Model-Based Engineering Design for Trade Space Exploration throughout the Design Cycle

Elisabeth S. Lamassoure*, Stephen D. Wall† and Robert W. Easter‡
NASA Jet Propulsion Laboratory, Pasadena, California 91109

This paper presents ongoing work to standardize model-based system engineering as a complement to point design development in the conceptual design phase of deep space missions. It summarizes two first steps towards practical application of this capability within the framework of concurrent engineering design teams and their customers. The first step is standard generation of *system sensitivities models* as the output of concurrent engineering design sessions, representing the local trade space around a point design. A review of the chosen model development process, and the results of three case study examples, demonstrate that a simple update to the concurrent engineering design process can easily capture sensitivities to key requirements. It can serve as a valuable tool to analyze design drivers and uncover breakpoints in the design. The second step is development of rough-order-of-magnitude, broad-range-of-validity design models for *rapid exploration of the trade space, before selection of a point design.* At least one case study demonstrated the feasibility to generate such models in a concurrent engineering session. The experiment indicated that such a capability could yield valid system-level conclusions for a trade space composed of understood elements. Ongoing efforts are assessing the practicality of developing end-to-end system-level design models for use before even convening the first concurrent engineering session, starting with modeling an end-to-end Mars architecture.

I. Introduction

Engineering design practices have been experiencing an increasing pressure to change, as more and more space projects have been suffering from freezing their system-level design too early in their lifecycle. This paper describes work performed at the Jet Propulsion Laboratory (JPL) under the Model-Based Engineering Design (MBED) task, by which point designs are complemented by system-level models, allowing trade space exploration during early stages of development as well as dramatically improved flexibility to customer requirements

Space missions are divided into four phases: conceptual design, formulation, implementation, and operations. Traditionally, the conceptual phase develops a feasible design, from which requirements are derived. When several options are possible, trade studies are carried out to select the design that best meets the mission objectives while meeting cost and programmatic constraints. The formulation phase transforms these requirements into a detailed, build-able point design to be carried forward into the implementation phase.

These traditional design practices stem from the assumption that high-level requirements are well developed and fairly stable in the formulation phase; if such is the case, these requirements can be used as a metric for optimization, and early selection of the "optimal" point design is a cost- and schedule-efficient choice. In practice however, requirements keep fluctuating for several reasons. First, deeper analysis of the design almost invariably uncovers issues, some of which involve reconsideration at the system or even requirements level; very high cost estimates are a typical example. Second, sponsor requirements frequently continue to change due to funds availability, changes in overall objective, or both. Third, requirements from the science team (or another customer) may change, due to maturation of the investigations, results from other missions, or changes in the target itself. The result is a critical need for the capability to keep an evolving design and the evolving user needs in balance with each other.

The Model-Based Engineering Design (MBED) initiative at the Jet Propulsion Laboratory (MBED) stems from the belief that early modeling of the design can offer a solution to this problem. The MBED Vision is to complement

* Staff System Engineer, Mission and Systems Architecture Section, 4800 Oak Grove Drive, M/S 301-445
† Insert Job Title, Department Name, Address/Mail Stop, and AIAA Member Grade for second author.
‡ Insert Job Title, Department Name, Address/Mail Stop, and AIAA Member Grade for third author.

the current textual description of requirements and point designs by end-to-end models that relate requirements, system design, system performance and project cost.

A number of the tools, processes and teams required to make this Vision a reality already exist or are in development. Mission design tools, and scenario-driven simulations are in wide use. Concurrent engineering design teams such as the Advanced Projects Design Team (Team X) have set the standard for fast translation of requirements into design and trade-offs of requirements versus cost[1]. Agile trade space manipulation and visualization tools have been introduced for easier analysis of trade results[2]. Only recently have a few projects started integrating their subsystem models into system-level "project trade models" with the goal to carry out systematic exploration of the trade space[3,4] and estimate the partial derivatives of their cost to their requirements[5,6].

The MBED is a research initiative into the tools and processes required to assemble these pieces into a coherent whole with general applicability, so as to make model-based engineering design a standard practice for projects from the earliest stages of the formulation phase and throughout the implementation phase. Its vision has been reviewed in previous papers[7]. The first year of this multi-year task focused on developing MBED capabilities for the *conceptual design* phase, at the very earliest stages of a project. Two distinct set of capabilities were developed to address the particular needs of two types of missions: Earth-orbiting satellites and deep-space missions. This paper will treat only the latter.

Definitions of the words *model, tool, model uncertainty*, and *model error* as they are used in this paper are provided in the Appendix for clarity.

II. Current State of Trade Space Exploration for Deep-Space Missions

A. The Current Conceptual Design Process is Very Efficient at Science-Driven Point Design Development

The design processes, methods and tools used in NASA organizations are geared towards the "ideal" science-driven requirements flow down, leading to a point design optimized for a set of science requirements:

1) Level 1 requirements are set based on the key mission and program goals.
2) The Science Team develops a *Science Traceability* matrix tracing the mission goals to key NASA goals, and tracing them down into qualitative measurement objectives and quantitative measurement requirements
3) The Science and Instruments Teams decide on an instrument suite, and flow measurement requirements down into quantitative instrument requirements, used to develop the design of the instruments.
4) The Instruments and Engineering Teams translate instrument designs into payload accommodation requirements, to be used together with Level 1 requirements as the basis for system design.
5) A mission-level tree of trade options is developed and analyzed mostly qualitatively, taking into account likely impact on cost, risk, and performance towards meeting the requirements; state of technology, potential design heritage, and programmatic considerations also play an important role. One or a few options are retained for further study.
6) A design team is formed, with expert representatives of all subsystems for the mission. The team first develops a feasible design that meets the mission objectives, and estimates its cost.
7) A – If the design and cost are satisfying, the team uses this baseline architecture and system-level design as the basis for subsystem-level trades, to finally conclude on a design that is "near optimal", i.e. meeting the science requirements while minimizing project cost, at an acceptable level of risk.
   B – If the cost estimate exceeds the cost cap for the project, the team revisits its architectural trades (5) and measurement and / or its instruments requirements (3-4). Several cycles through the whole process are often required to converge on a solution with satisfying design and cost.

The traceability from science objectives down to spacecraft and mission design provides direction for any spacecraft design trade-offs and optimization. The process has been improved further by the use of concurrent engineering design teams. By having all subsystem experts working concurrently in the same room with links between their design tools, system trades and design iterations can be carried out in real time[1]. At the Jet Propulsion Laboratory, Steps 1) through 5) are performed by a core dedicated mission study team, including at least a Principal Investigator (PI), a study lead and a systems engineer, in the span of a few days to a few months. Once the architecture and requirements for the project are defined, they are very often taken to an Advanced Project Design Team (typically "Team X") for Step 6) using concurrent engineering, which makes it possible to generate a feasible design, internally consistent system-wide, in less than two weeks. The converged system-level design is transitioned back to the study team, which at this point includes all the subsystem experts required to proceed with detailed subsystem-level trades (Step 8A).

