THÈSE

Présentée et soutenue par **Jessie BIRMAN**
le **19 décembre 2013**

Uncertainty quantification and propagation in Conceptual Aircraft Design: from deterministic optimisation to chance constrained optimisation

Quantification et propagation d'incertitude dans les phases amont de projets de conception d'avions : de l'optimisation déterministe à l'optimisation sous contraintes probabilistes

JURY

Laetitia Andrieu	Membre du Jury
Chef de groupe EDF-R& D Osiris (Clamart)	
Christian Bes	Président du Jury
Professeur, ICA Institut Clément Ader (Toulouse)	
Thierry Druot	Invité
Expert Ingénieur, Airbus Opération S.A.S. (Toulouse)	
René Henrion	Rapporteur
Professeur, Weierstrass Institute(Berlin)	
Jean-Baptiste Hiriart-Urruty	Directeur de thèse
Professeur, Institut de Mathématiques (Toulouse)	
Sophie Jan	Co-directeur de thèse
Maître de conférence, Institut de Mathématiques (Toulouse)	
Abdel Lisser	Rapporteur
Professeur, Université de Paris-Sud	

Nous cherchons la vérité, et ne trouvons en nous qu'incertitude.

Blaise Pascal

Remerciements

La thèse a été une expérience très riche tant sur le plan professionnel que sur le plan personnel. J'ai fait de nombreuses rencontres et je souhaite donc remercier toutes les personnes que j'ai pu côtoyer, de loin ou de près, durant ces trois dernières années.

Je souhaite dans un premier temps exprimer mes remerciements sincères à mes encadrants Thierry Druot (superviseur industriel), Jean-Baptiste Hiriart-Urruty et Sophie Jan (surperviseurs universitaires). Le travail accompli n'aurait pas été possible sans eux. Thierry, je te remercie pour tout le temps que tu m'as accordé afin de m'expliquer les divers aspects d'un dimensionnement avion. Je te remercie également pour ton soutien et tes encouragements constants. Sophie et Jean-Baptiste, un énorme merci pour vos conseils, vos relectures, et pour votre aide précieuse pendant la dernière ligne droite.

Je tiens également à remercier Christian Bès avec qui j'ai collaboré pendant les deux premiers mois de la thèse et qui m'a apporté une aide précieuse.

Mer remerciements vont également aux rapporteurs, René Henrion et Abdel Lisser, pour le temps qu'ils ont accordé à la lecture du manuscrit ainsi que pour leurs remarques.

Laetitia Andrieu, qui a toujours manifesté son intérêt pour les problèmes sous contraintes probabilistes, a accepté de participer au jury ; qu'elle en soit vivement remerciée.

Je souhaite remercier Phillippe Escarnot, Vincent Rivoire, Yannick Caillabet et Christophe Blondel de m'avoir accueillie dans leur équipe des Avant-projets à Airbus.

Je remercie également l'équipe des Avant-projets pour ces trois belles années pendant lesquelles j'ai beaucoup appris sur leur métier. C'est une équipe exceptionnelle. Merci à vous tous pour tous ces bons moments passés ensemble. Je remercie particulièrement Sole Martin-Hernandez, Michel Maspimby, Catherine Farjou, Céline Badufle et Serge Bonnet pour leurs soutien et conseils. Je tiens à remercier Sylvain Prigent qui m'a apporté une aide précieuse sur la propagation des moments.

Je remercie ensuite mon conjoint ainsi que mes amis qui ont toujours cru en moi et qui ont toujours été présents, pendant les bons et les mauvais moments.

Enfin, j'exprime un profond remerciement à ma famille, particulièrement à ma mère et à mes petites soeurs qui m'ont énormément encouragée. Elles sont une source certaine de force inépuisable.

Contents

List of Figures

List of Tables

Glossary

BIC Bayesian Information Criterion. 77

BPR By-Pass Ratio. 15, 29

CDF Cumulative Distribution Function. 53, 54, 59, 62, 82, 141, 142, 145, 146

COC Cash Operating Cost. 15, 28, 123

DOC Direct Operating Cost. 15, 28

FAR Federal Aviation Regulations. 8

FPO Future Project Office. 8, 25

ICA Initial Cruise Altitude. 10, 28

ICAO International Civil Aviation Organization. 10

ID4CS Integrated Design For Complex System. 34

ISA International Standard Atmosphere. 27

JAR Joint Airworthiness Regulation (Joint Aviation Requirement). 8

LW Landing Weight. 11

MFC Maximum Fuel Capacity. 10, 28

MLW Maximum Landing Weight. 12, 13

MRW Maximum Ramp Weight. 12

MTOW Maximum Take-Off Weight. 12, 13, 15, 16, 18, 21–25, 28, 40, 109, 111, 128

MWE Manufacturer Weight Empty. 11, 20–23, 123, 125, 146

MZFW Maximum Zero Fuel Weight. 12, 13, 31

OAD Overall Aircraft Design. 7, 8, 23, 25

OIW Operational Items Weight. 20

OWE Operator Empty Weight. 11–13

PDF Probability Density Function. 54, 59, 61, 62, 76, 82, 144, 145

Introduction

This Ph.D. thesis has been performed in the frame of a CIFRE contract in partnership with the Future Project Office (FPO) of Airbus France. CIFRE is a French acronym and stands for Conventions Industrielles de Formation par la REcherche.
Passenger transport aircraft is probably one of the most complex industrial products. With no surprise, the associated design process reflects this complexity. This process includes quite a lot of phases which can be grouped within three main stages: Conceptual Design, Preliminary Design and Detailed Design. This study is focused on the first phase where very impacting decisions are taken while very few information are available about the product itself.

Conceptual Design is carried out by the Future Project Office. The objective at this stage is to perform overall aircraft sizing (versus a given set of design and operational requirements) of several possible configurations in order to determine the best ones according to strategic and marketing criteria. This overall sizing is classically achieved through a global optimisation process under constraints. In most of the cases, many aircraft configurations have to be analysed and ranked according to various criteria which requires very time efficient computational processes. During conceptual design, only the main characteristics of the aircraft are defined. The outcome of this conceptual phase is a selection of very few number of airplane configurations (most often 2) that will be studied in more details in a preliminary design phase. Preliminary design phase is still based on simplified models so that global Multidisciplinary Optimisation could be achieved. The global computational process is very similar in Conceptual and Preliminary phases, only models are more refined in Preliminary Design, we use to speak about Overall Design Process. This second step involves FPO as well as some specialised departments that will work to reduce the uncertainty on some risky assumptions. After selection of the best candidate, the project may evolve towards detailed design phases. It is important to highlight one of the main difficulties of early design steps: in spite of a very limited knowledge available on the product, decisions that are taken have the greatest impact on further developments.

The Overall Design Process is complex. It is, at the same time, a:

- **constrained optimisation process**. It can be seen as a global optimisation where one seeks for the optimal aircraft configuration that allows minimising a given criterion (such as the weight) under operational and cost requirements. Optimisation methods are also used for other purposes within the preliminary design process: for example, they are involved in models calibration and are necessary to compute most of simulated operational performances of the aircraft (as Take Off Field Lengths or Ceilings).

- **multidisciplinary process**. An aircraft is a product where several physical fields are involved and interact such as aerodynamic, handling qualities, propulsion, weights or structure. All of these physics have to be represented when designing an aircraft, even in a preliminary study.

- **simulating-based process**. As many other engineering processes, aircraft design process is based on the use of numerical models that mimic the physics, in the most rigorous way.

The level of fidelity of these models varies depending on where we are in the design process and on the frequency of use of the model. During the last decade, FPO engineers have developed a platform providing models and processes which answer to the various aspects of preliminary aircraft design process. This platform has evolved as knowledge on the process increased.

Numerical models are generally constituted of a set of input parameters. For time saving, this set is generally a subset of the group of parameters which are involved in reality. The less parameter the model has, the less precise it will be. Then, the model may have a lack of accuracy because all physics are not represented. Consequently, there generally exist a discrepancy between the model estimation and experiments.

Aircraft design is classically based on a deterministic process: it assumes that there is no uncertainty. Engineers are used to managing the risk by launching sensitivity analysis that can possibly lead to take margins on some design parameters. Uncertainty is able to modify the response of a system and can degrade its performance, thus it is important to manage it as soon as possible. The objective of this study is to propose a new Design Philosophy based on robust optimisation technique in order to manage and integrate uncertainty into the existing aircraft design process. This approach requires that the main source of uncertainty is identified, quantified. Then, it can be propagated through the simulation code to launch robustness analysis. Moreover, since the involved models are simple and rapid to compute, quite a lot of evaluations can be made and classical techniques can be used to perform the optimisation.

Chapter 1 provides more details on the context of the study. The Future Project Office role and objectives are further explained, the stakes and the core of preliminary aircraft design process are presented (description of the simulation models, etc) and the strategy used by FPO engineers to manage risk and insure robustness is described. The general context, the industrial's stake and the aim of the thesis are detailed at the end of this chapter.

Chapter 2 deals with the state of the art of the fields related to the thesis study such as optimisation, uncertainty quantification, robust optimisation and response surfaces. This chapter will help in finding methods that can be applied to aircraft design.

Chapter 3 is about the uncertainty quantification. First, the uncertainty involved at prelim inary aircraft design is identified. Then, the selected method used to quantify this uncertainty is presented. A new generic distribution probability is introduced as well as the reasons motivating its set up. Moreover, focus is put on the difficulty to quantified uncertainty related to models with nested sub-models. A strategy is adopted to escape this difficulty and to be able to identify uncertainty for each model separately. A succinct correlation study is presented. In the last part of this chapter, an interpretation of the uncertainty quantified in terms of aircraft design freedom is presented.

Chapter 4 deals with uncertainty propagation through the aircraft simulation code. First, a short state of the art of existing propagation methods is presented. In this study, put the focus on Monte Carlo and moment propagation methods. We present the moment propagation method as well as the various steps involved in its implementation. Then, for a comparison sake, the robustness of the original process used by FPO engineers to manage risk is analysed by launching uncertainty propagation thanks to these two methods. In the last part of this chapter, we present two examples in which the moment propagation is used.

Chapter 5 aims to present the robust optimisation for aircraft design. A selection is done among the set of robust optimisation methods, the formulation of the robust aircraft design optimisation is described and the selection of an algorithm to solve the optimisation is justified.

The proposed methodology is tested on a twin-jet design problem. The results are presented and discussed.

Conclusion and perspectives of the thesis are presented at the end.

Chapter 1

Overall Aircraft Design in Future Project Office

Contents

1.1   Future Project Office

While being very similar to any other product development (see [Geb95]), the complexity of the Aircraft Design Process is in proportion to the complexity of the final product:

1. Idea generation (basic design)

2. Planning and definition (definition of the main function of the product)

3. Design (evaluation and definition of the project's goal)

4. Development/Realization: drafting, design, calculation (raw and detailed design, calculation and evaluation)

5. Test run

Time elapsing between the idea generation and test run varies according to the product development and is roughly proportional to the life of the product. In aircraft design context, this time is quite long. To give some idea, we can do the parallel with other product such as mobile phones, computers and cars (figure 1.1 and table 1.1). A mobile phone can be developed in about 2 to 5 months and will be in use a very few years (or even months) whereas an aircraft is developed in about 120 months (10 years) and may be in use other more than 50 years! This large design time is also related to the great amount of safety rules to satisfy.
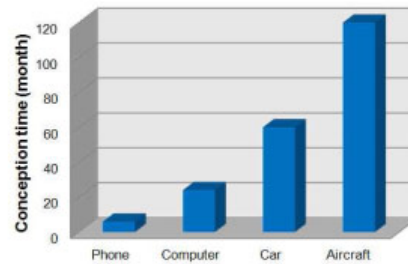
Figure 1.1: Comparison of time conception for different products

	Computer	Automobile	Aircraft
Pre-design	2	5	10
Production	3	5	20
Operating	5	10	30
Life cycle	10	20	60

Table 1.1: Product life cycle comparison

The aircraft design process can be split in four phases: feasibility, conceptual, definition and production.

These four phases are marked by a serie of important events called Maturity Gate (MG). MG are indicators of the evolution of the project. They represent the change from one status of the project to another one. Some MGs are particularly important. As an example, MG3 implies that the concept is frozen which has a great impact on future work and product life. Figure 1.2 presents the aircraft design process and its different MGs.

In this study, we will focus mainly on the first phase of the design process which is the feasibility phase. During this phase, the following actions are managed:

- Scenario analysis (corresponding to MG1),
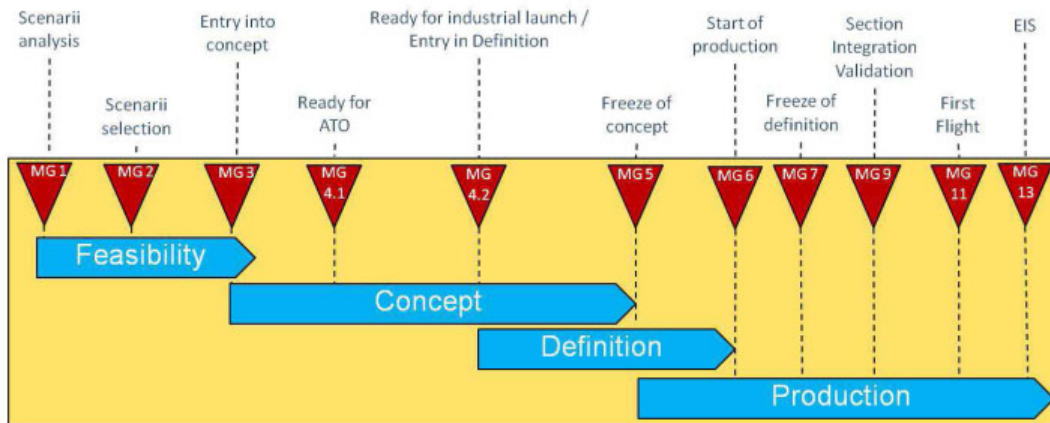
Figure 1.2: Aircraft design process maturity gates

- Scenario selection (corresponding to MG2),

- Finding out which product capabilities are really needed to make it attractive,

- Finding out which product concept does best address these capabilities,

- Definition of the specifications,

- Finding out how one could best make such product (technical, companies, partnership...),

- Entry into concept (corresponding to MG3).

The final goal of all these actions is the realisation of Overall Aircraft Design (OAD). OAD refers to the global definition of the aircraft according to a set of given requirements. The objective for engineers is to find, among all possible aircraft configurations, the 'best' one that allows fulfilling the requirements while optimising the selected criteria. Figure 1.3 presents, in a simple way, the process which leads to this 'best' aircraft configuration.
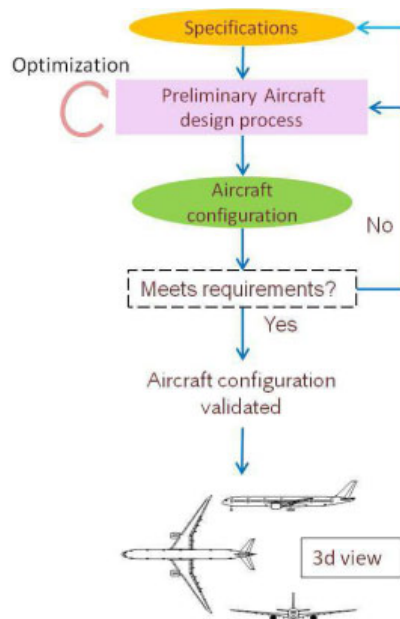
Figure 1.3: Process for future project conception

It is not a linear process: number of iterations has to be made in order to find the optimum design or, at least, to be convinced that the selected configuration is a pretty good one. Note that, OAD is not only dedicated to new aircraft project. It can be used when the evolution of an existing aircraft is at stake.

Future Project Office (FPO) plays a crucial and a major role in the aircraft design phase. As FPO is mainly involved at the beginning of the aircraft design process, its presence decreases as long as the project evolves to vanish at the beginning of the production phase. Consistently with the overall definition of the aircraft, FPO role is also to provide initial technical targets to the different groups that are in charge of component design (wing, fuselage, landing gear, empennage,...) .These groups take over at the detailed design phase starting between MG3 and MG4.

In the following section, we give further detail on the different items which composed the OAD process in figure 1.3.

1.1.1   Top Level Aircraft Requirements

Top Level Aircraft Requirements (TLARs) are supposed to be the most structuring requirements that will give shape to the future aircraft. They form the specifications and are the input of the design process. Actually, there are two types of requirements: flexible requirements (the aircraft "should" reach them) and hard one (the aircraft "shall" fulfil them). The selection of a well balanced set of TLARs is an essential step and also a tricky one as a clear view on commercial and economical stakes is necessary. A wrong set of TLARS may as well drive to an unfeasible aircraft or economically not viable. This selection is the result of quite a lot of iterations between several entities of the three domains: the environment (regulation authorities, competitors, providers,...), the airliners and the aircraft manufacturer (fig. 1.4).
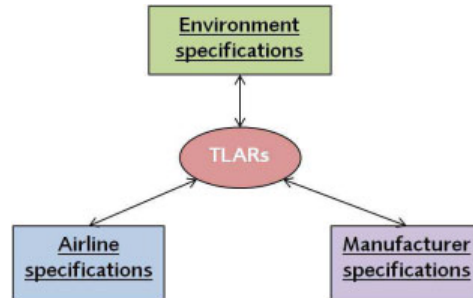
Figure 1.4: Entities which influence TLARs selection

Indeed, the requirements are not the same according to the entity considered. Sometimes, they can even be antagonistic. Then, it is necessary to come to a compromise in order to reach an aircraft configuration that satisfies more or less everyone (i.e. without letting somebody completely unsatisfied). Hereafter are some examples of requirements for each entity. Further details can be found in [Bad05].

Environment brings requirements on everything that is related to the environment in which the aircraft flies. It concerns:

- environmental constraints (noise, emissions...)

- Federal Aviation Regulations (FAR) or Joint Airworthiness Regulation (Joint Aviation Requirement) (JAR) regulations (in fact EASA, CS25 from 2008)(FAR in United States and EASA in European Union),

- competitor's aircraft,

- partnership (e.g. engine manufacturer)...

The most important for airline is to make profit. Then the requirements they bring are mainly related to passenger transportation aspect and on the operational aspect of the aircraft:

- range,

- capacity,

- noise,

- operational cost,

- cruise Mach,

- take-off field length...

For the manufacturer, the most important is to sell the maximum number of aircraft by minimising their cost production. Thus, they have requirements on:

- profitability,

- resources,

- product strategy...

There is another important point for the manufacturer that depends also on the TLARs values: it concerns the family concept. The manufacturer strategy is generally to design an aircraft family covering several seat capacities and not only a single aircraft. This practice is explained by the attempt to cover different market segments and to take advantage of technological evolutions. This evolution process is called "aircraft derivation". Indeed, it is less expensive and more rapid to "derive" an aircraft from an existing one than to start a complete new aircraft design. The capacity to downsize or extend an aircraft configuration depends on the original TLAR values. Indeed, with irrelevant TLAR values, it may be impossible to derive the related aircraft.
Figure 1.5 presents the family concept idea. With a family with three members, one could start with the basic aircraft configuration which is in the middle and then derive a stretch version and a shrunk version.
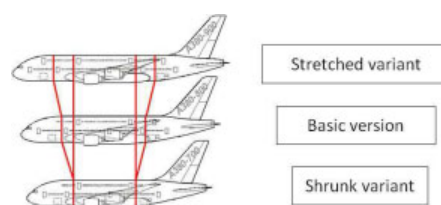
Figure 1.5: Family concept: basic, stretched and shrunk variants

The OAD role is to select those TLARs by taking into account the requirements coming from all the three entities: environment, airline and manufacturer. The set of requirements that we have finally retained in our study are:

- Range

- Total number of passengers

- Cruise Mach Number (Mn)

- Aircraft wing Span

- Take-Off Field Length (TOFL)

- Approach Speed (Vapp)

- Time To Climb (TTC)

- Initial Cruise Altitude (ICA)

- Flight ceilings

- Fuel Margin.

They are mostly related to performance aspects of the aircraft. Requirement on wing span is a geometrical limitation due to operational constraints aircraft have to satisfy. These constraints are defined by International Civil Aviation Organization (ICAO).

- TTC is related to the time the aircraft should take to reach cruise altitude.

- ICA refers to the capability to start cruise at sufficient altitude regarding to air traffic, meteorological conditions and best aircraft efficiency.

- Fuel margin corresponds to the difference between the total fuel necessary for the sizing mission and the Maximum Fuel Capacity (MFC) of the aircraft (see Payload-Range diagram for more explanation: figure 1.21).

- There are 4 types of flight ceilings: climb ceiling, cruise ceiling, buffeting ceiling and One-Engine-Inoperative (OEI) ceiling.

  - Climb ceiling corresponds to the highest altitude for which the aircraft is able to insure a given climb speed (or vertical speed).
  - The definition of cruise ceiling is almost the same. The difference is in the engine ratings: the engine rating is now configured for cruise instead of climb.
  - Buffeting refers to vibration of the aircraft caused by airflow separation. That may appear during cruise. Depending on the angle of attack, the airflow may separate which causes aerodynamic instability. Due to this instability, the structure of the aircraft shakes and the structural integrity of the aircraft is not ensured anymore. Buffeting is a dangerous state for the aircraft. Thus, it is important to identify the flight ceiling where this phenomenon appears and to add safety margins.
  - OEI ceiling refers to the highest altitude for which the aircraft is able to ensure a "PENTE" of at least 1,1% after an engine failure.

OAD role is also to estimate the cost associated to each TLARs. This amounts to asking question such as "what is the impact of requiring 4500NM range rather than 4000NM on aircraft performance and fuel consumption?". The answer to such question is looked for doing trade study.

1.1.2   Preliminary Aircraft Design Process

1.1.2.1   Multidisciplinary and simulation-based aspect

The preliminary aircraft design process has several characteristics: it is multidisciplinary and simulation-based. A process is defined as multidisciplinary when several disciplines are involved so that none of them could be individually determining. An aircraft is a system where many disciplines are highly contributing to achieve final objective which is to transport passengers

safely and economically. As an example, we have geometry, structure and weights, propulsion, aerodynamics, thermics, acoustics ... Each of these disciplines has its own characteristics, objectives and requirements. When designing an aircraft definition process, the purpose is to find the configuration which has the best performance as a global system. We do not look for the best of each discipline because it would not bring a global optimum. The optimum configuration is the one where a compromise is made between the different disciplines. Figure 1.6 shows the shape of the optimum aircraft configuration for each disciplines point of view.
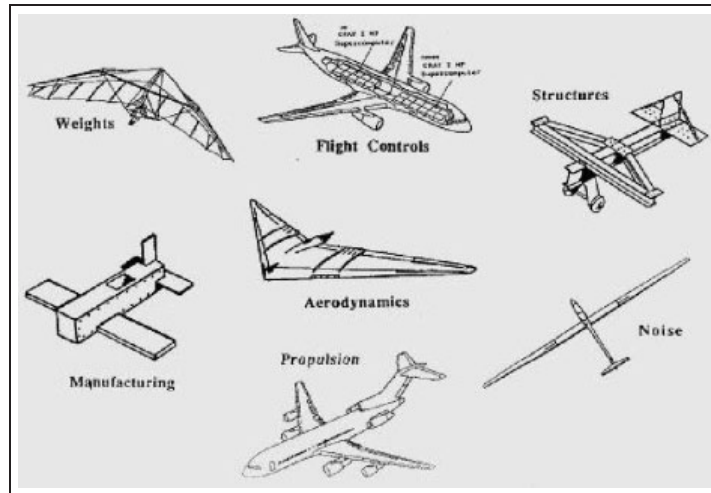
Figure 1.6: Idea of optimal configuration for each disciplines

The different aspects of those five disciplines are simulated/estimated thanks to models. That leads to the second characteristics of the aircraft design process: it is based on simulation. The aim in OAD is to find relevant models that allow predicting the physic behind each disciplines in short computation time so that a great number of evaluations could be performed (i.e. wide exploration of the design space). These models are presented in section 1.3.
The preliminary aircraft design process contains several internal sub-processes but we will focus on two of them: the Mass-Mission loop and the optimisation process. They are presented hereafter.

1.1.2.2 Aircraft Weights description

In the present study, we will refer to several characteristic weights of the aircraft: Maximum Take-Off Weight, Operator Empty Weight or again Landing Weight. They are used in the Mass-Mission loop and the Payload-Range diagram which will be presented later. Thus, to ease understanding, we present first the relations existing between all of these characteristic weights. In figure 1.7, we define:

- Manufacturer Weight Empty (MWE)

- Operator Empty Weight (OWE)

- Zero Fuel Weight (ZFW)

- Landing Weight (LW)

- Take-Off Weight (TOW)

- Ramp Weight (RW)

When ZFW, LW, TOW and RW are reaching their maximum value according to maximum load concideration, they are called:

- Maximum Zero Fuel Weight (MZFW)

- Maximum Take-Off Weight (MTOW)

- Maximum Landing Weight (MLW)

- Maximum Ramp Weight (MRW)

,
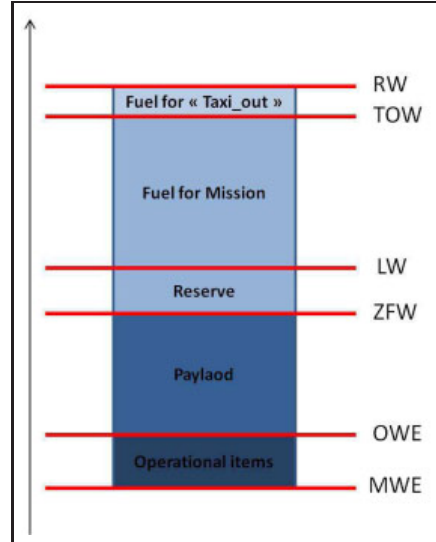
Figure 1.7: Design weights

1.1.2.3 Mass-Mission loop

The Mass-Mission loop is the process used to compute the $MTOW$ of an aircraft in accordance with the nominal payload and the nominal operational range. MTOW calculation encompasses two aspects: fuel consumption aspect and structural aspect. To carry out the mission the aircraft must take a necessary quantity of fuel on board, which impacts the $MTOW$. Moreover, the structure of the aircraft must be adapted to characteristics masses: MLW, $MTOW$, $MZFW$... Thus, a system needs to be solved in order to find the MTOW value.

Design Mission aspect: range constraint
An aircraft is expected to perform a mission. Specifically, it is designed to respect the Payload-Range diagram (see figure 1.21). A nominal range RA_{nom} is associated to a nominal payload $PLnom$. This nominal payload corresponds to the maximum number of passengers (and their luggage) in a typical configuration. To be able to perform the mission with the $MTOW$, it is necessary to have a certain quantity of fuel $Fuel_{nom}$. This quantity of fuel will have an impact on the $MTOW$ of the aircraft. Then, we have the following relation:

$$MTOW = ZFW_{nom} + Fuel_{Nom}.$$

The ZFW is the nominal Payload plus the OWE. OWE corresponds to the weight of the aircraft regarding its structure, propulsion and equipement but without any payload nor fuel. Then we have:

$$ZFW_{nom} = OWE + PL_{Nom}.$$

This means that:

$$MTOW = OWE + PL_{Nom} + Fuel_{Nom}.$$

So, we have:
$$OWE = MTOW - PL_{Nom} - Fuel_{Nom}.$$

Then, for a given mission, OWE is a function of $MTOW$:

$$OWE = f(MTOW).$$

Structural aspect: weight estimate

The structure weight, and then the OWE must be estimated to calculate the MTOW. The OWE depends on:

- MTOW

- MLW

- MZFW

- PL_{Max}: Maximum Payload

PL_{Max} is linked to the fuselage volume that a is geometrical limitation. The OWE can be written as a function of these four previous variables:

$$OWE = g(MTOW, MLW, MZFW, PL_{Max}).$$

As MLW is linked to MZFW (MLW = 1.06*MZFW), it comes that:

$$OWE = g(MTOW, MZFW, PL_{Max}).$$

And by definition,
$$MZFW = OWE + PL_{Max}.$$

Then
$$OWE = g(MTOW, MZFW).$$

For a given MTOW, we deduced that MZFW has to be adapted to get a OWE compatible with PL_{Max}. Then we obtain the following equation:

$$OWE = g(MTOW).$$

Figure 1.8 illustrates the idea behind the iterative resolution done (black arrow on the figure).

The blue line represents the relation between the range and $MTOW$ (mission aspect) and the yellow one represents the relation between payload and $MTOW$. The solution is situated at the intersection of the two linear operational constraints.

There exists a snowball effect behind the Mass-Mission loop: a small deviation on structure value can lead to a huge deviation on the final aircraft MTOW after looping. Figure 1.9 illustrates this phenomenon. Consider an initial deviation on OWE from the initial aircraft. We solve the Mass-mission loop starting from this deviation. The result is immediate: the looped aircraft is heavier than the initial one and the deviation is more important compared to the initial deviation introduced on OWE. For example, +1 tonne on initial OWE can lead to +2.4t on final MTOW and 1.4 on final OWE. Indeed, additional fuel is necessary to carry additional structural weight required.
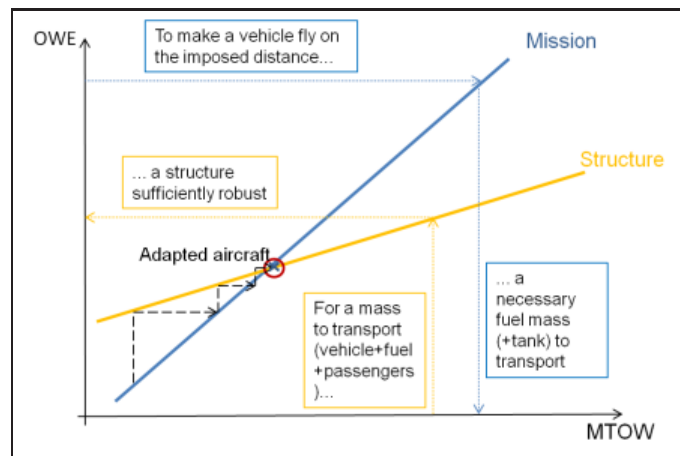
Figure 1.8: Mass-Mission loop
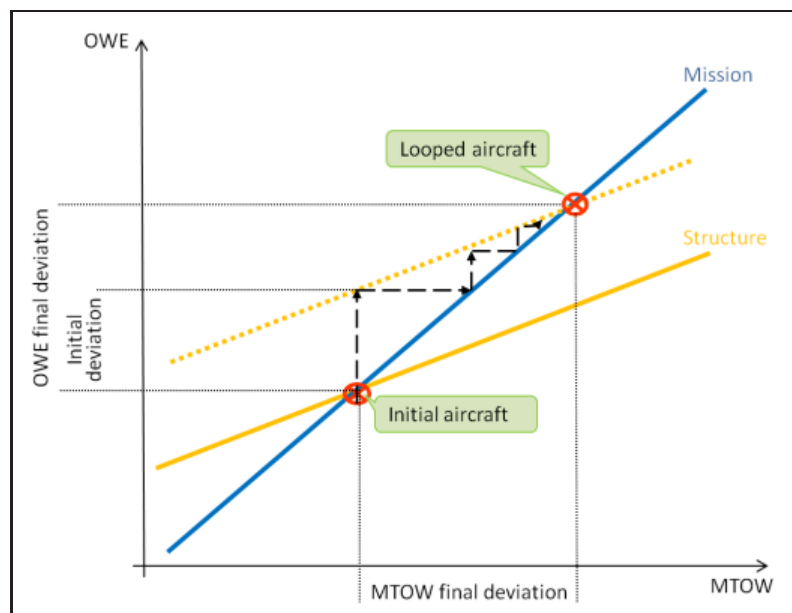
Figure 1.9: Mass-Mission loop: snowball effect

1.1.2.4 Optimisation under constraints

Among the small group of high level parameters that are managed at preliminary design level, three have probably the most important impact on the future product:

- Wing Area denoted *WingArea*,

- Engine Size driven by *SLSThrust*,

- Maximum Take-Off Weight (*MTOW*).

Wing Area and Engine Size are the most important components on which relies fly ability and performance of the aircraft. They are typically counting for half of the structural mass of the aircraft (see figure 1.10).
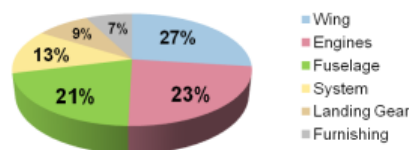
Figure 1.10: Typical split of the Manufacturer Weight Empty of an aircraft

A complete aircraft configuration can be described from the value of these two parameters. Moreover, modification on any TLARS has generally impact on both Wing Area and Engine Size. That places these parameters at the top of design parameters list.

MTOW is also an important parameter. It has a great impact on structural loads (as many maximum loads occur at maximum mass) and thus on the necessary amount of material. It is closely coupled to any form of cost associated to aircraft manufacturing and operation. For instance, it has a strong impact on all classical cost criteria (such as Direct Operating Cost (DOC) or Cash Operating Cost (COC)) as well as on all criteria related to energy assumption. It is an output from the preliminary design stage and an input for the detailed design phase. Regarding these reasons, *MTOW* appears as one of the most suited top level criteria to optimise. Generally, when designing an aircraft, we always look for optimising a target function. Nevertheless, with the fuel price increasing, other criteria such as the block fuel or climate impact parameters can be considered as top level criteria to optimise.

Assessing as soon as possible correct values for these parameters may reduce significantly the number of design loops (involving high fidelity tools) that will be necessary to converge towards the final design. The core process of the overall aircraft design has aim to define consistent values for these three parameters. Traditionally, this process is based on a deterministic optimisation. The aim is to look for the best couple *WingArea* and *SLSThrust* that minimize *MTOW* and satisfy operational requirements (take-off field length, approach speed, ceilings...), under a set of design assumptions (number of engines (Ne), wing aspect ratio (WingAR), By-Pass Ratio (BPR),...). By varying them, a set of possible aircraft configurations can be determined. Only configurations allowing to simultaneously reach the set of requirements and to minimize criterion are selected.

The structure of the optimisation process is in figure 1.11.
This process is clearly multidisciplinary. We distinguish at least five disciplines: geometry, propulsion, aerodynamics, weight and performance.
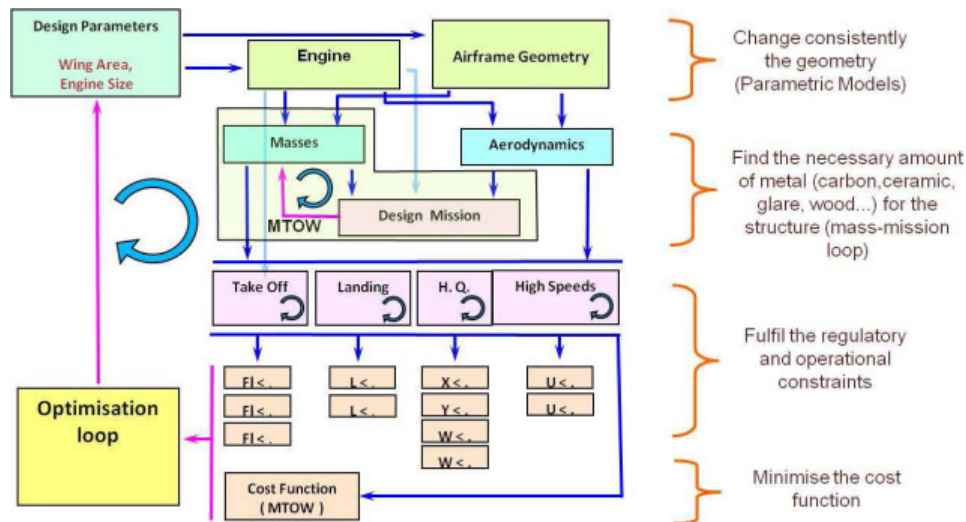
Figure 1.11: Overall Aircraft Design Process

The first step of the process is the definition of the aircraft geometry (see figure 1.12). From the design parameters (wing area and engine size being the most important ones), we drive the entire shape of the aircraft. This is done through parametric model which is in charge to change consistently the geometry (including the engine) of the aircraft. To do so, the parametric model implements feasible strategies to change the shape of the aircraft: for instance, if engines are attached to the wing, moving the wing sweep angle shall have the consequence to move also the engines. In addition to this type of geometrical constraints, it exists generally several ways to implement a given change: for instance, the area of the wing can be changed keeping constant the span or the aspect ratio. The choice of the geometrical constraints that are applied depends on the TLARs and influences the design space where the solutions are looked for.
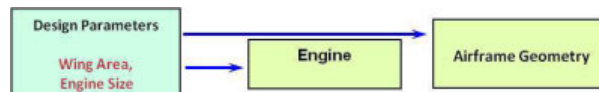
Figure 1.12: Step 1 - Definition of the aircraft geometry

Once the complete description of the aircraft geometry is available, we can move to step 2 which is about the estimation of aerodynamic forces and structural masses of the aircraft (see figure 1.13). To initiate structural mass estimation, we need to take assumptions of characteristic weights and especially the $MTOW$. In parallel, from masses and aerodynamic estimation, we can also simulate the mission of the aircraft. At the end of this simulation, a new value for the $MTOW$ is obtained according to the required payload. This leads to the Mass-Mission loop. The purpose of the Mass-Mission loop is to find the necessary amount of material for the aircraft structure. Further detail on the Mass-Mission loop is given in subsection 1.1.2.3.

At the end of step 2, we have the complete aircraft description regarding its geometry, engine, aerodynamics and weight. Then, in step 3 (figure 1.14), we are able to compute its operational performance and compare them to the requirements. We remind that the final objective of the design process is to find an aircraft configuration that fulfils all of these requirements at the same time by optimising a target function.

This final step leads to the top level aircraft design process which is the optimisation under constraints (figure 1.11).
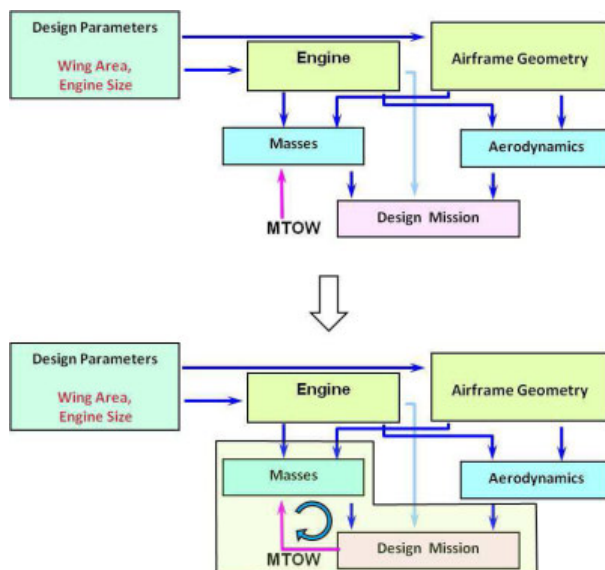
Figure 1.13: Step 2 - Estimation of aerodynamic forces and structural masses with Mass-Mission loop
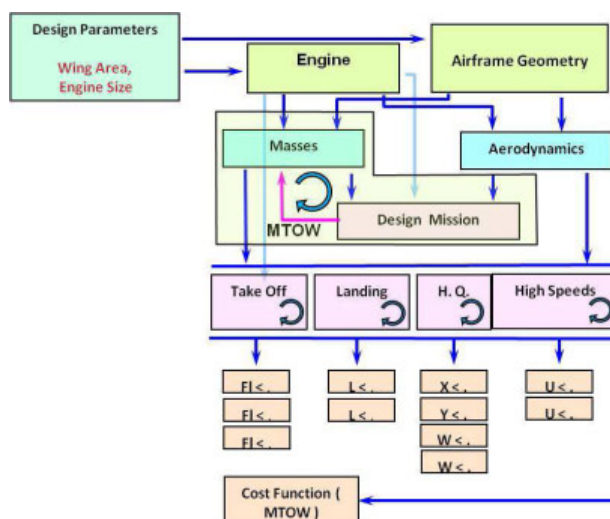
Figure 1.14: Step 3 - Operational performance computation and comparison to requirements

There is not always a solution to the optimisation problem. Indeed, it is not always possible to satisfy all requirements at the same time. Nevertheless, for ranges below 9000 NM and capacities not higher than 1000 seats, the specificities of the overall physical problem allow to find a feasible solution. This solution is represented graphically (see figure 1.15).
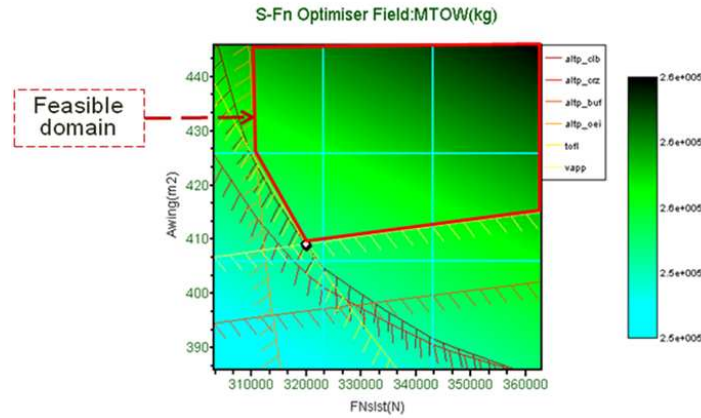
Figure 1.15: Overall Aircraft Design Process: graphical result

The design parameters engine size (driven by its max thrust) and wing area are respectively on X-axis and Y-axis. Each point of the surface stands for a given aircraft configuration. The colour field shows the evolution of the *MTOW* of these configuration: the darker, the heavier. According to this field, best aircraft has to be searched at the lower left corner (in the gradient direction). Each line of this graphic represents a constraint and divides the field in two areas: the hatched part is the area where the constraint is not satisfied. The feasible domain appears in the part of the field where all the constraints are satisfied. According to this diagram, it seems that it always exists a feasible area for high wing area and high engine power, in fact, wing area and engine power cannot be grown infinitely...

The solution for this classical optimisation is denoted:

$$WingArea_{deterministic} : \text{Wing Area value}$$

$$SLSThrust_{deterministic} : \text{Engine thrust value}$$

$$MTOW_{deterministic} : \text{Maximum Take-Off Weight value}$$

In this optimisation, the constraints are deterministic: the optimum point found is situated at the intersection of two active constraints. Any modification of the operational requirements would lead to get the optimum point be inaccurate and make obsolete the associated aircraft configuration (see figure 1.16). For example, if the estimation of the take-off field length reveals to be (a little bit) underestimated, the optimum solution found with the initial value of take-off field length will finally not satisfy its requirements. Such a solution is said to be not robust.

Robustness is not a new concept in aircraft design process. It had been developed in order to improve product quality and reliability. Robustification has aim to master the effect of the uncertainties along the design process in order to ensure requirements satisfaction. Classically, risks are covered by oversizing of some design parameters; the difference between oversized and nominal values of design parameters is called design margin. As oversizing generally leads to overcosting, the problem is of course to select the best set of margins that covers the widest uncertainty field at lower cost. From a formal point of view, robustness is acting as an independent new criterion (in addition to costing for instance) which makes the Robust Design a basic multicriteria problem. A particular Design can be defined as robust when its overall performances are able to resist to alterations of the assumptions taken for its elaboration. Then,
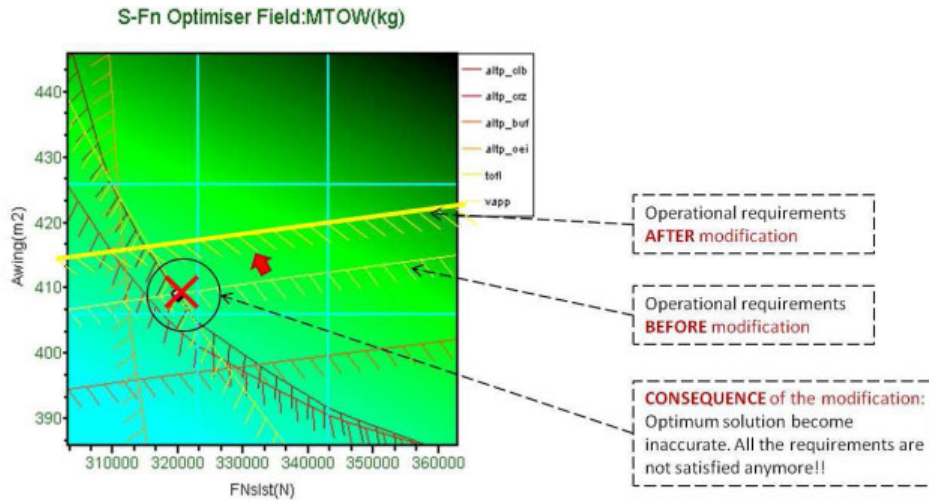
Figure 1.16: Consequence of modification of operational requirement value on deterministic solution

the problem was for engineers to define a methodology allowing to elaborate robust Design. A solution, integrating the risk concept, had been exhibited, and gave rise to a process: margin setting process. The process is presented in section 1.2.

1.2 Risk Management: Margin Setting Process

In many areas of aircraft design activity, technical risks are mitigated by introducing margins or safety factors on design parameter values [GNG11]. The purpose is to ensure the satisfaction of an aircraft design configuration according to an amount of uncertainty. These margins are often fixed values assessed according to experience gained by designers. Sometimes, they are computed according to simple processes as worst case or sensitivity analysis. But, even if data and know-how are available and if possible significant gain has been identified, these margins are rarely assessed according to uncertainty propagation techniques (which in the end could allow for the reduction of these margins, improving the overall performance of the product). This situation can be explained by the difficulty to change a proven and widely accepted way of doing when so important issues are involved such as meeting performance guarantees or passenger safety. A Margin Setting Process (MSP) based on uncertainty propagation would require defining acceptable probability of failure in fulfiling requirements. As no experience exists in using these techniques, we are in a 'chicken and egg' situation where things keep going as usual. Moreover, values of these parameters have a great impact since they are used to set up targets for the Center of Competence (CoC). These targets have to be ambitious enough to let room for challenge and innovation but realistic enough to maintain as much as possible overall consistency of the first design loops. The first definition of an aircraft configuration is driven by conflicting tendencies: if technically too comfortable, detailed design will most probably reveal an aircraft too expensive; if too ambitious, detailed design will reveal so many requirements not reached that a complete redesign may be necessary. In both cases, an important amount of work provides poor outcomes. The question we address is: does it exist a computational process that could ensure best chances to produce a relevant compromise for a new aircraft?

The margin setting process is an answer to that question. We present the rationale of this process in the following section. For that, we first have to introduce the Breguet-Leduc equation.

1.2.1 Breguet-Leduc Equation

The aim of a commercial aircraft is to carry an amount of passengers from one place to another one by air route. For that, we need three technological pillars:

- A structure to carry the passenger,

- A particular form called aerodynamic form to efficiently fly through the air,

- Engines to create propulsive forces called thrust.

Aerodynamics, thermodynamic of propulsion and structure-system weight are the three technological pillars that make possible air transport. These three ingredients are well integrated into the Breguet-Leduc equation, through combination of efficiency factors, to produce the range for a given amount of fuel. The Breguet-Leduc equation is given by:

$$Range = \frac{1}{g} \cdot \frac{L}{D} \cdot \frac{V}{SFC} \cdot \ln\left(1 + \frac{FuelBurn}{MWE + OIW + Pl + Reserve}\right) \tag{1.1}$$

where:

- V is the Ground speed (m/s),

- g is the Gravity acceleration (m/s^2),

- Pl is the Payload (kg),

- MWE is the Maximum Weight Empty (kg),

- OIW is the Operational Items Weight (kg),

- SFC is the Specific Fuel Consumption (kg/s/N),

- $\dfrac{L}{D}$ is the Lift over Drag ratio (no dimension) (further denoted L/D),

- $FuelBurn$ is the quantity of fuel burnt by the aircraft during a given mission (kg).

- $Reserve$ is additional quantity of fuel taken on board for holding and reroute (see detailed definition hereafter).

The MWE (see 1.7) corresponds to the sum of the weights of structural, system and engine components of the aircraft. That includes the weight of wing, fuselage, vertical tailplane (VTP), horizontal tailplane (HTP), landing gear, pylons, engines, systems and furnishings weight (toilets, floor covering, crew seats, overhead bins, emergency oxygen,...).

The *Operational Items Weight (OIW)* (see 1.7) is the sum of the weights of all equipment introduced for the operator of the aircraft such as passenger seats, catering, crew, unusable fuel, emergency equipment,...

The Specific Fuel Consumption (SFC) is the fuel flow rate required for a unit of thrust. For more details on this equation, see annex A.

The reserve is fixed by the regulation Jar 3% or Jar 5%. This additional quantity of fuel is constitute by 3% (or 5%, depending on the regulation) of the Block fuel plus fuel necessary for 30 minutes of holding starting from arrival airport plus fuel necessary for a reroute of 200 NM also starting from arrival airport.

In this equation, the aircraft velocity V and the payload Pl are fixed inputs because their value is generally the result of developments that are more strategic than technical.

This relation combines the influence of the three main drivers of air transport efficiency that are:

- L/D, which quantifies the aerodynamic efficiency of the airplane (the higher is L/D, the more efficient is the airplane);

- SFC, which quantifies the propulsion efficiency (the lower is SFC, the more efficient is the airplane);

- OWE, which is a measure of the structural efficiency (the lower is OWE, the more efficient is the airplane).

Consequently, from this equation, we understand that any problem which could appear in the design project of an aircraft, can be interpreted as a variation of L/D, a variation of SFC and a variation of MWE. It is then possible to simulate design perturbation by changing the value of these three variables. The rationale of the MSP is based on this fact.

1.2.2 Margin Setting Process principle

The MSP is a classical way in Future Project Office to design an aircraft which integrates margins. The process has two objectives: compute value for the three important parameters $WingArea$, $SLSThrust$ and $MTOW$ and determine realistic but ambitious targets for detailed design stage. The process is composed of three steps. In step 1, we compute oversized value of $WingArea$ and $SLSThrust$ by applying degradation factor on aerodynamics, thermodynamics and structure. This is done through the three efficient factors combined in Breguet-Leduc equation meaning through L/D, SFC and MWE. This step is called "take margins" step. In step 2, we find out a value for $MTOW$ by computing the Mass-Mission loop. It is called "the most probable case" step. The last step deals with computation of target for the Center of Competence (CoC). This last step is refered as the "Challenge step".

1.2.2.1 Step 1: Take margins

The aim of the first step of the MSP is to determine oversized values for $WingArea$ and $SLSThrust$. In a classical way, their values are determined thanks to the optimisation process previously presented. In this optimisation, we have a set of nominal values for the efficient factors L/D, SFC and MWE. The first step of the MSP is also based on this optimisation under constraints, but it integrates hypotheses of deterioration of the efficient factors values. This deterioration corresponds to an increase or a decrease of their value in order to make them worse. The different phases of this step are:

1. deterioration of efficiency factor values,

2. deterministic optimisation under constraint of $MTOW$.

Phase 1 is about the deterioration of the value of the efficiency factors L/D, SFC and MWE. Deteriorating their values mean decreasing the nominal value of the L/D, increasing the nominal value of the SFC and increasing the nominal value of the Empty weight of the aircraft MWE. Phase 2 is related to the optimisation process. The value of $WingArea$ and $SLSThrust$ are determined by computing the classical deterministic optimisation of $MTOW$ but with poor efficiency factors. Values obtained at this step are noted: $WingArea_{altered}$, $SLSThrust_{altered}$, $MTOW_{altered}$.
Figure 1.17 displays the principle of this first step.

The aircraft configuration obtained at the end of this step is oversized. If we compare this result with the one of the classical optimisation we have:

- $WingArea_{altered} > WingArea_{nominal}$
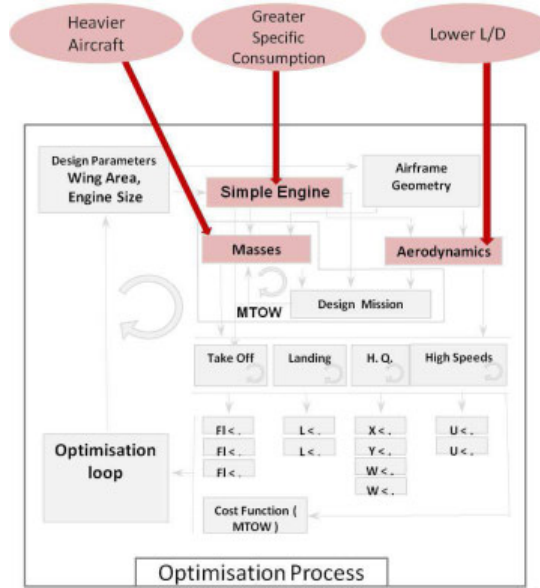
Figure 1.17: Step 1 of the MSP: risk cover

- $SLSThrust_{altered} > SLSThrust_{nominal}$

- $MTOW_{altered} > MTOW_{nominal}$

Indeed, if we want to fulfil the same set of requirements, the aircraft should have larger wings and more powerful engines. This is due to the degradation introduced. Moreover, for the same sizes ($WingArea$, $SLSThrust$), the aircraft is heavier in the altered case (with degradation) than in the classical one (without degradation)

Regarding to $MTOW_{altered}$, the value found is too conservative. Indeed, there is no real challenge in designing an aircraft with such $MTOW$ value. The objective is then to compute a less conservative value for this parameter meaning a smaller value. It is the aim of step 2.

1.2.2.2 Step 2: Most probable case

Once $WingArea$ and $SLSThrust$ values are fixed to $WingArea_{altered}$ and $SLSThrust_{altered}$, we have to determine a value for $MTOW$. This step is composed of the following phases:

1. setting of efficiency factors, $WingArea$ and $SLSThrust$ values,

2. determination of hypotheses for the $MTOW$ with the Mass - Mission loop.

Phase 1 is related to the setting of the different parameters. The efficiency factors L/D, SFC and MWE, are fixed to their nominal values: there is no deterioration considered. Regarding to $WingArea$ and $SLSThrust$, their values are set to $WingArea_{altered}$ and $SLSThrust_{altered}$ found in step 1. The risk is then taken into account by introducing these values into the calculation. In phase 2, we compute a value for $MTOW$ which is in line with oversized value of $WingArea$ and $SLSThrust$ but with no deterioration on efficiency factors. This is done by solving the Mass-Mission loop presented in section 1.1.2.3. The value obtained is noted $MTOW_{conservative}$. Figure 1.18 illustrates this second step.

