A methodology for system-of-systems design in support of the engineering team

G. Ridolfi [a,b,*], E. Mooij [b], D. Cardile [a,c], S. Corpino [a], G. Ferrari [c]

[a] *Politecnico di Torino, Corso Duca degli Abruzzi 24, Torino 10129, Italy*
[b] *Delft University of Technology, Kluyverweg 1, Delft 2629HS, The Netherlands*
[c] *Thales Alenia Space, Italy, Strada Antica di Collegno 253, Torino 10146, Italy*

ABSTRACT
Space missions have experienced a trend of increasing complexity in the last decades, resulting in the design of very complex systems formed by many elements and sub-elements working together to meet the requirements. In a classical approach, especially in a company environment, the two steps of design-space exploration and optimization are usually performed by experts inferring on major phenomena, making assumptions and doing some trial-and-error runs on the available mathematical models. This is done especially in the very early design phases where most of the costs are locked-in.
With the objective of supporting the engineering team and the decision-makers during the design of complex systems, the authors developed a modelling framework for a particular category of complex, coupled space systems called System-of-Systems. Once modelled, the System-of-Systems is solved using a computationally cheap parametric methodology, named the mixed-hypercube approach, based on the utilization of a particular type of fractional factorial design-of-experiments, and analysis of the results via global sensitivity analysis and response surfaces.
As an applicative example, a system-of-systems of a hypothetical human space exploration scenario for the support of a manned lunar base is presented. The results demonstrate that using the mixed-hypercube to sample the design space, an optimal solution is reached with a limited computational effort, providing support to the engineering team and decision makers thanks to sensitivity and robustness informa-tion. The analysis of the system-of-systems model that was implemented shows that the logistic support of a human outpost on the Moon for 15 years is still feasible with currently available launcher classes. The results presented in this paper have been obtained in cooperation with Thales Alenia Space—Italy, in the framework of a regional programme called STEPS.1

Keywords:
System-of-systems
Space-systems models Concurrent design methodology Factorial design
Sensitivity analysis

1. Introduction

In the last decades space missions have become more complex and more articulated than before. This happened mostly because the attention of the space agencies has shifted towards new and more ambitious mission architectures. As space programs are developed and implemented, it becomes more and more evident that it is necessary to efficiently manage the increased complexity to decrease the System-of-System (SOS) cost.

☆ This paper was presented during the 61st IAC in Prague.
* Corresponding author Tel.: +39 0110906857.
E-mail addresses: guido.ridolfi@polito.it, g.ridolfi@tudelft.nl (G. Ridolfi), e.mooij@tudelft.nl (E. Mooij), diego.cardile@polito.it (D. Cardile), sabrina.corpino@polito.it (S. Corpino), GiorgioLuigi.Ferrari@thalesaleniaspace.com (G. Ferrari).
[1] STEPS—Sistemi e Tecnologie per l'EsPlorazione Spaziale is a research project co-financed by Piedmont Region and firms and universities of the Piedmont Aerospace District in the ambit of the P.O.R–F.E.S.R. 2007–2013 program.

Nomenclature

ANOVA	Analysis Of Variance
CAP	Capsule
CCD	Central Composite Design
Delta-V	Velocity change
DS	Departure Stage
EOR	Earth Orbit Rendezvous
ESAS	Exploration System Architecture Study
I_{sp}	Specific Impulse
LAM	Lunar Ascent Module
LDM	Lunar Descent Module
LEO	Low Earth Orbit
LLO	Low Lunar Orbit
LOI	Lunar Orbit Injection
LOR	Lunar Orbit Rendezvous
LS	Lunar Surface
LSAMs	Lunar Surface Access Modules
LTO	Lunar Transfer Orbit
SM	Service Module
SOS	System of System
TLI	Trans-Lunar Injection

A System-of-Systems is formed of several interacting elements and sub-elements whose overall behaviour is usually different from the sum of the effects of the single elements. The analysis and comprehension of these interactions are of crucial importance for the proper design of such kind of complex systems.

The currently accepted approach to space System-of-Systems design is to create a fragmentation by systems and subsystems, and to decompose the design tasks into discipline tasks. Doing so, artificial boundaries are introduced in the process and in the organization thus creating communication problems in properly exchanging data. The system-of-systems design is accomplished by assembling these separately designed elements, iterating or balancing between conflicting objectives. In this process, the scope of systems engineering is to make sure that the development process leads to the most cost-effective final product. Before every decision is made, especially for those that are hard to undo in an advanced phase, the alternatives should be carefully assessed, understood and discussed. This can be achieved only if there is an efficient communication between the various disciplines/systems/subsystems that determine the performance(s) of the system-of-systems, enabled by a proper integration of all the mathematical models that concur to the determination of the System-of-Systems model.

The problem of designing using highly integrated models is on an open debate among the research groups dealing with complex, coupled systems [1,2]. Different classes of methodology can in principle be used to solve a system-of-systems, [3,4], but the heuristic/meta-heuristic methods are those which are more widely applied. The main disadvantages of these methodologies are that they are *simulation intensive*, causing long run times, providing almost no insight in the problem of interest and leading to poor convergence of the solution in some cases.

We are investigating a different and possibly more effective way to deal with this kind of systems, based on parametric design. We think, according to what has been addressed in previous works, [5–7], that supporting the design team with graphical information, instead of providing a ready solution, with no clear insight in the behaviour of the single elements and in the interactions, is crucial for System Engineering processes and tools.

Thus, the main objective of the approach presented in this paper is to support designers and decision makers during all the design phases of a complex system by keeping the concurrency of the design process and providing information on the behaviour of the system and its components, in terms of interactions and sensitivity.

The system-of-systems used as a case study in this paper is obtained by coupling the mathematical models of systems and subsystems belonging to a hypothetical human mission to support the return of mankind on the Moon with a permanent outpost, [21]. In particular, the models that have been developed and implemented enable the design of a manned re-entry capsule, a service module for the capsule, a lander system, an ascent vehicle, and an Earth–Moon transfer vehicle.

The remaining part of the paper is organized as follows. In Section 2 the modelling framework and the main characteristics of the mathematical models imple-mented are described. In Section 3 the main aspects and potentials of the implemented design methodology are described. The results of the design iterations are described and commented in Section 4. Finally, in Section 5, conclusions and final remarks are discussed.

2. Manned lunar-exploration system-of-systems

The design of a space exploration system-of-systems requires the definition of a mission architecture together with the mission concept stating how the mission will work in practice, and defining all the elements, i.e., the building blocks, that will take part in it. The mission concept includes also such issues as the synergies of manned and robotic resources, mission control, and the mission timeline.

Considering all the possible combinations of people, orbits, launch systems, space vehicles, surface facilities, and supporting infrastructures we end up with a large number of possible mission architectures. Obviously, not all of them are optimal, or even feasible. Thus, the very first part of the analysis is to reduce the mission architectures to a small number that will be developed in more detail.