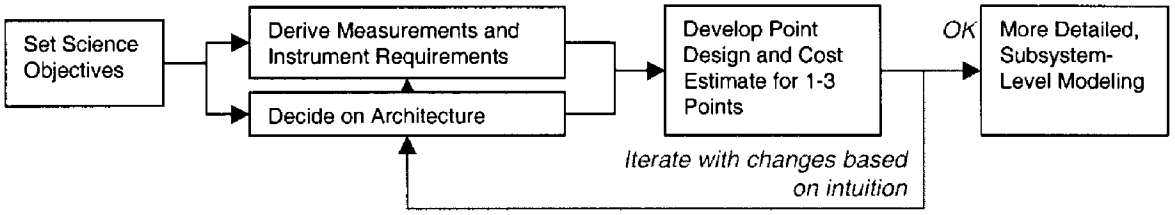
For missions with focused science objectives, mature instrument design, and small architectural trade space, this process is a very efficient way to optimize the end-to-end project design before moving into the implementation phase.

B. Increasing Need for Wider Trade Space Exploration and Point Design Sensitivities Analysis

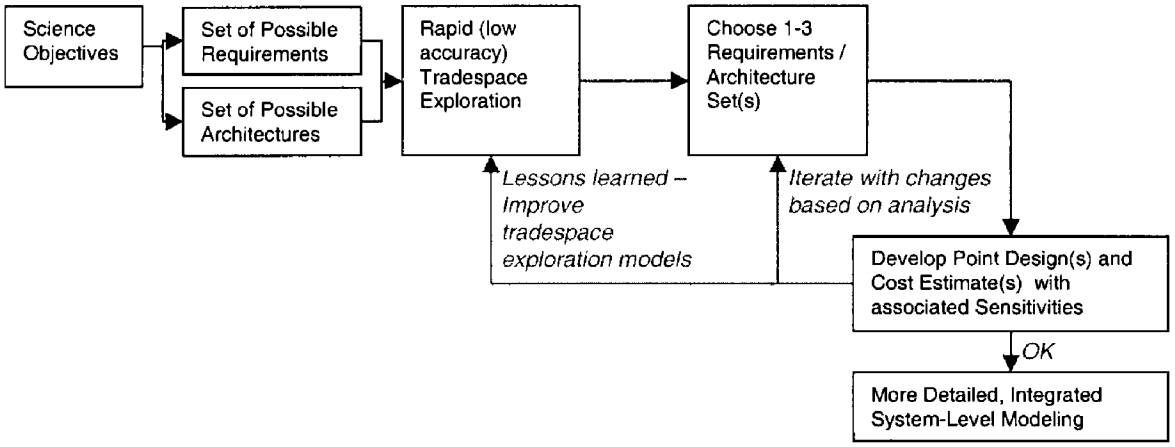
For more and more mission study teams however, improvements to the process are needed so as to be able to explore and understand a larger trade space. Trade-offs of project cost versus performance towards project and program goals are needed to reach a decision on the appropriate high-level requirements for the mission. Generating several point designs by repeating the cycle above provides an understanding of these trade-offs, but it very often too time- and resource-consuming to be considered.

Two additional capabilities can help improve the process:

1) An understanding of the *sensitivities* of the point design developed by the engineering team, with respect to its Level 1 requirements, payload accommodation requirements, and key design choices. How much heavier and how much more expensive does the flight system become for each added kg of instrument mass or instrument power, with each added km of orbital altitude, or with each added degree of latitude band access on the surface of the planet, with each added day of operations, or for a combination of the above? Design and cost sensitivities help discover what combination of requirements and design choices have the most impact on the design and cost of the project, thus providing invaluable guidance in iterating the requirements flow-down process.
2) For cases when the trade space is too large to know where to start, the capability to *rapidly explore a science or engineering trade space*, with moderate amount of effort, before deciding on the measurement requirements and key architectural features of the mission. By reducing the effort put into the analysis, the results for each point in the trade space would be expected to have higher deviation than would a conventional point design. But if the deviation can be kept small enough that *order of magnitude cost* and *relative conclusions* are accurate, this capability provides a quantitative map of a much larger trade space. The added value is a smarter choice of requirements and architecture before developing a point design.



a) Current Process, *simplified for the purpose of this discussion. "Shooting in the dark" might lead to many design iterations. Developing an understanding of a large trade space is prohibitive.*



b) Future Process, *as envisioned in this paper. Rapid trade space exploration and sensitivities analysis serve as guide for decision making and improve the quality of the design after the first iteration.*

Figure 1. Envisioned Improvements in the Conceptual Design Phase Process

For some design problems, the design team intuitively understands the drivers of the trade space, and quantitative analysis is not required. But as more and more complex flight system elements and programs are considered, these capabilities would increase the efficiency of the process and the value of the result, by avoiding any "shooting in the dark" and setting in on the first feasible result. Figure 1 compares the current conceptual design phase process with this new envisioned process.

The rest of the paper will describe recent work for development of two new functions: generating design sensitivity models as an output of a concurrent engineering design team, and developing models for rapid trade space exploration before concurrent engineering design sessions. A few case studies are provided to illustrate the potential offered by both capabilities. Future work will focus on bringing these developments into full operational status, so that they can become standard practice in the conceptual design phase of deep space missions.

III. Modeling Sensitivities around a Point Design in a Concurrent Engineering Team

A. A Model-Based Approach for a more Efficient Use of the Concurrent Engineering Design Sessions

1. Extracting the Whole Value of Concurrent Engineering Sessions

JPL's project-level concurrent engineering design team, Team X, typically takes as inputs project-level requirements (destination, lifetime, mission class, redundancy, measurement requirements) and sometimes payload accommodation requirements, and generates project-level outputs (project cost, schedule, risk) with component-level resolution (subsystem mass equipment list, mass, power, cost, block diagrams and configuration drawings). The outputs reflect a design that is converged at the system-level, and analyzed in sufficient depth at the subsystem-level to ensure feasibility, adequate choice of subsystem components, and reasonable cost estimate accuracy. Depending on the complexity of the mission (from simple Earth orbiter, to Mars Sample Return mission, through solar electric propulsion missions to outer planets), this point design accuracy is the work of ten to twenty senior engineers, over two to four concurrent engineering design sessions. During the sessions, the engineers use their expert judgment to tailor their standard design tools for use on the specific mission concept they are developing, and choose appropriate components from their database. In particular, the relationships between subsystems, which corresponds to the way system requirements flow down into subsystem requirements, change for each particular point design. For example, instrument data rate could drive the telecom subsystem data rate capability, which could drive the telecom antenna beamwidth, which in turn could drive the attitude control subsystem's (ACS) pointing requirement; alternatively, instrument pointing requirement could drive the ACS pointing capability, which together with power constraints could drive the choice of telecom antenna beamwidth, in turn driving the maximum telecom data rate, finally resulting in a minimum duration for the telecom pass.

Although a description of a converged point design and its cost estimate are the final product currently delivered by concurrent engineering design teams, the development of these subsystem relationships is another critical value-added component of the design sessions, which could be used more effectively. If the Customer team wants to consider another option, or discovers that they have to slightly change one of the requirements or instrument design, they are currently faced with two choices: approximating the effect on the Team X study results with their own tools; or scheduling a new Team X session.