By integrating the risk in the calculation, the value found for the $MTOW$ at the end of the Mass-Mission loop is specific: it is higher than $MTOW_{nominal}$ computed in the classical optimisation but lower than $MTOW_{altered}$ computed in step 2. In other words, we have:

$$MTOW_{nominal} < MTOW_{conservative} < MTOW_{altered}.$$
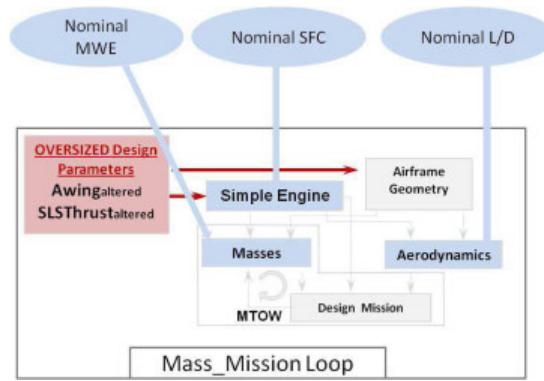
Figure 1.18: Step 2 of the MSP: most probable case

$MTOW_{conservative}$ value seems to be a balance between the better case and the worst case (according to a set of degradation).

We remind that OAD has 2 objectives: the first one is to give a global definition of an aircraft which fulfils a set of given requirements, and the second one is to compute target for the detailed design phase. The first objective is satisfied thanks to the first two steps of the MSP. The MSP assignes a value to the three main parameters $WingArea$, $SLSThrust$ and $MTOW$. Then, we have to define target for the Component Design. This is done through the last step of the MSP: challenge determination.

1.2.2.3 Step 3: Challenge determination

The aim of the last step of the Margin Setting Process is to determine targets for Component Design. There are two kinds of target: general targets and specific targets. General targets refer to those which concern the geometry and the mass of the aircraft. The specific targets refer to those which are transmitted separately to the different Centers. For instance, the SFC is a specific target for the engine manufacturer whereas the L/D is a specific target for the aerodynamic center. It is then important to find a way to give separately ambitious target to each discipline, which allows reaching a common goal but also let a room for expressing their creativity. For determining the targets, an hypothesis of improvement of the efficiency factor values is made. This last step of the MSP process is almost identical to the second step of the process. The main differences are located in the value of efficiency factors considered and the Mass-Mission loop condition. It is then composed of the following phases:

1. improvement of efficiency factor values and setting of $WingArea$ and $SLSThrust$ value,

2. computation of Mass-Mission loop with fixed hypotheses for $MTOW$.

Contrary to phase 2 of the risk cover step (step 1 of MSP), the value of the efficiency factors are improved: the L/D is increased, the SFC decreased and the MWE of the aircraft decreased. Moreover, as in step 2, value of $WingArea$ and $SLSThrust$ are fixed to $WingArea_{altered}$ and $SLSThrust_{altered}$ in order to always consider the risk cover in the calculation.

In step 3, computation of the targets is done. The Mass-Mission loop is run again but this fixed hypothesis about $MTOW$: the $MTOW$ value is fixed to the one found in step 2, meaning $MTOW_{conservative}$, and it does not change during the Mass-Mission loop (see fig 1.19).

This last step of the MSP leads to:

- risk cover with the choice of using $WingArea_{altered}$ and $SLSThrust_{altered}$,
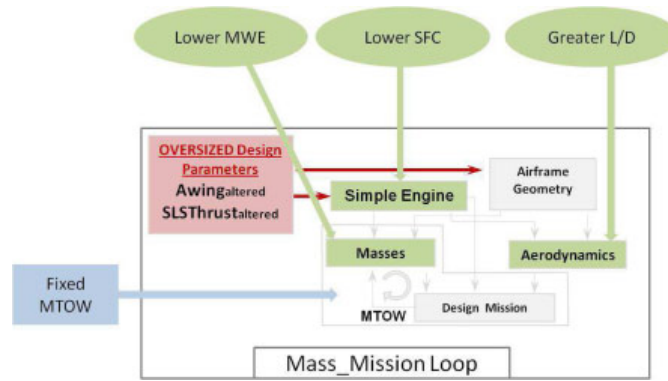
Figure 1.19: Step 3 of the MSP: challenge

- balance between risk and challenge with the value of $MTOW_{conservative}$,

- ambitious but realistic targets.

1.2.2.4 Graphical example

Hereafter is a graphical example which illustrates step 1 and step 2 of MSP.
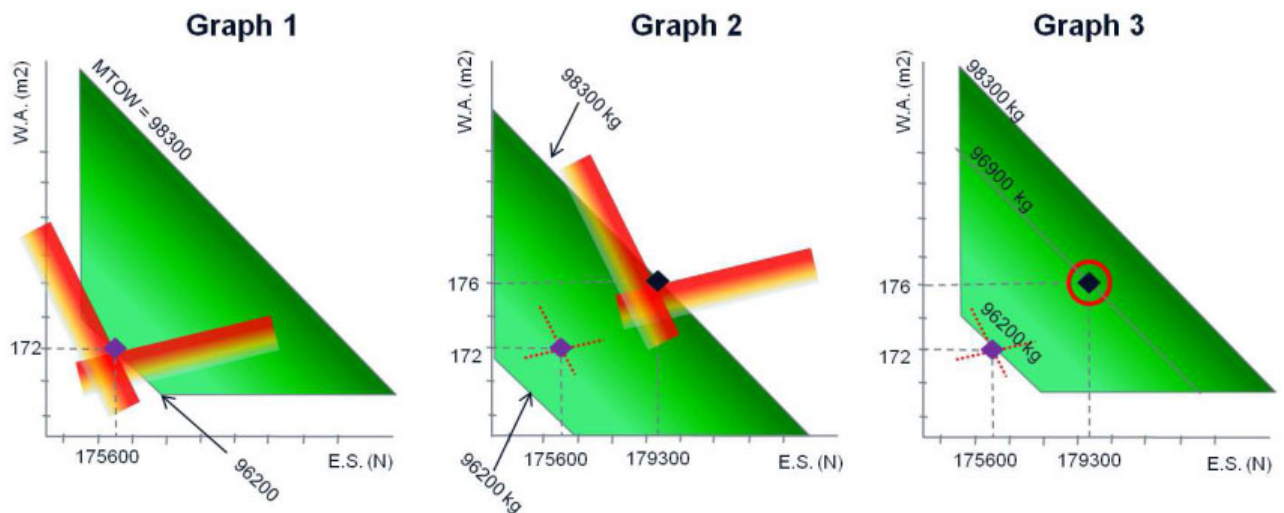
Figure 1.20: Graphical example of the MSP process

Graph 1 displays the result of the classical deterministic optimisation. In this case, the triplet of design variables (*WingArea*, *SLSThrust*, *MTOW*) of the optimal aircraft configuration is equal to (172 m^2, 175600 N, 96200 kg). It is represented by the purple diamond-shaped point.

Graph 2 displays the result of MSP's step 1. The deterministic solution is also visible on this graph. As described previoulsy, the aircraft configuration obtained at step 1 is bigger than the one found by the deterministic optimisation: the triplet is given by (176 m^2, 179 300 N, 98 300 kg). Indeed, because of the deterioration injected into the process, a larger wing and a more powerful engine are necessary to fulfil the same set of constraint than in the determistic case. Nevertheless, the value proposed for *MTOW* is too conservative. Moreover, for the same Wing Area and Engine Size, the aircraft is heavier (example: the value of *MTOW* obtained in this case for *WingArea* and *SLSThrust* respectively equal to 172 m^2 and 179600 N is higher than in graph 1).

Graph 3 displays the value proposed for *MTOW* by the step 3 of MSP. As explained afore-mentioned, this *MTOW* value is lower than the one in graph 2 and higher than the one in graph 1.

1.2.2.5 Conclusion

In this section, we have presented the principle of a Margin Setting Process dedicated to Aircraft Overall Design. The presented MSP makes use of the classical deterministic design process to build margins for overall design parameters and so make the final configuration more robust. Nevertheless, this method reveals two main drawbacks:

- It does not clearly explain which values of deterioration should be considered for the efficiency factors. Worst case is not applicable because it would result in a too much oversized and not competitive aircraft.

- Being based on a pure deterministic approach, the risk brought by constraints that are not active, but likely to be, is not taken into account.

Due to the level of granularity of selected models, the computation time of MSP process is relatively short. These models and their characteristics are presented in next section.

1.3 Models for Conceptual Aircraft design

In this study, we used an aircraft tool box called SiMCAD which is the acronym of Simple Models for Conceptual Aircraft Design. SiMCAD is dedicated to research activity in the domain of OAD. It has been designed to provide to academic people who are not necessarily familiar to aircraft design, the capability to tackle the problem from a systemic point of view. It is develop in Scilab [INR89].

It is a toolbox of models that allows to run simple, but realistic, pre-design processes involving all main physics. It offers the possibility to test multi-disciplinary, multi-level, multi-objective and robust optimisation strategies without having to manage huge amount of data and also huge computation time. By developing SiMCAD, the engineers have tried to capture the main effects that are contributing to give shape to an aircraft configuration. Moreover, the models have been designed so that they meet the following rules:

- The model toolbox must be able to compute all the parameters of the aircraft description knowing only the number of passengers, the nominal range and the cruise speed of the aircraft and a limited number of architectural choices, e.g. number of engines.

- All models must be as simple as possible and one run of a sequence, involving all the relations in the right order of call, should not exceed one second on a standard computer.

- The description must contain only top level parameters and the total number of parameters should not exceed a few hundreds (500 being an absolute limit).

- The description must be complete enough to give an overview of the configuration of the aircraft according to its geometry, weights, aerodynamics, low speed and high speed performances, route performances and cost.

The tool box has evolved during several years under the pressure of these rules and the various research activities that have taken benefit of it. For example, it is referred to as USMAC in [Bad05]. The level of granularity addressed by SiMCAD is low in order to respect the second rule. This type of model is also referred to Level Zero Models (LZ Models) by the engineers of FPO. The description of the aircraft is given by around 200 parameters. For each disciplines

(geometry, weights, aerodynamics, low speed and high speed performances, route performances and cost), only few aircraft parameters are selected to describe the aircraft. They are presented in the next section 1.3.1.

1.3.1   Aircraft description

1.3.1.1   Geometrical description

Five main aircraft components are considered in our model: cabin and fuselage, wing, horizontal stabilizer, vertical stabilizer, and nacelles. For all these components, only a limited number of very top level parameters have been selected. Most of these parameters are supposed to be measurable on pictures taken at a distance corresponding to several times the span of the aircraft. Some of them, like stabilizer volume factors or gross wetted areas are computable from the others. These parameters have been selected to give a global description of the aircraft that gets rid of all details. For instance, from the fuselage, only its average width, height, length and gross wetted area are taken. The complete geometry is represented by 36 parameters that are listed in Table A.1 with their name, usual unit and short description.

These parameters are considered as the minimum level of geometrical information that gives access to aerodynamic forces, component masses and propulsive efficiency estimations at a consistent level of accuracy for all the components.

1.3.1.2   Aerodynamic description

Only six parameters are used to define aircraft configurations from an aerodynamic point of view (table A.2).

The aerodynamic coefficients can be computed by aerodynamic models taking as input the previous geometrical description. One can ask how it is possible that so few aerodynamic parameters characterize an aircraft configuration? The following answer is clearly illustrating this fact: by looking for main part of aerodynamic contribution to aircraft design, we see that an aircraft must be able to take-off, to travel in cruise and to land. The three main parameters that are driving performances in such conditions are the maximum lift coefficients at take-off, the lift-to-drag ratio in cruise and the maximum lift coefficient at landing.

In this global approach, these three parameters have a slightly different meaning than in a pure physical approach. They are driving the aerodynamic performances of the aircraft from its overall geometry, provided that a competent design office is working to optimise the shape and robustness of the aircraft. The proposed aerodynamic model does not pretend to figure out the physical performances of a given shape. It gives a reachable level of performance corresponding to a given shape associated to current state of the art skills and technology. The numerical link between the shape and the aerodynamic coefficients is not only pure physics but also manufacturing know-how and technological assumptions. These assumptions are introduced in the models through statistical calibrations when enough data are available.

1.3.1.3   Mass description

The mass characterization is of similar level of granularity as geometry. It is done through the main component mass breakdown and a classical set of design weights (table A.3).

1.3.1.4   Engine performance

In this paragraph, one can find some information about engine performances in terms of available thrust in several characteristic flying conditions and SFC in cruise conditions. These values can be used to compare the engines of different aircraft configurations. Each typical rating of the

engine power management is characterized by one single set of flying conditions, except for take-off where two different sets of flying conditions are specified to be able to distinguish the effect of temperature (disa) (table A.4).

1.3.1.5 Operational performance

Considering that an aircraft should be able to take-off, climb, cruise and land, the constraints to be considered are related to these four flight phases. A particular attention is taken for constraints that can limit the top of climb point where cruise leg is starting. These constraints are called ceilings and they are of different types:

1. Minimum Vertical speed (climb and cruise ceilings, vertical speed (V_z) should not be lower than a given requirement).

2. Minimum Lift coefficient versus buffeting (buffeting ceiling, C_l should not be higher than $\frac{C_{l\ buffeting}}{1.3}$).

3. Minimum Climb path angle (OEI ceiling, path angle should not be lower than a given requirement).

4. Altitude of maximum Specific Air Range (SAR).

5. Time limited (TTC should not be higher than a given requirement).

In addition, the computation of the necessary time to reach the initial cruise altitude is introduced. It is not a ceiling but it can be an operational constraint. The Specific Air Range refers to the distance covered per unit of fuel mass burnt when considering the assumption of no wind. It is a measure of the efficiency of the aircraft. For information, the altitude of best SAR has been added and should be lower or equal to all top of climb altitude limitation. Indeed, when it is higher, it cannot be reached.

All these constrained outputs are computed at a given aircraft mass which is supposed to be a fraction of the MTOW (generally 0.98). In principle, the mass should result from the amount of fuel required to reach the targeted altitude but this complexity would be out of our scope. A constant factor on MTOW reveals to be a robust way to take into account the most important phenomenon which is: the heavier the aircraft takes-off, the heavier it reaches the cruise altitude. Relaxing this constraint (by managing a smoother climb) can be modeled by decreasing the factor on MTOW or simply decreasing the requested vertical speed. For most of these operational constraints, two computational points are proposed. Typically these points correspond to different flying conditions, for instance one condition could be sea level International Standard Atmosphere (ISA) and the other a hot and high condition for take-off, or two different vertical speed requirements corresponding to two different temperature deviations for climb and cruise ceilings (table A.5).

1.3.1.6 Mission performance

Five classical mission definitions are used to characterize transportation efficiency.

The first one, called Nominal Mission or Design Mission is used at design time to define the MTOW of the aircraft and correlatively the structural masses and the amount of fuel that must find place in the various tanks.

Then, three missions are used to define the so called payload/range diagram that usually looks like in Fig. 1.21.

1. Maximum Payload Mission (structural limitation): the aircraft is supposed to be loaded at its maximum payload capacity (passenger and also freight capability).

Figure 1.21: Payload range diagram.

2. *MTOW* (structural or performance limitation): payload and fuel added up to that TOW = MTOW.

3. *MFC*: all tanks are filled up and payload is reduced.

On branch 1 of Fig. 1.21, the fuel taken on board increases at constant maximum payload. On branch 2, fuel increase compensates payload decrease to keep the *TOW* constant and equal to its maximum (MTOW). That leads to increasing the range. Finally on branch 3, the payload is decreasing until the aircraft become the lighter one. The range reaches its maximum value after this. Most of the time, this limitation is required not to be active above a given payload. It may even not appear at all for some A/C.

The last mission, called Cost Mission is used to quantify an operational cost, that can be DOC or COC.

Each mission is characterized by its range and some important weights (Table A.6).

1.3.1.7   Operating cost

Two figures are most often used: the Direct Operating Cost (DOC) and the Cash Operating Cost (COC). The DOC is the COC plus Depreciation and Insurance. The DOC is used as far as specific economical context has to be taken into account or for mid and long term strategy analysis. For pure technical comparisons, COC is preferred. The selected formulas let two parameters to the user: the cost of human labor in $/h and the fuel price in $/USgal (Table A.7).

1.3.2   SiMCAD input data

SiMCAD tool box has 13 input parameters from which we are able to build a complete aircraft configuration. Some of them are coming from the TLARs (cf. section 1.1.1). They are:

- Total number of passengers (*Npax*)

- Nominal Range

- Cruise Mach Number (*Mn*)

- *ICA* (Initial Cruise Altitude)

- Center Wing Tank, Fuselage Tank, Tail Tank

- Containers and Pallets

- Number of engines (Ne)

- BPR (By-Pass Ratio)

- $WingAR$ (wing aspect ratio)

- Labor Cost

- Fuel Price

Total number of passengers, Nominal Range and Cruise Mach Number are most important TLARs and their values are decided by the aircraft manufacturer who wants to address a particular market area.

The Initial Cruise Altitude (which is also taken as the Reference Altitude) is an operational requirement.

The installation of a Center Wing Tank, a Fuselage Tank, a Tail Tank and the use of Containers and Pallets are configuration choices.

BPR refers to the ratio of amount of air blown outside the core (called secondary flow which is cold air) with respect to that moving through the core (called primary flow which is hot air) (see figure 1.22). A high BPR gives a lower exhaust speed. This reduces the SFC. Airbus' and
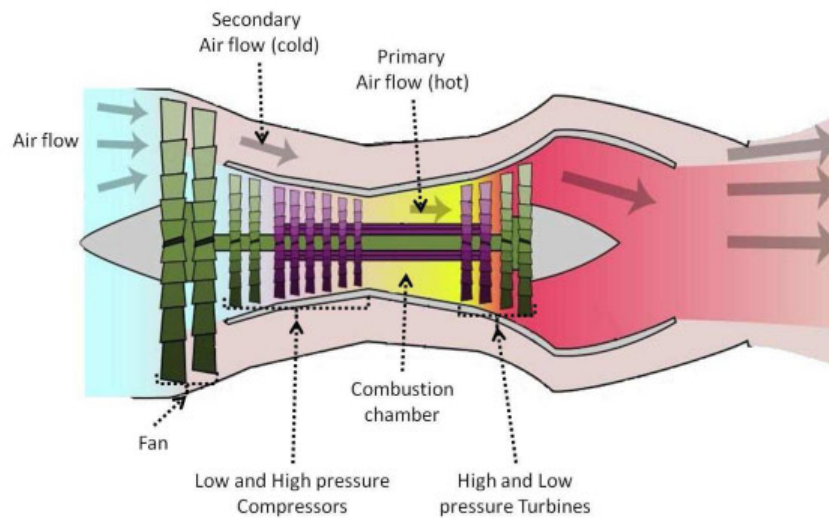
Figure 1.22: Engine By-Pass Ratio

Boeing's aircraft have a BPR value between 6 and 10. The BPR and the number of engines are mainly driven by availability of engines and/or technologies.

$WingAR$ is a measure of how long and slender is a wing: a high aspect ratio corresponds to long and narrow wings whereas a low one indicates short and stubby wings. It is the ratio between the square of the wing span with respect to the wing reference area. $WingAR$ of Airbus' and Boeing's aircraft is between 8 and 9.

1.3.3 SiMCAD outputs

The SiMCAD outputs are a list of operational performances of the aircraft which are:

- Fuel margin

- Take-Off Field Length (*TOFL*)

- Flight ceilings: Cruise, Climb, Buffetting and OEI

- Approach speed (*Vapp*)

- Time To Climb (TTC)

Most of them have been presented previously in section 1.1.1.

1.3.4   SiMCAD architecture

SiMCAD has a layered architecture (fig 1.23). The two lower layers (or inner layers) contain the Level Zero (LZ) Models which allow to estimate the aircraft component values within the following physics: geometry, weight, aerodynamics, propulsion, cost and performance. The higher layers (or outer layers) contain design processes which are used for running the Mass-Mission loop and the deterministic optimisation of *WingArea* and *SLSThrust* described previously.
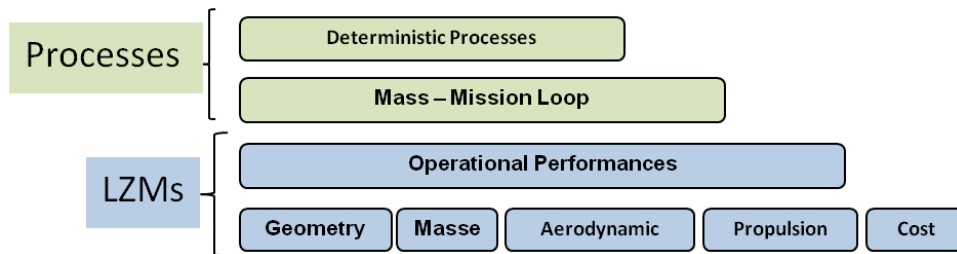
Figure 1.23: SiMCAD architecture

1.3.4.1   Level Zero Models

There are two levels of models in LZ Models. The first level is composed of basic models used to compute the parameters related to the geometry, the mass, the aerodynamics, the propulsion and the cost. This first level of models is then used and combined in the second level of models in order to compute the operational performances of the aircraft (previously listed) at given flying conditions.

    The LZ Models is composed of three types of function:

- Definition function,

- Regulation function,

- Models.

1.3.4.1.1   Definition function   Definition function refers to functions which give the expression of a physical parameter versus a set of other physical parameters. As an example, a function in SiMCAD gives the definition of the aircraft wing span versus its wing area and its wing aspect ratio. This model contains only one line.

Function Name: wing_span_
Input: *WingArea, WingAR* - **Ouput**: *WingSpan*
$WingSpan = (WingArea \cdot WingAR)^{\frac{1}{2}}$

1.3.4.1.2   Regulation function   Regulation function gives the value of a parameter fixed by regulation. For instance, there is a function which gives the value security coefficient kvs_TO (where vs is for $v$elocity $s$tall). It refers to the required margin to respect versus stalling velocity at Take-Off. This function also contains only one line.

Function Name: Kvs_Take_Off_	
Input: None - **Ouput**: kvs_TO	
Kvs_Take_Off_ = 1.13	

1.3.4.1.3   Models

1.3.4.1.3.1   Model building process   Models are the functions that most populate SiM-CAD toolbox. They are designed to capture the main part of involved physical links between the parameters to be estimated and the available information in the aircraft description. They have been designed by following the process hereafter:

1. Define a list of parameters that are supposed to capture the target level of model granularity.

2. Define the sub-list of parameters that is the most representative of top level aicraft requirements (TLARs) of the object. For instance: number of passengers, nominal range and Mach number.

3. Try to set up relations to compute as many parameters as possible directly from TLARs and cascade the relations to compute as many other parameters as possible. Try to have independent variables in the input list of relations.

4. Build the other relations by giving priority to expert judgment and statistical efficiency of the formula versus the available data without restriction concerning which variable depends on what. Try to have independent variables in the input list of relations.

5. When a sub-set of parameters are linked by too many relations, some static situations may occur. In that case, eliminate from the set the relations that have the lowest statistical efficiency. Since remaining as general as possible is a concern, keep preferentially the relations that are linking parameters according to the minimum set of assumptions. For example: prefer a relation linking fuselage mass to its geometry instead of a relation linking fuselage mass to $MZFW$. The last one has probably a validity domain which is narrower than the first one.

6. Possibly, add or remove some parameters from the initial list in order to get overall consistency of the set of relations.

7. In some cases, it is not possible to establish a pure sequence of relations to compute all the parameters because the calculation of the parameters is made by using a system. In this case, we use statistics to initialise the parameters of the system.

8. Proceed from step 1 to 7 until all the parameters of the description can be computed.

9. For all loops that have been created in the work-flow due to upstream relations, install the necessary iterative solving to compute them by minimizing the size of the system of equations to manage.

10. Finally, install the overall optimisation loop.

11. Evaluate the statistical efficiency of the whole process to predict available aircraft descriptions from the limited set of selected input parameters.

12. Eventually, adjust model constants in order to improve the global efficiency of the process versus all available aircraft descriptions.

1.3.4.1.3.2   Model input selection   The main problem here is to identify the relevant links. This approach has been illustrated for long in the area of so called Semi-Empirical Models. For this reason, some of the relations used to build the simplified models have taken their structure from various literatures as [VLC86] or simply have been selected according to their statistical efficiency. In each cases they all have the same very classical methodological basis. For example we consider here a variable to be estimated, Fuselage mass, denoted *FusMass*:

1. Look for the parameters that influence directly this variable. In the case of the *FusMass* variable, one can propose *FusWidth*, *FusHeight* and *FusLength*. If it appears that important information is missing in the data structure, the parameters have to be rethought in order to have a consistent set. From another side, it is very important not to add too many parameters by trying to "improve" the first order: this could bring instability and make fuzzy the granularity of the global model.

2. See if some parameters can be coupled into composite ones according to physical understanding of the phenomenon. This step is very important because it contributes with the same magnitude as step 1 to narrow the model uncertainty. In the case of fuselage weight, we can note that most part of the structure is concentrated on its surface and on its floors, both parts which areas are proportional to the multiplication of the equivalent diameter by the length of the fuselage. So it seems to be a good idea to build the term $FusDiameter = \sqrt{FusWidth \cdot FusHeight}$ (which is a sort of virtual diameter of the fuselage that takes account of both width and height of the fuselage) and multiply it by *FusLength*. The result behaves as a built surface that not only takes account of the skin but also of the floor(s).

3. Introduce tuning parameters T within the currently built formula in order to match at best to a set of measurements that is relevant for the purpose. Here also, a compromise is necessary: too few points would over determine the parameters and produce a non physical behaviour of the model but it is sometimes difficult to find many relevant measurements. Too many points may fuzzy the validity domain of the resulting model. Least squares method allows to optimise parameter values. This step is important to be able to customize the model in a particular study context (for instance Short Range or Long Range, Turboprop or Turbofan, ...).

4. If necessary, compose the function through a polynomial function (with possible fractional power) in order to improve data matching. This last step is sometimes useful to capture some industrial hidden properties related to the overall size of the component. For instance the mass of a main aircraft component is never fully linear versus its size due to some snowball effect.

The result in the example would be for instance:

$$FusMass = T_1 \cdot FusArea^2 + T_2 \cdot FusArea + T_3 \qquad (1.2)$$

where:

- T = [0.008, 11.8, 1500],

- and $FusArea$ can be computed as $FusArea = \pi \cdot FusLength \cdot FusDiameter$.

Fig. 1.24 shows a classical way to represent the accuracy associated to a model. Coordinate of each point is built by taking along the horizontal axis the value of the variable coming from the set of experiments and along the vertical axis the value coming out of the model. Of course, all corresponding inputs of the model have to be available consistently with the measurements.

The diagonal is the line of equation $Y = X$. As a result, a point below the diagonal means that the model underestimates the value, while a point above the diagonal means that the model overestimates the value.
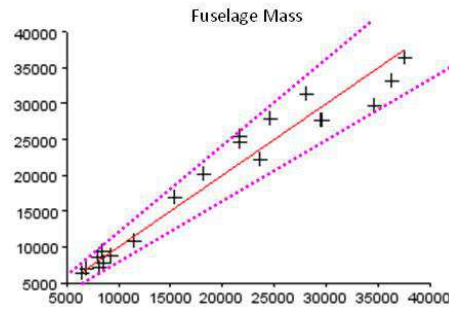
Figure 1.24: Model value versus Database value for Fuselage mass parameter

The example in Fig. 1.24 shows a typical behaviour of such models: absolute spreading increases with the order of magnitude of the known predicted data. It is to be noted that this behaviour is the most important justification to the classical operation consisting in tuning a formula with a multiplicative factor in order to capture a reference point. Models with such behavior are called *extensive models*. Conversely, models which are not extensive are *intensive* and have a behavior such as in figure 1.25.
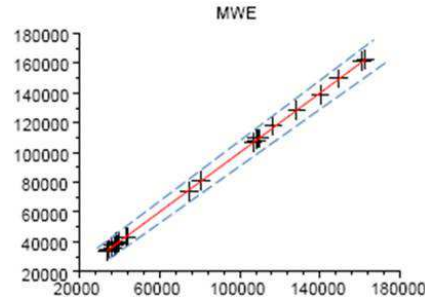
Figure 1.25: Model value versus Database value for MWE model

1.3.4.1.3.3 Examples of models Hereafter are some examples of models contained in SiMCAD.

The first example is a mass model which gives an estimation of the engine mass. The input data of this model are *SLSThrust* and the number of engines. The expression of the model is as follows:

Function Name: engine_mass_
Input: SLSThrust, Ne - **Ouput**: EngineMass
$T = 0.00247$
$EngineMass = T \cdot SLSThrust \cdot Ne$

A second example with a geometry model which computes the fuselage width. The fuselage width is estimated versus the number of passenger abreast and the number of aisles. The model expression is:

Function Name:	fuselage_width_
Input: NpaxAbreast, Naisle - **Ouput**: FusWidth	
$T = [0.38, 1.05, 0.55]$	
$FusWidth = T_1 \cdot NpaxAbreast + T_2 \cdot Naisle + T_3$	

Here is a last example with a performance model. The following model computes the value of the approach speed of the aircraft. It uses geometry and aerodynamics data in its calculation through *WingArea* and *CzMaxLD*.

Function Name:	approach_speed_
Input: mass, disa, Zp, kvsLD - **Ouput**: Vapp	
$global MODEL\_aircraft$	
$WingArea = MODEL\_aircraft.geom.wing(['WingArea'])$	
$CzMaxLD = MODEL\_aircraft.geom.wing(['CzMaxLD'])$	
$g = 9.80665$	
$T = 0.98$	
$[Pamb; Tamb; Rho] = non\_stand\_atmos\_(Disa; Zp)$	
$Vapp = 0.98 \cdot \sqrt{\frac{2 \cdot mass \cdot g}{Rho \cdot WingArea \cdot (CzMaxLD/KvsLD^2)}}$	

1.3.4.1.4 Dependency graph It is possible to group the different relations of SiMCAD into disciplinary modules. Figure 1.26 displays the dependency graph of the different modules. Hollow dots on the geometry line mean that the dependency is due to the *WingArea* only. It is to be noticed that the simplicity of the relations is responsible of the absence of some expected dependencies as the possible influence of the *L/D* upon the *TOFL* for instance. Dashed upward arrows are showing the reverse links that are the consequences of an optimisation process under constraints that can be played using SiMCAD. The following chapter presents this process and some others. Similar block diagrams can be made by exploring the content of each blue boxes.

All LZ Models can be put together in order to run classical design processes. In the next subsection, we give some details on them.

1.3.4.2 Design Processes

Using LZ Models, we are able to compute the four design processes that compose the higher layer of SiMCAD architecture, that is to say:

- Mass-Mission loop (cf 1.1.2.3),

- Deterministic optimisation (cf. section 1.1.2.4),

- Statistical design,

- Extended design.

All of these processes are very classical in the domain of aircraft design. They all have the objective to define the description of an aircraft that should fulfil a set of requirements.

1.4 Global context

The thesis subject is the continuation of one internal project and one external project which are respectively CRESCENDO [Pro09] and Integrated Design For Complex System (ID4CS) [ANR09].
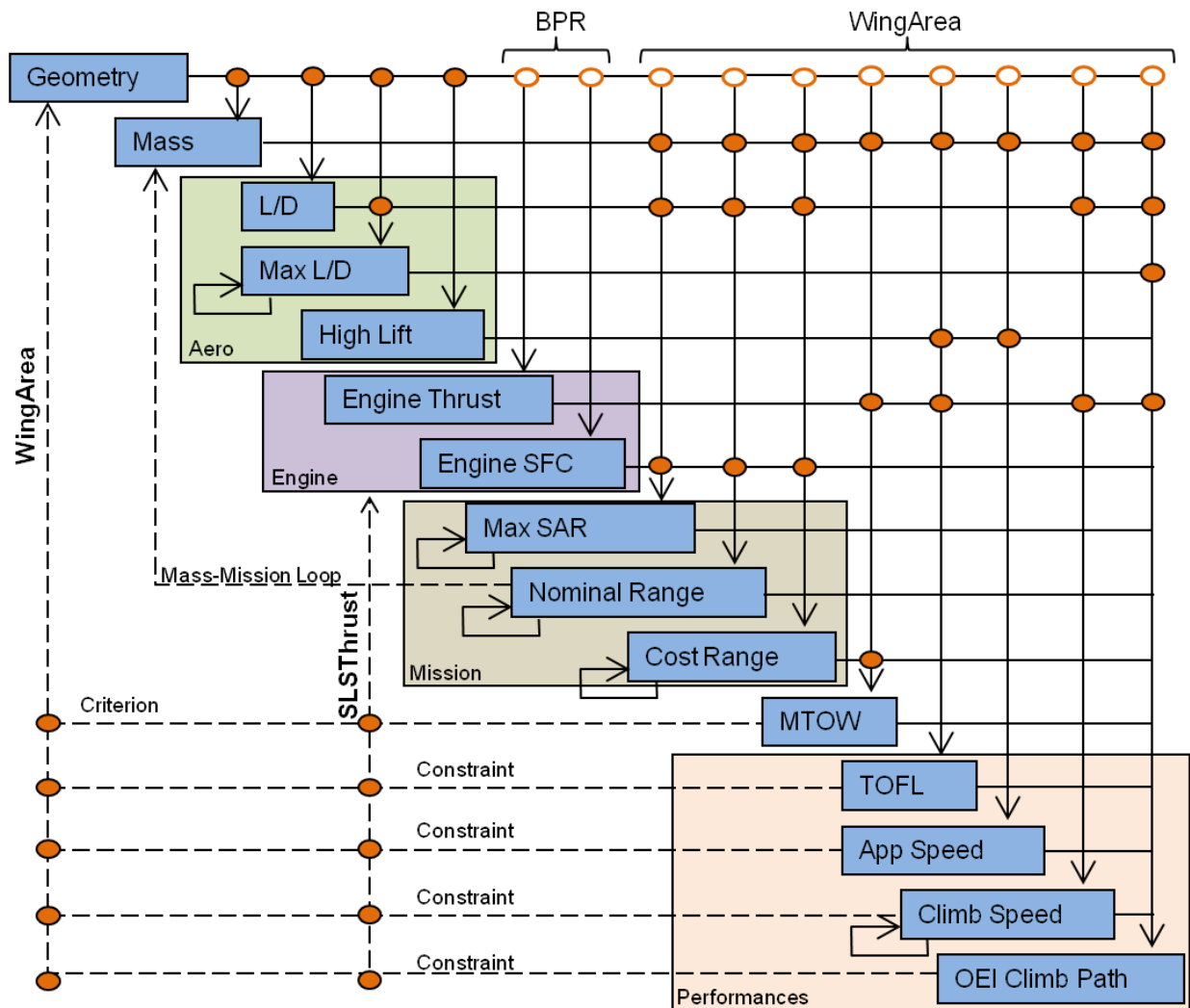
Figure 1.26: Dependency graph of SiMCAD models

CRESCENDO, for Collaborative and Robust Engineering using Simulation Capability Enabling Next Design Optimisation, is a 3-years project led by Airbus which brings together 13 European countries. The objective of the consortium is to bring methods for changing the way that Modelling and Simulation activities are carried out, in order to develop new aeronautical products in a more efficient manner in terms of cost and time. CRESCENDO will develop the foundations for the Behavioural Digital Aircraft (BDA) which will become the new backbone for the simulation (just as the Digital Mock-up (DMU) is today for the Product Life-cycle management). The main components of the BDA are: the model store, the simulation factory, the quality laboratory and the enterprise collaboration capabilities. This will lead to:

- a reduction of development lifecycle duration and cost,

- a reduction in rework,

- and a reduction in the cost of physical tests.

The CRESCENDO project will impact the three specific targets of the aeronautical industry's: Strategic Research Agenda. Note that CRESCENDO is the continuation of the project VIVACE. Information on the CRESCENDO project can be found in: [Pro09].

ID4CS, for Integrative Design for Complex Systems, is an ANR (French National Research Agency) project. The objective of the consortium is to propose a development and simulation environment for designing complex systems such as aircraft, by encompassing the complexity of the project which may come from several aspects:

- multi-disciplinary,

- multi-objective,

- multi-level,

- multi-attribute,

- multi-search-method,

- modelling uncertainty.

The idea is to use adaptive multi-agent technology in order to move toward a self-adaptative distributed and open architecture, enabling the integration of the previous multi-dimensional aspects. The final system will be able to communicate with several level of granularity. Information on the project can be found in: [ANR09]. The project is applied to the practical cases of preliminary aircraft design and aircraft engine design.

1.5   Industrial stake: need and motivation

As aforementioned, designing an aircraft is a long process which generally takes more than 9 years from the idea generation of the new aircraft to its first delivery. The phases which are necessary to move from a simple list of requirements to the fly-able aircraft configuration are numerous. Moreover, the process is not linear: number of iteration has to be made in order to access to the final product. Some of these iterations are unavoidable: they correspond to exchanges between the different poles of competence (Mass, performance,...) and between the actors of different phases (detailed phase to preliminary phase for example). In contrast, iterations that are necessary to correct wrong initial assumptions are avoidable. This category of iterations is the result of uncertainties which are present at some maturity gates and which

consequences are only revealed downstream in the design development process. Thus, the management of these uncertainties since the first stage of the design process might help in avoiding these unnecessary iterations. Depending on the actual technical difficulties of a given project, removing unnecessary iterations may lead to significant decrease of development cost.

Nowadays, engineers look for new strategies and new tools which can help them to improve the aircraft design process in three ways:

- by achieving a better initial effectiveness of the product.

- by focussing on the very final objectives of the customers, internal ones such as any design office department or final ones such as the airline companies and their passengers.

- by reducing the design cycle. This could be achieved by making the selected aircraft configuration more insensitive to variations which occur all along the design development process. That will allow to avoid trivial iterations between the different design phases.

As described before, the final aircraft design, appointed the nominal design, is classically the solution of a deterministic process. It is the best solution engineers are convinced they are able to produce according to the available technologies. Moreover, this final aircraft configuration integrates margins (or safety factors), that are taken to mitigate the risk and cover possible deteriorations of the product performances revealed by tests for instance. These margins are generally fixed values assessed according to experience compiled by the designers on previous aircraft project. There are *a priori* margins and generally are not the results of an explicit quantitative evaluation process. This way of doing is characteristic of a conservative approach which reveals unpleasant aspects: it promotes the accumulation of *a priori* margins which may lead to big oversizing of the product and jeopardise its economical efficiency as well as introduction of new technologies. Indeed, each expert at each stage of the design takes margins to avoid risks noted in the past or forecasted due to new technologies. At the end, the final product is probably in accordance with the requirements, but less attractive for customers.

The use of an explicit method for managing the uncertainty can help to fight against systematic conservative attitude and improve the performances and the attractivity of the final product. The only set of margins that are exported to the final customers (the airliners) are those who are attached to the guaranties of performances that are given by the aircraft manufacturer to trigger the first contracts. If we succeed in quantifying the whole chain of uncertainties along the project life, we can imagine being able to retro-propagate these uncertainties on guaranties at each stage of the design process which would be the most clever way to handle them. Moreover, we can imagine that some margins are taken on some detailed design parameters in early design stages on the basis of quantitative uncertainty propagations and released afterwards when it becomes clear that they are useless to achieve the final requested performances. Presently, we are far from being able to quantitatively manage uncertainty all along the project life. The presented work is a first step to introduce some techniques of Uncertainty Quantification and Management (UQ&M) that are able to make the link between detailed design levels, where UQ&M already exist or is beeing developed and Global Aircraft Level where final performance constraints criteria are expressed.

1.6 Global objective of the thesis

The global objectives of the thesis are to introduce and to implement new mathematical tools in order to initiate robust aircraft design practices. For that purpose, we will first define the mathematical methodologies involved and realise their state of the art in order to select the ones

that are the most adapted to aircraft design process.

The second task is to introduce the current approach used in FPO in order to manage the risk. The aim is to analyse the advantages and the weaknesses of these methods. Thanks to this analysis, we will be able to select another method to achieve our goal.

Another part of the objective, and one of the main, is to introduce the notion of uncertainty. The aim is to propose a new process based on uncertainty propagation which allows to improve the way the margins are taken on aircraft design while ensuring the satisfaction of the operational constraints. The first goal will be to carefully identify the uncertainty involved at FPO stage and then to propose effective methodology to propagate it through the simulation code.

The last part of the objective is to put in place the robust design. We have to find the relevant problem formulation depending on the FPO objective. One of the challenges is to find friendly way to explain to aircraft engineers the physical meaning of the results obtained with the new methodology and to make the link with the method currently used.

Chapter 2

State-of-the-art

Contents

2.1   Introduction

As described before, preliminary aircraft design consists in the global definition and sizing of a future aircraft configuration in order to fulfil specific requirements. It is a complex deterministic, mono-objective, constrained, multidisciplinary and simulated-based process. The engineer role is to determine the optimal aircraft configuration which permits to optimise a given criterion (for instance, to minimise the $MTOW$), while fulfilling operational (and cost) requirements. To find this optimal solution, he uses a mathematical method called optimisation.

Once the engineer has identified an optimal aircraft configuration, he has to realise it. This realisation is not a linear process: unforeseen turns of events, such as change of strategy or involvement of new supplier, generally occur at this stage. Moreover, errors are committed during the process. These errors are due to imprecision of measurement and calculation tools, to physical aspects out of control such as the whether (...). This kind of unforeseen event can degrade significantly the performance of the final product. The classical optimisation problem which gives the optimal configuration does not take into account these unforeseen turns of event: it supposes a complete absence of uncertainty whereas reality is more complex. Methods have been developed to take uncertainty into account during the optimisation process. The product quality obtained by optimising with uncertainty is improved and its sensitivity to uncertainty is reduced. These kind of methods are called Robust Optimisation methods.

This chapter is focused on the introduction of different methodologies existing for the resolution of an optimisation problem under uncertainty. On one hand, we will be able to fully realise the amount of work which has been done on the subject and on the other hand, we will be able to identify the methodology which can be applied in the case of aircraft preliminary design. Thus, in a first part, we briefly present an optimisation problem in its generality and introduce the notations. In section 2.3, we present multi-objective optimisation problem and some methods for solving it. In section 2.4, we introduce the notations and definitions related to statistical and probabilistical concepts necessary for the understanding. Section 2.5 is about the presentation of robust optimisation. We first introduce the notion of uncertainty and its measure. The different sources of uncertainty will be listed and methodology for integrating it in the optimisation process will be studied. Then, we present the existing methods for the optimisation of the robustness measures presented.

2.2   Optimisation

A general single-objective (or mono-objective) optimisation problem can be formulated as in equation (2.1):

$$\min_{x \in \Omega} f(x) \tag{2.1}$$

where $\Omega \subset \mathbb{R}^n$ is the design space, $x = (x_1, \ldots, x_n)$ is the vector of design variables (also called decision variables) and $f \colon \mathbb{R}^n \to \mathbb{R}$ is the objective function to be minimised (also called cost function, criteria, payoff function or value function). Depending on the problem considered, we will seek for the minimisation or the maximisation of the objective function f. Note that these two formulations are linked because maximising $f(x)$ comes down to minimising $-f(x)$. In this study, all the formulation will be written in the sense of the minimisation.

The aim of the optimisation is to determine the solution x^* for which the value $f(x^*)$ is minimal. We distinguish two types of solution: local and global ones. A local solution is the best solution within a neighbourhood set of feasible solutions whereas a global solution is the best one within the set of all feasible solutions. Figure 2.1 is an illustration of the optimisation concept in the case of the minimisation of an objective function f, where the notions of local and global solution are represented.
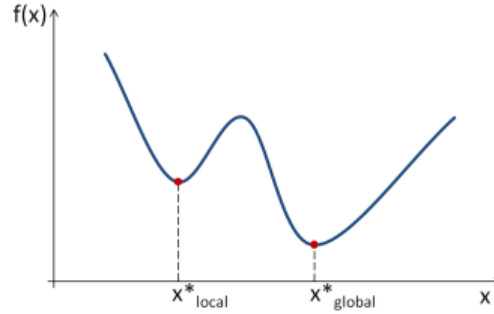
Figure 2.1: Illustration of the optimisation concept

The optimisation problem can be constrained by equality or/and inequality constraints. The addition of these constraints has the effect to reduce the design space. They are commonly denoted h for equality constraint and g for inequality constraint. This type of problem is in the class of *constrained optimisation*. Then, equation (2.1) becomes:

$$
\begin{aligned}
\min_{x \in \Omega} \quad & f(x) \\
\text{s.t.} \quad & g_j(x) \leq 0, \quad j = 1, \ldots, m, \\
& h_k(x) = 0, \quad k = 1, \ldots, p,
\end{aligned}
\tag{2.2}
$$

where:

- $\Omega \subset \mathbb{R}^n$ is the design space,

- $x = (x_1, \ldots, x_n)$ is the vector of design variables,

- $f \colon \mathbb{R}^n \to \mathbb{R}$ is the objective function,

- m is the number of inequality constraints and p the number of equality constraints,

- $g(x) = (g_1(x), \ldots, g_m(x))$ is the vector of inequality constraints,

- $h(x) = (h_1(x), \ldots, h_p(x))$ is the vector of equality constraints.

For more detailed information, the reader can refer to [Yan08] and [Bie06, NW06].

There exists a large list of methods to solve optimisation problems. The choice of the most efficient method to solve a given problem is guided by its nature! characteristics of the objective function, characteristics of the constraints (...). The choice is also guided by information available at this time such as derivative information. There are two families of methods: Derivative-based methods and Derivative-free methods. Derivative-based methods contain algorithms which use the derivative information to solve the problem. Obviously, there are applicable only when the derivative information is available or can be approximated (e.g. finite differences). Conversely, algorithms contained in the family of derivative-free methods do not require the derivative information. Figure 2.2 presents the tree of both derivative-based and derivative-free families.

In figure 2.2, an example of 1.5-order methods is quasi-Newton method [Cor91]. In this study, we will not go further in the presentation of these methods.
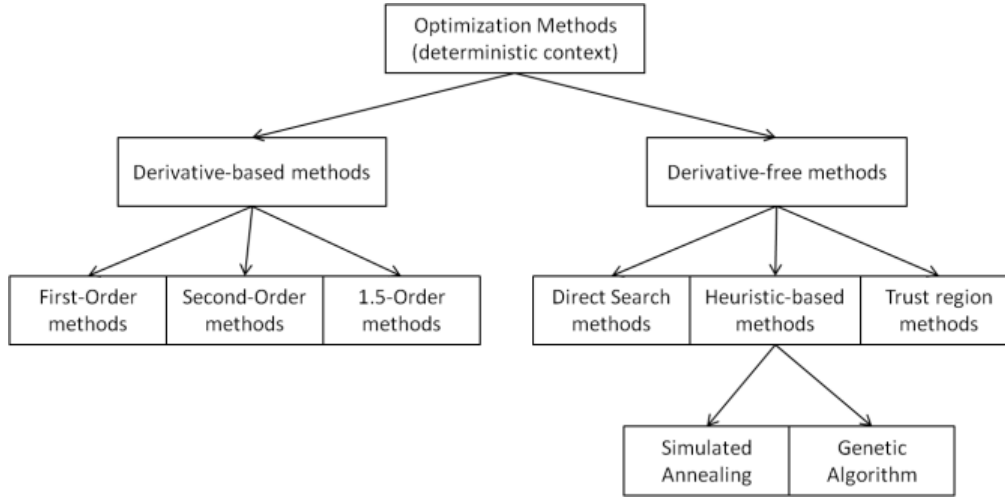
Figure 2.2: Methods for deterministic optimisation

2.3 Multi-Objective optimisation

Optimisation problems engineers have to deal with in the real-life have generally more than one objective to be optimised. This kind of problem with several objectives is refered to as *multi-objective optimisation* or *multicriteria optimisation*. Many examples of multi-objective optimisation can be found in the literature. For instance, in the field of finance, a common problem is to choose a portfolio for which the risk is minimised and the expected value maximised. Another example is in the paper industry, where the design of a paper mill requires to decrease the amount of capital invested while increasing the quality of the paper. Both of these previous examples are bi-objective optimisation problems.

A general multi-objective optimisation is formulated as follows:

$$
\begin{aligned}
&\min_{x\in\Omega} && F(x) = (f_1(x),\ldots,f_l(x)) \\
&\text{s.t.} && g_j(x) \leq 0, \; j = 1,\ldots,m, \\
& && h_k(x) = 0, \; k = 1,\ldots,p,
\end{aligned}
\tag{2.3}
$$

where l is the number of objective functions, m the number of inequality constraints and p the number of equality constraints. The feasible design space $\mathbf{X}$ (also called decision space or constraint set) is defined as the set $\{x \in \Omega : g(x) \leq 0 \text{ and } h(x) = 0\}$. The feasible criterion space $\mathbf{Z}$ (also called feasible cost space or feasible objective space) is defined as the set $\{F(x) : x \in \mathbf{X}\}$. It is the image of the feasible design space.

In the case of multi-objective optimisation, the solution is generally not unique: it is unlikely to find a solution that optimise each objectives at the same time. In fact, the solution is more a set of solutions. The aim, in multi-objective problem, is to find a solution that gives a satisfactory design and this is done thanks to a trade-off between the different objectives. Thus, it is necessary to extend the notion of optimum for the case of multi-objective optimisation. This leads to the introduction of Pareto-Optimality and dominance notions.

2.3.1 Dominance relation and Pareto-optimality

2.3.1.1 Dominance relation

The notion of dominance defines a relation between two feasible points x_1 and x_2 in order to determine if one of them is better than the other one or if they are equivalent. It is said that

x_1 dominates x_2 (denotes $x_1 \preceq x_2$ [Ehr05]) if the following conditions are true:

1. the solution x_1 is not worse than x_2 in all objectives,

2. and the solution x_1 is strictly better than x_2 in at least one objective.

That leads to:
$$x_1 \preceq x_2 \Leftrightarrow \begin{cases} \forall i \in 1, 2, \ldots, l, & f_i(x_1) \le f_i(x_2), \\ \exists j \in 1, 2, \ldots, l, & f_j(x_1) < f_j(x_2). \end{cases}$$

2.3.1.2 Pareto Optimality

The notion of dominance is used to defined the Pareto Optimality. A point $x^* \in \Omega$ is Pareto optimal if and only if it is a feasible solution and if there is no other feasible point $x \in \Omega$ such that $x \preceq x^*$ (i.e. there is no other point such that $F(x) \le F(x^*)$ and $f_i(x) < f_i(x^*)$ for at least one $1 \le i \le l$). The points which are Pareto optimal are also called efficient points. In fact, the term efficiency is used for the design variables and the term dominance for the function in the criterion space. The set of all Pareto optimal solutions constitute the Pareto front.

Multi-objective optimisation has two goals [Deb01]:

- find a set of solutions as close as possible to the Pareto Front,

- find a set of solutions as diverse as possible.

The points of the Pareto front are called non-dominated solutions. They are located on the boundary of the criterion space **Z**. Moreover, they are all equivalent: no solution is objectively better than another. Pareto solutions are such that any improvement in one criterion can occur only if another criterion is degraded. The selection of one of them over the other as final "best" solution is done by using subjective rules such as expert judgement and knowledge about the problem. Figure 2.3 illustrates the notion of Dominance and shows a Pareto front.
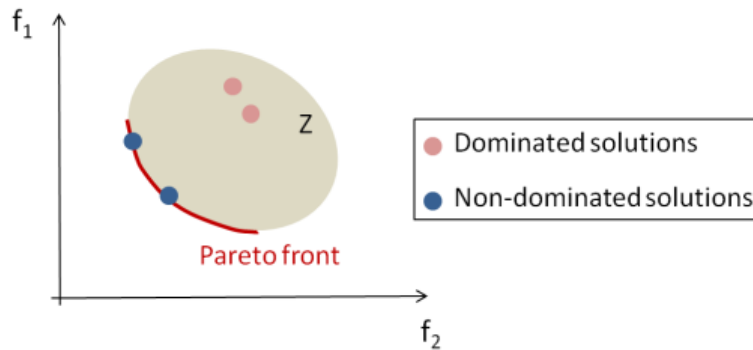
Figure 2.3: Example of dominance relation and Pareto front in the criteria space

2.3.2 Methods for multi-objective optimisation

Methods for solving multi-objective optimisation can be categorised in five main classes [MA04]:

- Methods with a priori articulation of preferences,

- Methods with *a posteriori* articulation of preferences,

- Methods with progressive articulation of preferences,

- Methods with no articulation of preferences,

- Evolutionary algorithms.

The previous classification depends on how the decision-maker specifies his/her preferences in the methods. Preference in this context refers to the opinion of the decision-maker on the importance of the objectives. This opinion is expressed thanks to preference functions (abstract functions (of points in the criterion space) in the mind of the decision-maker, which perfectly incorporates his preferences) or utility functions (functions which represent the relative importance of one objective depending on the decision-maker preferences). In the following subsection, the rationale of some common methods is presented.

2.3.2.1 Methods with *a priori* articulation of preferences

In this class of methods, the decision-maker states his preferences with *a priori* articulation meaning before the resolution of the optimisation problem: he indicates the relative importance of the objectives or desired goals before viewing points in the criterion space. It is the most common way to solve a multi-objective optimisation. Most of these methods incorporate parameters and use scalar functions. Scalar function is the term used in [MA04] to refer to the single objective function obtained after the combination of the components of a vector of objective functions (the term scalarisation method is coming from that). The idea is to transform the multi-objective optimisation into a mono-objective optimisation in order to be able to solve the problem with standard optimisation methods. These methods return a unique and single optimal point. Examples of such methods are Weighted Global Criterion, Goal Programming methods, Lexicographic or Exponential Weighted Criterion. We present the first three methods.

2.3.2.1.1 Weighted Global Criterion methods The Weighted Global Criterion method is a scalarisation method where all the objectives are combined to form one single function which is then minimised [Aro11, AP96]. The expression of this single function, called utility function, is given hereafter:

$$U = \left\{ \sum_{i=1}^{k} w_i [f_i(x) - f_i^o]^p \right\}^{\frac{1}{p}}$$

where:

- $F^o = (f_1^0, \ldots, f_k^0)$ is called utopia point (or ideal point) i.e. for each $i = 1, \ldots, k$, $f_i^o = \min \{ f_i(x) | x \in \Omega \}$.

- w is a vector of weights typically set by the decision-maker such that $\sum_{i=1}^{k} w_i = 1$ and $w_i > 0$ for $i = 1, \ldots, k$.

- The exponent p is also defined by the decision-maker. Its value depends on the distance measure the decision-maker wants to use.