This activity is usually performed at the highest levels of space agencies, mainly based on the inherent social, political, and economical situation. A very interesting example of this activity is the Exploration System Architecture Study (ESAS) by NASA, [11]. The mission architecture design activity goes beyond the scope of the paper so we refer to the recommended mission architecture for Moon exploration described in Ref. [11], to focus on the development of few building blocks. The ESAS document is aimed at defining the top-level requirements and the configurations for manned and cargo elements to develop an exploration architecture concept, with the key technologies required to support sustained human and robotic lunar exploration operations. The reference architecture is studied to ensure global access to the Moon, i.e., the possibility of transportation of crew and cargo to and from anywhere on the lunar surface. The mission is very similar to that of Apollo's for what concerns the surface activities but differs from the Apollo missions in the location of the "*nodes*", i.e., the positions where a docking or a separation of modules occurs [12].

The most relevant building blocks considered for this mission architecture are listed in Table 1.

The capsule is the vehicle capable of transporting and housing crew from Low Earth Orbit (LEO) to Low Lunar Orbit (LLO). The Service Module (SM) is an unpressurized system that provides propulsion, power and other supporting capabilities for the capsule. The Lunar Ascent Module (LAM) and the Lunar Descent Module (LDM) are docked together forming the so-called Lunar Surface Access Module (LSAM). The LAM is the module that supports the crew members from LLO to the lunar surface and back from the lunar surface to LLO. The LDM is an unpressurized module that performs the descent manoeuvres from LLO. It provides life support for the crew members and power generation during the lunar activities. The Departure Stage (DS) is a propulsion module that provides the necessary thrust to leave the LEO and inject the payload, formed by all the other modules docked together, into the Lunar Transfer Orbit (LTO).

In Fig. 1 a schematic of the mission architecture is shown.

The recommended mission architecture is a two launch, EOR–LOR, mission mode. The first launcher puts the main propulsion stage and the lunar access modules in orbit while the second launcher puts the capsule and the service module in orbit. For this motivation two main mission nodes are considered, LEO and LLO. The first one is near the Earth, where the rendezvous and docking of the capsule and the service module with the main propulsion stage and the lunar access modules is performed. The second one is in the cis-lunar space. In particular, in the low lunar orbit the lunar surface access module undocks from the capsule and lands on the Moon. In a subsequent phase the lunar ascent module comes back from the lunar surface and docks whit the capsule.

Table 1
Description of the main building blocks with relative delta-V manoeuvres of the space scenario.

Description	Acronym	Symbol	ΔV[11]
Capsule	CAP		50 [m/s]
Service module	SM		1724 [m/s]
Lunar ascent module	LAM		1888 [m/s]
Lunar descent module	LDM		1900 [m/s] 1390 [m/s]
Departure stage	DS		3120 [m/s]

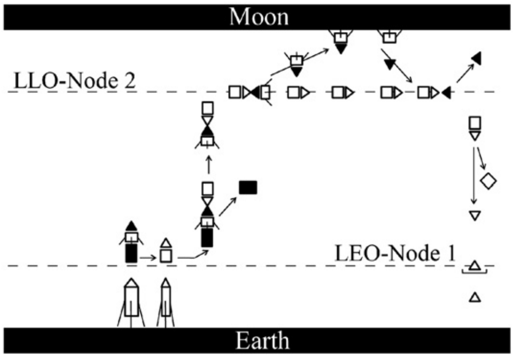

Fig. 1. Schematic of the reference mission architecture.

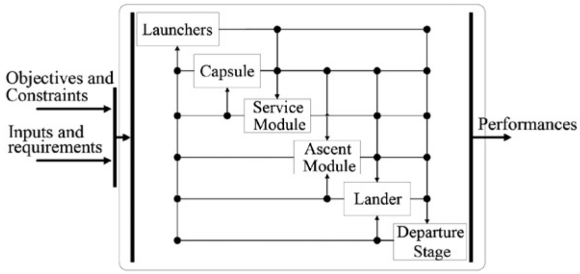

Fig. 2. System-of-Systems N^2 chart.

At system-of-systems level, multiple interactions amongst the building blocks are experienced. All the relationships and the interactions amongst the elements of the system-of-systems, which are then translated into mathematical coupling between the models, are schematically shown in Fig. 2. The blocks represent the mathematical models of the systems, while feed-forward and feed-back lines indicate a flow of data among the elements that are coupled together. The presence of feedbacks in the modelling framework forces the process to iterate before reaching convergence. Convergence is obtained once the state-variables of the system-of-systems are in equilibrium, i.e., do not change for successive iterations. This *non-hierarchical* decomposition of the mathematical models provided by the modelling framework, allows for a certain degree of flexibility in the modelling activity, [8–10]. The blocks can easily be substituted with more or less accurate models providing only the necessary interfaces with the other blocks of the system. At a lower level, thus within every block of Fig. 2, the mathematical modellization has been com-pleted with the same non-hierarchical decomposition approach.

The SOS design process needs to balance the performances to meet the constraints and fulfil the requirements. In the case of the Moon base support mission, explained in detail in Section 4, some of the SOS constrains and requirements are dictated by the Moon base characteristics. The operational life of the Moon base (t_{base}), the number of crew-members on the Moon (n_{crew_base}) and maximum period of time that astronauts can spend on the Moon (t_{max}) influence the total number of manned launches that must be performed to properly support the Moon base (n_{launch}):

$$n_{launch} = \frac{t_{base} n_{crew\text{-}base}}{t_{max} n_{crew}} \quad (1)$$

The number of crew members on the capsule (n_{crew}) is a very important design variable that influences many SOS performance factors, e.g., masses and geometries.

The modellization of the launchers is based on a user-customizable database where all the main characteristics are available, e.g., mass on orbit and volume availability. This model takes information about the mass and the geometry of all the other system-of-system modules as input and provides the maximum payload mass and available volume to the other SOS building blocks.

A very strong iteration loop exists amongst the mathematical models of the capsule and the service module. The capsule model provides to the model of the service module data about the interface geometry, external layout, electrical power, and thermal dissipation required. A similar interaction loop exists also between the model of the ascent module and the lander model. Between these two models there is a data-flow of interface geometry, external layout, electrical power, thermal dissipation required, and the amount of necessary oxygen and nitrogen to support the astronauts' life. The total amount of air is determined also by other system-of-system requirements and characteristics, i.e., the amount of pressurized volume, the number of re-pressurization cycles, the number of crew members, and the mission duration. Finally, the mass of all the building blocks is used by those systems providing thrust, i.e., the lander, the departure stage, and the service module, to properly compute the needed propellant and thus their mass.

To verify the goodness of the implemented SOS mathematical models, a simulation was performed to compare the outcome with data available in the literature [11]. The results of the comparison are shown in Table 2. There are relatively small differences between the calculated masses and the reference data.

Table 2
SOS model results compared to ESAS data.