For small excursions around the point design however, such as small change in instrument mass or power, orbit altitude or landing site latitude, a lot of the design choices made by the team would remain essentially the same. The session would consist in incremental design changes for the subsystems directly affected by the change in requirements, and propagation of these changes through the whole system using the same relationships as were developed for the original design. Unless new system-level issues are uncovered which require revisiting design decisions and subsystem relationships, such a mechanical convergence of already-established relationships is clearly not a good use of the Team's time and the Customer's resources. Nor does it make sense for the Customer team to try and re-create the subsystem relationships already developed by Team X.

A better use of resources is to extract from the concurrent engineering design team, not only the point design, but the set-up of subsystem relationships that were arrived at for the point design. Assembled together in an automated tool, these relationships can form a *project-level design model with subsystem-level resolution* and *local* range of validity around the point design, hereafter called *sensitivities model.*

Since the Team made a lot of decisions that are specific to the point design they were considering, understanding the range of validity of each relationship is then critical. Designs that would fall outside of the range of validity would indicate possible system-level issues, or changes in design choices, which are worth a closer look by the Team. Only design points that thus represent discrete steps in the trade space, for which the Team's time adds the

most value, would require a new design session. The new session would result in a second set of relationships, valid in a separate range of validity.

2. *Requesting Equations in addition to Equation Results*

A reasonably simple upgrade of the concurrent engineering design toolkit is all that is needed to generate such *sensitivities models.*

By the nature of their work, concurrent engineering design teams are equipped with the tools to automatically exchange design parameters. In the case of Team X, the toolkit consists of:

1) One spreadsheet-based design tool per subsystem. The only common point between all subsystem spreadsheets is their "send / request" feature by which Inputs from other subsystems can be automatically updated, and Outputs automatically sent. The rest of the spreadsheet includes the databases, design guidelines and design models specific to the subsystem.

2) A piece of software (ICEMaker©) that automatically distributes all requested Inputs and Outputs throughout the subsystem tools.

For each point design, each subsystem decides what portion of its design tool to use: which inputs from other subsystems will be the design drivers; for continuous design parameters (such as structural mass), which available model will represent the relationships; and for discrete design parameters (such as the choice of telecom radio), which component from the database best meets the mission needs. Based on these choices and the inputs received from other subsystems, the subsystem will output to ICEMaker one set of numbers representing one point design.

We are using the following approach to automate the development of *sensitivities models* in Team X:

1) Develop a list of types of requirements with respect to which we want to be able to estimate sensitivity. This includes instrument mass, power, and data rate; mission lifetime; level of redundancy; geographical access (orbital parameters or landing site latitude band and altitude).

2) Develop a matrix or diagram of the typical interdependencies between subsystems; this matrix is a guide as to how a change in requirements will propagate through the whole design. Figure 2 provides an example of such an interdependencies for a Mars Lander mission.

3) For each subsystem, determine the set of possible design drivers and update the interdependencies matrix accordingly. Extract the design drivers that could be changed through a change in one of the important requirements identified in (1).

4) For each subsystem, list all the possible subsystem component types. For each component type, determine which design drivers are most important in component selection and sizing.

5) Discrete component types, such as avionics components, represent possibly complex step functions. For the purpose of sensitivity analysis, they can be represented by a range of validity for the value of a key design driver, for in which the component remains the same. For a model with slightly larger range of validity, two additional set of outputs can be created, representing the components that the subsystem engineer would choose for a design driver value "below" (with lower bound) or "above" (with upper bound) the baseline range of validity.

6) Continuous component types, such as inert masses, are typically local functions of their design driver(s) with fixed form, but coefficients (and exponents) varying for each design. Two types of new outputs are sufficient to capture their sensitivities: the value of the equation coefficients, and the estimated range of validity of the equation.

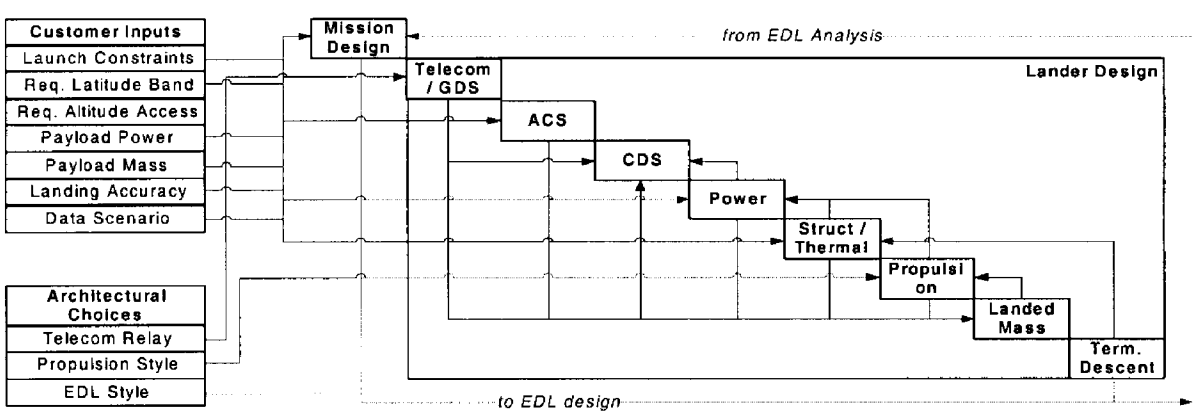


Figure 2. Example of Subsystem Interdependencies for a Mars Lander Mission

Table 1. Updates to Concurrent Engineering Design Tools to Generate Sensitivities Models

Type of Information	*Subsystem component with discrete choices*	*Continuous subsystem component*
Published before	▪ One set of information (mass, power, cost, performance)	▪ One set of information (mass, power, cost, performance)
Assumed in sensitivities model template	▪ Dependent on one key design driver ▪ Unchanged in small range around the point design	▪ Dependent on a set of design drivers ▪ Sizing relationship do not change form; coefficients are unchanged in small range around the point design
Published in addition	▪ Range of validity (on key design driver) –low, high ▪ One set of information for component "below" with range of validity ▪ One set of information for component "above" with range of validity	▪ One set of relationship coefficients per key output (mass, power, etc) ▪ Range of validity (on most important design driver)

7) Based on this information, a *sensitivities model template* can be created and set up to use the above information as input, and produce delta-point-designs and validity flags as outputs. In other words, the *sensitivities model template* is a *tool* that, once fed with equation coefficients are inputs, instantiates a *model.*

As summarized in Table 1, the process change requires additional work from each subsystem engineer to decide on the validity range within which they are comfortable with their design choice, and in the case of discrete components, think about what other choices they would make slightly beyond this range. This is work that the subsystem engineer would have to carry out in any case, if the Customer requests a second option be considered by the Team. By performing the work ahead of time instead, a variety of "delta-designs" can be automatically generated for much less extra effort.

3. Model Deviation and Range of Validity

If applied consistently, this method ensures that the model simulates the design change process that the concurrent engineering design team would follow if asked to provide a small design variation. The deviation on the model results is about as small as the deviation on the point design results, as long as the variations remain within the validity range identified by each subsystem.

As we will know illustrate on a few examples, the sensitivities model thus produced is much more than a series of partial derivative estimates. Instead, it is a multi-variant design model of the local project trade space.