The solutions obtained using this method depend on the value of w and p parameters. Depending on the value set for the parameter p, the Weighted Global Criterion method can be reduced to other common methods.

For example, the Weighted Sum method is a particular case of the weighted global criterion method with $p = 1$ and $f_i^0 = 0$. It is the most widely used approach because of its simplicity and ease of use. In this case, The utility function to be minimised is simply:

$$U = \sum_{i=1}^{k} w_i f_i(x).$$

The minimum of this utility function U is Pareto optimal if and only if all of the weights are positive.

Despite the simplicity of the Weighted Global Criterion method, it presents three main drawbacks [DD97]. First, the choice of the weights made by the decision-maker is not always intuitive even if some authors have developed methods for quantifying them (method based on fuzzy set theory [RR88] and ranking methods [YH95]). Second, the *a priori* choice of these weights does not guarantee that the final solution will be acceptable from the user point of view. Third, an uniform variation of the weights may not necessarily result in complete, accurate and uniformly distributed Pareto optimal points. Sometimes different weights lead to the same Pareto optimal points. Finally, it is not possible to find points on a non-convex part of the Pareto front.

2.3.2.1.2 Goal Programming methods   In this method, developed by [CCF55], the decision-maker first specifies goals, denoted b_j, that each criterion has to achieve in order to find a feasible solution. Then, he minimises the sum of the absolute deviation from the goals $\sum_{j=1}^{k} |d_j|$, where d_j is the deviation from the goal b_j for the jth objective. To model the absolute values, d_j is split into a negative deviation variable d_j^- and a positive deviation variable d_j^+, such that $d_j = d_j^+ + d_j^-$, with $d_j^+ \geq 0$, $d_j^- \geq 0$ and $d_j^+ d_j^- = 0$. d_j^+ denotes how much the decision has exceeded the goal and d_j^- denotes how far the decision is from the goal. Then, $| d_j |= d_j^+ + d_j^-$. Charnes and Cooper [CC77] present the formulation of this problem as in equation (2.4):

$$
\begin{aligned}
\min_{x\in\Omega,d^-,d^+} \quad & \sum_{i=1}^{k}(d_i^+ + d_i^-) \\
\text{s.t.} \quad & f_j(x) + d_j^- d_j^- = b_j, \quad j = 1,\ldots,k \\
& d_j^+, d_j^- \geq 0, \quad j = 1,\ldots,k \\
& d_j^+ d_j^- = 0, \quad j = 1,\ldots,k.
\end{aligned} \tag{2.4}
$$

The drawback of Goal Programming method is that there is no guarantee to obtain a solution which is Pareto optimal. Three popular subclasses of Goal Programming are Weighted Goal Programming (also called archimedean Goal Programming), Lexicographic Goal Programming and Min-max Goal Programming.

2.3.2.1.3 Lexicographic method   In this method, the decision-maker first sorts the objective functions by decreasing order of importance. Then, the following mono-objective optimisation problems are successively solved for $i = 1, ..., k$,

$$
\begin{aligned}
\min_{x\in\Omega} \quad & f_i(x) \\
\text{s.t.} \quad & f_j(x) = f_j^* \quad j = 1,\ldots,i-1,
\end{aligned}
$$

where f_j^* is the minimum of the jth objective function found in the jth iteration and i is the position of the function in the classification.

The main drawback of Lexicographic method is that the objectives have to be ordered before the optimisation. This arbitrary ordering leads generally to different solutions [CS02].

2.3.2.2 Methods with progressive articulation of preferences

Methods with progressive articulation of preferences are also called interactive methods. In this class of methods, the decision-maker continuously gives input to the algorithm doing the resolution of the optimisation. The inputs may concern the search direction, reference points,

weight vector ... These methods produce only one solution. Surrogate Worth Trade off method, Step method, Reference point method, Fandel method, Jahn method or Geoffrion-Dyer-Feinberg method are examples of such methods. We will not give more details on these methods but the reader can refer to the study of Miettinen [Mie99].

2.3.2.3   Methods with no articulation of preferences

Methods in this class do not require any articulation of preferences from the decision-maker. Moreover, these methods produce only one point of the Pareto optimal set. They can be split essentially into two classes: exponential sum and objective product. Some examples of these methods are the Min-max method, the Nash Arbitration method, the Global Criterion, or the Objective Product method.

2.3.2.3.1   Min-Max Method   The Min-max formulation is derived from the general Weighted Sum method where:

- there is no weights because the decision-maker does not have to express his/her preferences anymore ;

- and $p = \infty$: the norm used is the $\mathbf{L}_\infty$ norm.

The problem is formulated as follows:

$$\min_{x \in \Omega} \left\{ \max_i \ f_i(x) \right\}.$$

More information on this method can be found in [CCC99], [Osy78] and [Rao87].

2.3.2.3.2   Nash Arbitration   Nash Arbitration method is derived from Game Theory. The aim is to maximise the product of the objectives. It is formulated as follows:

$$\max_{x \in \Omega} F_g(x) = \prod_{i=1}^k (s_i - f_i(x)),$$

where $s_i > f_i(x)$ is selected as an upper limit value on each objective function. The solution proposed by this method depends on s_i values.

2.3.2.4   Methods with *a posteriori* articulation of preferences

In this class of methods conversely to methods with *a priori* articulation of preferences, the decision-maker states his/her preferences after the resolution of the optimisation. These methods have been developed because sometimes it is not easy for the decision-maker to state his/her preferences *a priori* or because he/she prefers to make a choice directly on a range of solutions. Weighted methods (aforementioned) are also a mean to generate Pareto optimal solutions. This is done by executing successively several mono-objective optimisations by varying the parameters. For example, one can vary the value of w_i in order to obtain a subset of the Pareto front. However, this successive execution while uniformly varying the parameters does not guarantee obtaining an even spread of the optimal solutions that accurately represent the Pareto Front. Consequently, some algorithms have been developed for this purpose. They are also called *Generate-first-choose-later* or *cafeteria* [BR00]. As an example, we find the Physical Programming or the Normal Boundary Intersection.

2.3.2.4.1   Physical Programming   The Physical Programming method was first developed for *a priori* articulation of preferences. It can be adapted for *a posteriori* articulation of preferences [MM02]. One of its advantages is that it generates points on the convex and non-convex part of the Pareto front. It consists in minimising the following function [MSC⁺99]:

$$g(x) = \log_{10} \left(\frac{1}{n_{sc}} \sum_{i=1}^{n} \bar{g}_i(g_i(x)) \right),$$

where

- g_i represents an objective to be optimised, or a constrained quantity to satisfy ;

- $\bar{g}_i$ is a preference function introduced according to the decision-maker preferences. Hard class-functions reflect the presence of constraints, and soft class-functions reflect the preferences settled for objective functions ;

- n_{sc} is the number of soft criteria.

Some region boundaries are introduced to translate the degrees of preferences of the decision-maker. They are composed of six ranges:

- unacceptable,

- highly-undesirable,

- undesirable,

- tolerable,

- desirable,

- highly-desirable.

The interest in introducing these regions is to compare the evolution of objective functions in a qualitative way. For instance, it can be better for one criterion to travel across the tolerable region than all the others to travel across the desirable region.

For more details on this method, the reader can refer to [Mes96] and [MM02].

2.3.2.4.2   Normal Boundary Intersection method   Normal Boundary Intersection (NBI) is another method for obtaining an approximation of the Pareto set in both cases of convex and non-convex Pareto optimal set. Das and Dennis [DD96] built this approach based on the idea that the intersection between the boundary of the feasible criterion space **Z** denoted ∂Z and the normal set of the Convex Hull Individual Minima is a point on the portion of ∂Z containing the efficient points. To present this method, we first have to define the concept of Convex Hull Individual Minima further denoted CHIM.

Let:

- x_i^* be the global minimum of the objective function f_i and $f_i^* = f_i(x_i^*)$,

- $F_i^* = F(x_i^*)$,

- $F^* = (f_1^*, \ldots, f_k^*)$ be the utopia vector.

We can define the $k \times k$ matrix Φ such that $\Phi = [F_i^* - F^*]$ for $i = 1, \ldots, k$ and which has the following properties:

- $\Phi(i, i) = 0$, for all $i = 1, \ldots, k$,

- $\Phi(i, j) > 0$, for all $i, j = 1, \ldots, k$ and $i \neq j$.

The CHIM is defined as:

$$CHIM = \left\{ \Phi\omega : \omega \in \mathbb{R}^k, \sum_{i=1}^{k} \omega_i = 1, \omega_i \geq 0 \right\}.$$

For a given convex weighting ω, $\Phi\omega$ represents a point in the CHIM. $\tilde{n}$ denotes the unit normal vector to the CHIM pointing to the origin. Then, $\Phi\omega + t\tilde{n}$, where t is a scalar, represents the set of points on that normal direction. The point of intersection between the normal and the boundary of the feasible domain can be obtained by solving the following optimisation problem:

$$\begin{aligned} \max_{x,t} \quad & t \\ \text{s.t.} \quad & \Phi\omega + t\tilde{n} = F(x), \\ & g(x) \leq 0, \\ & h(x) = 0. \end{aligned} \tag{2.5}$$

This method finds the part of the boundary of $\mathbf{Z}$ that contains the Pareto optimal set. Nevertheless, the method can also generate points which are not Pareto optimal. But this is not a disadvantage because these points can be useful in order to construct a "smoother approximation of the Pareto boundary".

Remark

In most of the previous methods described (except for methods with *a posterior* articulation of preferences), the objectives are combined into one single objective to solve the optimisation problem. This combination guarantees to find a solution which is Pareto optimal but the result is one single point. In addition, these methods may not be efficient with an objective function which is noisy or which has a discontinuous variable space. They can also be computationally very expensive. Moreover, the main drawback of the scalarisation methods is that the result fully depends on the weights assigned by the decision-maker: the decision-maker must have strong knowledge on the priority of the objective function in order to assign the weights. Thus, the solutions obtained for a same problem can be different.

2.3.2.5 Evolutionary algorithms

Evolutionary Algorithms (further denoted EA) are heuristic-based algorithms developed to solve multi-objective problems. They are gaining attentions from researchers because of their efficiency and robustness in finding a set of global solutions. They mimic natural evolutionary principles to solve complex optimisation problem. They incorporate vocabularies related to biology domain such as population, genotype, genetic code, self-adaptation... Candidate solution of the optimisation is represented by an individual in a population. [Bӱ96] says that this individual may also be "a container of current knowledge about the laws of the environment". The initial population evolves toward optimal point thanks to processes such as *mutation*, *selection* and *recombination*.

There are three mainstream in EA:

- Evolution Strategies developed by [Rec71] and [Sch81],

- Evolutionary Programming developed by [FOO66] (see also [Fog95]),

- Genetic Algorithm developed by [Hol75] and reviewed by [Gol89].

In EA, we also find algorithms such as Simulated Annealing, Taboo Search or Ant Colony. In this study, we describe Genetic Algorithm only. The authors in [CC10] make a list of references on evolutionary multi-objective optimisation.

Genetic Algorithms (GA) have been developed by Holland [Hol75] and are inspired by natural evolutionary principle formulated by Charles Darwin [Dar59]. They are part of derivative-free methods which make them effective regardless the nature of the objective function (discontinuous, non-differentiable or highly nonlinear). Moreover, they can be used to solve both constrained and unconstrained optimisation problems. They have been used in several domains such as science, commerce (trade) or engineering, for instance in Airbus with [Bad05] and [TC11] works. Before going further, a brief description of the principle of GA is given. We review the related vocabulary.

A population is a set of candidate solutions of the design space. A population is composed of *individual* (one point of the population). A *chromosome* (or *genotype*) is another name for individual but taken as a set of values. It represents a unique solution in the design space. A chromosome is composed of several discrete units called *genes*. A gene represents a component of the design vector (degree of freedom of the optimisation). The genes are assumed to be binary digits, and a *generation* is an iteration of the algorithm. The mapping between the design space and the chromosomes is called *encoding*.

The fitness function is another important element of GA. It assigns a fitness value to each individual. This fitness value can be seen as the survival probability of the individual: it represents the capacity of the individual to solve the problem. The more likely to fit the solution is, the higher its fitness value.

In GA, the generation of a new solution is done by using the following operators:

- selection operator,

- crossover (or reproduction) operator,

- mutation operator.

The selection operator allows to choose the individuals, called parents, which are retained for reproduction, i.e. to breed a new generation called children. This operator does not create new individuals, it only selects the more likely to fit ones. Indeed, it enables to keep the population size globally constant as worst individuals (between parents and children) are eliminated. The selection is done according to the fitness function. There exist several selection methods such as Elitism, Roulette wheel selection, Rank selection, Local selection, Steady-State selection or Tournament selection. In the simplest method, the parents are randomly chosen.

The crossover operator (also called reproduction operator) has the objective to combine the genes of two chromosomes called parents in order to form two new chromosomes called children. The way the parents are chosen depends on the type of algorithm used for the resolution. After selecting the parents, the crossover can be done using several strategies such as single point crossover, two points crossover or uniform crossover. Figure 2.4 illustrates the single point crossover method: first, a random single point is chosen on both parents; then, all data beyond that point are permuted in order to form the children.

The mutation operator introduces random changes in the value of one or more of the chromosome's gene. The probability of mutation is very small and depends on the length of the

Parents   Children

1 0 1 | 0 1 1 1 0 0   ⇨   1 0 1 0 0 1 0 1 1

1 1 0 | 0 0 1 0 1 1   1 1 0 0 1 1 1 0 0
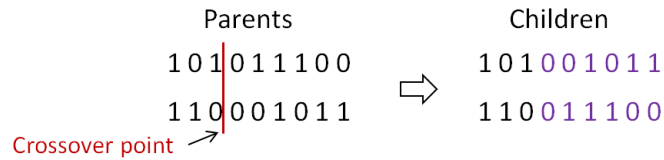
Crossover point

Figure 2.4: Single point crossover

chromosomes. This operator helps in keeping diversity in the population and helps the population to escape from a local optimum. The reader can find more detail on mutation operator in [Deb01]. An an example, the chromosome **11010011** can be mutated in **11011011**.

Figure 2.5 displays the principle of a simple GA.
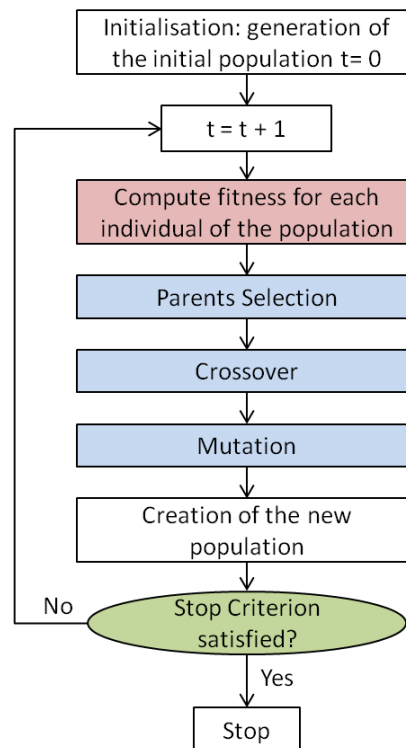
Figure 2.5: Flow chart of a simple genetic algorithm

GA are suitable to a large set of problems (multi-dimensional, non-differentiable, etc). There is no need to have a priori knowledge on the problem (derivability, convexity, etc): the main part of the work is in the encoding phase. Moreover, the calculation times does not increase with the number of variables but with the size of the population.

The main drawback of GA is the tuning of the parameter values such as the number of generation, the mutation rate, the size of the population...

The literature proposes a large number of GA. A short list of some of them is given in [KCS06]. Vector Evaluated Genetic Algorithm (VEGA) presented by Schaffer in 1985 [Sch85] is one of the first GA developed for multi-objective optimisation problems. After VEGA, many other algorithms have been proposed such as Multiple Objective Genetic Algorithm (MOGA) [FF93], Niched Pareto Genetic Algorithm (NPGA) [HNG94], Strength Pareto Evolutionary Algorithm (SPEA) [ZT99], Weight-Based Genetic Algorithm (WBGA) [HL92], Multi-objective Evolutionary Algorithm (MEA) [SL02], Non dominated Sorting Genetic Algorithm

(NSGA) [SD94] or again Fast Non dominated Sorting Genetic Algorithm (NSGA-II) [DPAM02]. In this study, we will use the Non dominated Sorting Genetic Algorithm (NSGA).

NSGA have been developed by Srinivas and al. [SD94]. The difference between NSGA and other basic GA comes from the selection phase: NSGA uses a ranking selection method based on non-domination sorting and a fitness sharing strategy based on niching in order to maintain diversity in the population [DPAM02, MA04]. The crossover and mutation phases are kept the same. The flow chart of NSGA is displayed in figure 2.6.
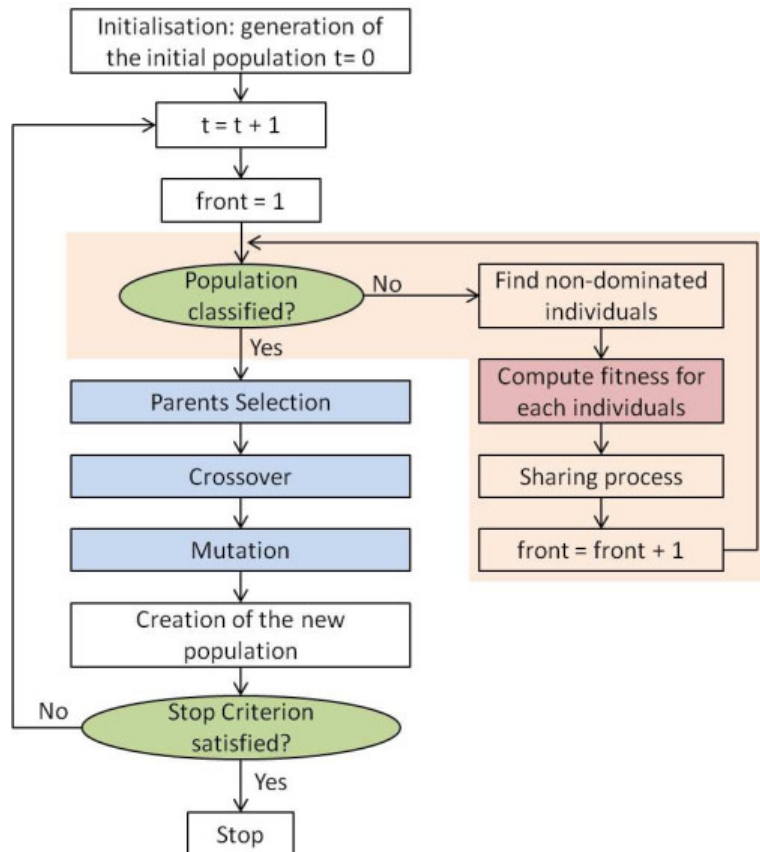
Figure 2.6: Flow chart of NSGA method

Step 1 in NSGA consist in ranking the population into several front. First, we look for a set F_1 of individuals which are non-dominated by the other individuals of the population. Solutions in this first front F_1 are said to belong to the best non-dominated front in the population. Since all solutions in F_1 are equally important in terms of their closeness to the pareto-optimal front relative to the current population, the same high dummy fitness value is assigned to all of them in order to guarantee an equal reproduction potential. Then, the rest of the population is processed of the same way until all the population is entirely classified into several fronts $F_1, F_2, \ldots, F_i$ where i is the total number of front (see figure 2.7). Note that the dummy fitness value attached to a front F_i is kept smaller than the minimum dummy fitness value of the previous front F_{i-1}. Doing that, the individuals of the first front F_1 get more copies as they have the higher dummy fitness value.

Task 2 in NSGA is to share the individuals. This process is done in each front F_i individually. Individuals of Front F_1, affected with their dummy fitness value, are the first ones to be shared thanks to a sharing function. The aim of this sharing process is to keep diversity in the population. The idea is to degrade the original fitness value of an individual by a quantity which is proportional to the number of individuals which are around it: the more there are
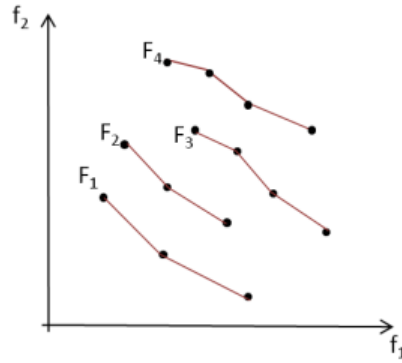
Figure 2.7: NSGA ranking process [Bau12]

individuals located in the individual's neighbourhood, the more its fitness value is degraded. Once the individuals of the first non-dominated front F_1 have been shared, the sharing process is repeated for the individuals of the other front F_2 to F_i.

The sharing function between two individuals i and j in the same front is expressed as follows:

$$Sh(d_{ij}) = \begin{cases} 1 - \left(\frac{d_{ij}}{\sigma_{share}}\right)^2 & \text{if } d_{ij} < \sigma_{share}, \\ 0 & \text{otherwise}, \end{cases}$$

where

- d_{ij} is the distance between two individuals i and j in the current front (d_{ij} defines the size of the neighbourhood of an individual). with n_1 is the number of individuals in front F_1, x_k^{max} and x_k^{min} respectively the upper and the lower bounds of x_k.

- σ_{share} is the maximum distance allowed between any two individuals to become members of a niche.

A parameter niche count is calculated for the ith individual by adding the sharing function values for all individuals of the current front as follows:

$$nC_i = \sum_{j=1}^{n_k} Sh(d_{ij}).$$

Finally, the shared fitness value of each individual is computed by dividing its dummy fitness value denoted f_i with its niche count:

$$f_i' = \frac{f_i}{nC_i}.$$

2.3.3 Summary

In this section, we have introduced the multi-objective optimisation problems and we have presented a short state of the art of methods existing to solve this category of optimisation problems. In the next section, we introduce probabilistic and statistical theories.

2.4 Probabilistic and statistical definitions and notations

Many uncertainty management tools are based on statistical and probabilistic concepts. So before going further, the following is an introduction to the basics of the probability theory. The reader can refer to Walter Appel's book [App13]. The following definitions are extracted from [Sap06] and [Bea04].

2.4.1   Probability space

A *random experiment* is an experiment for which the results can not be predicted beforehand. The set of all possible outcomes of the experiment is called the *sample space* and is generally denoted Ω. An element of Ω is an *elementary event* denoted ω, with $\omega \in \Omega$. A random event is defined as a set of outcomes ω.

The intersection of two events A and B is denoted $A \cap B$. Respectively, their union is denoted $A \cup B$.

The set of possible events associated to the sample space Ω is called an *event space*. This space forms a class $\mathcal{F}$ of part of Ω also called σ-algebra. It is defined by the three following axioms:

1. if $A \in \mathcal{F}$ then $\bar{A} \in \mathcal{F}$ (with $\bar{A}$ complement of A),

2. if A_1, A_2, ..., $A_n \in \mathcal{F}$ then $\bigcup_{i=1}^{n} A_i \in \mathcal{F}$,

3. $\mathcal{F}$ is non-empty then $\Omega \in \mathcal{F}$.

The pair $(\Omega, \mathcal{F})$ is called *probabilisable space*, with $\mathcal{F}$ a σ-algebra associated to Ω.

Each event is equipped with probabilities. They correspond to positive numbers which lie between 0 and 1. The probability of an event E is denoted $\mathbb{P}(E)$. This probability is defined by the axiom of Kolmogorov given by definition 2.4.1.

Definition 2.4.1. [**Kolmogorov axiom**] A probability is an application $\mathbb{P}$ with $\mathbb{P}$: $\mathcal{F} \mapsto [0,1]$ such as:
$$\mathbb{P}(\Omega) = 1,$$
$$\mathbb{P}(A) \geq 0, \quad \forall A \in \mathcal{F},$$
$$\mathbb{P}(A \cup B) = \mathbb{P}(A) + \mathbb{P}(B), \quad \forall A, B \in \mathcal{F} \text{ such that } A \cap B = \emptyset.$$

From the Kolmogorov axiom, we have the properties of a *probability measure*. $\mathbb{P}$ is a probability measure if:
$$\mathbb{P}(\emptyset) = 0,$$
$$\mathbb{P}(\bar{A}) = 1 - \mathbb{P}(A),$$
$$\mathbb{P}(A) \leq \mathbb{P}(B), \quad \text{if } A \subset B,$$
$$\mathbb{P}(A \cup B) = \mathbb{P}(A) + \mathbb{P}(B) - \mathbb{P}(A \cap B).$$

The triplet $(\Omega, \mathcal{F}, \mathbb{P})$ is called a probability space.

2.4.2   Random variable

A random variable (or stochastic variable) X is a variable whose values are outcomes from a random process. More precisely, a real random variable is an application from the probability space $(\Omega, \mathcal{F}, \mathbb{P})$ to the space of real number $\mathbb{R}$. There are two types of random variables: continuous variable and discrete variable. A random variable X is continuous if its domain of definition contains a real interval. Inversely, X is discrete if it can take a finite or a countable infinite set of discrete values. A random variable is generally associated with two functions: Cumulative Distribution Function and Probability Density Function.

2.4.2.1   Cumulative Distribution Function

The Cumulative Distribution Function (CDF) of a random variable X is the application $\mathbb{F}$ from $\mathbb{R}$ to [0,1] given by:
$$\mathbb{F}(x) = \mathbb{P}(X \leq x).$$

It is an increasing monotoneous function which describes the probability for the random variable X to be less or equal to a value x. It is defined for both discrete and continuous random variables. An example of such function is exhibited in figure 2.8.
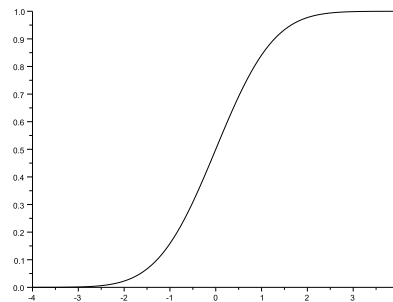
Figure 2.8: Cumulative Distribution Function of a continuous variable

By using the CDF, it is possible to determine the probability of x lying in a range $[a, b]$ with $a \leq b$:

$$\mathbb{P}(a \leq X \leq b) = \mathbb{F}(b) - \mathbb{F}(a).$$

2.4.2.2 Probability Density Function

The Probability Density Function (PDF) is defined only for continuous random variables. Indeed, a continuous random variable X has a density function denoted f_X and defined by:

$$P(a \leq X \leq b) = \int_a^b f_X(x)dx.$$

The PDF f_X is a positive function with integral equal to 1:

$$\int_{\mathbb{R}} f_X(x)dx = 1.$$
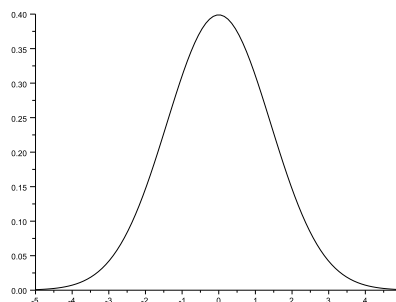
Figure 2.9: Probability Density Function

2.4.3 Statistical measure

A random variable X with a known PDF is characterised by some statistical criteria.

2.4.3.1 Mean

The mean (also called mathematical expectation or expected value) of a random variable X, denoted $E(X)$, is defined by

- $E(X) = \int_{\mathbb{R}} x f_X(x) dx$, if X is a continuous variable (a variable with a density function)

- and $E(X) = \sum_i x_i \mathbb{P}(X = x_i)$, if X is a discrete variable.

It gives the measure of the central tendency of values taken by X. The mean is also called first moment. It is commonly denoted μ_X.

2.4.3.2 Variance and Standard-deviation

The variance of a random variable X, denoted $V(X)$ is defined by:

$$V(X) = E((X - E(X))^2) = E(X^2) - (E(X))^2.$$

It is a measure of the dispersion of X around its mean value $E(X)$. The variance is called second moment and commonly denoted $\sigma^2 =$. From the definition of the variance, we can define the standard-deviation σ. It is the square root of the variance, meaning $\sigma_X = \sqrt{V(x)}$.

2.4.3.3 Other moments

The moment of order n is defined by:

$$m_n(X) = E(X^n).$$

In the literature, we can find the notion of centred moment and normalised (or standardised) moment. The centred moment of order n is defined by:

$$m_n(X) = E\left((X - E(X))^n\right),$$

and the normalised moment of order n is given by:

$$m_n(X) = E\left(\left(\frac{X - E(X)}{\sigma}\right)^n\right).$$

Let us particularly consider the third and the fourth moments. These two moments can give information about the shape of the distribution of a random variable X.

The third normalised moment is called *skewness* coefficient. It is the moment of order 3 and its centred expression is given by $E\left((X - E(X))^3\right)$. Its normalised expression corresponds to the centred moment of order 3 normalised by the cube of the standard-deviation σ and is defined by:

$$
\begin{aligned}
m_3(X) &= E\left(\left(\frac{X - E(X)}{\sigma}\right)^3\right) \\
&= \frac{E\left((X - E(X))^3\right)}{\sigma^3}.
\end{aligned}
\tag{2.6}
$$

The skewness coefficient gives information about the symmetry of the density function. A distribution with a negative skewness means that the distribution is asymmetric to the left (with a tail longer to the left than to the right). A distribution with a positive skewness means that the distribution is asymmetric to the right. And a distribution with a skewness equal to zero means that the distribution is symmetric. As an example, the normal density function has a

symmetrical distribution then its skewness coefficient is equal to zero.

The fourth normalised moment is called *kurtosis* coefficient, also called Pearson Kurtosis coefficient. Its centred definition is given by $E((X - E(X))^4)$ and its normalised one is defined by:

$$
\begin{aligned}
m_4(X) &= E\left(\left(\frac{X - E(X)}{\sigma}\right)^4\right) \\
&= \frac{E\left((X - E(X))^4\right)}{\sigma^4}.
\end{aligned}
\tag{2.7}
$$

The kurtosis coefficient gives information about the "peaky" aspect of the distribution: it measures the degree of the concentration of the observations in the distribution tail. The higher the value of the kurtosis coefficient, the more peaky the distribution. Kurtosis values are used to be compared to the kurtosis value of the Gaussian distribution which is equal to 3. A new expression of the kurtosis coefficient is given taking the Gaussian distribution as reference. It is called the excess Kurtosis:

$$
m_4(X) = \frac{(n+1)n}{(n-1)(n-2)(n-3)} \sum_{i=1}^{n} \left(\frac{x_i - E(X)}{\sigma}\right)^4 - \frac{3(n-1)^2}{(n-2)(n-3)},
\tag{2.8}
$$

where n is the size of the sample. A distribution with a positive excess Kurtosis coefficient is peaky: it rises up and falls abruptly. It is called *leptokurtic* distribution. One with a negative excess Kurtosis coefficient is flattened and is called *platykurtic* distributions.

2.4.3.4 Covariance

The covariance of two random variables X and Y is given by:

$$
cov(X,Y) = E(XY) - E(X)E(Y) = E\Big((X - E(X))(Y - E(Y))\Big).
$$

The covariance is a measure of the strength of the *linear* link between two random variables. The variance is a particular case of the covariance: $cov(X, X) = E\Big((X - E(X))(X - E(X))\Big) = V(X)$. The covariance coefficient depends on the units of the variables X and Y. It can be meaningless to compare the covariance of variables which are totally different (such as the mass of the fuselage and the sweep angle). To avoid this drawback, one can use the correlation coefficient.

2.4.4 Correlation

2.4.4.1 Correlation coefficient

The correlation coefficient, also called Pearson's coefficient in the literature, is a normalisation of covariance coefficient by the product of the standard deviation of the variables and is independent of the variable's units. It gives information on the existence of a linear relation between two variables. The correlation coefficient between X and Y is given by:

$$
r(X,Y) = \frac{E[(X - \mu_X)(Y - \mu_Y)]}{\sigma_X \sigma_Y},
\tag{2.9}
$$

where $E[(X - \mu_X)(Y - \mu_Y)]$ is nothing but the covariance between X and Y commonly denoted $cov(X,Y)$. The correlation coefficient $r(X,Y)$ is always in $[-1,1]$ and it is equal to:

- 0 if there is no linear relation between the samples;

- 1 if there is a maximal positive linear relation and it is in the range [0,1] if there is a simple positive linear dependence;

- -1 if there is a maximal negative linear relation and it is in the range [-1, 0] if there is a simple negative linear relation.

The correlation coefficient presents some drawbacks:

- It is sensitive to outliers i.e. values which are far from the others and which are considered as exceptions.

- It only gives information on the existence of a linear relation between two variables. Thus, two variables X and Y which are independent have a correlation coefficient equal to zero. Note that the converse is wrong. The correlation coefficient can be equal to zero even though there is a 'non linear' functional relation between the variables. Consider the function $Y = X^2$. Y is completely dependent of X but their correlation coefficient is equal to zero.

- It is a biased coefficient but this bias disappears when the size of the sample increases.

- Its interpretation is sensitive. A high correlation coefficient between two variables doesn't necessarily mean that there exists a causal relationship between them. It could just mean that the phenomenon represented by the variables have the same origin. For instance, let us consider the correlation between "the number of firemen on the fire area" and "the repayment price demanded by the insurance company". A study has shown that there is a high correlation between these events but it is obvious that in reality it is not the case. Although, if we consider the third event "the fire gravity", we see that the first two ones have the same cause: the third one.

In the case where the relation between the variables is non linear but monotonic, the correlation coefficient can highlight the existence of a relation but is not able to give information on the strength of the relation. One solution to avoid that problem is to make a transformation (of the variables) in order to obtain a linear relation. Nevertheless, this way of doing presents several drawbacks: the choice of the transformation function is not always obvious. Moreover, the number of transformation increases with the number of relationship to analyse. In this case, other correlation coefficients such as the Spearman's one are more relevant.

2.5 Optimisation with uncertainty

The optimal solution proposed by the deterministic optimisation methods does not take into account uncertainty that can affect the system. Thus, the performance of the system designed with this solution can be highly degraded when changes occur. Sometimes, it could be wiser to retain a solution that may not be optimal but that is less sensitive to changes. This type of solution is said to be robust.

Let us consider an example with figure 2.10. In this figure, x_1^* appears as the global optimal solution of the problem. Nevertheless, if any perturbation occurs, the performance associated to x_1^* can be highly degraded whereas the performance associated to x_2^* remains stable. In this example, x_2^* is the robust solution. Thus, it could be more interesting to consider solution x_2^* which is not optimal but insensitive to changes rather than solution x_1^* which is optimal but highly sensitive to changes.

Robust design has been introduced by Taguchi at the end of the 1940s [TCT00, Pha89]. The purpose was to improve the quality of engineering design process by achieving two goals: minimise the variability of the performance by integrating noise factors information and adjust
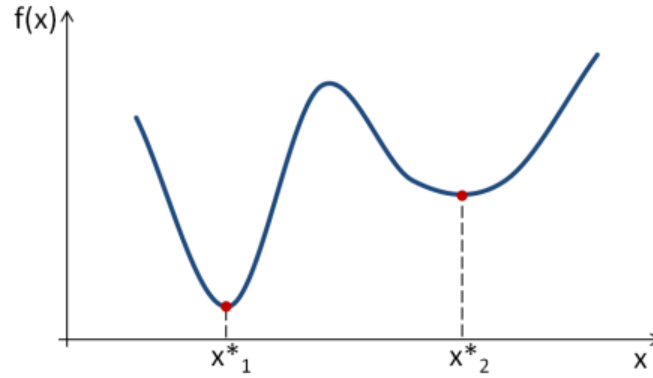
Figure 2.10: Illustration of the notion of robustness on a single objective optimisation

the mean performance to the desired target. Since this introduction by Taguchi, many methods have been put in place to launch robust optimisation.

To be able to determine robust solution, we first have to identify and quantify the uncertainty which affects the system. In the next part, we first present the notion of uncertainty. We review the different existing sources, the steps to follow in order to assess the uncertainty and we present the existing method to model the uncertainty. We then make a review of the robustness measures and we end up with the concept of robust optimisation.

2.5.1   Uncertainty

2.5.1.1   Definition and Source

Uncertainty is a concept increasingly used in a wide range of domains as economics, automotive [NAT07] or aeronautics [DDST98]. Several definitions of this concept can be found in the literature. Uncertainty is described in [Web07] as "a lack of certainty resulting from inaccuracy of input parameters, analysis process or both". In [CFS04], it is seen as "the fact that measured values frequently do not match the true values, but differ from them in a probabilistic manner". And in [Vos00], uncertainty is said to be "the assessor's lack of knowledge about the parameters that characterise the physical system that is being modelled". Moreover, when looking for information about sources of uncertainty and error in modelling of complex systems, one can also find various expressions referring to those concepts (e.g. random and epistemic uncertainty, model uncertainty [Dra95, NA03], predictive model uncertainty [Ril11, MMU99], model error [Kim01, Lun05]. Before going further, emphasis is put on the definition of these different sources in order to clarify the meaning we put in the concepts of uncertainty and error.

The classical definition of *error* is given by the experimentalists [Eur00]. It is defined as the difference existing between the measured value and the true value of a physical variable. In the context of complex system modelling, based on this definition, the error of model would be the difference between the model prediction and the true value of the physical variable. But it is more subtle. In [ODR$^+$02], error of model is defined as "a recognisable inaccuracy in any phase or activity of modelling and simulation that is not due to lack of knowledge". In this study, the error of model is defined as the discrepancy existing between the model prediction and the experimental value of the physical variable (as the true value is not known). This discrepancy is, among others, the consequence of simplifications made during the modelling process and corresponds to "acknowledged error" defined by the authors in [ODR$^+$02]: "an error can be either acknowledged or unacknowledged [...]. Examples of acknowledged error are assumptions and approximations made to simplify the modelling of a physical process".

According to [BS07] there are two ways to classify uncertainty: from an epistemological point of view and from a system one. We present both of them.

From a system point of view, uncertainty affecting the system are of 4 types (see figure 2.11):

- Uncertainties coming from the environment and from operating conditions. This type of uncertainty can not be controlled. It is represented by the parameter α in figure 2.11.

- Uncertainty on the design parameters. Generally, the system is realised with a certain degree of precision because it is not possible to make a design with a perfect accuracy. This type of uncertainty is introduced into the system by the parameter β.

- Uncertainty on the system output. It comes from imprecision in the evaluation of the system output and performance. Measuring errors and approximation errors (due to the use of models representing the reality) are contained in this type of uncertainty. They are represented by the parameter γ.

- Feasibility uncertainty. This type of uncertainty is related to the fulfilment of the constraints of the problem. It has an effect on the design space and not on the objective function. This last type is not represented on figure 2.11 because it can be seen as the consequences of the effect of the first three types of uncertainty.
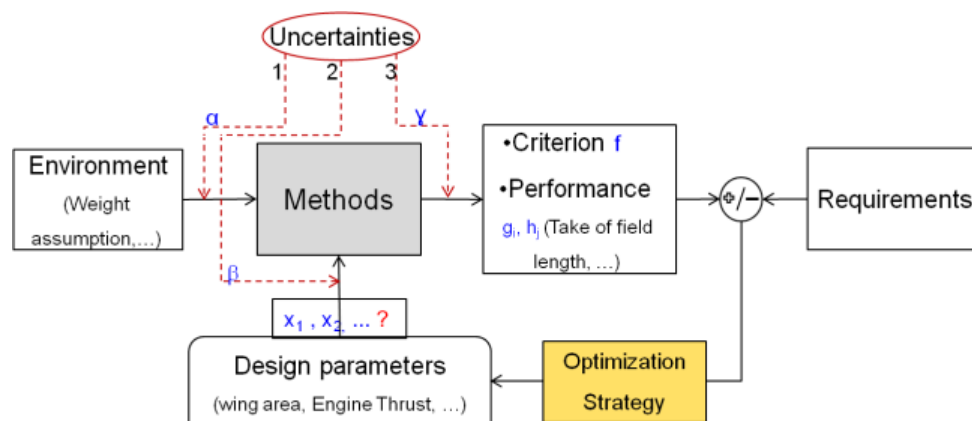
Figure 2.11: Uncertainty source from a system point of view

From an epistemological point of view, uncertainty can be categorised in two main groups: random uncertainty and epistemic uncertainty.

Random uncertainty is the inherent variation associated with the physical system under consideration or associated to the environment (meteorological, geographical variability). It is also called irreducible uncertainty, inherent uncertainty, stochastic uncertainty, objective uncertainty, and variability. This kind of uncertainty is completely out of control and does not decrease (even if the project gets in maturity). Random uncertainty is generally handled with probability methods through a PDF or a CDF.

Epistemic uncertainty refers essentially to a lack of knowledge or need for simplification of engineers about the behaviour of the system or the physic being modelled. It is also called reducible uncertainty, subjective uncertainty, and ignorance. In [ODR+02], the authors define it as "a potential inaccuracy in any phase or activity of the modelling process that is due to lack of knowledge". This type of uncertainty can be reduced through increasing knowledge about the physical system, increasing data available, the use of higher fidelity models, expert judgement, etc. Epistemic uncertainty is generally handled with Interval Analysis, Fuzzy Set theory, Possibility theory, Dempster-Shafer Evidence theory or with Second-order Probability [SP+09].

Let us consider the example of drag estimation in aircraft design process. Uncertainty appears to be linked with time and the use of models with higher and higher fidelity allows to reduce uncertainty as design time goes (see Figure 2.12.) At preliminary design phase, drag is estimated with Semi-Empirical Methods (SEM). At this stage, the uncertainty is large, the only data available for the studies are data about previous aircraft and the modelling methods are "rough". At detailed design phase, drag is first computed thanks to Computational Fluid Dynamic (CFD), then corrected by wind tunnel testing. CFD calculations are based on numerical integration of Navier-Stokes equations. This method is more precise than SEM and allows to reduce the uncertainty. Wind tunnel testing also helps to further reduce the uncertainty. Finally, at test and evaluation phase, drag is estimated from measurements made during flight tests.
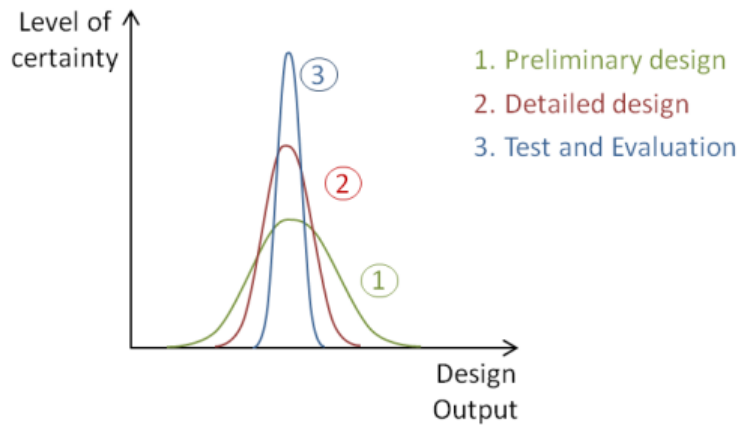
Figure 2.12: Evolution of uncertainty on drag estimation in aircraft design

During the last decade, focus has been put on the differentiation of these two types of uncertainty even if it is not always easy [Apo99, HB96]. This distinction helps, while trying to reduce the uncertainty, to focus on epistemic uncertainty which is reducible rather than random uncertainty which is not [Apo99]. Moreover, in some cases, both random and epistemic uncertainties can be propagated to obtain better results. It is the issue of Second Order Probability method [Zdr93]. In this study, we will focus on the assessment of epistemic uncertainty.

Uncertainty sources which create variation in the model prediction can be categorised in 4 groups:

- Simulation variability,

- Model parameter uncertainty,

- Structural model uncertainty (or model form uncertainty), and

- Predictive model uncertainty.

Simulation variability (also called numerical errors, numerical approximation) is related to uncertainty inherent to computational implementation. It includes truncation errors, iterative convergence errors, discretization error, round-off error,... Model parameter uncertainty refers to the uncertainty in parameters assigned to the model. It comes from a lack of knowledge (or limited amount of information) on the good values to assign to the model parameters. This concerns both input parameters and internal parameters of the model (*e.g.* what is the appropriate value for a constant in the model?). Structural model uncertainty refers to uncertainty related

to the structure of the model itself. Various models can emerge while trying to represent the behaviour of a physical system. The differences between these models come from the assumptions taken, their mathematical form, the levels of fidelity,... Predictive model uncertainty refers to the uncertainty existing on the error committed by the model. If y_{true} is the reality and $f_i(x)$ is the model prediction, then we have:

$$y_{true} = f_i(x) + \epsilon_{f_i}, \tag{2.10}$$

where ϵ_{f_i} is the error of model f_i. The aim is to quantify the predictive uncertainty on ϵ_{f_i}. Its quantification helps to know how good the model is in the predictability of the system behaviour.

2.5.1.2 Uncertainty assessment steps

Uncertainty assessment is split in four main steps:

1. Case-study specification,

2. Uncertainty Modelling,

3. Uncertainty Propagation,

4. Sensitivity Analysis (or Importance Ranking).

The goal of the step 1 is to describe the case-study. The specification of uncertainty sources (distinction between uncertain and fixed variable), model and criteria (variable of interest, quantity of interest,...) and uncertainty assessment framework (deterministic, probabilistic,...) are done at this stage.

The goal of step 2 is to characterise and quantify the source of uncertainty detected in step 1. This quantification will depend on the type of uncertainty framework selected. For example, it can be done by using a range in a deterministic framework, a PDF in a probabilistic framework, etc.

The goal of step 3 is to propagate the uncertainty which have been identified and quantified in the previous steps. The uncertainty is carried out through the system, in order to measure the effect of this uncertainty on the variable of interest.

The goal of the last step is to analyse the sensitivity of the component of the uncertain variables with respect to a given quantity of interest in the output of the system.

Several tools have been developed to perform uncertainty assessment. OpenTURNS is one of them. OpenTURNS, for Open source Treatment of Uncertainties, Risk'N Statistics, is an open source tool developed, since 2005, by EDF R&D (Électricité De France), EADS Innovation Works (European Aeronautic Defense and Space) and Phimeca Engineering. It provides a large set of methods for the management of the uncertainty through its quantification, its propagation and its classification.

For the uncertainty quantification, OpenTUNRS provides a set of non-parametric and parametric probabilistic density functions such as the Kernel smoothing method, Normal distribution, Logistic distribution, Exponential distribution or the Log-normal distribution. It also provides a statistical toolbox with methods for estimating the density function parameter values.

For the uncertainty propagation, OpenTURNS provides methods for deterministic or probabilistic study. It proposes Min-max criterion in a deterministic framework and Approximation methods and Sampling method in a probabilistic framework (as Monte Carlo, FORM-SORM, Importance Sampling, Latin Hypercube Sampling, etc).

For the sensitivity analysis, OpenTURNS provides ranking methods for the classification of the uncertainty source with a probabilistic criterion.

OpenTURNS is developed with Linux and can be downloaded as a C++ library, as a Python module or as an autonomous application with a graphical user interface. A rich documentation is available on the website of the software [EcP05].

2.5.1.3   Uncertainty Modelling

Uncertainty quantification is an important phase in uncertainty management. The goal of this step is to determine a model for representing the uncertain parameters. The uncertainty modelling can be done by using two frameworks:

- Non-probabilistic framework,

- Probabilistic framework.

The choice of the framework to be used depends on the amount of information available about the source of uncertainty (database, expert judgement, etc).

Within the set of non-probabilistic framework, we can find:

- Interval Analysis method,

- Fuzzy set,

- Possibility theory,

- Dempter-Shafer theory of evidence.

More information on these methods can be found in [DP86].

In the probabilistic framework, the uncertainty is considered as a random variable X which is represented by a PDF or by a CDF. A PDF is defined only for continuous variables whereas discrete variables can be only represented with a CDF. In this framework, the objective of uncertainty modelling phase is to identify the relevant PDF to be used for representing the random variable X. In the literature, one can find this process named as *distribution fitting* [KD00] or *density estimation* [Sil86].

Three alternatives can be considered to identify this PDF [Ass08]:

- Non-parametric estimation,

- Semi-parametric estimation,

- Parametric estimation.

In the non-parametric approach, the PDF is built using the available data. In [Ass08], the authors state that this method "let the data speak for themselves". Less assumptions are taken on the a priori nature of the distribution of the observed data. Kernel estimation is an example of non-parametric approach. More information on non-parametric methods can be found in [Tsy08, HMSW04].

Semi-parametric approach is used for models which have both a parametric (a regression function) and a non-parametric part (the distribution of the error of the model) [Pow08]. In semi-parametric approach, as in non-parametric one, the *a priori* shape of the PDF is not assumed.

In the parametric estimation, the data are assumed to be coming from a specific density function. This density function has parameters stored in θ vector, which have to be determined. The choice of the density function to be used is not obvious: several ones should be tested in order to find the one which best fit the data. A bad choice can lead to a wrong representation of the real PDF, then to wrong conclusions. For instance, in aircraft design context, it can lead to time and money losses. In this study, we use the probabilistic framework and specifically the parametric estimation method. The probabilistic approach have been already used in the context of aircraft design [War98].

The parametric density function estimation is made in three main steps:

1. Computation of the sample characteristics: moments (expectation, standard deviation...), mode (value that appears more often in the sample), histogram (etc),

2. Selection of a family of parametric density functions $f_X(x, \theta)$ which seems to fit well the distribution (Gaussian, Exponential, etc),

3. Estimation of the parameters of the density function stored in the θ vector using methods such as method of moments, maximum likelihood method, method of probability-weighted moments [Sor80].

2.5.2 Robustness measure

Once the uncertainty has been quantified, it can be used to analyse the robustness of different solutions regarding this uncertainty and retain the more robust one. This robustness analysis relies on robustness measures.

There exist several robustness measures. The relevant measure to be used depends on the type of knowledge available. In [BS07], the authors make a review of some of them. Table 2.1 displays the list of robustness measures presented by the authors depending on the framework of the study and on the function the measure is applied on. Indeed, there exist two concepts: robustness related to the objective function and robustness related to the constraints satisfaction.

Framework	Robustness measure applied on:	
	Objective function	Constraints
Deterministic	Robust Counterpart Approach	
Probabilistic	Expectancy measures	Statistical feasibility
Possibilistic	Bootstrap method, Fuzzy Logic and Evidence theory	

Table 2.1: List of robustness measures presented by authors in [BS07]

In the following, we give a brief description of the deterministic and probabilistic measures of robustness. For that, we denote ϵ_x the uncertainty affecting the parameter x.

2.5.2.1 Robust counterpart approach

A. Ben-Tal and A. Nemirovski give the description of the Robust Counterpart approach in [BTN98]. It is based on the construction of a robust counterpart of the function $f(x)$. This counterpart denoted $R(x, \epsilon)$ is given by:

$$R(x, \epsilon_x) = \sup_{\delta \in \mathbb{V}(x, \epsilon_x)} f(\delta)$$

where $\mathbb{V}(x, \epsilon_x)$ is a neighbourhood of the design point x and its size is defined by the parameter ϵ_x. This measure adopts the philosophy of the worst case method because it seeks for determining the worst value of f within a neighbourhood of the design point. This method is used in deterministic framework (there is no need to know the distribution of the random variable) for measuring uncertainty affecting design parameters (denoted uncertainty of type B in [BS07]). This approach is also called "minimax approach" [Jur07].

The advantage of this measure is that the size of the optimisation problem does not exponentially grow with the number of uncertain parameters. The drawback is in the determination of the ϵ_x parameter value defining the size of the neighbourhood. A large neighbourhood can lead to an optimal solution but with poor performances. This kind of solution has to be avoided because it leads to a design which is without interest for the industries.

2.5.2.2 Probabilistic robustness measure

In probabilistic robustness measures, the uncertainties α or β described in figure 2.11 become random variables and follow a distribution. This distribution can be provided by the decision-maker and represents the decision-maker's knowledge on the uncertainty attached to the problem at this moment.

In [Bau12] the authors sort the probabilistic robustness measures in two groups: measures based on expectancy and ones based on percentile.

2.5.2.2.1 Expectancy Measures
The group of expectancy measures is composed of two types of measures:

- Measure based on expectation computation,

- Measure based on dispersion computation.

2.5.2.2.1.1 Expectation measure
The expectation measure considers the mean value of the objective function in a neighbourhood of the design point x being studied. The robustness measure is expressed as follows:

$$R_U(x, \epsilon_x) = E[U(f(x, \epsilon_x))]$$

where U is called utility function. Depending on the choice of U, one will obtain different robustness measures. Using $U(f) = sign(f)|f|$, one obtains:

$$R(x, \epsilon_x) = \int f(x + \beta) p_{\epsilon_x}(\beta) d\beta,$$

where p_{ϵ_x} is the probability density function of ϵ_x.

2.5.2.2.1.2 Dispersion measure
Another expectancy measure is the dispersion measure. It evaluates the variance of the objective function in order to quantify its amplitude of variation in the neighbourhood of a solution x. It is expressed as follows:

$$
\begin{aligned}
R(x, \epsilon_x) &= V[f(x + \epsilon_x)] \\
&= E\left[(f(x + \epsilon_x) - E[f(x + \epsilon_x)])^2 \right] \\
&= E\left[f^2(x + \epsilon_x) \right] - E^2[f(x + \epsilon_x)]
\end{aligned}
$$

where

$$E[f^2(x + \epsilon_x)] = \int (f(x + \beta))^2 p_{\epsilon_x}(\beta) d\beta.$$

The higher the variance is, the more the performance of the solution is degraded due to the effect of uncertainty.

As one can see, the objectives of expectation and dispersion measures of robustness are not the same: the expectation measure works on the performance and the dispersion measure on the variability of the performance. There are two methods which allow to optimise both objectives at the same time. The first method comes down to consider the Pareto front. The second one is the aggregation method. In this approach, the two objectives are aggregated in a single one using a weighted sum. Then, the robustness measure is:

$$R(x, \epsilon_x) = E(f(x, \epsilon_x)) + C_2 \sqrt{V[f(x, \epsilon_x)]}. \tag{2.11}$$

2.5.2.2.2 Percentile measures of robustness The second type of probability robustness measure is the percentile measure. It is used when one seeks for robust solutions which have a certain level of performance f. In this case, we directly consider the distribution of the output f.

The percentile measure of robustness of order k (e.g. 90% quantile) is expressed as follows:

$$R(x, \epsilon_x) = inf \{q \in \mathbb{R} : \mathbb{P}(f(x, \epsilon_x) \leq q) \geq k\} \tag{2.12}$$

where q and k are thresholds fixed by the user. The level of robustness is controlled with the parameter k. Equation (2.12) means that the probability that the objective value is lower than a threshold q is higher than k.

