We are IntechOpen,
the world's leading publisher of
Open Access books
Built by scientists, for scientists

6,900
Open access books available

186,000
International authors and editors

200M
Downloads

154
Countries delivered to

Our authors are among the

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK CITATION INDEX
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Complex-Systems Design Methodology for Systems-Engineering Collaborative Environment

Guido Ridolfi[1,2], Erwin Mooij [2] and Sabrina Corpino[1]
[1]*Politecnico di Torino*
[2]*Delft University of Technology*
[1]*Italy*
[2]*The Netherlands*

1. Introduction

In the last decades man-made systems have gained in overall complexity and have become more articulated than before. From an engineering point of view, a complex system may be defined as one in which there are multiple interactions between many different elements of the system and many different disciplines concurring to its definition. However, the complexity seen from the system perspective is only partial. In more general terms complexity does not only regard the system *per se*, but it is also related to the whole life-cycle management of the system. This encompasses all the activities needed to support the program development from the requirements definition to the verification, validation and operation of the system in the presence of a large number of different stakeholders. These two interrelated views of complexity, being *bottom-up* in the first case and *top-down* in the second, both converge to the system defined as an entity formed by a set of interdependent functions and elements that complete one or more functions defined by requirements and specifications.

Systems Engineering processes have been increasingly adopted and implemented by enterprise environments to face this increased complexity. The purpose is to pursue time and cost reduction by a parallelization of processes and activities, while at the same time maintaining high-quality standards. From the life-cycle management point of view the tendency has been to rely more and more on software tools to formally applying modeling techniques in support of all the activities involved in the system life-cycle from the beginning to the end. The transition from document-centric to model-centric systems engineering allows for an efficient management of the information flow across space and time by delivering the right information, in the right place, at the right time, and to the right people working in geographically-distributed multi-disciplinary teams. This standardized implementation of model-centric systems engineering, using virtual systems modeling standards, is usually called Model Based Systems Engineering, MBSE.

On the other side, looking at the problem from the perspective of the system as a product, the management of complexity is also experiencing a radical modification. The former adopted approach of sequentially designing with separate discipline activities is now being replaced by a more integrated approach. In the Aerospace-Engineering domain, for instance, designing with highly integrated mathematical models has become the norm. Already from

the preliminary design of a new system all its elements and the disciplines involved over the entire life-cycle are taken into account, with the objective of reducing risks and costs, and possibly optimizing the performance.

When the *right people* all work as a team in a multi-disciplinary collaborative environment, the MBSE and the Concurrent Engineering finally converge to the definition of the system. The main concern of the engineering activities involved in system design is to predict the behavior of the physical phenomena typical of the system of interest. The development and utilization of mathematical models able to reproduce the future behavior of the system based on inputs, boundary conditions and constraints, is of paramount importance for these design activities. The basic idea is that before those decisions that are hard to undo are made, the alternatives should be carefully assessed and discussed. Despite the favorable environment created by MBSE and Concurrent Engineering for the discipline experts to work, discuss and share knowledge, a certain lack of engineering-tool interoperability and standardized design methodologies has been so far a significant inhibitor, (International Council on Systems Engineering [INCOSE], 2007). The systems mathematical models usually implemented in the collaborative environments provide exceptional engineering-data exchange between experts, but often lack in providing structured and common design approaches involving all the disciplines at the same time. In most of the cases the various stakeholders have full authority on design issues belonging to their inherent domain only. The interfaces are usually determined by the experts and manually fed to the integrated models. We believe that the enormous effort made to conceive, implement, and operate MBSE and Concurrent Engineering could be consolidated and brought to a more fundamental level, if also the more common design analytical methods and tools could be concurrently exploited. Design-space exploration and optimization, uncertainty and sensitivity analysis, and trade off analysis are certainly design activities that are common to all the disciplines, consistently implemented for design purposes at the discipline-domain level. Bringing fundamental analysis techniques from the discipline-domain level to the system-domain level, to exploit interactions and synergies and to enable an efficient trade-off management is the central topic discussed in this chapter. The methodologies presented in this chapter are designed for their implementation in collaborative environments to support the engineering team and the *decision-makers* in the activity of *exploring* the design space of complex-system, typically long-running, models. In Section 2 some basic definitions, terminology, and design settings of the class of problems of interest are discussed. In Section 3 a test case of an Earth-observation satellite mission is introduced. This satellite mission is used throughout the chapter to show the implementation of the methods step by step. Sampling the design space is the first design activity discussed in Section 4. Then in Section 5 and Section 6 a general approach to compute sensitivity and to support the engineering team and decision makers with standard visualization tools are discussed. In Section 7 we provide an overview on the utilization of a unified sampling method for uncertainty and robustness analysis. Finally, we conclude the chapter providing some recommendations and additional thoughts in Section 8.

2. Basic definitions

The discussion and the methodologies presented in this chapter are based on the assumption that the activity of designing a complex system is performed by a team of designers (the engineering team), using **mathematical models** to determine the physical and functional characteristics of the system itself. A mathematical model is a set of relationships, i.e.,

equations, providing figures-of-merit on the **performance(s)** of the system to the engineering team when certain **inputs** are provided. The inputs are represented by the **design variables**, i.e., factors that are responsible for influencing the performance(s) of the system. For this motivation, the design variables will also be called **design factors**, or more generally inputs, or simply variables. The domain of existence of the design variables forms the **design space**, where they can assume certain **values** between a minimum and a maximum. The **design-variable range** determined by the minimum and the maximum can of course only be as large as the domain of existence of the variable. Mimima and maxima for the design variables are usually set by the engineering team to limit the analysis to a specific region of the design space or to avoid infeasible conditions. For instance, the design range of the eccentricity, e, of a closed orbit about the Earth should not exceed the interval $0 \leq e < 1$. In the upper-left Cartesian diagram of Fig. 1 a hypothetical design space, formed by two variables, is shown. The limits of the variable ranges are represented by the dash-dotted lines. The subspace of the design space determined by all the design-variable ranges is addressed as the **design region** of interest, and it is represented by the rectangle formed by the dash-dotted lines and the vertical axis of the Cartesian diagram.
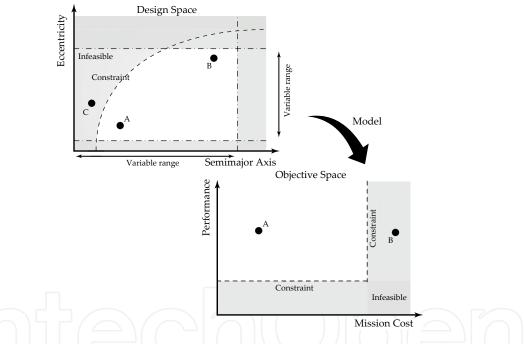

Fig. 1. Schematic representation of the design space and the objective space of the model.

Design variables can be **continuous** or **discrete**. A continuous variable can assume all the values between the minimum and the maximum. A discrete variable, instead, can assume only few specific values in the design-variable range. In this case the values are called **levels**. Discrete variables can be further distinguished into two classes, namely **ordinal** or **categorical**. The *length* of a solar array on a satellite system, for instance, is a continuous variable. It can assume, in principle, any value between a minimum and a maximum set to limit the weight or to provide a minimum performance under certain circumstances. The *number of cells* used to build the array is an ordinal variable. It can only assume the levels represented by the natural numbers, and certain characteristics increase (decrease) when the number of cells increases (decreases), e.g., the total mass. The *type of solar cell*, instead, is a categorical variable. This

means that it can only assume certain levels (e.g. *type#1*, *type#2*, and so on), but in this case the order is not important. It is not always the case that, for instance, the *efficiency* of the solar cells increases going from the first type to the second type and so on. It depends on the order in which they appear in a database, for instance, that may be an arbitrary choice of the engineering team. The model of the system may be also subject to other sources of variability representing the non-deterministically known parameters typical of the operating environment of the system. The residual atmospheric density on orbit, the solar radiation, the orbit injection errors, just to mention a few, are factors that may not be directly controlled at a design stage therefore they must be taken into account in a statistical sense. These factors are called **uncontrollable**.

One of the main tasks of the engineering team during the **design process** of the system is to set the values and/or the levels of the design variables in such a way that the performance(s) of the system assume a certain optimal level under *certain circumstances* (**optimal design**), and/or such that the final system is insensitive to variations of the uncontrollable factors (**robust design**). The performance(s) of interest are called **objective(s)** of the analysis. The space in which the objectives can be represented, i.e., the domain of the images of the mathematical equations of the model, is called **objective space**. Thus, the model is responsible for relating points in the design space with points in the objective space. The term *certain circumstances* is used to indicate the constraints and boundary conditions of the analysis. As already mentioned, the **boundary conditions** are represented by the design-variable ranges, the dash-dotted lines of Fig. 1. The **constraints**, instead, are determined by an infeasible condition in the objective space, e.g., the mass of the satellite exceeding the mass that the launcher is able to deliver in a given orbit. Further, the constraints can also be determined by infeasible conditions on the design space, when certain combinations of the values or levels of the design variables are not allowed. This may happen, for instance, with the eccentricity and the semimajor-axis of an Earth-orbiting satellite. Their combined values must ensure that the perigee altitude of the orbit is at least larger than the radius of the Earth. Constraints may be linear or non-linear, continuous or discrete. The dashed lines in Fig. 1 represent the constraints in the design space (non linear in this case), and in the objective space (linear in this case). The thick dots in Fig. 1 represent the **design points**. In the design space they are a representation of the values of the design variables, while on the objective space they represent the corresponding set of output values. Considering a deterministic model, there is a one-to-one correspondence between one point in the design space and one point in the objective space. However, the engineering team must make sure to provide design points that do not violate constraints in the design space. For instance, an orbit with a semi-major axis of 7000 km and eccentricity of 0.7 would lead to a negative value of the satellite altitude at perigee (i.e., non existing orbit) thus with the impossibility of computing relevant parameters such as, for instance, *time-in-view* at perigee passage on a specific region on Earth. Therefore, in Fig. 1 the design point C does not have a corresponding image on the objective space. In this case, the semi-major axis and the eccentricity are classified as correlated inputs.