	Total system mass		
	SOS model [kg]	ESAS [kg]	Δ
Capsule	10,442	9506	9%
SM	14,698	13,647	7%
LAM	9791	10,809	10%
LDM	35,406	35,055	1%
DS	120,263	227,250[a]	89%

[a] The DS mass evaluated in Ref. [11] is retrieved from different design assumptions: the ESAS DS performs three manoeuvres (LEO insertion, circularization and TLI burn); the DS modelled performs a single manoeuvre.

3. Mixed-hypercube approach in support of the engineering team

One of the main objectives of the engineering team during the design process of systems of any complexity, thus also of a system-of-systems, is to predict its behaviour in the operational environment, and to set all the physical and functional characteristics of the system such that it performs as required [22]. The cheapest and maybe fastest way to predict the behaviour of a system is to create a (n analytical) model and to extract information about its performance by executing multiple experiments with different levels (=values) of the design variables, i.e., the input to the model. There are many methods that can be used to extract information from an engineering mathematical model.

Deterministic methods (with multiple or single objective implementation) strongly rely on gradient-based techniques for exploring the design space in the search for the optimum [13]. Therefore first-derivatives and, sometimes, second derivatives of the mathematical equations that describe the physical phenomenon of interest are required. The derivatives are used to determine the search direction in the design space that should lead to the optimum. The gradient-based methods perform very well for problems in which the mathematical functions do not present disconti-nuities or plateaus, i.e., flat regions in the design space and have only a single optimum. They are not applicable when architectural, i.e., discontinuous, variables are present, as in the case presented in this paper.

Heuristic methods, such as those based on evolutionary strategies, have recently become very popular. They empirically demonstrate the advantage of reaching satisfactory results even in the presence of problems with discrete variables (thus without use of gradient information) [4,14]. One of the drawbacks of these methodologies, if observed from the perspective of the engineering team of a company, is that the analyses are not fully traceable, due to the partial random nature of those methods. Further those methodologies usually require computationally intensive processes to reach an optimal solution that has no practical value per se, due to the lack of sensitivity information.

Parametric Design is a deterministic method that can be used to provide the engineering team with information on the sensitivity of the factors on the responses of interest, without explicit use of derivatives and in a fully traceable way. A parametric design is often called factorial design. The term "*factors*" is used to address the design variables, while the term "*responses*" indicates the output variables (the objectives) that describe the performance of the system. Performing a factorial design means that the mathematical model of the system of interest is evaluated for all (or a subset in case of a fractional factorial design) combinations of factors levels. The levels of a factor are the values that the factor can assume, e.g., a minimum, nominal and maximum value. The advantage of using a parametric design approach is that the simulations are deterministically designed allowing for traceability of the results. Further, as we will see later in this section, if the settings of the design variables are well arranged the number of simulations needed to extract significant information from the model, e.g., sensitivity and robustness information, is significantly reduced if compared to a classical stochastic technique like the Monte Carlo analysis for instance [15,16].

The easiest way of performing a parametric design is to consider a *full factorial* design. Full factorial designs require the largest number of design-variable combinations to be evaluated, since all combinations of variables levels are tested. Consider, for instance, three design variables at two levels each. According to a full factorial design the total number of design points is 2^3 (=8), as shown in Fig. 3(top). The other class of factorial designs is called *fractional factorial*. In this case the required data points are only a fraction of those required by a full factorial design. There are three main subclasses of fractional factorial designs, namely Resolution III, Resolution IV, and Resolution V. From Resolution III to Resolution V the number of required experiments increases, thus providing more information on the behaviour of the problem in the design region of interest, [18]. I n Fig. 3(bottom), we show a Resolution III fractional factorial design that presents the least number of simulations needed with 3 factors at 2 levels. As can be seen, already with only 3 factors the required number of design points is half compared to the relative full factorial design.

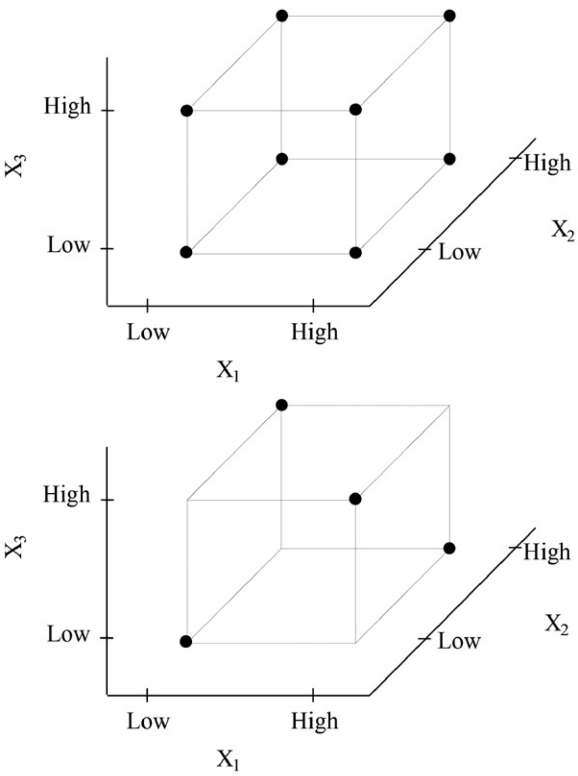

Fig. 3. Full factorial design: three factors at two levels (top). Fractional factorial design (Orthogonal Design): three factors at two levels (bottom).

Besides significantly reducing the computational effort in exploring the design space, fractional factorial designs enable an efficient determination of the effects of the design variables (or control parameters) on the outputs,(or performance parameters), due to their inherent symmetry in selecting the design-variable levels [17].

Whether to use a full factorial design or one of the fractional factorial design classes depends on many aspects. Low-resolution fractional factorial designs require fewer points to be sampled thus allowing for a faster experimentation (lower computational effort). However, in these cases less information on main effects and interactions can be analyzed [18]. High resolution fractional factorial designs, or full factorial in the extreme case, are advised when no *a-priori* information is available on the model of interest, so that fewer assumptions can be made regarding which interactions are negligible to obtain a unique interpretation of output data.

Two-level factorial designs allow for the determination of linear main effects and linear interaction effects. There are many cases in which the curvature of the design space is very important, especially when optima are inside the design space of interest instead of being localized on the border. When curvature is present, the two-level factorial design does not provide reliable results, since only limited curvature is detectable by the two-factor product terms. The second-order models of the following form are the most widely used, typically for most engineering problems [18,19]:

$$Y = \beta_0 + \sum_{i=1}^{k} \beta_i x_i + \sum_{i=1}^{k} \beta_{ii} x_i^2 + \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} \beta_{ij} x_i x_j \qquad (2)$$

β_0 is the mean response of the system, β_i are the coefficients for the linear main effects, β_{ii} are the coefficients for the quadratic effects, and β_{ij} represent the coefficients for the linear interactions.

In general, a second-order model like the one described in Eq. (2) will not represent a reasonable approximation on the whole design space, but for rela-tively small regions the results are usually very satisfac-tory. If this is still not the case, lack-of-fit will signal the need for a higher-order model, thus an increased number of experiments to be performed [18]. The discussion in this paper is based on Eq. (2), but it does not lose it general validity for higher-order models.

