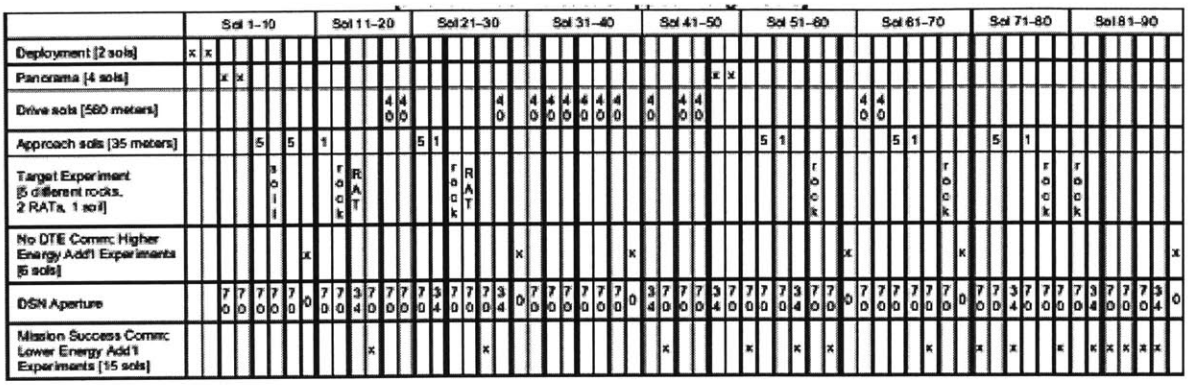


Figure 4-28: Sample of MER mission scenario

modeled by MSE are checked against this scenario in Figure 4-29. Each column

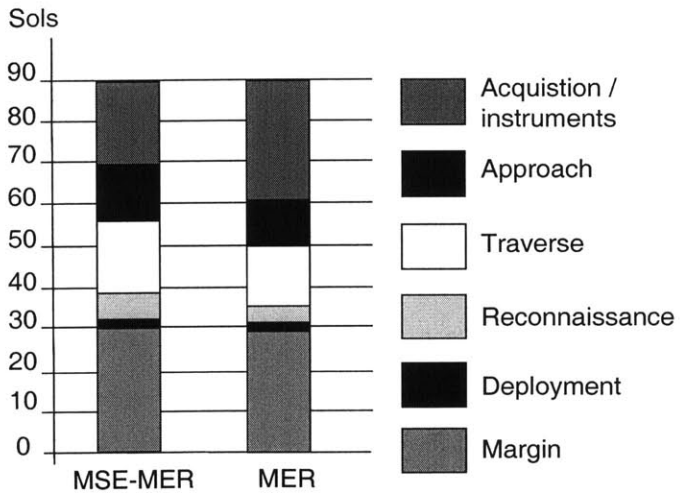


Figure 4-29: Autonomy validation

in this figure represents the breakdown in sols of the various tasks that the rover completes during its lifetime. The first column represents MER as modeled by MSE, and the second column represents MER based on the above mission plan. The figure shows that the required time for each surface operation is modeled closely but not

Table 4.7: Sols required for surface operations using A3 autonomy

Surface operations	**MSL-MSE [sols]**	**MSL-SDT [sols]**
3000 meters traversal	10	8
Sample approach	0.03	< 5
Approach, acquisition and rock processing	3	3

exactly. This is because of the difficulty of making a mission scenario general enough to be applicable to both rovers with A1 autonomy and those with A3 autonomy. For example, MSE models MER with more sols dedicated to reconnaissance because MSE's mission scenario assume a reconnaissance step is made upon arrival at each new site. Now, in reality for MER the sites are not so far apart that a reconnaissance step is necessary at each of them. But MSE's scenario is adapted to MSL's mission whose site to site traverse is 2500 meters and which, therefore, requires reconnaissance at each site.

The validation of a MER-like rover is only a validation for the model of the A1 level of autonomy. The validation of the A3 autonomy level model is done by using MSL as modeled by the 2001 MSL SDT report [7]. The models of MSL by the SDT report and by MSE are compared in Table 4.7 for various surface operations. The results derived from both models are consistent, therefore, the A3 autonomy modeling by MSE is validated.

4.11 Cost Module

The *Cost* module comprises design and operations costs, it does not calculates costs related to launch or entry descent and landing phases. Both design and operations costing models are simplistic and the overall *Cost* module is not reliable in its current state.

Design cost The design cost is the sum of the costs of the individual subsystems: *Acquisition*, *Instruments*, *Communications*, *Power* and *Rover*. *Acquisition* reports actual costs suggested from their references when available and estimated costs when

these actual costs were not available. The *Instruments* module costs each of its instrument at a default price of $10 million. The *Communications* subsystem team uses Wertz and Larson's relationships [56] and NASA references for their cost model, which is a function of the usage of the Deep Space Network. The *Power* subsystem uses several references including [56] to cost batteries and solar panels, and costing references regarding RTGs provided by Joe Parrish, president of Payload Systems Inc. Finally, the rover engineering costs (without its payload) are estimated based on a relationship provided by Robert Shishko from the Jet Propulsion Laboratory. This equation is homogeneous of degree 1/5, which means that if the mass and stowed envelope of the rover are doubles, costs go up about 15%. The cost model does not include any kind of cost for the development, validation or integration of autonomy.

Operations cost The operations cost is a linear function of the mission lifetime. The *Cost* module uses Pathfinder, the only rover for which monetary information regarding operations was found, as a reference point for that function. Thus, operations cost $14 millions every 53 sols of mission lifetime. Notice that operations cost as modeled does not depend on the level of autonomy of the rover. In reality, a rover with high levels of autonomy should require less operations staff on ground in order to make long mission lifetimes affordable. As the contribution of autonomy cost is not included in the design cost, it is also not included in operations cost, so that the *Cost* model as a whole does not capture any kind of autonomy cost. Paradoxically, this *Cost* model can then help to define the monetary value of autonomy by determining the maximum budget under which it should be developed and validated (Section 5.6, page 170).

The costs results for MER and MSL as modeled by MSE are shown in Table 4.8. The actual cost for the two MER rovers is $804 million, including the two launchers and operations for each rover's 90-sol prime mission [25]. The Delta II launchers are about $60 million each. However, the mission was first estimated to be $688 million but cost grew primarily to make the tight schedule. MSE does not model such scheduling issues; based on the latter cost estimation, the design and launch

Table 4.8: MER (1 rover) and MSL costs as modeled by MSE in FY09

	MER cost [$\$M$]	MSL cost [$\$M$]
Design	100	2190
Operations	30	165
Total	130	2355

costs of one MER rover are $224 million, which is 40% higher than MSE's evaluation. It is not unusual to use a 40% margin on a cost evaluated during the conceptual phase, still the MSE cost module need improvement.

4.12 System Validation

In this section, MSE is subjected to a credibility test. The science scenarios and certain design decisions are selected to emulate two rovers that have already been built – Sojourner and MER – and one that has been studied in depth at the Jet Propulsion Laboratory (JPL) – the MSL rover. The science and design vectors chosen to represent MSL in the MSE model are shown below:

$$V_{Science} = \begin{bmatrix} 49 \\ 1 \\ 2500 \\ 20 \\ 20\% \\ [-40, -5] \\ South \\ 185 \end{bmatrix} \begin{matrix} [kg] & \text{science payload} \\ [-] & \text{samples per site} \\ [m] & \text{site to site traverse} \\ [m] & \text{site diameter} \\ [-] & \text{rock coverage} \\ [^0] & \text{landing latitude} \\ [-] & \text{landing hemisphere} \\ [^0] & \text{areocentric longitude at landing} \end{matrix} \tag{4.38}$$

$$V_{Design} = \begin{bmatrix} 500 \\ 0.6, 0.7 \\ 1 \\ RTG \\ DTE \text{ and } X-UHF \\ A3 \\ A3 \\ A3 \end{bmatrix} \begin{matrix} [sol] & \text{lifetime} \\ [m] & \text{wheel size} \\ [-] & \text{number of computers} \\ [-] & \text{power source} \\ [-] & \text{communication type} \\ [-] & \text{long distance autonomy} \\ [-] & \text{short distance autonomy} \\ [-] & \text{acquisition autonomy} \end{matrix} \tag{4.39}$$

MSL's wheel size ranges from 0.5 meters to 0.7 meters, according to various references ([6] and [8]). Two wheel diameters – 0.6 and 0.7 meters – are kept for validation because they seem to represent the latest consensus.

Table 4.9 gathers all the validation information about MSE's ability to model the three rovers reliably. The rovers are organized at the top of the table as bi-columns –

Table 4.9: MSE validation table

	MPF		MER		MSL	
	MSE	Sojourner	MSE	Spirit	MSE	JPL design
No. samples [–]	3	N/A	**6**	**9**	28-33	28-74
Total mass [kg]	14.5	10.5	144	168-185	626-846	600-900
Science [kg]	1.44	1.35	16.5	15.5	49	49
Mobility [kg]	1.5	N/A	16	N/A	**223-355**	**173**
WEB [kg]	1	<2	17	N/A	89-136	96
Arm [kg]	0.05	N/A	4.3	3.5	**10-12**	**30**
Mast [kg]	0	0	2	N/A	**10-14**	**30**
Avionics [kg]	1.3	1	38.5	30	61	50
Comm [kg]	0.2	N/A	16	N/A	**16**	**28**
Battery [kg]	**5.46**	**1.24**	21	18-44	55-75	44
Total power [W]	16	15	140	100-140	272	220
Max. speed [m/s]	0.01	0.01	0.05	0.05	0.1-0.11	0.05-0.1
Avg. speed [m/s]	**0.004**	**0.007**	0.01	0.01	0.03	N/A

Mars Pathfinder, Mars Exploration Rovers, and Mars Science Laboratory. For each one, the right column represents the rover as it is built or modeled by JPL, the left column represents the same rover as modeled by MSE. The rows are design features deemed relevant to assess the reliability of MSE's models; they include science return,

mass break-down, power and rover speed. The major discrepancies – larger than 20% – are highlighted in bold characters.

The discrepancies regarding the Sojourner design concern the mass of its batteries and its average velocity. The former can be explained by the fact that Sojourner is a peculiar rover in the sense that it used primary batteries – non rechargeable batteries. Regarding the latter discrepancy, MSE uses the notion of *driving cycle* to model driving operations and to define the average speed of rovers with A1 and A3 autonomy; this notion may not apply to the Sojourner case, which had almost no on-board autonomy.

According to several sources, the reference mass of the MER Spirit rover rolling on the surface is either 168 kilograms or 185 kilograms. MSE models Spirit with only 144 kilograms, which is still within 20% of the smallest reference mass. Notice that all the numbers in the table, and the masses in particular, do not contain any margin. Usually during pre-phase A – design phase for which MSE is targeted – a margin of 30% is added to total mass estimations; such a margin would bring Spirit's mass as modeled by MSE to a value of 187 kilograms, very close to the upper-bound of the two reference masses. The discrepancy regarding the science return is due to the difference between the actual mission scenario of MER and the one modeled by MSE (Section 4.10.4, page 136). The mission scenario of MSE must be applicable to both A1 autonomy rovers (MER) and A3 autonomy rovers (MSL), and therefore, does not match MER's plan exactly.

The MSL design contains the most discrepancies, but it is the design for which there is the most uncertainty regarding its reference values. The column entitled *JPL design* tries to gather the most information about the current MSL design. However, this information comes from various sources and stages of design, and it is sometimes two or three years old. Additionally, when MSE's column contains two entries the lower one represents the rover with a 0.6 meter wheel and the higher one represents the 0.7 meter wheel rover. The MSL case shows the the limits of applications of several models, notably arm, mast and suspension hardware. The simple-beam structural model does not stand for large rovers like MSL, which is as big as a minivan.

The figures below show single-axis trends that can be used by mission designers for comparison with their own models. Figure 4-30 shows the behavior of the total mass of a MSL-like rover as its wheel diameter increases from 0.2 meters to 1 meter. Within this range the mass can be approximated to a polynomial function of the

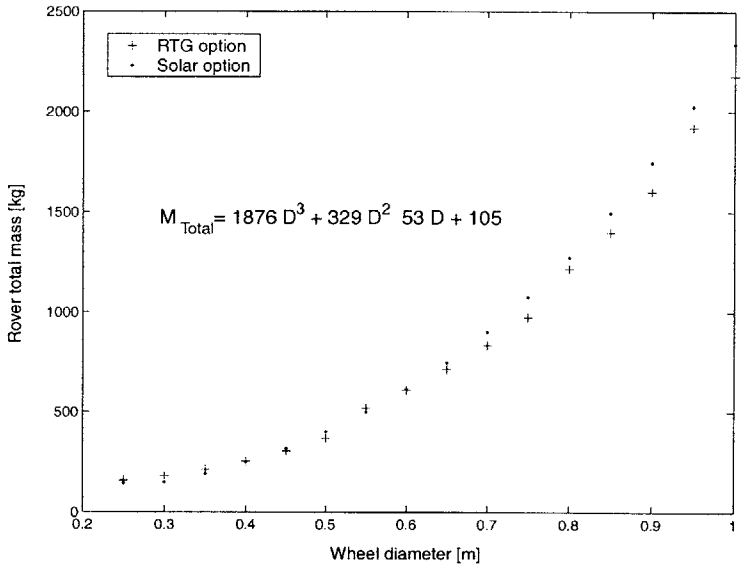
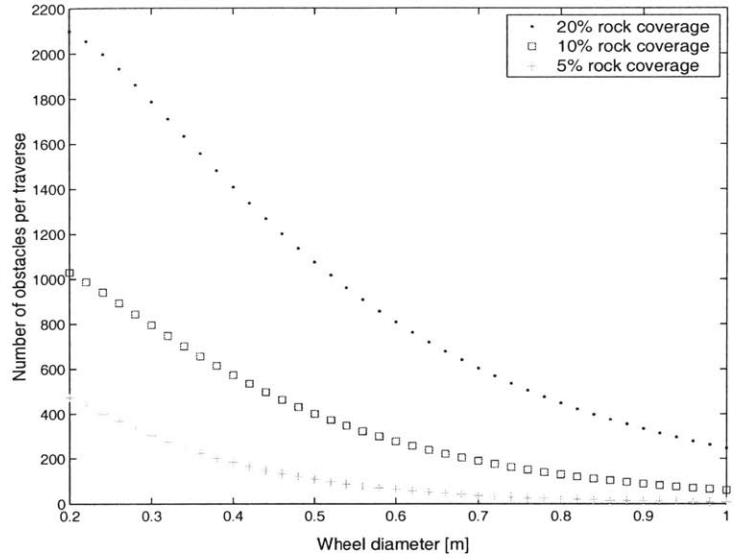


Figure 4-30: Rover mass as a function of the wheel size for solar and nuclear options

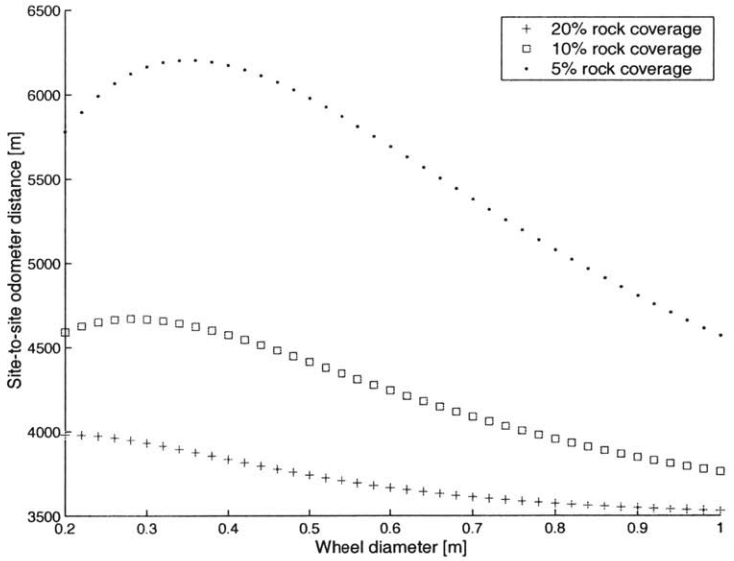
third order in wheel diameter.

Figure 4-31 shows the traverse ability of a MSL-like rover on terrains of various rock coverages. Figure 4-31(a) shows that as expected the number of obstacles encountered by the rover during a site to site traverse decreases as its wheel diameter increases. An obstacle is defined as a rock that the rover cannot traverse and must bypass. Since a rover can traverse rocks as large as its wheel size, the bigger its wheel is, the more rocks it can traverse and the less obstacles it has to go around.

Figure 4-31(b) shows the resulting odometer distance followed by the rover for an equivalent straight distance – as the crow flies – of 2500 meters. Interestingly, the graph shows a region of wheel sizes for which the odometer is maximum. Three phenomena interact in the calculations of the odometer distance. First, at the scale of the driving cycle (Section 4.7.5), when the rover drives forward a distance of half its length there is a pointing error. The rover does not move in the exact direction



(a) Number of obstacles



(b) Site-to-site odometer distance

Figure 4-31: Traverse ability of a MSL-like rover on terrains of various rock coverages

it is supposed to. These errors add up for each cycle and require corrections which induce more distance to be driven by the rover to meet its target. This phenomenon is handled by the *Autonomy Traverse* module which multiplies the site to site traverse given by the user by a factor of 1.4 in order to obtain the effective distance traversed. The second actor in the odometer distance is obstacle avoidance, the rover encounters obstacles on its way that it must bypass (Figure 4-31(a)). This fact tends to reduce the odometer distance as the wheel size increases. This fact explains why the odometer distance lengthens when the wheel size increases from 0.2 to 0.35 meters. This is due, in fact, to the buffer distance that is left between the obstacles and the rover during bypass operations. When the rover goes around an obstacle the distance between the center of the rover and the obstacle is half the width of the rover and the buffer distance, which is set also to half a rover width. Therefore, the total distance between the rover's center and the obstacle is a rover width. Now, as the wheel gets larger the rover width increases, hence, the rover must drive more distance to go around the rock. This fact tends to increase the odometer distance as the wheel size increases. The last two trends when added give the bell shape to the curves shown in Figure 4-31(b). A more complete study of the cost-benefits of large suspensions for rovers is presented in the next chapter (Section 5.3).

4.13 Conclusions

The reliability of MSE is not perfect. Still, the weaknesses of the model are identified. They concern chiefly the life cycle cost calculations, and the hardware modeling of rovers larger than MSL. Designers who have access to expertise in rover systems can easily fix these specific weak points and improve MSE's reliability. The validation table (Table 4.9) and the Figure 4-31 prove that MSE is able to model with system-level accuracy most features from existing rover designs and also to capture trade-offs inherent to rover systems. The next chapter demonstrates that in its current state MSE is already a tool that addresses pertinently design issues relevant to the rover mission community.

Chapter 5

MSE's Analysis Capabilities

5.1 Organization of the chapter

The purpose of this chapter is to demonstrate MSE's *usefulness* and *versatility*, as previously defined in Section 4.2.1 and that, consequently, it is a tool useful to the rover-mission community. This study chooses the 2009 Mars Science Laboratory (MSL) mission as a case example to prove MSE's ability to answer various critical design questions. MSL is the successor to the 2003 Mars Exploration Rovers (MER) mission in NASA's Mars Exploration Program. In comparison to MER, current plans call for MSL to have improved autonomous capabilities, to use nuclear power sources instead of solar power, and to be twice as large. These decisions are believed to increase science return through better use of operation time, longer mission lifetime and improved traverse abilities. This chapter evaluates these design decisions by assessing the cost-benefits of the technology they involve, and by exploring alternative solutions. It is organized in four sections each answering one of the subsequent questions:

1. What are the benefits of oversizing a rover's suspension? (Question asked by Charles Whetsel from JPL)

2. Is solar power a viable option for the MSL?

3. How big should the MSL rover be and how long should the mission last? (Ques-

tion asked by Charles Weisbin from JPL)

4. What are the cost-benefits of autonomy?

In addition to drawing pertinent conclusions from these questions, the intent of this chapter is to demonstrate the flexibility with which MSE can be used to deal with various trades. For example, in answering Question 3, the single-rover mission trade space of MSE is easily extended to include multi-rover missions.

5.2 Metrics

In order to make quick and grounded design decisions, simple metrics are necessary that are also relevant to the trade-offs being examined. As shown in Figure 4-3, one run of a trade space exploration results in a database of mission architectures. Each data point is a set of design variables that define it unambiguously, and of system-level characteristics that are calculated during the Rover Modeling phase. The latter includes the rover's total mass, number of samples analyzed, communication data, total odometry, number of sites visited, and more specific data, such as battery mass or length of the mast. All this information is conveniently available to the user via a Matlab graphical interface (Figure 5-1). It is among this pool of design and performance variables that metrics are chosen to best highlight the trade-offs.

The key notion in technology development is the cost-benefit trade-off. It is quantified by evaluating the investment and expected return on investment of a technology. If the return is deemed worth the investment, the technology can be developed. In the context of rover missions to Mars, the benefits are the science return of the mission, and the costs are what the tax-payers pay to finance the mission. Now, science return is difficult to quantify because it is mostly qualitative, and cost is delicate to model because it covers a multitude of aspects.

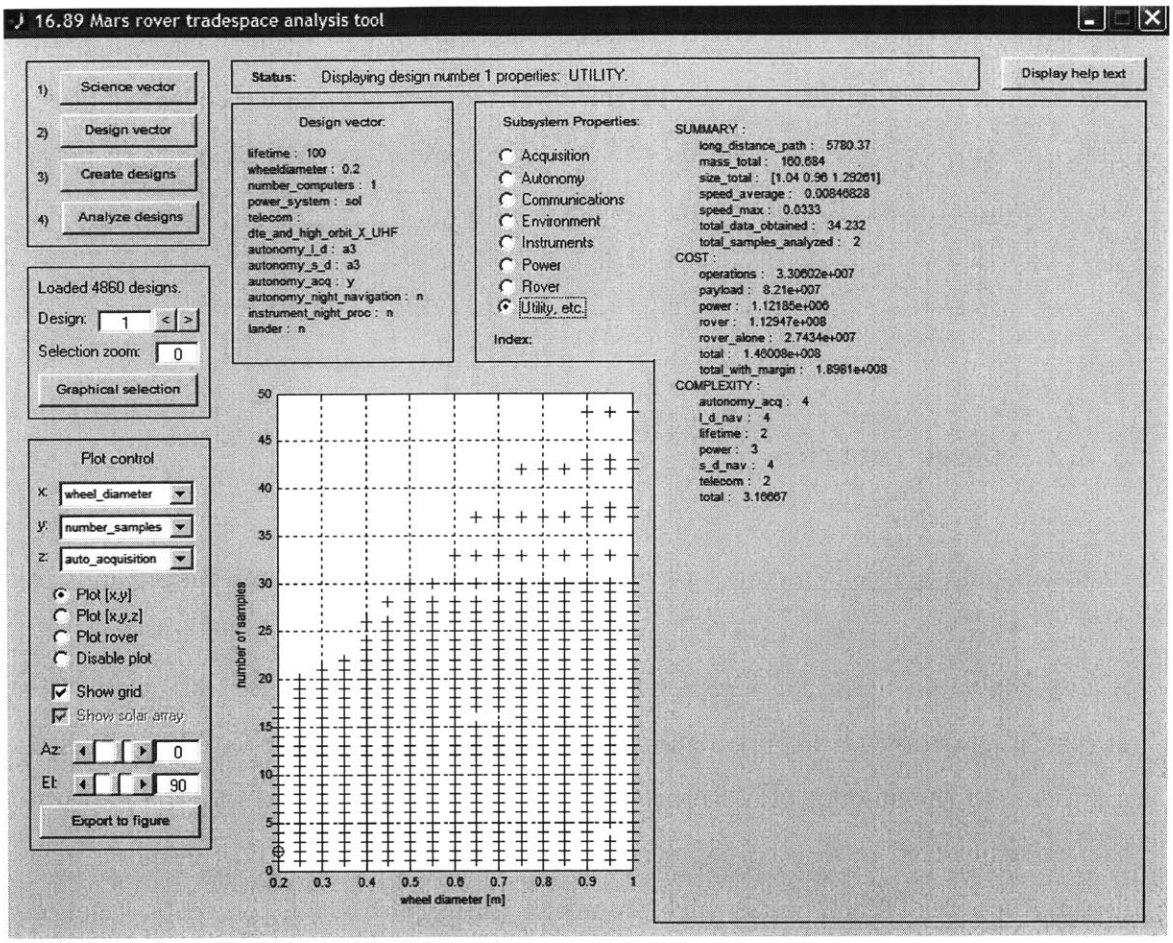


Figure 5-1: MSE's main graphical interface, designed by Mark Hisltad

5.2.1 Science return metric

The scientific value of a mission is driven more by the quality of the science returned than by its quantity. The word *quality* spans notions from unprecedented science to integrity of the data returned. Quantifying science return in detail is challenging and not within the scope of MSE.