The problem of developing and implementing the mathematical model of a complex system is beyond the scope of this chapter. However, a brief discussion on the type of modelization approach is beneficial for a better understanding of the discussed design methodologies. The development of a mathematical model is tackled considering two main sub-problems, namely *problem decomposition*, (Sobieszczanski-Sobieski, 1989), and *problem*

formulation, (Cramer et al., 1993; Tedford & Martins, 2006). In the literature, authors propose several model-decomposition techniques. However, two main classes may be identified, namely *Hierarchical Decomposition* and *Non-Hierarchical Decomposition* methods, (Sobieszczanski-Sobieski & Haftka, 1995). Non-Hierarchical Decomposition methods (NHD) are advised when there is no clear separation between two or more elements/disciplines, i.e. when the coupling between them is not negligible *a priori*. The formulation of the complex-system design problem is related to the allocation of the resources to the various elements of the architecture. Single- and multiple-level formulations are discussed in the literature, (Cramer et al., 1993; Tedford & Martins, 2006; Yi et al., 2008). The former must be executed on a single machine, the latter, instead, allows for more flexibility in allocating the computational resources. The mathematical models of a collaborative environment are most likely developed using a NHD approach, because it is the most general one, and with a multi-level architecture because resources are usually geographically distributed. An example of the multi-level architecture of a complex-system design problem is presented in Fig. 2. It represents the architecture most likely adopted in a collaborative environment with team-members responsible for element analysis and others responsible for system analysis. The thick lines represent input/output interfaces.
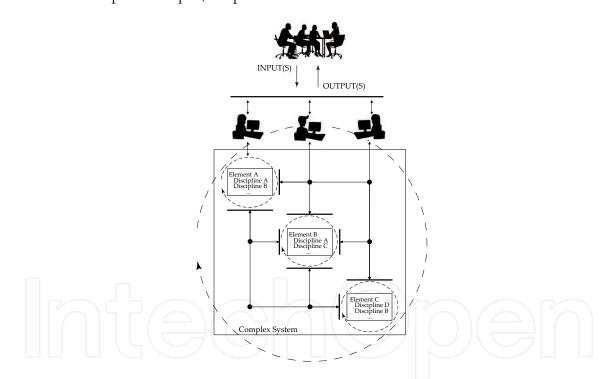

Fig. 2. Schematic of the Collaborative Bi-Level (COBiL) formulation for complex systems models.

3. Design case: Earth-observing satellite for natural disaster and land-use monitoring

Earth-observation satellites can observe areas over a wide range rather quickly. It is expected that their observation data combined with information obtained by aircraft and helicopters will be useful for a regular disaster condition assessment. This would make rescue

operations more effective, would allow for extracting topographical information reflecting latest land-usage changes, and identifying disaster risks.

In this chapter, the preliminary design of an Earth-observation mission to support the world-wide disaster management process and land-usage monitoring is deployed and discussed to show a practical implementation of the proposed design approaches. The following mission statement is considered as driver for the design process:

Design an Earth observation mission to provide world-wide disaster-management capabilities, for over a period of 7 years

The limited available space and at the same time the willingness to effectively convey the message of this chapter led us to take several assumptions to determine boundaries of the analysis presented here. A satellite system with an optical payload (staring sensor) is considered. The main purpose is to achieve a compromise between the design variables in such a way to obtain the best possible image resolution, at minimum cost. The satellite shall revisit the same area on the Earth surface within 24 hours, and shall be able to send the acquired data back, in real time, to any equipped ground station (the reference ground station is considered with 1 m aperture antenna diameter) with a link margin of at least 4 dB. The selected launcher is of the class of the *Delta II 6920/25*, with a maximum payload on polar orbit of 2950 kg. A highly inclined, circular orbit has been selected, with $i = 98°$. The main mission geometry parameters and few of the equations implemented for computing the coverage and the resolution are presented in Fig. 3.

In Table 1 the design variables taken into account in the analysis, and their intervals or levels (in case of discrete variables) are summarized.

The mathematical model of the satellite system is composed of all its main subsystems (i.e., payload, Attitude Dynamics and Control System (ADCS), communication system,

Design Variables	Code	Intervals Min	Max	Levels	Design Variables	Code	Intervals Min	Max	Levels
Number of days (rep. ground track)	A	1	3	3	Number of slew maneuvers [-]	G	10k	30k	–
Number of orbits (rep. ground track)[a]	B	1	3	3	Transmitting output RF power [W]	H	5	30	–
Instrument aperture diameter [m]	C	0.3	1	–	Antenna diameter [m]	I	0.1	1	–
Min. ϵ [deg]	D	5	50	–	Type of solar array [-]	J	1	2	2
Max. slew angle [deg]	E	0	50	–	Type of thrusters [-]	K	1	2	2
Min. maneuver time [s]	F	60	180	–	Payload heritage [-]	L	1	2	2

Table 1. Settings of the design variables.[a] When $A = 1$, $B = 13, 14$ or 15. When $A = 2$, $B = 28, 29$ or 30. When $A = 3$, $B = 43, 44$ or 45.
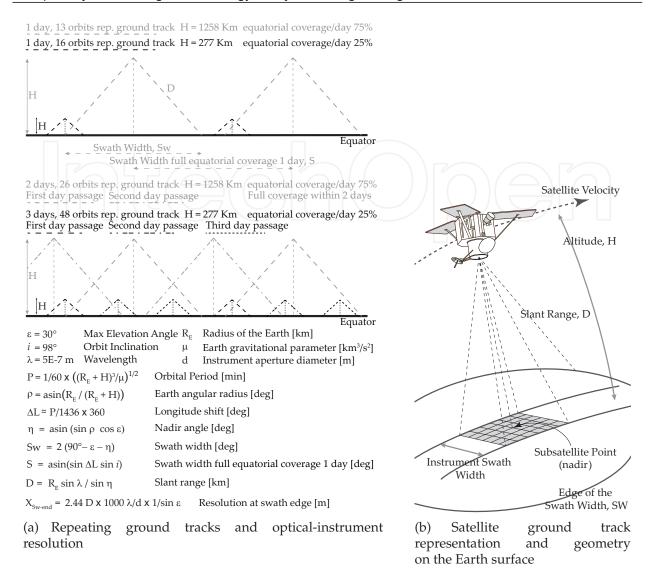
1 day, 13 orbits rep. ground track H = 1258 Km equatorial coverage/day 75%
1 day, 16 orbits rep. ground track H = 277 Km equatorial coverage/day 25%

H

D

H

Equator

Swath Width, Sw
Swath Width full equatorial coverage 1 day, S

2 days, 26 orbits rep. ground track H = 1258 Km equatorial coverage/day 75%
First day passage Second day passage   Full coverage within 2 days

3 days, 48 orbits rep. ground track H = 277 Km equatorial coverage/day 25%
First day passage Second day passage   Third day passage

H

H

Equator

$\varepsilon = 30°$	Max Elevation Angle	R_E	Radius of the Earth [km]
$i = 98°$	Orbit Inclination	μ	Earth gravitational parameter [km³/s²]
$\lambda = 5E\text{-}7$ m	Wavelength	d	Instrument aperture diameter [m]

$P = 1/60 \times ((R_E + H)^3/\mu)^{1/2}$ Orbital Period [min]

$\rho = \operatorname{asin}(R_E / (R_E + H))$ Earth angular radius [deg]

$\Delta L \approx P/1436 \times 360$ Longitude shift [deg]

$\eta = \operatorname{asin}(\sin \rho \cos \varepsilon)$ Nadir angle [deg]

$Sw = 2(90° - \varepsilon - \eta)$ Swath width [deg]

$S = \operatorname{asin}(\sin \Delta L \sin i)$ Swath width full equatorial coverage 1 day [deg]

$D = R_E \sin \lambda / \sin \eta$ Slant range [km]

$X_{Sw\text{-}end} = 2.44 D \times 1000 \lambda/d \times 1/\sin \varepsilon$ Resolution at swath edge [m]

Satellite Velocity

Altitude, H

Slant Range, D

Subsatellite Point (nadir)

Instrument Swath Width

Edge of the Swath Width, SW

(a) Repeating ground tracks and optical-instrument resolution

(b) Satellite ground track representation and geometry on the Earth surface

Fig. 3. Satellite mission geometry. Equations adapted from (Wertz & Larson, 1999).

power and avionics system, propulsion system, structure and thermal-control system) and a ground control station model. The cost is computed using the *Unmanned Spacecraft Cost Model* presented by Wertz & Larson (1999). Cost is mostly related to the mass and power consumption of the satellite, the type of technology used (e.g., type of payload or type of attitude control), and on the technology heritage of its components (the higher the cheaper). From database, two types of solar arrays and two types of thrusters are taken into account. The two types of solar arrays present an efficiency, *η*, of 0.14 and 0.2, and a power density of 115 [W/kg] and 100 [W/kg] respectively. The two thrusters are the *STAR48A* and the *IUS-SRM2* with a specific impulse of 250 [s] and 300 [s], (Wertz & Larson, 1999), and a percentage of inert mass with respect to the propellant of 0.13 and 0.21, respectively. The two levels of payload heritage foresee an *adapted* design from an existing one and a *new* design, respectively. The *new* design is more expensive, but allows for a better management of the acquired data on board, i.e., reduced data rate. The results of the analysis are discussed in the following sections, for every design step and for every type of design methodology presented.

4. Sampling the design space

Sampling the design space is the first step necessary when the mathematical model of a system needs to be studied. A sample is a set of points in the design region (a $k - dimensional$ hyperspace) whose coordinates are the values of the design variables taken from their variability ranges, $(x_1, x_2, \cdots, x_k)$, in their marginal (for independent inputs) or joint (for correlated/coupled inputs) distribution, see the black dots in Fig. 1.

The simplest, and possibly most straightforward approach to sampling is to generate a sequence of random points in the design region, as shown in Figure 4(a). The Latin Hypercube Sampling (LHS),developed by McKay, McKay et al. (1979), is an alternative method seen as a subclass of the stratified-sampling class. The LHS provides full stratification of the design region, thus increased design-space coverage characteristics if compared to the generic stratified sampling and the random sampling, see Figure 4(b). However, good space-filling characteristics are not always guaranteed, in the sense that points in the design space may still form separate and disordered bunches. Viana et al. (2010) propose an algorithm for near-optimal Latin hypercube designs (i.e., maximizing the distance between the samples) without using formal optimization, see Figure 4(c). This method provides results with a negligible computational effort if the number of design variables k is not so large. According to our experience using this algorithm, it requires the generation of matrices with at least 2^k elements, irrespective of the number of samples actually required. The number of matrix entries to be stored to compute the near-optimal LHS can become cumbersome already for 20 variables. The Sobol LP_τ sequence,Sobol (1979), is a quasi-random sampling technique that provides *low-discrepancy* sample points. Here discrepancy indicates a measure of *non-uniformity* and proximity between the samples. In Bratley & Fox (1988) and Press et al. (2007) there are useful indications on how a Sobol LP_τ sequence, or its variant proposed by Antonov & Saleev (1979), can be computed. The (modified) Sobol LP_τ sequence has the particular characteristic of providing a sequence of points for which successive points at any stage *know* how to fill in the gaps in the previously generated distribution, Press et al. (2007), see Figure 4(d). This aspect is particularly useful for the re-utilization of previous sample points when additional points shall be sampled to improve the quality of the results, as will be demonstrated later in the case of regression analysis. The modified Sobol LP_τ sequence demonstrates that the additional sample points, the circles in Fig. 4, are placed in such a way to fill the gaps following a pre-defined pattern, allowing for a more efficient re-utilization of the samples previously generated.