There are many designs, which allow for fitting a second-order regression model: Central Composite Design (CCD), three-level factorial design, Box-Behnken design, D-optimal design, and so on. The CCD is the most widely used class of factorial designs used for identifying second-order models. We choose the CCD also because of its heritage from the two-level factorial design [18,19]. T h i s particularly efficient design is formed by a two-level full factorial design for linear effects, plus axial points and a central point for curvature effects [19]. In this paper we implement the CCD by substituting its full-factorial part with a fractional factorial design, for reducing the computational cost of the analysis, as previously discussed. Adopting a fractional factorial structure for the non-axial points, will allow to proceed with an incremental approach in the planning and simulation of the experiments. Starting from the lowest possible design, Resolution III, experi-ments may be added if lack-of-fit results higher than a certain user-defined threshold. In Fig. 4, a full factorial CCD (top) and a CCD obtained using a fractional factorial (bottom) is presented.

Once the design space has been sampled using a CCD, the ANalysis Of VAriance, ANOVA, and variance decomposition are implemented to identify the relative effect of the different factors, and their interactions, on the overall variability of the performance detected during the simulations. To do so, the overall variance is partitioned in its components determined by the effect of the factors and the included interactions. The larger the contribution to the overall variance the larger is the effect on the analyzed performance.

Suppose we obtain a performance vector $\boldsymbol{y}=[y_1,y_2,\ldots,y_n]$, with n equal to the number of experiments, as a result from the simulations with the variable settings planned with the CCD. The total variation of the performance y, also called the total sum of squares, can be computed as follows:

$$SS_T = \sum_{i=1}^{n} (y_i-\overline{y})^2 \tag{3}$$

with $\overline{y} = \sum_{i=1}^{n} y_i$ being the mean response. Consider $SS(A)$ as the sum of squares associated with factor A, and $SS(A|B)$ as the sum of squares of the factor A given that factor B is already in the model, i.e., the variability added by factor A to the total variability computed with only factor B in the model. With this notation, the partial sum of squares of a factor C can be computed as follows:

$$SS(C) = SS(C|A,B,AB) = SS(A,B,C,AB) - SS(A,B,AB) \qquad 4)($$

Eq. (4) gives an indication of the change in variability of the data due to adding an extra term to the model, given that all the other terms have been added except for the terms that contain the effect under test. It provides a clear indication of the effect of excluding or including a term into the model. The partial sum-of-squares associated with each factor is a global measure of sensitivity since it is related to the variance. It is valid over the entire range of variation of the design variables. The regression coefficients could be used as sensitivity indices, but for more-than-linear terms they would provide only a local measure of sensitivity, [20].

However, the coefficients β_0, β_i, β_{ii} and β_{ij} that appear in Eq. (2), obtained using a least-squares interpolation, can be used to compute response surfaces and contour plots. For the presentation of the results and to the engineering team this is a fundamental step in the methodology. The results from the ANOVA on factor importance, allow the engineering team to focus only on the design region spanned by the most relevant factors concerning the performance of interest, thus saving time in the design process. Response surfaces are very power-ful in presenting the shape of the design space for fast modification of the design-variable settings and neighbourhood analysis of the design-point conditions. Further, contour plots allow for a fast and effective boundaries and constraints analysis, even with more objectives.

In the title of this section we introduced "*mixed hypercube*" as the name we gave to the design methodology presented in this paper to support the engineering team in the design of a system-of-systems. The term *hypercube* is used because when the design factors are more than three, the geometrical representation of all the design dimensions (with each one being a design variable) is a hyperspace. When we put boundaries to these dimensions we obtain a hypercube. The term *mixed* is mentioned, because we can use both continuous and discrete (nominal or architectural) design variables. The need of separating the design variables in these two classes arises from the fact that there are no intermediate levels that the discrete variables can assume. Therefore, for each design combination of architectural variables, a CCD analysis is performed on the continuous variables providing information on multiple objectives, trends and

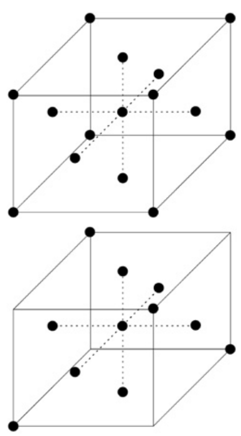

Fig. 4. Central Composite Design (CCD), including a full factorial (top), and a fractional factorial design (bottom).

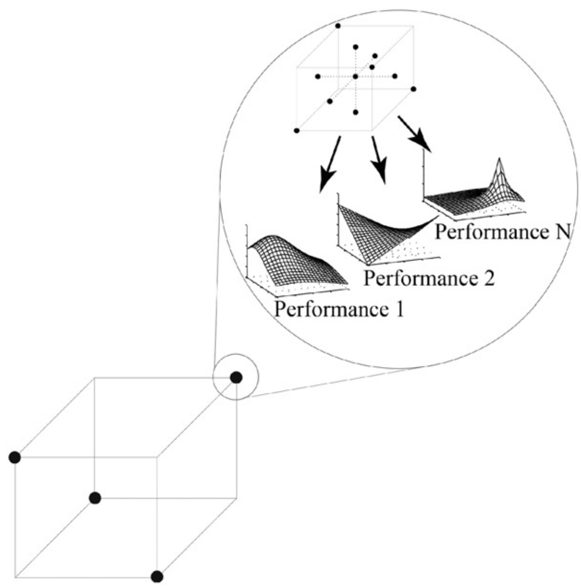

Fig. 5. Hypercube design with multiple performances and mixed continuous-discrete variables.

shape of the design region, best settings of the design variables, robustness and sensitivities, as shown in Fig. 5.

The lower left hypercube of Fig. 5 represents the fractional factorial design used for the discrete variables. For each point of that hypercube, another hypercube is build using the CCD with the continuous variables only.

4. System-of-systems performances

A hypothetical mission scenario to support a manned lunar outpost for a minimum of fifteen years has been designed using the modelling framework and the mixed hypercube approach described in the previous sections. The mission scenario supposes a certain number of astronauts that must be periodically transported to and from the Moon.

4.1. Design settings and analysis

The objective of the study is to minimize the number of manned launches to support the lunar outpost for the required time while minimizing the dimensions and the mass of the capsule and the service module. Cost models for such systems are not available in literature. However we consider minimizing the number of launches and the mass of the capsule and the service module reasonably similar to minimizing the cost of the mission as a whole, at least in a first approximation. In the future, a more detailed analysis will be performed to include relationships between mass, technology level and cost in the design process. In Table 3, the requirements and the design variables' levels taken into account as a baseline design are summarized. These settings are similar to those considered in Ref. [11]. The baseline design repre-sents the first tentative design-variable set used to solve the problem of minimizing the objectives while not violating the constraint. The baseline design leads to the following performances. The mass of the capsule and the service module is equal to 24 t and the number of (manned) launches to support the lunar base for 15 years is 113. The values of the design variables and require-ments have been assigned as a first guess in order to begin with the design process.