In our system-level model, several variables can play the role of a science return metric: total data returned, distance traveled, or number of samples collected. The last one is selected because it is intuitive and is commonly used to define the science objectives of Mars surface missions. This science metric is also consistent with the science utility used in the article by Lincoln et al. [42] which increases chiefly with

the number of samples analyzed. However, for this science utility, the benefits of collecting a new sample decrease as the total number of samples collected grows. This is an aspect of science relevance that is not captured by the current MSE model which assumes that all samples are equally worthy.

5.2.2 Cost metrics

Cost depends on a lot of factors, spread over several design phases and taking multiple forms. This is the reason several metrics are considered to evaluate mission cost, whereas only one metric for science return is judged to be enough. Besides evaluations of cost in monetary terms, mass and lifetime are used as surrogate variables. Mass is a measure of the amount of hardware that constitutes a rover. Therefore, it captures the effort involved in the development and design of that rover. Furthermore, mass directly impacts launch costs which are a substantial part of mission costs. Lifetime is a measure of a mission's operations costs, which are not negligible for missions like MSL that could last several hundreds sols. Surrogate variables are actually required because the *Cost* module does not model mission costs reliably enough (Section 4.11). Therefore, unless explicit costs are required, mass and lifetime are preferred metrics to evaluate mission cost.

5.2.3 Trade space example

As explained in the former chapter (Section 4.3.2), a trade space is defined by a science vector, $V_{Science}$, which sets all mission science requirements and a design vector, V_{Design}, which defines the technology alternatives to build a rover. The trade space

below is used in most of the subsequent sections:

$$V_{Science}= \begin{bmatrix} 49 \\ 1 \\ 2500 \\ 20 \\ 20\% \\ [-5,+5] \\ \text{North} \\ 330 \end{bmatrix} \begin{matrix} [\text{kg}] & \text{science payload} \\ [-] & \text{samples per site} \\ [\text{m}] & \text{site to site traverse} \\ [\text{m}] & \text{site diameter} \\ [-] & \text{rock coverage} \\ [^0] & \text{landing latitude} \\ [-] & \text{landing hemisphere} \\ [^0] & \text{areocentric longitude at landing} \end{matrix} \tag{5.1}$$

$$V_{Design}= \begin{bmatrix} \text{100 to 700, step of 200} \\ \text{0.20 to 1, step of 0.02} \\ \text{1, 1.5, 2} \\ \text{solar, RTG} \\ \text{DTE and X} - \text{UHF} \\ \text{A3, A1} \\ \text{A3, A1} \\ \text{A3, A1} \end{bmatrix} \begin{matrix} [\text{sol}] & \text{lifetime} \\ [\text{m}] & \text{wheel size} \\ [-] & \text{number of computers} \\ [-] & \text{power source} \\ [-] & \text{communication type} \\ [-] & \text{long distance autonomy} \\ [-] & \text{short distance autonomy} \\ [-] & \text{acquisition autonomy} \end{matrix} \tag{5.2}$$

All the designs of this trade space are plotted on a number of samples versus mass pair of axes in Figure 5-2. This same trade space is analyzed from different perspectives in the subsequent sections in order to analyze suspension, power, multi-rover missions and autonomy trades.

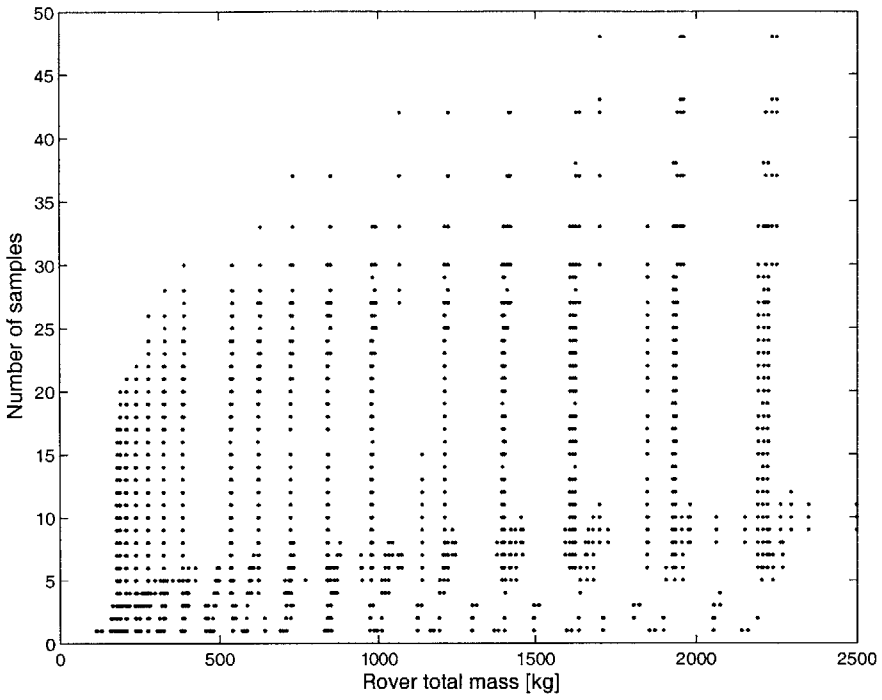


Figure 5-2: Trade space of MSL-like missions

5.3 What are the benefits of oversizing a rover's suspension?

5.3.1 Possible benefits of oversized suspensions

The size of a rover's suspension is usually scaled according to the payload that it must transport. Now, as stated in the former chapter (Equation 4.21), the suspension's dimensions are a direct function of the wheel diameter, which also scales with the size of the biggest rock the rover can traverse. Thus, providing a rover with a suspension larger than that required by its payload improves its traverse ability and, thereby, the mission's science return. Indeed, a rover with better ground clearance spends less time going around obstacles on site-to-site traverses; therefore, it visits more sites in the same time frame, and collects more samples overall. The gain in terrain clearance comes, however, at the cost of a heavier suspension system. The trade-off problem is then to identify the suspensions that maximize science return (through better traverse

ability), and minimize mass. The analysis of the cost benefits of larger suspensions is pertinent to the case of the MSL rover, for which various wheel sizes have been considered, ranging 0.5 to 0.7 meters. Notice that going over larger rocks imposes at least a perceived risk. A conservative operator may still choose to go around, rather than over, and thereby negate the benefit of oversized suspension. This risk management issue is not modeled here.

5.3.2 Analysis

Since the wheel diameter is the key parameter in the suspension's geometry, the present analysis compares the mass and traverse abilities of rovers with various wheel sizes. The impact of the wheel size on the total mass of the rover is discussed in the former chapter (Figure 4-30), which shows that the mass increases monotonically with the wheel size. The paragraphs below focus on the other aspect of the trade-off; they characterize the behavior of the science return as a function of the rover wheel size for missions with science scenarios similar to MSL. The design vector is defined in order to vary the wheel size in the vicinity of MSL's nominal value of 0.7 meters. The mission lifetime is allowed to change as well, in order to check for dependencies of the trade-off on this variable. The design vector entries are shown below:

$$
V_{Design} = \left[\begin{array}{c|l|l}
100 \text{ to } 700, \text{ step of } 200 & [\text{sol}] & \text{lifetime} \\
0.20 \text{ to } 1, \text{ step of } 0.02 & [\text{m}] & \text{wheel size} \\
1 & [-] & \text{number of computers} \\
\text{RTG} & [-] & \text{power source} \\
\text{DTE and X} - \text{UHF} & [-] & \text{communication type} \\
\text{A3} & [-] & \text{long distance autonomy} \\
\text{A3} & [-] & \text{short distance autonomy} \\
\text{A3} & [-] & \text{acquisition autonomy}
\end{array}\right] \tag{5.3}
$$

A total of 164 designs are thus evaluated and plotted on a number of samples versus wheel diameter graph (Figure 5-3). The three paragraphs below analyze this figure

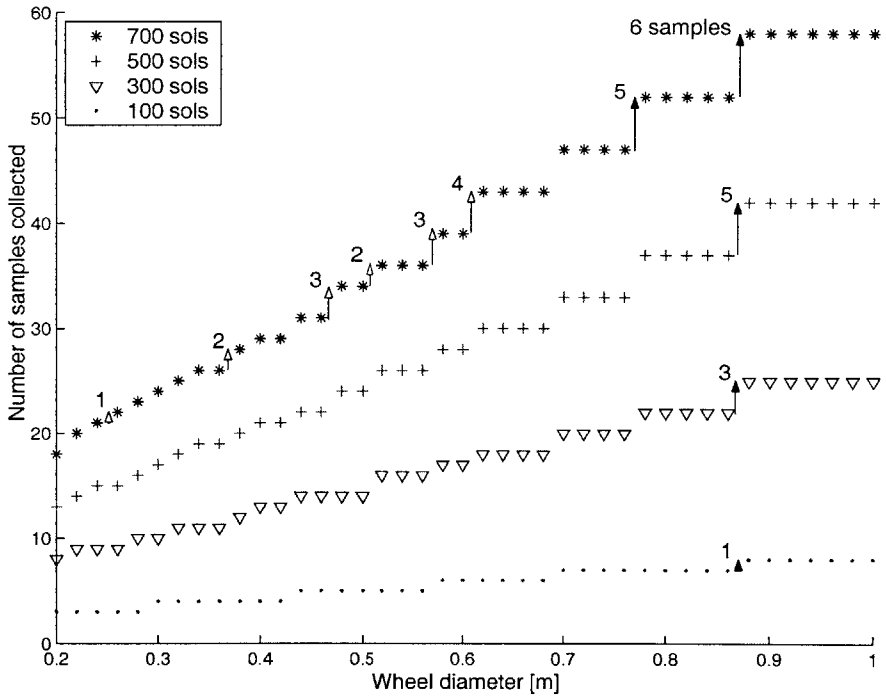


Figure 5-3: Science return of MSL-like missions vs. wheel size for various mission lifetimes

along mass and lifetime axes independently and then examine the coupling between the two.

For design points that have the same mission lifetime, the science return globally increases with the wheel size. With a discretization of the wheel size by a step of two centimeters this behavior appears to be, however, not continuous; the number of samples is a step function of the wheel size. This step function has a noteworthy trend, its steps are, indeed, wider and higher as the wheel size increases. In other words, as it gets more costly (in terms of mass) to increment the wheel size to the next step, it is also more rewarding (in terms of science return). The interpretation for this trend is twofold. First, as illustrated in Figures 4-31 and 5-4(a), when the wheel gets larger, the site-to-site odometer distance globally decreases (the explanation for the bell curve in Figure 5-4(a) is presented in Section 4.12). Second, as the wheel gets

larger the average velocity increases. Now, by definition, the time spent to go from one site to another is simply the odometer distance divided by the average velocity (Figure 5-4(b)). As expected, the figure shows that rovers with larger suspensions spend less time on the site to site traverse.

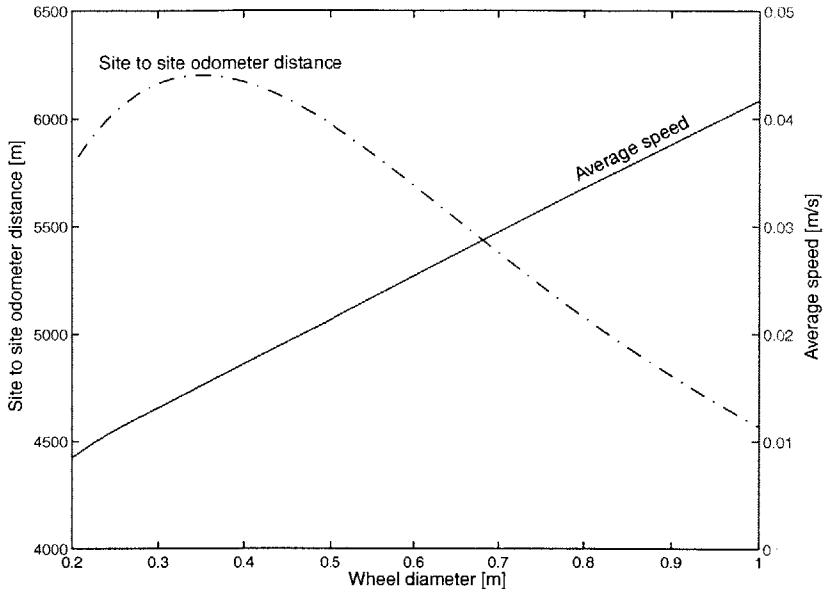
The odometer distance also changes depending on the ruggedness of the terrain that the rover traverses. Figure 5-5 shows the performance of 500 sol missions for various terrain ruggedness. As expected, terrains with high ruggedness penalize the mobility of rovers. Notice that a rover with a 0.5 meter wheel on 10% ruggedness terrain has the same return as smaller rover with a 0.38 meter wheel 5% terrain. Furthermore, the rate of increase of science return per unit length of wheel decreases for rockier terrains. In other words, the average slope of the number of samples versus wheel size function diminishes as the terrain ruggedness increases. Hence, not only ruggedness penalizes mobility, but its impact is stronger on larger suspensions.

Going back to the examination of Figure 5-3, one notices by comparing designs with similar wheel sizes that the steps tend to get higher as the lifetime lengthens. As missions get longer, the advantage of having a large suspension gains in importance. This fact is intuitive since a longer lifetime means more traverses, and therefore, more opportunities for the larger suspensions to take advantage of their better terrain clearance. For example, increasing the wheel diameter from 0.68 to 0.70 meters, which is one of the wheel sizes considered for MSL, enables a gain of one sample every 200 extra sols of the mission lifetime.

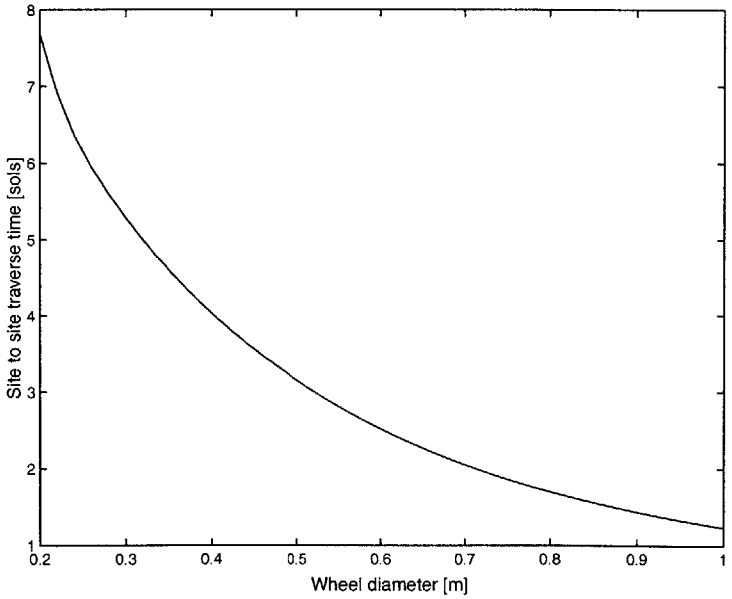
The optimal wheel sizes are those for which the number of samples versus wheel size function makes a step. The optimal wheel sizes define the designs that belong to the Pareto front. For designs with 300 sol lifetimes, these Pareto wheel sizes are:

$$D_{Pareto}^{300\ sols} = \{0.20,\ 0.22,\ 0.28,\ 0.32,\ 0.38,\ 0.40,\ 0.44,\ 0.52,\ 0.58,\ 0.62,\ 0.70,\ 0.78,\ 0.88\} \tag{5.4}$$

A noteworthy result is that the Pareto wheel sizes are conserved across the increasing lifetime axis. For example, the Pareto wheel sizes for designs with 300 sol mission lifetimes, $D_{Pareto}^{300\ sols}$, are included in the one for designs with 500 sol lifetimes, $D_{Pareto}^{500\ sols}$.



(a) Average speed and site to site odometer distance



(b) Time to go from one site to the following one

Figure 5-4: Traverse characteristics as the wheel size increases

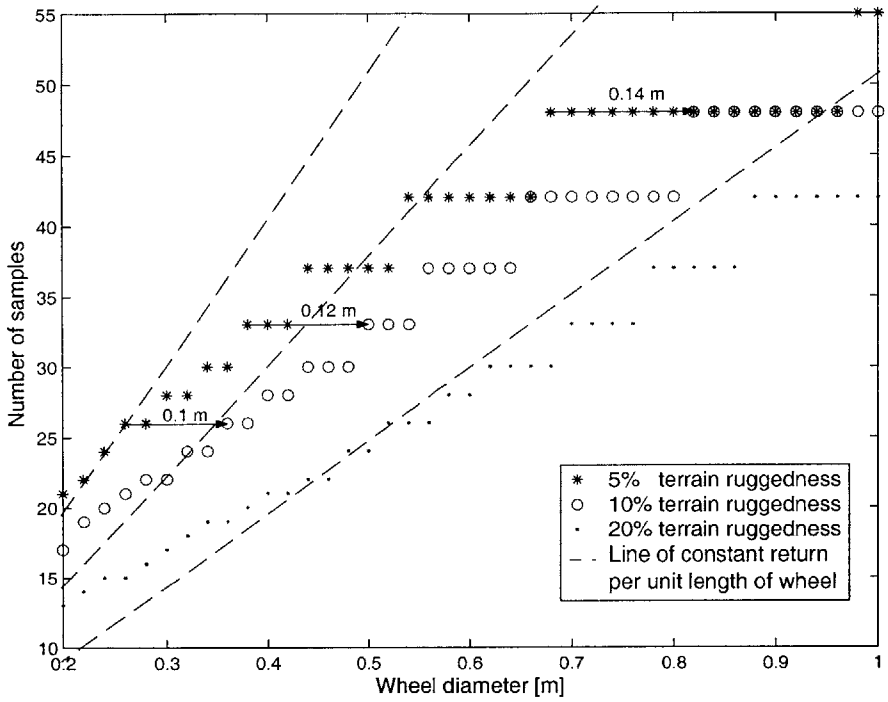


Figure 5-5: Science return of MSL-like missions vs. wheel size for various terrain ruggedness

Interestingly, the Pareto wheel sizes are conserved exactly from 500 sols to 700 sols.

$$D_{Pareto}^{300\ sols} \subset D_{Pareto}^{500\ sols} \tag{5.5}$$
$$D_{Pareto}^{500\ sols} = D_{Pareto}^{700\ sols}$$

This result is relevant to rover design because it permits finding optimal suspensions early in the design process when the mission lifetime might not yet be decided. For example, a wheel size of 0.7 meters is optimal for a MSL mission lifetime of 500 sols, but in case the mission is extended it will still remain optimal.

The same result is true if the total mass of the rover is used as the cost metric. The Pareto front for the number of samples versus mass is conserved across increasing mission lifetimes (Figure 5-6). The front spreads out from designs of 200 to 1500 kilograms, or in terms of wheel size, 0.2 to 0.88 meters, respectively. This proves that mission designers have a wide range of optimal designs to choose from.

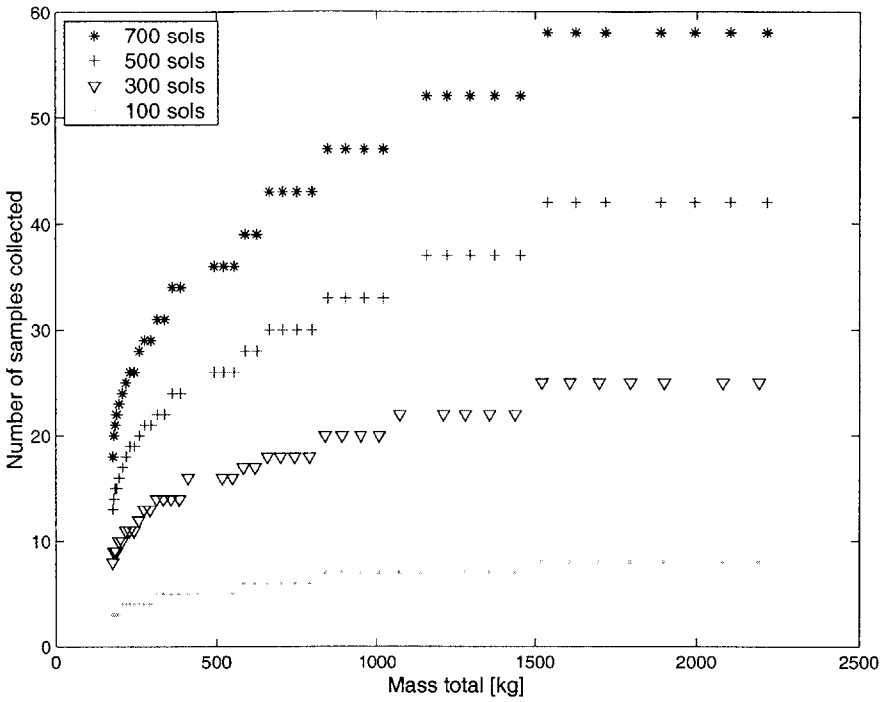


Figure 5-6: MSL-like mission's science return versus mass for various mission lifetimes

5.3.3 Conclusions

This analysis concludes that it is actually beneficial to make rovers larger than otherwise required by their payload. Designs with oversized suspensions have improved science return for terrain with low and high ruggedness, and their benefits increase with mission lifetime and site-to-site traverse. Another advantage of oversized suspensions, not captured by this study, is that they allow payload parts to grow in size. Even if large suspensions are more massive and expensive, they save money due to the fact that payload elements do not have to be miniaturized or custom made, as was necessary on MER. MER's payload customization issue was a critical driver of MER design costs.

5.4 Is solar power a viable option for the MSL?

5.4.1 The case for nuclear power

In order to be allowed the use of a nuclear power source (RTG) on a spacecraft, mission designers must demonstrate that their science goals cannot be met with a solar-powered spacecraft. Now, Mars is the furthest planet from the Sun where using solar power is still an option[1]. Nuclear power was used on both Viking landers, the first two probes that operated on the Martian surface. The two lander missions were highly successful, one lasting for three and half years, the other one for six and half years[2]. However, since then landed probes have only used solar power and for missions of a hundred sols maximum.