4.1 Design of experiments

An experiment is a test performed to evaluate the outcome of a process given certain settings of the factors believed to influence it. The *experiments* considered in this chapter are all *computer experiments* performed on the mathematical model in correspondence of the sample points. However, the Design of Experiment (DoE) practice has older origins than the computer era, indeed it was first introduced by Fisher in 1935. The sampling methods belonging to the category of DoE can be distinguished in Factorial Designs (full or fractional), Orthogonal Arrays and other methods, amongst which, for instance, Central Composite Design (CCD). The common characteristic of these sampling methods is that they are all deterministic. The samples are placed on the design space according to a certain pre-defined geometry, so that
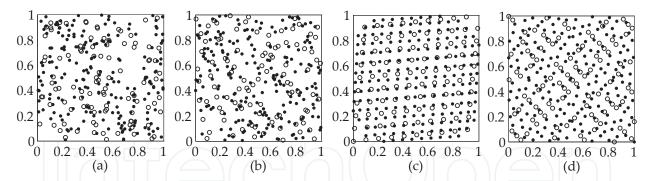
Fig. 4. Scatterplots of sampling points in a 2-dimensional design space based on (a) random sampling, (b) Latin Hypercube sampling, (c) sub-optimized Latin hypercube sampling, (Viana et al., 2010), (d) modified Sobol LP_τ sequence. • Initial sample, 100 points. ○ Additional sample, 100 points.
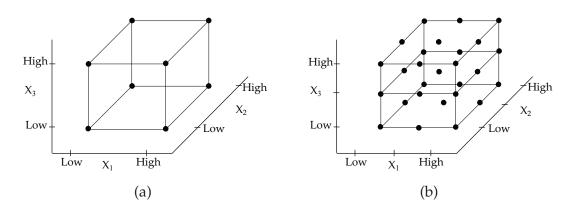

Fig. 5. Full factorial design with (a) 2 variable-levels and (b) 3 variable-levels in a 3-dimensional design space.

also *ordinal* and *categorical* variables can be used in the analysis, rather than only *cardinal* (i.e., continuous) variables as in the previously described sampling techniques. In this case the values of the variables are more properly called *levels*.

4.1.1 Factorial design

Factorial design, or *full* factorial design, is a sampling method that foresees one experiment for each possible combination of the levels of the factors. If factor *A* has *a* levels, factor *B* has *b* levels and factor *C* has *c* levels, the total number of experiments is $N = a \cdot b \cdot c$. There are special cases of factorial design where for all the factors only 2 or 3 levels are considered. They are usually called 2^k and 3^k factorial designs respectively, where *k* indicates the number of factors. The experimental structure obtained for 2^k and 3^k factorial designs is shown in Fig. 5 where the dots indicate the sample points.

Full-factorial design requires a number of experiments that increases with the power of the number of factors. Thus, already in the case of 2^k or 3^k factorial designs, the experimentation (i.e., the simulation of the model) can become cumbersome very soon. Therefore, fractional factorial designs were introduced as an attempt to reduce the computational effort for the analysis. As the name suggests, fractional-factorial designs only foresee a fraction of the

Experiment	Factors Assignment						
	A	B	C	D	E	F	G
1	1	1	1	1	1	1	1
2	1	1	1	2	2	2	2
3	1	2	2	1	1	2	2
4	1	2	2	2	2	1	1
5	2	1	2	1	2	1	2
6	2	1	2	2	1	2	1
7	2	2	1	1	2	2	1
8	2	2	1	2	1	1	2

Table 2. L_8 orthogonal array

number of experiments required by a full-factorial design with the same number of factors and the same number of levels. For instance a *one-half* fractional factorial design, or 2^{k-1} design, requires half of the experiments of the original 2^k design.

All the designs belonging to the category of DoE are also called matrix designs. Indeed their visualization, and their construction, is better understood if represented in the form of a matrix with the factors in the columns and the experiments to perform in the rows. A graphical structure for more than 3 variables becomes hard to visualize, see Table 2. A 2^{k-1} design is also called *Resolution 5* design (for $k > 4$). It is also possible to generate fractional-factorial designs that require less experiments than Resolution 5. However, the smaller the number of experiments, the lesser the information that can be obtained, as will be discussed in Section 5.2. Box et al. (1979) provide a thorough discussion on DoE, in general. Montgomery (2001), instead, present a complete overview of factorial designs, methods for obtaining several kinds of designs and their implications. For more detailed analysis we advise to refer to their original work.

4.1.2 Orthogonal arrays

Orthogonal Arrays, OAs, are special matrix designs originally developed by Taguchi (1987). OAs can be used as Resolution 3, Resolution 4, and Resolution 5 designs by properly arranging the columns of the design matrices, (Phadke, 1989). The term *orthogonal* is related to the balancing property, which means that for any pair of columns, all combinations of factor levels are present an equal number of times. In Table 2, the 1s indicate the *low* levels, while 2s indicate the *high* levels of the design factors.

The L_8 orthogonal array of Table 2 is only one amongst the many OAs discussed in (Taguchi, 1987). It is possible to build also three-, four-, and five-level OAs, and also mixed-levels OAs for factors having a heterogeneous number of levels, (Phadke, 1989). An efficient algorithm to generate three-level OAs is discussed by Mistree et al. (1994) while standard tables for other types of orthogonal arrays can be found in (Taguchi, 1987) and (Phadke, 1989).

4.1.3 Other experimental designs

The major distinction amongst the experimental designs is usually made between first- and second-order designs, as already hinted before. In the first case the design variables can assume only two levels, while in the second case at least three levels per design variable
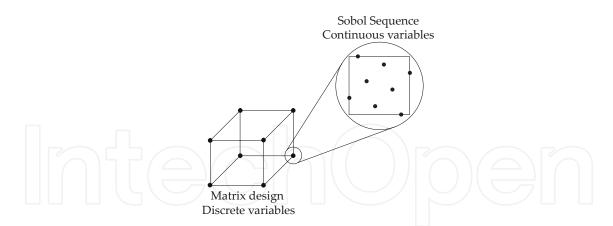
Fig. 6. Mixed-hypercube sampling with 3 discrete and 2 continuous variables.

are considered. The development of second-order designs is mainly related to the need of obtaining information on the curvature of the design space for fitting second-order response surfaces. Box et al. (1979) present a method to compute fractional 3^k factorial designs, the Box-Behnken designs, obtained by combining two-level factorial designs with balanced incomplete block designs. The Central Composite Design, CCD, introduced by Box & Wilson (1951), is build instead using a 2^k factorial design, plus a central point (in the geometric center of the design hyperspace), plus $2k$ points on the axis of every design variables at a distance α from the center. In a hyperspace normalized in the interval $[-1, 1]$, a CCD with $\alpha \neq 1$ will present 5 levels for each variables, while with $\alpha = 1$ it will only require the variables to assume 3 different levels. The interested readers may refer to Box et al. (1979) and Montgomery (2001) for a good overview and discussions on the many types of available experimental designs.

4.2 The mixed-hypercube approach

The purpose of the mixed-hypercube approach is to exploit both stratified sampling and DoE to efficiently sample the design space for obtaining information on the effect of both the continuous and the discrete design variables on the performance(s) of interest. The main idea is to separate the continuous variables and the discrete ones in two groups. A matrix design is then created for the discrete variables while for every row of the matrix design a Sobol sequence is generated for the remaining continuous variables. An example with three discrete and two continuous variables is presented in Fig. 6.

The advantage of using a matrix design instead of a space-filling technique for the discrete variables is that it allows to deterministically select the levels of the factors. When only few factor-levels can be selected (e.g., in a database there is a certain number of batteries, or only a limited number of thrusters is considered in the analysis of a satellite system) the maximum number of simulations is determined by a full factorial design. Therefore, its relative Resolution 5, 4, and 3 designs are the best way of obtaining samples by avoiding to disrupt the balance characteristics of the sampling matrix. The modification of a random or pseudo-random technique for sampling only at certain levels does not immediately provide such a balance, especially when the number of samples is kept low. On the other hand, in case of continuous variables matrix designs alone are less flexible in *filling* the design region and less suitable for the *re-sampling* process than the Sobol technique. The proposed

mixed-hypercube sampling approach allows for covering the design region more uniformly when compared to all the other techniques mentioned in this section, already with a low number of samples. The sensitivity-analysis technique described in Section 5, will directly benefit from these characteristics, since convergence of the variance is obtained with a reduced computational effort, for instance. Further, response surfaces for the continuous variables, and linear and interaction graphs for the discrete ones can be directly computed from the outcome of the simulations, with no additional data manipulation, see Section 6. A more detailed description of the implications of using specific implementations of the mixed-hypercube sampling method in combination with the design approaches presented in this chapter is discussed in the following sections.

5. Sensitivity analysis

Sensitivity analysis can be defined as the study of the *effect* of a certain input x on a given output Y. This *effect* can be the result of a local measure, e.g., the measure of a derivative as for instance $(\partial Y/\partial x)_{x=x^*}$, which requires an infinitesimal variation of the input x around a specific value x^*. However, the measure of sensitivity can also be obtained when the input ranges in a specified finite interval. In this case sensitivity analysis is valid over the entire interval of variation spanned by the input factor rather than only a single point. Therefore this type of sensitivity analysis is often called *global*. The settings of the problem of designing a complex system by selecting the most appropriate combination of input-factor levels is particularly suitable for the implementation of global sensitivity analysis. Indeed, in this context sensitivity analysis is aimed at finding the set of relevant factors in the determination of the output, providing information that is valid over the entire design region, even if it represents only a (small) subset of the design space. The main design questions that can be answered by using the global sensitivity analysis technique are the following:

Amongst all the design factors of the system model, what are those actually influencing the performance of interest? To what extent do these factors influence the performance?

The answer to these questions, already at an early stage of the design, could bring several advantages to the engineering team. First, it allows to identify the *design drivers*, i.e., those factors or group of factors that shall be carefully assessed, because they will be the main responsible for determining the performance of the system. The extent of the influence identified may be useful for checking the adequacy of the model being used for the analysis and for corroborating the underlying analysis assumptions.

5.1 Sensitivity indices

The relative importance of the factors can be determined on the basis of the reduction of the (unconditional) variance of the output Y, $V(Y)$, due to fixing that factor to a certain (yet unknown) value. A global quantitative measure for the *importance* of the factors, based on their contribution to the variance of the response, was first introduced by Sobol (1993). In (Sobol, 1993) and in (Sobol, 2001), the author presents a formal demonstration of his approach and a method for computing the sensitivity indices (sometimes called Sobol indices). Consider $Y = f(X)$ as the model of interest. Y is the response vector while $X = (x_1, x_2, \cdots, x_k)$ is

the vector with the k independent input factors. The method of Sobol discussed here and the regression-based sensitivity analysis described later in this section are in general valid for independent input factors. The case with correlated inputs implies that the correlation structure must be taken into account during the sampling of the design space, leading to higher computational cost, (Saltelli et al., 2004). An effective technique for imposing the correlation between input variables has been proposed by Iman & Conover (1982). However, in the case of systems design using mathematical models, dependencies between factors are very often accounted for within the model itself, leaving only the independent factors as design variables. Sometimes instead, input variables can still be considered independent if the design ranges are carefully selected. In the case of the semi-major axis and the eccentricity discussed in Section 2 one could limit the value of the eccentricity to the maximum possible with the minimum semi-major axis, for instance.