Table 3
Settings of the design-variables and requirements for the baseline design.

Requirements	
Lunar outpost operational life	15 [years]
Maximum time of the crew in the outpost	0.2 [years]
# Crewmembers in the outpost	6
Design variables	
Capsule crewmembers comfort level Capsule mission	2
duration	13.5 [days]
# Capsule crewmembers (n_{crew})	4
Capsule sidewall angle	32.5 [deg]
Capsule I_{sp}	274 [s]
Service module I_{sp}	364 [s]
Lander system I_{sp}	435 [s]
Ascent module I_{sp}	364 [s]
Transfer vehicle I_{sp}	451 [s]
Launcher class	25 [tons]

In Table 4, the ranges of the design variables are described and the nomenclature used in the graphs shown later in this section is indicated.

The experiments designed with the mixed hypercube, considering the discrete and continuous variables indicated in Table 4, have been analyzed with ANOVA. The total number of simulations is 145. This allowed for a reduction of the computational effort when compared to a full factorial design. Indeed, a full factorial design with 8 design variables at three levels would have required 6561 ($=3^8$) simulations. Regression analysis provided the regression coefficients of the second-order model with a negligible lack-of-fit: the coefficient of multiple determination is equal to 0.98. This means that no more experiments need to be added and that the model in Eq. (2) represents a reasonable approximation.

The factor contribution to the performances and constraint violation, i.e., sensitivity of performances and constraints to the design factors and their interactions, are shown in Figs. 6–8.

In these figures only the relevant factors and interactions are shown. When the same letter is repeated twice, it indicates that we refer to a quadratic effect of that factor. When two different factors are indicated, we refer to the effect of their interaction. The column labelled "*Other*" represents the sum of all the linear, quadratic, and interaction effects that are not specifically shown.

As we can see in Fig. 6, the number of (manned) launches is most affected by the lunar base operational life, that is a requirement, and by the crewmembers that the capsule can host. Also their interaction contributes to the determination

Table 4
Design variables taken into account in the design process. Type A: Architectural (discrete) variable. Type C: Continuous variable.

Design variables	Factor	Type	Levels		
			Level−1	Level 0	Level 1
Launcher Class	*A*	*A*	25 [tons]	50 [tons]	75 [tons]
Outpost operational life	*B*	*A*	10 [years]	15 [years]	20 [years]
# Capsule crewmembers (n_{crew})	*C*	*A*	3	4	5
Capsule crewmembers comfort level	*D*	*A*	1	2	3
Capsule I_{sp}	*E*	*C*	200 [s]	264 [s]	375 [s]
Capsule sidewall angle	*F*	*C*	28 [deg]	32.5 [deg]	37 [deg]
Service module I_{sp}	*G*	*C*	200 [s]	350 [s]	500 [s]
Ascent module crewmembers comfort level	*H*	*A*	1	2	3

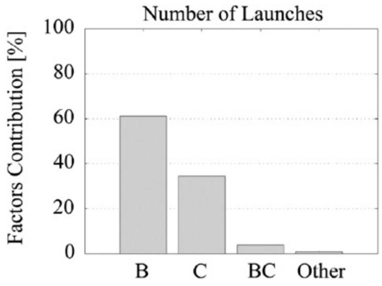

Fig. 6. Factors' contribution to the number of (human) launches.

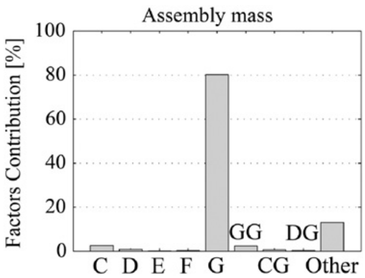

Fig. 7. Factors' contribution to the mass of Capsule plus Service module.

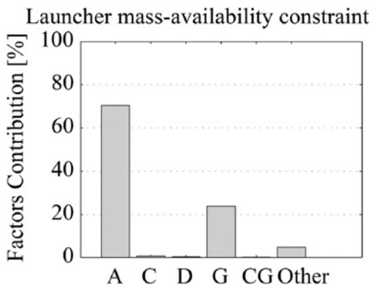

Fig. 8. Factors' contribution to the launcher on-orbit mass availability constraint violation.

of the performance, but with a reduced importance. This means that the number of launches needed to support the lunar base is very sensitive to the duration of the nominal operational life of the base itself and quite sensitive to the dimensions of the capsule. Also the launcher class, thus the maximum payload capability of the launcher, does play an important role. Indeed, in Fig. 8 we see that the launcher class affects the constraint violation up to 70%. The con-straint is violated each time that the capsule/service module assembly mass exceeds the launcher payload capability. In Fig. 7, the capsule/service module assembly mass is considered. We can observe that the design variables directly linked to the design of the capsule contribute the most, together with the specific impulse of the service module propellant. The specific impulse of the capsule does not contribute much to the mass since the ΔV to be delivered by the capsule itself is much lower than the ΔV to be delivered by the service module (in the simulation we used 50 m/s for the capsule and 1700 m/s for the service module). To gain more insight in the behavior of the system-of-systems, the information gathered with ANOVA and variance decomposi-tion is used for the regression analysis to plot the response surfaces of the performances and constraints as a function of the most relevant parameters.

In Figs. 9 and 10, the trends of the number of launches needed to support the outpost, as a function of the outpost operational life and the maximum number of crewmembers that can be hosted in the capsule is shown, for different levels of the remaining design variables. The response surfaces involving architectural variables do not have physical relevance, since only few of the points on the surface are valid. However, they are still very useful to understand the trends of the performances and to provide visual information to the engineering team.

In Fig. 9, we can read that with the operational life of the outpost that increases and the number of crewmem-bers hosted in the capsule that decreases, the number of launches increases consequently. This is probably an expected result, given the problem of interest. Maybe the mild interaction between the two parameters identi-fied with the variance decomposition (see Fig. 6) and corroborated by the trends of Fig. 9 is a less trivial result. Indeed, it can be clearly read that the sensitivity to variations of factor *B* increases when factor *C* decreases.

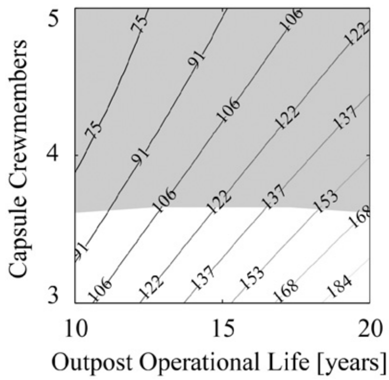

Fig. 9. Number of launches as a function of the outpost operational life and number of capsule crewmembers. The other parameter levels are as follows $A_{\mathrm{Level}-1}$, D_{Level1}, $E_{\mathrm{Level}-1}$, F_{Level0}, G_{Level0}, H_{Level0}. The grey area is infeasible.