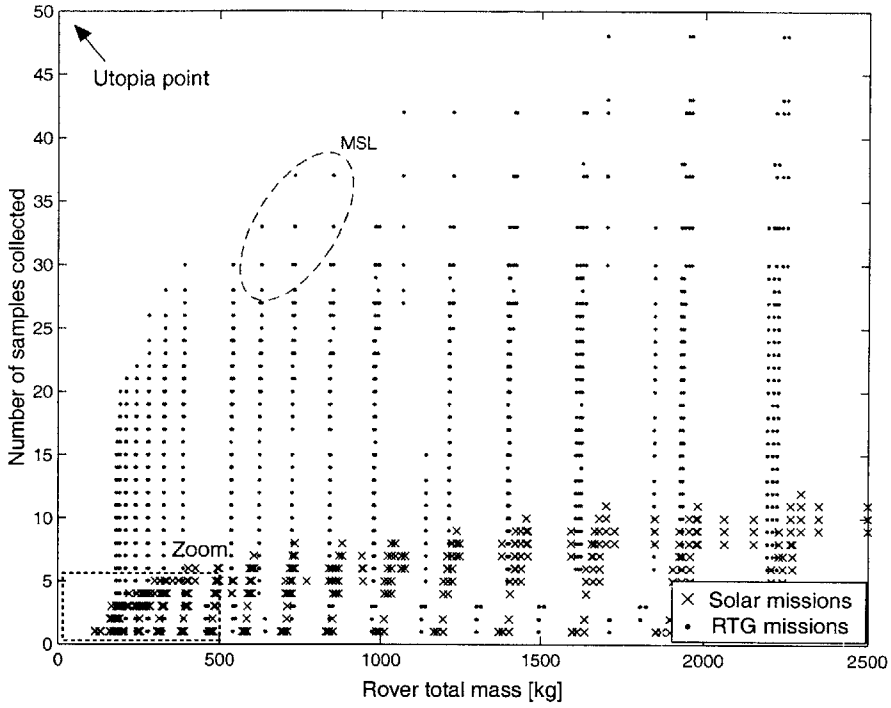
Solar panels can be used on the Martian surface only for a limited period of time due to two environmental reasons. First, Martian dust covers the solar panels continuously, thereby decreasing the amount of solar cells receiving sunlight. Second, rovers powered by solar energy are incapacitated during the Martian winter which has shorter and colder days. This is the reason nuclear power is considered for the MSL rover, whose mission lifetime is almost a Martian year. Still, the demonstration must be made that a solar-powered MSL rover would not be able to meet the mission's goals, which in the MSE model translate to bringing back at least 28 samples.

5.4.2 Trade analysis

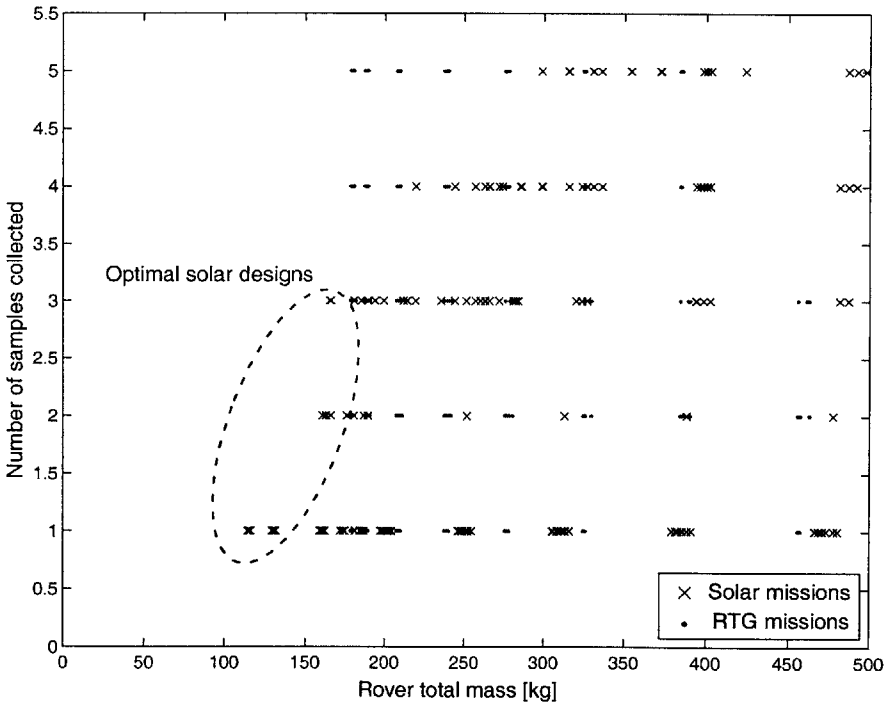
The trade space of MSL-like missions introduced in Section 5.2.3 is plotted in Figure 5-7 with markers that differentiate rovers using solar power from ones using nuclear power. Notice that the nuclear designs tend to lie on vertical lines which are lines of equal mass. The mass of an RTG-powered design does not depend on the mission lifetime, it changes chiefly with the wheel size of the rover. The wheel size determines

[1]The Rosetta spacecraft is actually stretching the limit by using solar power to explore the Churyumov-Gerasimenko comet in the vicinity of Jupiter, 675 million km from the Sun. The mission designed by ESA was launched on March 2, 2004 and will reach the comet by 2014.

[2]The Viking 1 Lander touched down at Chryse Planitia on July 20, 1976 and ended communications on November 13, 1982. The Viking 2 Lander touched down at Utopia Planitia on September 3, 1976 and ended communications on April 11, 1980.



(a) Trade space of MSL-like missions



(b) Zoom of the boxed area showed in the figure above

Figure 5-7: MSL-like mission's differentiated by their power source

the dimensions and, thereby, mass of the suspension system, increments of which appear on the mass axis between each vertical line. The wheel size is the major mass driver (x axis) whereas the lifetime and autonomy levels are the major science return drivers (y axis). As the mission lifetime of a nuclear rover lengthens, its science return increases and the design moves vertically on the plot of Figure 5-7(a), resulting in the vertical lines mentioned above. Furthermore, the lifetime of a nuclear rover is mostly limited by the time the operators on ground are willing to keep it functioning. Regarding the solar designs, rovers with longer operating time need larger panels and, therefore, are heavier. Hence, solar designs lie on slanted lines in the Figure 5-7(a). Moreover, the maximum lifetime, and therefore the science return, of solar designs is limited by the Martian environment.

The difference in performance between solar and nuclear architectures is dramatic. Solar powered architectures are limited to a science return of 12 samples compared to 48 samples for architectures using RTG's. Furthermore, the solar designs that return more than 5 samples are highly suboptimal compared to nuclear ones. The boxed region of Figure 5-7(a) is actually the only region of the trade space where the solar power is the best option, in terms of science return and mass. This region is represented specifically in Figure 5-7(b) which reveals that optimal solar designs are very few and bring back at most 3 samples.

The science scenario specifies a landing date on Mars at the beginning of Spring ($L_s = 330^0$). Even under these good conditions, the longest a solar rover can operate is 200 sols. That amount of time is not sufficient for rovers to meet the science goal of 28 samples; the time spent by the rovers on the site-to-site traverse of 2500 meters is too long. In comparison, the MSE designs that represent the future RTG-powered MSL rover (wheel size between 0.6 and 0.7 meters, computational power between one and two RAD6000) are optimal in terms of science return and mass (Figure 5-7(a)). They fulfill the science goal by collecting between 28 and 37 samples.

5.4.3 Conclusions

None of the solar architectures are successful and, thus, nuclear power systems are required to meet the mission goal. The science scenario of the MSL mission is, in fact, too demanding in terms of mission lifetime, site to site traverse, and payload power draw. However, important aspects of the discussion are not captured by the current MSE model. An argument that goes against the use of solar power is that it is geographically constraining and limits the number of scientifically interesting sites that can be explored. Then, there are two arguments that go in favor of solar power. First, because the packaging of the payload on the WEB is not modeled with enough fidelity, the current hardware model permits a rover even smaller than MER to carry two RTG's. In reality, two RTG's probably could not fit on a rover that small. RTG's are indeed large pieces of hardware and require special packaging. Hence, the lightweight nuclear architectures shown in Figure 5-7(b) would in reality be invalid designs. Therefore, in that same region of the trade space, solar architectures would become optimal. Second, and most importantly, the benefits of nuclear power systems must be offset by their costs. The pre-development costs for the MSL radioisotope power source is $195 million, including development, qualification and cost of nuclear fuel and fueling ($20−30 million alone). Nuclear power sources are far more expensive, by orders of magnitude, than solar plants. However, the development cost of such a technology should be amortized over the several missions which would benefit it. The next section analyzes whether teams of solar rovers working together could meet the MSL mission goal.

5.5 How big should the MSL rover be and how long should the mission last?

5.5.1 Mass versus lifetime Trade-off

This question, raised by Charles Weisbin from JPL, focuses on mission cost expressed in terms of mass and lifetime. Its translation in mathematical terminology is: among

all architectures that provide a given science return, what are the architectures that minimize mass and mission lifetime? The answer to this question for MSL-like missions is illustrated in Figure 5-8. The missions represented in the figure are those

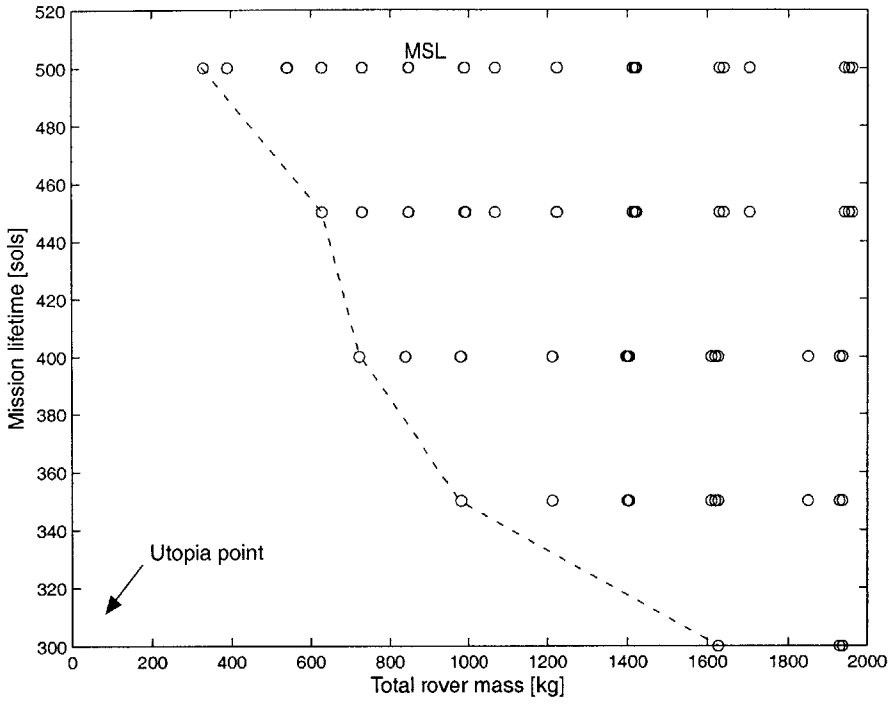


Figure 5-8: Single-rover missions that collect over 28 samples on lifetime vs. mass axes

from the trade space, defined in Section 5.2.3, that meet MSL's science requirement of 28 samples. The figure indicates that only five architectures are optimal; they constitute the Pareto front drawn with a dashed lined. The design that represents MSL in this study is a rover with a 0.7 meter wheel, equipped with full autonomy and a RAD6000. This design fulfills the science requirement by collecting 33 samples. In the figure, however, the point labeled *MSL* appears to be dominated by rovers which have smaller wheels, and are consequently lighter, and have more computational power for a science return of 28 samples. All of the missions shown on this figure involve only one rover operating on the surface of Mars. A more pertinent answer to Charles Weisbin's question should include other mission types, such as multi-rover missions.

This study examines the cost benefits of having multi-rover missions instead of

single-rover missions to complete the science goals of the MSL mission. Multi-rover missions are defined in this chapter as missions that have at least two identical rovers operating independently and simultaneously on the surface of Mars. The MER mission, which operates two identical rovers, each on opposite sides of the planet, is an example of a multi-rover mission. A multi-rover mission is analogous to rovers collecting samples in parallel, whereas a single rover mission collects samples one after the other. Hence, the former minimizes mission lifetime, whereas the latter, based on intuition, would minimize mass. To examine this trade-off in more detail, the trade space of single-rover missions shown in Section 5.2.3 is expanded, so that it includes multi-rover missions, by using the method presented below.

5.5.2 Search method

This three-step method finds all the single and multi-rover missions that meet a given science return requirement.

Step 1: Trade-space exploration A traditional trade space exploration is run; the trade space used in the present calculations is the one defined in Section 5.2.3. The resulting database of architectures is comprised of only single-rover missions, and is extended to multi-rover missions on the third step.

Step 2: Science requirement definition The science requirement selects the single and multi-rover missions of the trade space that return a number of samples within a specified range. The purpose of this requirement is to study a particular science scenario relevant to the mission designers and, additionally, to narrow down the size of the trade space, which otherwise would be too large if it contained all possible single and multi-rover missions. The required science return can be seen as a new variable of the science vector that focuses the science scenario even more.

As an example, the following condition, which expresses the science return of successful missions, $R_{success}$, in terms of the MSL mission, R_{MSL}, focuses the trade space exploration on multi-rover missions that would perform similarly or better than

the MSL mission.

$$R_{success} \in [R_{MSL},\ R_{MSL} + 50\%\ R_{MSL}] \tag{5.6}$$

The trade space is extended to multi-rover missions and at the same time reduced to missions that fulfill the above requirement. This science requirement of Equation 5.6 applies to the rest of this section.

Step 3: Generation of the multi-rover missions The multi-rover missions are created out of the database of single rover missions. Successful multi-rover missions are constituted of rovers that independently return less science than MSL, but have an added return that satisfies the science return requirement (Equation 5.6). The number of rovers in a multi-rover mission is called the multiplicity factor, n_{multi}. As all the rovers in a team are identical, they all collect the same amount of samples. Therefore, the science return of the multi-rover mission, R_{multi}, equals the multiplicity factor times the return of one of its rovers, R_{single}.

$$R_{multi} = n_{multi} \times R_{single} \tag{5.7}$$

Then, to be successful, a multi-rover mission must satisfy the subsequent science return relationship:

$$R_{MSL} \leq n_{multi} \times R_{single} \leq R_{MSL} + 50\%\ R_{MSL} \tag{5.8}$$

Therefore, the algorithm to build all successful multi-rover missions is:

For every single rover mission

If $R_{single} < R_{MSL}$ **then**

n_{multi} is the smallest number of replicates of the current rover

needed to fulfill the science requirement lower-bound

$n_{multi} = C\left[\frac{R_{MSL}}{R_{single}}\right]$

where C is the ceiling function.

If $n_{multi} \times R_{single} \leq R_{MSL} + 50\%\ R_{MSL}$ **then**

the mission with multiplicity n_{multi} and return $n_{multi} \times R_{single}$ qualifies as a multi-rover mission

end if

end if

end for

Notice the advantage of using a full-factorial exploration in Step 1. With just one run of this step, unlimited cases of science objectives scenarios can be analyzed by simply changing the science requirement in Step 2 and going through Step 3; both of these steps are far less computationally expensive than Step 1.

5.5.3 Trade-off analysis

The above method is applied to examine the multi-rover missions that would perform as well or better than the MSL mission as modeled by MSE. The initial single-rover trade space used for Step 1 is the one shown in Section 5.2.3. Step 2 sets the science requirement as in Equation 5.8, and Step 3 is then applied to these sample points. All the successful missions are then plotted on a lifetime versus mass perspective in Figure 5-9. The mission points are represented by different markers according to their multiplicity factors. Notice that for multi-rover missions the total mass, M_{multi}, represented on the plot is the sum of of the masses, M_{single}, of each rover belonging to the mission:

$$M_{multi} = n_{multi}\ M_{single} \tag{5.9}$$

The boxed points belong to the Pareto front for the objectives of mass and lifetime minimization; the utopia point is in the lower-left corner. All the represented architectures return at least the same amount of science as the MSL mission, which is also represented on the plot. On this trade space, the MSL mission point is even farther from the Pareto front than in Figure 5-8; it is dominated by many designs, most of which are multi-rover ones. The missions that dominate MSL are the ones that weigh less and have a shorter lifetime than MSL. These lie at the bottom left of the MSL

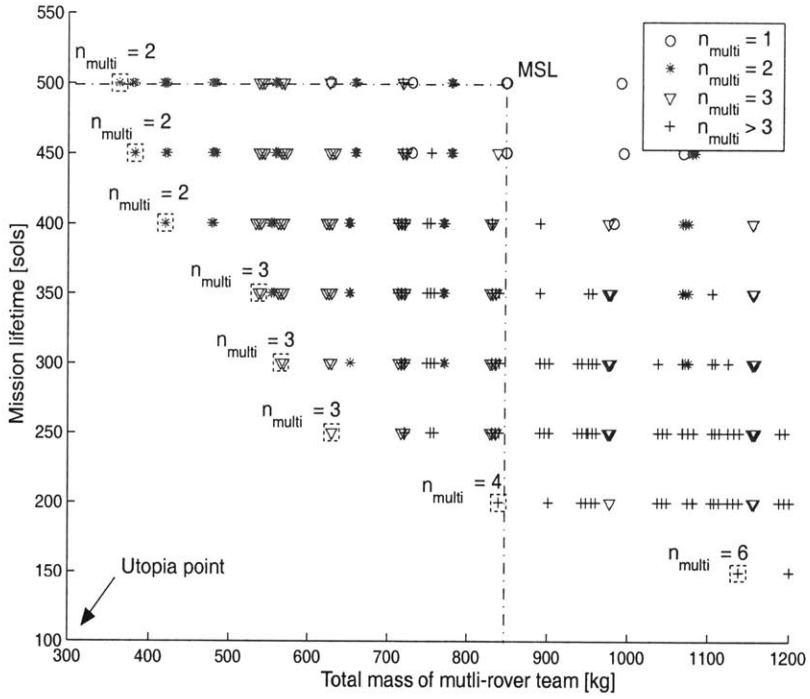
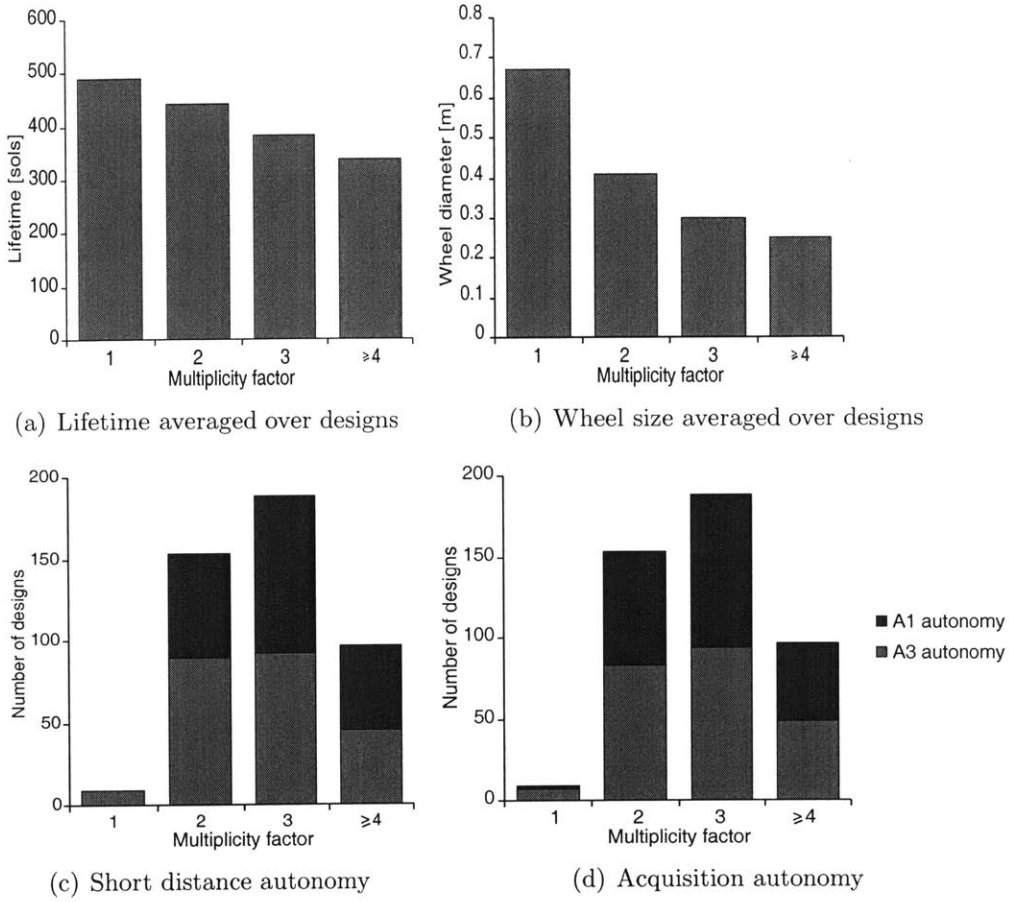


Figure 5-9: Single- and multi-rover missions that perform better than MSL

reference point, in the surface delimited by the two dashed lines and the two axes. The next paragraph examines these designs in further detail.

The designs are regrouped according to their multiplicity factors, against which their design characteristics are compared in Figure 5-10. Figures 5-10(a) and 5-10(b) show the lifetime and wheel sized averaged over designs that have a similar multiplicity factor. As expected, because teamed rovers work in parallel, they collect the required amount of samples in a shorter time than a single rover. In particular, three rovers are enough to reduce the lifetime by 100 sols, namely 20%. Still, even for multi-rover missions, the lifetime is not sufficiently reduced to make solar power a viable solution. All the designs of the trade space shown in Figure 5-9 use nuclear power. Hence, from this result and the conclusion of Section 5.4, it appears that the MSL science objectives can only be fulfilled by RTG-powered vehicles. Additionally, Figure 5-10(b) shows that rovers working in teams do not need to be as big as the single-rover MSL architecture; the latter needs a wheel of 0.7 meters in diameter,



(a) Lifetime averaged over designs

(b) Wheel size averaged over designs

(c) Short distance autonomy

(d) Acquisition autonomy

Figure 5-10: Missions' characteristics as a function of their multiplicity factor

whereas a wheel of 0.25 meters is enough for teams of four rovers and more. This last result suggests that the current single-rover architecture considered for MSL could be replaced, for better performance, by four MER-like rovers, assuming that each can carry the science payload of MSL and that four RTG's are affordable. However, this latter solution would incur other costs due to the necessity of customizing and miniaturizing the science payload.

Figures 5-10(c) and 5-10(d) show, for each of the multiplicity groups, the proportions of designs that have high levels of short-distance autonomy and acquisition autonomy. The first interesting result is that the group of three-rover missions is the largest one. Only nine single-rover missions, including MSL, can fulfill the science requirement, whereas there are 188 three-rover missions that meet the same require-

ment. Moreover, for the three-rover group the proportion of designs with high (A3) versus low (A1) levels of autonomy is equally distributed, whereas most single-rover missions require high levels of autonomy for both short distance traverse and sample acquisition. Therefore, by considering multi-rover architectures, mission designers have a large pool of successful missions to chose from, most of which use current state of the art technology.

Another noteworthy result is that designs from all groups require high level autonomy for long distance traverse (this result is not illustrated in the figures). Compared to rovers equipped with A3 long-distance autonomy, rovers with less autonomy lose too much time on the very long site-to-site traverse of 2500 meters and cannot meet the science requirement. In comparison, A3 short-distance autonomy is less crucial because the typical size of a site (20 meters) is small enough to allow rovers with A1 autonomy to perform well enough. Now, it must be noted that in the case of multi-rover missions the requirement for a 2500 meters site-to-site traverse should be redefined. Indeed, the need for large site-to-site traverses is less stringent as science diversity is already provided by the fact that rovers land in different regions of the planet. Instead of a single rover sequentially exploring three sites far from one another, three rovers could explore each of the three sites simultaneously with no need for long-distance traverse. Moreover, the latter option allows the simultaneous comparison of the science results from the various sites; thus, science experiments on one site can be planned consistently with those performed on the other sites. The redefinition of the traverse requirements for multi-rover missions is part of the future work objectives.