To compute the sensitivity, a sample of N points is taken from the model Y (performing N evaluations of the model Y). The unconditional variance $V(Y)$ can be decomposed as shown in Eq. (1), (Sobol, 1993). The expression in Eq. (1) is the ANOVA (Analysis Of Variance) representation of $V(Y)$, (Sobol, 2001).

$$V(Y) = \sum_i V_i + \sum_i \sum_{j>i} V_{ij} + \cdots + V_{12\cdots k} \tag{1}$$

All the terms of Eq. (1) are conditional variances of the factors indicated by the subscript indices. For instance V_i is the fraction of $V(Y)$ due to factor x_i only. V_{ij}, instead, represents the contribution of the interaction of x_i and x_j to $V(Y)$. The Sobol sensitivity indices are defined as in Eq. (2), (Sobol, 1993). S_i, S_{ij}, or $S_{i,j\cdots k}$ are sometimes called *first-order sensitivity indices*. They refer to the contribution to the variance of the single factors of Eq. (1). An additional measure of sensitivity is represented by the so-called total-effect sensitivity indices, S_{Ti}. A total-effect sensitivity index takes into account the unconditional variance of a certain variable x_i considering the first-order and all the higher-order effects in which it is involved. The total-effect sensitivity indices can be computed using Eq. (2) where V_{-i} indicates the contribution to the variance due to all factors but x_i and all the higher-order effects in which it is involved (Saltelli et al., 2004).

$$S_i = \frac{V_i}{V(Y)} \qquad\qquad S_{Ti} = 1 - \frac{V_{-i}}{V(Y)} \tag{2}$$

Global sensitivity indices can be estimated using qualitative or quantitative methods, it depends on the purpose of the analysis, on the complexity of the problem and on the available computational resources. A qualitative approach, like the method of Morris, (Morris, 1991), allows to determine the relative importance of the factors with a relatively limited computational effort. It is not possible to obtain a precise measure of the percent contribution of the factors to the unconditional variance, thus these methods are usually used as a preliminary analysis to detect and fix the unimportant factors. Therefore, qualitative methods are also called *screening methods*. Techniques like the method of Sobol, (Sobol, 1993), or the FAST (Fourier Amplitude Sensitivity Test), (Cukier et al., 1978), require a large number of model evaluations to provide quantitative sensitivity indices of the design factors, especially the terms like V_{ij}, or $V_{ij\cdots k}$. The regression-based sensitivity analysis method described in

the following section provides a quantitative measure of the global sensitivity indices with a limited computational effort. The sensitivity indices computed with this method are based on the decomposition of the variance computed by a regression model, providing information on the first-order as well as on higher-order effects of the factors on the response.

5.2 Regression based sensitivity analysis

If the design region of interest is not stretched out so much, a polynomial regression model is often sufficient to accurately describe the behavior of the system. This is true for typical models of engineering systems, especially when the source of complexity is represented by the large number of elements and their interrelated behavior rather than the mathematical models of every single component. However, also when the complexity is related to the highly nonlinear and non-smooth behavior of the mathematical equations linking the design variables, in a relatively small portion of the design space a polynomial regression model is still able to describe the system and explain most (if not all) of the variability of the data.

The Regression-Based Sensitivity Analysis (RBSA) method proposed here, is general enough to be applicable to regression models of any order. However, the choice of the regression-order depends on several aspects that will be discussed throughout this section. For ease of the discussion the method will be explained using the second-order model presented in Eq. (3) as a reference.

$$Y = \beta_0 + \sum_{i=1}^{k} \beta_i x_i + \sum_{i=1}^{k} \beta_{ii} x_i^2 + \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} \beta_{ij} x_i x_j \tag{3}$$

Here, β_i, β_{ii} and β_{ij} are the so-called regression coefficients that are calculated by fitting a response surface through the points sampled from the model, using the least-squares method. The estimate of the regression coefficients can be computed with the least-squares estimator, for instance:

$$\hat{\boldsymbol{\beta}} = \left(\mathbf{X}\mathbf{X}'\right)^{-1} \mathbf{X}'\mathbf{Y} \tag{4}$$

The fitted model is therefore represented by the following equation:

$$\hat{\mathbf{Y}} = \mathbf{X}\hat{\boldsymbol{\beta}} \tag{5}$$

Given a set of observations of a mathematical model, the variance of the data can be computed with the well-known equation:

$$\hat{V} = \frac{\sum_{i=1}^{N} \left(Y_i - E(\mathbf{Y})\right)^2}{N-1} \tag{6}$$

where $E(\mathbf{Y})$ is the expected value, or mean value, of the model output. The expression at the numerator of Eq. (6) is called sum of squares. Since in this case all the observations are taken into account we will refer to it as the total sum of squares, SS_T. The sum of squares of the regression only, instead, can be computed as follows:

$$SS_R = \sum_{i=1}^{N} \left(\hat{Y}_i - E(\mathbf{Y})\right)^2 \tag{7}$$

The SS_R, represents the portion of the total variability that can be explained by the regression model. In case the regression model perfectly fits the data then $SS_T = SS_R$. When residuals are present, in case of lack-of-fit, the portion of the total variability not explained by the regression model can be computed in the form of the error sum of squares, SS_E:

$$SS_E = \sum_{i=1}^{N} (Y_i - \hat{Y}_i)^2 \tag{8}$$

The regression sum of squares, as already mentioned, indicates how much of the observed variability is explained by the fitted model. To obtain the sensitivity indices of all factors that contribute to the total variability of the regression model, the regression sum of squares should be divided into its components, as done in Eq. (1). The main idea is to associate a sensitivity index to the additional variability calculated when a factor is added to the regression model. In Eq. (9) an alternative form of Eq. (5), combined with Eq. (4), is presented.

$$\hat{\mathbf{Y}} = \mathbf{X} \left(\mathbf{X}\mathbf{X}' \right)^{-1} \mathbf{X}'\mathbf{Y} = \mathbf{H}\mathbf{Y} \tag{9}$$

The matrix $\mathbf{X} \left(\mathbf{X}\mathbf{X}' \right)^{-1} \mathbf{X}'$ is called the *hat* matrix. It transforms the vector of the observed responses $\mathbf{Y}$ into the vector of the fitted values $\hat{\mathbf{Y}}$. Using the hat matrix, the total, regression, and error sums of squares can be expressed with the following relationships:

$$SS_T = \mathbf{Y}' \left[\mathbf{I} - \frac{1}{N}\mathbf{J} \right] \mathbf{Y} \qquad SS_R = \mathbf{Y}' \left[\mathbf{H} - \frac{1}{N}\mathbf{J} \right] \mathbf{Y} \qquad SS_E = \mathbf{Y}' \left[\mathbf{I} - \mathbf{H} \right] \mathbf{Y}$$

where $\mathbf{I}$ is a $N \times N$ identity matrix, and $\mathbf{J}$ is a $1 \times N$ vector of ones. Given these settings the RBSA is easy to compute. Let us consider a model in the form of Eq. (3) with three variables only. The compact notation Y_{full} denotes the model computed taking into account all the three factors, 2-factor interactions and quadratic terms, Eq. (10). In this notation $Y_{-x_1x_2}$ denotes the model computed excluding the factor x_1x_2., Eq. (11). The sensitivity index for the factor x_1x_2 can thus be computed as shown in Eq. (12).

$$Y_{full} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_{11} x_1^2 + \beta_{22} x_2^2 + \beta_{33} x_3^2 + \beta_{12} x_1 x_2 + \beta_{13} x_1 x_3 + \beta_{23} x_2 x_3 \tag{10}$$

$$Y_{-x_1x_2} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_{11} x_1^2 + \beta_{22} x_2^2 + \beta_{33} x_3^2 + \beta_{13} x_1 x_3 + \beta_{23} x_2 x_3 \tag{11}$$

$$S_{x_1x_2} = \frac{V(Y) - V_{-x_1x_2}}{V(Y)} = \frac{SS_T - SS_R(Y_{-x_1x_2})}{SS_T} \tag{12}$$

The conditional variance term $SS_R(\mathbf{X}_{-x_i})$ can also be computed and interpreted as the variance determined by excluding the ith design variable from the model. It is equivalent to the notation V_{-i} used before. In this case the sensitivity indices provide a measure of the total contribution of the variable x_i to the variance of the performance, considering all the interactions and higher order effects in which x_i is involved, see for instance Eq. (13) and Eq. (14). The sensitivity indices S_i are computed for all the terms of the model indicated in Eq. (3) while the total sensitivity indices S_{Ti} are computed for every design variable.
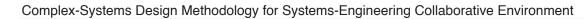
$$Y_{-x_1} = \beta_0 + \beta_2 x_2 + \beta_3 x_3 + \beta_{22} x_2^2 + \beta_{33} x_3^2 + \beta_{23} x_2 x_3 \tag{13}$$

$$S_{Tx_1} = \frac{V(Y) - V_{-x_1}}{V(Y)} = \frac{SS_T - SS_R(Y_{-x_1})}{SS_T} \tag{14}$$

The validity of the sensitivity indices computed with RBSA depends on the lack-of-fit of the regression model with respect to the sample data. Indeed, particular attention must be paid to the ratio between the regression and the total sum of squares. If SS_R is close to SS_T, then the regression model is able to account for a large part of the output variance, and as a consequence the sensitivity indices are meaningful measures. If this is not the case, lack-of-fit is present meaning that important terms are missing from the initially assumed regression model. However, this information is still important to decide whether to proceed with sensitivity analysis anyway or to modify the initial assumption and increment the order of the regression model by adding extra terms, i.e., higher-order terms like cubic or higher order interactions. Regression models of higher order require a larger number of samples to estimate the effect of all the terms included in the model.

The minimum number of samples for building a regression model is equal to the number of factors present in the model plus one. However, we suggest to collect a set of additional samples that may vary from 4 to 6 times the number of variables to allow for the values of the SS_T and SS_R to stabilize. At first sight, this iterative approach may seem inefficient, due to the re-sampling of the design region. However, if the design space is sampled using the mixed hypercube approach presented in the previous section, the samples taken in one iteration can be efficiently re-used also for the subsequent one. For continuous variables this is demonstrated in Fig. 4. For discrete variables the possibility of reusing the previous samples to compute new results is due to the deterministic structure of a factorial design. Going from a Resolution 3, to Resolution 4, Resolution 5, or eventually to a full factorial design guarantees that the additional samples are different from the previous ones allowing to maintain the balancing structure of the matrix design.