B. Example 1: Solar Electric Propulsion mission options on the propulsion, power, and mission designs

Figure 3 illustrates the importance of modeling to estimate the value of sensitivities and its value to explore the local tradespace around a point design. It represents the results of a concurrent engineering study for a deep space mission using solar electric propulsion. The team developed a point design that, although meeting all mission requirements, was slightly above the acceptable cost cap. The Customer team agreed to consider trading off three elements of performance to save cost: dry mass contingency (which would increase the risk that the spacecraft would not fit on the launch vehicle), solar array technology (using a newer, lighter weight array would decrease mass, but increase cost and risk), and the use of reaction wheels for precise attitude control (which would reduce the science value of the measurements).

The trade is driven by the interdependencies between the trajectory, the number of operating electric engines, and the power available from the solar arrays. The optimum trajectory depends on the available power, number of operating engines and spacecraft launch mass. It turn, if affects launch energy, which drives launch vehicle capability; and propellant mass, which affects the number of engines required to meet throughput requirements. Figure 3 plots the cost savings versus dry mass contingency trade space for four different options relating to number of engines, solar array technology, and presence of reaction wheels. For each case, several points represent different levels of solar array power.

For each option, the chart identifies the "maximum" power level over which spending more money on the solar array doesn't increase the trajectory performance enough to increase launch mass margin. It shows how in the cost / mass margin space, two options clearly dominate (squares for low cost, triangles for mass performance). However, none of them carries reaction wheels, which would provide additional science value.

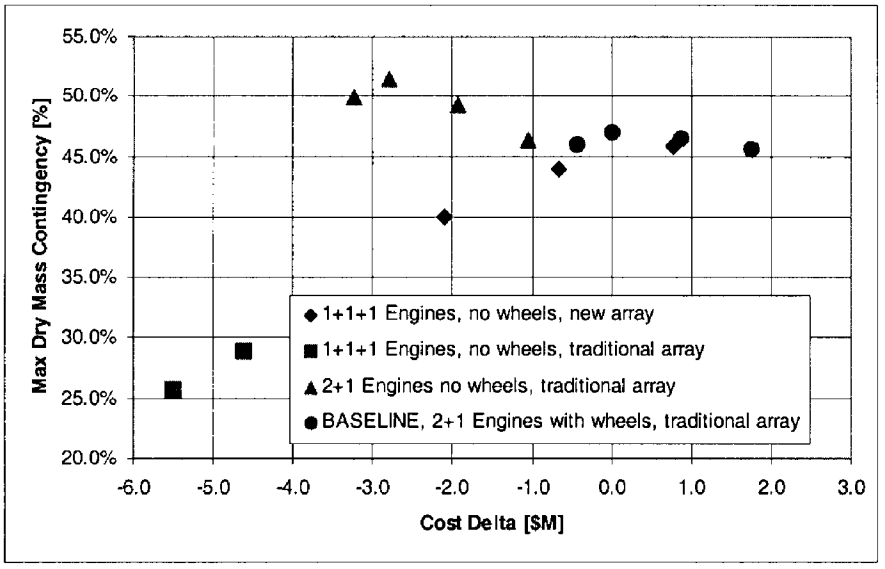


Figure 3. Example of Sensitivities Study Results for a Solar Electric Propulsion Mission

Instead of defining an "optimal" design based on some overarching value metric, delivering a plot such as Fig. 3 helps the Customer understand the trade space landscape, and make their own decision based on a variety of important metrics.

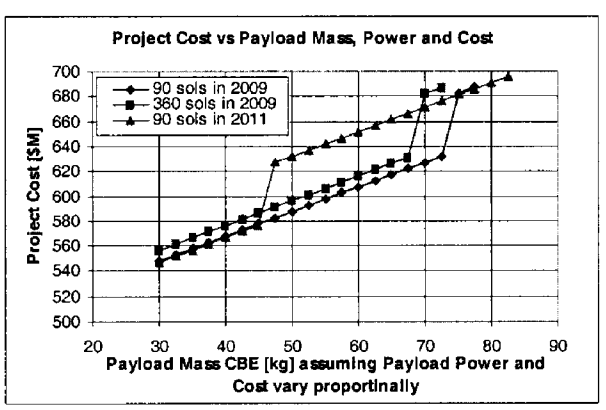
The very same sensitivities model could be used by technologists, to evaluate sensitivities of the mission design and project cost to technology performance parameters, such as the specific impulse of the propulsion system or the efficiency of the solar array. The sensitivity results would provide an estimate of the value of the technology improvement to the mission, which could be compared to the cost of technology development.

C. Example 2: Sensitivities Models for Mars Landers

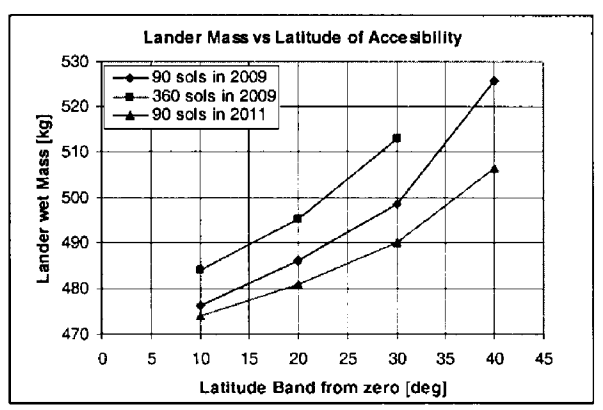
The next examples illustrate how the same process can be used to model mission types with different challenges and different questions.

Figure 4 shows the sensitivities results for the conceptual study of a small human precursor Lander mission to Mars. The requirements for such a mission are still immature. Instead of trying to optimize the design for set science requirements, this time the sensitivities are used to map out the requirements space.

Given all the unknowns, the Customer was interested in sensitivity to launch date (which changes trajectory characteristics), payload mass and power (which relates to various instrument suite options), and latitude access



a) *Project Cost Sensitivity to Payload Mass and Power*



b) *Wet Mass Sensitivity to Latitude Band of Access*

Figure 4. Example Sensitivities Model Results for a Mars Human Precursor Lander Study. *Model results are graphed beyond their likely validity range to better illustrate break points.*

requirements. The Customer was also interested in knowing the breakpoints in the design of the entry system, over which heritage systems would not be appropriate.

Figure 4 illustrates how the sensitivities model results improve on what is learnt from the point design. It shows that project cost grows about linearly with payload size, but identifies the breakpoints incurred when a larger launch vehicle is needed. It also shows how instrument mass and power, mission lifetime, and latitude access band can be traded-off against each other to reach the same wet mass, launch vehicle choice and project cost. Some less-intuitive conclusions can also be reached; for example, launching in 2011 reduced Lander wet mass because of the more benign arrival conditions (summer arrival). But above a given payload mass, it increases project cost significantly because the higher launch energy requires a larger launch vehicle.