This first analysis proves that successful single-rover missions are not only few, but they are heavy, require long operation times, and require high levels of sophistication. Now, this study focuses solely on the space segment; before drawing definite conclusions, a follow-up study would need to examine thoroughly the interactions with the ground segment, as well as assess cost and risk. Increasing the number of rovers on the surface of Mars, indeed, raises a number of issues about the management of the ground segment. The idea was implemented by the MER mission which sent two

rovers working in parallel to opposite sides of the same planet. At the same time on Earth two science teams, each dedicated to one rover, rotated in the operations room. The overall operation time is therefore maximized; almost at all times, one of the two rovers is working on Mars. A single-rover mission with current autonomy would instead be working only half of that time. Still, missions with more than two rovers functioning concurrently would present an operational challenge. Science teams cannot be replicated in large numbers because of issues of cost, management, and communication between teams. Additionally, the communication links between Mars and Earth are limited to a few hours due to Deep Space Network's restricted availability; little command and telemetry communication would consequently be left for each of the rovers. Hence, detailed models of the interactions with the ground segment may challenge the claim that multi-rover missions do not require sophisticated autonomy.

Regarding monetary cost, multi-rover missions have different effects on launch, design and construction, and operations costs. Multi-rover missions may require multiple launches and, thus, cause an increase in launch costs. Additionally, launch windows to Mars are every two years and last roughly two months; hence, all the launches must happen within the same two months which can be a big challenge if their number is more than two or three. There is a trade-off regarding the design and construction cost. Because multi-rover missions have redundant elements they are more expensive, but for the same reason a learning curve factor can be applied on each rover developed after the theoretical first unit, and thus, reduce the total production cost. There is another trade-off regarding the behavior of operations cost as the number of rovers increases. On the one hand, multi-rover missions shorten the total operation time as shown in Figure 5-10(a). On the other hand, multi-rover missions require more operations staff.

Regarding operational risk, multi-rover missions have a higher probability of success, notably for critical phases like launch and Entry Descent and Landing (EDL). There are two reasons for this. First, since from a probabilistic point of view each rover represents a trial with a certain probability of success, every extra rover decreases the

probability of total system failure (assuming there is no common cause failure). Second, for launch and EDL phases, which happen sequentially, the operations team can learn from the experience of the first trials to possibly make necessary corrections on the following spacecraft operations. As an example, the MER EDL operations team corrected the planning of Opportunity's[3] EDL maneuvers, after examination of Spirit's EDL progress.

5.5.4 Conclusions

Even if this study does not provide a definite answer about the benefits of multi-rover missions, it brings out important elements and shows that such missions are interesting solutions to decrease mass, mission lifetime, and software sophistication while returning more science than single rover missions. The idea of using replicated rovers has many supporters, such as Bod Cradlock, science advisor for the undersecretary for science at the Smithsonian Institution, who recently said [20]:

> "In fact, I would argue that we need to send even more robotic missions identical to the MER [...] NASA's intrepid use of the same basic Mariner spacecraft bus led to an enormously successful phase of exploration in the 1960s. [...] Instead of reinventing the wheel for each successive mission, which is both risky and costly, NASA needs to look for ways of utilizing the technology it has in hand."

Interestingly, the three-rover missions, which showed the most potential in this study, are very close to the MER design. One strategy may be to design a new rover for every other rover mission, with the others being multi-rover missions based upon the proven design.

[3]Spirit is the first of the two MER rovers to have landed on Mars on January 3, 2004. It was followed by Opportunity on January 25, 2004.

5.6 What are the cost-benefits of autonomy?

5.6.1 The challenge of assessing autonomy's cost-benefits

The necessity for more autonomous capabilities to equip future Mars rover missions is arguable. Higher levels of autonomy would surely benefit missions such as MSL, but the question is how well does improved autonomy compare to alternative solutions? Sections 5.3 and 5.5 demonstrate that oversizing a rover's suspension or sending teams of rovers are also interesting options that would enhance the science return of future missions. These solutions do not require rovers with more sophisticated autonomy software. In order to rigorously compare autonomy to technology alternatives, it is necessary to assess its cost-benefits. Yet, regarding these cost-benefits, while many papers praise the advantages of developing more autonomy ([39, 54, 55, 19]), very few actually deal with, or even raise the question of its cost ([50]), which would cover development, integration, and, most importantly, testing phases. This study tackles the autonomy costing problem by providing a method to quantify the maximum budget under which autonomous capabilities should be developed, in order to still be worthy in comparison to other technologies.

Because the cost model used in the current version of MSE is not reliable enough, the rover mission budgets and autonomy development costs that are given in the next paragraphs are not relevant for immediate use. This study does not claim to put a price-tag on autonomy at the end of this chapter; it provides mission designers with a method to determine an autonomy development budget. This budget is based on the designer's cost models of other subsystems, with which they are more familiar.

As mentioned in the *Autonomy* module section (Section 4.10), no methods were found that reliably model the costs involved in developing autonomy software; this lack actually reflects the difficulty of putting a price on autonomy. For that reason, in all mission-cost calculations, autonomy software is assumed to be available free of charge, whatever its degree of sophistication. This analysis assesses how much future autonomy would cost if its cost-benefit were equivalent to lower autonomy options. Then, the mission developer can evaluate whether or not it is reasonable to develop

the future level of autonomy for less than that cost.

As explained in Section 4.10, *A1* level of autonomy refers to the current state of the art in Mars rover autonomy, the autonomy used by MER, and *A3* refers to the higher level of autonomy that will be used by MSL in 2009. Furthermore, two main applications of autonomy are considered: autonomy for long-distance traverse, namely from site to site, and autonomy for short-distance traverse, within a site during sample approach. Autonomous acquisition of samples could be added as a third application field. There are two levels of autonomy and two types of applications; hence, a rover is equipped with one the following four autonomy configurations:

$$A1/A1, \quad A1/A3, \quad A3/A1, \quad A3/A3$$

where in the notation *Ai/Aj*, *Ai* refers to the long-distance autonomy level and *Aj* to the short-distance autonomy level. Within this chapter, *A3 autonomy* refers to the three autonomy configurations that have at least long or short distance autonomy at the level A3, as opposed to the A1/A1 configuration. To emphasize what is mentioned above, these four autonomy combinations are all considered to be free technologies. Following on this note, the *partial cost* of a mission is defined, for the rest of this section, as the total life cycle cost of the mission, minus the cost, $\$_{Ai/Aj}$, associated with the development and validation of its level of autonomy Ai/Aj:

$$\$^{Total}_{Ai/Aj} = \$^{Partial}_{Ai/Aj} + \$_{Ai/Aj} \tag{5.10}$$

The cost $\$_{Ai/Aj}$ is the unknown in the Equation 5.10. It is assumed to have the same value for all missions equipped with Ai/Aj levels of autonomy. The cost $\$^{Partial}_{Ai/Aj}$ is the mission cost provided by the *Cost* module. Using Equation 5.10, this analysis compares the cost-benefit of the missions modeled by MSE with that of a reference mission, for which the total cost is known, in order to put a cap on the autonomy development cost $\$_{Ai/Aj}$.

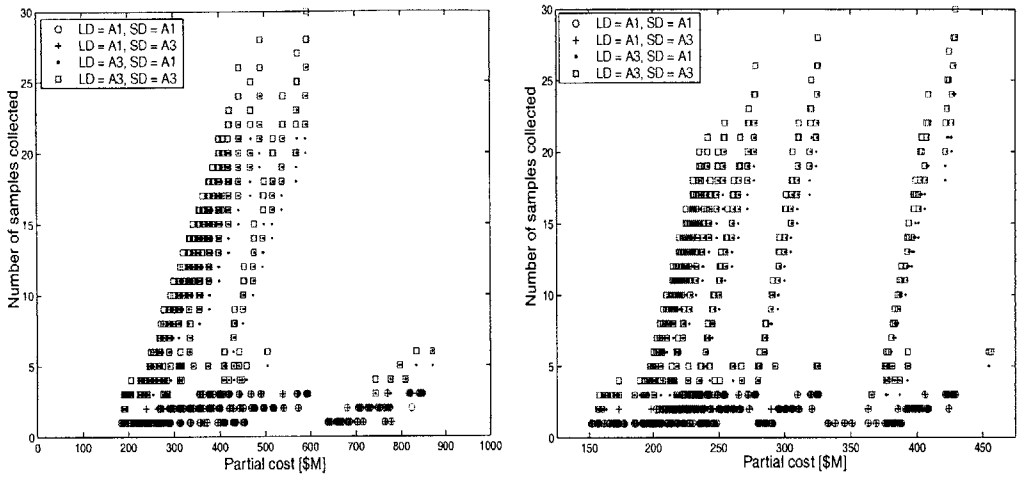


Figure 5-11: Missions' science return as a function of their partial cost

5.6.2 Autonomy costing method

The trade space introduced in Section 5.2.3 is plotted with respect to the number of samples versus partial cost axes in Figure 5-11. The design points are represented by markers that differentiate rovers according to the autonomy configuration with which they are equipped. In Figures 5-11(a) and 5-11(b), *LD* stands for long-distance autonomy and *SD* for short-distance autonomy. Figure 5-11(a) presents an overview of the trade space shape for the designs which cost less than $1 billion, and Figure 5-11(b) zooms in the region of that space where the optimal designs lie. Both views indicate that rovers with A1/A1 and A1/A3 configurations are limited to a maximum science return of three samples, which is roughly one order of magnitude less than the maximum science return of A3/A1 and A3/A3 architectures. The benefits of long-distance autonomy are undeniable for an MSL mission scenario that has a particularly long site-to-site traverse of 2500 meters. The challenge is then to determine the range of costs for which developing long-distance autonomy to a level A3 is still a worthy option.

The notion of *samples per dollar* is introduced as a metric similar to that of cost per function defined in the GINA methodology [37]. Architectural performances are compared against the same metric defined as the number of samples they return,

$R_{mission}$, divided by their total life cycle cost, $\$^{Total}_{mission}$. This metric is consistent with the idea of cost-benefits: if an architecture maximizes science return and minimizes cost, it maximizes the sample per dollar metric. Furthermore, the metric permits linking the relative benefits of two missions to their relative costs. For a rover with a science return, R_{rover}, to perform better than a reference rover, $(\$^{Total}_{ref}, R_{ref})$, its total cost, $\$^{Total}_{rover}$, must verify

$$\$^{Total}_{rover} < \frac{R_{rover}}{R_{ref}} \$^{Total}_{ref} \tag{5.11}$$

This inequality is used to set the cap on the A3 autonomy development budget by comparing missions using A3 autonomy to reference missions. These reference missions are identified by arguing that A3 autonomy is worthy of development if the following two conditions are fulfilled. First, for the same science scenario, rovers with A3 autonomy must perform better, in terms of samples per dollar, than rovers with A1/A1 autonomy. Second, rovers with A3 autonomy must perform better than rovers of former missions. In the case of MSL, the only former mission is MER, given the fact that the Mars Pathfinder mission is considered as a technology-driven rather than science-driven mission. Thus, it is worth developing a high level of long- or short-distance autonomy for a mission if its function per cost ratio, $R_{A3}/\$_{A3}$, is larger than the one for all A1/A1 missions of the trade space, as well as the one for MER:

$$\frac{R_{A3}}{\$^{Total}_{A3}} > f \times Max\left(\frac{R_{A1/A1}}{\$^{Total}_{A1/A1}}\right) \tag{5.12}$$

$$\frac{R_{A3}}{\$^{Total}_{A3}} > f \times \frac{R_{MER}}{\$^{Total}_{MER}} \tag{5.13}$$

In these two inequalities, f is a constraint factor larger than unity imposed by mission designers to quantify by how much they want new architectures to outperform reference missions. Equation 5.12 is rewritten in a more convenient form. Let us call $A1^*$ one A1/A1 design that maximizes the function per cost of all A1/A1 designs. Equation 5.12 is then equivalent to Equation 5.14. The autonomy A1/A1 is an existing technology developed and implemented on MER by the Jet Propulsion Laboratory;

thus, cost information about A1/A1 autonomy exists. Still, if the cost of implementing it on a new rover remains unknown or unavailable, Equation 5.15 is used instead of Equation 5.14. Indeed, since $\$_{A1^*}^{Total}$ is larger than $\$_{A1^*}^{Partial}$ (Equation 5.10), the inequality 5.15 implies the inequality 5.14. In other words, inequality 5.15 imposes a higher constraint on $\frac{R_{A3}}{\$_{A3}^{Total}}$.

$$\frac{R_{A3}}{\$_{A3}^{Total}} > f \times \frac{R_{A1^*}}{\$_{A1^*}^{Total}} \tag{5.14}$$

$$\frac{R_{A3}}{\$_{A3}^{Total}} > f \times \frac{R_{A1^*}}{\$_{A1^*}^{Partial}} \tag{5.15}$$

In summary, missions with A3 autonomy are worth developing if their function-per-cost ratio satisfies inequalities 5.13 and 5.15. The right hand side of these two inequalities is known. The life cycle cost of the MER mission is evaluated at $804 million, including the two launchers and the operations of each rover during a period of 90 sols [25]. The targeted science return of MER used in subsequent calculations is 14 samples for both rovers combined. The function-per-cost ratio $\frac{R_{A1^*}}{\$_{A1^*}^{Partial}}$ is provided by MSE.

For $f = 1$, the inequalities 5.13 and 5.15 are graphically represented by two straight lines that cut the samples versus cost plane into two regions (Figure 5-12). The function per cost of MER imposes a more stringent constraint than the one of A1/A1 designs. All the architectures that are below the MER constraint line have a function per cost ratio that does not satisfy the inequality 5.13. In other words, these architectures are dominated by MER. Conversely, designs that lie above this constraint line are non-dominated and perform better than MER and all A1/A1 designs. Among the designs that are non-dominated, several are A3/A1 designs (A3 long-distance autonomy and A1 short-distance autonomy) whose total costs remain to be determined.

The A3/A1 design points of Figure 5-12 are plotted on Figure 5-13, along with MER constraint lines for various values of the constraint factor f. The figure shows that no A3/A1 designs satisfy the MER inequality for a constraint factor of 3 or more ($f \geq 3$). A fraction of these architectures seems to satisfy the inequality for $f = 2$,

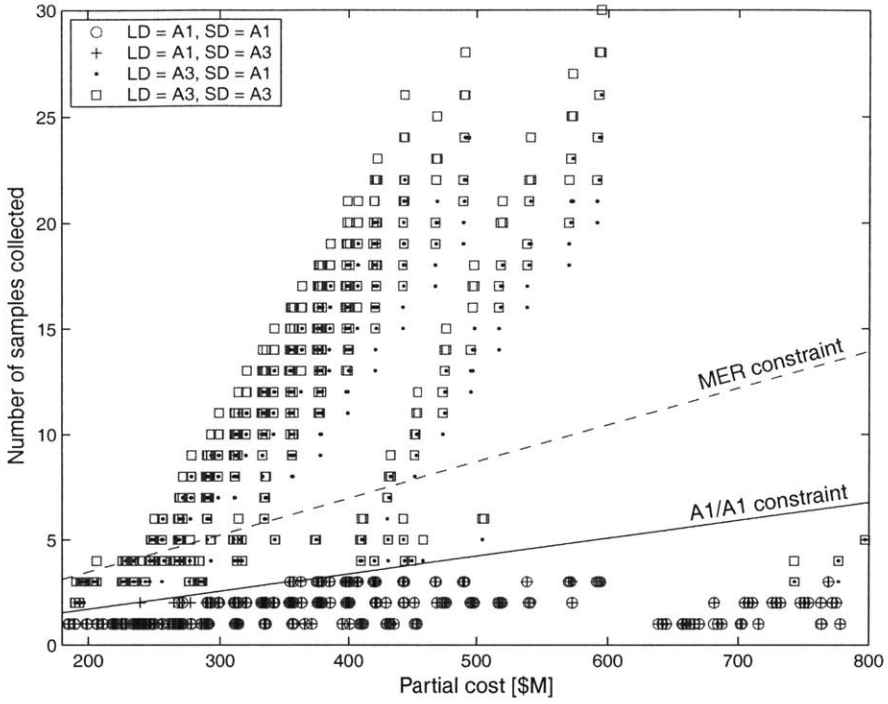


Figure 5-12: MSL-like mission's science return versus cost trade space

which means they perform twice as well as MER. Notice, though, that the plots in Figure 5-12 and 5-13 have partial costs axes whereas the constraint inequalities (Equation 5.13 and 5.15) are expressed in terms of total cost, $\$_{A3}^{Total}$.

The transformation of Figure 5-13 from a partial to a total cost scale is an horizontal shift to the right (higher costs, constant science return) of all the design points. This shift corresponds to $\$_{Ai/Aj}$, the cost of developing the autonomy configuration, Ai/Aj, relative to each architecture. Hence, a design that seems non-dominated in Figure 5-13 may actually be dominated if that shift leads the design point within the dominated region (Figure 5-14). Therefore, the horizontal distance between a design point and a constraint line is the maximum value of $\$_{Ai/Aj}$ for which the design is not dominated (Figure 5-15). As mentioned in the introduction to this section, the monetary value $\$_{Ai/Aj}^{Max}$ shown in the figure is not relevant as such. Figure 5-15 illustrates the final step of an autonomy costing method that should be applied with reliable partial-cost models contributed by the users.

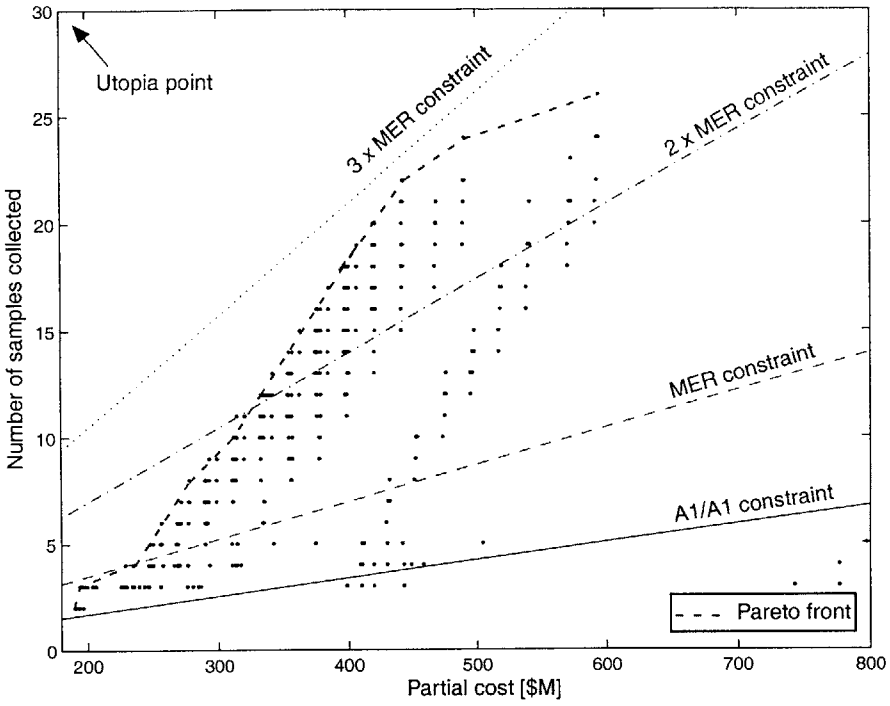
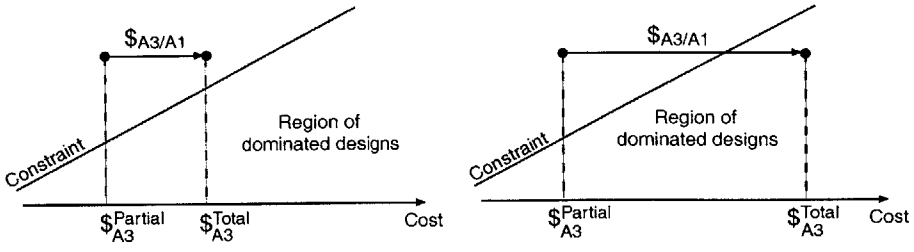


Figure 5-13: Trade space of A3/A1 designs

This graphical determination of $\$^{Max}_{Ai/Aj}$ translates into the following mathematical expression:

$$\$_{Ai/Aj} < \frac{R_{Ai/Aj}}{f \times R_{ref}} \$^{Total}_{ref} - \$^{Partial}_{Ai/Aj} \tag{5.16}$$

The right hand side of this equation represents the maximum amount of money that should be spent in developing and validating future levels of autonomy, $\$^{Max}_{Ai/Aj}$. Notice that $\$^{Max}_{Ai/Aj}$ tends to increase for missions that collect more samples (Figure 5-15); the



(a) Non-dominated design (b) Dominated design

Figure 5-14: Transition from partial cost to total cost

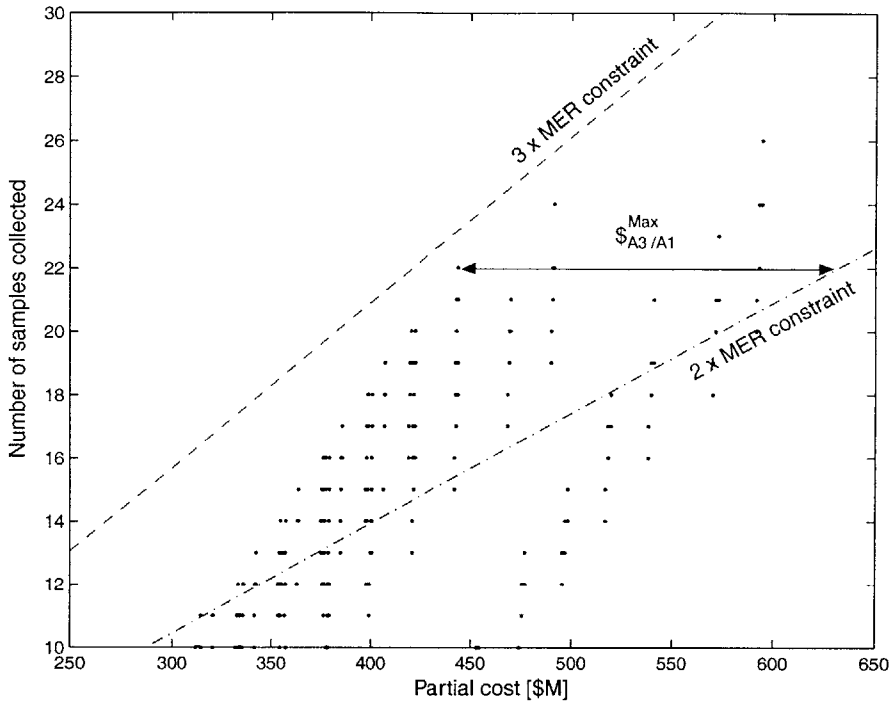


Figure 5-15: Zoom of the Figure 5-13

value of autonomy development is larger for more capable systems. Also, this model assumes that this development is amortized over one mission. Multiple missions could easily be analyzed in a similar fashion as in Section 5.5.

5.6.3 Conclusions

This study presents a method to quantify the autonomy development and validation budget by linking its cost to its benefits. This budget is calculated indirectly by assessing, first, the performance of the design, and second, its partial cost, which does not comprise any autonomy cost. These two evaluation tasks, for which engineering models exist, replace the more difficult task of assessing autonomy development cost directly.