When working with factorial design, the problem of *aliasing*, or *confounding*, is often experienced. The aliasing effect is the impossibility of discerning the effect of two different factors or interactions of factors. Observing Table 2, it is clear that the effect of factor C is equal to (is confounded with) the effect of interaction AB. In fact, column C is obtained by *xor* operation between columns A and B. In general, for a Resolution 3 design no main effects are confounded with any other main effect, but main effects are confounded with two-factors interactions (and higher order) that may also be confounded with each other. The design in Table 2 is a Resolution 3 design, for instance. For a Resolution 4 design no main effects confounded with any other main effect or with any two-factor interaction, but two-factor interactions can be confounded with each other and with higher-order interactions. Resolution 5 designs allows for experimentation with no main effect or two-factor interaction confounded with any other main effect or two-factor interaction, although two-factor interactions can be confounded with higher-order interactions, (Box et al., 1979) and (Montgomery, 2001). For this motivation, when selecting the type of matrix design for the discrete variables in the mixed-hypercube sampling approach it is necessary to match the *resolution* of the matrix design with the number of samples required to compute the desired effects. For instance, a Resolution 3 design is sufficient to compute linear effects only, more sample points are needed to take into account also the interactions (Resolution 4 and 5 for
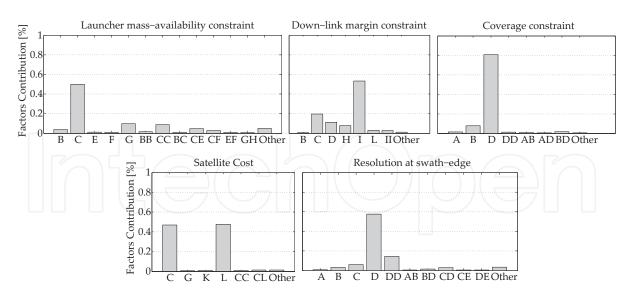
Fig. 7. Bar plots indicating the first-order sensitivity indices computed with the RBSA method.

2-factor interactions, full factorial for higher-order interactions) and, as mentioned already, even more than two levels per variable are required to estimate quadratic effects.

5.3 The Earth-observation mission, sensitivity analysis

In Fig. 7 the results from the sensitivity analysis on the model of the Earth-observation mission, computed using RBSA, are presented. The first-order sensitivity indices are visualized for the constraints (top three graphs) and the objectives (lower two graphs) discussed in Section 3. The results are obtained using a second-order model, see Eq. (3), re-sampled for additional cubic terms of the factors. Two full-factorial designs (3-level and 2-level) have been used for the discrete factors (A) and (B), and (J), (K), and (L), respectively (Table 1). For the continuous variables, instead, the Sobol sequence required 60 samples. The bars represent the percent (divided by 100) contribution of the factors indicated on the horizontal axis of the graphs, their interactions (when the product of two factors is indicated), and their quadratic effects (when the product of the factor by itself is indicated) to the variability of the constraints and the objectives. Cubic effects were limited. Their contribution and the contribution of all the other effects that are not explicitly shown in the bar plots, have been encapsulated in the bars named *Other*.

The first conclusion is that the factors (E), (F), (G), (J), and (K) have a limited effect on the objectives and constraints, probably less then one would expect since some of them are related to the propellant utilization on-board, which is usually a mass driver, thus with an effect on the cost. They can eventually be fixed to a certain level/value with a minor impact on the mission. The other design variables, instead, present contrasting behaviors. The instrument aperture diameter (factor C), for instance, affects the mass of the satellite and the satellite cost (the larger the diameter the larger the mass and the cost, reasonably) but also the down-link margin. The minimum elevation angle for the observation (factor D) has an effect on coverage (the smaller D is, the better) and on the resolution at the edge of the swath (the larger D is, the better). However, factor (D) also has some influence on the down-link margin constraint.

The effect of factors (C) and (D) on the down-link margin constraint, rather than the more obvious impact of the antenna diameter (factor I) and the transmitter RF power output (factor H), can be explained as follows. After these results were obtained, a close investigation on the model lead us to the relationship between the instrument aperture diameter and the *angular resolution*, that is related to the *pixel angular resolution*, thus to the *number of pixels* and finally to the *real-time data rate*, which causes the influence on the link margin. The elevation angle, instead, is related to the atmospheric attenuation that increases as the path to the receiver increase (so as the minimum elevation angle decrease). Many conservative assumptions were made for this applicative case. One of them is actually the fact that communication takes place with a ground station at the edge of the instrument swath width. The results of the sensitivity analysis will be used in the subsequent phase of the design methodology, as presented in the following section.

6. Graphical support to the engineering team

The information gathered during the sensitivity analysis is a roadmap for the engineering team to efficiently direct the design effort. The non-influential design factors can be fixed to a pre-determined level, because they will not affect the performance much, *de facto* reducing the dimensions of the design search-space. However, the influential design variables and the behavior of the system under the effects caused by their variation and their interactions shall be investigated in more detail. Indeed, the same samples used for sensitivity analysis can be used again to compute and present the response surfaces and the variable-trends linking the most influential design factors to the performance, in case of continuous variables. For discrete variables, linear and interaction graphs are computed and presented instead. The design questions that need an answer at this stage of the design process of a complex system are the following:

What is the shape of the design-region? What are the best parameter settings to optimize the objectives and meeting the constraints? What are the best system alternatives?

6.1 Response surfaces for continuous variables

The subject of Response Surface Methods, RSM, includes the procedures of sampling the design space, perform regression analysis, test for model adequacy and optimize the response, (Kuri & Cornell, 1996). The first three steps of the RSM are already in place, as previously discussed. The iterative approach of the RBSA, besides giving quantitative information on the sensitivity indices, also provides the regression coefficients, computed with Eq. (4), related to the best-found sample-fitting regression model. Thus, at this stage of the methodology, a surrogate model that links the design variables to the performance is available, see Eq. (5). Therefore, it is possible to visualize the trends of the objectives and the constraints as a function of the continuous design variables for each combination of discrete-variable levels. Response surfaces, and their bi-dimensional representation called contour plots, can effectively represent the shape of the subspace formed by two continuous variables. When only one continuous variable is of interest, single-variable trends are a valid alternative to contour plots.

Contour plots and single-variable trends could in principle also be computed for discrete variables, since the regression coefficients are available from the RBSA. However, the regression of a continuous function for intermediate discrete-variables levels would not be significant. To visualize the average effect of the discrete variables on the objectives and the constraints, linear and interaction graphs can be computed instead with the method shown in the following subsection.

6.2 Linear and interaction graphs for discrete variables

Consider the analysis of a system with M discrete factors $[A, B, \cdots, M]$, each with a different number of levels $[a, b, \cdots, m]$, and L continuous ones. Thus, there are $M + L = K$ design variables that form a $k - dimensional$ design space. Referring to Figure 6, the matrix-design for the discrete variables would be a $a \times b \times \cdots \times m$ hypercube (considering a full-factorial) while, concerning the Sobol sequence for the continuous factors, let us assume that l sample points are required for each combination of discrete design-variable levels. Once the design space has been sampled and the simulations executed, the responses of the system's model can be analyzed.

Let $Y_{...}$ represent the sum of all the responses obtained during the simulations, $Y_{...} = \sum y = \sum_{i=1}^{a} \sum_{j=1}^{b} \cdots \sum_{w=1}^{m} \sum_{s=1}^{l} y_{ij...ws}$. Let $Y_{i...}$ represent, the sum of all the responses with the factor A at level i, $Y_{i...} = \sum_{j=1}^{b} \cdots \sum_{w=1}^{m} \sum_{s=1}^{l} y_{ij...ws}$.

Considering the values of $Y_{i...}$ normalized with the number of experiments, $n = b \times \cdots \times m \times l$, for which the variable A is at level i, we compute the average value of the performance for A at level i:

$$C_{A_i} = \frac{Y_{i...}}{n} \tag{15}$$

The values of the C_{A_i} plotted against the objectives values provide the so-called linear graphs. Besides showing the trend of the objectives to the variation of a single discrete variable (with all the other variables effects averaged out), they also show the eventual presence of higher order effects, if more than two levels per factor are available from the sampling procedure. In case of ordinal discrete variables, e.g., the number of batteries in a satellite, the higher-order effects may have a certain significance indicating that the performance is not linear with increasing value of that factor. In case of categorical variables instead, e.g., the type of batteries to be implemented in the power subsystem or the type of launcher to be used for the mission, the higher-order effects are not so significant *per se* since there is no *increasing* or *decreasing* direction. This aspect have an implication on the type of matrix design selected for sampling the sub-space formed by the discrete variables only. In principle all the combinations of categorical design factors shall be experimented. Each one of these combinations represent a different system architecture that needs to be explicitly assessed. For the ordinal design factors instead, fractional-factorial designs may suffice to compute their effect on the output due to the fact that these types of variables usually have monotonic trends. However, this does not always have to be the case thus accurate matrix-design selection have to be made by the engineering team depending on the type of problem at hand.

The interaction between two discrete variables can be computed using an approach similar to that used before. For the interaction between factor A and factor B, for instance, a matrix with

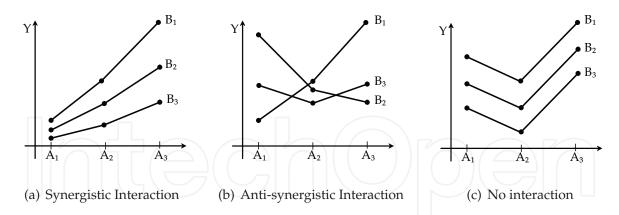(a) Synergistic Interaction   (b) Anti-synergistic Interaction   (c) No interaction

Fig. 8. Interaction graphs with 2 discrete variables at 3 levels. Adapted from (Phadke, 1989).

dimensions equal to $a \times b$ is filled with the following coefficients:

$$C_{A_i B_j} = \frac{Y_{ij\cdots}}{r} \qquad (16)$$

In this case $Y_{ij\cdots}$ indicates the sum of the $r = c \times \cdots \times m \times l$ responses with the factor A at level i and factor B at level j. For each level of A, for instance, b average performance can be plotted against the objectives values, providing the so-called interaction graphs, see Fig. 8. When the lines of an interaction graph are not parallel it indicates the presence of synergistic or anti-synergistic effects, i.e., interactions. A synergistic effect is present when the improvement of a performance given the variation of a factor is enhanced by the variation of another one. An anti-synergistic effect is the exact opposite, (Phadke, 1989). In Fig. 8, the higher-order behavior of the objective to the variation of the variable levels is indicated by the fact that the lines are not perfectly straight over the three levels of variable A, for instance.

The interactions between continuous and discrete variables, eventually detected by sensitivity analysis, can be graphically presented using a mix of contour plots, or single-variable trends, and linear graphs as will be shown in the following subsection.

The synergistic utilization of the results from sensitivity analysis with the RSM and linear and interaction graphs allows the engineering team to study only on the most relevant trends, identified with the sensitivity analysis, and to more effectively select the best combination of design-variable levels. The purpose of this methodology is to support the engineering team and the decision-makers design process and trade-off analysis, and we believe that with this combination of mathematical techniques and graphical results the initial goal is accomplished. However, at this stage of the methodology, the surrogate model could also be used with automatic optimization techniques to provide an optimum (in case of single objective) or a Pareto front of optima (in case of multiple objectives) solutions. A discussion on single- or multiple-objectives optimization techniques is beyond the scope of this chapter. A vast amount of literature dealing with this topic can be found by the interested readers. Coello Coello et al. (2007) and Back et al. (2000), for instance, provide a broad overview and many references.

6.3 The Earth-observation mission, visualization of the design region

The results obtained with the sensitivity analysis in the previous section suggested that some variables influence the objectives and the constraints more than others. This allowed us to reduce the number of important graphs and to focus the attention on only a few of them. Indeed, the graphs in Fig. 9 are an alternative, more detailed and more focused, way of looking at the same data used to compute the sensitivity analysis.