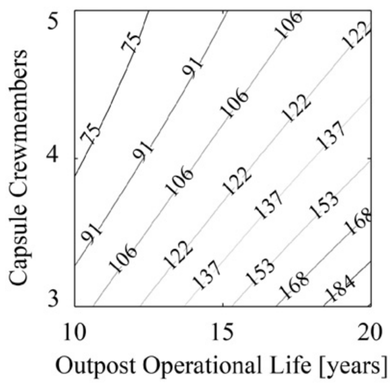

Fig. 10. Number of launches as a function of the outpost operational life and number of capsule crewmembers. The other parameter levels are as follows A_{Level0}, $D_{\mathrm{Level}-1}$, E_{Level1}, F_{Level0}, G_{Level1}, H_{Level0}.

This is an indication of the presence of an interaction between the two parameters. The influence of the number of crewmembers increases as the outpost operational life increases, i.e., when the total number of astronauts to be landed on the Moon increases.

The grey area in Figs. 9–13 represents the infeasible region of the design space. In that region the assembly mass, i.e., capsule mass plus service module mass, exceeds the mass deliverable by the selected launcher.

The fact that the other design parameters do not affect this performance much is confirmed by the trends shown in Fig. 10. These trends have been obtained as a function of factor *B* and factor *C*, with the other factors at a level that increases the size of the feasible region, especially because factor *G* is increased. The trend of the perfor-mance is not significantly different when compared to Fig. 9, and this result was already anticipated from the factor-importance analysis, see Fig. 6. The most relevant consequence is related to the fact that the launcher class is set to 50 t in Fig. 10, thus causing the shift of the design region into an area that is far from the constraint.

In Figs. 11 and 12, the trend of the capsule and service module assembly mass, expressed in kg, is shown as a

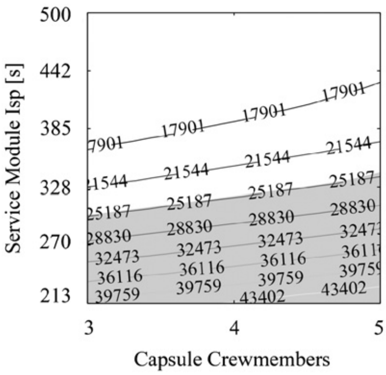

Fig. 11. Capsule and service module assembly mass [kg], as a function of the service module propellant I_{sp} and number of crewmembers hosted in the capsule. The other parameter levels are as follows $A_{\text{Level}-1}$, $B_{\text{Level}-1}$, $D_{\text{Level}-1}$, E_{Level0}, F_{Level0}, H_{Level0}.

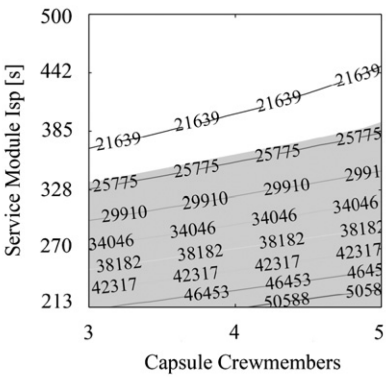

Fig. 12. Capsule and service module assembly mass [kg], as a function of the service module propellant I_{sp} and number of crewmembers hosted in the capsule. The other parameter levels are as follows $A_{\text{Level}-1}$, $B_{\text{Level}-1}$, D_{Level1}, E_{Level0}, F_{Level0}, H_{Level0}.

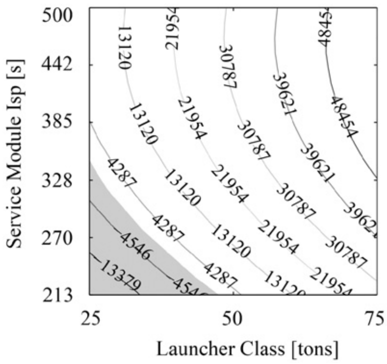

Fig. 13. Constraint as a function of the launcher class and service module propellant I_{sp}. The other parameter levels are as follows B_{Level0}, C_{Level0}, D_{Level0}, E_{Level0}, F_{Level0}, H_{Level0}.

function of the number of crewmembers hosted in the capsule and the service-module propellant I_{sp}, with the capsule crewmembers comfort level at $D_{\text{Level}-1}$=1 and D_{Level1} =3.

The results presented in Fig. 7 helped us to identify the design parameters to which this performance is most sensitive to capture most of the variability detected during the simulations. Factor *G* (=Service module I_{sp}) clearly dominates the performance if compared to the influence of factor *C* (=n_{crew}). In particular we read that with the I_{sp} that increases and the number of crewmem-bers hosted in the capsule that decreases, the assembly mass decreases.

The difference between Figs. 11 and 12 is attributed to the variation of factor *D* (=Crewmembers comfort level). Indeed, in Fig. 7 certain relevance in the determination of the assembly mass had already been discovered. In particular, with the comfort level that increases the assembly mass increases as well, moving the result closer to the constraint. With factor *G* (=I_{sp}) at its lowest level, there is no way to avoid violating the constraint by only acting on the number of crewmembers hosted by the capsule and the other design parameters set as specified.

The specific impulse of the service module propellant is much relevant concerning the determination of the constraint violation, but is not the only one, as can be observed in Fig. 8. Indeed, in Fig. 13, the constraint is plotted as a function of launcher class and service module propellant I_{sp}. What we see is that the only possible approach to obtain mission feasibility and not violating the constraint (while considering a hypothetical Ariane5 launcher class with 25 t of available mass), is to increase the propellant I_{sp}. I n Fig. 13 we also see that with increasing launcher-mass availability the sensitivity of the constraint to the propellant I_{sp} decreases, thus provid-ing more flexibility concerning this particular design choice. Of course, the launcher class is almost always a given reality. However the analysis showed that using the mixed hypercube approach and the variance decomposi-tion proposed in this paper, the engineering team has the possibility of identifying the impact of the requirements on the performances, to adjust the free parameters meet-ing the requirements and optimizing the performances.

Further, the fact that we considered the outpost operational life as a design variable, see Figs. 9 and 10, shows that this methodology can be used as a tool to negotiate the requirements. With considerations on affordability, reliability, flexibility, and cost in mind (not taken into account in the mathematical model, but supposedly clear in the engineering-team background) requirements can be adjusted in an informed way, considering their effect on the system-of-systems as a whole.

The combinations of design variables to be plotted in pairs to study the objectives and constraints are much more than those reported in the contour plots shown in this section, and they increase when the number of design parameters increases. The analysis of variance and its decomposition allowed us to reduce the number of graphs actually needed to plot all the design variables pairs to only a few, and still retaining most of the variability experienced during the simulations.