5.7 Conclusions on MSE's analysis capabilities

This chapter focused on the description of the Analysis segment of the MSE framework. It demonstrates MSE's usefulness by the contributions MSE provides to addressing several questions relevant to, and raised by, the rover mission community. An extensive amount of information can be gleaned out of a single full-factorial trade space exploration by analyzing it from appropriate perspectives, that is choosing the right variables to represent the trade-offs. The whole diversity of analyses is possible because MSE models the variables that are relevant to many system-level design issues. Furthermore, the variety of questions tackled in this study shows the versatility with which MSE can be used. MSE's capabilities can easily be extended to include more complex systems, such as multi-rover missions.

Regarding future work on MSE, a major improvement to the framework would be the implementation of a risk model that would encompass the whole life cycle of a rover mission. In its current state MSE can already be used to perform uncertainty analyses. For example, turning parameters belonging to the science vector into design variables reveals the performance and mass uncertainties due to uncertainty in the science scenario. Figure 5-16 shows rover designs for which the science scenario is that of MER. Two dimensions of uncertainty are considered. First, the terrain roughness is allowed to vary $\pm$ 50% from its nominal value (10% terrain roughness). Second, the site-to-site distance is allowed to change $\pm$ 20% from its nominal value (70 meters). The effect of uncertainty in terrain roughness is mainly on the rover's mass (x-axis) whereas the separation between sites affects the number of samples (y-axis). As a consequence, the probabilistic regions are diamond shaped. However, some point designs do not show these regions around them which means they are robust to the changes in the science vector. These designs, numbered from 1 to 8, are actually rovers with an A3 level of autonomy for traverse whereas the science vector-sensitive rovers have only an A1 level of autonomy. In agreement with intuition, one benefit of investing in autonomy is that it frees a rover's performance from uncertainties in traverse and terrain knowledge.

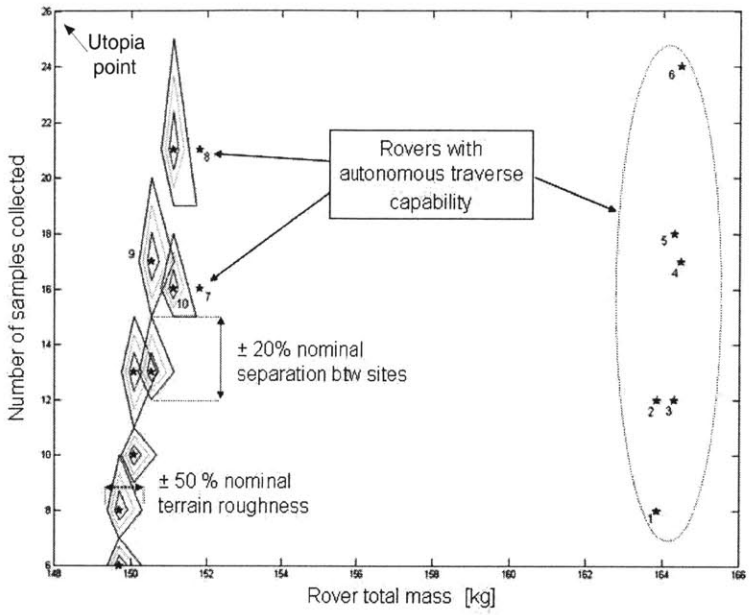


Figure 5-16: Sensitivity of MER missions to science scenario definition

Additionally, as each design has a different sensitivity, the design ranking varies with the science vector; as uncertainties grow, the relative merits between one architecture and another may become less obvious or actually switch. Such an uncertainty analysis capability will help direct program resources toward those uncertain science and engineering aspects to which mission function-per-cost is most sensitive. This is the essence of design for uncertainty.

Chapter 6

Propellant Production In Mars Orbit

6.1 Introduction and motivation

In its most recent document stating the scientific goals for the exploration of Mars, the Mars Exploration Program Analysis Group (MEPAG) identified sample return as a crucial capability of future Mars missions [34].

> "Orbital and landed packages can make many of the high priority measurements, but others absolutely require that samples be returned from Mars. There is a strong consensus on the need for sample return missions [...] study of samples collected from known locations on Mars and from sites whose geological context has been determined from remote sensing measurements has the potential to revolutionize our view of Mars."

While the first sample return mission to the Moon was achieved in 1970 by the Russian mission Luna 16, the first sample return mission to Mars is scheduled for no sooner than 2014 (it was first planned for 2003). The exact date for a sample return mission to Mars is still elusive because many challenges must be overcome for such an endeavor to succeed. One of the difficulties is to conceive the appropriate propulsion system that will propel the space probe on its return journey to the planet.

The difference between a probe that just goes to Mars and one that must come back is that the latter must be equipped with its own complete propulsion system and fuel (not just for the purpose of station keeping). Therefore, a return probe is a lot heavier and more costly to send into space on its way to Mars. As an example, out of the forecast 2700 kilograms total mass for the MSR 2003 orbiter, the share of propellant was 1400 kilograms (for the rendezvous phases with the sample canisters and subsequent departure from Mars[1]), for only one kilogram of sample brought back [47]. In this design, the orbiter used a standard chemical propulsion system. Electric propulsion is now considered as an option for the new design of the orbiter [22]. On the one hand, the chemical propulsion has the advantages of being widely used and permitting a shorter travel time (Section 6.4). Travel time is generally a main mission driver [59] and definitely a critical parameter in the context of human exploration. On the other hand, the electric propulsion is more fuel efficient (Section 6.6). The trade-off that must be solved is, therefore, in terms of payload capacity and travel time. This thesis proposes a novel propulsion method that is a compromise between the former two systems. It uses the existing idea of resource utilization, but instead of having a plant on the Martian ground, the plant is on board the orbiter.

6.1.1 In orbit resource utilization

Extensive research has been done on the subject of resource utilization on Mars. The planet's atmosphere is particularly appropriate because it is 95.3% carbon dioxide. With the addition of hydrogen, many valuable elements can be derived from carbon dioxide, such as water, oxygen, and methane. So far, the resource utilization concepts have involved chemical plants on the surface that would produce fuel for unmanned as well as manned Mars ascent vehicles, and water for human colonies. The technology (Mars In-Situ Propellant Production Precursor) was to be demonstrated on board the 2001 Surveyor Lander [49], which was eventually canceled after the failures of the Mars Climate Orbiter and Mars Polar Lander in late 1999. The idea presented

[1]The corresponding velocity impulse budgets are 550m/s for rendezvous and 2370m/s for departure [41].

in this thesis associates the techniques of in-situ propellant production with that of aerocapture and acrobraking.

Aerocapture and aerobraking Aerocapture and aerobraking are methods that take advantage of a spacecraft's drag in the atmosphere of a planet in order to slow the craft down and bring it to a closed orbit around the planet, namely an elliptic orbit.

On its first approach to Mars, an orbiter passes deep into the planet's atmosphere. This maneuver is called aerocapture because the high resistance of the atmosphere slows the spaceship from a hyperbolic orbit (open orbit) to an elliptic orbit (closed orbit); hence, on the first pass the spacecraft is *captured* by the planet. The method was to be demonstrated by the orbiter designed by the Centre National d'Etudes Spatiales (CNES) for the Mars Sample Return 03 mission, which was then canceled [41]. The orbiter was designed to reach altitudes as low as 40 kilometers on its first pass.

The subsequent passes are higher in the atmosphere, where there is less drag. They are meant to slowly decrease and circularize the orbit. Figure 6-1 illustrates the procedure for the case of the NASA Mars Odyssey orbiter. Aerobraking maneuvers

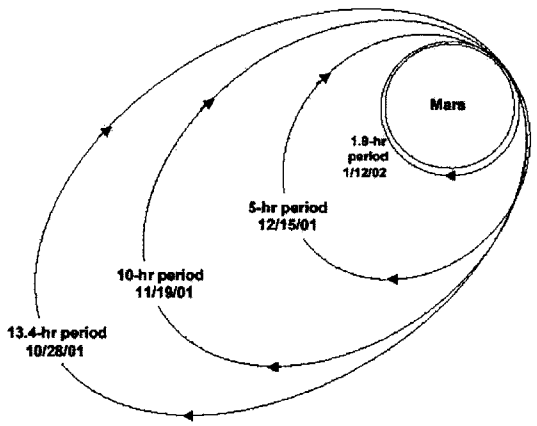


Figure 6-1: Mars Odyssey aerobraking orbits [5]

were used over a total of nine months to gradually reduce the initial apoapsis of Mars

Global Surveyor's orbit from 56,000 kilometers to 400 kilometers [45]. During these phases, the drag of the spacecraft passing through the atmosphere converts the craft's kinetic energy into heat. The idea is to utilize that heat to help the production of fuel inside a spaceship.

Propellant Production In Mars Orbit The Propellant Production In Mars Orbit (PPIMO) involves a spacecraft, propelled by a chemical propulsion system, that produces in-situ the fuel for its return. The craft collects amounts of Martian atmosphere during the phases of aerocapture and aerobraking. At the same time, it uses the heat generated by the drag to initiate chemical reactions that produce fuel from the carbon dioxide collected. In summary, the PPIMO is an in-situ propellant production plant running in orbit and using the energy coming from the braking phases of the spacecraft into the Martian atmosphere.

On its departure from Earth, a spacecraft using the PPIMO system does not need to be loaded with fuel in prevision of its return. Such a probe is, therefore, less expensive to launch. It will produce the fuel required for Mars escape once it arrives at the planet. The PPIMO system is a compromise between the standard chemical propulsion and the electric devices. It is as quick as the former and has a larger payload capacity like the latter.

6.1.2 Study goal

The purpose of the study is to determine with what efficiency a PPIMO system should be conceived in order to be competitive. The work presented in this thesis is not a feasibility study. It formulates the concept of a PPIMO system and assesses its effectiveness with respect to that of traditional chemical and electrical systems in terms of payload capacity and travel time.

6.1.3 Approach

The system considered is a spacecraft, initially in orbit around the Earth and loaded with a given payload mass, which must complete a return journey to Mars. The ship is equipped with any one of the chemical, PPIMO, and electrical propulsion systems. The preferred propulsion method is the one that minimizes the initial mass of the spacecraft and the duration of the journey for the mission scenario described the in subsequent paragraph.

The spaceship propels itself from a 700 kilometer altitude Low Earth Orbit (LEO) to a Mars transfer orbit. Once on Mars approach, the spacecraft performs orbit insertion by aerocapture. Then, successive aerobraking passes circularize its orbit on a 200 kilometer Low Mars Orbit (LMO). The travel back follows the same pattern. The spacecraft propels itself out of Mars orbit and ends its journey in direct entry to Earth's surface. The scenario does not take into account any scientific operation of the orbiter while in Mars orbit. As soon as the spacecraft has circularized around Mars at the specified altitude, it begins the escape maneuvers to travel back to Earth (assuming it has the opportunity to do so).

In the case where the spacecraft uses chemical engines (standard chemical and PPIMO propulsion devices), it is only thrusting twice during the journey: the first time to escape Earth's attraction, the second time to escape Mars' attraction. In the case where it uses electric propulsion, it is firing continuously and spirals slowly around the planets. Sections 6.4, 6.5, and 6.6 provide the models used to calculate the journey duration and the amount of propellant used for each propulsion method.

6.2 Celestial mechanics approximations

This section presents the assumptions made regarding the transfers between Earth and Mars. These assumptions apply to all three propulsion methods. Other method-specific assumptions are detailed in the subsections related to each method.

Two main assumptions are made regarding the planets' orbits around the Sun. First, the inclination angle of Mars' orbital plane with respect to the ecliptic is ne-

glected [2]. Second, Earth's and Mars' orbits are treated as circles with radii of their mean distances to the Sun (Table 6.1).

Two other approximations are made which concern the spacecraft's travel. Along its trajectory the spacecraft is under the gravitational influence mainly of three bodies, namely the Earth, the Sun, and Mars. Therefore, the spacecraft is part of a four-body interaction system. In order to simplify calculations, the transfer is divided in successive parts. For each part the spacecraft is considered to be in interaction with only one reference body: the Earth, the Sun and Mars are assumed to act on the ship one at a time. The notion of *sphere of influence* of a reference body, introduced by Pierre-Simon de Laplace, is useful for that purpose. This sphere is the region of space where a body exercises a predominant gravitational influence compared to the influence of neighboring bodies. A mass in the sphere of influence of a reference body is considered to be close enough to that body, that only this body's gravitational influence affects it [13]. For example, the Moon is in the sphere of influence of the Earth. Hence, when calculating the Moon's trajectory, the Sun's influence can be neglected in a first approximation, in comparison to that of Earth. Outside of each planet's sphere of influence masses are under the Sun's influence. The radius R_{SI} of the sphere of influence of a planet of mass M_{planet} is given below:

$$R_{SI} = \frac{M_{planet}}{M_{Sun}} \rho_{planet} \tag{6.1}$$

where M_{Sun} is the mass of the Sun and ρ_{Planet} the mean distance of the planet to the Sun [13]. The spheres of influence of Earth and Mars are given in Table 6.1.

Table 6.1: Celestial characteristics of Earth and Mars [13]

Planet	Mean distance [a.u.]	Mass ratio planet/Sun	Radius of sphere [km]
Earth	1.000000	0.000002999	923,763
Mars	1.523691	0.00000032	574,536

Hence, the problem of space travel can always be broken down into two body interaction systems. In the case of a round trip to Mars, the travel breaks down into

[2]The angle between the orbital planes of Earth and Mars is 1.85^o.

three steps for the journey to Mars and three for the journey back. The spacecraft is initially in LEO and consequently in the Earth's sphere of influence. In the case of a chemical propulsion, it leaves LEO with a velocity impulse and then follows a hyperbolic orbit which leads it outside of the sphere. In the case of electric propulsion, it is thrusting continually and it gradually raises its orbit by spiraling around the Earth. In both cases, the spaceship travels in the Earth's sphere of influence as long as its distance from the Earth is less than 923,763km (Table 6.1). Then the influence of the Sun is predominant on the spacecraft; its trajectory is calculated with reference to the Sun.

The spacecraft travels in the Sun's sphere of influence as long as its distance from Mars is more than 574,536km (Table 6.1). From there, the spacecraft is under Martian influence and it circularizes in Low Mars Orbit via aerocapture and aerobraking into the Martian atmosphere.

On the way back, the spheres of influence remain the same. The propulsion pattern used to escape Mars is the same used to escape Earth. The probe is assumed to end its travel by an Earth direct entry. In other words, the spacecraft follows a ballistic trajectory before landing on Earth. It is only slowed down by Earth's atmosphere. Finally, gravity losses are not taken into account in the chemical propulsion model.

6.3 Chemical propulsion

6.3.1 Velocity impulse derivation

This section provides a derivation for the required velocity impulses and journey durations to achieve an Earth to Mars round trip with a PPIMO or standard chemical propulsion system. The equations presented are from the Astrodynamics class [14] taught at the Massachusetts Institute of Technology by Richard H. Battin. These relations can also be found in altered forms in [13]. As stated in the above section, the path of the interplanetary spacecraft is broken down into three steps for each way. However, the spacecraft fires altogether only two times during the round trip. The

first time is to leave Low Earth Orbit (LEO) and the second is to leave Low Mars Orbit (LMO). Each firing power must be tailored so that the spacecraft can escape Earth's attraction and also rendezvous with the other planet at the exact time and location. The firing impulse is chiefly determined by the type of transfer chosen in the Sun's sphere of influence, or in other words what kind of orbit the spacecraft is following in the Sun's sphere of influence.

The greater part of the travel is, indeed, made in the Sun's sphere of influence. As the spacecraft is not firing, it is merely in free flight under the action of solar gravity. Under the assumption that Earth's and Mars' orbits are coplanar and circular, the most fuel efficient path is the Hohmann transfer [13]. It is a half ellipse, with the Sun at one focus, whose perihelion[3] is tangent to Earth orbit and aphelion tangent to Mars orbit (Figure 6-2).

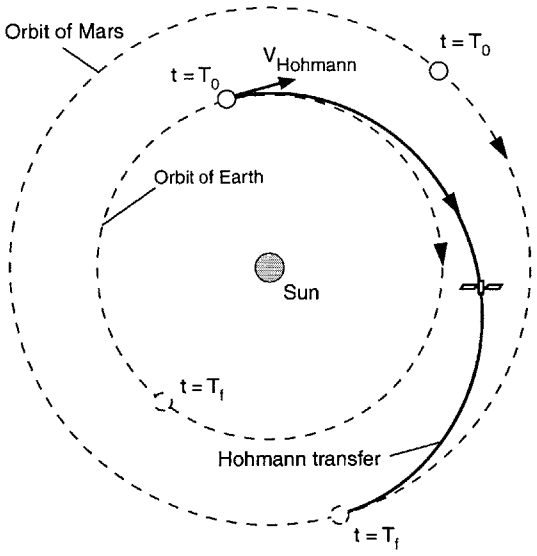


Figure 6-2: Hohmann transfer from Earth to Mars, adapted from [13]

The following section addresses the calculations of the total velocity impulse required to perform the round trip. First, it determines the impulse required to perform the transfer in the Sun's sphere of influence. This impulse is larger than what the

[3]The pericenter and apocenter of an ellipse are respectively the closest and farthest points to its focus. For an orbit around the Sun the words perihelion and aphelion are used instead.

spacecraft must actually provide. Indeed, the spacecraft benefits from its initial velocity in orbit around the planet it is departing from, and also from the velocity of this planet in rotation around the Sun.

The round trip to Mars involves two Hohmann transfers. The first one is to get to Mars and the second to come back. One Hohmann transfer is a two-impulse type transfer. To get to Mars, it first requires an impulse, $V_{Hohmann}^{Earth}$, at the perihelion to leave Earth's orbit. Once at aphelion, in Mars' vicinity, another one is needed to circularize the spacecraft in Mars' orbit. The first boost is an acceleration and is provided by the active propulsion system on board the spacecraft. The second one is actually a deceleration and is provided by aerocapture in Mars' atmosphere. Therefore, this deceleration is considered to be fuel free. The same process is used for the trip back to Earth. A departure acceleration impulse, $V_{Hohmann}^{Mars}$, is provided by an active system to leave Mars. Then the spacecraft eventually lands on Earth with a direct entry without any fuel consumption.

These impulses fired in the planets' spheres of influence must result in specific velocities in the Sun's sphere of influence so that the transfer occurs as expected. The two required velocities, in the Sun's sphere of influence, are determined by the Sun's gravitational constant, μ_{Sun}, and Earth's and Mars' mean distances to the Sun, R_{Earth} and R_{Mars}, respectively .

$$V_{Hohmann}^{Earth} = \sqrt{\frac{\mu_{Sun}}{R_{Earth}} \frac{2 \times R_{Mars}}{(R_{Earth} + R_{Mars})}} \tag{6.2}$$

$$V_{Hohmann}^{Mars} = \sqrt{\frac{\mu_{Sun}}{R_{Mars}} \frac{2 \times R_{Earth}}{(R_{Earth} + R_{Mars})}}$$

Now, because the boosts happen in the planets' spheres of influence, $V_{Hohmann}^{Planet}$ must be expressed in the planets' frames of reference. The velocity of the spacecraft in the Sun's frame of reference is the composition of two velocities (Figure 6-3). The first is the planet's rotation around the Sun. The second is the spacecraft's velocity in the planet's frames of reference. Hence, $V_{Hohmann}^{Planet}$ is the composition of the planet's rotation with the ship's velocity at the time it reaches the limit of the planet's sphere

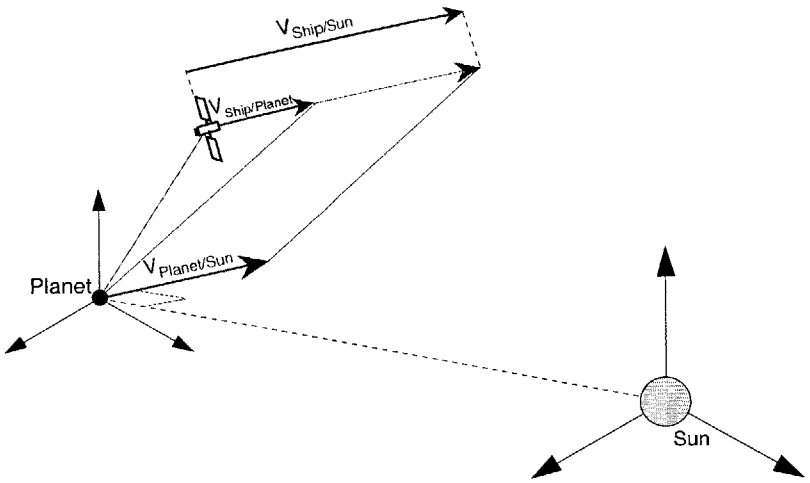


Figure 6-3: Velocity composition

of influence. This velocity is approximated as the excess hyperbolic velocity, V_∞^{Planet}.

To escape the attraction of a body, a spacecraft must follow an open orbit. There are only two types of open Keplerian orbits, the parabola and the hyperbola. A spacecraft following a parabola has a zero asymptotic velocity with respect to the reference body it departs from. One following a hyperbola has a strictly positive asymptotic velocity. This positive velocity is therefore an *excess hyperbolic velocity* compared to the parabolic case. It is assumed that the velocity of the spacecraft reaching the sphere of influence limit is very close to its asymptotic velocity, V_∞^{Planet}. Hence, the impulse provided by the spacecraft, expressed in the planet's referential, is the difference between $V_{Hohmann}^{Planet}$ and the planet's velocity around the Sun. The resulting excess hyperbolic velocities for Earth, V_∞^{Earth}, and Mars, V_∞^{Mars}, escapes are given below.

$$
\begin{aligned}
V_{Planet/Sun} &= \sqrt{\frac{\mu_{Sun}}{R_{Planet}}} \\
V_\infty^{Earth} &= V_{Hohmann}^{Earth} - \sqrt{\frac{\mu_{Sun}}{R_{Earth}}} \\
V_\infty^{Mars} &= V_{Hohmann}^{Mars} - \sqrt{\frac{\mu_{Sun}}{R_{Mars}}}
\end{aligned}
\tag{6.3}
$$

In order to follow the escaping hyperbolic orbit, the spacecraft fires an impulse V_{Esc}^{Planet} (Equation 6.4) tangent to its initial low orbit around the planet (Figure 6-4). Since the spacecraft trajectory is in the planet's sphere of influence, the gravitational

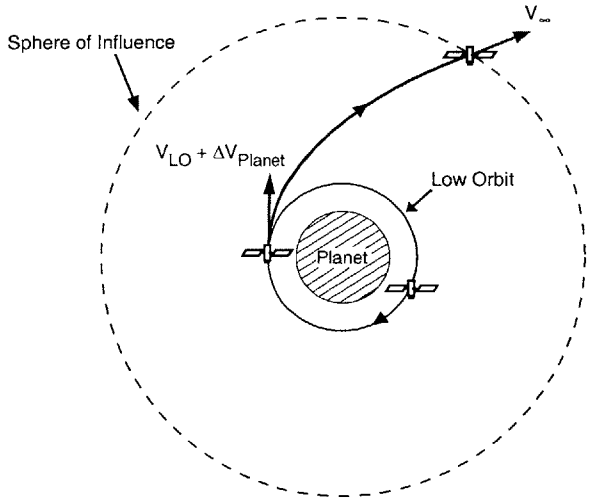


Figure 6-4: Spacecraft's trajectory in sphere of influence of a planet

constants are those of Earth and Mars.

$$\begin{aligned} V_{Esc}^{Earth} &= \sqrt{\frac{2\,\mu_{Earth}}{R_{LEO}} + V_{\infty\ Earth}^2} \\ V_{Esc}^{Mars} &= \sqrt{\frac{2\,\mu_{Mars}}{R_{LMO}} + V_{\infty\ Mars}^2} \end{aligned} \tag{6.4}$$

Because the spacecraft is already in low orbit with velocity V_{LO}, it only needs to provide the complementary part of V_{Esc}^{Planet} (Equations 6.5). This complement is the actual boost that the spacecraft fires. It is from that impulse that fuel consumption is calculated in the sections specific to the PPIMO and standard chemical systems.

$$\begin{aligned} V_{Low\ Orbit} &= \sqrt{\frac{\mu_{Planet}}{R_{Low\ Orbit}}} \\ \Delta V_{Earth} &= V_{Esc}^{Earth} - \sqrt{\frac{\mu_{Earth}}{R_{LEO}}} \\ \Delta V_{Mars} &= V_{Esc}^{Mars} - \sqrt{\frac{\mu_{Mars}}{R_{LMO}}} \end{aligned} \tag{6.5}$$

Finally, the effective velocity impulse that the spacecraft must provide for the whole round trip is the sum of the Earth and Mars impulses (Equation 6.6).

$$\triangle V_{Total} = \triangle V_{Earth} + \triangle V_{Mars} \tag{6.6}$$

The numerical values for all the velocity impulses are detailed in Table 6.2. An

Table 6.2: Velocity impulse values along the round trip

	$V_{Hohmann}$	V_{∞}	V_{Esc}	$\triangle V$	Units
Earth	32,729	2,945	11,014	3,509	[m/s]
Mars	21,480	2,649	5,553	2,102	[m/s]
Total	54,209	5,594	16,567	5,611	[m/s]

important point is that the velocity impulses do not depend on either the spacecraft's mass or the engine characteristics. They are determined purely by celestial mechanics and the transfer type selected. The calculated Mars escape impulse is 13% less than that predicted for the Mars Sample Return orbiter designed by the CNES (2370 meter per second) [41]. This discrepancy is mostly due to the approximation made in this study that Earth and Mars have circular orbits around the Sun.