In the interaction graph of Fig. 9(a) the two discrete variables related to the orbit of the satellite are considered. For all the levels of (A) and (B) the average (as previously discussed in this section) value of the equatorial coverage is plotted. The number of days for a repeating ground-track and the total number of orbits in that time period have a synergistic effect on the coverage. In particular, as expected with a higher orbit (e.g., 13 orbits in 1 day and $H = 1258.6$ km) the average equatorial coverage is larger compared to a case with a lower orbit (e.g., 29 orbits in 2 days and $H = 725.2$ km). The combinations of factors levels A1-B3 (i.e., 15 orbits in 1 day), A2-B3 (i.e., 30 orbits in 2 days), and A3-B3 (i.e., 45 orbits in 3 days) lead to the same configuration since the altitude of the orbit is the same, $H = 567.5$ km. The comparison between the performances of an A1-B1 configuration and A3-B2 configuration on the resolution at swath-edge, and on the equatorial coverage, as a function also of the minimum elevation angle and the instrument aperture diameter (factor C) is presented in Fig. 9(b). The light-gray area represents the revisit time constraint for the A3-B2 configuration, set as 100% of equatorial coverage in 24 h. The dark-gray area represents the same constraint for the A1-B1 configuration. A higher orbit (dashed lines in Fig. 9(b)) allows to meet the re-visit constraint with a larger minimum elevation angle thus also improving the resolution performance at the edge of the swath. For the A3-B2 configuration, with $\epsilon = 30°$ and the instrument aperture diameter equal to 0.7 m the resolution at the edge of the swath is 12.7 m/pixel, and 1.26 m/pixel at subsatellite point. For the A1-B1 configuration, instead, the resolution at subsatellite point is slightly worse, i.e., 2.2 m/pixel, but at the edge of the swath a resolution of 7 m/pixel can be obtained. Further, for an A1-B1 configuration, the fact that the minimum elevation angle can be up to 30° gives the satellite the possibility to actually observe over the entire geometrical swath width with the maximum possible slewing angle, i.e., $(E) = 50°$, and at a higher resolution than a A3-B2 configuration. The aperture diameter of the instrument, paradoxically, plays a more relevant role in the determination of the data rate, thus on the down-link margin than on the actual resolution, as demonstrated by the sensitivity analysis. Indeed, in Fig. 9(d) the down-link margin constraint is plotted as a function of the instrument aperture diameter and the minimum elevation angle, for the configuration A1-B1 and with $(H) = 30$ W and $(I) = 1$ m. An A3-B2 configuration would push the coverage constraint down, with the side result of allowing less flexibility in selecting the instrument aperture diameter. The effect on the cost is plotted in Fig. 9(c). The assumption is that a higher orbit, would require less manouvers for pointing the instrument of the satellite in one particular direction and the effect is in a reduced cost (difference between the full and the dashed lines). The constraint on the launcher mass availability is mainly driven by the instrument aperture diameter. Indeed the mass and power consumption of the payload is scaled with the diameter, and so does the mass of the satellite and its cost. The *Delta II* class of launchers allows for enough flexibility until the payload aperture diameter of about 0.9 m.
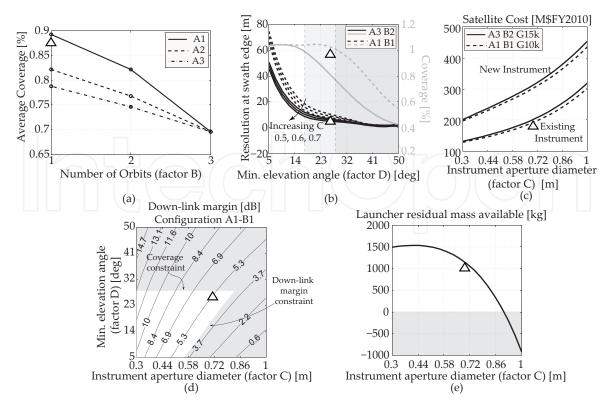
Fig. 9. Analysis main results. Δ is a tentative selected baseline. The light-gray area of (b) represents the revisit time constraint for the A3-B2 configuration, set as 100% of equatorial coverage in 24 h. The dark-gray area of (b) represents the same constraint for the A1-B1 configuration.

The triangles in Fig. 9 represent a tentative selection of the baseline. In particular an A1-B1 architecture has been selected, with $(C) = 0.7$ m, $(D) = 30°$, $(E) = 50°$, $(F) = 120$ s, $(G) = 10000$, $(H) = 30$ W, $(I) = 1$ m, $(J) = 2$, $(K) = 2$, $(L) = 1$. With these settings of the design variables a confirmation experiment was performed on the model. The simulation yield to a cost of the satellite of 188 $M\$FY2010$, a mass of 1330 kg and an overall power consumption of 1 kW. The resolution at the edge of the swath is 7.3 m/pixel and 2.2 m/pixel at sub-satellite point. The equatorial coverage after 24 h is 100% and the down-link margin is 4.1 dB. The results from the verification experiment are very close to the values that can be read from the graphs in Fig. 9. This indicates that the sampling technique and the regression analysis provided reliable results. Sensitivity analysis and graphical support in the form of contour plots, variable trends and interaction graphs enabled a thorough reasoning on the phenomena involved. This allowed us to quickly select a system baseline that meets the constraints balancing the objectives under analysis.

7. Uncertainty analysis and robust design

Uncertainty analysis and robust design are often considered complementary design activities implemented for determining the performances of the system under uncertain operating conditions. In particular, uncertainty analysis is the study of the uncertain distribution characteristics of the model output under the influence of the uncertainty distributions of the model inputs. With these settings, the purpose of uncertainty analysis is to *simply*

propagate the uncertainty trough the model. When the analysis presents both controllable and uncontrollable factors, the latter being intrinsically uncertain parameters (e.g., operating environmental conditions), the purpose of the uncertainty analysis is to obtain settings of the controllable design variables that optimize the performances while at the same time minimize the impact of the uncertainties on the system. In this case we talk about robust design.

In general, uncertainty can be classified in two types, stochastic and epistemic. The stochastic or aleatory uncertainty describes the inherent variability associated with a certain phenomenon. It is usually modeled by stochastic processes when there is enough information to determine the probability distributions of the variables. The epistemic uncertainty is characterized instead by the lack of knowledge about a specific characteristic of the system. If seen in this perspective, the value of the controllable design variables and its related uncertainty can be classified as epistemic and, as discussed in the previous section, these variables are modeled as uniformly distributed between a minimum and a maximum value. However, epistemic uncertainty can also be related to uncontrollable factors for which there is too little information for determining a proper probability distribution. In this case the use of uniform distributions to characterize their uncertainty has been criticized for the main reason that a phenomenon for which there is lack of knowledge cannot be represented by any specific probability distribution (Helton et al., 2006).

For the design of a complex system, in case of both epistemic and stochastic uncertainty, probability theory alone is considered to be insufficient for a complete representation of the implications of the uncertainties on the performances. Therefore, in the following subsection we introduce a unified method for propagating the uncertainty through the model, in the presence of stochastic and epistemic uncertain factors. The main design questions we will try to answer in this section are the following:

In case of uncertainties of any type, how do they propagate through the model of the system? What are the factors that are mostly responsible for performance dispersion? How robust is the design?

7.1 The unified sampling method

In this subsection we introduce a modified implementation of the Sobol sampling technique. A Sobol sequence only allows to sample uniformly on the design space. Uniform distributions are the only necessary distributions of the design variables when the purpose of the analysis is to select a certain baseline that optimize the performances, as discussed in the previous sections. The unified sampling technique, instead, allows to cope with any type of epistemic and stochastic distributions of the uncertain factors, typical when the focus of the analysis is that of propagating the uncertainty throughout the model.

The problem of determining the probability distribution of the output, given the probability distributions of the inputs of a model is related to the computation of a multi-dimensional integral. A direct numerical integration or the analytical solution of the integral can become practically infeasible with already few uncertain variables. Therefore, the direct Monte-Carlo simulation is amongst the most widely adopted methods for uncertainty analysis, since it does not require any type of *manipulation* of the model. When it comes to long-running models, as is usually the case for complex space systems in a collaborative environment, the method of Monte Carlo, using random-sampling techniques, has the recognized disadvantage of
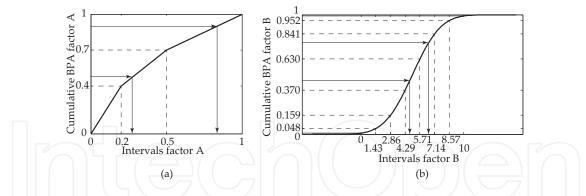
Fig. 10. Representation of the cumulative distributions of (a) the epistemic uncertain variable, and (b) the stochastic (normal) uncertain variable. The dashed lines connect the BPAs to the relative uncertainty intervals. The arrows represent the projection of the sample points from the BPSs domain to the uncertainty-intervals domain.

being computationally expensive, since it generally requires a large number of simulations to compute the mean, the variance and a precise distribution of the response (Rubinstein, 1981). Helton & Davis (2003) compare LHS with a random sampling technique for the propagation of uncertainty into mathematical models. Their analysis corroborates the original results obtained by McKay et al. (1979), and demonstrates that stratified sampling provides more stable Cumulative Distribution Functions (CDFs) of the output than random sampling, with the result that less samples are required for a given accuracy in the determination of the CDFs.

As discussed previously, also epistemic uncertainty must be considered for the design of a complex system. Thus, for the development of the unified sampling technique presented in this section we inherit some ideas and some nomenclature from the evidence theory derived from the initial work of Dempster (1967; 1968) and Shafer (1976). When lack of knowledge about a certain system behavior is present, and when the available historical and statistical sources become sparse, the engineering team is forced to evaluate and combine disparate data sources not perfectly tailored to the purpose at hand based on judgmental elements. Structured expert judgment is increasingly accepted as scientific input in quantitative models, and it is dealt in a number of publications, see for instance (Cooke, 1991) and (O 'Hagan & Oakley, 2004). The result of the combination of experts judgments on the uncertainty of a specific phenomenon leads to the creation, for every single uncertain factor, of so-called *Basic Probability Assignments*, BPAs. The BPAs represent the level of confidence that the engineering team has on the fact that the value of the factor of interest lies in a certain interval of possible values. The uncertainty interval is divided into *n* subsets and for each of them a certain belief, or probability, that the actual value of the uncertain parameter will lie within that subset is assigned. The set of the *n* beliefs form the BPA for the factor under analysis.