Percentile measures of robustness are widely used to evaluate the robustness of constraints g. In this case, it is called statistical feasibility robustness. The feasibility of the constraints is probabilistically guarantee as in equation 2.13.

$$R(x, \epsilon_x) = \mathbb{P}(g(x, \epsilon_x) \leq 0) \geq P_0 \tag{2.13}$$

where $\mathbb{P}(g(x, \epsilon_x) \leq 0)$ is known as the *probability of safety* and P_0 is the so called *level of confidence* or *level of reliability* representing the safety margin predefined by the decision-maker. P_0 can also be seen as the minimum allowable probability of safety. This way of formulating feasibility robustness referred to as Chance Constrained Programming (CCP) in the Stochastic Programming context [BL97, Liu09, Hen10]. Chance constrained programming has been introduced by Charnes and al. [CCS58], then was further expanded by Miller and al [MW65] and Prékopa [Pré70]. Optimisation problems integrating probabilistic constraints are referred to as Reliability-Based Design Optimisation (RBDO) in mechanical engineering.

2.5.3 Formulation of a robust optimisation

Once the uncertainty is identified, we have to formulate the robust optimisation problem we want to solve. There are several ways to formulate the robust problem depending on the robustness measure used. The choice for a suitable formulation is led by two points: the type of information available at the time the study is done (e.g. available information on the noise) and the decision-maker demand on the system performance (e.g. focus on performance only or on both performance and variability).

To transform a classical deterministic optimisation as in Equation (2.3) in a robust optimisation, we can use one of the following strategy [Bau12]:

- Replace each initial objective function of the problem by robustness measure,

- Keep initial objective functions and add one or more robustness measures in the problem as additional objectives,

- Add new constraints to the initial problem.

In the first strategy, each initial objective function of the problem is replaced by its robustness measure. As an example, one can minimise the expectation of the objective function:

$$\min \ E(f(x + \epsilon_x))$$

or minimise a percentile robustness measure such as the probability of failure:

$$\min \ \mathbb{P}(f(x + \epsilon_x) \leq q).$$

We can also use the aggregation method aforementioned to optimise performance and variability of the performance (see equation (2.11)):

$$\min \ C_1 E(f(x, \epsilon_x)) + C_2 V(f(x, \epsilon_x))$$

where C_1 and C_2 are positive constants fixed by the decision-maker (the result depends on the value set for them).

In the second strategy, the initial objective functions are kept and one or more robustness measures are added in the problem as additional objectives. As an example:

$$\min \ (f(x), R[f(x + \epsilon_x)]) .$$

The problem becomes a multi-objective optimisation and we have both objectives with no uncertainty and robust objectives. This strategy presents one drawback: the resolution of the problem can become tedious as the number of objective increases.

In the last strategy, new constraints are added to the initial optimisation problem. It allows to quantify the sensitivity threshold the decision-maker wants. As an example, one can minimise the expectation of the objective function with a constraint on the performance variability:

$$\min \quad E(f(x + \epsilon_x)) \qquad (2.14)$$
$$s.t. \quad \sqrt{V(f(x, \epsilon_x)} < \lambda,$$

where λ is a positive constant to be chosen. Equation (2.15) refers to the notion of M-robust defined by [LSW04].

2.5.4 Method to solve robust optimisation

When solving a robust optimisation problem, the first task is to compute the robustness measure used in its formulation. There exist several approaches to do so. They have been sorted in two groups in [BS07]:

- Methods that compute the robustness measure by using numerical techniques. These methods transform the robust problem into an ordinary one. They are referred to as deterministic approaches to robust optimisation.

- Methods that use directly the value of the objective and which do not need the first or second order derivative information. They are referred to as randomised approaches.

In this section, we give a short description of some of these methods.

2.5.4.1 Deterministic approaches to robust optimisation

In deterministic approaches to robust optimisation, the robustness measure is computed using numerical techniques. There are two methods: Feasibility robustness method and Expected Value Robustness and Related Measures method.

Feasibility robustness methods are used in the case where we are facing expectancy measure of robustness and statistical feasibility constraints. An example of approach with Constraints with Build-in Constraints Variation, First-Order Reliability Method (FORM) and Second-Order Reliability Method (SORM). Constraints with Built-in Constraints Variation approach is viewed as worst case analysis [REF]: it uses Taylor approximation for replacing the constraints (by taking into account the uncertainty). First-Order Reliability Method (FORM) and Second-Order

Reliability Method (SORM) are techniques which also use Taylor series. In the first case, the Taylor series approximating the probability $\mathbb{P}(g_i(x, \alpha) \leq 0)$ is truncated after the linear term and in the other one the Taylor series is truncated after the quadratic terms. More details can be found in [Rac01, EK05].

Expected Value Robustness and Related Measures methods are used to compute robustness measures based on expectancy and dispersion. They rely on approximation methods because it is almost always impossible to analytically compute these measures.

2.5.4.2 Randomised approach to robust optimisation

Randomised approaches differ from deterministic approaches to robust design in the sense that they directly introduce the uncertainty into the initial optimisation problem. The uncertainty considered is probabilistic uncertainty. There are several categories of approaches. In following, we present two of them: Monte Carlo and Meta-Model strategies.

2.5.4.2.1 Monte Carlo strategy Monte Carlo strategy, denoted MC further, refers to a family of methods using sampling for evaluating a quantity or for solving a system. They can be used to compute integrals, to solve partial differential equations, linear systems or optimisation problems. We present the general idea of the method hereafter.

Consider one wants to evaluate the following probability:

$$I = \mathbb{P}[f(X) \leq 0] = \int_{f(X) \leq 0} p_X(x) dx \tag{2.15}$$

where X is a random variable which have a probability density function p_X. We introduce the indicator function $\mathbf{1}_{f(X) \leq 0}$ such that:

$$\mathbf{1}_{f(X) \leq 0} = \left\{ \begin{array}{ll} 1 & \text{if } f(X) \leq 0, \\ 0 & \text{otherwise.} \end{array} \right.$$

Then equation (2.15) becomes:

$$I = E[\mathbf{1}_{f(X) \leq 0}]. \tag{2.16}$$

If we consider a set $\{x_1, \ldots, x_n\}$ of N independent samples of X, the probability I can be approximated by:

$$\hat{I} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}_{f(X) \leq 0}$$

The law of large numbers (denoted LLN, see [App13]) guarantees that $\hat{I}$ converges to I as the size N of the sample tends to infinity. Moreover, thanks to the Central Limit Theorem [Joh04], we have access to the Monte Carlo confidence interval of order α:

$$ConfInt = \left[\hat{I} - \alpha \left(\frac{\hat{I}(1 - \hat{I})}{N} \right)^{0.5}, \hat{I} + \alpha \left(\frac{\hat{I}(1 - \hat{I})}{N} \right)^{0.5} \right].$$

With this approach, it is also possible to estimate the mean and the variance of a random variable X for which the distribution p_X is not explicitly known. They are estimated as follows:

$$\begin{aligned} E[X] &\simeq \bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i, \\ V[X] &\simeq \frac{1}{n} \sum_{i=1}^{n} (x_i - \bar{x})^2. \end{aligned}$$

MC strategy is a method widely used to evaluate the robustness because of its simplicity and its generality: there is no need to put forward hypothesis on the output distribution (directly linked to the distribution of the inputs). Its main drawback is that it can be computationally expensive: it is necessary to compute a large number of evaluations of the function f in order to converge to an estimation which have a pertinent level of precision. This large number of evaluations can lead to high computation time specifically when the simulation of the function f is expensive. Indeed, much of today's engineering numerical simulations representing a system are complex simulations which are expensive to run. That is why, most of the time, we do not use the simulation of f directly but we use a meta-model. Nevertheless, even if the use of surrogate models allows reducing computation cost, MC method remains costly.

2.6 Meta-modelling method

The numerical tools used by engineers to simulate the behaviour of a system can be very complex and expensive to evaluate. Sometimes, a simple evaluation can take several hours or several days. It is avoided to use them directly in methods such as optimisation or uncertainty propagation. Indeed, these methods realise many evaluations of the analysis code which are time consuming.

An idea to reduce significantly the computation time is to replace the complex computer code by a much simpler model. When physics allows it, this model takes the form of a basic simplification of the computer code, otherwise it takes the form of an approximation, called meta-model, obtained by evaluating f at selected design points and interpolating or smoothing the function values obtained [JCS01].

The design of a meta-model involves the following three phases:

1. Elaborate an experimental design to generate a subset of design points. This can be done by using sampling methods such as Random sampling, Latin Hypercube Sampling (LHS), ...

2. Choose a family of models representative of a set of data (polynomials, neural networks, kriging, etc),

3. Fit the model on the subset of design points defined in step 1 by using method such as Least Square Regression or Weighted Least Square Regression.
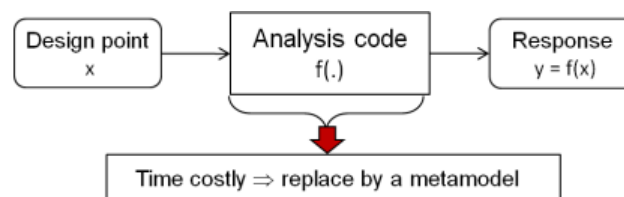
Figure 2.13: Idea of meta-modelling techniques

In this section, we present three of the most popular meta-modelling techniques which are:

- Polynomial regression models,

- Artificial neural networks,

- Kriging models.

2.6.1 Polynomial Regression Models

Polynomial Regression Models, commonly called Response Surface Model (RSM), are the most popular methods used to fit a sample of data because of its simplicity and its ease of implementation.

In RSM, the model is expressed as follows:

$$f(x) = P(x) + \epsilon$$

where

- $f(x)$ is the unknown function of interest,

- $P(x)$ is the known polynomial function of x,

- ϵ is a random vector with $\epsilon \sim \mathrm{N}(0,\sigma^2)$ (with $\sigma^2 \in \mathbb{R}$).

The polynomial models involved are generally low-order polynomial models such as linear or quadratic polynomials. Linear polynomials are used as approximation method when the surface to be approximated presents little curvature. In this case, the form of the model is:

$$P(x) = b_0 + \sum_{i=1}^{n} b_i x_i. \tag{2.17}$$

Quadratic polynomials are used when the surface presents significant curvature. The quadratic polynomial model form becomes:

$$P(x) = b_0 + \sum_{i=1}^{n} b_i x_i + \sum_{i=1}^{n} \sum_{j=i}^{n} b_{ij} x_i x_j. \tag{2.18}$$

The number of unknown coefficients in Equation (2.17) is $n+1$ and the number of unknown coefficients in Equation (2.18) is $\frac{(n+2)(n+1)}{2}$. Theses coefficients are evaluated using Least Squares Regression analysis to fit the meta-model to the existing set of data. Moreover, to fit the meta-model, the sample size should be at least two or three times the number of unknown coefficients [JDC03].

2.6.2 Artificial Neural Networks

The Artificial Neural Networks concept, further denoted ANN, is inspired from the functionality of the human brain behaviour where billions of neurons are interconnected by synapses. An ANN is composed of neurons (also called unit or single-unit perceptron), which are multiple nonlinear regression models which non-linearly transform their input signals to produce an output. ANN are well known because they present the advantage to be able to approximate any type of function and particularly, to successfully model abrupt changes in the function. They are also appreciated for their capacity to learn.

An ANN is defined by:

- its architecture (Feed-Forward or Feed-Back network, number of neurons, number of inputs, . . .),

- its learning process (e.g. weight estimation).

Feed-Forward ANN are networks that allow signals to go only in one way: from input to output. At the opposite, Feed-Back ANN (actually contained in recurrent neural network) are networks where the signals can go in both directions. It can produce complex networks.

Figure 2.14 presents the general architecture of an ANN. It is composed of an input layer, an hidden layer and an output layer. It can have several hidden layers containing intermediate variables. In this example, the ANN has 3 neurons in its input layer, one hidden layer with 4 neurons and one neuron in its output layer.
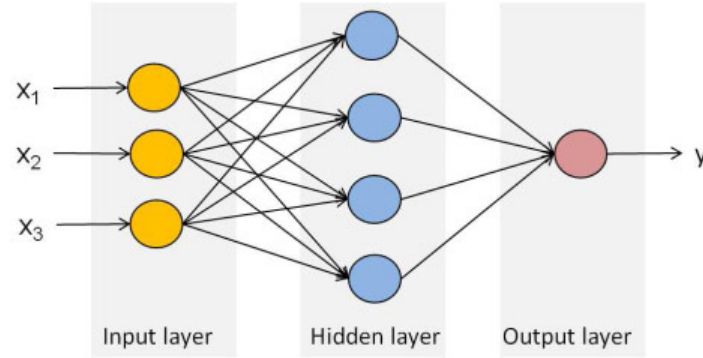
Figure 2.14: General architecture of ANN

Figure 2.15 gives more details on how works an ANN. In this figure, $(x_1, \ldots, x_n)$ are the inputs of each neuron contained in the input layer, $(w_1, \ldots, w_n)$ are weights affected to each input, f is an activation function and b is a bias.
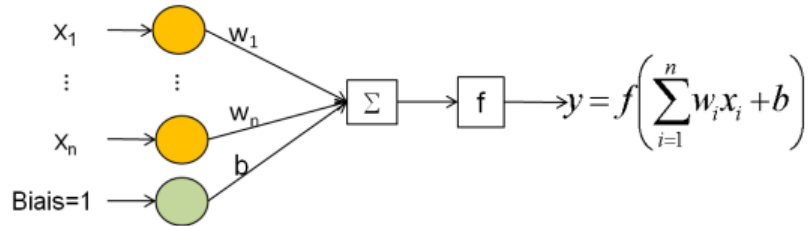
Figure 2.15: Details on the ANN working process

The neuron behaviour depends on the synaptic weights w and on the activation function f which aim is to transform non-linearly the inputs of the neurons. Activation functions most often used are Heaviside function, sigmoid function and linear function [REF] which are expressed as follows:

- Heaviside function:

$$f(x) = \begin{cases} 0 & \text{if } x < k \\ 1 & \text{if } x \geq k \end{cases}$$

  where $k \in \mathbb{R}$ is a fixed threshold.

- Sigmoid function:

$$f(x) = \frac{1}{1 + e^{-\lambda x}}$$

  which is a differentiable function where λ is a threshold which governs the slope of the function.

- Linear function:

$$f(x) = x.$$

Once the architecture of the ANN is fixed for the approximation of a particular system, that network has to be trained: it is fit to *training data* coming from the original complex problem by adjusting the value of the weights w_i (which are the degree of freedom of the ANN). The objective is to configure the network such that the application of a given set of inputs results in a desired output. This training process is called learning process. There are several learning processes: supervised, unsupervised and reinforcement learning.

The most popular ANN are:

- Multi-Layer Perceptron (MLP),

- Radial Basis Function (RBF),

- Support Vector Machine (SVM).

More details on these ANN can be found in [Bad05, BL00, VGS97, SS04].

2.6.3 Kriging Interpolation

The kriging interpolation is another approximation technique. It has been first introduced by D.G. Krige in the field of spatial statistics and geostatistics [Kri51, Cre93]. Kriging models are often used to approximate response data coming from deterministic simulations.

In Kriging interpolation, the unknown model is expressed as follows [LS08]:

$$f(x) = P(x) + Z(x) \tag{2.19}$$

where:

- $f(x)$ is the unknown function we look for,

- $P(x)$ is a known polynomial regression model,

- $Z(x)$ is the realisation of a stationary, normally distributed random Gaussian process with mean zero, variance $\sigma^2 \in \mathbb{R}$ and non-zero covariance matrix.

In equation (2.19), the term $P(x)$ represents the global trend of the process. Depending on the form considered for $P(x)$, the kriging process is said to be general, ordinary or simple. In the general case, also called universal kriging, the polynomial function $P(x)$ is a linear weighted combination of n known functions r_i, with weighted coefficients β_i: $P(x) = \sum_{i=1}^{n} \beta_i r_i(x)$. In the ordinary case, $P(x)$ is an unknown constant: $P(x) = p \in \mathbb{R}$. Finally, in the simple case, $P(x)$ is a known constant: as an example $P(x) = 0$.

The term $Z(x)$ creates "localised" deviations from the polynomial part $P(x)$ so that the kriging model interpolates the observations at the sampling points. As a result, the output of the kriging model at the sampling points is equal to the real observations. The construction of the interpolation is based on spatial covariance between points. The covariance is expressed as follows:

$$Cov(Z(x), Z(x')) = \sigma^2 R(x, x')$$

where R is the correlation function such that

$$R(x, x') = \prod_{k=1}^{n} R_k(x_k, x'_k).$$

The previous correlation function is specified by the user. He has the choice between a large set of correlation functions (see [SWMH89]). The more frequently used is the Gaussian correlation function:

$$R_k(|x_k - x'_k|) = e^{-\theta_k (x_k - x'_k)^2}$$

where $\theta \in \mathbb{R}^+$.

Kriging interpolation technique offers the advantage to provide, with the prediction of the function, an error of the prediction.

Chapter 3

Model Uncertainty Assessment in Preliminary Aircraft Design

Contents

One of the main sources of uncertainty detected in conceptual and preliminary aircraft design is coming from the model. It is often called model predictive uncertainty. It is part of the group of epistemic uncertainties. As aforementioned in the previous chapter, preliminary aircraft design relies, as many other engineering activities, on computational simulation models. In this study, the models are organized in the SiMCAD toolbox presented in chapter 1. At very early stage of aircraft design, predictive uncertainty has a larger amplitude than in any other following stages. The possibility that the final product reaches its requirements is linked to the know-how of the manufacturer which has not yet been acting in the new proposed development. This know-how would be available if we could involve very precise models and engineering skills but this is not possible in early design stages. In fact, at early design stages, manufacturer know-how can mostly be extracted from already built airplanes. Existing aircraft data base is clearly the most rigorous source of information where to look for a relation between physical dimensions and properties, as also for resulting performances obtained by the way of a specific know-how. As we have seen in chapter 1, semi-empirical models are an effective way to formalize these relations. Nevertheless, as only top level physical parameters of the product can be managed in early design stages, semi-empirical models reveals an important predictive uncertainty. Here again, the existing aircraft database can provide quantitative information about this uncertainty. This chapter is dedicated to the presentation of the methodologies used to assess the uncertainty in Preliminary aircraft design. We present in details the second step of the uncertainty assessment process described in chapter 1. In section one, we present the process used to identify the error of each SiMCAD models based on the comparison with experimental data stored in an aircraft database. Section 2 is devoted to the introduction of the framework used to quantify the predictive uncertainty. In these studies we use probability tools. The aim of section 3 is to present the reasons which have motivated the design of a new generic distribution function and to introduce its characteristics. Section 4 is focused on the particular case of nested models where uncertainty of sub-parts of the models cannot be addressed without being superimposed to uncertainty coming from other parts. In section 5, we perform a short correlation study in order to analyse the dependences existing between the errors of SiMCAD models. Finally, in section 6, an interpretation of the uncertainty is given in terms of freedom design.

3.1   SiMCAD models error identification

The error of the SiMCAD model is the discrepancy between the SiMCAD model prediction and the experimental values of the physical variable represented by the model ([ZGD95]). In [Lun05], the author used this method to identify the error of fire models. It is also the first step in model validation procedures ([RO10]). In this study, the experimental values come from an in-house aircraft database containing information about existing aircraft but also about future projects of aircraft which are sufficiently advanced (i.e. project for which we have enough data). We extract from this database a selection of the aircraft for which all important parameters are available. This leads to a specific database containing around 50 aircraft descriptions. The scope of data covers the main five important domains described previously: geometry, weight, aerodynamic, propulsion and performance.
If we use equation 2.10, the error of model f_i is given by:

$$\epsilon_i = y_{DB} - f_i(x), \tag{3.1}$$

where y_{DB} is the experimental value taken in the database. This function can be assessed according to a given aircraft j. Thus, the model error ϵ_{ij} of model f_i versus the experimental value taken in the description of aircraft j is:

$$\epsilon_{ij} = y_{DBj} - f_i(x_j), \tag{3.2}$$

provided that the model number i takes its input in the description of aircraft j.

The previous definition does not take into account intensive or extensive characteristics of the model defined in paragraph "Model Input Selection". Indeed, as explained before, extensive models have a non-linear scatter plot form and the error identification process should be representative of this phenomenon.The previous definition of error model is suitable only for intensive models. So, it is necessary to define another error definition for the extensive model case. The solution is to use relative error for extensive models. Finally, we consider:

$$\epsilon_{ij} \;=\; y_{DBj} - f_i(x_j) \text{ for intensive models, and} \tag{3.3}$$

$$\epsilon_{ij} \;=\; \frac{y_{DBj} - f_i(x_j)}{f_i(x_j)} \text{ for extensive models.} \tag{3.4}$$

The extensive and intensive characteristics also have to be taken into account during the propagation phase of the uncertainty. Thus, the error is applied to the model output as in 3.1. This way of doing is similar to the adjustment factor approach described in [Zio09].

$$f_i(x) \;=\; f_i(x) + \epsilon \text{ for intensive models,} \tag{3.5}$$

$$f_i(x) \;=\; f_i(x)(1 + \epsilon) \text{ for extensive models.} \tag{3.6}$$

By computing the error for the model f_i for each aircraft j of the database, we obtain a sample of error $(\epsilon_{i1}, \epsilon_{i2}, ..., \epsilon_{ij}, ..., \epsilon_{in})$. This sample can be represented as an histogram in order to observe its behaviour. The model error identification process is summarized in Figure 3.1. With
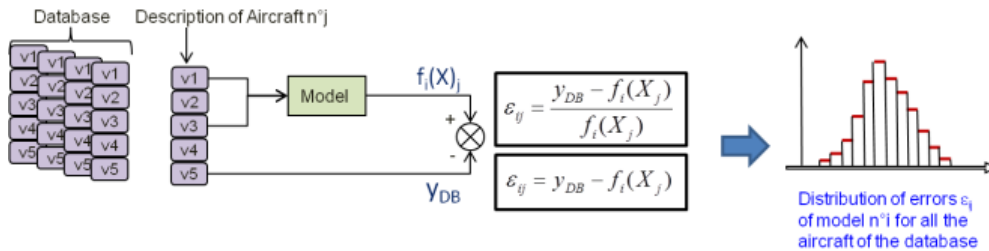
Figure 3.1: Identification of aircraft model error versus an aircraft database.

these histograms, we are able to proceed to a PDF distribution fitting. It is the aim of the next section: uncertainty quantification.

3.2   SiMCAD model uncertainty modelling

We first choose a probabilistic methods for representing the predictive uncertainty of each model. Then, we have to answer the following question: which density function best fits the histogram of each SiMCAD model error? We first launch a graphical analysis of each SiMCAD model error histogram. Then, we perform a study of normality.

3.2.1   Histogram of SiMCAD model error

A quick look at histograms of SiMCAD model errors leads to the main conclusion that the SiMCAD model errors are not all coming from a symmetrical probability distributions. It supposes that we use also asymmetrical density functions. Moreover, it also gives relevant information about the normality of errors. Up to now, noise model is assumed to follow a Normal distribution, but this assumption has never been checked (the normality assumption on the data is generally necessary to be able to compute some statistical tests). One of the characteristics of

the Normal distribution is to be symmetrical. Consequently, the previous remark leads straightforwardly to the conclusion that SiMCAD models which have an asymmetrical distribution of error are not following a Normal distribution.

Remark 3.2.1. Symmetry is necessary but not sufficient to conclude in normality. Indeed, other distributions such as Cauchy and Student distributions have a PDF of the same "shape" as the Normal distribution. Then, it is not straightforwardly possible to conclude in the normality of distribution of data which have a symmetrical distribution.

We check the normality property of the model errors. For this purpose, we use the Henry line method and launch the Anderson Darling test.

3.2.2 Result of the Normality tests for the SiMCAD model error

Figures 3.2 and 3.3 present examples of obtained Henry lines. They confirm our first conclusions. As we can see in figure 3.2, the hypothesis of normality seems acceptable for the error distribution of models like the one computing the Take-Off Field Length ($TOFL_1$ and $TOFL_2$) or the furnishing weight (*Mfurn*). Conversely, in figure 3.3, we clearly see that the hypothesis of normality is not appropriate for representing the behaviour of the SiMCAD model error.

It is important to notice that the previous method is a graphical analysis: consequently, it
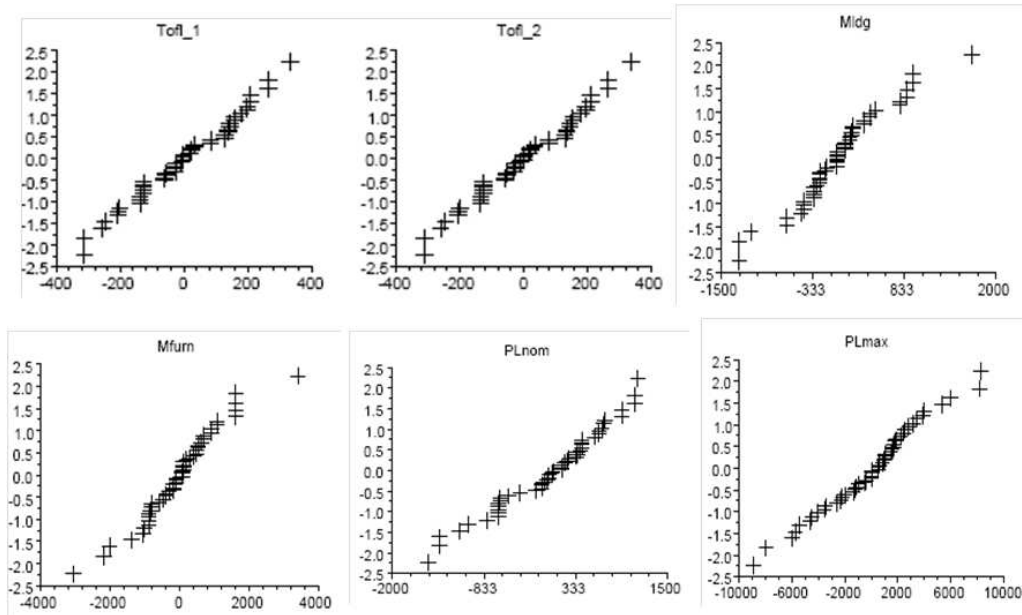
Figure 3.2: Example of SiMCAD error models which are normal: Henry line ok

remains fully subjective. Thus, more quantitative techniques (such as statistical tests) should be carried out to confirm conclusions obtained graphically.

3.2.3 SiMCAD: goodness-of-Fit test

In the previous subsection, we conclude that some of SiMCAD model errors were normally distributed. Thus, we have to find the adequate density function to use in order to fit the remaining SiMCAD model error.

OpenTURNS (cf. [EcP05]) provides a set of tools for checking the compatibility between a sample of data and a set of density functions. As for example: Chi-squared test (for discrete
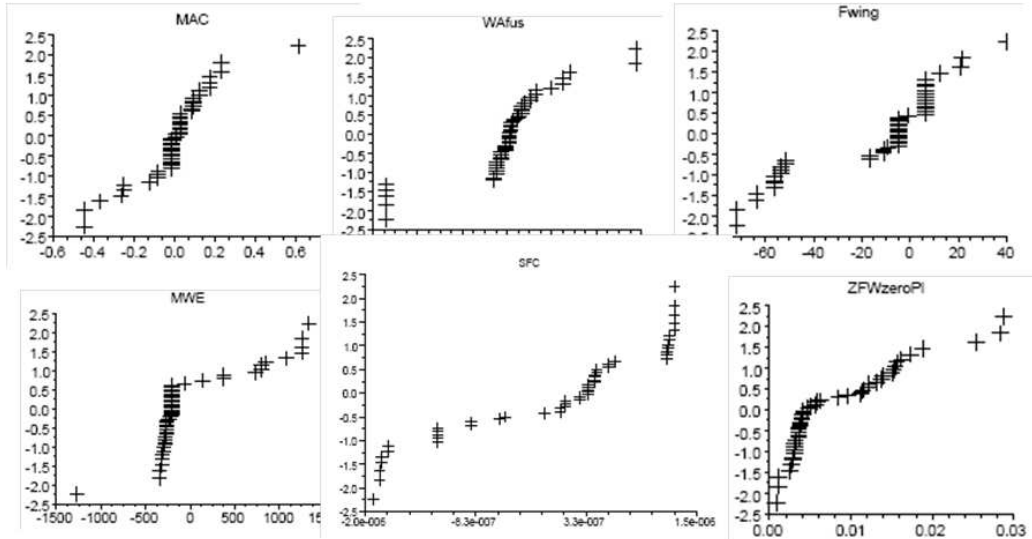
Figure 3.3: Example of SiMCAD error model which are not nomal

distributions), Kolomogorov-Smirnov test, Cramer-Von Mises test or Anderson Darling test. In this study, we use the Bayesian Information Criterion (BIC). The BIC is based on the maximum likelihood estimates of the model parameters. To use the BIC, we first have to select a list of candidate density functions. In the literature, we found a large set of probability distributions ([ME98]). By using the description of some of them, we identified a short list of possible candidates. These distributions can be classified into two groups: symmetrical and asymmetrical distributions.

In our study, two symmetrical distributions have been retained: the Uniform distribution and the Normal distribution (mentioned before). The Uniform distribution is retained to fit the SiMCAD model error distributions which have no particular form. Regarding asymmetrical distributions, the question is the same: how to select one distribution? This selection is discussed in [Ass08], [TNB02] and [CS97]. By using the expert judgment, we have retained the two following asymmetrical distributions: Gumbel and Gamma distributions.

3.3 Build-up of a generic density function

Among the density functions presented before, three of them do not have a compact support. Using this kind of distribution functions for representing the uncertainty of SiMCAD model errors could lead to obtain unphysical data (e.g.: a negative weight). Thus, we have decided to set up a new generic density function.

We look for a density function that could cover a wide range of possible shapes including uniform, triangular, normal distribution and possible variations of these distributions. Moreover, it should fulfil the following requirements:

- to have a compact support,

- to be unimodal,

- to be continuous,

- the position of its mode can be controlled,

- the shape of its mode (sharp or flat) can be controlled.

In addition to these 5 requirements, we add another one, more subjective: the behaviour of the shape of the law when its parameters vary should be as intuitive as possible. The aim of this last point is to make easy the use of the distribution in engineers' daily life. To a great extent, a density function with only four parameters are required to satisfy these requirements. Within the existing classic density functions, the four parameters Beta distribution (see [Bur99, NIS12]) appear to be the perfect candidate. It allows to cover a large range of shapes. Nevertheless, it does not satisfy the fourth and fifth requirements. Indeed, with the Beta distribution, the third and fourth parameters control independently the left and the right part of the curve. Moreover, the way the shape of the distribution is controlled by the parameters is not intuitive enough. Therefore, we have decided to build a new generic distribution based on the density function of the four parameters Beta distribution. Thus, we have set the following rules for the proposed generic distribution:

- The first parameter a represents the lower bound of the support;

- The second parameter b represents the upper bound of the support;

- The third parameter Z, by varying between -1 and +1, controls the symmetry of the density;

- The fourth parameter P will control the spreading in reference to the Normal distribution law, that is to say: the Normal law should be approximated by a rounded value of the parameter (for instance 1 or 10).

Hereafter, we present two versions of density functions that fulfil all the above requirements: the first one is built by assembling two polynomial functions and the second one is based on Beta distribution. Because of the ability of these laws to correctly mimic various distributions as the Uniform, Triangular, Normal or Gamma ones, we decided to name them respectively the **Mystique distribution** and the *Beta-Mystique distribution*, in reference to the character of Mystique in the X-Men movies.

3.3.1   Mystique distribution

The definition of the Mystique distribution is given hereafter: (3.7).

$$D(x,a,b,Z,P) = \begin{cases} \dfrac{2}{b-a}\left(\dfrac{1-K^2}{4}\left(\dfrac{1+X}{1+W}\right)^K + \dfrac{K(1+K)}{4}\left(\dfrac{1+X}{1+W}\right)^{K-1}\right) \\ \qquad\qquad\qquad\qquad\qquad\qquad \text{if } -1 \le X \le W \\[2ex] \dfrac{2}{b-a}\left(\dfrac{1-K^2}{4}\left(\dfrac{1-X}{1-W}\right)^K + \dfrac{K(1+K)}{4}\left(\dfrac{1-X}{1-W}\right)^{K-1}\right) \\ \qquad\qquad\qquad\qquad\qquad\qquad \text{if } W < X \le 1 \\[2ex] 0 \qquad\qquad\qquad\qquad\qquad\qquad \text{if } X < -1 \text{ or } 1 < X \end{cases} \tag{3.7}$$

where

$$\begin{cases} X &= x\left(\dfrac{2}{b-a}\right) - \left(\dfrac{b+a}{b-a}\right), \\ K &= 1 + 8.437 \cdot \max(\varepsilon, P) \text{ and } 0 \le P, \\ W &= \max\left(\varepsilon - 1, \min(Z, 1-\varepsilon)\right) \text{ and } -1 \le Z \le 1. \end{cases}$$

Here, ε is the smallest significant positive value on the used computer. The introduction of such a ε, as it appears in the formula, avoids computational troubles when the P and Z parameters reach their respective bounds.

The expression of the cumulative law is given in the equation (3.8) with the same notations as before.

$$C_1(x, a, b, Z, P) = \begin{cases} \dfrac{(1-K)(1+W)}{4}\left(\dfrac{1+x}{1+W}\right)^{K+1} + \dfrac{(1+K)(1+W)}{4}\left(\dfrac{1+x}{1+W}\right)^{K} \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{if } -1 \le X \le W, \\[1em] 1 - \dfrac{(1-K)(1-W)}{4}\left(\dfrac{1-x}{1-W}\right)^{K+1} - \dfrac{(1+K)(1-W)}{4}\left(\dfrac{1-x}{1-W}\right)^{K} \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{if } W < X \le 1, \\[1em] 0 \qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{if } X < -1 \text{ or } 1 < X. \end{cases}$$

(3.8)

If we denote C_2, C_3, C_4, C_5 the integrals of higher degrees of the distribution law, we can obtain the mean, variance, skewness and kurtosis coefficients as shown in equations (3.9), (3.10), (3.11) and (3.12).

$$\mu = b \cdot C_1(b) - C_2(b), \tag{3.9}$$
$$\sigma^2 = b^2 \cdot C_1(b) - 2b \cdot C_2(b) - 2 \cdot C_3(b) - \mu^2, \tag{3.10}$$
$$\begin{aligned} skew = {}& b^3 \cdot C_1(b) - 3b^2 \cdot C_2(b) + 6b \cdot C_3(b) - 6 \cdot C_4(b) - 3 \cdot \mu \cdot \sigma^2 \\ & -\mu^3, \end{aligned} \tag{3.11}$$
$$\begin{aligned} kurt = {}& b^4 \cdot C_1(b) - 4b^3 \cdot C_2(b) + 12b^2 \cdot C_3(b) - 24b \cdot C_4(b) + 24 \cdot C_5(b) \\ & -4 \cdot \mu \cdot skew - 6 \cdot \mu^2 \cdot \sigma^2 - \mu^4. \end{aligned} \tag{3.12}$$

Figure (3.4) shows the evolution of the shape of the Mystique distribution law when P varies between 0 and 5, with $Z = 0$.

Figure 3.4: Evolution of the shape of the distribution law ($0.1 \le P \le 1$ and $-1 \le Z \le 0$)

Figure 3.5 illustrates the evolution of the shape for P varying between 0.1 and 1 and Z varying between -1 and 0 (the behaviour is symmetrical when 0<Z<1).

The ability of our flexible distribution to mimic Normal, GumbleMax, Uniform or Triangular laws is illustrated in the graphs from figure 3.6.

Observe that the Normal distribution $N(0, \sigma)$ is approximated by $Myst(-6\sigma, 6\sigma, 0, 1)$ and the Uniform $U(min, max)$ is approximated with $Myst(min, max, 0, 0)$. The Mystique law has

Figure 3.5: Evolution of the shape of the distribution law ($0.1 \leq P \leq 1$ and $-1 \leq Z \leq 0$)
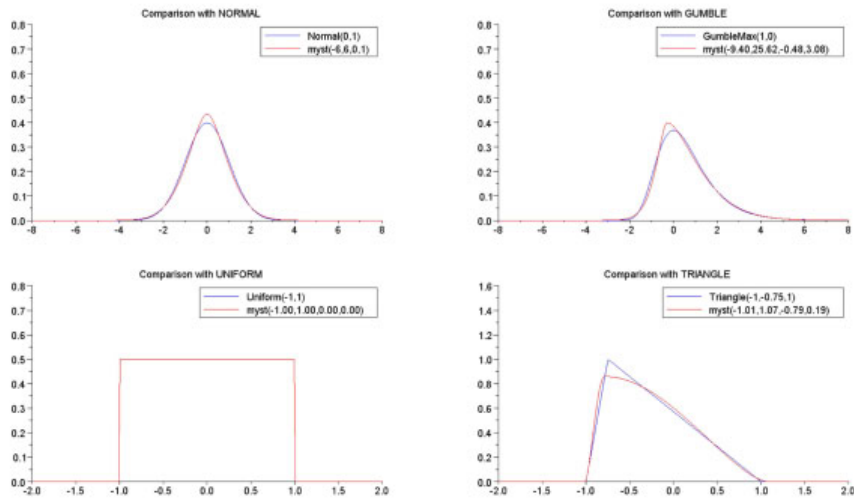
Figure 3.6: Illustration of the versatility of the Mystique law

some difficulties to mimic the shapes of the peaks of Normal, Gumble or Triangular laws. This drawback is acceptable if one is mainly interested in the distribution tails where the Mystique law matches quite well the reference distributions.

3.3.2   Beta distribution

Among the set of classical laws, there is one which has the same characteristics as the Mystique distribution: the Beta distribution. The Beta distribution has a compact support and it can take a wide variety of forms depending on the values of its parameters. The characteristics of Beta distributions are:

- Parameters: α and β (both shape parameters)

- PDF:

$$
\begin{aligned}
f(x) &= \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1}(1-x)^{\beta-1} \\
&= \frac{1}{\mathbb{B}(\alpha,\beta)} x^{\alpha-1}(1-x)^{\beta-1},
\end{aligned}
\tag{3.13}
$$

 where α is the shape parameter, β is the scale parameter, $\Gamma(z)$ the gamma function and $\mathbb{B}(\theta)$ is the Beta function defined by

$$
\beta(x,y) = \int_0^1 t^{x-1}(1-t)^{y-1} dt
$$

 .

- CDF:

$$
\begin{aligned}
F(x) &= \frac{\mathbb{B}_x(\alpha,\beta)}{\mathbb{B}(\alpha,\beta)}. \\
&= \mathbb{I}_x(\alpha,\beta)
\end{aligned}
\tag{3.14}
$$

 where $\mathbb{B}_x(\alpha,\beta)$ is the incomplete beta function and $\mathbb{I}_x(\alpha,\beta)$ is the regularized incomplete beta function.

- Mean: $\frac{\alpha}{\alpha+\beta}$

- Standard Deviation: $\sqrt{\frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}}$.

The estimation of the beta distribution parameters with the method of moments gives:

$$
\alpha = \frac{E[X]^2 - E[X]^3 - V[X]}{V[X]},
$$

$$
\beta = (1 - E[X]) \left(\frac{E[X](1 - E[X])}{V[X]} - 1 \right).
$$

The previous expression was for a Beta distribution defined on the interval [0,1]. The use of Beta distributions can be extended (generalised) to any interval $[a,b]$ by performing a change of variable. Indeed, if X follows a Beta distribution defined on [0,1], then $Y = a + X(b-a)$ follows a Beta distribution defined on $[a,b]$. This leads to a four parameter Beta distribution which is expressed as following:

- Parameters: a, b, α and β

- PDF:

$$
f(x) = \begin{cases} \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)(b-a)} \left(\frac{x-a}{b-a}\right)^{\alpha-1} \left(1 - \frac{x-a}{b-a}\right)^{\beta-1}, & a \le x \le b \\ 0, & otherwise \end{cases}
$$

- Mean: $E[X] = a + (b-a)\frac{\alpha}{\alpha+\beta}$

- Standard deviation: $V[X] = \sqrt{(b-a)^2 \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}}$.

Remark 3.3.1. Outside its range of validity, meaning for $x \notin [a,b]$, the Beta distribution is not null. Thus, the user has to be careful when using the Beta distribution and has to stay in its domain of definition.

This Beta distribution is also able to mimic various forms. The first two parameters define the support, while the last two parameters drive independently the left part of the curve and the right part of the curve. This behaviour was not found intuitive enough by our collaborators. For that reason, we introduced a specific transformation in order to satisfy our initial requirements. That leads to the definition of a new distribution, the Beta-Mystique distribution.

3.3.3 Beta-Mystique distribution

The expression of the Beta-Mystique distribution is given in equation (3.15).

$$D(x,a,b,Z,P) = \begin{cases} \dfrac{(b-x)^{q_1-1}(x-a)^{p_1-1}}{\beta(p_1,q_1)(b-a)^{p_1+q_1-1}} & \text{if} \quad a \leq x \leq b \\ \\ 0 & \text{if} \quad x < a \quad or \quad b < x \end{cases} \tag{3.15}$$

$$\text{where} \quad \begin{cases} W = \max(\varepsilon-1, \min(Z, 1-\varepsilon)) \quad \text{and} \quad -1 \leq Z \leq 1, \\ K = \dfrac{3.3\max(\varepsilon,P)}{b-a} \quad \text{and} \quad 0 \leq P, \\ M = \dfrac{1}{2}(a.(1-W)+b.(1+W)), \\ p_1 = 1 + K(M-a), \\ q_1 = 1 + K(b-M), \\ \beta(p_1,q_1) = \dfrac{\Gamma(p_1)\Gamma(q_1)}{\Gamma(p_1+q_1)}, \quad \Gamma \text{ being the Gamma Law.} \end{cases}$$

Here again, ε is the smallest significant positive value on the used computer. Introducing such an ε, as shown in the formula above, avoids computational problems when the P and Z parameters reach their respective bounds.

In the case of the Beta-Mystique law, the expressions of Mean, Variance, Fisher Skewness and Kurtosis Coefficients [JKB95] are simple to obtain as shown in equations (3.21), (3.22),

(3.23) and (3.24).

$$Z = max(-1 + \epsilon, min(Z, 1 - \epsilon)), \tag{3.16}$$

$$m = \frac{a(1 - Z) + b(1 + Z)}{2}, \tag{3.17}$$

$$r = \frac{3.3 \times P}{b - a}, \tag{3.18}$$

$$p1 = 1 + r(m - a), \tag{3.19}$$

$$q1 = 1 + r(b - m), \tag{3.20}$$

$$Mean = \frac{p_1 b + q_1 a}{p_1 + q_1}, \tag{3.21}$$

$$Var = \frac{p_1 q_1 (b - a)^2}{(p_1 + q_1 + 1)(p_1 + q_1)^2}, \tag{3.22}$$

$$Skew = \frac{2(q_1 - p_1)\sqrt{p_1 + q_1 + 1}}{(p_1 + q_1 + 2)\sqrt{p_1 q_1}}, \tag{3.23}$$

$$Kurt = 6\frac{(p_1 + q_1 + 1)(q_1 - p_1)^2 - p_1 q_1 (p_1 + q_1 + 2)}{p_1 q_1 (p_1 + q_1 + 2)(p_1 + q_1 + 3)}. \tag{3.24}$$

The graphs from figure 3.7 show the evolution of the shape of the distribution law when P varies between 0 and 10 and Z equals 0.
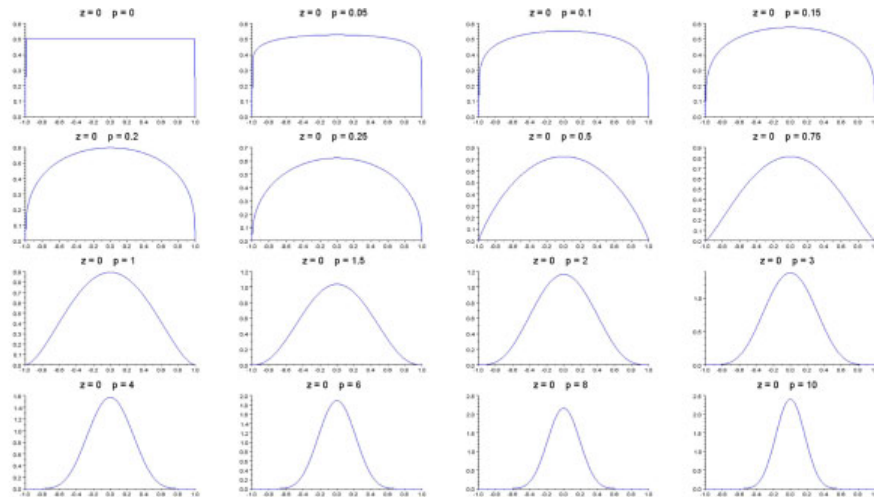
Figure 3.7: Evolution of the shape of the distribution law ($0 \leq P \leq 10$ and $Z = 0$)

The graphs from figure 3.8 illustrate the evolution of the shapes for P varying between 0.5 and 4 and Z varying between -1 and 0 (the behaviour is symmetrical when $0 < Z < 1$).

The ability of the law to mimic the Normal, GumbleMax, Uniform or Triangular laws is illustrated on the graphs from figure 3.9. Note that the Normal Law $N(0, \sigma)$ is approximated by $BetaMyst(-6\sigma, 6\sigma, 0, 10)$ and the Uniform $U(min, max)$ is obtained with $BetaMyst(min, max, 0, 0)$. The Beta-Mystique distribution seems to perform better than the Mystique one for capturing the shapes of the pick of Normal, Gumble laws, but it turns out not to be very accurate for the Triangular one.
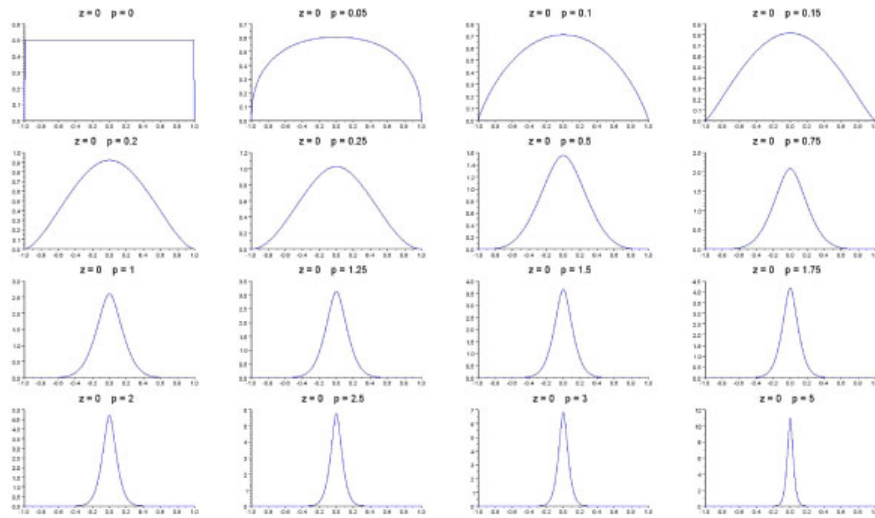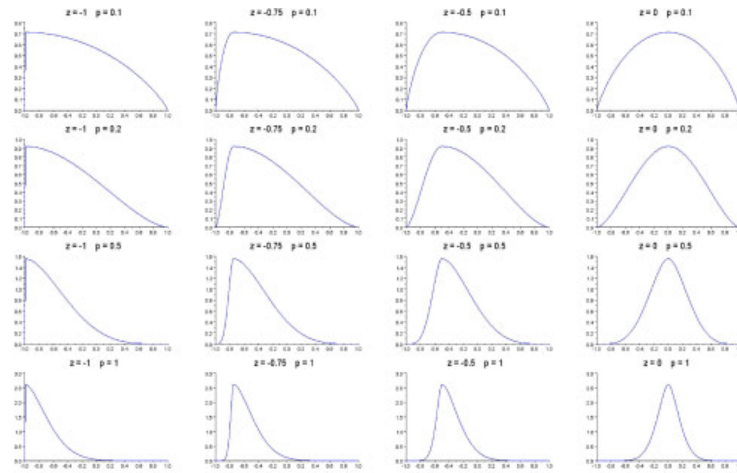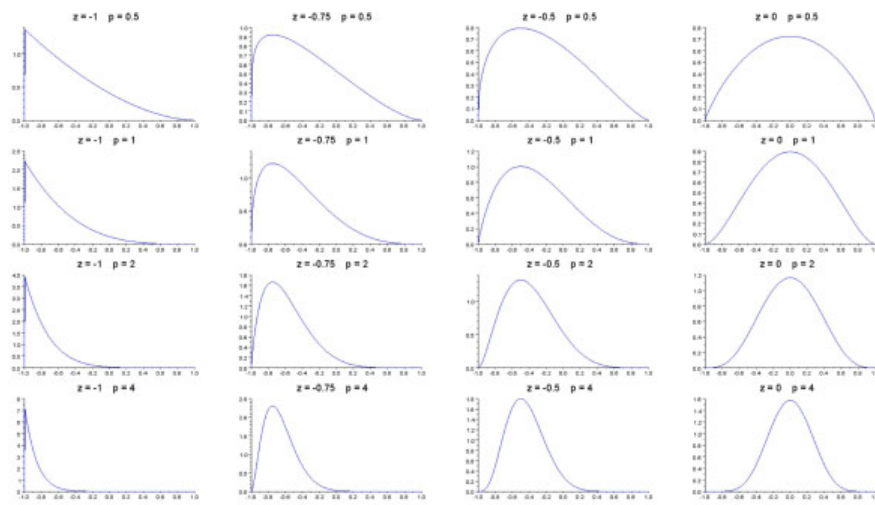
Figure 3.8: Evolution of the shape of the distribution law ($0.5 \leq P \leq 4$ and $-1 \leq Z \leq 0$)
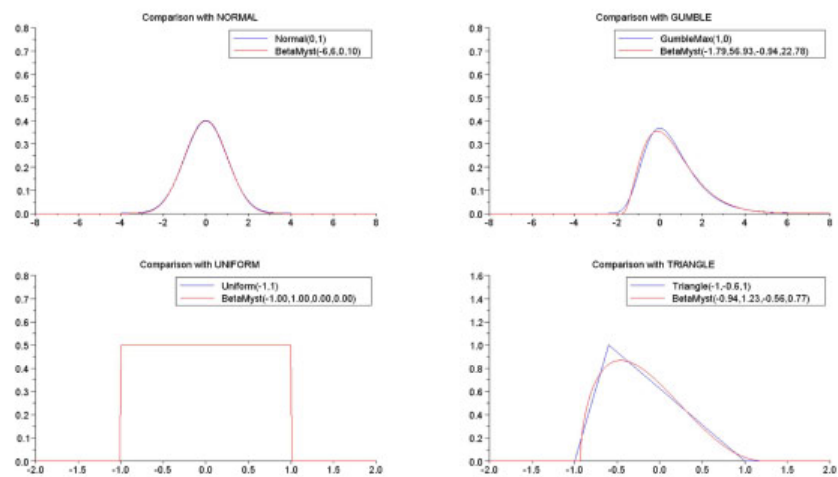
Figure 3.9: Illustration of the flexibility of the Beta-Mystique law

In both cases (Mystique and Beta-Mystique distribution), the best values of parameters that allow matching the different distributions (Gumble, Triangular and Normal) have been obtained by solving an optimization problem with the parameters (to be chosen) as unknowns and the likelihood as the criterion to be maximized.

3.3.4   Results of SiMCAD predictive uncertainty quantification

The uncertainty on the error of each aircraft model within the five domains (geometry, aerodynamics, mass, propulsion and performance) has been quantified using the Beta-Mystique distribution. Figure 3.10 displays examples of this quantification on three aircraft models corresponding to the aircraft aerodynamics. The values of the Beta-Mystique parameters are stored in table 3.1.
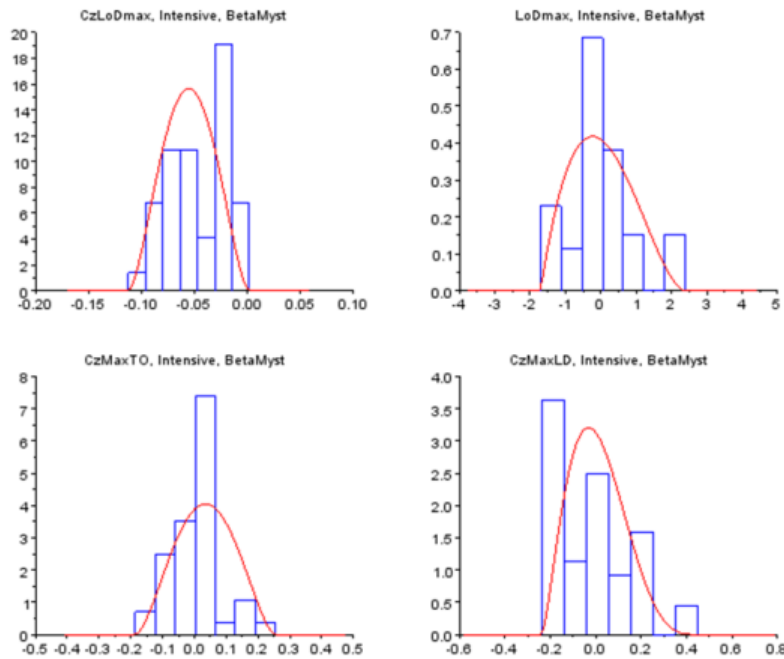
Figure 3.10: Uncertainty quantification of our aerodynamic models using Beta-Mystique law

| | Beta-Mystisque parameter | | | |
Model name	a	b	Z	P
LoDmax	-1.7020	2.3807	-0.28	0.79
CzMaxTO	-0.1864	0.2548	0	1
CzMaxLD	-0.2376	0.4481	-0.4	1.44

Table 3.1: Beta-Mystique parameter values

A graphical editor of the Beta-Mystique distribution (see figure 3.11) has been developed so that experts could easily define what they have in mind concerning uncertainty of some given input parameters. It has been proved user friendly by users such as Airbus engineers.