6.3.2 Travel time

The time required to complete a transfer is also determined by celestial mechanics. The period of motion, $T_{complete}$, on an ellipse of semi-major axis a is obtained from Kepler's second law.

$$T_{complete} = 2\Pi \sqrt{\frac{a^3}{\mu}} \tag{6.7}$$

In this equation, μ is the gravitational parameter. This relationship needs to be adapted to the context of a spacecraft performing a round trip to Mars. On its journey to Mars, the spacecraft covers half an ellipse. Therefore, the time $T_{E \rightarrow M}$ required for

that travel is half that for covering the total ellipse given in Equation 6.7.

$$T_{E\rightarrow M} = T_{M\rightarrow E} = \Pi \sqrt{\frac{a^3}{\mu}} \tag{6.8}$$

$$T_{E\rightarrow M} = 8.6 \text{ months} \tag{6.9}$$

The same is true for the travel time, $T_{M\rightarrow E}$, of the journey back to Earth. Notice that the total travel time is not just the sum of $T_{E\rightarrow M}$ and $T_{M\rightarrow E}$. Indeed, its calculation must account for the frequency of Hohmann transfer opportunities. When the spacecraft leaves the Earth for Mars ($t = T_0$ in Figure 6-2) the relative position of the planets must be such that the spacecraft will intercept Mars right when it reaches its orbit apogee ($t = T_f$). Such a configuration of the planets happens roughly every 25 months which is a longer period than a one-way transfer between the planets. Therefore, the total travel time decomposes as follows:

$$T_{Journey} = T_{E\rightarrow M} + T_{Wait} + T_{M\rightarrow E} \tag{6.10}$$

$$T_{Wait} = 16.4 \text{ months} \tag{6.11}$$

$$T_{Journey} = 33.6 \text{ months} \tag{6.12}$$

In the case of a Hohmann transfer, the duration of the journey is constant and equal to a little less than three years. This result is consistent with the schedule of the CNES MSR orbiter, whose departure was planned for August 2005 and return to Earth for April 2008 [41].

6.4 Standard chemical propulsion

6.4.1 Initial mass

A spacecraft using a standard chemical propulsion system must carry all the fuel required for its round trip. Its initial mass on LEO, M_{LEO}^{Chem}, is that of its payload, $M_{payload}$, engines, $M_{engines}$, and fuel, M_{fuel}^{Chem}. The mass of the payload is given and

common to all propulsion systems examined. It includes the scientific payload and also the structure of the spacecraft. The mass of the engines is the same for the PPIMO and standard chemical propulsion systems. The dry mass, M_{dry}, is defined as the sum of the payload's and engines' masses. The dry mass is the mass of the spacecraft after it has escaped Mars and consumed all its fuel.

$$\begin{aligned} M_{dry} &= M_{payload} + M_{engine} \\ M_{LEO}^{Chem} &= M_{dry} + M_{fuel_{Earth}}^{Chem} + M_{fuel_{Mars}}^{Chem} \end{aligned} \tag{6.13}$$

The variables $M_{Fuel_{Earth}}^{Chem}$ and $M_{Fuel_{Mars}}^{Chem}$ are the amount of fuel needed to escape Earth and Mars, respectively. The spacecraft's initial and final masses are related by the rocket equation below.

$$M_{LEO}^{Chem} = M_{dry}\, e^{\frac{\Delta V_{Total}}{g\ Isp}} \tag{6.14}$$

The fuel consumptions for Earth's and Mars' escapes are given below along with the total fuel consumption:

$$M_{fuel_{Earth}}^{Chem} = \left(M_{dry} + M_{fuel_{Mars}}^{Chem}\right)\left(e^{\frac{\Delta V_{Earth}}{g\ Isp}} - 1\right) \tag{6.15}$$

$$M_{fuel_{Mars}}^{Chem} = M_{dry}\left(e^{\frac{\Delta V_{Mars}}{g\ Isp}} - 1\right) \tag{6.16}$$

$$M_{fuel}^{Chem} = M_{dry}\left(e^{\frac{\Delta V_{Total}}{g\ Isp}} - 1\right) \tag{6.17}$$

Hence, the initial mass of the spacecraft on LEO can be derived from given payload and engine masses. The duration of the round-trip is given by Equation 6.12.

6.5 Propellant Production In Mars Orbit (PPIMO) system

The PPIMO is an alternative to the traditional chemical propulsion. A spaceship equipped with a PPIMO system is able to produce at Mars (in-situ) the fuel it needs

for its return to Earth. The main motivation for this solution is that the amount of fuel required to send the spacecraft to Mars is far less than that with a traditional chemical system. Indeed, for a spacecraft able to produce its return fuel in-situ, $M_{fuel_{Mars}}$ is equal to zero in Equation 6.15 and thus $M_{fuel_{Earth}}^{PPIMO}$ is less than $M_{fuel_{Earth}}^{Chem}$.

Extensive research has been done on the subject of resource utilization on Mars but in the context of a human exploration of its surface. Therefore, all the work has focused on production plants located on the surface of Mars. In this present study the plant is on board an orbiter. Initially, the spacecraft is loaded with only a portion of the fuel required for it to come back. The amount of complementary fuel produced in-situ depends on the orbiter's mass and on the efficiency of the PPIMO system. The fuel production procedure is the following: during the phases of aerocapture and aerobraking the spacecraft collects a certain quantity of Martian atmosphere from which fuel is generated via several chemical reactions. Additionally, the heat generated by the capture and braking of the craft is used to run these chemical reactions. The PPIMO system benefits from the high power of the aerocapture, whereas propellant production plants situated on ground suffer from low available power (unless they use nuclear sources). Finally, the vehicle uses the fuel produced by the PPIMO system to propel itself on a return journey to Earth.

This section first addresses the thermodynamic aspect of the fuel production and then presents the rationale for the computation of the initial mass of a spacecraft that would use a PPIMO system.

6.5.1 In situ fuel production

The subsequent chemical process produces oxygen that can used by a bi-propellant rocket with hydrogen brought from Earth. The Reactions 6.18 and 6.19 present a two-step production of oxygen from carbon dioxide. First, a Sabatier reaction produces water from carbon dioxide and hydrogen reactants. Then, the water is electrolyzed

to produce hydrogen and oxygen.

$$CO_2 + 4\ H_2 \rightarrow CH_4 + 2\ H_2O + 165\ \text{kJ/mol} \tag{6.18}$$

$$483.6\ \text{kJ/mol} + 2\ H_2O \rightarrow 2\ H_2 + O_2 \tag{6.19}$$

There are two properties that must be considered to quantitatively assess the feasibility of this process. The first is the amount of heat that must produced in the reactors for the reactions to occur. The Sabatier reaction (Reaction 6.18) is exothermic, in other words, it produces thermal energy. However, the second reaction is endothermic and requires more heat than is initially produced by the Sabatier process. Therefore, the overall fuel production process is endothermic. It requires 159.3 kilo Joule per mole of water produced, assuming all the heat generated by the Sabatier reaction is used for the electrolysis. The second property is the temperature of the reactors. On the one hand according to Le Châtelier's rule[4], the Sabatier process gives a larger amount of products for low temperatures because it is exothermic. On the other hand, from a production rate point of view, the Sabatier reaction is faster for higher temperatures. Thus, a temperature exists, still to be determined, that maximizes the productivity of the Sabatier process. The details of how the reactors would be designed and implemented into the spacecraft have not yet been addressed. The next section describes the advantages of using the heat generated by the aerocapture and aerobreaking phases for running the fuel production process.

6.5.2 Energy transfer

The phases of aerocapture and aerobraking slow the spacecraft from a hyperbolic orbit to a series of elliptic orbits that lead to final circular orbit around Mars. Each time the spacecraft traverses the Martian atmosphere, it encounters resistance. The friction of the spacecraft's outer structure with the atmosphere converts the kinetic energy of the vehicle into heat. This energy transfer is localized on the vehicle's

[4]In simple terms Le Châtelier's rule is a principle of equilibrium. For an exothermic reaction, if the temperature of the reactor is lowered, the reaction will generate more products and heat in order to raise the temperature.

surface. The innovative idea of the PPIMO system is to direct the heat flow into reactors inside the spacecraft where the endothermic chemical process(Reactions 6.18 and 6.19) takes place. The concept is similar to that used on the Toyota Prius, which converts the energy released from braking the car into electric energy.

The spacecraft arrives at Mars with the hyperbolic velocity V_{∞}^{Mars} and circularizes around Mars at an assumed altitude of 200 kilometers. The variation of kinetic energy per unit mass of the spacecraft, ΔE_K, is given by the equation below.

$$\Delta E_K = \frac{\Delta E_{Kinetic}}{M_{Arrival}^{Mars}} = \frac{1}{2}\left[\left(V_{Esc}^{Mars}\right)^2 - \frac{\mu_{Mars}}{R_{LMO}}\right] \tag{6.20}$$

$$\Delta E_K = 9.46 \text{ MJ/kg}_{\text{s/c}} \tag{6.21}$$

This equation assumes that the spacecraft's mass is constant during the braking phases and equal to its mass when it approaches Mars, $M_{Arrival}^{Mars}$. In fact, the mass of the spacecraft should increase during these phases by the amount of atmosphere captured (propellant produced). Additionally, as a first order approximation, this model does not account for the energy spent in accelerating that amount of atmosphere collected by the spacecraft.

According to Reactions 6.18 and 6.19, the chemical process requires the following energy for each kilogram of fuel produced:

$$\Delta E_{Chem} = 8.85 \text{ MJ/kg}_{\text{fuel}} \tag{6.22}$$

The transformation factor, β, is then defined as the ratio of the mass of the fuel produced in-situ, M_{ISPP}, to that of the spacecraft on its arrival at Mars.

$$\beta = \frac{M_{ISPP}}{M_{s/c}} = \frac{\Delta E_K}{\Delta E_{Chem}}\,\eta \tag{6.23}$$

$$\beta = 1.07\,\eta \tag{6.24}$$

In these equations, η is the general efficiency of the transfer of kinetic energy to thermal energy in the reactors. Section 6.8 determines the efficiency and mass the PPIMO system must have in order to be a preferable solution to the traditional

chemical and electrical propulsion.

6.5.3 Mass calculations

Reacting hydrogen In order for the Sabatier reaction to take place, hydrogen must be brought from Earth to react with the carbon dioxide collected from the Martian atmosphere (Reaction 6.18). This initial amount of hydrogen is twice that present in the produced fuel; the rest of it is used to make methane.

$$n_{H_2}^{Earth} = 2\ n_{ISPP} \tag{6.25}$$

$$M_{H_2}^{Earth} = \frac{2}{9} M_{ISPP} \tag{6.26}$$

Equation 6.25 relates the number of moles of hydrogen brought from Earth to that of the fuel produced. Equation 6.26 provides the same relationship but in terms of mass.

Initial mass on LEO Depending on the production efficiency of the PPIMO system, the spacecraft may initially have to carry some fuel to complement that produced in-situ in prevision of Mars escape. This section derives the initial mass of a PPIMO spacecraft in LEO, along with the minimum PPIMO efficiency for which the spacecraft produces all its return fuel in-situ. It is assumed that the dry mass of a PPIMO spacecraft is the same as a spacecraft using standard chemical propulsion. In other words, the mass the PPIMO hardware is not yet taken into account in the spacecraft's total mass. Section 6.8 presents a method to determine the maximum mass that should be allocated to the PPIMO system for it to be perform better than chemical and electric propulsion engines.

The expression of the initial mass of a PPIMO spacecraft on LEO is given below. This mass takes into account the amount of hydrogen required for the initiation of the Sabatier reaction in-situ. The formula is the rocket equation expressed for the

Earth impulse.

$$M_{LEO}^{PPIMO} = \left(M_{Dry} + \alpha\, M_{fuel_{Mars}}^{Chem} + M_{H_2}^{Earth}\right)\, e^{\frac{\Delta V_{Earth}}{g\ Isp}} \tag{6.27}$$

$$\alpha = \frac{M_{fuel_{Mars}}^{Chem} - M_{ISPP}}{M_{fuel_{Mars}}^{Chem}} \tag{6.28}$$

In this equation, $\alpha\, M_{fuel_{Mars}}^{Chem}$ is the amount of fuel loaded on the spacecraft in prevision of the return journey. This amount is a fraction of what would be required by a traditional chemical system, M_{fuel}^{Mars}. It is assumed that the propellant plant does not produce more propellant than required for the return journey. Therefore, α is a positive number. The amount of fuel produced in-situ, M_{ISPP}, is derived from Equation 6.23.

$$\beta = \frac{M_{ISPP}}{\left(M_{Dry} + \alpha\, M_{fuel_{Mars}}^{Chem} + M_{H_2}^{Earth}\right)} \tag{6.29}$$

$$M_{ISPP} = \frac{\beta}{1 + \frac{7}{9}\beta}\left(M_{Dry} + M_{Fuel_{Mars}}^{Chem}\right) \tag{6.30}$$

The combination of this last result with Equations 6.16 and 6.27 yields another expression of the PPIMO system's mass on LEO:

$$M_{LEO}^{PPIMO} = \frac{9}{9 + 7\,\beta}\, M_{Dry}\, e^{\frac{\Delta V_{Total}}{g\ Isp}} \tag{6.31}$$

$$M_{LEO}^{PPIMO} = \frac{9}{9 + 7\,\beta}\, M_{LEO}^{Chem} \tag{6.32}$$

As expected, the initial mass ratio of the PPIMO system to the standard solution is always less than unity.

Because a PPIMO spacecraft does not produce more fuel than needed for it to escape Mars, the possible values of the transformation factor, β, have an upper bound.

$$\beta \leq \frac{9\left(e^{\frac{\Delta V_{Mars}}{g\ Isp}} - 1\right)}{7 + 2\, e^{\frac{\Delta V_{Mars}}{g\ Isp}}} \tag{6.33}$$

$$\beta \leq 0.54 \tag{6.34}$$

It must be emphasized that this inequality is not due to a physical limitation. It is a limitation related to the use of the PPIMO system. The spacecraft does not need more than 54% of its mass in fuel to be produced in-situ. This value would change for a different mission scenario, such as a Mars sample return which would require the orbiter to produce additional propellant for a Mars ascent vehicle.

The combination of Equations 6.24 and 6.34 sets the following upper bound on the PPIMO system's efficiency:

$$\eta \leq 0.51 \tag{6.35}$$

Because the spacecraft only requires at most 54% of its mass in fuel, the efficiency of the system does not need to be more than 51%. For this efficiency, the initial mass of the PPIMO spacecraft is 30% less than that of the standard one, as illustrated in Figure 6-5.

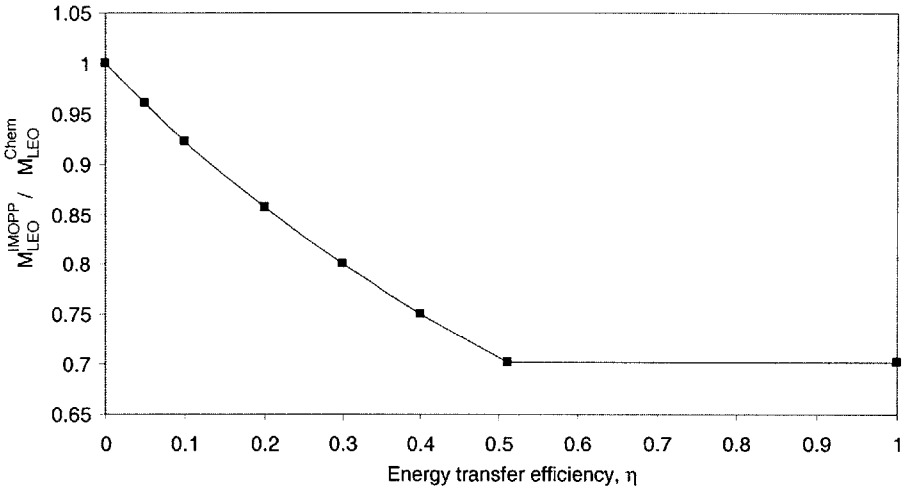


Figure 6-5: Ratio of the initial masses of standard chemical and PPIMO systems as a function of efficiency

6.6 Electric propulsion fundamentals

Electric propulsion is a low-thrust but high-specific impulse type of propulsion [53]. It can be used for several broad applications like north-south station keeping, orbit

raising and interplanetary travel. However, electric thrusters provide insufficient acceleration to put payloads in orbit. Spacecraft using electric propulsion need to be brought to reduced-gravity or gravity-free space by another propulsion means. This study focuses mainly on two types of electric devices, the ion engine and the Hall effect thruster.

These two types of thrusters are more fuel efficient than chemical systems. In the context of space applications, fuel efficiency of an engine is measured in terms of *specific impulse*, Isp. The Isp is the total impulse per weight of propellant [53]. It is regarded as an important figure of merit for a spacecraft, similar in concept to that of the kilometers-per-liter parameter used for cars. For constant thrust, $T_{s/c}$, and mass flow, $\dot{M}_{s/c}$, the Isp is given by Equation 6.36.

$$Isp = \frac{T_{s/c}}{g\ \dot{M}_{s/c}} \tag{6.36}$$

Regarding spaceflight, electric propulsion is not yet as mature as chemical propulsion. Up to now, it has been used only twice, on missions dedicated to test high-risk technologies. NASA's 1998 Deep Space 1 spacecraft successfully used an ion engine to encounter asteroids and comets. ESA's SMART-1 mission (Small Missions for Advanced Research in Technology) is the second example of electric propulsion used for space exploration. A Hall effect thruster propels the SMART-1 spacecraft on a 16 month transfer from Low Earth Orbit (LEO) to the Moon. In comparison NASA's 1968 Apollo 8 mission, which used chemical propulsion, reached lunar orbit in 3 days. This last application points out a genuinely important feature of electric propulsion: its good fuel efficiency is traded for very long travel times. NASA's 2006 Dawn and ESA's 2011 BepiColumbo missions are the future applications of electrical propulsion to space exploration.

6.6.1 Astrodynamics

Unlike a chemical device, an electric engine fires over long sequences; as a consequence, a spacecraft equipped with electric propulsion does not follow a Keplerian orbit. In

the context of this study, the spacecraft is assumed to be thrusting continuously during its transfer between planets, and always in the direction perpendicular to its position vector. Furthermore, the thrust of the spacecraft, $T_{s/c}$, is assumed be constant throughout the round trip (Equation 6.37). The thrust is expressed as a function of the engine's specific impulse, Isp, the electric plant's power and efficiency, P_{Plant} and η_{Plant}, and Earth's gravitational acceleration, g :

$$\begin{aligned} \overrightarrow{T_{s/c}} \cdot \overrightarrow{r} &= 0 \\ T_{s/c} &= \frac{2\ \eta_{Plant}\ P_{Plant}}{g\ Isp} \end{aligned} \tag{6.37}$$

The energy of the thruster is converted into potential energy. For an escape maneuver, the spacecraft slowly raises its orbit by spiraling around the reference body (Figure 6-6). The process involves many revolutions around that body. The spacecraft escapes

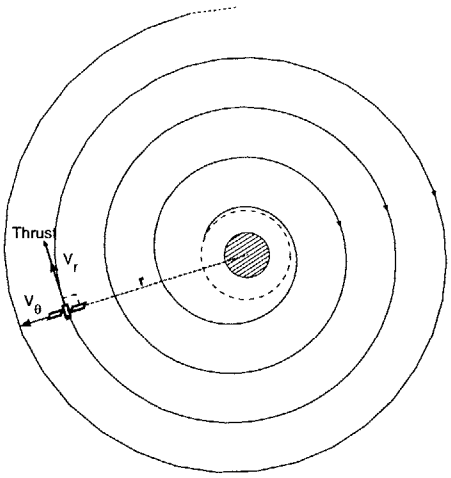


Figure 6-6: Orbit raising around the Earth with electric propulsion

a planet's attraction by spiraling until it reaches the planet's sphere of influence. Then, it spirals around the Sun until it arrives at the other planet's orbit.

6.6.2 Numerical Model

Contrary to the chemical propulsion case (Section 6.3), there is no analytical expression to estimate the fuel consumption of a spacecraft using electric propulsion. In

this situation, one recourse is to utilize time discretization methods that solve the motion's differential equations time step by time step. Furthermore, in the chemical propulsion case, the initial mass of the spaceship is derived from a given final mass (referred to as *dry mass* in Section 6.3). For the electric propulsion case, using numerical methods forces the procedure to be reversed. The model starts with a given initial mass. At each time step, amounts of mass are withdrawn that correspond to the spacecraft's fuel consumption for that step; and so on until the transfer is finished.

Fuel mass and travel time are the only results needed for the purpose of assessing electric propulsion's efficiency. Still, the determination of these two properties requires side variables to be calculated, such as velocity and position. Position indicates when the spacecraft reaches its destination. Velocity determines the trajectory followed by the vehicle to which travel duration and fuel consumption are directly related. The following numerical model solves a problem with five unknowns: mass $M_{s/c}$, time t, radial velocity V_r, orthoradial velocity V_θ and position r.

In the electric propulsion case, the probe is firing in four of the six travel phases. Two of these phases are orbit raising procedures around Earth and Mars. The two other phases involve orbit raising around the Sun during the journey to Mars, and orbit lowering during the journey back to Earth.