Consider for instance the epistemic uncertain factor (A) in Figure 10(a). The uncertainty interval of factor A is equal to $[0, 1]$, divided into 3 subsets $[0, 0.2] \cup [0.2, 0.5] \cup [0.5, 1]$. Suppose that the judgment of the engineering team-members on the uncertainty structure of factor A leads to the conclusion that the actual value of A will lie in the subset $[0, 0.2]$ with a probability equal to 0.4, in the subset $[0.2, 0.5]$ with a probability equal to 0.3 and in the subset $[0.5, 1]$ with a probability of 0.3. Thus the BPA of factor A is equal to $[0.4, 0.3, 0.3]$ and its cumulative function is reported on the y axis of Figure 10(a). The idea is to extend
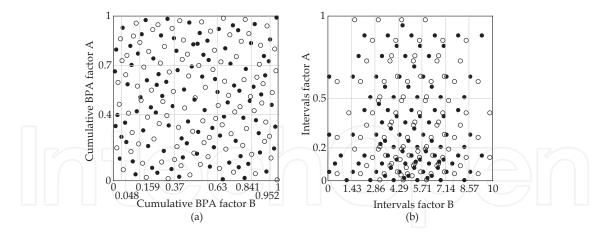
Fig. 11. Unified sampling method. Representation of (a) the uniform sampling in the BPAs domain, and (b) the corresponding sample points in the uncertainty-intervals domain.

the concept of the BPA also to the stochastic variables in such a way to obtain a unique representation of the uncertainty structure of the inputs. For a stochastic variable the cumulative distribution function is continuous. If the uncertainty interval of the stochastic factor is discretized in m subsets, then the discretized CDF can be expressed in the form of Basic Probability Assignments as in case of the epistemic uncertain factors. Consider the normally distributed uncertain factor (B) of Figure 10(b). Its uncertainty interval is equal to $[0, 10]$, divided in 7 subsets, for instance, as reported on the x axis of Figure 10(b). According to the CDF of the normal distribution, the BPAs associated to this discretization are the following $[0.0480, 0.1110, 0.2106, 0.2608, 0.2106, 0.1110, 0.0480]$. The cumulative BPAs are reported in the y axis of Figure 10(b). In the case of stochastic uncertainty, there is the possibility of having infinite tails of the distributions, as in the case of the normal one. However, if the minimum and the maximum values of the uncertainty intervals represent a high percentile, e.g., 0.95 and 0.05, or 0.99 and 0.01 (as in the case of factor A), the error is acceptably small in most of the cases. In Figure 10(b) the gray areas represent the error that arise when considering a truncated normal distribution. The probabilities of the first and the last intervals are overestimated of a quantity equal to the smallest truncation percentile (0.01 in this case).

The unified sampling method is executed in two steps. First, a uniform sampling on the space formed by the cumulative values of the BPAs is executed, Fig. 11(a). Then, each sample point within each interval in the BPA domain is mapped to the corresponding point in the uncertainty interval domain, Fig. 11(b). This passage from the BPAs domain to the uncertainty intervals domain is also represented by the arrows in Figure 10. With this procedure the final sample is collected according to the aleatory/epistemic probability distribution of the factors.

Experience and common sense tells that the more the BPA intervals, the better the approximation of the output probability distribution. However, in the case of epistemic-factor uncertainty the number of BPA intervals depends on the degree of knowledge of the engineering team on the behavior of the factors themselves. If the initial uniform sampling is performed according to a stratified technique, the resulting response CDF will be more stable than what could be obtained by using a random technique, as demonstrated by Helton & Davis (2003) and McKay et al. (1979). Further, if a Sobol sequence is implemented all the advantages already discussed in the previous chapters would still hold. This is particularly

true if seen in the perspective of computing the sensitivity analysis using the RBSA, which is directly applicable if the unified sampling method is used. The computation of sensitivity analysis under uncertainty settings allows to identify the contribution of the inputs to the uncertainty in the analysis output, so to drive the effort in better describing the uncertainty of only the most relevant factors.

7.2 The Earth-observation mission, uncertainty analysis for cost, mass and power budgets

In the traditional system engineering process, design margins are used to account for technical budget uncertainties, e.g., typically for cost, mass and power. A certain percentage of the baseline's performance is added to account for both uncertainties in the model and uncertainties about eventual assumptions made at a preliminary phase that will likely be modified in advanced phases, due to an increased level of detail and knowledge. For instance, the results presented in the previous section were obtained with a 15% margin on the total satellite mass, total power consumption and propellant stored on board. The results without margins would be different. In particular, the satellite mass would be equal to 1048 kg, the power consumption euqual to 830 W and with a cost saving of 15 $M\$FY2010$. The unified sampling method allows the engineering team to obtain more insight in the uncertainty structure of the solution by focussing on every single source of uncertainty. This will enable a more informed decision-making process on the allocation of the budgets to each subsystem and each element.

In the case of the Earth-observation mission we considered the uncertain parameters and the uncertainty structure presented in Table 3. A mix of normal, log-normal and epistemic distributions has been considered. The normal and the log-normal uncertain variables are centered on the values needed to obtain the results presented before. The epistemic uncertain intervals and BPAs are determined in such a way that the value of the factors needed to obtain the previous results is at the center of the first epistemic interval. Using the unified sampling method, with 200 samples, we obtained the results shown in Fig. 12. In Fig. 12(a,b,c) the probability density estimates of the performances are presented. The histograms are plotted with an adjusted scale, so to obtain a total area of the bars equal to 1. The black and gray arrows are positioned in correspondence to the values of the performance previously computed with and without margins, respectively. It is clear that the margins approach does not provide the same insight provided by the PDFs and the histograms on the performances of the system under uncertain factors. In particular, the PDF and CDF trends shown in Fig. 12 allow the engineering team to better understand the behavior of the system under analysis, bringing two main advantages. First, the uncertainty can be allocated to single subsystems and single elements more effectively using the unified sampling method. Second, the final performance can be precisely assessed according to the desired confidence level. Further, having a precise distribution of the performances allows for more effective budget-allocation management for subsequent phases of the design process. In Fig. 12(d,e,f) the empirical cumulative distribution functions of the performances are presented. The CDF estimate, computed with 2000 samples using a random sampling method, is also represented. The fact that the empirical CDF and the CDF estimate are very close to each other corroborates the initial statement that the unified sampling method, being a stratified sampling method, is able to provide accurate results with a reduced computational effort.

| Uncertain Variables | Intervals | | Distribution | Uncertain Variables | Intervals | | Distribution |
	Min	Max			Min	Max	
Margin δV [%]	0	0.25	Epistemic[a]	Solar array power dens. $[W/Kg]$	90	110	Normal[d]
Specific Impulse $[s]$	280	320	Normal[d]	Batteries energy dens. $[W-h/Kg]$	25	75	Normal[d]
Thrusters inert mass fraction [%]	0.2	0.4	Epistemic[b]	PCU mass $[Kg]$	27	50	Log-normal[e]
ADCS sens. mass $[Kg]$	58	70	Log-normal[e]	Regulators mass $[Kg]$	33	55	Log-normal[e]
ADCS sens. power $[W]$	33	45	Log-normal[e]	Thermal subs. mass $[Kg]$	20	50	Log-normal[e]
Antenna mass density $[Kg/m^2]$	9	11.5	Normal[d]	Struct. mass margin [%]	0	1	Epistemic[c]
Solar cells η [%]	0.17	0.23	Normal[d]				

Table 3. Settings of the design variables.[a]Intervals $[0, 0.04, 0.1, 0.17, 0.25]$, BPA $[0.4, 0.3, 0.2, 0.1]$. [b]Intervals $[0.2, 0.25, 0.3, 0.4]$, BPA $[0.4, 0.35, 0.25]$. [c]Intervals $[0, 0.25, 0.5, 0.75, 1]$, BPA $[0.4, 0.3, 0.2, 0.1]$.[d]$\mu = 0$ $\sigma = 1$, Min and Max are the 0.01 and 0.99 percentile respectively.[e]$\sigma = 1$, Max is the 0.99 percentile, Min corresponds to $X = 0$.

Fig. 12. Uncertainty-analysis results

7.3 Robust design and the augmented mixed hypercube approach

Robustness is a concept that can be seen from two different perspectives, at least according to the discussion so far. One can define robustness of the system with respect to the effect of uncontrollable factors (aleatory and/or epistemic) and, if interested in obtaining a robust design, one can select that combination of controllable design-factor values that minimizes the variance while optimizing the performance. This concept was already expressed in the previous section, and it is the most common way of thinking of robust design. However, robustness can also be defined as the insensitivity of a certain design baseline to modification of the design variables in subsequent phases of the design process, thus providing an intrinsic design-baseline robustness figure. The modification of the levels of the design variables is likely to happen, especially when the baseline is at an early stage of the design process (phase 0/A). In this sense, robustness can be linked to the programmatic risk encountered when modifying a set of design parameters at later stages of the design process. In the first case, instead, robustness is more related to the operational-life risk of the system (if the uncertainties derive from the operational environment, for instance).

In both cases the Mixed Hypercube approach and the unified sampling method, and the utilization of the design techniques proposed in this chapter, provide a valuable tool in the hand of the engineering team. The sampling approaches described in this chapter are summarized in Fig. 6 and Fig. 13. When the purpose of the analysis is to study the best settings of the controllable design variables to optimize the performances while meeting the constraints, the mixed hypercube approach (see Fig. 6 in conjunction with RBSA, response surfaces and linear and interaction graphs) provide a way to answer many of the most common design questions. When the purpose of the analysis is to obtain a robust design, thus studying the settings of the controllable design factors that optimize the performances while keeping the system insensitive to uncertain factors, then the Augmented Mixed Hypercube, AMH, approach shall be used, see Fig. 13(a). For every combination of the levels of the controllable design variables, an uncertainty analysis can be executed using the unified sampling method to obtain the performance of the system, and the relative statistics, due to uncertain factors. When, instead, the effect of the modification of the controllable design variables in later stages of the design process is under investigation, the general case presented in Fig. 13(b) can be implemented. The variables used to determine the baseline can be studied in perspective of their uncertain future variation. The continuous variables are more likely to be modified, since the discrete ones commonly represent different architectures of the system (whose change usually bring more radical modifications of the design, thus most likely high costs). However, in general a figure of robustness can be computed for each combination of discrete-factor levels. The approach in Fig. 13(b), without architectural variables, was also used for the budget-margins analysis presented in the previous section.

One last remark regards the possibility to use the Augmented Mixed Hypercube for a wider search. The analysis performed with the AMH, as presented in this chapter, is restricted to the portion of the design space delimited by the variability ranges of the design variables. Sometimes a single hypercube is sufficient to entirely cover the design space, sometimes instead a narrower hypercube might be needed to avoid major lack-of-fit conditions. In this case more than one hypercube may be implemented to study different regions of the design space as different alternative baselines of the system. In this case, the methodologies presented
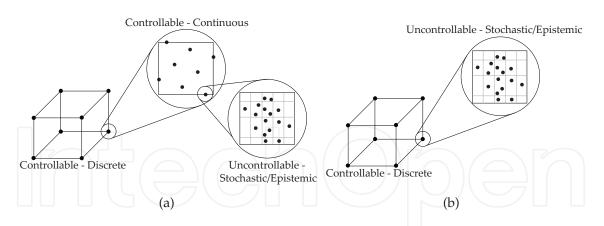
Fig. 13. Augmented Mixed Hypercube sampling procedure for robust design.

in this chapter will not only support the engineering team in selecting the best configuration for each single baseline, but will also allow to compare and trade between the baselines based on their performances, constraint-violation conditions and robustness.