The expert can define graphically the shape of the distribution or enters directly the parameters of the Beta-Mystique law. Note that the support of the distribution is defined by deltas *vs* a so called *nominal value* which can be different from the most probable value (the peak) since it happens quite often that the nominal value, being altered by margins, is different from the final
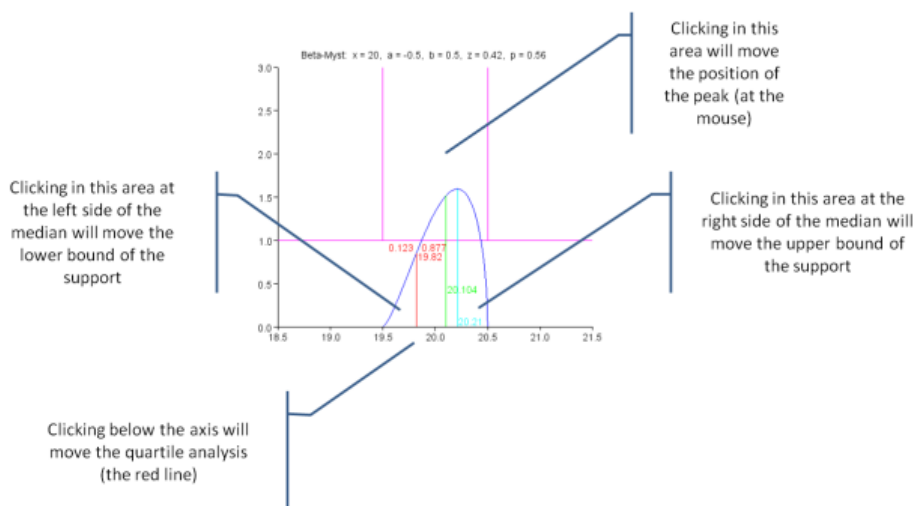
Figure 3.11: Graphical editor of the Beta-Mystique distribution

most probable value.Once the Beta-Mystique parameters have been captured, the information is translated into moments (using formulas (3.16) to (3.24)).

3.4 Particular case of performance models error

In this section we focus on the quantification of the uncertainty on the error of models computing performance of the aircraft. The method used in section 3.1 for evaluating the model error is not applicable in the case of the performance models because of their structure.

3.4.1 Problem definition

Not all aircraft models do have the same structure. We list two types of structures: simple structure and interlinked structure. We say that an aircraft model has a simple structure if it does not use sub-models for which the uncertainty can be measured independently. It can be represented as in figure 3.12. This is typically the case for aerodynamic and engine models. Conversely, we say that an aircraft model has an interlinked structure if it uses other models for its calculation. These sub-models have both inputs coming from outside of the main model and inputs that are strictly internal. It can be represented as in figure 3.13. Input 1 and input 2 are coming from outside and i3 and i4 are strictly internal inputs. All models computing the performance of the aircraft have an interlinked architecture. Indeed, they need aerodynamic and engine data for their calculation.
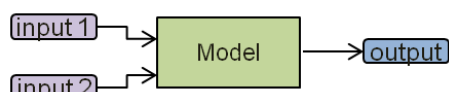
Figure 3.12: Simple structure
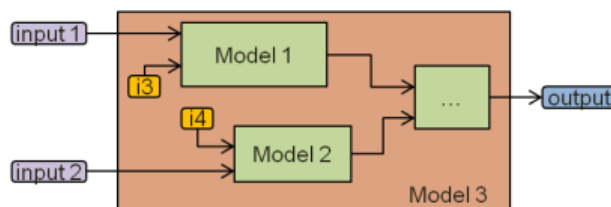
Figure 3.13: Interlinked structure

The interlinked structure of these models leads to an inability to apply the method presented in section 3.1 to identify their errors. As aforementioned, the error of each model is identified by confronting the output value of the model with the experimental value available in the aircraft

description. Doing so, we identify an error distribution for both types of models (fig. 3.14). Nevertheless, because of the interlinked structure of performance models, the error distribution
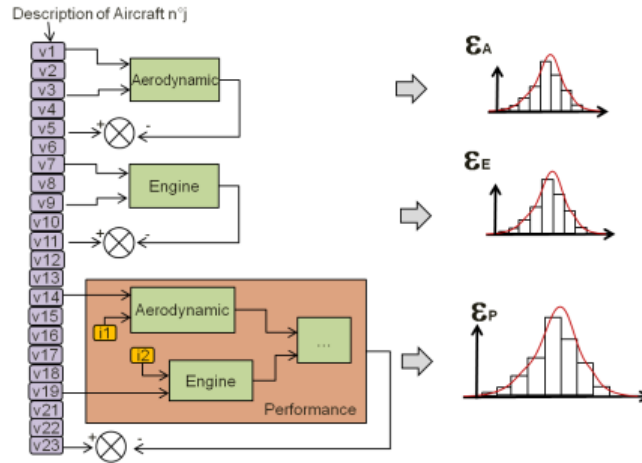
Figure 3.14: Error identification

identified for them implicitly contains errors of model coming from aerodynamic and engine models. Then, if we want to quantify the intrinsic error of performance model by not taking into account uncertainty coming from nested models, it is necessary to develop a strategy.

3.4.2   Methodology

The strategy used is based on an optimisation process and is split into four steps.

1. Generation of three samples of errors ϵ_A, ϵ_E and ϵ_P of size n respectively for aerodynamic, engine and performance models from their corresponding Beta-Mystique distribution previously identified: $\epsilon_A = (\epsilon_{A1}, \ldots, \epsilon_{An})$, $\epsilon_E = (\epsilon_{E1}, \ldots, \epsilon_{En})$, $\epsilon_P = (\epsilon_{P1}, \ldots, \epsilon_{Pn})$ (the same three samples are used during the whole process in order to avoid sampling noises).

2. Building of the reference CDF of performance output. In this step, we first launch a performance calculation without propagating any uncertainty. This gives a single value for performance model. Then, we add noise to this single value with error values contained in ϵ_P. At the end of this step we obtain a sample of values for the performance model. We represent this sample through an empirical CDF. This step is described in Figure 3.15.
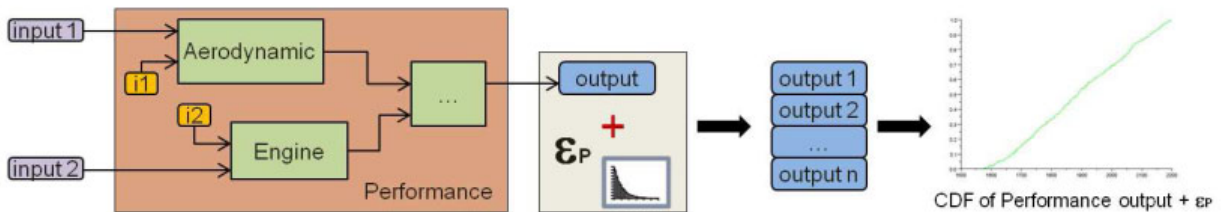
Figure 3.15: Process to build the reference CDF

3. Generation of a sample of errors ϵ_I (size = n) from a Beta-Mystique distribution with an arbitrary parameter value $\theta = (a, b, Z, P)$ for representing the intrinsic uncertainty on errors of performance model. We build this sample thanks to the CDF inversion method previously described (cf. theorem A.2.1). We first generate random variate according to the Uniform(0,1) distribution and then transform them in order to obtain random variate

of the Beta-Mystique distribution. The same random variate of the Uniform(0,1) will be used all along the process in order to avoid sampling noise.

4. Building of the parametric CDF and launching an optimisation process on θ (which is directly linked to the contents of ϵ_I) in order to minimize the distance between the reference CDF and the parametric CDF. In this step, we launch another performance calculation but, this time, by propagating errors stored in ϵ_A, ϵ_E and also ϵ_I. That leads to a sample of values for the performance model. It is described in Figure 3.16. The objective is then
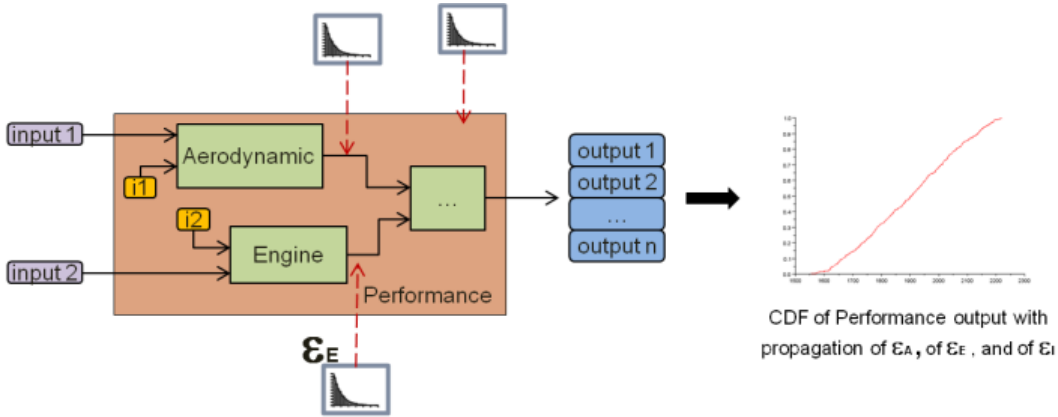
Figure 3.16: Process to build the parametric CDF

to find the value of θ that minimises the distance between the reference CDF built in step 2 and the parametric one. The distance between the two CDF is expressed as in equation 3.25 (see figure 3.17).

$$D = \sum_i \left(F_i(\epsilon_P) - \tilde{F}_i(\theta, \epsilon_A, \epsilon_E, \epsilon_U) \right).$$

(3.25)
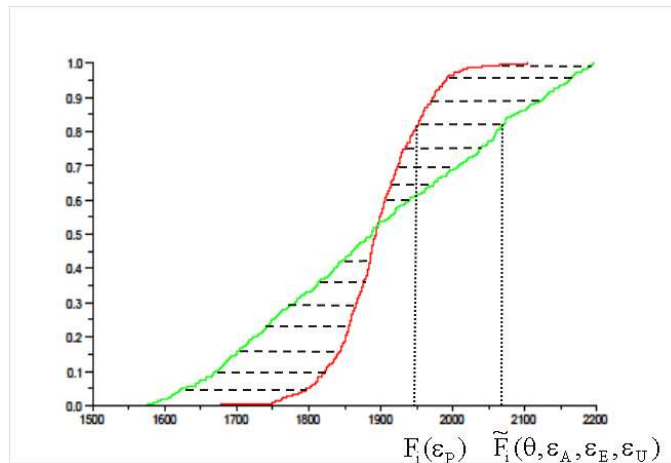
Figure 3.17: Distance between two CDF

The solution of this optimisation process is denoted by θ_{opt} and is the point that produces a $\tilde{F}_i(\theta, \epsilon_A, \epsilon_E, \epsilon_U)$ close of the reference CDF $F_i(\epsilon_P)$. The intrinsic uncertainty ϵ_I of performance model is then represented by a Beta-Mystique distribution with a parameter value fixed as θ_{opt}.

We used two mehods to minimise D: a classical optimisation method and the maximum likelihood method aforementioned (cf. section A.2.3.2). Moreover, the convergence of the optimisation is not easy because its depends on the starting point θ_0: the more θ_0 is far from the starting point, the more difficult the convergence is to access. Some strategies can be put in place in order to select the starting point (cf. [GS10]). Within the two methods, the classical optimisation method was the most robust regarding to the starting point θ_0.

3.4.3 Results

Figures 3.18 and 3.19 display the comparison betwen the reference (green curve) and the parametric CDF (red curve) built with the result θ_{opt} of the optimisation.
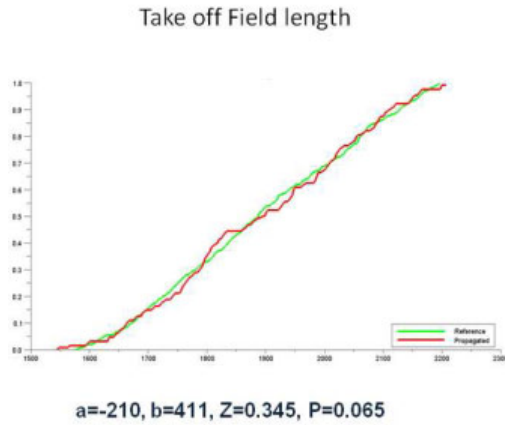
Figure 3.18: Parametric CDF obtained for the Take-Off Field Length model
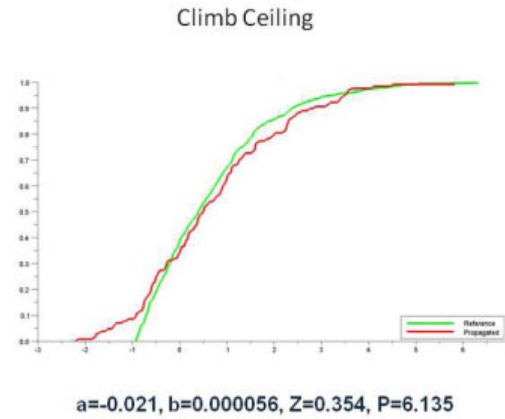
Figure 3.19: Parametric CDF obtained for the Climb ceiling model

In example 1 in figure 3.18, the result obtained is correct: the difference between the two CDF is minimised and the parametric CDF is "uniformly" close to the reference curve. In the second example (see figure 3.18), the result is not so good. There is some area where the red curve is far from the reference CDF (in the neighborhood of f).

3.5 Correlation study

The study carried out is based on the hypothesis of a complete independence of model errors as measured in reference to the airplane data base. In order to check the validity of this assumption, we performed a correlation study. For this purpose, we have computed the correlation coefficients $r(X, Y)$ (also referred as Pearson's coefficient in the literature) which informs on the existence of a linear relation between two sets of data (or two random variables). In our case, X and Y are two sets of aircraft model errors. The correlation coefficient is given by Equation (3.26):

$$r(X, Y) = \frac{E[(X - \mu_X)(Y - \mu_Y)]}{\sigma_X \sigma_Y},$$ (3.26)

where $E[(X - \mu_X)(Y - \mu_Y)]$ is the covariance between X and Y commonly denoted $cov(X, Y)$. The correlation coefficient $r(X, Y)$ lies always in [-1, 1] and it is equal to:

- 0 if there is no linear relation between the samples;

- 1 if there is a maximal positive linear relation, and it is in the range [0,1] if there is a simple positive linear dependence;

- -1 if there is a maximal negative linear relation, and it is in the range [-1, 0] if there is a simple negative linear relation.

The results highlights a few correlations. Table 3.2 displays some of those which have a correlation coefficient higher than 0.90 in absolute value. Figure 3.20 illustrates the positive correlation between error of *FNslst* and *MTOW* models.

Model name	Correlated with	Correlation coefficient
FNslst	MTOW	0.93
ToCk	ToCt	0.93
MCL	MCR	0.97
ClbCeil	CrzCeil	0.94
ZFWplMax	FuelMaxFuel	-0.99
FuelplNom	ZFWplNom	-1

Table 3.2: Correlation coefficient $|r| > 0.90$
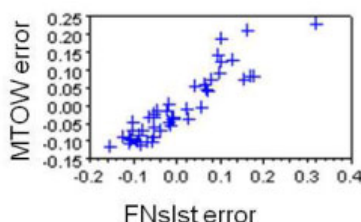
Figure 3.20: Correlation between samples of 2 model errors

We could distinguish three different categories:

- A correlation appears between variables that are linked by a deterministic relation, this is the case for instance for *FuelPlNom* and *ZFWplNom*. Correlations of this category are not a problem because the uncertainty is attached to only one variable while the other one is computed without uncertainty using the deterministic relation.

- A correlation appears between variables of the same nature, which values are resulting from a unique sub-model. This is the case for instance for *ToCk* and *ToCt*, *MCL* and *MCR* or *ClbCeil* and *CrzCeil*. In this case, the uncertainty measured is attached to the sub-model (see figure 3.21).
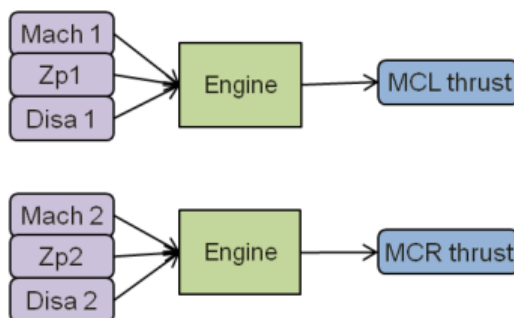
Figure 3.21: Same model used to compute 3 parameters related to the engine

- A correlation appears because of a strong physical link between the variables which are not taken into account as a model but as a process involving the models. This is the case for instance for *FNslst* and *MTOW* which are linked by overall optimisation of the aircraft submitted to the necessity to fulfill operational requirements. In this case, it is important not to use both models in the same process. For example, if the *MTOW* is assessed using the corresponding model and its uncertainty, then *FNslst* must be computed according to operational requirements satisfaction without introducing any other uncertainty.

3.6 Design freedom

Figure 3.22 gives a graphical explanation of the model error distribution according to a given set of experiments (*i.e.*, the content of the aircraft database).
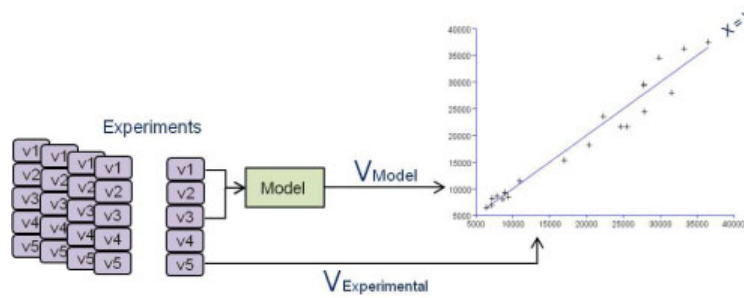
Figure 3.22: Aircraft model error distribution.

For a given output value of the model f_i, the range in brackets in Figure 3.23 contains the possible variability ϵ_{f_i} of the real output that would result from a real design process.
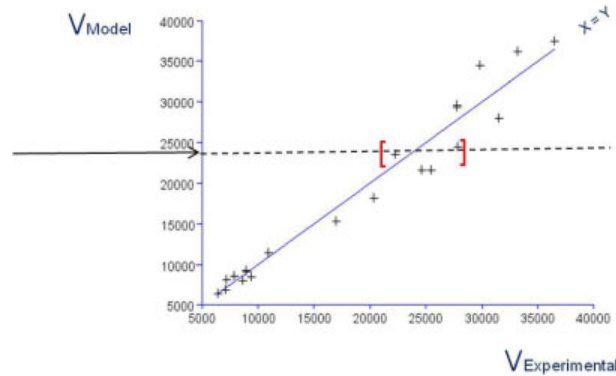
Figure 3.23: Visualisation of the possible variability of the real output.

$$y_{true} = f_i(x_n, ..., x_p) + \epsilon_{f_i},$$

ϵ_{f_i} is the predictive uncertainty of the model f_i at a very early design stage, let's say for $t = t1$. Some months later, at $t = t2$, if the design process goes on well, the knowledge acquired on the studied aircraft configuration has increased and we can assume the existence of a refined version of model f_i (which probably requires new input parameters). Similarly, at the end of the design process, let's say $t = t3$, we assume the existence of the most refined version of model f_i denoted F_i giving the "best" (i.e. the most accurate) value of variable y taking as input the detailed design parameters z_q. The remaining uncertainty ϵ_{F_i} will be drastically reduced in comparison

with ϵ_{f_i}. Let us note E_i the difference $F_i - f_i$, then we have:

$$y_{true} = f_i(x_n, ..., x_p) + E_i(x_n, ..., x_p, ..., z_q, ...) + \epsilon_{F_i} \text{ with } \epsilon_{F_i} << \epsilon_{f_i}.$$

Finally:

$$\epsilon_{f_i} = E_i(x_n, ..., x_p, ..., z_q, ...) + \epsilon_{F_i}. \qquad (3.27)$$

Formula (3.27) is developing the true value of predictive error of ϵ_{f_i} but a new question is raised: are we sure that such a function F_i exists?

Actually, we can find an indication (but not a proof) of the existence of the function F_i in the two following remarks:

- According to most of the recent aircraft developments that have been carried to their conclusion, final aircraft performances have been quite well predicted by last versions of models.

- All the points in Figure 3.22 correspond to aircraft which development have been carried to the end.

If we assume the existence of the function F_i (which is not a so strong assumption in fact), then for most of the quantities y_{true} (corresponding to continuous physical phenomena) we can deduce that it exists particular set of values for the z_q parameters that allow $F_i(z_q) + \epsilon_{F_i}$ to reach any value within the range of ϵ_{f_i}. In other words, for a given set of values $x_n, ..., x_p$, it exists a set of detailed parameters z_q that allows F_i to reach most values within the range of uncertainty of $f_i + \epsilon_{f_i}$. Trying to reformulate again, we are just saying that uncertainty on ϵ_{f_i} could be a measurement of the freedom given by further detailed design.

This last interpretation is particularly interesting because it suggests a kind of equivalence between the amplitude of predictive uncertainty at a given design process stage and the amplitude of freedom given by effective further design work. But this does not mean that any deviation from the nominal design within the range of ϵ_{f_i} can be reached with the same level of difficulty. The opposite is most probably true, as it is well known that it is much more difficult to make a structural part lighter than to make it heavier for a given set of load cases...

Practically, it is quite easy to guess, for each y_i, which side is the most difficult to reach (see Figure 3.24). This side can be used to orientate the measured distribution of errors so that the position of the peak corresponds naturally to the common state of the art: low values of quantile correspond to deviations that are not challenging while high values of quantile correspond to deviations that are very challenging according to the current state of the art (but not unreachable and even which can be overtaken).

This information can be enriched by expert judgment in order to rank the y_i parameters from those which are quite easy to deviate by the way of an adequate design process to those which cannot be managed easily (as maximum aerodynamic efficiency for instance). Thanks to this set of information, robust design can be achieved by looking for the best set of deviations (according to design facility) that is able to contain the risk on the quantities of interest below a required value while minimizing selected cost criteria.

At this stage, a last question requires some clarification: how do evolve the uncertainty ϵ_{F_i} when a deviation F_i is assumed for the purpose of risk quantification. Three considerations are helping us:

- The closer the deviation is from the maximum challenging value, the more reduced is the uncertainty because the use of refined models is compulsory in this situation.

- Here also, expert judgment can be helpful to determine the minimum amplitude of uncertainty.
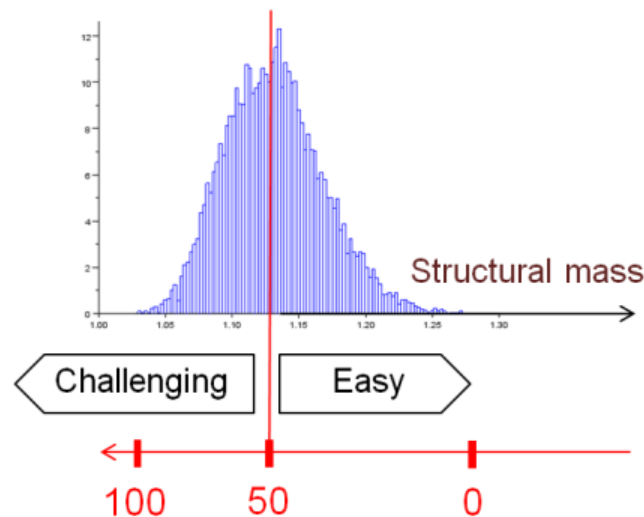
Figure 3.24: Challenging and easy side.

- Closer the design process is to its end, lower is this uncertainty.

In order to match with the first and second points, we propose the following mechanism:

- The original distribution is just scaled so that the peak remains at the current deviated value and the most challenging deviated value remains unchanged (anchored).

- When minimum uncertainty range is reached according to expert advice, the scaled distribution is not narrowed anymore and, consequently, the support will move accordingly.

- If the deviation overtakes the anchor value (limit of the support of the original distribution), then the difficulty increases linearly according to the relative distance to the nominal value of the original distribution.

- The quantile corresponding to the deviated value according to the original distribution is interpreted as the level of difficulty of the deviation.

- A symmetrical behavior is assumed on the "easy" side but with a decrease of the difficulty quotation according to the quantile versus the original distribution (quotation being floored to zero).

This mechanism is illustrated in Figure 3.25. It may be seen as artificial but it has some interesting properties:

- The most probable value is always the current one, which is in line with the intuition.

- The most challenging deviation is clearly identified by a quotation of 100 but can be overtaken (on expert advice for instance).

- The quantile corresponding to the deviated value according to the original distribution being interpreted as the level of difficulty of the deviation, this quotation evolves accordingly to the shape of the distribution, which is also in line with intuition: for non symmetrical distributions, steep ramps reveal hardly possible deviations.
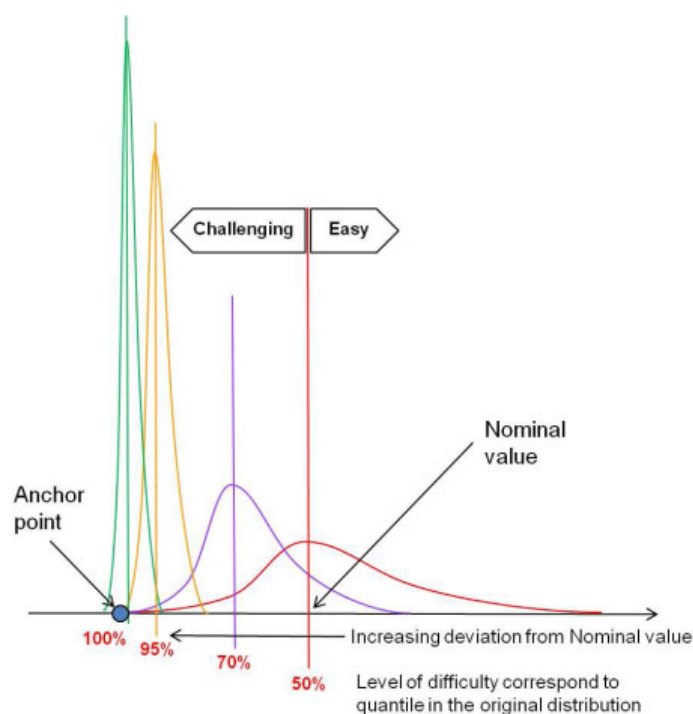
Figure 3.25: Model uncertainty distribution arbitrarily correlated to the design effort.

3.7 Conclusion

In this chapter, a methodology for quantifying and modelling the model error uncertainty for aircraft design is given.

First, the process used to identify the error of each aircraft design model is presented. It is based on a comparison between the model prediction and data contained in the available aircraft database.

Then, we first consider a classical distribution function for the uncertainty quantification but finally, we build up a generic distribution function called Beta-Mystique. It is generic and is used to quantify the uncertainty on errors of all aircraft design models.

Then, the particular case of performance model is pointed out. A specific methodology has been put in place in order to quantify the uncertainty of performance models which have a different structure compare to geometry, weight, aerodynamic and propulsion models. For this purpose, a methodology based on an optimization process is presented.

A correlation study is done in order to test the independence of aircraft design model errors. These correlations have induced some adaptations in the architecture of the models and given some insight in the way to use them into a computational process. Finally, in the last section of this chapter, we performed a correlation study. The results highlight dependences between some model errors. Further investigations and adapted method can be computed to be able to take into account this correlation. This is the subject of future work.

Finally, an interpretation of the predictive uncertainty in terms of design freedom has been proposed.

Chapter 4

Uncertainty propagation in Overall Aircraft Design

Contents

In the previous chapter, we have identified and quantified the main source of uncertainty involved in preliminary aircraft design. As aforementioned, the final objective of this study is to use this uncertainty into a robust optimisation method in order to directly generate robust aircraft configuration. This uncertainty can first be used in the frame of uncertainty propagation. As we have seen in chapter 2, the aim of the uncertainty propagation is to carry out the uncertainty through the system thanks to a numerical method, in order to assess the uncertainty effect on the output of the system. In this study, we apply the propagation in order to:

- Evaluate the effect of SiMCAD model error uncertainty on the operational performances of the aircraft,

- And evaluate the protection offered by the MSP (cf. section 1.2.2) used by FPO engineers to generate oversized aircraft configuration.

We use the well-known uncertainty propagation method which is the Monte Carlo method. We also present another uncertainty propagation method called moment propagation and compare the results with those obtained with Monte Carlo method.

4.1 Uncertainty propagation methods

There exist several uncertainty propagation methods. Table 4.1 displays the main groups.

Methods	Criterion	Method calculation	Computation time
Deterministic method (min-max)	Range	First order optimisation method	Potentially high
Numerical Integration	All probabilistic criteria	Integration by quadrature	Exponential
Taylor Approximation	Variance	First order Taylor development	Low
Monte Carlo Simulation	All probabilistic criteria	Random sampling	High
Variance-Reduction Techniques	Exceedance probability	Conditional MC, Importance sampling, Directional sampling	Reasonable
FORM-SORM methods	Exceedance probability	Numerical approximation	Low
Hybrid methods	Exceedance probability	Importance sampling	Reasonable
Method based on stochastic development	Moment - PDF - Exceedance probability	Development onto functional basis	Reasonable

Table 4.1: Uncertainty propagation methods

4.1.1 Monte Carlo method

Monte Carlo (MC) simulation (already presented in chapter 2) is the most widely used method. It presents the advantage to provide a confidence interval around the estimated value, and this is a measure of the degree of precision of this estimation. Nevertheless, its main drawback is the prohibitive computation time. This is particularly the case when the simulation models used are

themselves time consuming. However, MC simulation may be used at preliminary design phase because the models used are most of the time simple, consequently, computationally inexpensive. As our study is done at preliminary design stage, we first use MC simulation for propagating the uncertainty into the aircraft design process. Nevertheless, in order to further reduce the computation time, we used the moment propagation method.

4.1.2 Method of moment propagation

Monte Carlo method is known to be time consuming. Thus, we have decided to test another much cheaper method: moment propagation method.

The method of moment propagation is an analytical method. As it is based on Taylor series expansion, it requires derivatives information. The method is called *First order method of moments* when only the first-order approximation is considered and *Second order method of moments* when both first and second orders approximation are considered. Moment propagation method have been already used in aircraft design context such as in [GLK02, DMC12]: for example, first order method of moment is widely used in CFD calculation context [PNT04].

4.1.2.1 Description of the method

This method refers to the propagation of the moments through a function or a system in order to compute statistical quantities of the system output according to a given uncertainty. Consider a function $f \in C^1(\mathbb{R}^n, \mathbb{R})$ such that $y_0 = f(X)$ where $X = (x_1, \ldots, x_n)$. Let ϵ_{x_i} and ϵ_y be the uncertainty respectively attached to x_i and to the output y where ϵ_{x_i} is defined by the mean-variance couple $(\mu_{x_i}, \sigma_{x_i}^2)$ and ϵ_y by (μ_y, σ_y^2). We have thus:

$$y_0 + \epsilon_y = f(x_1 + \epsilon_{x_1}, \ldots, x_n + \epsilon_{x_n}). \tag{4.1}$$

With Equation (4.1), the first order Taylor development of f gives:

$$y_0 + \epsilon_y = f(X) + \sum_i \partial_{x_i} f(X)\epsilon_{x_i} + o(||(\epsilon_{x_1}, \ldots, \epsilon_{x_n})||).$$

and then

$$\epsilon_y = \sum_i \partial_{x_i} f(X)\epsilon_{x_i} + o(||(\epsilon_{x_1}, \ldots, \epsilon_{x_n})||). \tag{4.2}$$

We can use Equation (4.2) to evaluate the first four moments of ϵ_y by using the following formula presented in Chapter 2:

$$m_n(X) = E\left(\left(X - E(X)\right)^n\right), \quad \forall n \geq 2.$$

Keeping only the first order terms, we obtain:

$$
\begin{aligned}
\mu_y &= E(\epsilon_y) \\
&\simeq \sum_i \partial_{x_i} f(X)\mu_{x_i},
\end{aligned}
\tag{4.3}
$$

$$
\begin{aligned}
\sigma_y^2 &= E((\epsilon_y - E(\epsilon_y))^2) \\
&\simeq \sum_i (\partial_{x_i} f(X))^2 \sigma_{x_i}^2,
\end{aligned}
\tag{4.4}
$$

$$
\begin{aligned}
s_y &= E\left((\epsilon_y - E(\epsilon_y))^3\right) \\
&\simeq \sum_i (\partial_{x_i} f(X))^3 s_{x_i},
\end{aligned}
\tag{4.5}
$$

$$
\begin{aligned}
k_y &= E\left((\epsilon_y - E(\epsilon_y))^4\right) \\
&\simeq \sum_i (\partial_{x_i} f(X))^4 k_{x_i} + 6\sum_{j=1}^{n}\sum_{i=1}^{j-1}\left((\partial_{x_i} f(X))^2(\partial_{x_j} f(X))^2\sigma_{x_j}^2\sigma_{x_i}^2\right).
\end{aligned}
\tag{4.6}
$$

To be more precise, the previous formulas should incorporate derivatives of higher order of function f. Nevertheless, in this case, the formulas become so enormous that they turn out to be unmanageable. Besides, practical tests made with computational processes used in this study have shown sufficiently accurate results with only first order approach. This way of propagating uncertainty is more precise when variability (errors) are contained into limited variations or when the modelled phenomena are varying smoothly (which is often the case in our business).

Equations 4.5 and 4.6 have been obtained by using the centered definition of the third and fourth moments. In some cases, their values can become very big and then it is better to use the standardized form for the skewness and the kurtosis (see Section 2.4.3). Thus, the previous formulas for moments have to be adapted to be able to propagate the standardized moments. Then, we obtain the following propagation formulas:

$$Sk_y = \frac{1}{(\sigma_y^2)^{\frac{3}{2}}} \sum_i (\partial_{x_i} f(X))^3 Sk_{x_i} (\sigma_{x_i}^2)^{\frac{3}{2}}, \tag{4.7}$$

$$Ku_y = \frac{1}{(\sigma_y^2)^2} \sum_i (\partial_{x_i} f(X))^4 Ku_{x_i} (\sigma_{x_i}^2)^2. \tag{4.8}$$

Let us first prove the skewness formulae (4.7).

$$Sk_y = E\left(\left(\frac{\epsilon_y - E(\epsilon_y)}{\sigma_y}\right)^3\right) = \frac{1}{(\sigma_y^2)^{\frac{3}{2}}} s_y = \frac{1}{(\sigma_y^2)^{\frac{3}{2}}} \sum_{i=1}^n (\partial_{x_i} f(X))^3 s_{x_i}.$$

Since $s_{x_i} = (\sigma_{x_i}^2)^{\frac{3}{2}} Sk_{x_i}$, we have

$$Sk_y = \frac{1}{(\sigma_y^2)^{\frac{3}{2}}} \sum_{i=1}^n (\partial_{x_i} f(X))^3 Sk_{x_i} (\sigma_{x_i}^2)^{\frac{3}{2}}.$$

The proof for the excess kurtosis is given hereafter.

$$\begin{aligned} Ku_y &= E\left(\left(\frac{\epsilon_y - E(\epsilon_y)}{\sigma_y}\right)^4\right) - 3 \\ &= \frac{1}{(\sigma_y^2)^2} k_y - 3 \\ &= \frac{1}{(\sigma_y^2)^2} \left(\sum_{i=1}^n (\partial_{x_i} f(X))^4 k_{x_i} + 6 \sum_{j=1}^n \sum_{i=1}^{j-1} \left((\partial_{x_i} f(X))^2 (\partial_{x_j} f(X))^2 \sigma_{x_j}^2 \sigma_{x_i}^2\right)\right) - 3 \\ &= \frac{1}{(\sigma_y^2)^2} \left(\sum_{i=1}^n (\partial_{x_i} f(X))^4 k_{x_i} + 6 \sum_{j=1}^n \sum_{i=1}^{j-1} \left((\partial_{x_i} f(X))^2 (\partial_{x_j} f(X))^2 \sigma_{x_j}^2 \sigma_{x_i}^2\right) - 3(\sigma_y^2)^2\right). \end{aligned}$$

Since $k_{x_i} = (Ku_{x_i} + 3)(\sigma_{x_i}^2)^2$ and $\left(\sum_{i=1}^{n} a_i b_i \right)^2 = \sum_{i=1}^{n} a_i^2 b_i^2 + 2 \sum_{j=1}^{n} \sum_{i=1}^{j-1} a_i b_i a_j b_j$, we have:

$$
\begin{aligned}
Ku_y &= \frac{1}{(\sigma_y^2)^2} \left(\sum_{i=1}^{n} (\partial_{x_i} f(X))^4 (Ku_{x_i} + 3)(\sigma_{x_i}^2)^2 + 6 \sum_{j=1}^{n} \sum_{i=1}^{j-1} \left((\partial_{x_i} f(X))^2 (\partial_{x_j} f(X))^2 \sigma_{x_j}^2 \sigma_{x_i}^2 \right) \right. \\
&\quad \left. - 3 \left(\sum_{i=1}^{n} (\partial_{x_i} f(X))^2 \sigma_{x_i}^2 \right)^2 \right) \\
&= \frac{1}{(\sigma_y^2)^2} \left(\sum_{i=1}^{n} (\partial_{x_i} f(X))^4 (\sigma_{x_i}^2)^2 Ku_{x_i} + 3 \left[\left(\sum_{i=1}^{n} (\partial_{x_i} f(X))^4 (\sigma_{x_i}^2)^2 \right) \right. \right. \\
&\quad \left. \left. + 2 \sum_{j=1}^{n} \sum_{i=1}^{j-1} \left((\partial_{x_i} f(X))^2 (\partial_{x_j} f(X))^2 \sigma_{x_j}^2 \sigma_{x_i}^2 \right) \right] - 3 \left(\sum_{i=1}^{n} (\partial_{x_i} f(X))^2 \sigma_{x_i}^2 \right)^2 \right) \\
&= \frac{1}{(\sigma_y^2)^2} \sum_{i=1}^{n} (\partial_{x_i} f(X))^4 (\sigma_{x_i}^2)^2 Ku_{x_i}.
\end{aligned}
$$

Coupling phenomenon The previous formulas have been established in the case of a complete independence of the model errors as measured in reference to the airplane database. Nevertheless, within a computation flow as we can find in the expression of a model, variables are most often correlated. Figure 4.1 illustrates the coupling phenomenon that naturally appears within the computation flow. Models $M1$ and $M2$ are both using the input X_2. Consequently,
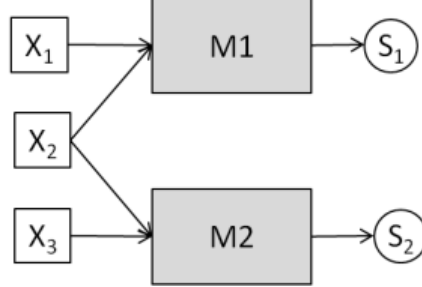
Figure 4.1: Coupling phenomenon within the computation flow

the outputs S_1 and S_2 are not independent. In this case, it is not allowed to use the previous formulas in order to add or substract S_1 tO S_2. Consequently, we have to look for a way to propagate the uncertainty by taking into account this coupling phenomenon. In this study, we consider that overall inputs of the whole computation are independent.

To achieve this objective, we compute the partial derivative of any internal variable X versus any overall input a_i denoted $\partial_{a_i} X$ such as

$$
\frac{\partial X}{\partial_{a_i}} = \partial_{a_i} X.
$$

Consider a generic function f with 2 entries x_1 and x_2:

$$
y = f \left(x_1 + \epsilon_{x_1}(\mu_{x_1}, \sigma_{x_1}^2, Sk_{x_1}, Ku_{x_1}), x_2 + \epsilon_{x_2}(\mu_{x_2}, \sigma_{x_2}^2, Sk_{x_2}, Ku_{x_2}) \right).
$$

For this function f, the formulas 4.3,4.4, 4.7 and 4.8 becomes:

$$\mu_y = \sum_{i=1}^{2} \partial_{x_i} f(X) \mu_{x_i}, \tag{4.9}$$

$$\sigma_y^2 = \sum_{i=1}^{2} (\partial_{x_i} f(X))^2 \sigma_{x_i}^2 + 2 \prod_{i=1}^{2} \partial_{x_i} f(X) \sum_{j} \partial_{a_j} x_1 \partial_{a_j} x_2 \cdot \sigma_{a_j}^2, \tag{4.10}$$

$$Sk_y = \frac{1}{(\sigma_y^2)^{\frac{3}{2}}} \sum_{i=1}^{2} (\partial_{x_i} f(X))^3 Sk_{x_i} (\sigma_{x_i}^2)^{\frac{3}{2}} + \dots \tag{4.11}$$

$$3 \left((\partial_{x_1} f(X))^2 \partial_{x_2} f(X) + (\partial_{x_2} f(X))^2 \partial_{x_1} f(X) \right) \times \dots \tag{4.12}$$

$$\sum_{j} \left((\partial_{a_j} x_1)^2 \partial_{a_j} x_2 + (\partial_{a_j} x_2)^2 \partial_{a_j} x_1 \right) Sk_{a_j} \sigma_{a_j}^2)^{\frac{3}{2}}, \tag{4.13}$$

$$Ku_y = \frac{1}{(\sigma_y^2)^2} \sum_{i=1}^{2} (\partial_{x_i} f(X))^4 Ku_{x_i} (\sigma_{x_i}^2)^2 + \dots \tag{4.14}$$

$$4(\partial_{x_1} f(x_0))^3 \partial_{x_2} f(x_0) \sum_{j} (\partial_{a_j} x_1)^3 \partial_{a_j} x_2 Ku_{a_j} (\sigma_{a_j}^2)^2 + \dots \tag{4.15}$$

$$4\partial_{x_1} f(x_0) (\partial_{x_2} f(x_0))^3 \sum_{j} \partial_{a_j} x_1 (\partial_{a_j} x_2)^3 Ku_{a_j} (\sigma_{a_j}^2)^2 + \dots \tag{4.16}$$

$$6(\partial_{x_1} f(x_0))^2 (\partial_{x_2} f(x_0))^2 \sum_{j} (\partial_{a_j} x_1)^2 (\partial_{a_j} x_2)^2 Ku_{a_j} (\sigma_{a_j}^2)^2, \tag{4.17}$$

where $X = (x_1, x_2)$. The propagation stream must know how to carry as well the gradient of overall inputs g_y.

These formulas have been implemented in the models and processes such that mean, variance, kurtosis and skewness of the current calculated value are available at each step of the calculation. This way of doing allows to save a lot of time compared to Monte-Carlo method: the uncertainty distribution of all outputs is obtained in only one run.

4.1.2.2   Practical computing: overloading operation

Our main objective is to capture the uncertainty into a structure in order to be able to use it afterwards. The uncertainty has been captured thanks to the Beta Mystique distribution which is characterised by its four parameters a, b, Z, P. Moreover, we have presented earlier the moment propagation formula of order 1 to 4 through a function or a system (cf. Equations (4.3), (4.4), (4.7) and (4.8)). The purpose now is to find a way to put together all these informations in order to explain to computers how to manipulate the Beta Mystique distribution. The first idea is to concentrate this information into a vector. Nevertheless, vectors already have their own algebra in the computers and the way computers are manipulating vectors is not suited for the parameters of the Beta Mystique distribution. In fact, we have to create our own data type called 'bmyst' and denoted { 'bmyst',x,a,b,Z,P}. Then, we have to define a new algebra, which propagates the values and the moments. This is done by overloading the basic operators.

4.1.2.2.1   Overloading operators   Any modern language offers the possibility to overload its basic operators. This operation is completely independent of the coding of the models.

4.1.2.2.2   An example with the addition operator   We consider the function f defined by

$$y = f(x_1, x_2) = x_1 + x_2$$

and its gradient

$$\begin{cases} \partial_{x_1} f = 1 \\ \partial_{x_2} f = 1. \end{cases}$$

The vectors $\begin{pmatrix} \mu_{x_1} \\ \sigma_{x_1}^2 \\ Sk_{x_1} \\ Ku_{x_1} \end{pmatrix}$ and $\begin{pmatrix} \mu_{x_2} \\ \sigma_{x_2}^2 \\ Sk_{x_2} \\ Ku_{x_2} \end{pmatrix}$ contain the first four moments of x_1 and x_2. The objective is to determine the vector $(\mu_y, \sigma_y^2, Sk_y, Ku_y)$ containing the first four moments of y in order to define the addition operator $\oplus$.

Let us first determine the expression of μ_y. From Equation (4.3), we have

$$\mu_y = [\partial_{x_1} f, \partial_{x_2} f] \begin{pmatrix} \mu_{x_1} \\ \mu_{x_2} \end{pmatrix} = \mu_{x_1} + \mu_{x_2}.$$

Doing the same for σ_y^2, we obtain:

$$\sigma_y^2 = \sigma_{x_1}^2 + \sigma_{x_2}^2.$$

The expression of the skewness of y, denoted Sk_y, is given by:

$$\begin{aligned}
Sk_y &= \left[(\partial_{x_1} f)^3, (\partial_{x_2} f)^3 \right] \begin{pmatrix} \dfrac{Sk_{x_1}(\sigma_{x_1}^2)^{\frac{3}{2}}}{(\sigma_y^2)^{\frac{3}{2}}} \\ \dfrac{Sk_{x_2}(\sigma_{x_2}^2)^{\frac{3}{2}}}{(\sigma_y^2)^{\frac{3}{2}}} \end{pmatrix} \\
&= \frac{Sk_{x_1}(\sigma_{x_1}^2)^{\frac{3}{2}} + Sk_{x_2}(\sigma_{x_2}^2)^{\frac{3}{2}}}{(\sigma_y^2)^{\frac{3}{2}}} \\
&= \frac{Sk_{x_1}(\sigma_{x_1}^2)^{\frac{3}{2}} + Sk_{x_2}(\sigma_{x_2}^2)^{\frac{3}{2}}}{(\sigma_{x_1}^2 + \sigma_{x_2}^2)^{\frac{3}{2}}}.
\end{aligned}$$

Doing the same for the kurtosis Ku_y, we have:

$$Ku_y = \frac{Ku_{x_1}(\sigma_{x_1}^2)^2 + Ku_{x_2}(\sigma_{x_2}^2)^2}{(\sigma_{x_1}^2 + \sigma_{x_2}^2)^2}.$$

Finally, the addition operator $\oplus$ is defined as follows:

$$\begin{pmatrix} y \\ \mu_y \\ \sigma_y^2 \\ Sk_y \\ Ku_y \end{pmatrix} = \begin{pmatrix} x_1 \\ \mu_{x_1} \\ \sigma_{x_1}^2 \\ Sk_{x_1} \\ Ku_{x_1} \end{pmatrix} \oplus \begin{pmatrix} x_1 \\ \mu_{x_2} \\ \sigma_{x_2}^2 \\ Sk_{x_2} \\ Ku_{x_2} \end{pmatrix} = \begin{pmatrix} x_1 + x_2 \\ \mu_{x_1} + \mu_{x_2} \\ \sigma_{x_1}^2 + \sigma_{x_2}^2 \\ \dfrac{Sk_{x_1}(\sigma_{x_1}^2)^{\frac{3}{2}} + Sk_{x_2}(\sigma_{x_2}^2)^{\frac{3}{2}}}{(\sigma_{x_1}^2 + \sigma_{x_2}^2)^{\frac{3}{2}}} \\ \dfrac{Ku_{x_1}(\sigma_{x_1}^2)^2 + Ku_{x_2}(\sigma_{x_2}^2)^2}{(\sigma_{x_1}^2 + \sigma_{x_2}^2)^2} \end{pmatrix}.$$

4.1.2.2.2.1   An example with the subtraction operator   Following the same process with

$$y = f(x_1, x_2) = x_1 - x_2$$

and

$$\begin{cases} \partial_{x_1} f = 1 \\ \partial_{x_2} f = -1, \end{cases}$$

we obtain the formulas for the subtraction operator $\ominus$:

$$\begin{pmatrix} y \\ \mu_y \\ \sigma_y^2 \\ Sk_y \\ Ku_y \end{pmatrix} = \begin{pmatrix} x_1 \\ \mu_{x_1} \\ \sigma_{x_1}^2 \\ Sk_{x_1} \\ Ku_{x_1} \end{pmatrix} \ominus \begin{pmatrix} x_1 \\ \mu_{x_2} \\ \sigma_{x_2}^2 \\ Sk_{x_2} \\ Ku_{x_2} \end{pmatrix} = \begin{pmatrix} x_1 - x_2 \\ \mu_{x_1} - \mu_{x_2} \\ \sigma_{x_1}^2 + \sigma_{x_2}^2 \\ \dfrac{Sk_{x_1}(\sigma_{x_1}^2)^{\frac{3}{2}} - Sk_{x_2}(\sigma_{x_2}^2)^{\frac{3}{2}}}{(\sigma_{x_1}^2 + \sigma_{x_2}^2)^{\frac{3}{2}}} \\ \dfrac{Ku_{x_1}(\sigma_{x_1}^2)^2 + Ku_{x_2}(\sigma_{x_2}^2)^2}{(\sigma_{x_1}^2 + \sigma_{x_2}^2)^2} \end{pmatrix}.$$

The overloading operation is not enough to develop applications based on Beta-Mystique distribution and moment propagation method. The processes used contain also System Solvers and Optimizers. These objects also have to be overloaded.

4.1.2.2.3   Overloading System Solvers   Overloading a system solver is not so difficult: we have to accept to stay at the first order level that means to take into account only the Jacobian information.

Any system of n equations with n unknown variables can be formalized as follows:

$$F \text{ being a function from } \mathbb{R}^n \longrightarrow \mathbb{R}^n,$$
$$\text{Find } X_0 = [x_{0_1}, \ldots, x_{0_n}]$$
$$\text{such that } F(X_0) = 0.$$

The variable X_0 being the output of the system, the objective is to compute the variability denoted ϵ_x in function of the variability denoted ϵ_F coming from the function F. By introducing uncertainty propagation, the problem becomes:

$$F \text{ being a function from } \mathbb{R}^{n+p} \longrightarrow \mathbb{R}^n,$$
$$\text{Find } X_0 = [x_{0_1}, \ldots, x_{0_n}] \text{ and } \epsilon_{X_0} = [\epsilon_{x_{0_1}}, \ldots, \epsilon_{x_{0_p}}]$$
$$\text{such that: } F(X_0 + \epsilon_{X_0}, Y + \epsilon_Y) = 0,$$

where $Y = [y_1, \ldots, y_p]$ are parameters associated to $\epsilon_Y = [\epsilon_{y_1}, \ldots, \epsilon_{y_p}]$. The first order Taylor's development of F gives:

$$F(X_0 + \epsilon_{X_0}, Y + \epsilon_Y) \simeq F(X_0, Y + \epsilon_Y) + \left[\frac{\partial F}{\partial X}\right] \epsilon_{X_0}.$$

We obtain:

$$X_0 \text{ being solution of the system: } F(X, Y) = 0,$$
$$F(X_0, Y + \epsilon_Y) + \left[\frac{\partial F}{\partial X}\right] \epsilon_{X_0} = 0 \iff \epsilon_{X_0} = -\left[\frac{\partial F}{\partial X}\right]^{-1} F(X_0, Y + \epsilon_Y).$$

In other terms, the uncertainty output $U_{X_0} + \epsilon_{X_0}$ is computed by replacing the original non - linear system by its local linear approximation (this is the Newton-Raphson scheme) and propagating the uncertainty ϵ_Y through it :

$$U_{X_0} = X_0 - \left[\frac{\partial F}{\partial X}\right]^{-1} F(X_0, Y + \epsilon_Y).$$

In doing so, we are supposing that the variability of the partial derivative $\left[\frac{\partial F}{\partial X}\right]$ is negligible. In addition, moment propagation offers an interesting opportunity to compute the Jacobian. If we apply propagation formula of the first moment (Equations (4.3), (4.4), (4.7), (4.8)) to function F, we obtain:

$$\mu_F = \sum_i \left[\frac{\partial F}{\partial X}\right]_{X_0} \mu_{x_i},$$

where μ_{x_i} is the first moment of the variability of the component x_i of the vector X. Consequently, if we propagate the following input through the function F:

$$X_i = \begin{pmatrix} x_1 = 'uv', x_{0_1}, 0, 0, 0, 0 \\ \dots \\ x_i = 'uv', x_{0_i}, 1, 0, 0, 0 \\ \dots \\ x_n = 'uv', x_{0_n}, 0, 0, 0, 0 \end{pmatrix},$$

we obtain in its output the column i of the Jacobian of F:

$$F(X_i) = \begin{pmatrix} \frac{\partial F_1}{\partial x_i} \\ \dots \\ \frac{\partial F_j}{\partial x_i} \\ \dots \\ \frac{\partial F_n}{\partial x_i} \end{pmatrix}.$$

Doing so for each i will produce the complete Jacobian. This method is well known in the field of automatic derivation.

The Scilab function *fsolve* is an example of System Solvers which has been overloaded for this study.

4.1.2.2.4 Overloading Optimizers Optimisation Operators cannot be overloaded in a generic way. Nevertheless, in some particular cases which are frequent in Overall Aircraft Design context, Optimization Operators can be overloaded. Two cases are described hereafter.

As aforementioned in Chapter 1, most of the time, the optimum of the OAD optimization is located at the intersection of two active constraints. If we assume that these active constraints are not impacted by the injected uncertainty (which has to be carefully checked), then the Optimisation Operator can be replaced by a System Solving Operator and uncertainty can be propagated through it.

There is another situation where overloading an Optimization Operator can be done: when

- there is no constraint on output;

- and uncertainty does not push the optimum outside a fixed domain in which a polynomial approximation of the function remains valid.
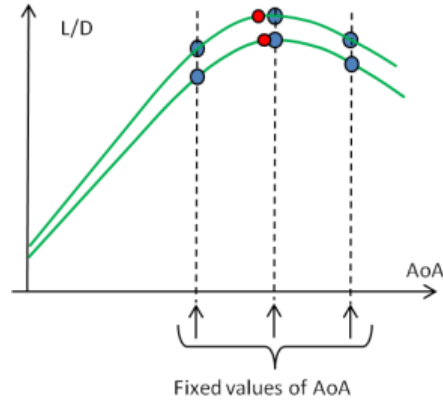
Figure 4.2: Maximum Lift over Drag ratio of an aerodynamic polar

A good illustration of this situation is the maximum Lift over Drag ratio (LoD) of an aerodynamic polar (cf. Figure 4.2).

Three points are computed at fixed values of Angle of Attack. Coefficients and maximum of the second degree polynomial function that goes through these three points can be computed using a pure algebraic direct process. As a result, propagation of uncertainty through this process can be managed simply using basic operators overloading.