Body of the Model (Illustrated in Figure 6-7)

The following equations apply to each of the three spheres of influence by using the appropriate gravitational constant. The numerical model uses four main equations in order to solve for the four unknowns of mass $M_{s/c}$, radial velocity V_r, orthoradial velocity V_θ and radial distance r. The aim is actually to develop mathematical expressions for all the derivatives $\dot{M}_{s/c}$, $\dot{V}_r$, $\dot{V}_\theta$ and $\dot{r}$. Indeed, given a time step Δt and the value of property P at time t, it is then possible to find the property's new value at time $t + \Delta t$ as shown in Equation 6.38 (the time derivative of the variable x is noted $\dot{x}$).

$$P(t + \Delta t) = P(t) + \dot{P}\ \Delta t \tag{6.38}$$

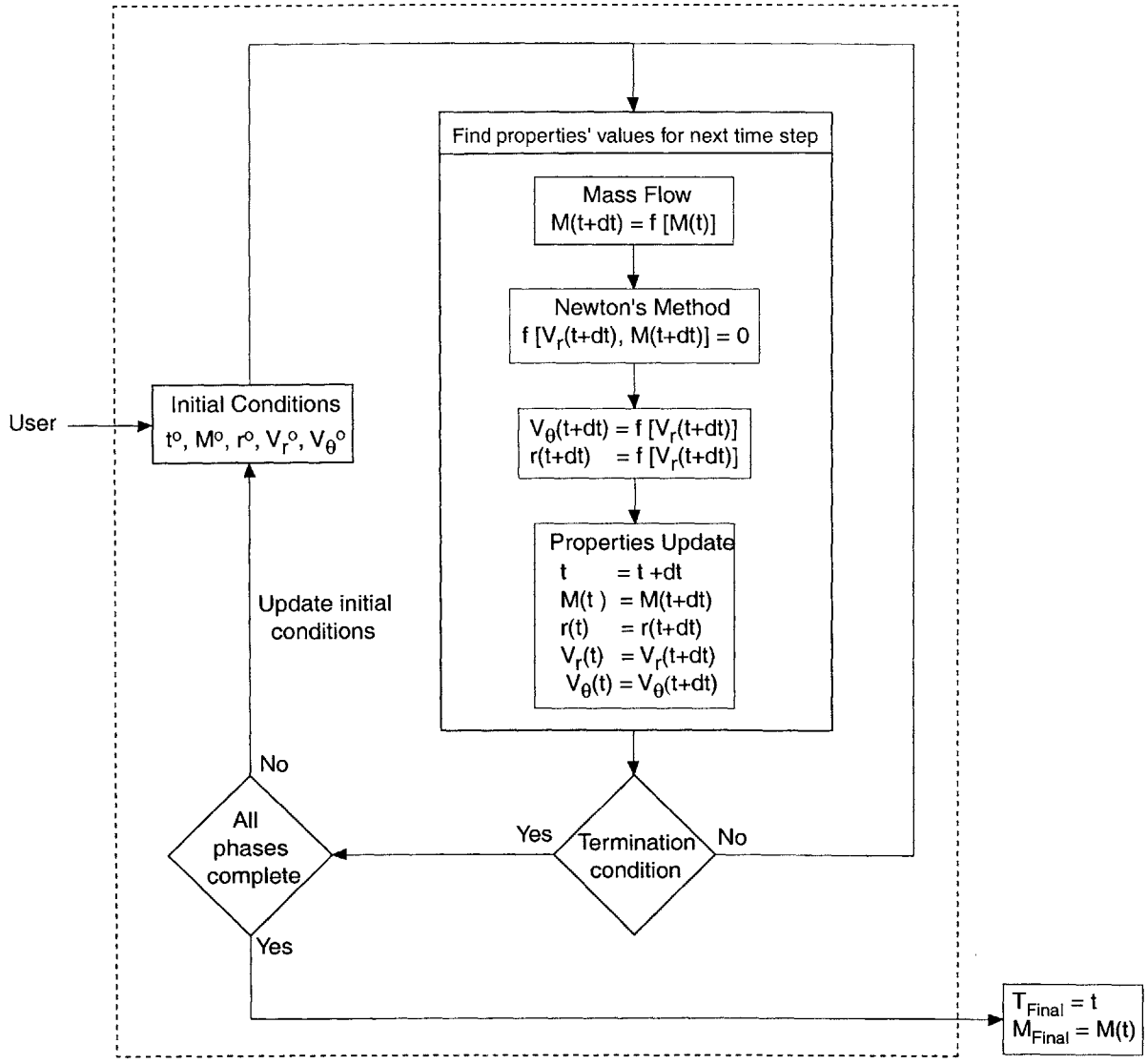


Figure 6-7: Numerical model for the electric propulsion

Therefore, with given initial values it is possible to find, step by step, states' values for the whole round trip. In order for the iterations to converge, it is necessary to use a backward Euler method. The difference between forward and backward Euler methods is represented in Equations 6.39 and 6.40, respectively.

$$\dot{P}(t) = \frac{P(t+\Delta t) - P(t)}{\Delta t} \tag{6.39}$$

$$\dot{P}(t+\Delta t) = \frac{P(t+\Delta t) - P(t)}{\Delta t} \tag{6.40}$$

The forward Euler method is not apppropriate because, unlike the backward Euler one, it can be divergent for large discretization steps. However, the backward Euler has a more complicated formulation since the derivative of the state, $\dot{P}$, must be expressed at time $t + \Delta t$.

Mass flow The first equation of the numerical model relates the fuel mass flow, $\dot{M}_{s/c}$, to the engine's specific impulse, Isp, and thrust, $T_{s/c}$. It is the same relationship as Equation 6.36.

$$\dot{M}_{s/c} = -\frac{T_{s/c}}{g\, Isp} \tag{6.41}$$

Energy The following set of equations are energy relations. Equation 6.42 states that the spacecraft's total mechanical energy, $E_{s/c}$, is the sum of its specific kinetic energy $\frac{1}{2}\, V^2$ and potential energy $-\frac{\mu}{r}$. The following two equations (Equations 6.43 and 6.44) express the spaceship's mechanical power, $P_{s/c}$, first via the thrust and velocity product $T_{s/c} \times V_\theta$, and second via the energy derivation $\dot{E}_{s/c}$.

$$\frac{1}{2}\, V^2 - \frac{\mu}{r} = E_{s/c} \tag{6.42}$$

$$P_{s/c} = T_{s/c} \times V_\theta \tag{6.43}$$

$$P_{s/c} = \dot{E}_{s/c} \tag{6.44}$$

This set of equations provides a relation between the derivatives of the radial velocity, $\dot{V}_r$, and orthoradial velocity, V_θ. Indeed, the equality between the derivative of the

left term of Equation 6.42 and the right term of Equation 6.43 yields:

$$V_r \dot{V}_r = \frac{T_{s/c}}{M_{s/c}} V_\theta - \frac{\mu}{r^2} \dot{r} - V_\theta \times \dot{V}_\theta \tag{6.45}$$

Angular Momentum The final set of equations expresses the ship's angular momentum derivative, $\dot{h}$, via the cross-product of position and velocity (Equation 6.46), and via that of position and force (Equation 6.47).

$$\dot{h} = r \times \dot{V}_\theta + \dot{r} \times V_\theta \tag{6.46}$$

$$\dot{h} = \frac{T_{s/c}}{M_{s/c}} r \tag{6.47}$$

Equations 6.46 and 6.47 are combined to give Equation 6.48. This expression of V_θ is then be inserted back into Equation 6.45 to provide an expression of V_r without any derivatives on the right hand side (Equation 6.49). The last derivative to be expressed is the one of the position vector r; it is by definition the radial velocity, V_r (Equation 6.50).

$$\dot{V}_\theta = \left[\frac{T_{s/c}}{M_{s/c}} - \frac{V_r \times V_\theta}{r}\right]_{t+\Delta t} \tag{6.48}$$

$$\dot{V}_r = \left[\frac{V_\theta^2}{r} - \frac{\mu}{r^2}\right]_{t+\Delta t} \tag{6.49}$$

$$\dot{r} = V_r(t + \Delta t) \tag{6.50}$$

The expressions for the derivatives of $M_{s/c}$, V_θ, V_r and r are determined in Equations 6.41, 6.48, 6.49, and 6.50, respectively. The subsequent task is to determine $M_{s/c}(t + \Delta t)$, $V_\theta(t + \Delta t)$, $V_r(t + \Delta t)$ and $r(t + \Delta t)$ from the derivatives' expressions and Equation 6.39.

Equation 6.41 can be used directly to update $M_{s/c}(t + \Delta t)$ as $\dot{M}_{s/c}$ depends only on the engine's characteristics. The remaining three equations (6.48, 6.49, and 6.50) are interrelated. Combining each of these equations with the derivative expressions

of Equation 6.39 provides the following set of relations.

$$V_\theta(t+\Delta t) = \left[\frac{T_{s/c}}{M_{s/c}} - \frac{V_r \times V_\theta}{r}\right]_{t+\Delta t} \times \Delta t + V_\theta(t) \quad (6.51)$$

$$V_r(t+\Delta t) = \left[\frac{V_\theta^2}{r} - \frac{\mu}{r^2}\right]_{t+\Delta t} \times \Delta t + V_r(t) \quad (6.52)$$

$$r(t+\Delta t) = V_r(t+\Delta t) \times \Delta t + r(t) \quad (6.53)$$

The position vector at time $t+\Delta t$, $r(t+\Delta t)$, is expressed in terms of the radial velocity,$V_r(t+\Delta t)$, in Equation 6.53. The insertion of this equation in Equation 6.51 yields an expression of $V_\theta(t+\Delta t)$ in terms of $V_r(t+\Delta t)$ (Equation 6.54).

$$V_\theta(t+\Delta t) = \left[\frac{T_{s/c}}{M_{s/c}(t+\Delta t)} + V_\theta(t)\right] \frac{V_r(t+\Delta t)\ \Delta t + r(t)}{2\ V_r(t+\Delta t)\ \Delta t + r(t)} \quad (6.54)$$

Finally, Equations 6.53 and 6.54, when combined with Equation 6.52, provide Equation 6.55 to be solved for $V_r(t+dt)$.

$$V_r(t+\Delta t) - V_r(t) - \frac{\mu\ \Delta t}{(V_r(t+\Delta t)\ \Delta t + r(t))^2} - \left[\frac{\left(\frac{T_{s/c}}{M_{s/c}(t+\Delta t)} + V_\theta(t)\right)^2 (V_r(t+\Delta t)\ \Delta t + r(t))}{2\ V_r(t+\Delta t)\ \Delta t + r(t)}\right] \Delta t = 0 \quad (6.55)$$

All the variables of Equation 6.55 are known but $V_r(t+dt)$. The other variables are either constant or they have been calculated at the former time step (time t). The equation is non-linear for the variable $V_r(t+dt)$ and is solved by using Newton's method. This method is an iterative procedure to solve equations of type $f(x^*) = 0$. The algorithm is detailed below:

while $\|x^{k+1} - x^k\|$ and $\|f(x^{k+1}\|$ are not small enough

do $x^k = x^{k+1}$

and $x^{k+1} = x^k - \left[\frac{df}{dx}(x^k)\right]^{-1} f(x^k)$

end

$x^* = x^{k+1}$

Note that the method requires $\frac{df}{dx}(x^k)$ to be non-zero. The value of $V_r(t+\Delta t)$ resulting from this solution method is an approximate value. Still, satisfactory precision is accessible relatively quickly since the method converges quadratically. Once $V_r(t+\Delta t)$ is found, $r(t+\Delta t)$ and $V_\theta(t+\Delta t)$ follow from Equations 6.53 and 6.54, respectively.

6.7 Propulsion hardware

6.7.1 Chemical thruster

The engine used in this study is similar to the third-stage booster (H-10) of the former Ariane 4 developed by Arianespace. It has a mass of 1300 kilograms and a specific impulse of 444 seconds [56]. It burns liquid oxygen and liquid hydrogen, and thus, is appropriate for PPIMO applications.

6.7.2 Electric thrusters

Two types of devices are considered: ion engines and Hall-effect thrusters. The ion engine chosen for this study has the characteristics of the 1998 NSTAR Deep Space1 ion engine, which remains to date the only such engine used for the exploration of the solar system. The Hall-effect thruster's characteristics are that of the SPT engine as presented in Table 19-7 of [53]. All these engines' characteristics are gathered in Table 6.3

Table 6.3: Performance characteristics for the chemical, ion and Hall-effect engines

Engines	Type	Isp [s]	Power [W]	Mass	Efficiency	# units	Sources
H10	Chemical	444	N/A	1300 kg	N/A	1	[56]
NSTAR	ion engine	3100	2300	45 W/kg	0.6	5	[53, 59]
SPT	Hall-effect	1600	1500	150 W/kg	0.48	7	[53]

Additionally, the mass of the electric propulsion system accounts for the mass of the solar panels, which collected power is needed to thrust. The mass of the solar

arrays is calculated based on the power requirements from the engines and by using a specific power of 100 watts per kilogram.

6.8 Comparison of the propulsion systems

The initial mass estimate performed in Section 6.5 does not account for any mass of the propellant production hardware. This system includes the atmosphere collector, reactors, pipes, radiators and tanks. The purpose of this section is to determine the mass of the whole system and its efficiency in order for it to be preferable to traditional chemical and electric systems. PPIMO propulsion is first compared to standard propulsion and then to electric propulsion.

6.8.1 PPIMO versus standard chemical propulsion

Initial mass metric Equation 6.32 shows that the initial mass ratio of a spacecraft using standard chemical propulsion over one using a PPIMO system is constant for a given system's efficiency. The difference between the two spacecrafts' masses is:

$$M_{LEO}^{Chem} - M_{LEO}^{PPIMO} = \left(\frac{7\frac{\Delta E_K}{\Delta E_{Chem}}\,\eta}{9 + 7\frac{\Delta E_K}{\Delta E_{Chem}}\,\eta} \right) e^{\frac{\Delta V_{Total}}{g\ Isp}} \left(M_{payload} + M_{engine}\right) \qquad (6.56)$$

The mass difference is an affine function[5] of the payload mass variable. The function is represented in Figure 6-8 for various efficiencies. Notice that the mass savings created by the PPIMO solution are larger for missions with more payload. For a given PPIMO system efficiency, the difference in mass of the two spacecraft increases with the payload mass. The rate of increase, a, is derived from Equation 6.56:

$$a(\eta) = \left(\frac{7\frac{\Delta E_K}{\Delta E_{Chem}}\,\eta}{9 + 7\frac{\Delta E_K}{\Delta E_{Chem}}\,\eta} \right) e^{\frac{\Delta V_{Total}}{g\ Isp}} \qquad (6.57)$$

$$a_{max} = 1.08 \qquad (6.58)$$

[5]Affine functions are of the type $f : x \mapsto a\ x + b$. Linear functions are a particular case of affine functions for which $b = 0$.

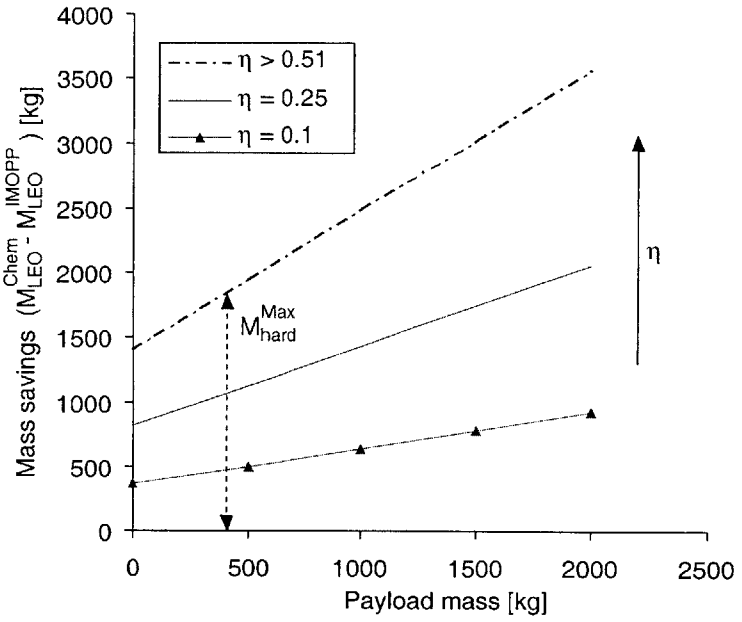


Figure 6-8: Difference of the initial masses of a spacecraft using standard propulsion and one using PPIMO propulsion as a function of the payload mass

When the effective efficiency is maximum ($\eta \geq 0.51$), the rate of increase is the largest and equals 1.08. In other words, the PPIMO device permits saving an amount of mass equivalent to a little more than the payload mass. According to [41], a typical payload mass for a Mars sample return mission is 400 kilograms. For that payload mass, a PPIMO system with an efficiency of 0.51 (maximum effective efficiency) permits savings of 1,834 kilograms, as illustrated in Firgure 6-8 (M_{hard}^{Max}). This mass savings is optimistic since it should be diminished by an amount equal to the mass of the PPIMO hardware. Yet, this mass savings is valuable information which asserts that, within the context of this sample return mission, the fuel production hardware must weigh less than 1,834 kilograms (M_{hard}^{Max}) for the PPIMO system to be favored over standard propulsion systems. More generally, for a system with a given performance and payload mass, $M_{LEO}^{std} - M_{LEO}^{PPIMO}$ is the maximum mass under which designers must conceive the fuel-production plant.

In the case where the efficiency is considered as a free parameter, there is an interesting trade-off between the fuel-generation plant's mass and its efficiency. Figure 6-9

shows the curves of iso-payload-mass (lines of constant payload mass) plotted against efficiency and PPIMO hardware mass. The iso-payload lines are represented for pay-

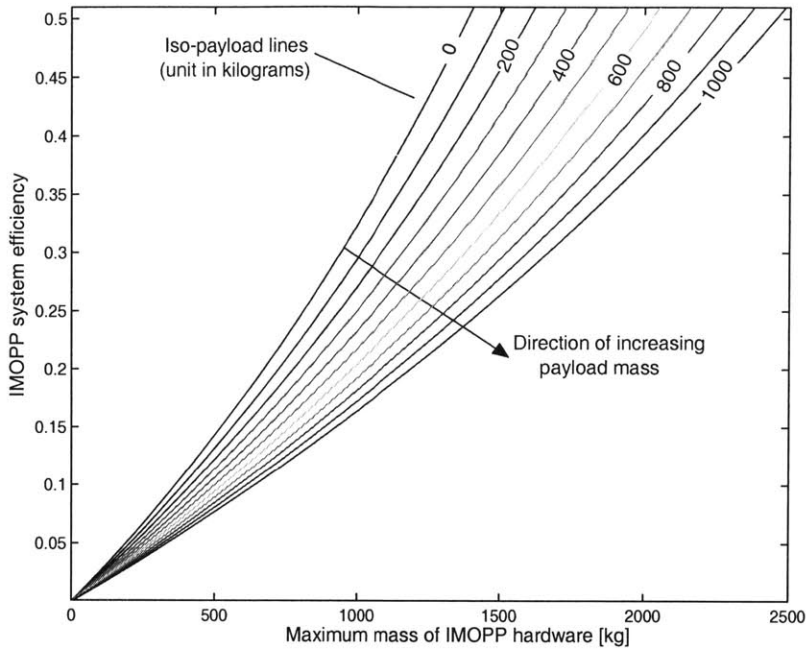


Figure 6-9: Trade-off between the fuel-production plant's efficiency and mass

load masses ranging from 0 to 1000 kilograms. Notice that while the zero kilogram line is a physical boundary (there can be no other iso-payload curve above that line), the 1000 kilogram iso-payload line is an arbitrary upper bound (other curves could be shown below that line). When the design is moved up along an iso-payload line, the efficiency, as well as the maximum fuel-production hardware mass, increase. On the one hand, the propellant generation plant is less constrained regarding its mass; on the other hand, the system must achieve a higher performance. Given a payload mass, designers can utilize Figure 6-9 to analyze how production efficiency can be traded for mass and the converse.

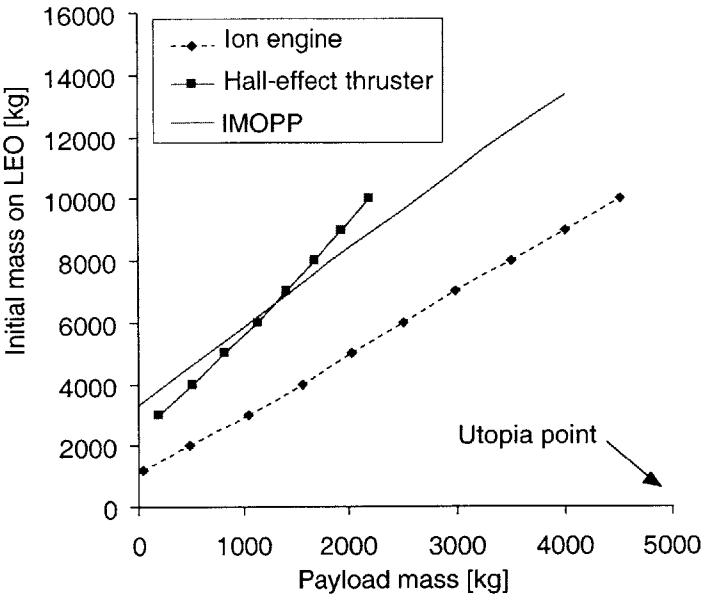
Travel time Assuming the aerobraking phases take less than 16 months (Equation 6.10), the PPIMO and standard propulsion have the same journey duration. They both come back to Earth using the Hohmann transfer opportunity subsequent to their departures.

6.8.2 PPIMO versus electric propulsion

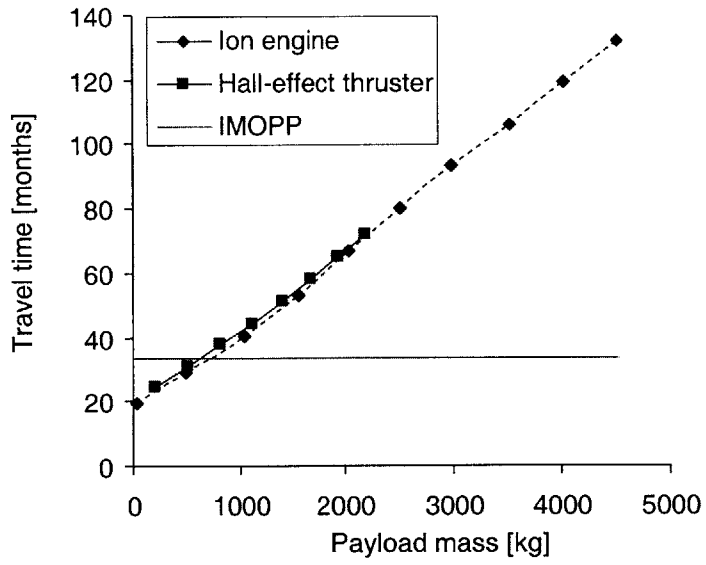
The performance of the PPIMO system is compared to that of electric devices in terms of payload capacity, travel time, and transportation ratio, which is the ratio of the first two variables [46].

Initial mass metric Figure 6-10(a) shows the spacecraft's initial mass in LEO as a function of its payload mass for the three propulsion systems: ion engine, Hall-effect thruster, and PPIMO. According to this figure the ion engine technology is the best because it provides the largest payload capacity for the smallest initial mass on LEO. For the three systems, the initial mass is an affine function of the payload capacity. The ion engine system is represented by the function with the smallest slope; in other words, the largest increase of payload mass for the smallest increase in initial mass. That slope is equal to 1.97 kilograms of increased initial mass per extra kilogram of payload. Notice the cross-over between the PPIMO and Hall systems for a payload mass of 1270 kilograms. For payloads larger than 1270 kilograms the PPIMO is the better solution. Still, for payload masses ranging from 0 to 2000 kilograms, the PPIMO and Hall-effect propulsions have globally a similar behavior with respect to the initial mass.