8. Conclusions

Design-space exploration is the fundamental activity with which the model of a complex system is analyzed to understand the effect of the design choices on the performance(s) and to set the values of the variables in such a way that the final product will perform as required by the customer(s). This activity often involves many stakeholders, with many objectives to be balanced, many constraints and many design variables, thus posing the problem to be extremely difficult to solve with a non-structured approach. The purpose of this chapter was to discuss on subsequent analysis steps and synthesis methodologies that could serve as a guideline for exploring the design space of complex models in a standardized and possibly more efficient way. In particular, the goal was to bring fundamental analysis techniques from the discipline domain level to the system domain level. This is done to support the decision-making process and provide a unified design approach that could be implemented in a collaborative environment, in the presence of long-running models with limited time available for the design process. For this motivation, all the proposed techniques do not require any kind of manipulation of the original model and they are developed pursuing the reduction of the required computational effort as much as possible.

The Augmented Mixed Hypercube approach developed and presented step-by-step in this chapter demonstrated to be a flexible sampling method with which many fundamental design questions could be answered. The AMH is slightly more elaborated than other conventional sampling techniques but it allows the engineering team to gain a great deal of insight in the problem at hand with continuous and discrete, controllable and uncontrollable design variables with one unified method. The final baseline of the Earth-observing satellite, for instance, was selected according to a non-conventional mission architecture for an observation satellite, i.e., quite a high orbit altitude. This choice was mostly driven by the need to balance the coverage requirement and the resolution performance, while keeping the cost down. The *risk* of obtaining conventional design baselines is behind the corner when non-structured, expert-judgment driven, approaches are implemented. However, very often, especially in preliminary design phases, expert judgment is a fundamental ingredient to a good system

baseline. In fact the AMH also allows to take expert-judgment into account with a unified epistemic-stochastic sampling approach. The Regression Based Sensitivity Analysis presented in this chapter, coupled with the AMH, allows to compute global variance-based sensitivity indices with a reduced computational effort if compared to other global sensitivity analysis methods. The great advantage of sensitivity analysis performed already at an early stage of the design process, as demonstrated with the Earth-observation mission, is that it could speed up the process itself providing the engineering team with an *X-ray machine* that allows to efficiently understand the effect of their design choices on the system.

We would like to close this chapter with few final thoughts on possible implementation of the methods proposed here. In case of low-complexity systems, when few variables are under analysis, and when previous experience on similar systems is present, these methods could be used as a confirmation of the expected trends, or as a proof for the analysis underlying assumptions. For complex and new systems, the implementation of the methods could reduce the engineering-team effort in exploring different solutions and architectures. In the cases where very experienced specialists are present within the engineering team (that would probably have already a clear picture of the priorities of the factors for the inherent problem), the standardized graphical approach could be a valid tool for them to explain thoughts and solutions. However, understanding performance trends in the presence of constraints and multiple objectives beforehand could also for them be a non-trivial task. On the other hand, the less experienced team members could benefit from the tool even with easy problems and expected behaviors, thus improving the overall design process, quality and effectiveness.

The contribution of the human factor is fundamental for obtaining a final product with a high cost/effectiveness value. With the integrated design approach presented in this chapter we do not mean to substitute the humans in the process of designing but, quite on the contrary, to better support their activities.

9. Acknowledgments

10. References

Antonov, I. & Saleev, V. (1979). An economic method of computing lpt sequences., *USSR Computational Mathematics and Mathematical Physics* 19(1): 252–256.

Back, T., Fogel, D. & Michalewicz, Z. (2000). *Evolutionary Computation.*, Vol. 1-2, Institute of Physics Publishing, Bristol.

Box, G., Hunter, W. & Hunter, J. (1979). *Statistics for Experimenters. An Introduction to Design, Data Analysis and Model Building*, Wiley, New York.

Box, G. & Wilson, K. (1951). On the experimental attainment of optimum conditions., *Journal of the Royal Statistical Society* 13(B): 1–45.

Bratley, P. & Fox, B. (1988). Implementing sobol's quasirandom sequence generator, *ACM Transactions on Mathematical Software* 14(1): 88–100.
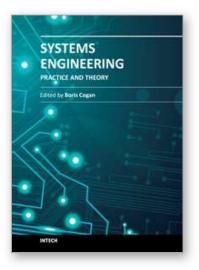
Coello Coello, C., G.B, L. & Van Veldhuizen, D. (2007). *Evolutionary Algorithms for Solving Multi-Objective Problems.*, Springer Science and Business Media, New York, NY. Genetic and Evolutionary Computation Series.

Cooke, R. (1991). *Experts in Uncertainty.*, Oxford University Press, New York, NY.

Cramer, E. J., J. E. Dennis, J., Frank, P. D., Lewis, R. M. & Shubin, G. R. (1993). Problem formulation for multidisciplinary optimization, *AIAA Symposium on Multidisciplinary Design Optimization*, Houston, TX. CRPC-TR 9334.

Cukier, R., Levine, H. & Shuler, K. (1978). Nonlinear sensitivity analysis of multiparameter model systems, *Journal of Computational Physics* 1(26): 1–42.

Dempster, A. (1967). Upper and lower probability inferences based on a sample from a finite univariate population., *Biometrika* 2-3(54): 515–28.

Dempster, A. (1968). A generalization of bayesian inference., *Journal of the Royal Statistical Society* (30): 205–47.

Helton, J. & Davis, F. (2003). Latin hypercube sampling and the propagation of uncertainty in analyses of complex systems., *Reliability Engineering and System Safety* 1(81): 23–69.

Helton, J., Johnson, J., Oberkampf, W. & Sallaberry, C. (2006). Sensitivity analysis in conjunction with evidence theory representations of epistemic uncertainty., *Reliability Engineering and System Safety* 1(91): 1414–1434.

Iman, R. & Conover, W. (1982). A distribution-free approach to inducing rank correlation among input variables., *Commun. Statist. - Simula. Computa.* 3(B11): 311–334.

International Council on Systems Engineering [INCOSE] (2007). Systems engineering vision 2020, *Technical Report INCOSE-TP-2004-004-02*. Available from `http://incose.org/ProductsPubs/products/sevision2020.aspx`.

Kuri, A. & Cornell, J. (1996). *Response Surfaces. Design and Analyses*, Marcel Dekker, Inc., New York. Second Edition, Revised and Expanded.

McKay, M., Beckmank, R. & W.J., C. (1979). A comparison of three methods for selecting values of input variables from a computer code, *Technometrics* 21: 239–245.

Mistree, F., Lautenshlager, U., Erikstad, S. & Allen, J. (1994). Simulation reduction using the taguchi method., *NASA Contractor Report* (CR-93-4542): 252–256.

Montgomery, D. (2001). *Design and Analysis of Experiments*, John Wiley & Sons, New York. Fifth Edition.

Morris, M. (1991). Factorial sampling plans for preliminary computational experiments, *Technometrics* 33(2): 161–174.

O 'Hagan, A. & Oakley, J. (2004). Probability is perfect, but we can 't elicit it perfectly., *Reliability Engineering and System Safety* (85): 239–248.

Phadke, M. (1989). *Quality Engineering Using Robust Design.*, Prentice-Hall, Englewood Cliffs, NJ.

Press, W., Teukolsky, S., Vetterling, W. & Flannery, B. (2007). *Numerical Recipes. The Art of Scientific Computing*, Cambridge University Press, New York. Third Edition.

Rubinstein, R. (1981). *Simulation and the Monte Carlo method.*, Wiley, New York, NY.

Saltelli, A., Tarantola, S., Campolongo, F. & Ratto, M. (2004). *Sensitivity Analysis in Practice*, John Wiley & Sons Ltd, Chichester, West Sussex, England.

Shafer, G. (1976). *A mathematical theory of evidence*, Princeton University Press, Princeton, NJ.

Sobieszczanski-Sobieski, J. (1989). Multidisciplinary optimization for engineering systems: Achievements and potential, *Technical Report NASA Technical Memorandum 101566*, NASA Langley Research Center.

Sobieszczanski-Sobieski, J. & Haftka, R. T. (1995).   Multidisciplinary aerospace design
        optimization: Survey of recent developments, *34th Aerospace Sciences Meeting and
        Exhibit*, Reno, Nevada. AIAA 96-0711.

Sobol, I. (1979). On the systematic search in a hypercube., *SIAM Journal on Numerical Analysis*
        16(5): 790–793.

Sobol, I. M. (1993).   Sensitivity analysis for nonlinear mathematical models, *Mathematical
        Modeling and Computational Experiment* 1: 407–414.

Sobol, I. M. (2001).   Global sensitivity indices for nonlinear mathematical models and their
        monte carlo estimates, *Mathematics and Computers in Simulation* 55: 271–280.

Taguchi, G. (1987).   *System of Experimental Design. Engineering Method to Optimize Quality and
        Minimize Costs.*, UNIPUB/Kraus International Publications, New York.

Tedford, N. P. & Martins, J. R. (2006).   On the common structure of MDO problems:
        A comparison of architectures, *11th AIAA/ISSMO Multidisciplinary Analysis and
        Optimization Conference*, Portsmouth, VA. AIAA 2006-7080.

Viana, F., Venter, G. & Balabanov, V. (2010). An algorithm for fast optimal latin hypercube
        design of experiments, *Int. J. Numer. Meth. Engng* (82): 135–156.

Wertz, J. & Larson, W. (1999). *Space Mission Analysis and Design*, Springer, New York.

Yi, S. I., Shin, J. K. & Park, G. J. (2008).   Comparison of MDO methods with mathematical
        examples, *Structural and Multidisciplinary Optimization* 35: 391–402.

Systems Engineering - Practice and Theory

Edited by Prof. Boris Cogan

The book "Systems Engineering: Practice and Theory" is a collection of articles written by developers and researches from all around the globe. Mostly they present methodologies for separate Systems Engineering processes; others consider issues of adjacent knowledge areas and sub-areas that significantly contribute to systems development, operation, and maintenance. Case studies include aircraft, spacecrafts, and space systems development, post-analysis of data collected during operation of large systems etc. Important issues related to "bottlenecks" of Systems Engineering, such as complexity, reliability, and safety of different kinds of systems, creation, operation and maintenance of services, system-human communication, and management tasks done during system projects are addressed in the collection. This book is for people who are interested in the modern state of the Systems Engineering knowledge area and for systems engineers involved in different activities of the area. Some articles may be a valuable source for university lecturers and students; most of case studies can be directly used in Systems Engineering courses as illustrative materials.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Guido Ridolfi, Erwin Mooij and Sabrina Corpino (2012). Complex-Systems Design Methodology for Systems-Engineering Collaborative Environment, Systems Engineering - Practice and Theory, Prof. Boris Cogan (Ed.), ISBN: 978-953-51-0322-6, InTech, Available from: http://www.intechopen.com/books/systems-engineering-practice-and-theory/complex-systems-design-methodology-for-systems-engineering-collaborative-environment

INTECH
open science | open minds