From a general point of view, moment propagation can be achieved easily each time a numerical result can be obtained using a direct sequence of algebraic computations (non iterative, except for system solving).

4.1.2.2.4.1 Numerical experiments The purpose of these numerical experiments is to compare results given by the two propagation methods: Monte Carlo sampling and moment propagation.

Consider two variables x and y following a Beta-Mystique distribution such as $x \sim bmyst(a_x, b_x, Z_x, P_x)$ and $y \sim bmyst(a_y, b_y, Z_y, P_y)$. Consider two samples of data S_1 and S_2, both of size n, coming from x and y. We proceed as follows:

- First, we perform the addition of x and y using the moment propagation method combined with the overloading addition. The result of this addition gives the variable w such as $w \sim bmyst(a_w, b_w, Z_w, P_w)$: $(w, a_w, b_w, Z_w, P_w) = (x, a_x, b_x, Z_x, P_x) \oplus (y, a_y, b_y, Z_y, P_y)$.

- Then, we perform the addition of the two samples S_1 and S_2. We obtain the third sample S_3. From this third sample S_3, we compute the parameters of the Beta-Mystique distribution associated. We say that S_3 is a sample of the variable v such as $v \sim bmyst(a_v, b_v, Z_v, P_v)$.

- Finally, we compare $w \sim bmyst(a_w, b_w, Z_w, P_w)$ and $v \sim bmyst(a_v, b_v, Z_v, P_v)$.

Let us consider the following values: $x \sim bmyst(-0.4, 0.4, -0.5, 1)$, $y \sim bmyst(-0.5, 0.5, -0.42, 0.56)$. With those values, we obtain $w \sim bmyst(-1.078, 0.845, 0.12, 2.86)$. The result is displayed on Figure 4.3.

The addition of the two samples S_1 and S_2 gives $v \sim bmyst(-1.062, 0.854, 0.1, 2.77)$. The result is displayed on Figure 4.4.

A graphical comparison can be done with the third graphics of Figure 4.3 and 4.4. We can see that the results are practically the same. Thus, with these results we are able to conclude that the moment propagation method is as efficient as the Monte Carlo method. Nevertheless, even if the two methods give close results, the execution time makes a difference between
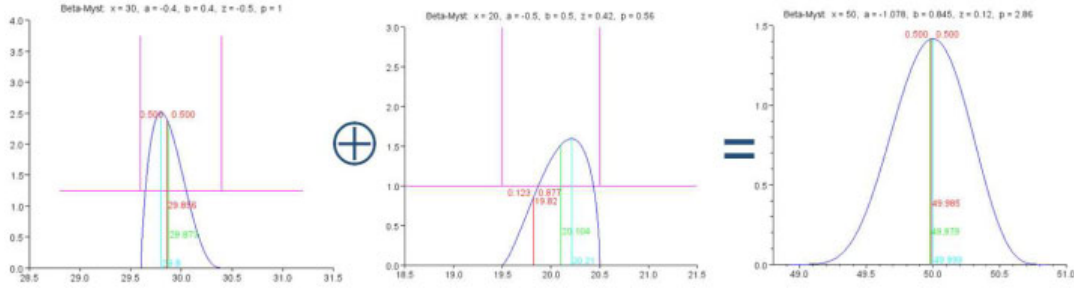
Figure 4.3: Example of addition with moment propagation combined with operation overloading
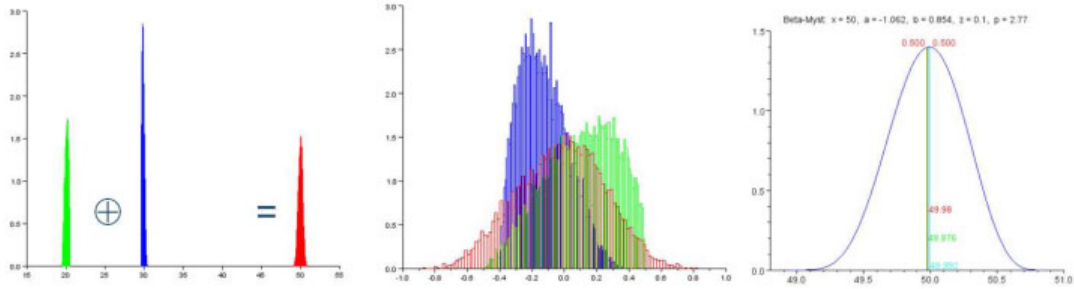
Figure 4.4: Example of addition of two samples of bmyst data

them.Indeed, the moment propagation method is less greedy in terms of computation time than the MC method. Thus, if execution time is an important issue, it is more interesting to use moment propagation method (then to put more effort on the code development) than MC method. This choice is possible only in the case where the derivative information is available.

4.1.2.3 Limitation of moment propagation method

The moment propagation method presents a drawback: its precision decreases in case of high non-linearity. This can be improved by taking into account the second order derivative information.This work is currently in progress. It is not an easy task to carry out because of the overloading operation. Note that this limitation does not matter in this study. Even in the case where all the uncertainty has been considered, the results stayed inside the frontier or at most, on the frontier.

Otherwise, when distortions brought by non-linearity are not acceptable, the way to proceed is to use Monte Carlo method.

4.1.2.4 Problem when transforming Beta-Mystique moments into parameters

In Chapter 3, we have presented the Beta Mystique distribution and its characteristics. Thanks to the method of moment propagation, we have access to the formula that allows to transform the Beta-Mystique parameters into Mean, Variance, Skewness and Kurtosis and conversely:

$$a = Mean - p_1 \cdot fac, \tag{4.18}$$

$$b = Mean + q_1 \cdot fac, \tag{4.19}$$

$$Z = \frac{2 \cdot Mean + q_1 \cdot fac}{b - a}, \tag{4.20}$$

$$P = \frac{p_1 + q_1 - 2}{3.3}, \tag{4.21}$$

where

$$
\left\{
\begin{array}{rcl}
sum & = & \dfrac{6 \cdot \left(Skewness^2 - Kurtosis - 2\right)}{2 \cdot Kurtosis - 3 \cdot Skewness^2}, \\[2ex]
prod & = & \dfrac{6 \cdot (sum + 1) \cdot sum^2}{Kurtosis \cdot (sum + 2) \cdot (sum + 3) + 30 \cdot sum + 36}, \\[2ex]
det & = & sum^2 - 4 \cdot prod, \\[1ex]
sign & = & \dfrac{Skewness}{|Skewness|}, \\[2ex]
p_1 & = & \dfrac{sum - sign \cdot \sqrt{det}}{2}, \\[2ex]
q_1 & = & \dfrac{sum + sign \cdot \sqrt{det}}{2}, \\[2ex]
fac & = & \sqrt{Var \cdot \dfrac{p_1 + q_1 + 1}{p_1 \cdot q_1}}.
\end{array}
\right.
$$

The possibility to compute the (Z, P) parameters of the Beta-Mystique law from (Skewness, Kurtosis) does not mean that any couple of values (Skewness, Kurtosis) can be reached from a couple of values (Z, P). It appears that some positive Kurtosis values are not reachable for low absolute values of Skewness, as shown in Fig. 4.5 and Fig. 4.6.
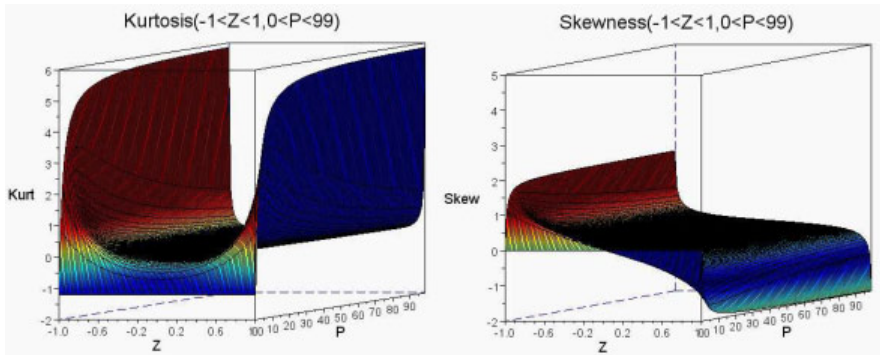
Figure 4.5: Illustration of values unreachable by Skewness and Kurtosis
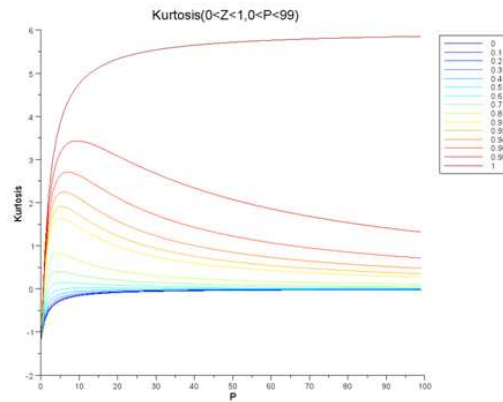
Figure 4.6: Illustration of values unreachable by Skewness and Kurtosis

In practice, we have proposed the following work around (equations (4.22), (4.23), (4.24)

and (4.25)) which has the effect to put a ceiling on the resulting value of P to 99:

$$P_{max} = 99, \tag{4.22}$$

$$K_{limit} = \min\left(Kurtosis, \frac{(9.9 \cdot P_{max} + 12) \cdot Skewness^2 - 12}{6.6 \cdot P_{max} + 10}\right), \tag{4.23}$$

$$sum = \frac{6 \cdot \left(Skewness^2 - K_{limit} - 2\right)}{2 \cdot K_{limit} - 3 \cdot Skewness^2}, \tag{4.24}$$

$$prod = \frac{6 \cdot (sum + 1) \cdot sum^2}{K_{limit} \cdot (sum + 2) \cdot (sum + 3) + 30 \cdot sum + 36}, \tag{4.25}$$

where K_{limit} is the maximum value of Kurtosis that can be reached with the current Skewness and a maximum value of P_{max} for the fourth parameter of the Beta-Mystique Law. In doing so, we alter the identification of some distribution shapes that have both a "thick base" and "high pick" as illustrated on Fig. 4.7. Further analysis is required to better manage this limitation of the Beta-Mystique law.
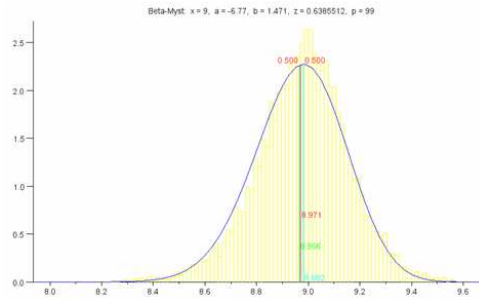
Figure 4.7: Example of limitation of Beta-Mystique law

4.2   Aircraft design process for uncertainty propagation

The aircraft design process has been presented in chapter 1 (cf. Figure 1.3). In the frame of the uncertainty propagation, we did not use the same process. We used a process without neither optimisation loop nor mass-mission loop. In this case, the aircraft characteristics are estimated with a sequential process. Starting from the values of *WingArea*, *SLSThrust* and *MTOW*, the values of each aircraft characteristics are evaluated from the geometry to the operational performances respectively in this order: geometry, engine, aerodynamics, masses and finally operational performances (this sequence is the backbone of the aircraft design process). The process is presented in Figure 4.8.

Figure 4.9 displays how the uncertainty is propagated into the sequential process with the Monte Carlo method. First, we generate a sample of model errors for each SiMCAD models with respect to their distribution. Then, we run the sequential design process for each individual of the sample. At the end, we obtain a sample of values for each operational performance. With these samples, we are able to evaluate the number of aircraft that do not fulfil the given requirements on each operational performance.

In Figure 4.9, the error ϵ_I introduced for the performance models is the intrinsic error evaluated in Chapter 3 Section 3.4. We remind that ϵ_I of performance models does not contain errors coming from aerodynamic and engine models which are nested in performance models.
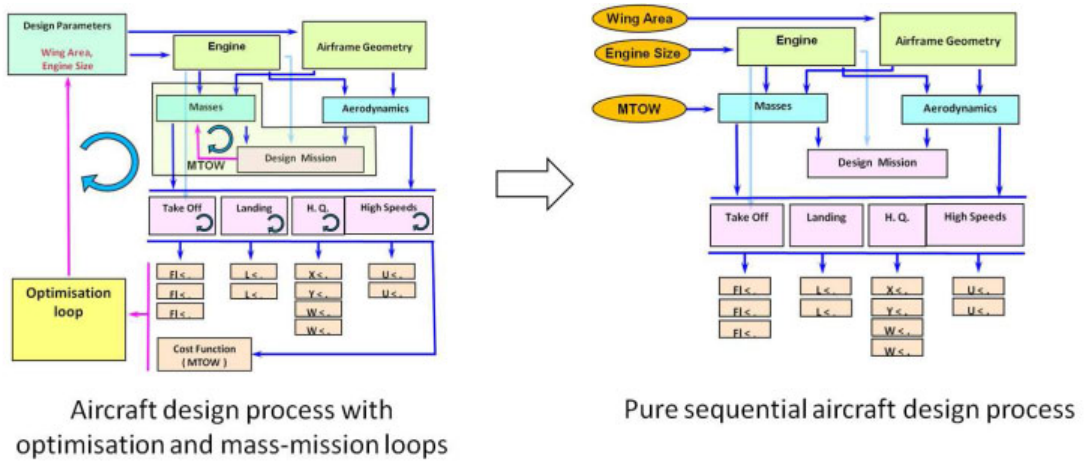
Figure 4.8: Pure sequential aircraft design
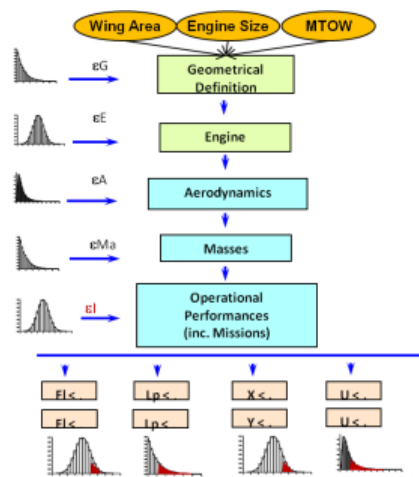
Figure 4.9: Uncertainty propagation in the sequential design process

4.3 Robustness study of Margin Setting Process

4.3.1 Strategy

As presented in section 1.2.2, MSP is a method used in FPO to compute aircraft configurations which integrates margins. The objective is to compute values for the three parameters *WingArea*, *SLSThrust* and *MTOW* and determine realistic but ambitious targets for detailed design stage. In this study, we focus only on the first objective.

The purpose of this section is to evaluate the protection offered by the MSP. Evaluating the protection offered by the MSP requires that we guess the performances of the final aircraft which is designed during detailed design phase with high level fidelity models. But models used for the simulation at preliminary design stage (i.e. SiMCAD models) are very simple and they have a much lower level of fidelity than the models used at the detailed design phase. Thus, we have to keep in mind that the evaluation we make at preliminary design is done by using simpler models which are less precise than the high fidelity models used in reality.

The strategy used to evaluate the MSP is based on the process presented in Figure 4.9. It is composed of the following steps:

1. Launch step 1 and step 2 of MSP in order to have values for *WingArea*, *SLSThrust* and *MTOW*. They are denoted $WingArea_{MSP}$, $SLSThrust_{MSP}$ and $MTOW_{MSP}$.

2. Define the set S of SiMCAD models which have an uncertainty on their error of simulation.

3. Generate a sample of model errors for each model in S from the probability density function identified during uncertainty quantification step.

4. Launch the sequential process of Figure 4.9 with $WingArea_{MAP}$, $SLSThrust_{MAP}$ and $MTOW_{MAP}$ as inputs.

5. Store the probability to not fulfil the requirement for each operational performance.

For that purpose, we used two methods: Monte Carlo method and moment propagation method. The results obtained are presented in the next section on an example of a twinjet aircraft design.

4.3.2 Test case and results

In this example, we look for an aircraft which fulfils the specifications stored in Table 4.2. It is a twin-jet aircraft, which can carry 160 passengers and which has a 3500 NM design range.

The first step of the strategy is to launch MSP process. The deterioration coefficients used in step 1 are fixed to: $kL/D = -1$, $kSFC = +1$ and $kMWE = +1$ (we remind that this deterioration coefficients are applied to the efficiency factors L/D, SFC and MWE and that they are equal to $kL/D = 0$, $kSFC = 0$ and $kMWE = 0$ in step 2: there is no deterioration considered in step 2). With this initialisation, the aircraft configuration proposed by MSP is given by the triplet $(WingArea, SLSThrust, MTOW) = (131.3~m^2, 145330.77~N, 77976.8~kg)$.

In this example, we analysed the capacity of this aircraft configuration to fulfil the requirements when all SiMCAD models are considered as uncertain: the set S contains all SiMCAD models.

Design Assumptions	
WingAr	8
BPR	8
Design range	3500 NM
Cruise mach	0.76 mach
Number of engines	2
Number of passengers	160
Reference altitude	35000 ft
Requirements	
Range	3500 NM
Take Off Field Length	1700 m
Climb ceiling	33000 ft
Cruise ceiling	33000 ft
Approach speed	137 kt

Table 4.2: Design specifications

Constraint Name	Probability of satisfaction
Range	0.41
TOFL	0.73
Climb	1
Cruise	1
Vapp	0.98

Table 4.3: Results of MSP robustness study with Monte Carlo method

4.3.2.1 Propagation with Monte Carlo method

By using the Monte Carlo method, we obtained the results stored in Table 4.3. Most of the operational constraints seems to be well protected by the MSP except two of them: the range and the Take-Off Field Length (TOFL) constraints. Nevertheless, these constraints are very important ones. TOFL is a strong constraint: an increase of a few % of the necessary Take Off Field Length can lead to the inability of the aircraft to operate in some airports.

A sensitivity test has been carried out in order to find the values of degradation which reduce the probability of not fulfilling the constraint. The results of this test are stored in Table 4.4.

L/D	MWE	SFC	Range	Climb	Cruise	TOFL	Vapp
-1	+1	+1	0.41	1	1	0.73	0.98
-2	+1	+1	0.41	1	1	0.74	0.98
-1	+2	+1	0.41	1	1	0.84	0.98
-1	+1	+2	0.41	1	1	0.79	0.98
-2	+3	+2	0.4	1	1	0.95	0.99

Table 4.4: Sensitivity study of TOFL constraint variation regarding MSP degradation values

The variations of degradation values have a great effect on TOFL constraint. Penalizing the MWE parameter risk (i.e. increasing MWE value) leads to a significant increase of the probability of fulfilling TOFL constraint. Nevertheless, the probability of fulfilling the nominal range constraint is not improved. This result can be explained by the fact that the MTOW given by the MSP is too low. Thus, we launch again steps 4 and 5 of the strategy but with a higher MTOW value than the one given in by MSP. We operate an 1% increasing. The results

obtained are in Table 4.5.

Constraint Name	Probability of satisfaction
Range	0.42
TOFL	0.57
Climb	1
Cruise	1
Vapp	0.98

Table 4.5: Result of MSP robustness study with a higher MTOW value

The results show that increasing MTOW value helps in decreasing the probability to miss the constraint on range but, at the opposite, it has a negative effect on the probability to fulfil TOFL constraint: both constraints are acting in opposite ways. This situation seems to reveal an impossibility to fulfill all the constraints at the same time.

4.3.2.2 Propagation with moment propagation method

We carry out another uncertainty propagation to evaluate the robustness of the MSP result, but this time, with the moment propagation method. The results of the analysis are given in table 4.6. The results are slightly different from those found with Monte Carlo method.

Constraint Name	Monte Carlo	Moment propagation
Range	0.41	0.35
TOFL	0.73	0.67
Climb	1	0.82
Cruise	1	0.79
Vapp	0.98	0.79

Table 4.6: Result of MSP robustness study with the moment propagation method

Nevertheless, the same pattern emerges: Range and TOFL are the less protected constraints and the protection on Range does not exceed 50%. The significant difference between both methods is the computation time. The execution of Monte Carlo with a sample size equal to 500 took around 300 seconds while the execution with the moment propagation method took around 10 seconds.

4.3.3 Conclusion

The analysis of the MSP robustness with the two methods reveals that the process does not offer a perfect protection. Most of the constraints seem to be well protected but the method does not protect TOFL enough, which is the most threatening constraint and it does not ensure the nominal range for a large number of possible occurrences. Moreover, we have seen that TOFL and Range constraints react inversely versus an increase of MTOW value coming out of MSP. Even more, TOFL constraint is much more sensitive than the Range one. Furthermore, the results also show that combining different assumptions for degradations can protect efficiently against TOFL constraint but has no significant effect on Range. This last remark requires to be further investigated. Besides, the method does not explain how to assess the different degradation coefficient. Finally, it does not take into account the threat level of each constraint.

Figure 4.10 illustrates the threat level of constraints: in case of uncertainty, some the constraints threat levels of are not the same. Consequently, it is interesting to have a method which
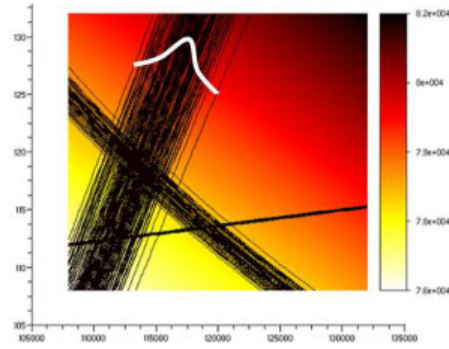
Figure 4.10: Illustration of the constraints threat level

allows us to enforce more protection on threatening constraints. For all these reasons, we have decided to experiments technics of robust optimisation.

Finally, the study conducted in this section has revealed the effectiveness of the moment propagation method compared to the Monte Carlo method.

4.4 Other example of use of moment propagation method

4.4.1 Illustration of the Design Freedom

We can illustrate the section 3.6 *Design freedom* page 92 using the Breguet-Leduc equation presented in chapter 1 (see Eq 1.1). As aforementioned, this relation is a simple but fundamental model of air transport. It makes the link between the possible range that an airplane can fly and the necessary amount of fuel.

$$Range = \frac{1}{g} \cdot \frac{L}{D} \cdot \frac{Mach \cdot Vsnd}{SFC} \cdot \ln\left(1 + \frac{FuelBurn}{OWE + Pl}\right) \tag{4.26}$$

where:

- *Range* is the distance that can be covered by the airplane (m),

- *Mach* is the cruise Mach number,

- *Vsnd* is the sound speed (m/s),

- *g* is the Gravity acceleration (m/s^2),

- *Pl* is the passenger weight (payload) (kg),

- *OWE* is the weight of the airplane without passengers and without fuel (kg),

- *SFC* is the Specific Fuel Consumption (kg/s/N),

- $\frac{L}{D}$ is the Lift over Drag ratio (no dimension) (further denoted *L/D*),

- *FuelBurn* is the quantity of fuel burnt by the aircraft during a given mission (kg).

This relation combines the influence of the three main drivers of air transport efficiency that are *L/D*, *SFC* and *OWE*. At a given point in project time line, the values of *L/D*, *SFC* and *OWE* are attached to a certain amount of uncertainty that can be estimated. The following values are typical of the amount of uncertainty that can be encountered within the first half of project development of a short range aircraft:

- $\epsilon_{L/D} = $ BmystLaw(x=19, a=-0.04, b=0.005, Z=0.75, P=1),

- $\epsilon_{SFC} = $ BmystLaw(x=1.68·10^{-5}, a=-1·10^{-7}, b=2·10^{-7}, Z=0.25, P=1),

- $\epsilon_{OWE} = $ BmystLaw(x=45000, a=-200, b=1500, Z=0, P=1).

A possible targeted maximum range for a modern short range aircraft could be 4000 NM. Additionally, the following characteristic values of cruise conditions have been used:

- $g = 9.806\ m/s^2$,

- $Mach = 0.76$,

- $Vsnd = 296.53\ m/s$,

- $PL = 11000$ kg,

- $FuelBurn = 19000$ kg.

With these assumptions, we obtain the distribution law displayed in Figure 4.11 for the Range that can be covered by the airplane:
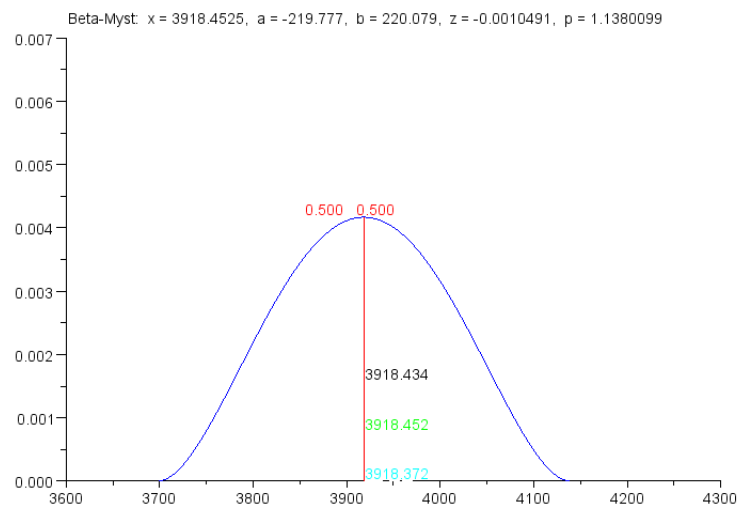
Figure 4.11: Distribution for the Range that can be covered by the airplane.

By analyzing this output, we found that:

- the probability that the Range become higher or equal to 4000 NM is 18.8%,

- the range value which probability to be achieved is 80% is 3840 NM.

These values are related to a level of difficulty of 0.5 which corresponds to a nominal (medium) investment in term of design effort.

If we assume more investment in the design effort corresponding to a level of difficulty of 0.7, uncertainty on efficiency drivers is modified as shown by the graphs in Figure 4.12. Blue curves correspond to nominal design effort while red curves correspond to an index of design effort of 0.7:

By analyzing the corresponding output, we found that:

- the probability that the Range become higher or equal to 4000 NM is now: 63.7%,
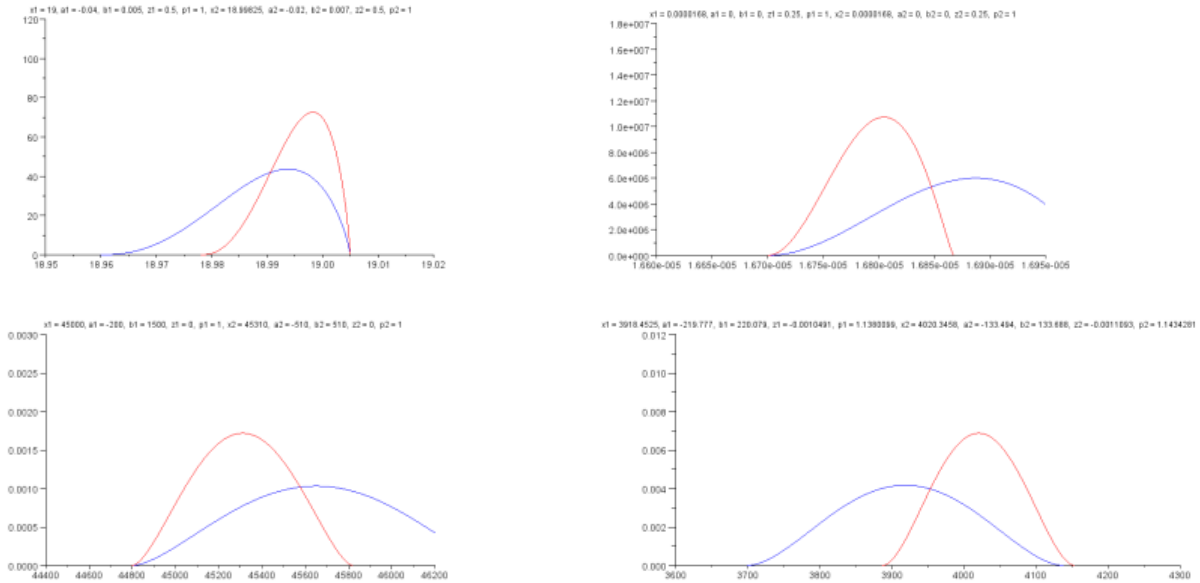
Figure 4.12: Modification of uncertainty on efficiency drivers: Blue curves correspond to nominal design effort and red curves correspond to an index of design effort of 0.7.

- the range value which probability to be achieved is 80% becomes: 3973 NM.

We can see that the probability to reach the targeted maximum range has greatly increased when the design effort has gone from 0.5 to 0.7. An effort a little bit higher than 0.7 would achieve it with a probability of 80%. Of course this method does not tell what the cost of such an effort is. This has to be figured out from deeper industrial analysis. At this stage, the method gives a link between the design effort (not yet scaled) and the achievable performance of the product inside the validity domain that appears according to the existing aircraft.

4.5 Conclusion

The aim of this chapter was to present the propagation methods that we used in the study. We focused on two methods: Monte Carlo sampling and moment propagation methods.

The moment propagation method has been put in place: we have overloaded the basic algebraic operators, the optimizer and the system solvers. We have a way to store the uncertainty measured with the Beta-Mystique distribution into a generic format. Moreover, we have put in place the necessary tools in order to propagate the uncertainty through simulation models.

A robustness study of the Margin Setting Process (MSP) has been carried out thanks to Monte Carlo and moment propagations. Both methods have highlighted the weakness of the MSP: the method does not offer the best possible protection. Moreover, moment propagation method turns up to be much more effective than the Monte Carlo method in terms of computation time.

Finally, in the last part of this chapter, we have presented two examples in which moment propagation method has been used to:

- transform the Beta Mystique parameters into moments (mean, variance, skewness and kurtosis) and conversely. That allows to obtain the Beta-Mystique parameters when having only the moments of a sample of data;

- understand the effect of uncertainty affecting the three pillar parameters (L/D, SFC and MWE) on the range performance.

Chapter 5

Robust Optimisation in Aircraft Design

Contents

The study carried out in the previous chapter has brought out the weakness of the protection offered by the Margin Setting Process (MSP) against risk. The solution obtained with MSP integrates margins but can lead very easily, according to uncertainties, to aircraft configurations which do not fulfil all the operational requirements. That leads us to the conclusion that the MSP is not the more efficient method to use to design aircraft configuration that are protected against risk.

In chapter 2, we have introduced the field of robust optimisation. Robust optimisation methods take into account uncertainties during the process in order to produce solutions which are insensitive to this uncertainty. These methods are new for future project engineers and none of them are implemented or used currently. The use of such methods will help to generate aircraft configurations which are robust according to identified sources of uncertainty.

The aim of this chapter is to present the methodology put in place to achieve robust aircraft design. For that purpose, we display hereafter the steps to formulate and solve a robust optimisation problem in the context of overall aircraft design. First, a robustness measure is selected, then the mathematical formulation of the robust aircraft optimisation is made explicit, and finally the method used to solve that robust optimisation problem is presented.

5.1   Modelisation of the robust aircraft design problem

The modelisation of robust optimisation problem is an important step. Indeed, the result of the optimisation will depend on it. As introduced in chapter 2, we proceed in two steps to model a robust problem: first we select a robustness measure, then we select the formulation of the robust optimisation problem itself (integrating the robustness measure selected in the first step).

5.1.1   Choice of a robustness measure

The choice of the robustness measure is crucial. The result of the study relies on it and may be different according to the measure used. This choice is guided by the type of information available such as:

- the framework used to quantify the uncertainty,

- the decision-maker expectation about the robust solution,

- the constraints on the computation time.

Overall aircraft design problem has been presented in chapter 1. An aircraft configuration is said to be optimal with respect to a set of design assumptions, if it is solution of the deterministic optimisation problem 5.1:

$$\min_{x} \quad MTOW(x)$$
$$\text{s.t.} \quad g_j(x) \leq 0, \quad j = 1, \ldots, n \tag{5.1}$$

where $x = (WingArea, SLSThrust)$, the $g_j(.)$ refer to the operational constraints that the aircraft have to fulfil and n is the number of operational constraints. The fulfilment of those constraints is the very first difficulty in aircraft design processes since some parts of the models are just not computable if some related constraints are not satisfied. Afterwards, the capability of an airplane to fulfil these constraints is later converted into Guaranty of Performances for the Airliners. Important effort is done to guarantee their feasibility so the robustness measure considered should underscore this important aspect. Computation time is also an important issue at this stage: the used methods have to be computationally cheap in order to allow a high number of design studies. Then, the robustness measure should be, computationally, as cheap as possible. At last, the robustness measure should be adapted to uncertainty quantified with

probabilistic tools.

Three categories of measures have been presented in chapter 2: deterministic, probabilistic and possibilistic. Within these categories, probabilistic measures seem to be the most adapted to overall aircraft design problem.

In this study, two probabilistic measures are considered:

- expectation measure of robustness on the objective function (the objective function is also represented by a robustness measure because it is also subject to uncertainty);

- statistical feasibility robustness measure applied on constraints (chance constraints).

The measure proposed for constraints is the probability $\mathbb{P}_G$ of satisfying all constraints (see Eq. 5.2):

$$\mathbb{P}_g = \mathbb{P}\left[g_j(x, \epsilon) \leq 0, j = 1, \ldots, n\right] \geq P_0, \tag{5.2}$$

where ϵ is a random variable representing the uncertainty on aircraft model error. This probability is called *joint chance constraint* because the probability is calculated for the event *all the constraints are satisfied simultaneously*.

The measure proposed on the objective function is the expectation measure (Eq. 5.3):

$$R_f(x, \epsilon) = E[f(x, \epsilon)]. \tag{5.3}$$

5.1.2   Formulation of the robustness problem

Once the robustness measures are selected, the next task is the formulation of the robust optimisation problem to solve. The robustness measures have to appear in this formulation. Here again, the choice of a formulation depends on the characteristics of the problem and on the information available. In this study, the strategy adopted to transform equation (5.1) is to replace the objective function and the constraints by their robustness measures. Thus, the formulation proposed for the robust aircraft design optimisation is equation (5.4):

$$\begin{aligned} \min_{x} \quad & E(f(x, \epsilon)) \\ \text{s.t.} \quad & \mathbb{P}\left[g_j(x, \epsilon) \leq 0, j = 1, \ldots, n\right] \geq P_0, \end{aligned} \tag{5.4}$$

where $x = (WingArea, SLSThrust)$ and $P_0 \in [0, 1]$ is the confidence level fixed by the decision-maker. This formulation means that we look for a couple of values of *WingArea* and *SLSThrust* that gives an aircraft configuration for which the constraints are satisfied with a probability P_0 at least.

In equation (5.4), the chance constraint considered is called a joint chance constraint. There exists another type of chance constraint: the individual chance constraint. In the latter, the confidence level P_0 may be not the same for each constraint. It is useful because sometimes all constraints do not have the same criticality. For example, one could ask for an aircraft design for which the constraint on Take-Off Field Length is fulfilled at least 95% and the constraint on Approach Speed is satisfied only at least 50%. The formulation of the robust problem with individual chance constraints is as in equation (5.6):

$$\min_{x} \quad E(f(x, \epsilon)) \tag{5.5}$$

$$\text{s.t.} \quad \mathbb{P}[g_j(x, \epsilon) \leq 0] \geq P_{0_j}, \quad j = 1, \ldots, n \tag{5.6}$$

where $x = (WingArea, SLSThrust)$.

Equation (5.6) is not the final expression of the robust aircraft design problem to calculate. An information is missing.

Consider again the deterministic optimisation problem in equation (5.1) associated to figure 1.11 page 16. In this optimisation problem, the design variables are *WingArea* and *SLSThrust* and the criterion to minimise is *MTOW*. Actually, there is a third design variable which is hidden in the Mass-Mission loop: it corresponds to the *MTOW* used as hypothesis for structural design. Consider figure 1.13 page 17 which focuses on the mass-mission loop step. As explained, to initiate the mass-mission loop, an assumption on *MTOW* value is taken. Moreover, once the complete geometry of the aircraft is available, as well as aerodynamic forces and structural mass, the mission of the aircraft is simulated and a new value of *MTOW* is obtained. To differentiate the two variables and to avoid misunderstanding, the assumed *MTOW* value is denoted $MTOW_{DV}$ (as design variable) and the value obtained after mission simulation is denoted $MTOW_{OF}$ (as objective function). What does not appear in figure 1.13, is that $MTOW_{OF}$ may be lower than (or at most equal to) $MTOW_{DV}$. And the objective of the mass-mission loop is to ensure that this inequality is verified. Thus, if $MTOW_{OF}$ is higher than $MTOW_{DV}$, it is recalculated so that its value becomes equal to $MTOW_{DV}$. As $MTOW_{DV}$ is an input of the structural design sub-process, it could be considered as a design variable of the whole optimisation. Thus, the new formulation of the deterministic aircraft design problem is :

$$\min_x \quad \mathbf{MTOW_{OF}(x)}$$
$$\text{s.t.} \qquad g_j(x) \leq 0, \qquad j = 1, \ldots, n \tag{5.7}$$

where $\mathbf{x} = (\mathbf{WingArea}, \mathbf{SLSThrust}, \mathbf{MTOW_{DV}})$. With these new variables, figure 1.11 page 16 is transformed and becomes figure 5.1.
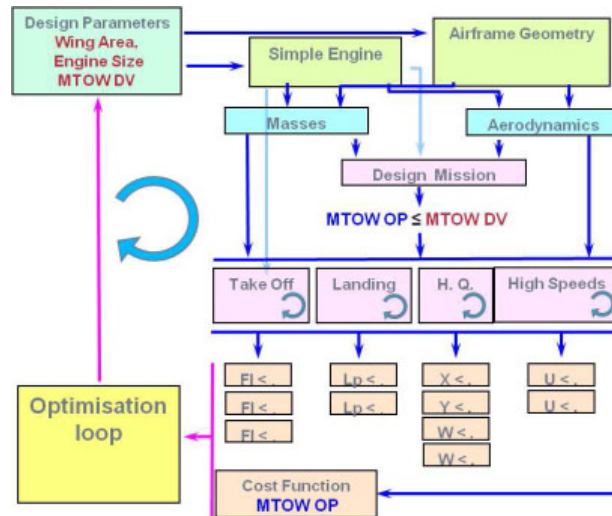
Figure 5.1: Overall Aircraft Design Process with a third design variable

Finally, regarding equations (5.6) and (5.7), the robust formulation proposed is (eq. 5.8):

$$\min_x \qquad E(f(x, \epsilon))$$
$$\text{s.t.} \quad \mathbb{P}[g_j(x, \epsilon) \leq 0] \geq P_{0_j}, \quad j = 1, \ldots, n \tag{5.8}$$

where $x = (WingArea, SLSThrust, MTOW_{DV})$.

5.2 Strategy to compute the robustness measures

Two classes of approaches have been presented in the state-of-the-art (chapter 2) to compute the robustness mesures P_g and R_f:

- Deterministic approaches corresponding to methods that compute the robustness measures by using numerical methods;

- Randomised approaches referring to methods that use directly the values of the objective and which do not need the derivative information.

In aircraft design, the derivative information is not available at any stage of the design. Indeed, it is possible to have access to it at overall stage but it is quasi impossible at detailed stage.

In this study, we will compare two strategies:

- the first one is to build surrogate models of the robustness measures then to use them for the optimisation;

- the second one is to directly use moment propagation method to solve the problem.

Both strategies offer the advantage not to focus on the computation of the robustness measure during the optimisation. Moreover, in the first case, the uncertainty is involved only during the design of the surrogate models and not anymore during the optimisation. Other methods such as FORM or SORM are able to directly realise estimations in the range which is in concern.

In the first strategy, two levels of surrogate models are generated: the first level contains surrogate models of the criterion and the operational performances and the second level contains surrogate models of the feasibility probability. The process launched to generate these surrogate models is the following (cf. figure 5.2):

- **STEP 1:** Generate n vectors of size m of sample of model errors. That gives a matrix ϵ of size $n \times m$:

$$\epsilon = \begin{pmatrix} \epsilon_1 \\ \vdots \\ \epsilon_n \end{pmatrix} = \begin{pmatrix} \epsilon_{11}, \ldots, \epsilon_{1m} \\ \vdots, \ldots, \vdots \\ \epsilon_{n1}, \ldots, \epsilon_{nm} \end{pmatrix} \tag{5.9}$$

  Generate a design of experiments (DOE) of size $q \times p$ of the design parameters ($WingArea$, $SLSThrust$, $MTOW_{DV}$), where $q = 3$ is the number of design parameters and p is the number of samples of each design parameter. The DOE is built thanks to Latin Hypercube Sampling (LHS) method.

- **STEP 2:** For each point of the DOE, launch a linear aircraft design analysis by propagating uncertainty stored in vector ϵ_1 (first line of ϵ). That gives a set of p aircraft configurations. With these p aircraft configurations, we are able to compute response surfaces for each operational constraint and also for the criterion.

  Step 2 is repeated n times, where n is the number of vector of model errors. We obtain n response surfaces for each operational constraint and n response surfaces for the criterion

- **STEP 3:** Using the surrogate model generated in step 2, we compute, for each constraint, the frequencies of aircraft that fulfil requirements. With these frequencies, we are able to build response surfaces representing the feasibility probability of each operational constraint.

This approach in two steps allows the user to change the value of the operational requirements and to recalculate the feasibility probabilities (step 2) without having to recalculate the performances (step 2). Changing operational constraints is sometimes performed when engineers want to understand the sensitivity of the design point to the set of Top Level Requirements itself which can lead to reconsider the requirements. Nevertheless, this method presents one drawback: the two levels of surrogate models introduce two levels of modelling errors which can be difficult to manage. But it is not limitative.
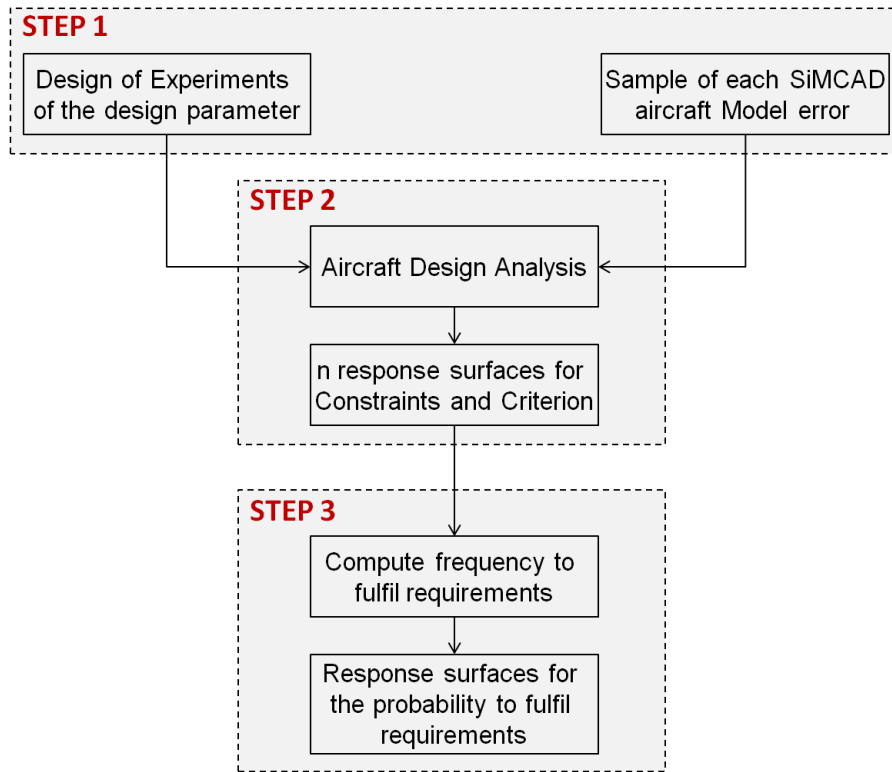
Figure 5.2: First strategy: compute robustness measures with surrogate models

5.3   Algorithm for the resolution

Once the robustness measures have been computed with the first strategy, they are used in the optimisation process in order to compute the robust solutions. For that purpose, an efficient optimisation solver has to be selected. This selection is done according to the characteristics of the problem being solved such as the nature of the objective function, constraints, number of design variables or the decision-maker expectation. Several solvers can be retained for solving the same problem. In this case, the final choice can be done regarding to the solver execution time.

In the case of robust overall aircraft design, the solver should be adapted to mono or multi-objective constraints optimisation problems in which function evaluations are fast because of the use of surrogate models to retrieve constraints and criteria.

Figure 2.2 page 42 presents the two families of deterministic optimisation methods. The study is oriented toward derivative-free optimisation algorithms. Three methods have been taken into consideration:

- Nonlinear Simplex with penalty method for constraints (Nelder-Mead)

- Genetic Algorithm (GA) with Non dominated Sorting Genetic Algorithm (NSGA) for multi-objective optimisation

- Mesh Adaptive Direct Search (MADS)

In this study, we have used the Nonlinear Simplex with penalty method for constraints for the mono-objective optimisation and the Genetic Algorithm with NSGA for the multi-objective optimisation.

5.4 The test case

Table 5.1 shows the specification of the test case. We want to design a twin jet aircraft, able to carry 180 passengers, and to reach a range of 3000 nautical miles. Moreover, this aircraft should have for example an approach speed lower than 130 kt and a Take-Off Field Length lower than 1800 meters. The criterion to minimise is either the Cash Operating Cost (COC).

Design Assumptions	Definition/Unit	Value
WingAr	Wing Aspect Ratio	9
BPR	By Pass Ratio	10
Range	Design Range (NM)	3000
Mn	Cruise Mach number (mach)	0.76
Ne	Number of engine	2
Npax	Number of passenger	180
Alt_{Ref}	Reference altitude (ft)	35000
Requirements	Definition	Value
Range	Design Range (NM)	≥ 3000
TOFL	Take Off Field Length (m)	≤ 1800
ClVz	Climb vertical speed (ft/min)	≥ 500
CrVz	Cruise vertical speed (ft/min)	≥ 300
Vapp	Approach speed (kt)	≤ 130

Table 5.1: Design specifications

5.5 Deterministic aircraft design optimisation

The deterministic optimisation (cf. Eq. (5.7) page 120) is first launched. The results will be considered as a baseline for comparison. They are stored in table 5.2.

Design Variables	Value
Wing Area (m^2)	140
Engine size (N)	120910
$MTOW_{DV}$ (kg)	76306
Criterion/Constraints	
COC (\$/trip)	4259
Range (NM)	3000
TOFL (m)	1799
ClVz (ft/min)	597
CrVz (ft/min)	400
Vapp (kt)	129

Table 5.2: Deterministic optimisation results

In this example, the active constraints are Take-Off Field Length (orange line) and Approach Speed (yellow line). They are displayed in Figure 5.3.

The optimal aircraft design is located at the intersection of the two active constraints. The robustness of this design is analysed by using a moment propagation method. In this study, we consider as uncertain the same three variables presented in chapter 1 when describing the margin assessment process: *L/D*, *SFC* and *MWE*. The among of uncertainties propagated is
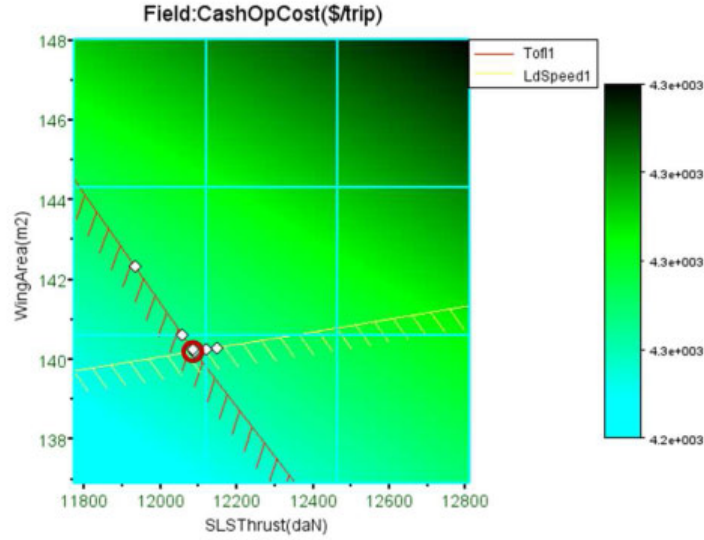
Figure 5.3: Deterministic optimisation result

fixed according to expert judgement, through Beta Mystique distribution. We consider: $\epsilon_{L/D} \sim$ bmyst(1,-0.02,0.01,0.33,1), $\epsilon_{SFC} \sim$ bmyst(1,-0.01,0.02,-0.33,1) and $\epsilon_{MWE} \sim$ bmyst(1,-0.01,0.02,-0.33,1). Figures 5.4 to 5.6 present these uncertainties.
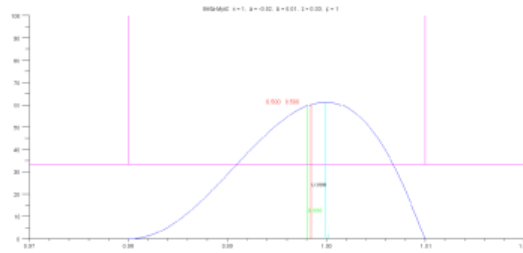
Figure 5.4: Uncertainty on LOD
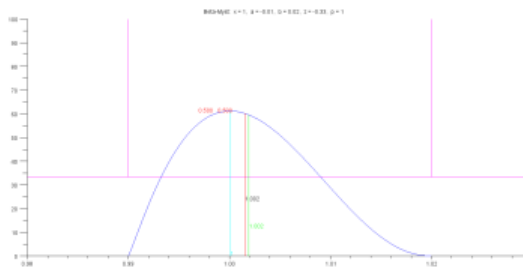
Figure 5.5: Uncertainty on SFC

Table 5.3 displays the results of the analysis.

With this aircraft configuration, requirement on *Range* is 51% of the times fulfilled, the one on *TOFL* is 33% of the times fulfilled, and the one on *Vapp* is 40% of times satisfied. The probability on range and approach speed constraints are very low. It is not acceptable because the fulfillment of the constraint on range is crucial.
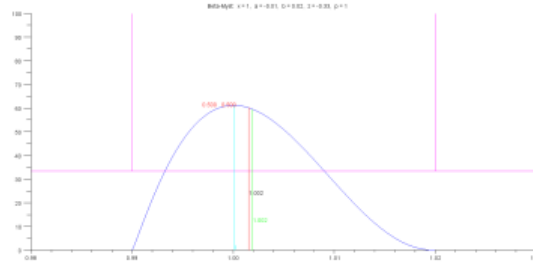
Figure 5.6: Uncertainty on MWE

Constraint names	Probability of satisfaction
Range (NM)	0.51
TOFL (m)	0.33
ClVz (ft/min)	1
CrVz (ft/min)	1
Vapp (kt)	0.40

Table 5.3: Probabilistic analysis of the deterministic design with the three uncertain variables *L/D*, *SFC* and *MWE* by using moment propagation method

5.6 Chance constrained optimisation

5.6.1 Mono-objective optimisation

The chance constrained optimisation (cf. Eq. (5.8)) has been solved thanks to the Simplex algorithm. We compare the results of the strategy where the robustness measures are represented by surrogate models (further denoted SurrMod), with the results of the strategy based on moment propagation (further denoted MomProp). The confident levels P_0 considered are displayed in table 5.4.

Probability of safety	Confident level P_0
$\mathbb{P}(\text{Range} \geq 3000)$	0.85
$\mathbb{P}(\text{TOFL} \leq 1800)$	0.95
$\mathbb{P}(\text{ClVz} \geq 500)$	0.75
$\mathbb{P}(\text{CrVz} \geq 300)$	0.95
$\mathbb{P}(\text{Vapp} \leq 135)$	0.75

Table 5.4: Description of the Probability of reliability and the level of confidence P_0

The result of the deterministic optimisation is taken as a starting point of the chance constrained optimisation.

In the following, we carry two mono-objective studies: one with low amount of uncertainties and the other one with high amount of uncertainties. In the first case, only the three variables *L/D*, *SFC* and *MWE* are considered as uncertain. In the second case, we introduce all the uncertainty quantified in chapter 3 for all SiMCAD models. The aim is to compare the aircraft configuration proposed in both cases.

5.6.1.1 Low amount of uncertainties

Table 5.5 provides the results obtained with the robust optimisation according to the confidence levels P_0 and according to the two strategies SurrMod and MomProp, in the case with low

amount of uncertainties.

Design Variables	RD with SurrMod	RD with MomProp
Wing Area (m^2)	149	141
Engine size (N)	124940	123190
$MTOW_{DV}$ (kg)	79349	77119
Criterion/constraints		
COC (\$/trip)	4298	4275
$\mathbb{P}$(Range$\geq$ 3000)	0.85	0.85
$\mathbb{P}$(TOFL $\leq$ 1800)	0.95	0.95
$\mathbb{P}$(ClVz $\geq$ 500)	0.96	1
$\mathbb{P}$(CrVz $\geq$ 300)	0.96	1
$\mathbb{P}$(Vapp $\leq$ 135)	0.99	0.75
Computation time (s)	4	400

Table 5.5: Robust Mono-Objective optimisation results with surrogate models and moment propagation strategies in the case of low amount of uncertainties

The resulting robust aircraft designs are heavier than the deterministic design for both SurrMod and MomProp methods. The robust configurations proposed by the Surrogate Models and moment propagation strategies are respectively: 4% and 1% heavier with a 6% and 1% bigger wing area and a 3% and 2% bigger engine size than the configuration obtained with the deterministic approach. At the end, that will cost less than 1% more on both to guarantee the satisfaction of the requirements at the confident level P_0. With these results, we are able to apprehend the margins that robust optimisation bring to the design.

Furthermore, when focusing on the computation time, Moment Propagation method seems to be longer than the Surrogate Models strategy. This is only possible because the step of building the two stages of surrogate models have been done upstream. In fact, when considering all the steps, Surrogate Models strategy turns to be longer than Moment Propagation strategy.

By using a Monte Carlo sampling (50000 points), we can check the validaty of the results obtained in Table 5.5 with the robust optimisations. The results of the analysis are presented in Tables 5.6 and 5.7.

Constraint names	Required	RD with SM	Monte carlo Probabilistic analysis
Range (NM)	**0.85**	**0.85**	1
TOFL (m)	0.95	0.95	1
ClVz (ft/min)	0.75	0.96	1
CrVz (ft/min)	0.95	0.96	1
Vapp (kt)	**0.75**	**0.99**	1

Table 5.6: Probabilistic analysis of the robust design obtained with surrogate models strategy, with the three uncertain variables L/D, SFC and MWE by using moment propagation analysis

The active constraints for both optimisations are the Range and the Approach Speed (*Vapp*). If we focus on the probability of satisfaction for these constraints, we see that, with the moment propagation method, the probabilistic analysis gives exactly the required value.