Travel time PPIMO spacecrafts use Hohmann transfers, and therefore, have a constant journey duration equal to 33.6 months (Equation 6.12). Figure 6-10(b) shows that both ion engines and Hall-effect thrusters follow the same travel time law. The plot represents the journey duration as a function of the payload mass. The figure shows that the rate of increase of the travel time is a little more than a week per 10 kilogram of payload for both electric devices. For a payload mass of 540 kilograms, there is a cross-over between the PPIMO propulsion and the electric propulsion devices. Below that value electric propulsion is quicker. With respect to the travel time metric, the PPIMO system is advantageous for large payloads.



(a) Paylaod capacity



(b) Travel time

Figure 6-10: Comparison of PPIMO and electric propulsions for the metrics of payload capacity and travel time

Transportation ratio The transportation ratio, q, is defined in this study as the normalized ratio of the payload capacity divided by the travel time.

$$q = \frac{M_{payload}}{T_{Journey}} \left. \frac{T}{M} \right|_0 \tag{6.59}$$

This metric captures both the aspects of payload mass and journey duration in a single parameter. It assumes that mission designers give an equal importance to mass and time; the weighing of the two variables is the same. In that case, the bi-objective problem of maximizing payload capacity and minimizing travel time is equivalent to the single-objective problem of maximizing the transportation ratio.

The transportation ratio is represented in Figure 6-11 as a function of the payload mass. For a the range of payload capacity from zero to 1000 kilograms, the

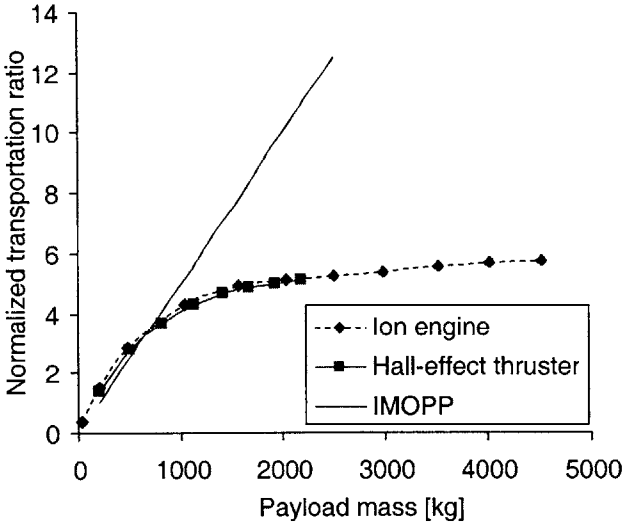


Figure 6-11: Transportation ratio as a function of the payload capacity

transportation ratio is sensibly the same for the three types of engines. For larger payloads, the PPIMO system shows a better transportation ratio. In that case, a study similar to that perform in Section 6.8.1 would derive the PPIMO's mass and efficiency required for it to remain competitive with respect to electric engines.

Conclusions Based on the results of this section, designers who are chiefly concerned about mass budgets should prefer the ion-engine solution. In the case where mass and travel time are equally important, the PPIMO technology has the best performance for payloads of more than 1000 kilograms. For smaller payloads, the results of this preliminary study are not sufficient to sort the propulsion alternatives.

6.9 Conclusions and future work

Benefits of PPIMO propulsion The comparison of the propulsion technologies shows that the PPIMO system is a advantageous compromise between the traditional chemical and electric propulsion systems. PPIMO propulsion permits shorter journeys than electric propulsion. Furthermore, PPIMO systems have a better overall fuel-efficiency than standard chemical propulsion sytems. The effective specific impulse, Isp_{eff}, of the PPIMO system is defined in Equation 6.60, and expressed as a function of the transformation factor β in Equation 6.61 (derived from Equation 6.32).

$$M_{LEO}^{IMOPP} = M_{dry}\, e^{\frac{\Delta V_{total}}{g\ Isp_{eff}}} \tag{6.60}$$

$$Isp_{eff} = \frac{1}{\frac{\Delta V_{total}}{g\ Isp} - ln\left(\frac{9+7\ \beta}{9}\right)} \frac{\Delta V_{total}}{g} \tag{6.61}$$

The effective specific impulse increases with β, and therefore, with the efficiency of the system. For the maximum efficiency, the effective specific impulse of the PPIMO system is 612 seconds, which represents an improvement of 28% compared to the standard chemical system.

The PPIMO system could play a more decisive role for a Mars sample return mission scenario. Such a mission involves the delivery to Mars of a Mars Ascent Vehicle (MAV). Once on the surface and loaded with Mars samples, the MAV propels itself from the surface and sends the samples in a return trajectory to Earth. To propel itself from the Martian surface, a MAV requires high-thrust chemical propulsion. Therefore, a PPIMO spacecraft could be used to deliver the MAV to Mars, and to produce in-situ its own return fuel along with the fuel needed for the MAV. Moreover,

the payload that must be delivered to the Martian surface has an estimated mass of 1800 kilograms [47]. This payload capacity is in the region where PPIMO is preferable to electric devices with respect to the transportation ratio metric.

In situ propellant production There are two aspects to the production of propellant in orbit: the interactions between the spacecraft and the Martian atmosphere, and the chemical process producing the fuel in the vehicle's reactors. Regarding the former, a more detailed energy transfer formulation must be derived. Notably, the capture of the atmosphere and its effect on the spacecraft's momentum must be defined. The study has to take into account the conservation of momentum for the system composed of the spacecraft and the atmosphere it captures. The spacecraft enters Mars' atmosphere with a mass, M_{arr}^{Mars}, and a velocity V_{hyp}^{Mars}. The vehicle captures and accelerates carbon dioxide until the vehicle reaches its circularization velocity.

$$M_{arr}^{Mars}\ V_{Esc}^{Mars} + m_{CO_2}\ V_{CO_2} = \left(m_{CO_2} + M_{arr}^{Mars}\right)\ V_{Circ}^{Mars} \tag{6.62}$$

The initial velocity, V_{CO_2}, of the carbon dioxide is negligible compared to that of the spacecraft. The maximum mass of carbon dioxide that can be captured by the spacecraft is then given by the equation below.

$$\frac{m_{CO_2}}{M_{arr}^{Mars}} \leq \sqrt{2 + \frac{R_{LMO}\ \mu_{Sun}}{R_{Mars}\ \mu_{Mars}} \left(1 - \sqrt{\frac{2\ R_{Earth}}{R_{Earth} + R_{Mars}}}\right)^2} - 1\ \approx 0.61 \tag{6.63}$$

The maximum amount of carbon dioxide that can be captured by the spacecraft is equal to little more than 61% of its initial mass at the time of Mars approach. The expression assumes that whole kinetic energy of the vehicle between the entry and the circularization is used to accelerate the carbon dioxide captured. In reality, some of that energy is converted into heat, and used in the reactors (Equations 6.20 to 6.24).

Regarding the fuel production in the reactors, the thermodynamics of the Sabatier and electrolysis reactions need to be examined thoroughly. The derivation of the temperatures that optimize the productivity of the overall process is of particular

importance. In parallel, other chemical options should be investigated. For example, an alternative is to produce carbon monoxide and oxygen from carbon dioxide.

$$CO_2 + \ 483 \text{ kJ} \rightarrow \text{CO} + \frac{1}{2}\text{O}_2 \tag{6.64}$$

This solution is appealing because it does not require any reactant to be brought from Earth. The feasibility study of the PPIMO system should benefit from research done in the fields of atmosphere capture during flight [43, 31] and ground in-situ propellant production [49].

Chapter 7

Conclusions and recommendations

This chapter contains the conclusions of the thesis on its three topics: Sample Preparation And Transfer (SPAT), Mars Surface Exploration (MSE), and Propellant Production In Mars Orbit (PPIMO). The chapter is organized into four subsections that include summary, contributions, future work, and usefulness of the three research topics.

7.1 Thesis summary

SPAT The goal of the SPAT study is to provide designers with a mathematical rationale for the development of shared-preparation facilities. The approach is as follows: first, all the possible SPAT architectures are generated, then, these architectures are compared and ranked with respect to their mass, sample throughput and operational risk. Mass is modeled by taking into account economy of scale for facilities with shared preparation. Sample throughput is characterized by the presence of bottlenecks in the sample flow. Operational risk is calculated via a reliability analysis based on Markov state models theory. The reliability analysis models two kinds of redundancies, cold and warm, for the elements constituting the SPAT system.

MSE The MSE study is motivated by the need for a broad systems engineering tool able to perform preliminary analyses on rover missions. For that purpose, MSE

applies multidisciplinary system design optimization techniques to the design of Mars rovers. Chapter 4 describes how the MSE framework is constructed. In addition, the chapter contains detailed modeling methods and assumptions for each rover subsystem. The validation section demonstrates that MSE is able to model with system-level accuracy most features from existing rover designs, and also to capture trade-offs inherent to rover systems. Chapter 5 applies MSE's analysis capabilities to design issues emerging from the rover design community. Furthermore, the chapter demonstrates the convenience of MSE's configuration. All the valuable results of that chapter are gleaned out of a single exploration of the trade space.

PPIMO This thesis presents the preliminary study of a new propulsion system, the PPIMO, based on the innovative idea of regenerative aerocapture and aerobraking. Propulsion models for the standard chemical, electric, and, especially, PPIMO systems are developed. The description of the PPIMO system includes a quantification of the energy gained by the regenerative aerocapture and aerobraking concept and a detailed derivation of its payload capacity. The three propulsion technologies are then compared with respect to their payload capacity, travel time, and transportation ratio.

7.2 Contributions

SPAT The study identifies the *sample path* as a key notion to depict SPAT systems from architectural and performance points of view. Optimal SPAT architectures comprise the shared and distributed preparation architectures as well as multiple hybrid architectures. Furthermore, the results of the research demonstrate that warm redundancy is advantageous for SPAT systems because it improves productivity by both reducing operational risk and removing sample throughput bottlenecks. Finally, a method is presented that determines what level of economy of scale the shared preparation architecture must meet in order to be competitive in comparison to the distributed architecture in terms of productivity per unit mass of the system.

MSE Other rover models already exist that take the approach of interconnecting sophisticated software design environments to conduct detailed analyses of a particular architecture. What these techniques gain in fidelity, they lose in breadth and agility. MSE's approach is a good complement to these techniques. Chapter 5 contains the contribution of MSE analysis capabilities to four engineering design issues. First, MSE is used to assess the benefits of oversizing a rover's suspension and provides the optimal dimensions of a suspension in relation to its payload. Second, MSE examines the trade-off between solar versus nuclear powered rovers and concludes that solar powered rovers cannot meet the Mars Science Laboratory mission requirements. Third, MSE assesses the advantages of multi-rover missions as opposed to single-rover missions. The analysis concludes that missions involving a team of three rovers are the most promising. Finally, the MSE study presents a method for budgeting the future development and validation of rover autonomy.

PPIMO The comparison of PPIMO propulsion with standard chemical and electric propulsions shows that PPIMO is a concept worthy of more research. On one hand, the payload capacity of the PPIMO system is larger than the capacity of a chemical propulsion system but less than that of an electric propulsion system. On the other hand, the duration of a return journey to Mars is shorter for a spacecraft using the PPIMO technology than for one using electric thrusters. The combination of these results into the metric of transportation ratio shows that PPIMO and electric propulsions perform similarly for payloads ranging from 0 to 1000 kilograms. Above that payload capacity, the PPIMO system shows the best performance. The study does not yet deal with the PPIMO hardware design. However, designers are provided with a method to determine the mass budget and efficiency requirement that PPIMO must exhibit to be a competitive technology.

7.3 Future work

SPAT There are two tasks related to the reliability analysis that should be subjects of future work. First, the redundancy analysis should be extended to all elements of the SPAT system. In the current model, redundancy can be applied only to the instruments of a SPAT system. The analysis would become more relevant if redundancy could be applied to processors as well. More redundancy possibilities entail a larger trade space of architectures. If the full-factorial search of the trade space becomes too computationally expensive, partial search methods should be developed. Second, the SPAT model would benefit from a more refined science utility function. With the current utility, an architecture with one failed instrument but a high sample throughput is preferred over an architecture with all its instruments but a slower throughput. This behavior does not reflect the high scientific value of each instrument.

MSE To improve MSE's reliability, some models need to be refined. The rover hardware and cost models are currently the weak points of the tool. In addition, a major improvement of MSE capabilities would be the implementation of a risk model that would encompass the whole lifecycle of a rover mission. The tool should be augmented with four technical risk elements: technology development, design uncertainty, verification and validation, and operational degradation. MSE's analyses would also gain in relevance with a more realistic science utility function. The article by Lincoln et al. [42] provides such a science utility function that captures the decrease in value of new samples as the total number of samples grows. Finally, as the complexity of the tool increases, intelligent search methods should be implemented to explore the trade space efficiently.

PPIMO The details of the transfer of the spacecraft's mechanical energy into thermal energy in the fuel reactors remains to be precisely quantified in order to make a solid case for the PPIMO propulsion. The energy transfer involves modeling of the transformation of the spacecraft's kinetic energy into heat and into kinetic energy of the atmosphere captured. The complexity of the system lies in the thermodynamics

of the atmosphere and heat flows to the reactors.

7.4 Usefulness

SPAT The SPAT engineering model is a valuable complement to the science-oriented research done on the Sample Preparation And Distribution system [17]. The study provides designers with a process for making traceable decisions about the degree of shared preparation and the type of redundancy that optimize a SPAT system.

MSE MSE has proved that it is capable in its current state of providing valuable insight to the design issues of rover mission developers. The MSE tool can easily be modified to model rover missions on other bodies, such as lunar missions for exploration or resource utilization. MSE itself is an example of a systems engineering design tool for complex systems. More than a rover-specific tool, MSE is a framework that can inspire the creation of similar tools for application to other complex systems.

PPIMO Neither standard chemical nor electrical systems are suited for Mars return missions. The former is not fuel efficient. The later requires excessively long travel times. PPIMO technology is an appropriate propulsion method for the scenario of Mars return missions: it shows better fuel efficiency than chemical propulsion and shorter journey durations than electric propulsion.

Bibliography

[1] Database available at http://www.lmd.jussieu.fr/en/Welcome.html.

[2] http://www.mysportscar.com/features/wheelweight.htm.

[3] Design handbook for honeycomb sandwich structures, 1970. Hexcel publication.

[4] The Viking mission to mars, 1975. Prepared by Martin Marietta Corporation.

[5] 2001 Mars Odyssey Arrival, October 2001. Press Kit.

[6] Mars 2007 Smart Lander mission – Science Definition Team Report, October 2001.

[7] Mars 2007 Smart Lander: reference science scenario for SDT report, September 2001. PowerPoint presentation.

[8] Mars Science Laboratory mission 2009 landed science payload – Proposal information package, November 2003. Draft.

[9] Project Science Integration Group instruments summary document. March 2003.

[10] Results from field/lab tests document of 1 and 10m drill, 2003. Honeybee Robotics publication.

[11] J. Appelbaum, G. Landis, and I. Sherman. Solar radiation on Mars - Update 1991. *Solar Energy*, 50(1):35–51, 1993.

[12] J. Appelbaum, G. Landis, and I. Sherman. Solar energy on Mars: Stationary photovoltaic array. *Journal of Propulsion and Power*, 11(3):554–561, 1995.

[13] Richard H. Battin. *An introduction to the mathematics and methods of astrodynamics, revised edition.* American Institute of Aeronautics and Astronautics, 1999.

[14] Richard H. Battin. Astrodynamics I, 2001. Massachusetts Institute of Technology.

[15] David Beaty, Sylvia Miller, Wayne Zimmerman, Terry Huntsberger, and Jeff Simmonds. Mars in-situ Sample Preparation And Distribution. JPL presentation, April 2002.

[16] David Beaty, Wayne Zimmerman, Sylvia Miller, and Jeff Simmonds. Plans for a Mars sample transfer/processing study. JPL presentation, February 2002.

[17] D.W. Beaty, S. Miller, W. Zimmerman, J. Bada, P. Conrad, E. Dupuis, T. Huntsberger, R. Ivlev, S.S. Kim, B.G. Lee, D.Lindstrom, L. Lorenzoni, P.Mahaffy, K. McNamara, D. Papanastassiou, S. Patrick, S. Peters, N. Rohatgi, J.J. Simmonds, J. Spray, T.D. Swindle, L. Tamppari, A. Trieman, and J.K. Wolfenbarger adn A. Zent. Planning for a Mars in situ sample preparation and distribution (SPAD) system. *Planetary and Space Science*, 52:55–66, 2004.

[18] Dimitri P. Bertsekas and John N. Tsitsiklis. Introction to probability, 2002. Notes from the Probabilistic Systems Analysis class.

[19] John Bresina, Gregory A. Dorais, Keith Golden, David E. Smith, and Richard Washington. Autonomous rovers for human exploration of Mars.

[20] Robert Roy Britt. Evidence of past water on Mars sets stage for the Future. *Space News International*, 15(13), March 2004.

[21] J.A. Crisp, J.K. Erickson R.B. Roncoli, P.C Theisinger, and R.A. Cook. Mars Exploration Rover (MER) Project Mission Plan, April 2002. Approved for external release.

[22] Leonard David. Returning rocks from mars: The latest plans, October 2001. Space.com.

[23] Olivier de Weck and Karen Wilcox. *Multidisciplinary System Design Optimization*, 2002. Multidisciplinary Design and Analysis Framework and Approaches.

[24] Centre National d'Etudes Spatiales, editor. *Missions, technologies et conception des véhicules mobiles planétaires / Missions, technologies and design of planetary mobile vehicles.* Cépaduès, Toulouse, September 1992.

[25] Michael A. Dornheim. Mars rovers will search for signs of ancient water. *Aviation Week & Space Technology*, may 2003.

[26] Emily Eelkema. MSL overview, April 2003. Presentation at the Jet Propulsion Laboratory.

[27] NASA Engineering for Complex Systems Program Office. Research opportunities in Engineering for Complex Systems, February 2003. NRA2-38150 (WLT).

[28] David W. Miller et al. *Rapid Modeling of Mars Robotic Explorers.* MIT graduate Space Systems Engineering class project, Spring, 2003.

[29] Orlando Figueroa and Jim Garvin. Following the water: the mars exploration program, 2003. Power point presentation.

[30] B. Gardini. CDF study report ExoMars 09.

[31] Archibald Gay. Method and apparatus for coordinating propulsion in a single stage space flight, September 1973. United States Patent number 3,756,024.

[32] Golombek, Haldermann, Forsberg-Taylor, DiMaggio, Schoeder, Jakosky, Mellon, and Matijevic. Rock size-frequency distributions on Mars: At the Pathfinder landing site, and in boulder fields, thermal inertia of rock populations, and rock shape and burial and implications for Mars Exploration Rover landing safety and operations. *Journal of Geophysical Research*, December 2002. MER special issue.

[33] Golombek and Rapp. Size-frequency distributions of rocks on Mars and Earth analog sites: Implications for future landed missions. *Journal of Geophysical Research*, 102(E2):4117–4129, February 1997.

[34] Mars Exploration Program Analysis Group. Scientific goals, objectives, investigations, and priorities: 2003, March 2004.

[35] A. F. C. Haldemann, E. T. Baumgartner, G. H. Bearman, D. L. Blaney, D. I. Brown, B. P. Dolgin, L. I. Dorsky, T. L. Huntsberger, A. Ksendzov, J. C. Mahoney, M. J. McKelvey, B. E. Pavri, G. A. Post, E. F. Tubbs, R. E. Arvidson, N. O. Snider, S. W. Squyres, S. Gorevan, G. Klingelhofer, B. Bernhardt, and R. Gellert. FIDO science payload simulating the Athena science payload. *Journal of Geophysical Research*, 107(E11), October 2002.

[36] F.P. Incropera and D.P. DeWitt. *Fundamentals of heat and mass transfer.* John Wiley & Sons, 1996. fourth edition.

[37] Cyrus D. Jilla and David W. Miller. *A multiobjective, multidisciplinary design optimization methodology for the conceptual design of distributed satellite systems.* PhD thesis, Massachusetts Institute of Technology, May 2002.

[38] JPL. In-situ sample transfer and processing. JPL presentation, March 2002.

[39] James A. Kurien, P. Pandurang Nayak, and Brian C. Williams. Model-based autonomy for robust Mars operations.

[40] Geoffrey A. Landis and Phillip P. Jenkins. Dust mitigation for Mars solar arrays, May 2002.

[41] Wayne Lee, Louis D'Armario, Ralph Roncoli, and John Smith. Mission design overview for the Mars 2003/2005 sample return mission, 1999. AAS 99-305.

[42] W. P. Lincoln, A. Elfes, T. Huntsberger, G. Rodriguez, and C. R. Weisbin. Relative benefits of potential autonomy technology investments, July 2003. International Conference on Space Mission Challenges for Information Technology.

[43] Michael A. Minovich. Self-refueling space propulsion system and operating method, July 1988. United States Patent number 4,754,601.

[44] Andrew H. Mishkin, Jack C. Morrison, Tam T. Nguyen, Henry W. Stone, Brian K. Cooper, and Brian H. Wilcox. Experiences with operations and autonomy of the Mars Pathfinder microrover.

[45] Michelle M. Munk and Richard W. Powell. Aeroassist technology planning for exploration, January 2000. 10th AAS/AIAA Space Flight Mechanics Meeting, Clearwater, Florida, AAS 00-169.

[46] David Y. Oh, Scott Kimbrel, and Manuel Martinez-Sanchez. End to end optimization of three dimensional chemical-electric orbit raising missions. International Electric Propulsion Conference IEPC-03-036.

[47] H. Price, K. Cramer, S. Doudrick, W. Lee, J. Matijevic, S. Weinstein, T. Lam-Trong, O. Marsal, and R. Mitcheltree. Mars Sample Return spacecraft systems architecture.

[48] Robert Shishko and Robert G. Chamberlain. *NASA Systems Engineering Handbook.* Number SP-610S. NASA, June 1995.

[49] K.R. Shridhar and M. Gottmann. 2001 Mars in-situ oxygen production flight demonstration. 99(2413), June 1999. 35th Joint propulsion conference and exhibit.

[50] Jeffrey H. Smith, Julie Wertz, and Charles Weisbin. Predicting the cost of new technology - An approach and case study for autonomy technologies, October 2003.

[51] C.R. Stoker, L. Richter, W.H. Smith, L.G. Lemke, P. Hammer, J.B. Dalton, B. Glass, and A. Zent. The Mars Underground Mole (MUM): A subsurface penetration device with in situ infrared reflectance and Raman spectroscopic sensing capability. 2003.

[52] H. Stone. Mars Pathfinder microrover, a small, low-cost, low-power spacecraft, August 1996. Available at http://mpfwww.jpl.nasa.gov/roverctrlnav/publications.html.

[53] George P. Sutton and Oscar Biblarz. *Rocket Propulsion Elements.* Wiley-Interscience, 2001. seventh edition.

[54] Walt Truszkowski, Harold Hallock, and James A. Kurien. Agent technology from a NASA perspective.

[55] Richard Washington, Keith Golden, John Bresina, David E. Smith, Corin Anderson, and Trey Smith. Autonomous rovers for mars exploration.

[56] James R. Wertz and Wiley J. Larson. *Space Mission Analysis and Design.* Space Technology Library, 1999. third edition.

[57] Julie Wertz and David Miller. Reliability and productivity modeling for the optimization of separated spacecraft interferometers. Master's thesis, Massachusetts Institute of Technology, May 2002.

[58] C. Whetsel. Surface system design and sizing relationships, 2003. Internal document.

[59] Byoungsam Woo, Victoria L. Coverstone, John W. Hartmann, and Michael Cupples. Outer-planet mission analysis using solar-electric ion propulsion, February 2003. 13th AAS/AIAA Space Flight Mechanics Meeting.