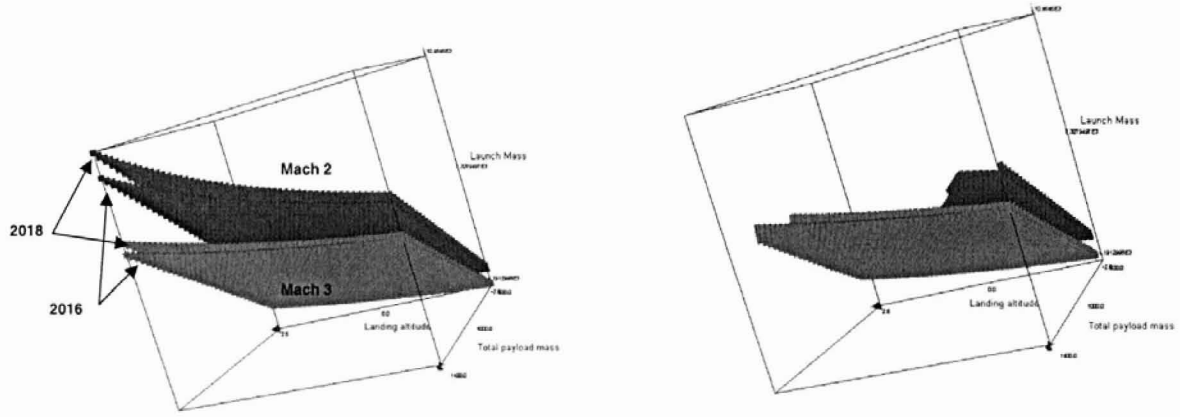
Figure 5 displays the results of another Mars human precursor Lander study. Lower down the line of human precursor missions, this study considered demonstrating new entry, descent and landing (EDL) technologies to land a heavy human precursor payload, such as a long-life drilling platform and subscale in-situ propellant production and storage plant. With help from Johnson Space Center and Langley Research Center, the Team considered the use of a new entry and descent configuration, a mid-Lift-to-Drag ratio aeroshell and high-Mach parachute configuration for the mission, as a demonstration of the type of technology required for the eventual human mission to Mars. Whereas the overall architecture was chosen, the payload mass was known only very approximately, and the landing site requirements remain very uncertain. Understanding sensitivities to these parameters was therefore an important complement to the point design.

Here again, the results of the sensitivities model (Fig. 5) help quantify a number of valuable results, such as the launch mass increases sensitivity payload mass and landing altitude. It also helps identify breakpoints in technology: a Mach 3 parachute could achieve the same landed mass at a much smaller launch mass than a Mach 2 parachute. Alternatively, for a given altitude, there is a maximum payload mass that can be delivered by each type of parachute.

IV. Broad Trade Space Exploration Models for Initial Point Design Selection

A. Models at Varying Levels of Error for Varying Levels of Detail

The previous examples dealt with local trade space exploration around an existing point design. Before reaching



a) *Model results are shown beyond actual validity to emphasize the shape of the sensitivities surface.* **b)** *Only designs that can fit in the launch vehicle are shown.*

Figure 5. Example Sensitivities Model Results for a Mars Heavy Landing Study
Horizontal axis represents landing altitude. Out-of-page axis represents total payload mass. Vertical axis represents launch mass.

that point however, mission objectives and Level 1 requirements had to be known, and a number of architectural decisions had to be made. Making these decisions represents slicing through a much larger initial trade space, but deciding, for example, what type of platform to use (lander or rover, or both?), what instruments to land, what landing accuracy to require with what landing style, etc.

1. The Theoretical Foundations for these Improvements Have Been Laid Out by University Research

Ref. 8 provides an excellent overview of recent research in methods for rapid trade space exploration which can pave the way for practical applications. The paper describes a method for building Customer-specific trade space exploration models. The level of resolution is controlled by focusing the modeling effort on a limited number of attributes that constitute what the Customer really cares about. New multi-attribute utility assessment processes and tools are proposed to provide an optimization metric representative of the concerns from a variety of stakeholders. Design variables are selected using methods such as Quality Function Deployment (QFD). The models that translate a design vector into values for the attributes leverage existing orbit simulation software packages, and the well-established rules-of-thumb for design of Earth orbiters[9]. The result is efficient, value-oriented rapid trade space exploration, and a modeling of utility that can be brought as added value into the concurrent engineering design session.

2. Adapting the Theory for Successful Practical Use

The theoretical foundations briefly summarized above require adaptation for successful practical use in the conceptual design phase of deep space missions. It relies on the assumption that rules-of-thumb exist for the design of the spacecraft, is the case for Earth orbiters. Before deep space missions can take full advantage of advanced trade space exploration techniques however, they need to start from "square 1" and develop adequate rules-of-thumb design models. Given the one-of-a-kind nature of deep space missions, the validity of rules-of-thumb can be questionable. This raises a number of challenges.

First, the development needs to be complemented by a method to *track deviation and validity range*, so that results can be effectively used as decision making aids. Deciding to choose an area of a large trade space for closer study by a design team, relies on the conclusions that the project cost has an acceptable order-of-magnitude, the mission an acceptable level of performance; and that the mission showed either lower cost, and/or higher performance, than a number of alternatives. To know whether this conclusion is valid, estimates of the *absolute* and *relative* deviation of the model are required.

An estimate of the *absolute* deviation will help determine that the results are "in the right ballpark". This will typically be sufficient for very early, large trade space exploration, as long as "the model is valid for *relative* conclusions". This statement, however, is not obvious. Considering the example of Fig. 6, analysts would agree that design C is more expensive than design A, but could argue about design B. Some people would conclude that, the difference between B and C being smaller than the deviation on both, no conclusion can be reached; they are making the assumptions that the deviations are uncorrelated. Others would argue that although the absolute deviation is large, trends are well captured, therefore C is more expensive than B; they are making the assumption that the deviations are fully correlated. Reality probably lies in between these two viewpoints.

For rapid tradespace exploration models to provide useful results, some level of *absolute* and *relative* validation of the model is therefore required. This is a non-trivial problem when modeling new types of flight systems.

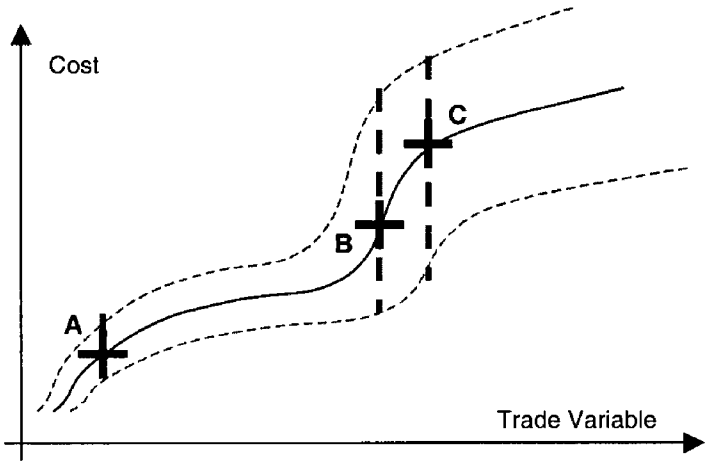


Figure 6. The Need for *Relative* Deviation Estimates: *Is C more expensive than B?*

This leads to the second implementation challenge. Assembling rapid trade space exploration models with sufficient levels of accuracy requires some analysis by the right combination of experts. For deep space missions in particular, there is no published work of design rules-of-thumb, such as can be found for Earth orbiters[9]. Experts in their engineering fields are not necessarily expert software developers. Unless an efficient process is developed, the modeling effort is likely to require too much time from too many experts than is valuable or affordable to Customers.