5.6.1.2   High amount of uncertainty

We launched another chance-constrained optimisation by taking into account all the uncertainty quantified for all SiMCAD models. The required levels of confidence are kept identical than in

Constraint names	Required	RD with MP	Monte carlo Probabilistic analysis
Range (NM)	**0.85**	**0.85**	**0.84**
TOFL (m)	0.95	0.95	1
ClVz (ft/min)	0.75	1	1
CrVz (ft/min)	0.95	1	1
Vapp (kt)	**0.75**	**0.75**	**0.75**

Table 5.7: Probabilistic analysis of the robust design obtained with moment propagation strategy, with the three uncertain variables L/D, SFC and MWE by using moment propagation analysis

Table 5.4. To solve the problem, we used only the moment propagation method because the strategy based on the Monte Carlo method would have required billions of computations to reach the same level of precision. The results are stored in Table 5.8.

Design Variables	RD with MomProp
Wing Area (m^2)	165
Engine size (N)	173300
$MTOW_{DV}$ (kg)	90489
Criterion/constraints	
COC (\$/trip)	4788
$\mathbb{P}(\text{Range} \geq 3000)$	0.85
$\mathbb{P}(\text{TOFL} \leq 1800)$	0.98
$\mathbb{P}(\text{ClVz} \geq 500)$	0.95
$\mathbb{P}(\text{CrVz} \geq 300)$	0.95
$\mathbb{P}(\text{Vapp} \leq 135)$	0.75

Table 5.8: Robust Mono-Objective optimisation results with moment propagation strategy in the case of full model uncertainties

The aircraft configuration solution, in this case with high amount of uncertainty, is much larger than the other configuration proposed so far in Table 5.5 (with the moment propagation method).

From the designer and the airliner point of view, these results are not manageable: on one hand this aircraft would be too costly and on the other hand it would not be economically attractive for the airliners. An aircraft design expert would say that a lighter aircraft could be made for sure that would achieve the same requirements. In other terms: such an aircraft do not exploit really the possibilities of the technology. This situation is the result of the way models has been calibrated by minimising the least square error versus the aircraft database. Doing so, the level of technical skills captured by the model corresponds to an average of what can be actually done. We would want to have a way to tune this implicit technical skill incorporated into the models. The Design Freedom Parameter introduced in chapter 3 can play this role. Of course, it would be interesting to have a quantitative relation between the level of skill and the cost of the associated design, indeed, a high level of skill is most costly than an average one. Such a relation would allow to incorporate this cost in the optimisation criteria.

5.6.2 Multi-objective optimisation

Engineers are also interested in the case of optimisation problem where several objectives are involved i.e. in multi-objective optimisation. For example, one would like to minimise the

Maximum Take-Off Weight (MTOW) while maximising the range that the aircraft should cover. This deterministic multi-objective optimisation problem is formulated as follows:

$$\min_{x} \quad (MTOW_{OF}(x), -Range(x))$$
$$\text{s.t.} \qquad g_j(x) \leq 0, \qquad\qquad j = 1, \ldots, n \qquad (5.10)$$

where $x = (WingArea, SLSThrust, MTOW_{DV})$.

The formulation proposed for the robust multi-objective optimisation is as in Eq. (5.11):

$$\min_{x} \quad (E(f_1(x, \epsilon)), \ldots, E(f_l(x, \epsilon)))$$
$$\text{s.t.} \qquad \mathbb{P}[g_j(x, \epsilon) \leq 0] \geq P_{0_j}, \qquad j = 1, \ldots, n \qquad (5.11)$$

where $x = (WingArea, SLSThrust, MTOW_{DV})$ and l is the number of objective functions. Its resolution is carried out thanks to the Genetical Algorithm NSGA.

The first numerical experiment is to solve the bi-objective optimisation which confronts a cost criterion and one representing the level of robustness. The criterion representing the level of robustness is the probability not to fulfil all operational constraints at the same time, noted nJP. This problem has been solved thanks to the NSGA algorithm set up with a population size equal to 800, a number of children by generation equal to 800 and a maximum number of generation equal to 500. The result of the bi-objective optimisation is Figure 5.7 showing the Pareto Front. We recall that the Pareto Front is the set of points representing the compromises for which it is not possible to improve one objective without deteriorating another one.
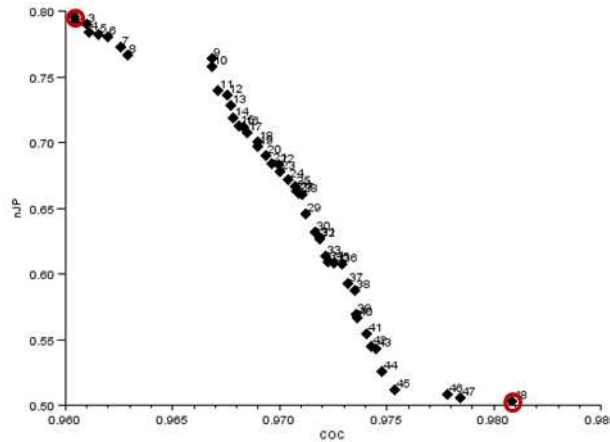
Figure 5.7: Pareto Front

All the points on the Pareto front are solutions of the bi-objective optimisation problem. The two extreme points of the Pareto Front are reported in Table 5.9.

Min Objective	COC	nJP
COC($/trip)	4802	4904
nJP (%)	79	53
Wing Area (m^2)	144	161
Engine size (N)	136421	146333
$MTOW_{DV}$ (kg)	78797	84985

Table 5.9: Result of the robust bi-objective optimisation COC-nJP
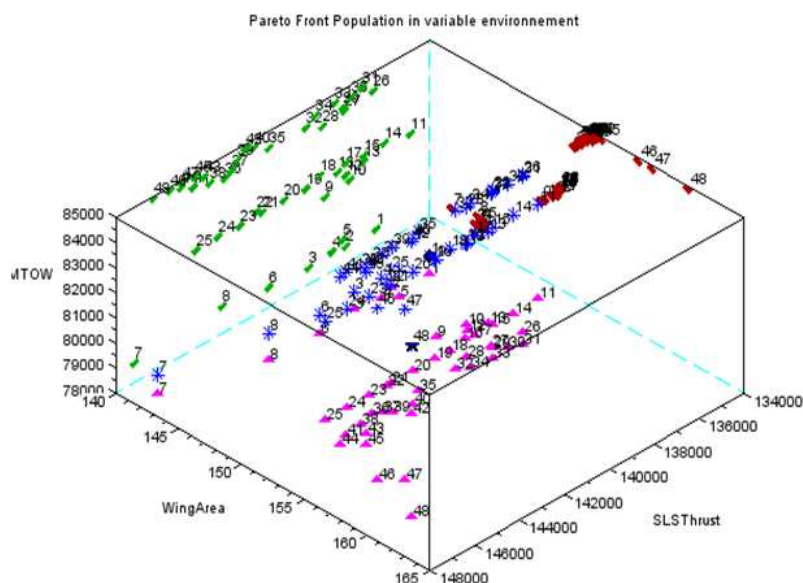
Figure 5.8: Pareto Front population in variable environment

Thanks to the Pareto Front, we are able to visualise the trade-off between cost and robustness. Indeed, it costs more to build an aircraft with high level of robustness (meaning with a low probability not to fulfil all constraints together) than one with a low level of robustness.

The selection of a given solution in the Pareto Front is not necessarily done by the engineer as some strategic and political issues may place the decision at higher hierarchical level. Sometimes, the task is not easy because of the high number of solutions especially when more than 2 criteria are involved. Methods have been developed in order to help the user in this task. More details can be found in [Bau12].

5.7 Conclusion

The aim of this chapter was to present the methodologies proposed to handle robust aircraft design by the way of chance constrained optimisation. The test case is based on the toolbox of Scilab codes named SiMCAD presented in chapter 1.

For comparison sake, we have proposed two strategies to solve it:

- the first strategy was to approximate the feasibility probabilities with surrogate models and to solve the optimisation problem with the surrogate models (in place of the real probabilities);

- use moment propagation method to get out directly output probabilities and solve the optimisation problem.

The resolution of the robust optimisation problem has been done thanks to the Simplex for mono-objective optimisation and thanks to the Genetical Algorithm NGSA for multi-objective optimisation. The different experiments carried out have first shown the advantages of chance constrained method compare with deterministic one. The results obtained help in understanding the way the margins are bring to the design in a robust optimisation problem. This study has also highlighted the weakness and the potential of each of the experimented strategies. The moment propagation method turns out to have high potential in comparison with the method using surrogate models.

Several improvements can be made to the current process:

- increase the level of fidelity of the response surfaces used to approximate the probabilistic constraints or use another type of surrogate models such as Kriging or polynomial chaos;

- compute the probability not to fulfil all requirements together using the moment propagation method;

- generate output graphics in order to present from a synthetic point of view the results of the chance constrained optimisation (inspired from the one produced at the end of the deterministic optimisation);

- propose a method that is able to highlight some particular compromises inside the Pareto front on the basis of engineering knowledge. This could help the decision maker to select a final solution.

Conclusion

This last chapter presents the contributions, the conclusions and the perspectives of the thesis. This study was intended to propose a new Design Philosophy based on robust optimisation technique in order to manage and integrate quantitative uncertainty management into the very early phase of aircraft design processes in Airbus. First, we have developed a new generic distribution function to quantify the uncertainty which have been propagated through the simulation code thanks to Monte Carlo and moment propagation methods. Then, we have exhibited a strategy to move from a deterministic aircraft design problem toward a chance constrained optimisation. This work has lead to the creation of a toolbox of Scilab functions which propose mathematical tools covering from automated uncertainty quantification and propagation to the chance constrained optimisation.

Contribution and conclusion

Our first objective was to identify and quantify the source of uncertainty involved at preliminary aircraft design stage. Such objective has been achieved by adopting a probabilistic framework, particularly a parametric estimation.

Uncertainty on error modelling, named predictive uncertainty, has been identified as the main source of uncertainty in aircraft design. It has been quantified using a new generic distribution function called Beta-Mystique distribution that we have specifically built for that purpose. It is a four parameters distribution which has the characteristic to adapt itself on the shape of most of unimodal distributions. The use of a unique distribution function offers the advantage to simplify the quantification process because it is no more necessary to select a family of density function. Thanks to the propagation method, we were able to transform the Beta-Mystique parameters into moments and conversely. Nevertheless, this operation is not always possible in an exact manner and a strategy has been adopted to manage this conversion in all cases, paying a decrease of precision when exact conversion is not possible.

The particular structure of the models computing the aircraft performances has been highlighted. We have defined, implemented and validated a specific strategy, based on an optimisation process, in order to assess the predictive uncertainty of such interlinked models. Important effort has been done to select the more efficient method to solve the problem.

The second objective of the thesis was to select an efficient method for propagating the uncertainty through the aircraft simulation code. Two methods have been selected: Monte Carlo method and moment propagation method. The principle of the moment propagation method has been presented as well as the implementation necessary for its proper integration into the current conceptual aircraft design process. The implementation method has been based on overloading the basic operators, system solvers and local optimization sub-processes. This strategy has been selected because it was judged as the less intrusive regarding the existing model source codes.

A comparison between the two methods has been carried out and the moment propagation method turns out to be more efficient than Monte Carlo method in terms of computation times. Moment propagation method reveals to be very adapted to the context of conceptual design and allows to handle many uncertainty sources without being submitted to combinatorial explosion. Nevertheless, a limitation of the moment propagation has been pointed out regarding its loss of precision in case of important non-linearity. A solution to overcome this drawback has been identified: taking into account the second order derivative information. Further investigation is currently in progress. Both of these methods have been used to analyse the robustness of an existing process used in Airbus to manage the risk called the margin setting process (MSP). The results have revealed the weaknesses of the MSP: poor protection on the most threatening constraints, no explicit explanation on how to assess the degradation values and no consideration of the level of threat of each constraint.

Our last objective was to put in place mathematical tools which allow to design aircraft by mastering the final risks. The selection of the appropriate method to use have been done regarding to the characteristics of the problem. As the feasibility of the constraints is an important issue in aircraft design, we had to select a method which put importance on that point. That led us to focus on the chance constrained optimisation where the constraints are probabilistically guarantee. After exhibiting the path followed to transform the deterministic aircraft design optimisation into a chance constrained optimisation, two strategies have been used to compute the probabilistic constraints: the use of surrogate models for approximating the probabilistic constraints or directly the use of moment propagation method.

The resolution of the optimisation has been done with the Nonlinear Simplex with penalty method for constraints (Nelder-Mead) in the case of mono-objective optimisation and with the Genetic Algorithm (GA) with NSGA in the case of multi-objective optimisation.

To summarise, the contribution of the thesis are:

- automation of a process for calibrating the SiMCAD models on a given aircraft database;

- definition, implementation and validation of a methodology for quantifying the predictive uncertainty of performance models with an interlinked structure;

- automation of a process to identify the predictive uncertainty of all SiMCAD models (with simple or interlinked structure);

- definition, implementation and validation of a generic four parameter distribution function called Beta-Mystique for fitting most of unimodal distributions;

- integration and validation of the algebraic propagation of the moments in the current conceptual aircraft design;

- definition and validation of a numerical strategy to convert the moments into Beta-Mystique parameters and the reversal;

- proposition of an interpretation of the predictive uncertainty, measured according to a database, as a design freedom;

- specification and integration of a chance constrained optimisation based on the strategy of surrogate models and Monte Carlo sampling in conceptual aircraft design;

- specification and integration of a chance constrained optimisation based on moment propagation method in conceptual aircraft design.

Perspectives

Future work can be focus on some points to improve this presented study. The moment propagation method can be improved by taking into account the second order derivative information. In addition to the gradient, the operator overloading can also be developed to be able to propagate the Hessian. Consequently, the coupling terms in the moment propagation formulas will incorporate some more terms. This work has been investigated in the frame of an internship and will be applied in another thesis which has started recently.

The capability to propagate gradient and Hessian through the computational process can also be greatly profitable to solve optimisation problems. Late versions of the SiMCAD toolbox are being adapted to provide Gradient and Hessian to imbedded optimisation methods that can use them. The use of the uncertainty on model error in terms of design freedom should be experimented. This would allow to point out the problematic parts of the design and to mitigate the associated risks by transferring part of the design effort inside the domains which have the best return on investment in term of technical results.

Clearly, the use of linear interpolation as surrogate models to approximate the probabilistic constraints is not satisfactory. Other solutions should be used such as Kriging or Polynomial Chaos in order to reduce (and control) the approximation error.

The possibility to calculate the joint probability (to satisfy all the constraints) thanks to the moment propagation method has to be investigated. The answer is not obvious because of the coupling between all constraints through uncertain inputs parameters. The use of copulas may provide a way to compute this joint probability.

The NSGA algorithm used in this study for solving the multi-objective optimisation problem might be replaced by its successor, the NSGA II algorithm. In the literature it is presented as more efficient than NSGA. Another algorithm might be interested to use: the Mesh Adaptive Direct Search (MADS). It has already been tested in the frame of a thesis on the context of aircraft departure procedure and has given promising results.

All the processes developed are interesting tools in the frame of new technology evaluation. In this type of study, we try to evaluate the economical benefit of using a new technology as fuel cells or electrical engines for instance according to various technical implementation scenario. Indeed, in this context, only very few data are available, some of them being very fuzzy. That implies the use of models with low level of fidelity. Then, the quantitative consideration of the uncertainty allows to sort the different technological scenario according to their credibility, evaluated in term of probability. A study is currently carried out on this subject in the frame of another thesis.

Appendix A

Appendix

A.1 SiMCAD front end

In order to be able to replay all the important steps of this work, a front end has been developed to be able to launch the corresponding scripts. The front end is able to launch various sub-process of the work from the calibration of the models on a given aircraft database to the robust optimisation process including specific illustration of propagation techniques. For each sub-process, some input variables are offered to the user for modification but the list is not exhaustive and any other variables can be added just by modifying the scripts. The aim of this paragraph is to briefly present the use of the SiMCAD front end also called abusively tool box (the words SiMCAD tool box referencing the internal modules of SiMCAD in other parts of this report).

The SiMCAD front end is accessible from a Batch file ('.bat'). This file can be launched from Windows(R) by a double click or from a Disk Operating System (DOS) session. After double clicking on the Batch file, the user can visualise the GUI of SiMCAD front end, presented in Figure A.1.

Figure A.1: SiMCAD Architecture

Several choices are proposed to the user as a list of tabs. To activate his choice, he just has to click on the corresponding tab. In the remainder of this section, we give examples on the

steps to follow to carry out specifics studies such as a deterministic aicraft optimisation or a robust aircraft optimisation.

A.1.1 Deterministic Optimisation

To launch a deterministic aircraft optimisation, the user has to click on the tap 'deterministic studies' then on 'Optimised Design'. Another window opens so that the user can enter its preferences on the requirements the aircraft should fulfil: the total number of passengers, the range, the Cruise Mach number, the reference altitude, the number of engines, the By Pass Ratio, the Wing Aspect Ratio, the Take-Off Field Length and finally the approach speed. The user can visualise the process running (see Figure A.3).
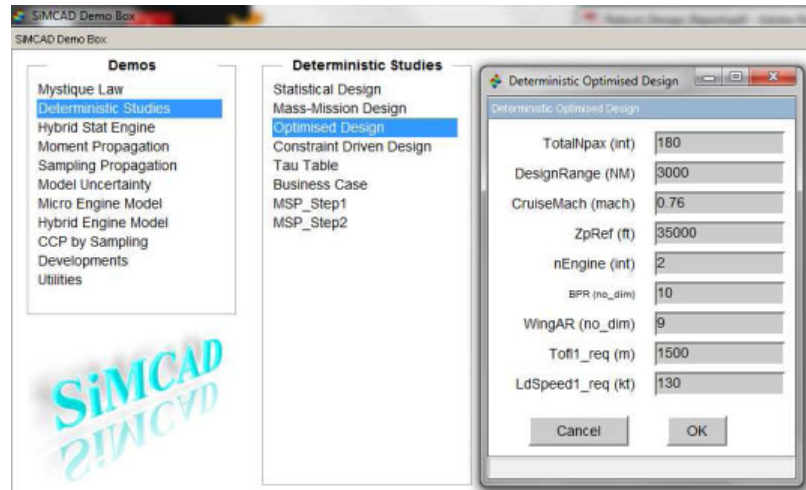
Figure A.2: Deterministic optimisation
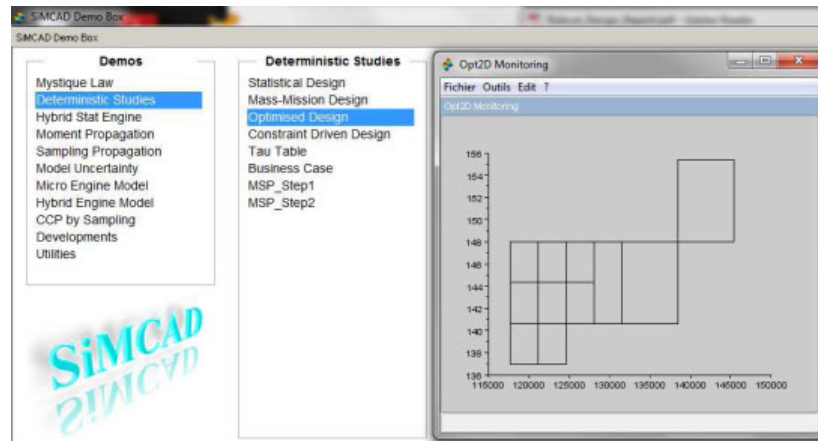
Figure A.3: Deterministic optimisation running

The result of the optimisation is presented to the user through three files: a Scilab file and a text file containing the description of the optimal aircraft configuration and a graphic representing the position of the solution regarding the active constraints.

A.1.2 Uncertainty

The tool box offers several tabs related to uncertainty management. The first tab is related to the Beta-Mystique (Figure A.5). The user has the possibility to display a graphical setting
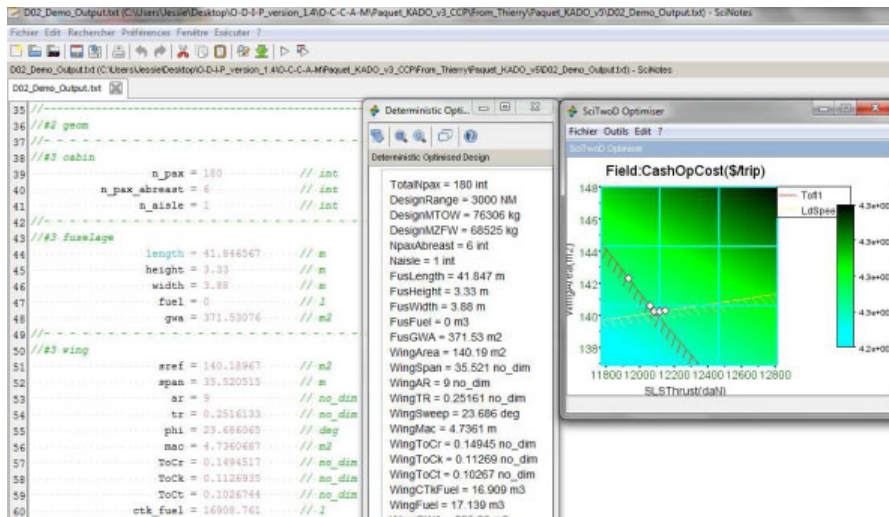
Figure A.4: Results of the deterministic optimisation

of the Beta-Mystique distribution. It can also launch basic operations with this distribution: addition, subtraction, multiplication, division, power, log, and exponential. Figure A.6 presents the addition of two Beta-Mystique distributions.
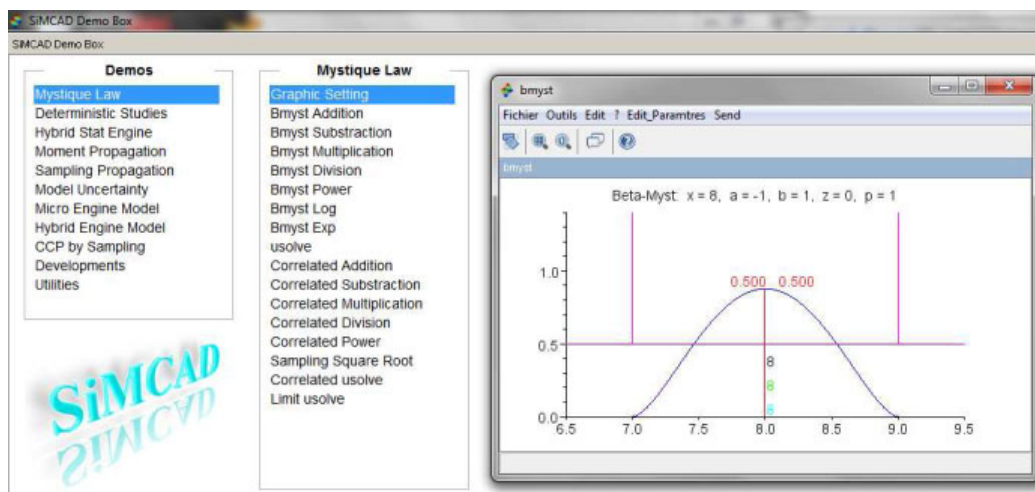
Figure A.5: Beta-Mystique distribution

Uncertainty propagation study can be executed with SiMCAD tool box. Two methods are proposed: the moment propagation method and the Monte Carlo method. The moment propagation method is accessible via the tab 'moment propagation' and the Monte Carlo method under the tab 'sampling propagation'. Based on these propagation methods, we have implemented the whole code sequence for the robustness study of some processes such as the Margin Setting Process (see Figure A.7).

A.1.3 Robust aircraft optimisation

To launch a robust aircraft optimisation, the user has to click on the tab 'CCP by sampling' for Chance Constraint Programming. The user will be offered a list of control to carry out the robust optimisation from the generation of error sampling to the resolution of the optimisation. He will just have to follow the different steps from the first tabs to the optimisation.
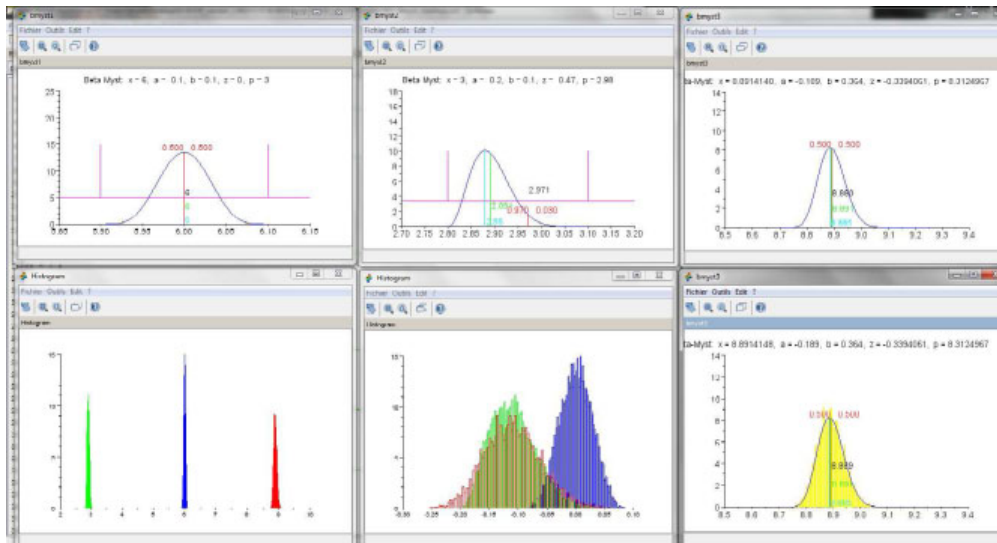
Figure A.6: Addition of two Beta-Mystique distributions
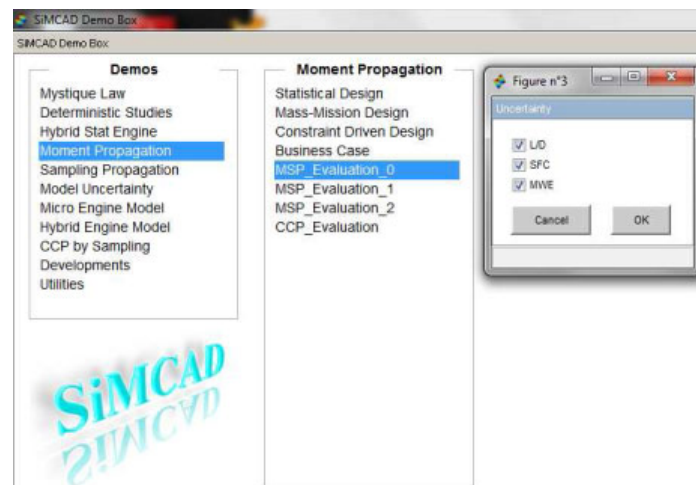
Figure A.7: Robustness study of the Margin Setting Process with the moment propagation method

The user can launch the robust optimisation by using two strategies: the first one is to build two levels of surrogate models in order to approximate the feasibility probability and the second one is to use moment propagation method.

To launch the first strategy with the surrogate models, the user has to execute the following steps:

- generate the model error sampling,

- build the performance data,

- build the surrogates models to approximate the aircraft performances,

- build the surrogates models to approximate the feasibility probabilities,

- launch the robust optimisation named 'CCP by MC'.

To launch the second strategy with the moment propagation, the user has to execute the following steps:

- generate the model error sampling,

- launch the robust optimisation named 'CCP by FOM'.

Whatever the strategy selected by the user, when clicking on the taps for the resolution of the optimisation, the user is asked to enter its preference on the value of the confident level P_0, as in Figure A.8.
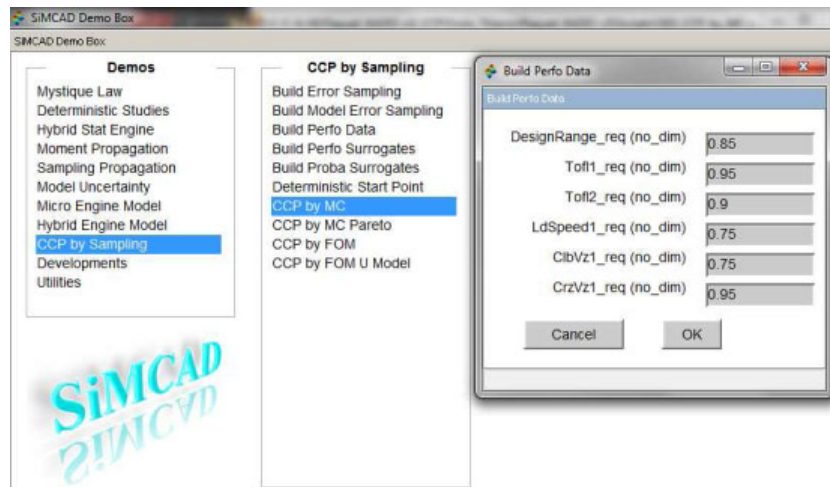
Figure A.8: Optimisation with Surrogate Models strategy

SiMCAD front end also proposes a tab to launch multi-objective optimisation. The currently proposed resolution used the two level of surrogate models to approximate the feasibility probability and the Genetical Algorithm Non dominated Sorting Genetic Algorithm (NSGA). During the execution, the user can visualise the point into the parameter space and the evolution of the Pareto Front.
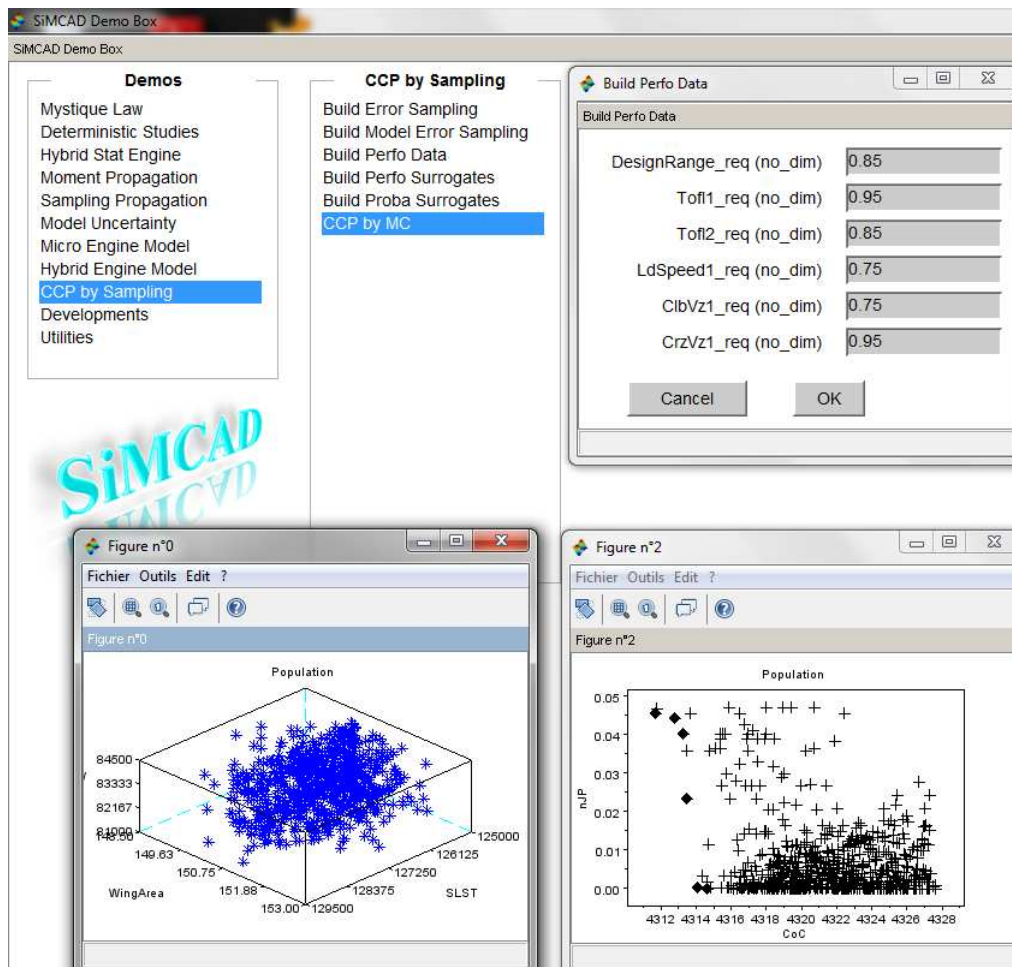
Figure A.9: Multi-Objective optimisation with Surrogate Models strategy and Genetical Algorithm NSGA

A.2 Some statistical tools

Several methods exist to validate the choice of the density function. Their objective is to test the adequacy between the sample of data and a candidate density function. They can be split into 2 groups: empirical method and statistical test (also called goodness-of-fit tests). In the empirical method, we find histogram and quantile-quantile plot (Q-Q plot). They are both graphical tools. Examples of goodness-of-fit tests are Chi-square, Kolmogorov-Smirnov, Cramer-Von Mises or Anderson Darling tests. In the next paragraphs, we give a brief description of Q-Q plot method and Anderson-Darling test (the definitions are extrated from OpenTURNS documentation).

A.2.1 Quantile-Quantile plot

Let X be a scalar uncertain variable modelled as a random variable. Quantile-Quantile plot is a graphical tool used to test whether two samples of data $\{x_1, \ldots, x_n\}$ and $\{x'_1, \ldots, x'_n\}$ are coming from the same probability distribution or not.

The α-quantile $q_X(\alpha)$ of X, where $\alpha \in (0, 1)$ is defined as follows:

$$\mathbb{P}(X \leq q_X(\alpha)) = \mathbb{F}_X(q_X(\alpha)) = \alpha. \tag{A.1}$$

Then:

$$q_X(\alpha) = \mathbb{F}_X^{-1}(\alpha). \tag{A.2}$$

The 50%-quantile ($q_x(50\%)$) is called the median, and the 25%-quantile ($q_x(25\%)$) and 75%-quantile ($q_x(75\%)$) are called the quartiles.

The value of the quantile is estimated empirically by using the sample. It is done in two steps:

1. the sample $\{x_1, \ldots, x_n\}$ is sort in increasing order to obtain the ordered statistics $x_{(1)}, \ldots, x_{(N)}$,

2. the α-quantile value is estimated by $\hat{q}_X(\alpha) = x_{([N]+1)}$

where $[N]$ is the intregral part of N.

A second estimate $\hat{q}'_X(\alpha)$ of the quantile can be calculated with the sample $\{x'_1, \ldots, x'_n\}$. If $\mathbb{F}$ is its cummulative distribution function, then we have:

$$\hat{q}'_X(\alpha) = \mathbb{F}_X^{-1}(\frac{j}{N+1}). \tag{A.3}$$

Then, if the sample of data is really coming from the distribution $\mathbb{F}$, the points $\{\hat{q}_X(\alpha), \hat{q}_X(\alpha)'\}$ should form a line (see figures A.10 and A.11).

Henry plot (or Henry line) is a classical, simple and quick way to test the adequation between the normal distribution and a sample of data. It is a particular case of Quantile-Quantile plot. Nevertheless, as the Cumulative Distribution Functions of the normal distribution has no simple expression, we used one of the properties of the Normal distribution given by: if X follows a Normal distribution with mean μ and standart deviation σ, then $\frac{X-\mu}{\sigma}$ follows a Normal distribution with mean 0 and standart-deviation 1.

Let us consider F the empirical CDF of the sample and G the CDF of the Normal distribution. The Henry line is built by following these steps:

1. Classification of the sample $\{x_1, \ldots, x_n\}$ in an increasing order.

2. Calculation of values of the CDF F_i of the observed sample, with $F_i = \frac{i-3/8}{n+1/4}$ (see [CC88]).
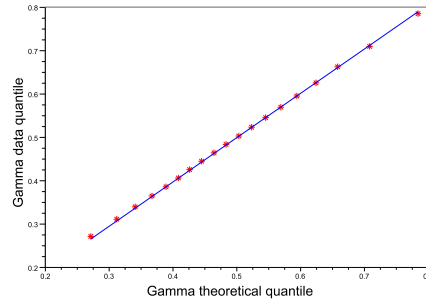
Figure A.10: Quantile-Quantile plot of Gamma data quantile versus Gamma theoretical quantile: the adequation between the two samples is total and it is materialized by a line.
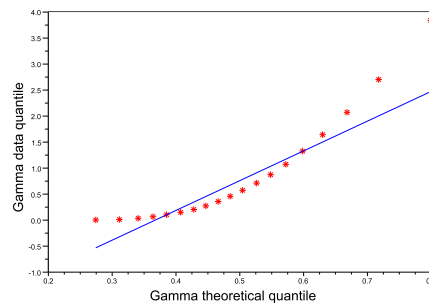
Figure A.11: Quantile-Quantile plot of Chi-square data quantile versus Gamma theoretical quantile: there is no adequation between the two samples.

3. Calculation of quantiles $z^*(i)$ associated to each value Fi by using the inverse of the normal ditribution with mean 0 and standart-deviation 1.

4. Representation of $(x(i), z^*(i))$.

If the distribution of observed sample is really a Normal distribution, then there exists a linear relation between the quantile of the observation and the theoretical Normal function with mean 0 and standart-deviation 1. The points $(x(i), z^*(i))$ represent a line called "Henry line".

A.2.2   Anderson-Darling test principle

The principle of the test is based on the calculation of the distance between the CDF of the sample of data $x_1, \ldots, x_n$, noted $\mathbb{F}_N$ and that of the probability distribution which is tested, noted $\mathbb{F}$. This distance belong to the class of quadratic statistical measure and is equal to:

$$D = \int_{-\infty}^{+\infty} \frac{[\mathbb{F}(x) - \mathbb{F}_N(x)]^2}{\mathbb{F}(x)(1 - \mathbb{F}(x))} d\mathbb{F}(x). \tag{A.4}$$

With the sample of data $x_1, \ldots, x_n$ sorted in an increasing order $x_{(1)}, \ldots, x_{(n)}$, this distance D is estimated with:

$$\hat{D} = -N - \sum_{i=1}^{N} \frac{2i - 1}{N} [ln(\mathbb{F}(x_{(i)}) - ln(1 - \mathbb{F}(x_{(N+1-i)}))]. \tag{A.5}$$

The probability distribution of the distance $\hat{D}$ is asymptotically known. If N is sufficiently large, one can calculate the threshold d_α such that:

- $\hat{D} > d_\alpha$, the candidate distribution is rejected with a risk of error α,

- $\hat{D} < d_\alpha$, the candidate distribution is considered acceptable.

d_α depends on the candidate distribution $\mathbb{F}$ being tested. In the context of OpenTURNS, it is limited to the case of a Normal distribution. The assumption which is tested is: "The SiMCAD model error is normally distributed". Then, d_α is calculated such that:

- $\hat{D} > d_\alpha$, SiMCAD model error is not normally distributed with a risk of error α,

- $\hat{D} < d_\alpha$, SiMCAD model error is normally distributed.

A.2.3   Parameter estimation of Density function

Once a density function is selected, we have to evaluate its parameter values that allow fitting the sample of data as well as possible. We present two methods hereafter: Method of Moments and Maximum Likelihood Method.

A.2.3.1   Method of Moments

The Method of Moments (MM) is the simplest existing method in order to evaluate the value of the parameter θ of a probability distribution that allows to best fit the probability on a sample of data $\{x_1, \ldots, x_n\}$. The method is based on the calculation of the empirical moments (see section 2.4.3.3 page 55) of the observed data (see [FEHP11]).
Suppose we look for an estimator $\hat{\theta}$ of the parameter $\theta = (\theta_1, \theta_2, \ldots, \theta_p)$ (where p is the number of scalar parameters of f_X) of the probability distribution f_X according to a sample of data $\{x_1, \ldots, x_n\}$. We know the expression of the first p moments of the probability distribution f_X which are defined as follow:

$$
\begin{aligned}
m_1(\theta) &= E(X) \\
m_2(\theta) &= E(X^2) \\
&\vdots \\
m_p(\theta) &= E(X^p).
\end{aligned}
\tag{A.6}
$$

Moreover, we can compute the first p empirical moments of the sample of data $\{x_1, \ldots, x_n\}$ given as:

$$
\mu_1 = \frac{1}{n} \sum_{i=1}^{n} x_i
$$

$$
\mu_2 = \frac{1}{n} \sum_{i=1}^{n} x_i^2
$$

$$
\ldots
\tag{A.7}
$$

$$
\mu_p = \frac{1}{n} \sum_{i=1}^{n} x_i^p.
\tag{A.8}
$$

The method of moment estimator $\hat{\theta}_{MM}$ is the value of θ solution of the following system:

$$
m_r(\theta) = \mu_r \text{ for } r = 1, \ldots, p.
\tag{A.9}
$$

Example: Suppose X follows a gamma distribution with parameter $\theta' = (\alpha, \beta)$ and with density function:

$$f(x; (\alpha, \beta)) = \frac{\beta^\alpha x^{\alpha-1} exp(-\beta x)}{\Gamma(\alpha)}.$$

The theoretical moment of X are:

$$E(X) = \frac{\alpha}{\beta} \tag{A.10}$$

$$E(X^2) = \frac{\alpha}{\beta^2}. \tag{A.11}$$

Suppose we have a sample of data $\{x_1, x_2, ..., x_n\}$, assuming to follow a gamma distribution. The first two empirical moments are given by:

$$\mu_1 = \frac{1}{n} \sum_{i=1}^{n} x_i = \bar{x} \tag{A.12}$$

$$\mu_2 = \frac{1}{n} \sum_{i=1}^{n} x_i^2 = s^2. \tag{A.13}$$

According to the method of moments, the estimator $\hat{\theta}'_{MM} = (\hat{\alpha}, \hat{\beta})$ of θ is solution of the system:

$$\begin{cases} E(X) = \mu_1 \\ E(X^2) = \mu_2. \end{cases}$$

Thus, the estimators of $\hat{\alpha}$ and $\hat{\beta}$ are:

$$\hat{\alpha} = \frac{\bar{x}^2}{s^2} \tag{A.14}$$

$$\hat{\beta} = \frac{\bar{x}}{s^2}. \tag{A.15}$$

A.2.3.2 Maximum Likelihood Method

The Maximum Likelihood Estimation (MLE) is another method to estimate the value of θ. It has been popularized by the statistician, biologist, eugenicist and geneticist R. A. Fisher. The MLE method is based on the calculation of the likelihood function from the density function taken as assumption and the sample of observed data.

Suppose we have the sample of data $\{x_1, \ldots, x_n\}$ iid: independent and identically distributed. The expression of the likelihood function is given by:

$$L(x_1, \ldots, x_n; \theta) = \prod_{i=1}^{n} f_X(x_i, \theta). \tag{A.16}$$

It can be seen as a the joint PDF of the sample $(x_1, \ldots, x_n)$. The estimator $\hat{\theta}_{MLE}$ of θ is the value of θ that maximise the likelihood function A.16. It is an optimisation process. In practice, it is more convenient to maximise the log-likelihood function than the likelihood one itself. The expression of $\hat{\theta}_{MLE}$ is given by:

$$\hat{\theta}_{MLE} = \text{argmax}_{\theta \in \mathbb{R}^n} \ln L(x_1, \ldots, x_n; \theta)). \tag{A.17}$$

Example

Consider the exponential probability distribution with parameter λ ($\lambda > 0$) and with PDF given by:

$$f(x,\lambda) = \begin{cases} \lambda e^{-\lambda x} & \text{if } x \geq 0 \\ 0 & \text{else.} \end{cases}$$

The likelihood function is expressed as follow:

$$L(x_1, \ldots, x_n; \lambda) \;\; = \;\; \prod_{i=1}^{n} \lambda e^{-\lambda x_i} \tag{A.18}$$

$$= \;\; \lambda^n e^{-\lambda \sum_{i}^{n} x_i}. \tag{A.19}$$

The log-likelihood associated is:

$$\ln L(x_1, \ldots, x_n; \lambda) = n\ln(\lambda) - \lambda \sum_{i}^{n} x_i. \tag{A.20}$$

It is a concave function of λ. The first derivative regarding λ is:

$$\frac{\partial \ln L(x_1, \ldots, x_n; \lambda)}{\partial \lambda} = \frac{n}{\lambda} - \sum_{i}^{n} x_i. \tag{A.21}$$

The previous first derivative is null for:

$$\hat{\lambda} = \frac{n}{\sum_{i}^{n} x_i}. \tag{A.22}$$

Thus, the MLE estimator $\hat{\lambda}_{MLE}$ of λ is:

$$\hat{\lambda}_{MLE} = \frac{n}{\sum_{i}^{n} x_i}. \tag{A.23}$$

A.2.4   Random variate generation

A.2.4.1   Cumulative Density Function Inverse Method

To propagate uncertainty source which are quantified with a PDF, it is necessary to first generate random variates from this probability distribution. To generate random simulations, we use a sample of random simulations of the Uniform(0,1) distributions which is transformed in order to obtain random simulations of the selected distribution. Several methods exist in the literature to do this transformation: the CDF inversion method, the rejection method (commonly called the acceptance-rejection method), the decomposition as discrete mixture method, the acceptance-complement method... Devroye ([Dev86]) gives a detailed description of some of them. Some techniques are specific to the type of distribution considered: [MT00] deals with a method for generating gamma samples and [Doo05] with one for generating normal samples. In our study, we used the CDF inversion method (previously described). It is the most simple and direct method for simulation. It is based on the following theorem:

Theorem A.2.1. *Let $\mathbb{F}$ be a continuous CDF, with an inverse function $\mathbb{F}^{-1}$ given by:*

$$\mathbb{F}^{-1}(u) = \inf\{x : \mathbb{F}(x) \geq u, u \in [0,1]\}. \tag{A.24}$$

If U is a variable following an uniform probability distribution on [0,1], then $\mathbb{F}^{-1}(U)$ is distributed from $\mathbb{F}$. Moreover, if X has $\mathbb{F}$ as CDF, then $\mathbb{F}(X)$ is uniformly distributed on [0,1].

The method is illustrated in figure A.12. Proof of theorem A.2.1 is available in [Dev86]. This

method is generally used when a simple analytical expression of the inverse CDF is available.
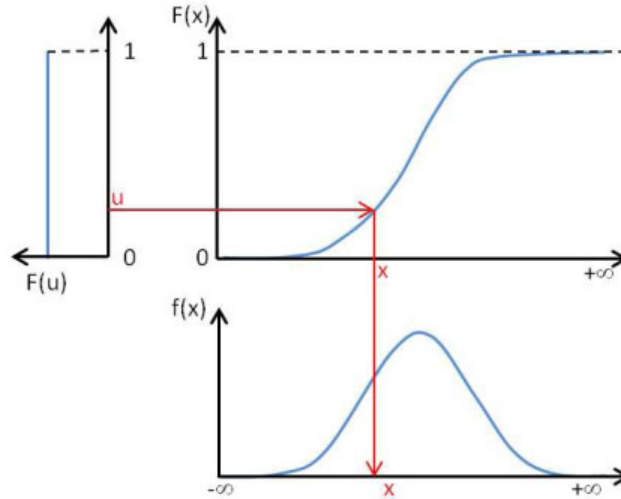
Figure A.12: Graphical illustration of the inverse method

A.2.4.2 Gaussian case: Box-Muller Method

The Inversion Method previously described is not effective when facing the Normal density as there is no simple expression of the Gaussian CDF. In 1958, George Box and Mervin Muller has created an ingenious transformation to overcome this issue: Box-Muller transformation. The Box-Muller transformation maps an Uniform(0,1) distribution to a standard Normal distribution. If U_1 and U_2 are random variables following a Uniform(0,1) distribution, then the random variables

$$N_1 = \sqrt{-2 * \ln(U_1)} * \cos(2\pi U_2)$$

and

$$N_2 = \sqrt{-2 * \ln(U_1)} * \sin(2\pi U_2)$$

follow a standard normal distribtuion. Thus, variables

$$S_1 = \mu + \sigma N_1$$

and

$$S_2 = \mu + \sigma N_2$$

follow a normal distribution with mean μ and standard-deviation σ.

A.3 Breguet-Leduc equation

In this section, we give further details on the three technological pillar that are combined in the Breguet-Leduc equation.

Breguet-Leduc is important because through its expression it appears that any technical modification of the aircraft can finally be expressed in term of L/D, SFC and MWE. Of course, many constaints will drive the design of the aircraft, but, at the end of the day, the combination of the resulting values of L/D, SFC and MWE will mostly contribute to its overall performance. We present hereafter each of these three pillars.

A.3.1 Aerodynamic efficiency

An aircraft in movement is subject to 4 main forces (see A.13):

- Lift (L)

- Weight (mg)

- Thrust (Fn)

- Drag (T)
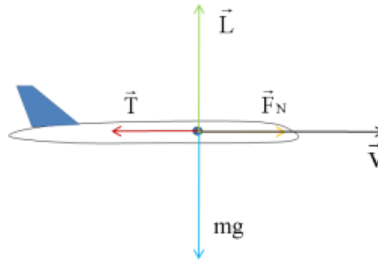
Figure A.13: Forces applied to an aircraft in movement

As any other object on the Earth, the aircraft is subject to the force called **Weight** due to the gravity. To be able to go up, a force is necessary to overcome the Weight of the aircraft. This force is called **Lift**. It is created due to the way the air moves around the aircraft wing. When having lift, we need help to move forward and create more air movement around the wing. This is done with the engine. The engines create a force called **Thrust** which help to go forward. Thrus also help to overcome **Drag** which is the consequence of the movement of the aircraft through the air.

These four forces are applied to any powered aircraft (see figure A.14).
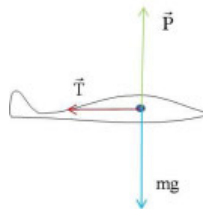
Figure A.14: Forces applied to a glider in movement

In the case of a glider there is no thrust to balance the drag, thus The glider can only move in the air mass that surrounds it. The projection of its weight along the glide path is playing the role of the engine thrust. The glider can rise above the ground when the air mass that surrounds it is rising quicker than the glider is going down.

The flight path angle γ can be expressed from thrust force T and lift force L:

$$\tan(\gamma) = \frac{T}{L}$$

. Thrust and lift force are generally represented by the product of a pressure with a surface $T = p * S$ and $L = p * S$ where:

- p is the pressure given by $\frac{1}{2} * \rho * V^2$

- S is an arbitrary surface. Wing area is generally taken as reference in this study.

Then we have $T = \frac{1}{2} * \rho * V^2 * S$ and $L = \frac{1}{2} * \rho * V^2 * S$. As S is an arbritrary value, it is necessary to add a coefficient which will allow to find the exact value of drag and lift. These coefficients are denoted Cx for the drag and Cz for the lift. They are called aerodynamic coefficient. That lead to

$$T = \frac{1}{2} * \rho * V^2 * S * Cx$$

and

$$L = \frac{1}{2} * \rho * V^2 * S * Cz$$

. Then, the expression of the angle of attack α is:

$$\tan(\alpha) \quad = \quad \frac{T}{L} \tag{A.25}$$

$$= \quad \frac{\frac{1}{2}\rho V^2 S * Cx}{\frac{1}{2}\rho V^2 S * Cz} \tag{A.26}$$

$$= \quad \frac{Cx}{Cz} \tag{A.27}$$

$$= \quad \frac{D}{L} \tag{A.28}$$

This last equation is the expression of an efficiency denoted L/D ratio which is an indicator of the quality of the aerodynamic form of the aircraft.

A.3.2 Propulsive efficiency

The SFC is related to the efficiency of the propulsion system. It represents the weight of Fuel burnt by unity of time (hours) and by unity of thrust (newton).

$$SFC = \frac{Fuel flow}{Thrust} = \frac{[M]}{[T][F]} = \frac{kg}{h * N} \tag{A.29}$$

It varies according to the flying conditions. It generally takes its value in the range [0.45 - 0.65] (even if most of aircraft have one between 0.58 and 0.63). The efficiency of the propulsion system deacreases when the SFC increases.

A.3.3 Structural efficiency

The structural efficiency can be seen as the ability to move from one point to another one by using the minimum quantity of structure. This efficiency is given by:

$$\eta = \frac{Arrival\ Weight}{Departure\ Weight} \tag{A.30}$$

This efficiency is related to the technology level. Its values is nowadays around 0.5. A technological breakthrough is needed if we want to move this value significantly lower.

A.3.4 Combination in Breguet Leduc equation

Hypothesis: SFC, f, Q = Departure Weight - Arrival Weight = Weight of fuel burnt. We look for the relation existing between Q and Ra.

$$\frac{dx}{dQ} = \frac{V}{SFC \cdot F}$$

$$Ra = \int_A^B \frac{V}{SFC \cdot F} dQ$$

$$Ra = \frac{V \cdot f}{SFC \cdot g} \int_A^B \frac{dQ}{m}$$

Fuel loss involving weight loss, we have: $dQ = -dm$. Then, we have:

$$Ra = \frac{V \cdot f}{SFC \cdot g} \int_A^B \frac{dm}{m} \tag{A.31}$$

$$= -\frac{V \cdot f}{SFC \cdot g} \int_{M_1}^{M_2} \frac{dm}{m} \tag{A.32}$$

$$= -\frac{V \cdot f}{SFC \cdot g} \left(\log(M_2) - \log(M_1)\right) \tag{A.33}$$

$$= -\frac{V \cdot f}{SFC \cdot g} \log\left(\frac{M2}{M1}\right) \tag{A.34}$$

$$= \frac{V \cdot f}{SFC \cdot g} \log\left(\frac{M1}{M2}\right) \tag{A.35}$$

where $M1$ is the aircraft weight at departure and $M2$ the aircraft weight at arrival.