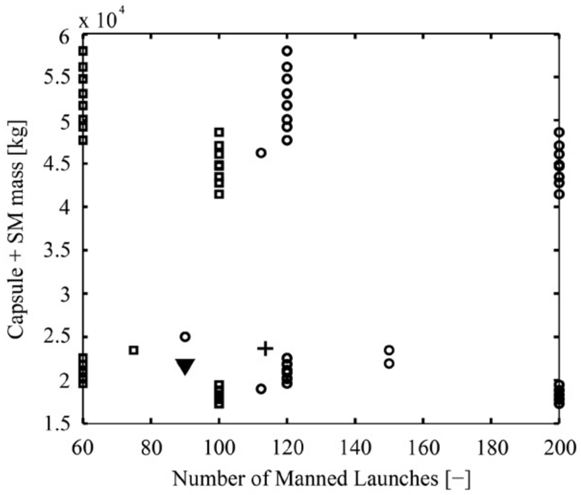

Fig. 14. Objective space, feasible solutions (□ Feasible solutions with 10 years outpost operational life, ○ Feasible solutions, ▼ Selected design point, + Initial baseline design point).

4.2. Design session close-up

To close the design session, we collect all the information gained by analyzing the sensitivity analysis and the response surfaces graphs to select the best combination of design-variable levels to determine the baseline for successive design iterations, maybe at a deeper level of detail.

Suppose that the 25-tons launcher class is selected as baseline. It seems to be the most reasonable choice, especially because we discovered that there can be feasible SOS architectures considering this launcher class. Unless programmatic and cost analyses are performed, given the information we derived from the mathematical models there is no particular motivation of applying for a requirement change request, thus we cope with the value of 15 years for the lunar-outpost operational life. Concerning the dimensions of the capsule it seems advantageous to strive for a large one. The effect on the reduction of the number of launches is larger than the effect on the increase in the mass of the capsule, and the mass-constraint can be met anyway. The crewmembers comfort level depends on the choice of the service-module propellant I_{sp}. Choosing an I_{sp} of 500 s will bring the design point to be far enough from the constraint, thus allowing the comfort level to be high. With an intermediate level of I_{sp}, that will cause the mass of the propellant to increase (but probably the cost to decrease), a high comfort level can still be chosen, but with the warning of putting the design point close to the constraint. On the other hand, the combination of factor G ($=I_{sp}$) at the intermediate level and factor C ($=n_{\text{crew}}$) at the high level, with a high capsule comfort level would lead to unfeasibility, see Fig. 9. Thus, it seems wiser to set the service module propellant I_{sp} at the highest level. Concerning the Capsule I_{sp} and sidewall angle, we did not experience any major impact on the objectives and the constraints. The selection of the levels of these parameters would need to be taken considering other issues not included in the current version of the mathematical models, like cost and re-entry conditions and requirements, for instance, thus for the time being we could cope with the values of the baseline. The ascent module crewmembers comfort level can be set equal to the baseline value for the same reason.

With these settings of the design variables, the design point of the system-of-systems is placed in an interesting position on the objective space, with 90 launches and an assembly mass of almost 22 t, see Fig. 14. Here, all the design points computed with the mixed hypercube method are shown. As we can see, the response surface analyses, with constraint and sensitivity analyses allowed us to improve the baseline design. Indeed, the selected design point dominates the baseline design point, i.e., it is better considering all the objectives at the same time. Further, there are no design points on the graph that are better than the one selected, besides those obtained with a relaxed requirement on the lifetime of the lunar outpost. In Fig. 15, the non feasible solutions obtained during the design session are shown.

The design points computed with the mixed hypercube approach depend on the initial design point chosen as baseline. If the baseline point (the initial experts' guess) would have been far away from the actual one, the solution we found would probably not have been discovered, especially with reduced dimensions of the hypercube. This is the main drawback of working with a local design methodology. It provides information in the region within the hypercube only, thus not suitable to find globally optimal solutions. However, multiple initial baseline points may be analyzed in a single design session. This would allow the engineering team to per-form a more complete quantitative trade-off analysis.

Two main conclusions can be drawn from the analysis. The launch of a capsule and a service module to support a human outpost on the Moon is still possible with currently available launcher classes (even though some design changes would have to be discussed to be able to "launch" humans). For successive design phases, the design team must be very careful in handling the service module I_{sp}, since it is the main driver concerning performances and constraint.

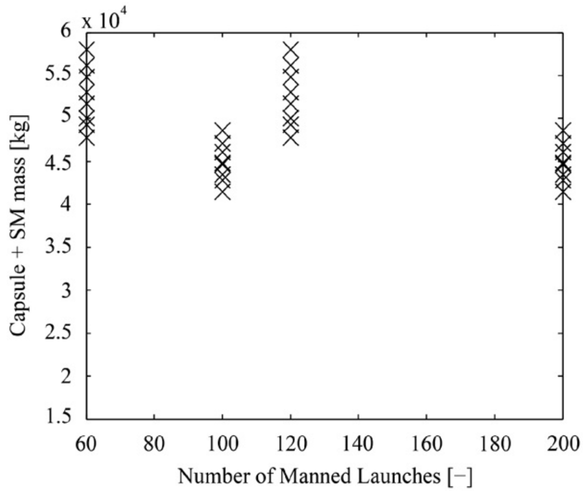

Fig. 15. Objective space, infeasible solutions.

5. Conclusions and final remarks

In the present paper a modelling framework for system-of-systems and a methodology for supporting the engineering team during its design have been presented.

The modelling framework resulted particularly useful in decomposing the system-of-systems to obtain modular and flexible software architecture. This feature allows the engineering team to work with interfaces, thus saving time during the modelling phase.

The mixed hypercube approach enabled a quantitative and traceable analysis to be performed with a limited computational effort. Continuous or discrete variables may be used in the methodology without major complication. This is particularly useful when dealing with architectural configurations, as shown in the example.

The analysis of variance and variance decomposition techniques allowed us to perform a sensitivity and factor-importance analysis over the entire design region within the hypercube. The response–surface analysis based on the results coming from ANOVA resulted faster and more effective since only the most relevant parameters were taken into account.

The visualization of the results in the form of bar plots for factor sensitivities, contour plots with constraints for the objectives trends as a function of the design variables is crucial for a methodology to support the design team to make informed decisions. "*What-if*" scenarios can be easily analyzed, so that fast decisions on heterogeneous systems and architectures can be made in a reasonable time with all the needed information available. The proposed approach allows the customer to be actively involved in the design process, since requirements can be treated as design variables thus having the possibility of discussing on them based on quantitative figures. The mixed hypercube approach allowed us to determine a design point that is better than the initially selected baseline, even if it may not be a global optimal solution. However, the results provide indications in the region of the design space in which investing more effort for the search of the optimum solution for eventual subsequent phases of the design process.

The mixed hypercube approach is very general, it can be implemented as a relatively cheap (computationally) local design method, for supporting the engineering team, with in principle any type of model, with continuous and discrete variables. It could also be used as a local refinement method after an optimization process, to provide more insight in the optimum solution(s) to the engineering team.

One of the main advantages of this integrated approach is also its main limitations. Expert knowledge can in general not be encapsulated in a set of equations. Experts' judgment in selecting good initial baselines is still a fundamental aspect, especially at an early design stage.