A third practical limitation of the theoretical approach is less critical in the short term. Many research efforts are developing methods for aggregating performance metrics into one, overarching value metric for the mission, on the basis of which optimal designs can be sought[10]. Although this goal is noble and attractive, its application is not always practical, but fortunately not always necessary either. It is indeed very hard to get the required information (basis for a utility metric) from the Customers and other stakeholders, because of the nature of the information being requested, but also because of the time the process requires, and how often it would need to be repeated. However, optimizing the design at early stages of the trade space exploration is not a necessity and could be premature. Understanding all the dimensions of the trade space, and its bearing on all metrics of interest (or attributes), helps the Customers as well as the team understand the trade space they are building, the interdependencies between elements of the mission, the design and cost drivers. This learning experience is one of the key values added of the trade space exploration, which could be undermined by attempts at reducing the number of dimensions too early. Once trade space exploration for deep space missions is as well understood as it is for Earth missions, the methods will prove very valuable. Ongoing research into reducing the interview time will help standardizing their use.

B. Proof of Concept Example: a New use of Concurrent Engineering Design Teams

In spite of these challenges, there are cases when rough design models for trade space exploration before point can be successfully put into practice for selection of a point design to develop. A study of a small, Europa lander mission proved so and provided interesting lessons-learned.

1. The Customer's Problem

The Customer for the study came to the concurrent engineering team with a "reverse" problem: given three different wet mass allocations, what amount of science can be performed by a Europa lander, as a function of a few key design choices: landing style, power source technology, and lifetime on the surface?

Such a study is not easily answered by the traditional conceptual design process, which would require developing more than 27 point designs, with several iterations each time to converge on a science package that fits the mass allocation. The problem was made even more challenging when discussion with the design team uncovered several other trade space dimensions: choice of propulsion technology; assumptions on technology development in lightweight, low power instruments; and assumptions on technology development in radiation-hard components. The Customer being more interested in a picture of the trade space than in the level of accuracy of any point in the space, the study was very well suited to a rapid trade space exploration experiment.

2. Approach

The approach taken to solve the Customer's problem is probably representative of a practical rapid trade space exploration process for missions with no existing rules-of-thumb:

1) Based on discussion with the design team, decide what part of the trade space can be the object of rapid trade space exploration. Parts of the trade space that represent totally new areas of design, for which the team has no previous experience, are not suited for rules-of-thumb modeling; any conclusions reached would depend on too many assumptions. Such was the case of some of the landing methods in the trade space. They were kept for later closer analysis and detailed mechanical modeling, only method by which the required knowledge could be generated. Any rules-of-thumb reached after more detailed modeling could be incorporated in the future.

2) Break down the system-level trade into subsystem-level relationships. This work, to be orchestrated by the system engineer, is very similar to the development of interdependencies matrices described for sensitivity analysis, only at a lower level of resolution (subsystem-level instead of component-level).

3) Carry out a design session dedicated to development of a trade space exploration model. Based on the system breakdown, each subsystem is allocated a set of inputs and outputs, and responsible for providing the relationships between its inputs and its outputs. For example, the power subsystem would be responsible for translating total spacecraft power requirement, mission duration, and level of technology development, into mass of the power subsystem. The instruments subsystem would be responsible for translating total instruments mass and power allocations, and level of instrument technology development into a choice of an instruments suite, which could be used as a metric for science value.

4) As relationships come in during the session, the system engineer assembles them into a model of the design that takes all trade space variables as input, and outputs key metrics. In this case, inputs included: instruments mass and power, mission duration, choice of power source, choice of propulsion option, state of development of radiation-hard technologies, and state of development of lightweight instruments. The two main outputs were the choice of an instrument suite, and the total spacecraft wet mass.

5) After the session, run the model in automated mode to generate outputs on the whole trade space and generate summary tables and plots to be discussed with the Customer.

6) Hold a meeting with the Customer, and a subset of the design team to analyze the trade space exploration results and choose a few point designs for further analysis. In this case, only a few cases appeared to fit within one of the possible mass allocations, because of the challenges to soft-land on Europa, and survive in its high-radiation environment.

7) Reconvene the whole design team to develop point designs for the chosen point(s). In this case, three point designs were developed by the team. Design sessions helped uncover more issues, which increased the mass estimates of some subsystems. But it also provided enough time, mission information and brain power for new ideas to emerge on how to design and configure the spacecraft based on the specific challenges of the mission at hand. This reduced the mass of other subsystems.

8) Compare the results of the rapid trade space exploration with the results of the point designs to estimate the error bar of the trade space exploration model and conclude on the validity of its conclusions.

9) Finally, update the trade space exploration model with the new information generated during the design session (issues and new ideas). Run the full trade space exploration again to generate a more accurate map of the mission trade space.

Table 2 summarizes the results of the error bar analysis. They are interesting as representative of the type of errors to be expected from a system-level design model with subsystem-level resolution. The absolute errors at the subsystem level can very high, and are higher for the subsystems that were not represented in the modeling session. However, the subsystem-level errors are independent enough, that they get statistically reduced once added-up, in spite of the interdependencies of the subsystem designs. The absolute error on the system-level mass is reasonable for order-of-magnitude conclusions. Even more interestingly, the *relative* error on the system-level mass is low, ensuring that the right conclusions where reached when comparing the total mass of one design to another. Some of the subsystem-level relative error bars are beyond -100%, indicating that the model captured the wrong subsystem-level trend. This confirms that such a model is not suited for subsystem-level conclusions.

3. *Conclusions and Lessons Learned*

This study provided a number of lessons-learned.

First, it is feasible to rapidly generate system-level design models with subsystem-level resolution for rapid trade space exploration and selection of a point design, provided that the right expertise is available, the right process is followed, and the team works with the right mindset.

Second, that the accuracy to be expected from such models is good for relative system-level conclusions, adequate for order-of-magnitude system sizing, but inadequate for any subsystem-level conclusion.

Third, that only the part of the trade space for which the team has an existing knowledge-base can be rapidly modeled. Any design option relying on new concepts would require more detailed analysis.

Finally, that the model will not include any creative design the team might invent. Design models assume that design is deterministic, thus ignoring the large contribution of team creativity to improving the design.

Table 2. Error Bars Estimates on Example Broad Tradespace Exploration Model.

	Absolute Difference			*Relative Difference*		
Mass Element	Option 1	Option 2	Option 3	Op 3 / 1	Op 2 / 1	Op 1 / 3
Attitude Control	-30%	-30%	-30%	0%	0%	0%
Command and Data Handling	-55%	-55%	-55%	0%	0%	0%
Power	+31%	+19%	+0%	-59%	-27%	+350%
Propulsion	+19%	+19%	+19%	0%	+6%	+15%
Structures and Cabling	+66%	+68%	+66%	+2%	13%	+28%
Telecom	+92%	+92%	+92%	0%	0%	0%
Thermal	-7%	-1%	+138%	-103%	+374%	-95%
Total Dry	*+31%*	*+31%*	*+30%*	*-7%*	*-1%*	*8%*
Propellant	+25%	+25%	+25%	-7%	-1%	8%
Total Wet	*+29%*	*+29%*	*+28%*	*-7%*	*-1%*	*8%*

C. Approach for a More Systematic Capture of General Rules-of-Thumb in the Existing Knowledge Base

The Europa lander case study proved the feasibility of rapid trade space exploration models, but produced a model of little use beyond the Customer of the study. Its relationships were tailored towards answering the particular questions asked, and would not constitute a general model. How general a model is it possible to build for deep space missions?