References

[1] J. Sobieszczanski-Sobieski, K. Hongman, J. Mullins, S. Ragon, G. Soremekun, A parallel trade study Architecture for Design Optimization of Complex Systems, AIAA ASME (2005) 2005–2201 April, Texas, AIAA.

[2] S. Kodiyalam, C. Yuan Evaluation of methods for multidisciplinary design optimization (MDO), part II, NASA/CR-2000-210313.

[3] J. Sobieszczanski-Sobieski, Multidisciplinary optimization for engineering systems: achievements and potential, NASA TM 101566, 1989.

[4] C.A. Coello Coello, D.A. Van Veldhuizen, G.B. Lamont, Evolutionary Algorithms for Solving Multi-Objective Problems, Kluwer Academic Publishers, e-ISBN 978-0-387-36797-2, 2002.

[5] G. Ridolfi, E. Mooij, S. Chiesa, A modeling framework for the concurrent design of complex space systems, in: Proceedings of the AIAA MST Conference, Toronto, August 2010.

[6] G. Ridolfi, E. Mooij, S. Corpino, A system engineering tool for the design of satellite subsystems, in: Proceedings of the AIAA MST Conference, August 2009, Chicago, AIAA p. 6037, 2009.

[7] G. Ridolfi, E. Mooij, S. Chiesa, A parametric approach to the concurrent design of complex systems, in: Proceedings of the 4th International Workshop on System & Concurrent Engineering for Space Applications, European Space Agency, Lausanne, October 2010.

[8] N.P. Tedford, J.R. Martins, On the common structure of MDO problems: a comparison of architectures, 11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Portsmouth, Virginia, Sep. 6–8, 2006.

[9] E.J. Cramer, J.E. Dennis Jr., P.D. Frank, R.M. Lewis, G.R. Shubin, Problem formulation for multidisciplinary optimization, CRPC-TR93334.

[10] J. Sobieszczanski-Sobieski, Optimization by decomposition: a step from hierarchic to non hierarchic systems. NASA CP 3031.

[11] D. Stanley, S., Cook, J. Connolly, NASA's exploration systems architecture study. NASA, TM 214062, 2005.

[12] B. Lyndon, Apollo Program Summary Report, Johnson Space Center, NASA, TM X 68725, Texas, 1975.

[13] P.M. Pardalos, H.E. Romeijn, Handbook on Global Optimization, Kluyver Academic Publishers, ISBN 1-4020-0742-6, 2002.

[14] K. Deb, Multi-Objective Optimization using Evolutionary Algorithms, Wiley, ISBN 0-471-87339-X, 2001.

[15] U. Lautenshlager, S.O. Erikstad, J.K. Allen, F. Mistree, Experimental satellite trajectory analysis using decision-based robust design, Journal of Guidance, Control, and Dynamics 18 (5) (1995).

[16] S.O. Erikstad, U. Lautenshlager, B. Bras, J.K. Allen, F. Mistree, Integrating robustness into multiobjective space vehicle design process, Journal of Guidance, Control, and Dynamics 18 (5) (1995).

[17] M.S. Phadke, Quality Engineering Using Robust Design, Prentice-Hall, Englewood Cliffs, NJ, 1989.

[18] D.C. Montgomery, Design and Analysis of Experiments, John Wiley and Sons, New York, ISBN 0-471-31649-0, 2001.

[19] A.I. Khuri, J.A. Cornell, Response Surfaces. Design and Analysis, Second ed., Marcel Dekker Inc., New York, ISBN 0-8247-9741-8, 1996.

[20] J.C. Helton, F.J. Davis, Illustration of sampling-based methods for uncertainty and sensitivity analysis, Risk Analysis 22 (3) (2002).

[21] D. Cardile, G. Ridolfi, E. Mooij, S. Chiesa, G. Ferrari, A system-of-systems approach for the concurrent design of space missions, IAC 2010, Prague, IAC-10.B5.2.3.

[22] G. Ridolfi, E. Mooij, D. Cardile, S. Corpino, F. Stesina, Orthogonal arrays based design methodology for complex coupled space systems, IAC 2010, Prague, IAC-10.B5.2.4.

Guido Ridolfi received his MSc in Aerospace Engineering from Politecnico di Torino, Italy, in 2008. He is a PhD candidate at the Department of Aeronautics and Space Engineering (DIASP) of the Politecnico di Torino since January 2009. His Ph.D. is carried out in the framework of an international cooperation with Delft University of Technology, Space Missions and Systems department. His activity is mainly focused on the development of methodologies for space missions and systems design, and for supporting of the engineering team and decision makers. He is also involved as thermal engineer in the e-st@r cubesat project at Politecnico di Torino.

Erwin Mooij received his MSc and Ph.D. in Aerospace Engineering from Delft University of Technology, The Netherlands, in 1991 and 1998, respectively. From 1995 until mid 2007 he was working for Dutch Space, The Netherlands, on re-entry systems and (real-time) simulator development. Currently, he is an Assistant Professor in the Faculty of Aerospace Engineering, Delft University of Technology. His research interests include re-entry systems, trajectory optimization, guidance and control system design, and design methods and data-analysis techniques. He is a senior member of the American Institute of Aeronautics and Astronautics, and a member of the AIAA GNC Technical Committee.

Diego Cardile is a Ph.D. candidate at the department of Aeronautics and Space Engineering at Politecnico di Torino since January 2010. The Ph.D. theme is about the integration of System of Systems methodologies in a Model Based System Engineering (MBSE) environment. The research activity is financed by Thales Alenia Space where he actually works. In July 2009, he received his MSc in Aerospace Engineering at Politecnico di Torino. From June to July 2005, he carried out an internship at Alenia Aeronautica. The activity was performed in the structural test division of Pomigliano D'Arco where he worked on performance characterization and integration of test equipments.

Sabrina Corpino is assistant professor in the Aerospace System Engineering Area at the Department of Aeronautics and Space Engineering (DIASP) of the Politecnico di Torino. She is involved in the education activities as well as in the research projects of the AeroSpace Systems Engineering Team (ASSET). She worked on the PiCPoT nanosatellite system design, physical simulation, and manufacturing and assembly definition. Since mid 2007 ASSET has been working at the design, manufacturing, test and launch into LEO of another university satellite (CubeSat standard), named e-st@r. Within this project she is Project Manager coordinating the team constituted by other researchers of Politecnico di Torino and some thirty students mainly from the Aerospace Engineering courses. Her research activities include low-cost systems performing scientific missions, and development of hardware-in-the-loop techniques for systems simulation and verification.

Giorgio Luigi Ferrari received his MSc in Aeronautical Engineering from University of Pisa, Italy, in 1986. He works at Thales Alenia Space-Italy where he is currently involved as System Engineer for the ATV ILR and work-package responsible for the STEPS program. During his career he was involved in several engineering activities as system engineer of the SOHO-UVCS, MPLM, CTV, Cupola, IHAB, and FLECS projects, and in support of the Aurora Exploration study by ESA.