To answer this question, we chose two classes of deep-space missions that seemed general enough to cover a lot of potential conceptual design studies; specific enough that they can make the object of *one* design model; and conventional enough that there would be an existing knowledge base from which to draw rules-of-thumb design relationships. If having general system-level design models is feasible at all, this would be the place to start. The choice was therefore: Planetary Orbiters; and Mars Lander missions, which include models for a Lander, an Entry System, and a Cruise Stage.

1. Model Development Approach

Under the name "System-level design model of a planetary Orbiter", many models with totally different inputs and outputs could be envisioned. When developing a design model without a particular trade in mind, one must decide on the type of questions they would like the model to be able to answer. The level of resolution a design model needs to go into depends on the type of trades it needs to conclude on. As a principle, the level of resolution should be kept to a minimum while capturing all the important trades with the right level of accuracy.

Figure 7 summarizes Inputs representative of the type of trade space of importance to conceptual design studies. As outputs of the model, we need enough information to get a system-level summary of the design, and provide inputs to a system-level cost model. The list of inputs identified requires subsystem-level resolution. Based on this list of inputs and outputs, a system engineer can break down the question into subsystem-level inputs and outputs, and a team of subsystem experts can develop rules-of-thumb based on design experience.

Once system-level models are developed, they can be assembled into a project-level model as graphically represented on Fig. 8.

Mars Lander Inputs			
Requirements	Landing style	Propulsion Choice	Telecom Choice
Instruments mass, power, data rate	Rough	Monoprop / Biprop	Relay only
Other carried elements (Rover)	Airbags	Landing only / + Entry / + Cruise	Direct-to-Earth only
Landing season	Soft, Viking-style		Relay and Direct-to-Earth
Surface lifetime	Soft, Skycrane-style	Power Choice	
Landing accuracy required		Batteries only	
Mission class / redundancy		Solar (articulated / non-articulated)	
Launch date		Nuclear	
Mars Entry System Inputs			
Requirements	Aeroshell Type	Subsystems	
Lander Wet Mass	Blunt (cone angle)	Propulsion on EDL or Lander	
Arrival Velocity, EFPA	Ellipsled (Lift/Drag)	Avionics on EDL or Lander	
Landing Altitude		Lander characteristics	
Landing Ellipse	Parachutes	Min / Max thrust, Isp	
Launch Vehicle Diameter	Mach 2	(for calculation of terminal descent propellant based on	
Mission class / redundancy	Mach 3	parachute release altitude)	
Launch date			
Carrier Stage Inputs			
Requirements		Subsystems (on Carried Elements / on Cruise)	
Carried Elements Mass and Number		Attitude Control	
Trajectory Correction Maneuvers		Command and Data Handling	
Cruise duration		Telecom	
Mission class / redundancy		Propulsion	
Launch date			
Planetary Orbiter Inputs		Propulsion Choices	ACS Choices
Requirements		Monoprop	3-axis stabilized
Planet		Biprop	Spin-stabilized
Instruments: mass, power, data rate, pointing accuracy		Solid	
Desired orbital elements			
Delta-V for all required maneuvers		Telecom / Operations Options	
Mission class / redundancy		Frequency band	
Launch date		Choice of Earth station antenna size	
Launch vehicle		Choice of Earth pass duration per day	

Figure 7. Typical Trade Space Options for Mars Missions

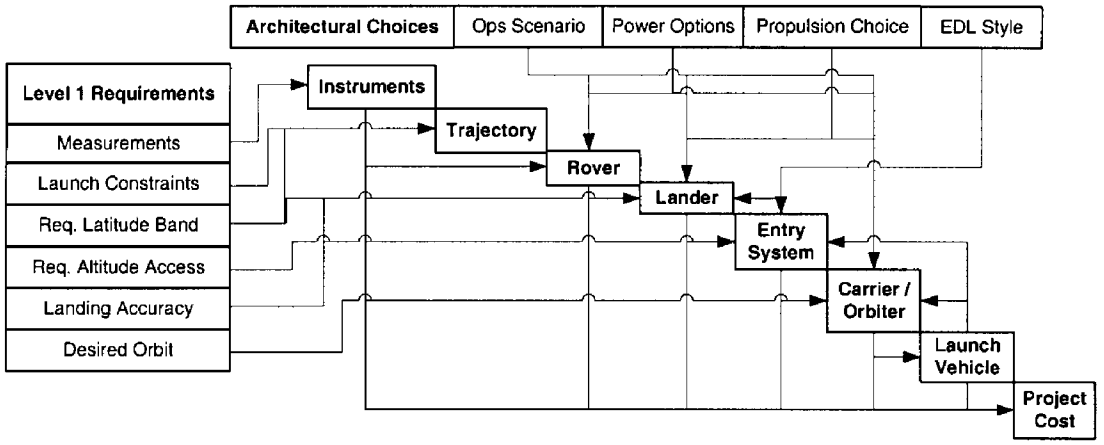


Figure 8. Mars Project-Level Model Interdependencies

2. On Modularity, and Model Flexibility versus Optimization

When generating models with broad applicability, model flexibility is the number one priority, which the choices of model architecture, model design and software platform very important.

To ensure maximum re-usability and generality, all models need to be built in a modular fashion. By developing a suite of compatible modular units, various trade questions can be addressed by assembling various models in different ways.

For ease of review, re-use, change and improvements, but also to improve the knowledge-gain in understanding the trade space, it is very important that the model be totally transparent to all. All model assumptions, including all numerical constants in a sizing rule-of-thumb and all automated logical design decisions need to be visible to the developer as well as all users and reviewers, and very easy to change. Providing an override function for any parameter of the model that doesn't result directly from the laws of physics makes it possible to immediately capitalize on any more accurate result, or on any better expertise found in the room. It also provides a convenient basis for "what if?" analyses.

To maximize the flexibility in assembling models, the capability to easily change what parameters are inputs and outputs of a given model is very useful. A simple example would be an orbits model, which could have period as input and semi-major axis as output, or the opposite, depending on what matters most to the measurement objectives. Even better flexibility is provided if even unforeseen inputs, or new options, can be added. This flexibility can be facilitated by a central parameters exchange feature. With direct link between models, all related models would need to be changed when changing what parameters are inputs and outputs of a model. Instead, there can be one central repository of design parameters where all models deposit their outputs; they all get their inputs from this same place as well, no matter where the parameter came from.

High performance computer languages, together with algorithms tailored to optimize the speed of iterative calculations in the design, would help build robust, high-precision, high-turnaround models. However, flexibility considerations, for reasonably simple calculations, led us to choose a spreadsheet-based architecture instead. A good spreadsheet design has self-documenting assumptions, and editing the documentation immediately updates the model to reflect the changes. Total user override capabilities are straightforward. Sharing models, and transitioning models from design teams to customer teams, is made easier by using a software environment familiar to all. Better software environments that could provide the same features will be interesting to research.

Appendix: Definition of Terms

There is the need for a common language for the model-based engineering community players (modelers, mission architects and designers, managers, software programmers) to understand each other. We did not attempt to propose a consensus lexicon. For the sake of clarity however, the following definitions hold within the paper:

Model

The terms "spacecraft model" has a variety of meanings, depending on the type of model it refers to. It could be a piece of hardware, a subscale version of the physical spacecraft (*Physical Model*). It could be a diagram representing the functional relationships between the various subsystems of the spacecraft (*Functional Model*). It

could be a dynamic document capturing the relationships between all the system requirements (*Requirements Model).* It could be a set of equations that simulate the behavior of the system (*Analytical Model).* All types of models are important at one part or another of the design cycle, but this paper focuses on one class of models: for the purposes of this paper, we use the term *Model* to mean *Analytical Model.* In this definition, a *Model transforms* information (Inputs into Outputs) using logical relationships and equations that represent the behavior of the system being modeled. We are interested in two main types of analytical models:

Design Model

Design Models transform system requirements and qualitative design choices into system design parameters. For example, the solar array area is a function of the total power required for all spacecraft subsystems, and the choice of solar cell chemistry.

Simulation Model

Simulation Models, or *Performance Simulation Models*, calculate system performance based on system design parameters, operational scenario(s), and a chosen set of performance metrics. For example, *orbit simulation* software packages calculate the planet coverage statistics of a given spacecraft based on orbital parameters, and characteristics of the instrument field of view. Similarly, a *landing hazard model* would calculate the probability of hazardous landing based on a description of the surface features a Lander is designed to survive, such as maximum rock size and maximum slope.

Tool

The word *Tool* is even more general than *Model.* We will refer to *Tools* to mean *Decision Support Tools*, to be distinguished from models in the following way: *Tools organize* information in a way that facilitates human understanding, analysis, and decision making. They do not *add* informational content. Multi-dimensional plotting and visualization tools are obvious examples of decision support tools.

With these definitions, a *Modeling Tool* is a tool that facilitates building, organizing and running Models.

Level of Detail, Level of Resolution

The *Level of Detail* of a model refers to the depth of its Inputs and Outputs. Since we are concerned with space mission models, the level of detail will refer to the following hierarchy:

Enterprise $\supseteq$ Program $\supseteq$ Project $\supseteq$ Mission $\supseteq$ Flight & Ground Systems $\supseteq$ Spacecraft $\supseteq$ Subsystem $\supseteq$ Component

Thus, a Mission Design Model would take requirements at the mission level as inputs, such as destination planet, launch date, and measurement objectives; and return design parameters at the mission level as main outputs, such as launch mass, launch vehicle, or mission cost. Such a model can be used to trade off various measurement requirements against the mission cost. Of course, in order to translate inputs into outputs, the model is likely to involve sub-models at lower level of details, in this case, probably down to the spacecraft or subsystem level. It would be able to provide secondary outputs at that lower level. This can be referred to at the *Level of Resolution* of the model. We would then speak, for example, of a *Mission-Level Design Model at Subsystem-Level Resolution.*

Model Uncertainty, Model Error

There are a number of reasons why the results from a model might not match the final parameters of a flown space mission. When building models, it is important to be aware of these sources of error and their distinction.

Inherent uncertainty in a parameter is uncertainty that no level of modeling can reduce, because the parameter value can still be changed by future events, or because it relates to a phenomenon that (to our best knowledge) is stochastic by nature. This is the case, for example, of some spacecraft environment characteristics, such as radiation hazard caused by solar flares, or the winds in the atmosphere of Mars upon arrival; in most cases, the extent of this uncertainty is estimated, and can be accounted for with increased design margins. Also, there will usually be significant inherent uncertainty in project cost, no matter how sophisticated a cost model is used, because ultimate project cost depends on many human and organizational factors that are independent of the technical description of the mission, and likely to fluctuate during the course of the project. Cost reserves are held to account for this uncertainty. Inherent uncertainty is typically systematic; it will affect all model results with about the same magnitude.

We call *study uncertainties*, affecting design as well as requirements, the uncertainties that relate to the level of immaturity of the study, and at what level of detail it stopped. As design studies progress, the details of the design will be developed, some new system-level effects will be discovered, and new ideas might be found that can result

in design changes. These uncertainties are reduced by more detailed modeling and analysis on the design, and more refined decision making on the requirements.

By contrast with these sources of uncertainty, we call *model errors* the errors resulting from any deliberate approximation made in the model calculations, for the purpose of simplicity and speed of calculation

Study uncertainties and model errors are related, if only because they together result in the *model deviation*, which we can define as the standard deviation of the model results compared to the results that would be obtained using the most accurate available means (the "actual" numbers but for inherent uncertainties). The distinction between study uncertainties and model errors is important because it can provide a useful guide as the appropriate level of approximation for modeling. The model results are only as accurate as their "weakest link".

Finally *Relative error* and *Relative deviation* refer to the error in the relative model results using different outputs. It is the standard deviation of the relative difference between two designs as predicted by the model.

Acknowledgments

This work was performed at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration. Significant contributions were made by Bobak Ferdowsi, Gene Lee, Joseph Mrozinski, Corey Harmon and Jaret Matthews, from the Jet Propulsion Laboratory (JPL); Eric Mahr, on contract with JPL from the Aerospace Corporation; and undergraduate summer interns Alice Quan, Justin Smith and Lori Ziegler under a NASA Grant fellowship.

References

[1]Wall, S.D., Smith, D.B., Koenig, L.J. "Team structures and processes in the design of space missions" *1999 IEEE Aerospace Conference*, Aspen, CO; 6-13 March 1999, pp35-42, 1999

[2]Multidimensional visualization and its application to a design by shopping paradigm. Stump, G; Simpson, T W; Yukish, M; Bennett, L. *9th AIAA/ISSMO Symposium and Exhibit on Multidisciplinary Analysis and Optimization*, Atlanta, GA, 2002. AIAA Paper 2002-5622

[3]Chase, James et al. "System, Cost, and Risk Analysis for Access to Space." *Space Systems Engineering and Risk Management Symposium Proceedings*, Cambridge, Massachusetts, 2004.

[4]16.89 MIT Space Systems Class Design Document, "Rapid Modeling of Mars Robotic Explorers", Massachusetts Institute of Technology, 2003.

[5]Girerd, André R. and Robert Shishko. "Project Trades Model for Complex Space Missions." *AIAA Space 2003 Conference and Exhibit*, Long Beach, CA, 2003. AIAA Paper 2003-6325

[6]Min, I., Mahr, E., Vaughan, R., Kellogg, I. "Rapid Space Architecture Development and Analysis Using an Integrated Toolset". *2004 IEEE Aerospace Conference*, Big Sky, WY, Mar 6-13, 2004.

[7]Wall, S.D. "Model-Based Engineering Design for Space Missions". *2004 IEEE Aerospace Conference*, Big Sky, WY, Mar 6-13, 2004.

[8]McManus, H.L., Hastings, D.E. and Warmkessel, J. M. "New Methods for Rapid Architecture Selection and Conceptual Design" *Journal of Spacecraft and Rockets*. Vol 41 No1, pp10-19, January-February 2004

[9]Wertz, J. & W. Larson. *Space Mission Analysis and Design (SMAD)*. Space Technology Library, El Segundo, CA, 1999.

[10]Ross, A. M., Hastings, D. E. and Warmkessel, J. M. "Multi-Attribute Tradespace Exploration as Front End for Effective Space System Design." *Journal of Spacecraft and Rockets*. Vol 41 No1, pp20-28 January-February 2004