A.4 SiMCAD

A.4.1 Geometrical description

Table A.1: Description of geometrical parameters

Name	Usual Unit	Short Description
	Cabin	
TotalNpax	integer	Total number of passengers
NpaxAbreast	integer	Number of pax abreast in economic class
Naisle	integer	Number of aisle in economic class
	Fuselage	
FusLength	m	Total fuselage length
FusHeight	m	Fuselage section height
FusWidth	m	Fuselage section width
FusFuel	l	Fuel in fuselage tank(s)
FusGWA	m2	Fuselage gross wetted area
	Wing	
WingArea	m2	Wing reference area
WingSpan	m	Wing span
WingAR	no dim	Wing aspect ratio : $(WingSpan)^2$ over Wing Area
WingTR	no dim	Wing taper ratio : Tip Chord over Root Chord ratio
WingSweep	deg	Wing sweep angle at 25% chord
WingMac	m2	Wing mean aerodynamic chord
WingToCr	no dim	Wing thickness over Chord ratio at root
WingToCk	no dim	Wing thickness over Chord ratio at main kink
WingToCt	no dim	Wing thickness over Chord ratio at tip
WingCTkFuel	litre	Fuel volume in the wing centre tank
WingFuel	litre	Fuel volume in the wing
WingGWA	m2	Wing gross wetted area
	Horizontal stabiliser	
HtpArea	m2	Horizontal tail plane area
HtpAR	no dim	Horizontal tail aspect ratio
HtpTR	no dim	Horizontal tail taper ratio
HtpSweep	deg	Horizontal tail sweep angle at 25% chord
HtpLarm	m	Horizontal tail lever arm
HtpVolume	no dim	Horizontal tail volume factor
HtpFuel	litre	Horizontal tail fuel volume
HtpGWA	m2	Horizontal tail gross wetted area
	Vertical stabiliser	
VtpArea	m2	Vertical tail plane area
VtpAR	no dim	Vertical tail aspect ratio
VtpTR	no dim	Vertical tail taper ratio
VtpSweep	deg	Vertical tail sweep angle at 25% chord
VtpLarm	m	Vertical tail lever arm
VtpVolume	no dim	Vertical tail volume factor
VtpGWA	m2	Vertical tail gross wetted area
	Landing gear	
nWheel	int	Number of wheel on main landing gear
	Nacelles	
nEngine	int	Number of engines

NacHeight	m	Nacelle height
NacWidth	m	Nacelle width
NacLength	m	Nacelle length
NacGWA	m2	Nacelles and pylons gross wetted area
BPR	nodim	Engine by pass ratio

A.4.2   Aerodynamic description

Table A.2: Description of aerodynamic parameters

Name	Usual Unit	Short Description
ZpRef	ft	Reference altitude for aero data
CruiseMach	mach	Cruise Mach number
LoDmaxCz	no dim	Lift coefficient for maximum Lift over Drag ratio
LoDmax	no dim	Maximum Lift over Drag ratio
CzMaxTO	no dim	Maximum lift coefficient in Take-off for MTOW SL ISA
CzMaxLD	no dim	Maximum lift coefficient in landing configuration

A.4.3   Mass description

Table A.3: Description of mass parameters

Name	Usual Unit	Short Description
Mass breakdown		
WingMass	kg	Wing equipped structural weight
FusMass	kg	Fuselage equipped structural weight
HtpMass	kg	Horizontal tail equipped structural weight
VtpMass	kg	Vertical tail equipped structural weight
LdgMass	kg	Landing gear weight
PylonMass	kg	Engine pylons equipped structural weight
EngMass	kg	Total engine weight
SystMass	kg	System weight
FurnMass	kg	Furnishings
MWEadjust	kg	MWE Adjustment
Operational masses		
OpItemMass	kg	Operator items weight
ContPallMass	kg	Containers and Pallets weights (Tare)
Characteristic masses		
MWE	kg	Evaluation MWE
OWE	kg	Evaluation Operational empty weight
MZFW	kg	Maximum Zero Fuel Weight
MFW	kg	Maximum Fuel Weight
MLW	kg	Maximum Landing Weight
MTOW	kg	Maximum Take-off Weight
FuelDensity	kg/l	Fuel density

A.4.4   Engine Performance

Table A.4: Description of engine parameters

Name	Usual Unit	Short Description
Propulsion		
SLSThrust	lbf	Sea level static thrust (reference thrust)
Take-Off Normal: case 1		
TONzp1	ft	Altitude for TON thrust
TONmach1	mach	Mach number for TON thrust
TONdisa1	degK	delta ISA for TON thrust
TONkfn1	no dim	Throttle push factor on TO rating
TONthrust1	lbf	Normal take off rating thrust at sea level
Take-Off Normal: case 2		
TONzp2	ft	Altitude for TON thrust
TONmach2	mach	Mach number for TON thrust
TONdisa2	degK	delta ISA for TON thrust
TONkfn2	no dim	Throttle push factor on TO rating
TONthrust2	lbf	Normal take off rating thrust at sea level
Maxi Continuous		
MCNzp	ft	Altitude for MCN thrust
MCNmach	mach	Mach number for MCN thrust
MCNdisa	degK	delta ISA for MCN thrust
MCNkfn	no dim	Throttle push factor on MCN rating
MCNthrust	lbf	Maxi continuous rating thrust
Max Climb		
MCLzp	ft	Altitude for MCL thrust
MCLmach	mach	Mach number for MCL thrust
MCLdisa	degK	delta ISA for MCL thrust
MCLkfn	no dim	Throttle push factor on MCL rating
MCLthrust	lbf	Maxi climb rating thrust
Max Cruise		
MCRzp	ft	Altitude for MCR thrust
MCRmach	mach	Mach number for MCR thrust
MCRdisa	degK	delta ISA for MCR thrust
MCRkfn	no dim	Throttle push factor on MCR rating
MCRthrust	lbf	Maxi cruise rating thrust
SFC bucket		
BucketZp	ft	Altitude for bucket sfc
BucketMach	mach	Mach number for bucket sfc
BucketDisa	degK	delta ISA for bucket sfc
BucketK Sfc	no dim	Technological factor on bucket SFC
BucketThrust	lbf	Net thrust at bucket sfc conditions
BucketSfc	lb/lbf/h	Minimum specific fuel consumption

A.4.5   Operationnal performance

Table A.5: Description of operational parameters

Name	Usual Unit	Short Description
Take-off: case 1		
ToDisa1	degK	Temperature shift for take off case 1

ToZp1	ft	Pressure altitude for take off case 1
ToKmtow1	no dim	MTOW ratio for take off case 1
ToKvs1g1	no dim	Factor V2/Vs1g for take off case 1
Tofl1	m	Field length for take off case 1
Take-off: case 2		
ToDisa2	degK	Temperature shift for take off case 2
ToZp2	ft	Pressure altitude for take off case 2
ToKmtow2	no dim	MTOW ratio for take off case 2
ToKvs1g2	no dim	Factor V2/Vs1g for take off case 2
Tofl2	m	Field length for take off case 2
Landing		
LdDisa1	degK	Temperature shift for landing case 1
LdZp1	ft	Pressure altitude for landing case 1
LdKmlw1	no dim	MLW ratio for landing case 1
LdKvs1g1	no dim	Factor Vapp/Vs1g for landing case 1
LdSpeed1	kt	Approach speed for landing case 1
Ceiling Climb		
ClbDisa1	degK	Temperature shift for climb case 1
ClbVz1	ft/min	Required vertical speed for climb case 1
ClbKmtow1	no dim	MTOW ratio for climb case 1
ClbZp1	ft	Ceiling pressure altitude for climb case 1
Ceiling Cruise		
CrzDisa1	degK	Temperature shift for cruise case 1
CrzVz1	ft/min	Required vertical speed for cruise case 1
CrzKmtow1	no dim	MTOW ratio for cruise case 1
CrzZp1	ft	Ceiling pressure altitude for cruise case 1
Buffeting		
CeilBufDisa	degK	Temperature shift for buffeting ceiling
CeilBufKmtow	no dim	MTOW ratio for buffeting ceiling
CeilBufKcz	no dim	effective CL margin versus buffeting lift
CeilBufZp	ft	Ceiling pressure altitude for buffeting
One engine out		
CeilOeiDisa	degK	Temperature shift for one engine out ceiling
CeilOeiKmtow	no dim	MTOW ratio for one engine out ceiling
CeilOeiPath	rad	Effective climb path angle for one engine
outCeilOeiZp	ft	Ceiling pressure altitude with one engine out
Max sar		
MaxSarDisa	degK	Temperature shift for max SAR altitude
MaxSarKmtow	no dim	MTOW ratio for max SAR altitude
MaxSarZp	ft	altitude for max Specific Air Range
Time to climb		
TtcDisa	degK	Temperature shift for Time to climb
TtcKmtow	no dim	MTOW ratio for time to climb
TtcZp	ft	Altitude to reach for time to climb
TimeToClimb	min	Time to climb

A.4.6   Mission performance

Table A.6: Description of mission parameters

Name	Usual Unit	Short Description
Conditions		
HSperfoDisa	degK	Temperature Shift for all mission computations
Nominal		
NominalMisRange	NM	Range of nominal mission
NominalMisTOW	kg	Take off weight of nominal mission
NominalMisZFW	kg	Zero fuel weight of nominal mission
NominalMisTotalFuel	kg	Total fuel of nominal mission
NominalPayLoad	kg	Payload of nominal mission
Max payload		
MaxPlMisRange	NM	Range of max payload mission
MaxPlMisTOW	kg	Take off weight of max payload mission
MaxPlMisZFW	kg	Zero fuel weight of max payload mission
MaxPlMisTotalFuel	kg	Total fuel of max payload mission
MaxPayLoad	kg	Payload of max payload mission
Max fuel		
MaxFuelMisRange	NM	Range of max fuel mission
MaxFuelMisTOW	kg	Take off weight of max fuel mission
MaxFuelMisZFW	kg	Zero fuel weight of max fuel mission
MaxFuelWeight	kg	Total fuel of max fuel mission
MaxFuelMisPayload	kg	Payload of max fuel mission
Zero payload		
ZeroPlMisRange	NM	Range of zero payload mission
ZeroPlMisTOW	kg	Take off weight of zero payload mission
ZeroPlMisZFW	kg	Zero fuel weight of zero payload mission
ZeroPlMisTotalFuel	kg	Total fuel of zero payload mission
Cost		
CostMisRange	NM	Range of cost mission
CostMisTOW	kg	Take off weight of cost mission
CostMisZFW	kg	Zero fuel weight of cost mission
CostMisPayload	kg	Payload of cost mission
CostMisBlockFuel	kg	Block fuel of cost mission
CostMisBlockTime	min	Block time of cost mission

A.4.7 Operating cost

Table A.7: Description of Cost parameters

Name	Usual Unit	Short Description
LaborCost	$/h	Labor Cost (typically: 60 $/h)
FuelPrice	$/USgal	Fuel Price (typically: 2 $/USgal)
ZeroPlMisTOW	kg	Take off weight of zero payload mission

+1

A.1.8 Geometry Models

Table A.8: Equations

Formula		Type
$WingArea = 88 \cdot TotalNpax \cdot 1.10^{-9} \cdot DesignRange + 60$	(1)	Model
$SLSThrust = \dfrac{177 \cdot TotalNpax \cdot 1 \cdot 10^{-6} \cdot DesignRange + 100000}{Nengine}$	(2)	Model
$\begin{cases} TotalNpax \leq 8, NpaxAbreast = 2; \\ 8 < TotalNpax \leq 16, NpaxAbreast = 3; \\ 16 < TotalNpax \leq 50, NpaxAbreast = 4; \\ 50 < TotalNpax \leq 100, NpaxAbreast = 5; \\ 100 < TotalNpax \leq 225, NpaxAbreast = 6; \\ 225 < TotalNpax \leq 300, NpaxAbreast = 8; \\ 300 < TotalNpax \leq 375, NpaxAbreast = 9; \\ 375 < TotalNpax, NpaxAbreat = 10; \end{cases}$	(3)	Model
$\begin{cases} NpaxAbreast \leq 6, \quad Naisle = 1 \\ NpaxAbreast > 6, \quad Naisle = 2 \end{cases}$	(4)	Model
$FusWidth = 0.38 \cdot NpaxAbreast + 1.05 \cdot Naisle + 0.55$	(5)	Model
$FusHeight = 0.38 \cdot NpaxAbreast + 1.05 \cdot Naisle$	(6)	Model
$FusLength = 7.9 \cdot FusWidth + 0.0063 \left(\dfrac{TotalNpax}{NpaxAbreast} \right)^{2.2}$	(7)	Model
$FusFuel = sign(Ftk) \cdot 0.27 \cdot FusLength \cdot FusWidth \cdot max(0, FusHeight - 2.2)$	(8)	Model
$FusGWA = 2.47 \cdot FusLength \cdot \sqrt{FusWidth \cdot FusHeight}$	(9)	Model
$WingMac = 1.2 \cdot \sqrt{\dfrac{WingArea}{WingAR}}$	(10)	Model
$WingSpan = \sqrt{WingArea \cdot WingAR}$	(11)	Model
$K = 3 \cdot \left(\dfrac{WingSpan \cdot WingMac}{4 \cdot WingArea} \right)$	(12)	Model
$WingTR = \dfrac{0.8 \cdot ((2 \cdot K - 1) - \sqrt{4 \cdot K - 3})}{2 \cdot (1 - K)} + 0.15$	(13)	Model
$WingSweep = 1.59 \cdot \max 0, (CruiseMach - 0.5)$	(14)	Model
$WingToCr = -0.030 \cdot CruiseMach + 0.180 \cdot \sqrt{\cos(WingSweep)}$	(15)	Model
$WingToCk = -0.028 \cdot CruiseMach + 0.140 \cdot \sqrt{cos(WingSweep)}$	(16)	Model
$WingToCt = -0.016 \cdot CruiseMach + 0.120 \cdot \sqrt{cos(WingSweep)}$	(17)	Model
$WingGWA = (0.00005 \cdot WingArea + 1.6) \cdot WingArea$	(18)	Model
$\begin{cases} WingFuel = 0.2 \cdot WingArea \cdot WingMac \cdot ... \\ \qquad ... \cdot \dfrac{5 \cdot WingToCr + 3 \cdot WingToCk + 2 \cdot WingToCt}{10} \end{cases}$	(19)	Model
$WingCTkFuel = sign(Ctk) \cdot 1.3 \cdot FusWidth \cdot WingToCr \cdot WingMac^2$	(20)	Model
$NacWidth = 0.28 \cdot BPR + 0.000004 \cdot SLSThrust$	(21)	Model
$NacHeight = 0.28 \cdot BPR + 0.000004 \cdot SLSThrust$	(22)	Model
$NacLength = 0.63 \cdot BPR + 0.0000013 \cdot SLSThrust$	(23)	Model
$NacGWA = 0.9 \cdot Nengine \cdot \pi \cdot \sqrt{NacHeight \cdot NacWidth} \cdot NacLength + 16$	(24)	Model
$HtpVolume = -0.087 \cdot SLSThrust \cdot 1 \cdot 10^{-5} + 1.05$	(25)	Model
$HtpAR = 5.1$	(26)	Constant
$HtpTR = 0.35$	(27)	Constant
$HtpSweep = 0.7 \cdot abs(WingSweep) + 0.2$	(28)	Model
$HtpLarm = (0.0002 \cdot FusLength + 0.45) \cdot FusLength$	(29)	Model

$$HtpArea = 1.012 \cdot (HtpVolume \cdot WingArea \cdot \frac{WingMac}{HtpLarm}) + 0.16 \tag{30}$$ Model

$$HtpGWA = (-0.002 \cdot HtpArea + 1.82) \cdot HtpArea \tag{31}$$ Model

$$HtpFuel = sign(Ttk) \cdot 0.08 \cdot HtpArea \tag{32}$$ Model

$$VtpVolume = -0.0014 \cdot (WingSpan \cdot SLSThrust \cdot 1 \cdot 10{-}6) + 0.09 \tag{33}$$ Model

$$VtpAR = 1.7 \tag{34}$$ Constant

$$VtpTR = 0.4 \tag{35}$$ Constant

$$VtpSweep = 0.75 \cdot abs(WingSweep) + 0.3 \tag{36}$$ Model

$$VtpLarm = (0.00034 \cdot FusLength + 0.42) \cdot FusLength \tag{37}$$ Model

$$VtpArea = (VtpVolume \cdot WingArea \cdot \frac{WingSpan}{VtpLarm}) + 0.4 \tag{38}$$ Model

$$VtpGWA = (-0.002 \cdot VtpArea + 2) \cdot VtpArea \tag{39}$$ Model

A.1.9 Mass models

Table A.9: Equations

Formula		Type
$MTOW = 67 \cdot TotalNpax \cdot DesignRange \cdot 1.10{-}6 + 20500$	(40)	Model
$MZFW = 41 \cdot TotalNpax \cdot DesignRange \cdot 1.10{-}6 + 25000$	(41)	Model
$FusMass = 5.47 \cdot (\pi \cdot FusLength \cdot \sqrt{FusWidth \cdot FusHeight})^{1.2}$	(42)	Model
$\begin{cases} A = 3.5 \cdot WingSpan^3 \cdot \sqrt{MTOW \cdot MZFW} \\ B = 0.6 \cdot WingToCr + 0.3 \cdot WingToCk + 0.1 \cdot WingToCt \\ C = WingArea \cdot cos(WingSweep)^2 \\ D = 1 \cdot 10^{-6} \cdot \frac{1 + 2 \cdot WingAR}{1 + WingAR} \\ WingMass = 33 \cdot WingArea^{1.1} + 1.1 \cdot \frac{A}{B \cdot C} \cdot D \end{cases}$	(43)	Model
$PylonMass = 0.0034 \cdot SLSThrust \cdot Nengine$	(44)	Model
$EngineMass = 0.0247 \cdot SLSThrust \cdot Nengine$	(45)	Model
$HtpMass = (0.04 \cdot HtpArea + 21) \cdot HtpArea$	(46)	Model
$VtpMass = (0.008 \cdot VtpArea + 29) \cdot VtpArea$	(47)	Model
$LdgMass = 0.03 \cdot MTOW^{1.02}$	(48)	Model
$SystemMass = 0.545 \cdot MTOW^{0.8}$	(49)	Model
$FurnMass = (0.063 \cdot TotalNpax + 9.76) \cdot TotalNpax$	(50)	Model
$\begin{cases} MWE = WingMass + FusMass + HtpMass + VtpMass + LdgMass \\ \qquad + PylonMass + EngineMass + SystemMass + FurnMass \end{cases}$	(51)	Definition
$OpItemMass = 5.2 \cdot TotalNpax \cdot DesignRange \cdot 1 \cdot 10{-}6$	(52)	Model
$ContPallMass = sign(CnP) \cdot 4.36 \cdot FusWidth \cdot FusLength$	(53)	Model
$OWE = MWE + OpItemMass + ContPallMass$	(54)	Definition
$FuelDensity = 803$	(55)	Constant
$MFW = (FusFuel + WingCTkFuel + WingFuel + HtpFuel) \cdot FuelDensity$	(56)	Definition
$NominalMisTOW = MTOW$	(57)	Model
$NominalPayload = 102 \cdot TotalNpax$	(58)	Model
$MaxPlMisTOW = MTOW$	(59)	Model
$MaxPayload = 145 \cdot TotalNpax$	(60)	Model
$MaxFuelMisTOW = MTOW$	(61)	Model
$MaxFuelPayload = max(MTOW - MFW, OWE) - OWE$	(62)	Definition
$ZeroPlMisTOW = OWE + min(MFW, MTOW - OWE)$	(63)	Definition
$CostMisPayload = NominalPayload$	(64)	Model

$$MZFWeff = OWE + MaxPayload \qquad (65) \qquad \text{Definition}$$
$$MLW = 1.07 \cdot MZFW \qquad (66) \qquad \text{Model}$$

A.1.10 Atmospheric models

Function Name: non_stand_atmos	(67)

Input: $Disa, Zp -$ **Ouput** $: Pamb, Tamb, Rho$

$$
\begin{cases}
Zp \le 11000, \begin{cases} Pstd = (8.961962852 - Zp \cdot 2.021612304 \cdot 10^{-4})^{5.255880} \\ Tstd = 288.15 - 0.0065 \cdot Zp \end{cases} \\[2ex]
11000 < Zp \le 20000, \begin{cases} Pstd = 128244.6928 \cdot exp(Zp \cdot (-1.576885 \cdot 10^{-4})) \\ Tstd = 216.65 \end{cases} \\[2ex]
20000 < Zp \le 32000, \begin{cases} Pstd = (0.7055184555 + Zp \cdot 3.587686018 \cdot 10^{-6})^{-34.16322} \\ Tstd = 216.65 + 0.001 \cdot (Zp - 20000) \end{cases} \\[2ex]
32000 < Zp \le 47000, \begin{cases} Pstd = (0.3492686141 + Zp \cdot 7.033096869 \cdot 10^{-6})^{-12.20115} \\ Tstd = 228.65 + 0.0028 \cdot (Zp - 32000) \end{cases} \\[2ex]
47000 < Zp \le 50000, \begin{cases} Pstd = 41828.42421 \cdot exp(Zp \cdot (-1.2622656 \cdot 10^{-4})) \\ Tstd = 270.65 \end{cases}
\end{cases}
$$

$$Pamb = Pstd$$
$$Tamb = Tstd + Disa$$
$$Rho = Pamb/(287.05 \cdot Tamb)$$

Function Name: $pressure\_altitude\_$	(68)

Input: $Pamb -$ **Output:** Zp

$$
\begin{cases}
Pamb \ge 22632 , Zp = 44330.76923 - 4946.546863 \cdot Pamb^{0.1902630958} \\
22632 \ge Pamb \ge 5474.87 , Zp = 74588.16198 - 6341.616541 \cdot \log(Pamb) \\
5474.87 \ge Pamb \ge 868.014 , Zp = -196650 + 278731.1919/Pamb^{0.02927124551} \\
868.014 \ge Pamb \ge 110.906 , Zp = -49660.71227 + 142184.8751/Pamb^{0.08195948743} \\
110.906 \ge Pamb \ge 75.9443 , Zp = 84303.42544 - 7922.262953 \cdot \log(Pamb)
\end{cases}
$$

Function Name: $climb\_mode$	(69)

Input: $Cmode, Disa, Tamb, Zp, Mach -$ **Ouput** $: Acc, dTodz$

$$
\begin{cases}
Zp < 10950 , dTodZ = -0.0065 \\
10950 \le Zp < 11050 , dTodZ = -0.0065 + (0.0065/100) \cdot (Zp - 10950) \\
11050 \le Zp < 19950 , dTodZ = 0 \\
19950 \le Zp < 20050 , dTodZ = (0.0010/100) \cdot (Zp - 19950) \\
20050 \le Zp < 31950 , dTodZ = 0.0010 \\
31950 \le Zp < 32050 , dTodZ = 0.0010 + (0.0018/100) \cdot (Zp - 31950) \\
32050 \le Zp < 46950 , dTodZ = 0.0028 \\
46950 \le Zp < 47050 , dTodZ = 0.0028 - (0.0028/100) \cdot (Zp - 46950) \\
47050 \le Zp < 50000 , dTodZ = 0
\end{cases}
$$

$$
\begin{cases}
Cmode ='CAS' \quad , Acc = 1 + \dfrac{(1 + 0.2 \cdot Mach^2)^{3.5} - 1}{(1 + 0.2 \cdot Mach^2)^{2.5}} + \\
\qquad\qquad\qquad + 20.49029021 \cdot Mach^2 \cdot \dfrac{Tamb - Disa}{Tamb} \cdot dTodZ \\
Cmode ='MACH' \, , Acc = 1 + 20.49029021 \cdot Mach^2 \cdot \dfrac{Tamb - Disa}{Tamb} \cdot dTodZ
\end{cases}
$$

Function Name: $Vcas\_from\_Mach\_$	(70)
Input: $Pamb, Mach$ − **Ouput** : $Vcas$	

$$Vcas = 340.29 \cdot \sqrt{5 \cdot (((Pamb \cdot ((1 + 0.2 \cdot Mach^2)^{3.5} - 1)/101325) + 1)^{0.4/1.4} - 1)}$$

Function Name: $Mach\_from\_Vcas\_$	(71)
Input: $Pamb, Vcas$ − **Ouput** : $Mach$	

$$Mach = \sqrt{\frac{(((0.2 \cdot (Vcas/340.29)^2 + 1)^{3.5} - 1) \cdot 101325/Pamb + 1)^{1/3.5} - 1}{0.2}}$$

A.1.11 Miscellanous

Function Name: m_nm (*Translate nautical miles into metres*)	(72)
Input: nm − **Ouput** : m	
$m = nm \cdot 1852$	

Function Name: m_ft (*Translate feet into metres*)	(73)
Input: ft − **Ouput** : m	
$m = ft \cdot 0.3048$	

Function Name: s_min (*Translate minutes into seconds*)	(74)
Input: mn − **Ouput** : s	
$s = mn \cdot 60$	

Function Name: mps_kt (*Translate knots into metres per seconds*)	(75)
Input: kts − **Ouput** : mps	
$s = kts \cdot 1852/3600$	

Function Name: $Dpl\_DpUSgal$ (*Translate dollars per US gallon into dollars per litre*)	(76)
Input: $DpUSgal$ − **Ouput** : Dpl	
$Dpl = DpUSgal \cdot 0.264173$	

A.1.12 Aerodynamic models

Function Name: $CzMaxTO_-$	(77)
Input: $WingSweep$ − **Ouput** : $CzMaxTO$	
$CzMaxTO = 2.84 \cdot ((-1.3459506 * WingSweep + 0.7648987) \cdot WingSweep + 0.71)$	

Function Name: $CzMaxLD_-$	(78)
Input: $WingSweep$ − **Ouput** : $CzMaxLD$	
$CzMaxLD = 3.20 \cdot ((-0.4858553 \cdot WingSweep - 0.4277130) \cdot WingSweep + 1.11))$	

Function Name: $divergence\_mach_-$	(79)
Input: Cz − **Ouput** : $MachDiv$	

$$Czth = max([0.25, Cz])$$
$$MachDiv = 1.0358 \cdot \frac{(0.85 - 0.5 \cdot (Czth^2 + 0.0625) + 0.25 \cdot Czth)}{\sqrt{\cos WingSweep} + 0.1 \cdot WingToCr + 0.4 \cdot WingToCk + 0.5 \cdot WingToCt}$$

Function Name: $buffeting\_lift_-$	(80)
Input: $Mach$ − **Ouput** : $CzBuf$	

$CzBuf = 0.6$
for $j = 1 : 8$
  $[MachDiv] = divergence\_mach\_(CzBuf)$
  $x = max(0.277, Mach + (MachDiv - 0.82))$
  $y = 37.821274 \cdot x^5 - 133.87462 \cdot x^4 + 164.16013 \cdot x^3 - 86.945536 \cdot x^2 + 18.268534 \cdot x + 0.2362113$
  $CzBuf = 0.45 \cdot y + 0.35$
endfor

Function Name: $aero\_efficiency_-$	(81)
Input: $Pamb, Tamb, Mach, Cz$ − **Ouput** : LoD	

$$Fac = (1 + 0.126 \cdot Mach^2)$$
$$Re = 47899 \cdot Pamb \cdot Mach \cdot (Fac \cdot Tamb + 110.4)/(Tamb^2 \cdot Fac^{2.5})$$
$$FusCx0 = 1.4 \cdot ((0.455/Fac) \cdot \left(\frac{log(10)}{log(Re \cdot FusLength)}\right)^{2.58} \cdot \frac{FusGWA}{WingArea}$$
$$NacCx0 = 1.4 \cdot ((0.455/Fac) \cdot \left(\frac{log(10)}{log(Re \cdot NacLength)}\right)^{2.58} \cdot \frac{NacGWA}{WingArea}$$
$$TailCx0 = 1.4 \cdot ((0.455/Fac) \cdot \left(\frac{log(10)}{log(Re \cdot WingMac)}\right)^{2.58} \cdot \frac{(HtpGWA + VtpGWA)}{WingArea}$$
$$WingCx0 = 1.4 \cdot ((0.455/Fac) \cdot \left(\frac{log(10)}{log(Re \cdot WingMac)}\right)^{2.58} \cdot \frac{WingGWA}{WingArea}$$
$$Cx0 = FusCx0 + WingCx0 + NacCx0 + TailCx0$$
$$Cxi = \frac{(FusWidth/WingSpan)^2 + 1.02}{\pi \cdot WingAR} \cdot Cz^2$$
$$Mdiv = divergence\_mach\_(Cz)$$
$$Cxc = 0.0002 \cdot exp(45 \cdot (Mach - Mdiv))$$
$$Cx = Cx0 + Cxi + Cxc$$
$$LoD = Cz/Cx$$

Function Name: $lift\_of\_LoDmax\_$	(82)

Input: $Disa, Zp, Mach$ − **Ouput** : $CzLoDmax$

$[Pamb, Tamb] = non\_stand\_atmos\_(Disa, Zp)$
$dCz = 0.05$; $[Cz1, Cz2, Cz3] = (0, dCz, 2 \cdot dCz)$
$LoD1 = aero\_efficiency\_(Pamb, Tamb, Mach, Cz1)$
$LoD2 = aero\_efficiency\_(Pamb, Tamb, Mach, Cz2)$
$LoD3 = aero\_efficiency\_(Pamb, Tamb, Mach, Cz3)$
while $LoD3 > LoD2$
  $Cz1 = Cz2$; $Cz2 = Cz3$; $Cz3 = Cz3 + dCz$; $LoD1 = LoD2$; $LoD2 = LoD3$
  $[LoD3] = aero\_efficiency\_(Pamb, Tamb, Mach, Cz3)$
endwhile
$X = [Cz1 ; Cz2 ; Cz3]$
$A = [X^2, X, ones(X)]$
$B = [LoD1 ; LoD2 ; LoD3]$
$C = lsq(A, B)$
$CzLoDmax = -C(2)/(2 \cdot C(1))$

Table A.26: Equations

Formula		Type
$CzMaxTO = CzMaxTO\_(WingSweep)$	(83)	Model
$CzMaxLD = CzMaxLD\_(WingSweep)$	(84)	Model
$LoDmaxCz = lift\_of\_LoDmax\_(0, ZpRef, CruiseMach)$	(85)	Model
$(Pamb, Tamb) = non\_stand\_atmos\_(0, ZpRef)$	(86)	Model
$LoDmax = aero\_efficiency\_(Pamb, Tamb, CruiseMach, LoDmaxCz)$	(87)	Model

A.1.13 Engine models

Function Name: $engine\_thrust\_$	(88)

Input: $SLSThrust, Tamb, rho, Mach, rating$ − **Ouput** : $Fnet$

if $rating = "mto"$, $Kfn = 0.82$
  elseif $rating = "mcn"$, $Kfn = 0.61$
  elseif $rating = "mcl"$, $Kfn = 0.47$
  elseif $rating = "mcr"$, $Kfn = 0.44$
  endif
$MC = [-0.0017 , 0.0562 , 0.1693 ; 0.0043 , -0.1269 , -0.1914 ; -0.001 , 0.0279 , 1.0354]$
$kBprMach = [Mach^2 , Mach , 1] \cdot MC \cdot [BPR^2 ; BPR ; 1]$
$Fnet = Kfn \cdot SLSThrust \cdot \sqrt{rho/1.225} \cdot kBprMach$

Function Name: $engine\_wimd\_m\_drag\_$	(89)

Input: $Pamb, Mach$ − **Ouput** : Fd

$Fd = 0.7 \cdot Pamb \cdot Mach^2 \cdot (0.12 \cdot NacWidth \cdot NacHeight)$

Function Name: $engine\_sfc\_$	(90)
Input: $Pamb, Tamb, Mach, Fnet -$ **Ouput** $: Sfc$	
$Sfc = (-0.026 \cdot BPR + 0.76)/(10 \cdot 3600)$	

Table A.30: Equations

Formula		Type
$TONdisa1 = 15$	(91)	Constant
$TONzp1 = 0$	(92)	Constant
$TONmach1 = 0.25$	(93)	Constant
$(Pamb, Tamb, Rho) = non\_stand\_atmos\_(TONdisa1, TONzp1)$	(94)	Model
$TONthrust1 = engine\_thrust\_(SLSThrust, Tamb, Rho, TONmach1,' mto')$	(95)	Model
$TONdisa2 = 15$	(96)	Constant
$TONzp2 = m\_ft(5000)$	(97)	Constant
$TONmach2 = 0.25$	(98)	Constant
$(Pamb, Tamb, Rho) = non\_stand\_atmos\_(TONdisa1, TONzp2)$	(99)	Model
$TONthrust2 = engine\_thrust\_(SLSThrust, Tamb, Rho, TONmach2,' mto')$	(100)	Model
$MCNdisa = 15$	(101)	Constant
$MCNzp = m\_ft(25000)$	(102)	Constant
$MCNmach = 0.45 \cdot CruiseMach$	(103)	Constant
$(Pamb, Tamb, Rho) = non\_stand\_atmos\_(MCNdisa, MCNzp)$	(104)	Model
$MCNthrust = engine\_thrust\_(SLSThrust, Tamb, Rho, MCNmach,' mcn')$	(105)	Model
$MCLdisa = 0$	(106)	Constant
$MCLzp = m\_ft(25000)$	(107)	Constant
$MCLmach = CruiseMach$	(108)	Constant
$(Pamb, Tamb, Rho) = non\_stand\_atmos\_(MCLdisa, MCLzp)$	(109)	Model
$MCLthrust = engine\_thrust\_(SLSThrust, Tamb, Rho, MCLmach,' mcl')$	(110)	Model
$MCRdisa = 0$	(111)	Constant
$MCRzp = ZpRef$	(112)	Constant
$MCRmach = CruiseMach$	(113)	Constant
$(Pamb, Tamb, Rho) = non\_stand\_atmos\_(MCRdisa, MCRzp)$	(114)	Model
$MCRthrust = engine\_thrust\_(SLSThrust, Tamb, Rho, MCRmach,' mcr')$	(115)	Model
$Bucketdisa = 0$	(116)	Constant
$Bucketzp = ZpRef$	(117)	Constant
$Bucketmach = CruiseMach$	(118)	Constant
$(Pamb, Tamb, Rho) = non\_stand\_atmos\_(BucketDisa, Bucketzp)$	(119)	Model
$MCRthrustBucket = engine\_thrust\_(SLSThrust, ...$ $..., Tamb, Rho, BucketMach,' mcr')$	(120)	Model
$BucketThrust = 0.7472 \cdot MCRthrustBucket + 4233$	(121)	Model
$BucketSfc = engine\_sfc\_(Pamb, Tamb, BucketMach, BucketThrust)$	(122)	Model

A.1.14 Low speed operation models

A.1.14.1 Take off

Function Name: $take\_off\_$	(123)
Input: $mass, Disa, Zp, KvsTO -$ **Ouput** $: TOFL$	
$g = 9.80665$	

$$[Pamb, Tamb, Rho] = non\_stand\_atmos\_(Disa, Zp)$$
$$CzMaxTO = CzMaxTO\_(WingSweep)$$
$$MachStall = \sqrt{\frac{mass \cdot g}{0.7 \cdot Pamb \cdot WingArea \cdot CzMaxTO}}$$
$$Mach = KvsTO \cdot MachStall$$
$$Fnet = engine\_thrust\_(SLSThrust, Tamb, Rho, Mach,' mto')$$
$$K = \frac{mass^2}{\frac{CzMaxTO}{KvsTO^2} \cdot Fnet \cdot Nengine \cdot WingArea \cdot (Rho/1.225)^{0.8}}$$
$$TOFL = (-0.056 \cdot K + 34.5) \cdot K - 1600$$

Table A.32: Equations

Formula		Type
$ToKvs1g1 = 1.13$	(124)	Constant
$ToKmtow1 = 1$	(125)	Constant
$ToDisa1 = 15$	(126)	Constant
$ToZp1 = 0$	(127)	Constant
$Tofl1 = take\_off\_(ToKmtow1 \cdot MTOW, ToDisa1, ToZp1, ToKvs1g1)$	(128)	Model
$ToKvs1g2 = 1.13$	(129)	Constant
$ToKmtow2 = 1$	(130)	Constant
$ToDisa2 = 33$	(131)	Constant
$ToZp2 = m\_ft(1200)$	(132)	Constant
$Tofl2 = take\_off\_(ToKmtow2 \cdot MTOW, ToDisa2, ToZp2, ToKvs1g2)$	(133)	Model

A.1.14.2 Landing

Function Name: $approach\_speed\_$	(134)
Input: $mass, Disa, Zp, KvsLD$ − **Ouput** : $AppSpeed$	

$$g = 9.80665$$
$$[Pamb, Tamb, Rho] = non\_stand\_atmos\_(Disa, Zp)$$
$$CzMaxLD = CzMaxLD\_(WingSweep)$$
$$AppSpeed = 0.98 \cdot \sqrt{\frac{2 \cdot mass \cdot g}{Rho \cdot WingArea \cdot (CzMaxLD/KvsLD^2)}}$$

Table A.34: Equations

Formula		Type
$LdKvs1g1 = 1.23$	(135)	Constant
$LdKmlw1 = 1$	(136)	Constant
$LdDisa1 = 0$	(137)	Constant
$LdZp1 = 0$	(138)	Constant
$LdSpeed1 = approach\_speed\_(LdKmlw1 \cdot MLW, LdDisa1, LdZp1, LdKvs1g1)$	(139)	Model

A.1.15 High speed operation models

A.1.15.1 Ceilings

Function Name: $flight\_aero\_data\_$	(140)
Input: $Disa, Zp, Mach, mass$ − **Ouput** : $LoD, Cz, g, SoundSpeed, Pamb, Tamb, Rho$	

$g = 9.80665$
$[Pamb, Tamb, Rho] = non\_stand\_atmos\_(Disa, Zp)$
$SoundSpeed = sqrt(1.4 \cdot 287.05 \cdot Tamb)$
$Cz = (mass \cdot g)/(0.7 \cdot Pamb \cdot WingArea \cdot Mach^2)$
$LoD = aero\_efficiency\_(Pamb, Tamb, Mach, Cz)$

Function Name: $climb\_speed\_$	(141)
Input: $Cmode, mass, Disa, Zp, Mach, rating$ − **Ouput** : $Vz, Path\_$	

$g = 9.80665$
$[LoD, Cz, g, SoundSpeed, Pamb, Tamb, Rho] = flight\_aero\_data\_(Disa, Zp, Mach, mass)$
$Acc = climb\_mode\_(Cmode, Disa, Tamb, Zp, Mach)$
$Fnet = engine\_thrust\_(SLSThrust, Tamb, Rho, Mach, rating)$
$Path\_ = (Fnet \cdot nEngine/(mass \cdot g) - 1/LoD)/Acc$
$Vz = Mach \cdot SoundSpeed \cdot Path\_$

Function Name: $climb\_path\_$	(142)
Input: $Cmode, mass, Disa, Zp, Vcas, rating, Nei$ − **Ouput** : $Path\_, Vz$	

$g = 9.80665$
$[Pamb, Tamb, Rho] = non\_stand\_atmos\_(Disa, Zp)$
$Mach = Mach\_from\_Vcas\_(Pamb, Vcas)$
$LoD, Cz, g, SoundSpeed] = flight\_aero\_data\_(Disa, Zp, Mach, mass)$
$Acc = climb\_mode\_(Cmode, Disa, Tamb, Zp, Mach)$
$Fnet = engine\_thrust\_(SLSThrust, Tamb, Rho, Mach, rating)$
$Path\_ = (Fnet \cdot (nEngine - Nei)/(mass \cdot g) - 1/LoD)/Acc$
$Vz = Mach \cdot SoundSpeed \cdot Path\_$

Function Name: $cas\_max\_path\_$	(143)
Input: $mass, Disa, Zp$ − **Ouput** : Cas	

$g = 9.80665$
$[Pamb, Tamb, Rho] = non\_stand\_atmos\_(Disa, Zp)$
$Mach = \sqrt{\dfrac{mass \cdot g}{0.7 \cdot Pamb \cdot WingArea \cdot LoDmaxCz}}$
$Cas = Vcas\_from\_Mach\_(Pamb, Mach)$

Function Name: $buffet\_$	(144)
Input: $mass, Disa, Zp, Mach - $**Ouput** $: kCz$	
$[LoD, CzEff] = flight\_aero\_data\_(Disa, Zp, Mach, mass)$	
$CzBuf = buffeting\_lift\_(Mach)$	
$kCz = CzBuf/CzEff$	

Table A.40: Equations

Formula		Type
$ClbKmtow1 = 0.97$	(145)	Constant
$ClbDisa1 = 0$	(146)	Constant
$ClbZp1 = m\_ft(35000)$	(147)	Constant
$ClbVz1 = climb\_speed\_('MACH', ClbKmtow1 \cdot MTOW, ClbDisa1, ...$ $..., ClbZp1, CruiseMach,' mcl')$	(148)	Model
$CrzKmtow1 = 0.97$	(149)	Constant
$CrzDisa1 = 0$	(150)	Constant
$CrzZp1 = m\_ft(35000)$	(151)	Constant
$CrzVz1 = climb\_speed\_('MACH', CrzKmtow1 \cdot MTOW, CrzDisa1, ...$ $..., CrzZp1, CruiseMach,' mcr')$	(152)	Model
$CeilBufKmtow = 0.97$	(153)	Constant
$CeilBufDisa = 0$	(154)	Constant
$CeilBufZp = m\_ft(35000)$	(155)	Constant
$CeilBufKczMin = 1.3$	(156)	Constant
$CeilBufKcz = buffet\_(CeilBufKmtow \cdot MTOW, CeilBufDisa, ...$ $..., CeilBufZp, CruiseMach)$	(157)	Model
$\begin{cases} nEngine = 2 \ , CeilOeiMinPath = 0.011 \\ nEngine = 3 \ , CeilOeiMinPath = 0.013 \\ nEngine = 4 \ , CeilOeiMinPath = 0.016 \end{cases}$	(158)	Model
$CeilOeiKmtow = 0.97$	(159)	Constant
$CeilOeiDisa = 10$	(160)	Constant
$CeilOeiZp = m\_ft(19000)$	(161)	Constant
$CeilOeiCas = cas\_max\_path\_(CeilOeiKmtow \cdot MTOW, ...$ $CeilOeiDisa, CeilOeiZp)$	(162)	Model
$CeilOeiPath = climb\_path\_('CAS', CeilOeiKmtow \cdot MTOW, ...$ $..., , CeilOeiDisa, CeilOeiZp, CeilOeiCas,' mcn', 1)$	(163)	Model

A.1.15.2 Best cruise altitude

Function Name: $maximum\_SAR\_Zp\_$	(164)
Input: $mass, Disa, Mach - $**Ouput** $: Zp$	
$dZp = m\_ft(2500) \ ; \ [Zp1, Zp2, Zp3] = (0, dZp, 2 * dZp)$	
$Sar1 = cruise\_SAR\_(Disa, Zp1, Mach, mass)$	
$Sar2 = cruise\_SAR\_(Disa, Zp2, Mach, mass)$	
$Sar3 = cruise\_SAR\_(Disa, Zp3, Mach, mass)$	
while $Sar3 > Sar2$	
$\quad [Zp1, Sar1, Zp2, Sar2, Zp3] = (Zp2, Sar2, Zp3, Sar3, Zp3 + dZp)$	

$$Sar3 = cruise\_SAR\_(Disa, Zp3, Mach, mass)$$
endwhile
$$X = [Zp1 ; Zp2 ; Zp3]$$
$$A = [X^2, X, ones(X)]$$
$$B = [Sar1 ; Sar2 ; Sar3]$$
$$C = lsq(A, B)$$
$$Zp = -C(2)/(2 \cdot C(1))$$

Function Name: $cruise\_SAR\_$	(165)
Input: $Disa, Zp, Mach, mass$ − **Ouput** : Sar	

$[LoD, Cz, g, SoundSpeed, Pamb, Tamb, Rho] = flight\_aero\_data\_(Disa, Zp, Mach, mass)$
$Fnet = (mass \cdot g)/LoD$
$Sfc = engine\_sfc\_(Pamb, Tamb, Mach, Fnet)$
$Sar = 0.98 \cdot (Mach \cdot SoundSpeed \cdot LoD)/(mass \cdot g \cdot Sfc)$

Table A.43: Equations

Formula		Type
$MaxSarKmtow = 0.97$	(166)	Constant
$MaxSarDisa = 0$	(167)	Constant
$MaxSarZp = maximum\_SAR\_Zp\_(MaxSarKmtow \cdot MTOW, ...$ $..., MaxSarDisa, CruiseMach)$	(168)	Model

A.1.15.3   Time to climb

Function Name: $time\_to\_climb\_$	(169)
Input: $mass, Disa, Zp\_max, Mach\_max$ − **Ouput** : TtC	

$Zp\_ini = m\_ft(1500)$
$Vcas\_ini = mps\_kt(230)$
$Zp\_int = m\_ft(10000)$
$Vcas\_int = mps\_kt(250)$
$TtC1 = time\_to\_climb\_seg\_('CAS', mass, Disa, Zp\_ini, Zp\_int, Vcas\_ini)$
$Pamb\_brk = \left((1 + 0.2 \cdot (Vcas\_int/340.29)^2)^{3.5} - 1\right) \cdot 101325/((1 + 0.2 \cdot Mach\_max^2)^{3.5} - 1)$
$Zp\_brk = pressure\_altitude\_(Pamb\_brk)$

$$\begin{cases} Zp\_max < Zp\_brk \begin{cases} [TtC2, Mach\_top] = time\_to\_climb\_seg\_('CAS', mass, ... \\ \qquad\qquad\qquad\qquad ..., Disa, Zp\_int, Zp\_max, Vcas\_int) \\ TtC3 = 0 \end{cases} \\ \\ Zp\_int < Zp\_brk < Zp\_max \begin{cases} TtC2 = time\_to\_climb\_seg\_('CAS', ... \\ \qquad\qquad ..., mass, Disa, Zp\_int, Zp\_brk, Vcas\_int) \\ [TtC3, Mach\_top] = time\_to\_climb\_seg\_('MACH', ... \\ \qquad\qquad ..., mass, Disa, Zp\_brk, Zp\_max, Mach\_max) \end{cases} \\ \\ Zp\_brk < Zp\_int \ , \ print(\text{"Mach\_max is reached below "altp\_int"m, in time\_to\_climb\_"}) \end{cases}$$

$TtC = TtC1 + TtC2 + TtC3$

Function Name: $time\_to\_climb\_seg\_$	(170)

Input: $Cmode, mass, Disa, Zp1, Zp2, Speed$ — **Ouput** : $TtC, Mach2$

$$\begin{cases} Cmode = "CAS" \begin{cases} Pamb1 = non\_stand\_atmos\_(Disa, Zp1) \\ Mach1 = \sqrt{\left(\left(\left((0.2 \cdot (\frac{Speed}{340.29})^2 + 1)^{3.5} - 1\right) \cdot \frac{101325}{Pamb1} + 1\right)^{1/3.5} - 1\right)/0.2} \\ Pamb2 = non\_stand\_atmos\_(Disa, Zp2) \\ Mach2 = \sqrt{\left(\left(\left((0.2 \cdot (\frac{Speed}{340.29})^2 + 1)^{3.5} - 1\right) \cdot \frac{101325}{Pamb2} + 1\right)^{1/3.5} - 1\right)/0.2} \end{cases} \\ Cmode = "MACH" \begin{cases} Mach1 = Speed \\ Mach2 = Speed \end{cases} \end{cases}$$

$Vz1 = max\left(0.02, climb\_speed\_(Cmode, mass, Disa, Zp1, Mach1,' mcl')\right)$
$Vz2 = max\left(0.01, climb\_speed\_(Cmode, mass, Disa, Zp2, Mach2,' mcl')\right)$
$TtC = ((Zp2 - Zp1)/(Vz2 - Vz1)) \cdot \log(Vz2/Vz1)$

Table A.46: Equations

Formula		Type
$TtcKmtow = 0.97$	(171)	Constant
$HSperfoDisa = 0$	(172)	Constant
$TtcZp = m\_ft(33000)$	(173)	Constant
$TimeToClimb = time\_to\_climb\_(TtcKmtow \cdot MTOW, ...$ $... , HSperfoDisa, TtcZp, CruiseMach)$	(174)	Model

A.1.16 En Route Operation Models

A.1.16.1 Standard missions

Function Name: $mission\_$	(175)

Input: $TOW, fuel\_block, Disa, Zp, Mach$
Ouput : $Fuel, ZFW, Range, Time, Fuel\_div, Time\_div, Fuel\_hld, Time\_hld$

$Zp\_hld = m\_ft(1500)$
$Mach\_hld = 0.60 \cdot Mach$
$Time\_hld = s\_min(30)$
$Zp\_div = m\_ft(25000)$
$Mach\_div = 0.75 \cdot Mach$
$Leg\_div = m\_nm(200)$
$KF\_res = 0.05$
$mass = min(TOW, max(TOW/2, TOW - fuel\_block/2))$
$LDW = TOW - fuel\_block$
$[LoD, Cz, g, SoundSpeed, Pamb, Tamb] = flight\_aero\_data\_(Disa, Zp, Mach, mass)$
$Fnet = (mass \cdot g)/LoD$
$Sfc = engine\_sfc\_(Pamb, Tamb, Mach, Fnet)$
$Range = 0.98 \cdot (SoundSpeed \cdot Mach \cdot LoD)/(Sfc \cdot g) \cdot \log(TOW/LDW)$
$Time = 1.09 \cdot (Range/(Mach \cdot SoundSpeed))$
$[LoD\_div, Cz\_div, g\_div, SoundSpeed\_div, Pamb\_div, Tamb\_div] = ...$

$$flight\_aero\_data\_(Disa, Zp\_div, Mach\_div, LDW)$$

$Fnet\_div = (LDW \cdot g)/LoD\_div$

$Sfc\_div = engine\_sfc\_(Pamb\_div, Tamb\_div, Mach\_div, Fnet\_div)$

$Fuel\_div = LDW \cdot (1 - e^{-\dfrac{Sfc\_div \cdot g}{SoundSpeed\_div \cdot Mach\_div \cdot LoD\_div}} \cdot (leg\_div/0.98))$

$Time\_div = 1.09 \cdot (leg\_div/(Mach\_div \cdot SoundSpeed\_div))$

$[LoD\_hld, Cz\_hld, g\_hld, vsnd\_hld, Pamb\_hld, Tamb\_hld] = ...$

$$flight\_aero\_data\_(Disa, Zp\_hld, Mach\_hld, LDW)$$

$Fnet\_hld = (LDW \cdot g)/LoD\_hld$

$Sfc\_hld = engine\_sfc\_(Pamb\_hld, Tamb\_hld, Mach\_hld, Fnet\_hld)$

$Fuel\_hld = Sfc\_hld \cdot Fnet\_hld \cdot Time\_hld$

$Fuel = (1 + KF\_res) \cdot fuel\_block + Fuel\_hld + Fuel\_div$

$ZFW = TOW - Fuel$

Function Name: fct_miss	(176)

Input: $FuelBlock, Disa, Zp, Mach, TOW, OWE, Payload$ — **Ouput** : $delta$

$FuelTotal = mission\_(TOW, FuelBlock, Disa, Zp, Mach)$

$delta = TOW - OWE - Payload - FuelTotal$

Function Name: $standard\_operation\_$	(177)

Input: $TOW, OWE, Payload, Disa, Zp, Mach$ — **Ouput** : $RA, ZFW, FUEL$

$fct = list(fct\_miss, Disa, Zp, Mach, TOW, OWE, Payload)$

$[FuelBlock, Yres, rc] = fsolve((TOW - OWE - Payload), fct, 0.001)$

$[FUEL, ZFW, RA] = mission\_(TOW, FuelBlock, Disa, Zp, Mach)$

Table A.50: Equations

Formula		Type
$HSperfoDisa = 0$	(178)	Constant
$[NominalMisRange, NominalMisZFW, NominalMisTotalFuel] = ...$ $standard\_operation\_(NominalMisTOW, OWE, NominalPayload, ...$ $..., HSperfoDisa, ZpRef, CruiseMach)$	(179)	Model
$[MaxPlMisRange, MaxPlMisZFW, MaxPlMisTotalFuel] = ...$ $standard\_operation\_(MaxPlMisTOW, OWE, MaxPayload, ...$ $..., HSperfoDisa, ZpRef, CruiseMach)$	(180)	Model
$[MaxFuelMisRange, MaxFuelMisZFW, MaxFuelWeight] = ...$ $standard\_operation\_(MaxFuelMisTOW, OWE, MaxFuelPayload, ...$ $..., HSperfoDisa, ZpRef, CruiseMach)$	(181)	Model
$[ZeroPlMisRange, ZeroPlMisZFW, ZeroPlMisTotalFuel] = ...$ $standard\_operation\_(ZeroPlMisTOW, OWE, 0, ...$ $..., HSperfoDisa, ZpRef, CruiseMach)$	(182)	Model

A.1.16.2 Missions for costing

Function Name: fct_cost	(183)
Input: $X, Disa, Zp, Mach, OWE, Payload, Range$ — **Ouput** : Y	
$[TOW, FuelBlock[= (X(1), X(2))$	
$[FUEL, ZFW, RA[= mission\_(TOW, FuelBlock, Disa, Zp, Mach)$	
$Y = [Range - RA, ZFW - OWE - Payload[$	

Function Name: $cost\_operation\_$	(184)
Input: $Range, OWE, Payload, Disa, Zp, Mach$ — **Ouput** : $TOW, FuelBlock, TimeBlock$	
$fct = list(fct\_cost, Disa, Zp, Mach, OWE, Payload, Range)$	
$[Xres, Yres, rc] = fsolve(OWE \cdot [2.0, 0.5], fct, 0.001)$	
$[TOW, FuelBlock] = (Xres(1), Xres(2))$	
$[FUEL, ZFW, Range, TimeBlock] = mission\_(TOW, FuelBlock, Disa, Zp, Mach)$	

Table A.53: Equations

Formula		Type
$\begin{cases} DesignRange < m\_nm(4500) \ , \ CostMisRange = m\_nm(500) \\ m\_nm(4500) \leq DesignRange \ , \ CostMisRange = m\_nm(4000) \end{cases}$	(185)	Model
$[CostMisTOW, CostMisBlockFuel, CostMisBlockTime] = cost\_operation\_(..$ $CostMisRange, OWE, CostMisPayload, HSperfoDisa, ZpRef, CruiseMach)$	(186)	Model

A.1.16.3 Operating cost

Function Name: $Cash\_Op8cost\_$	(187)
Input: $LaborCost, FuelPrice$	
Ouput : $CashOpCost, FuelCost, FrameMC, EngineMC, CockpitCrew, ...$	
$...., CabinCrew, LDGfees, NAVfees$	
$BH = CostMisBlockTime/3600$	
$Tt = BH + 0.25$	
$WG = MTOW \cdot 1e - 3$	
$Wf = (MWE - EngineMass) \cdot 1e^{-5}l$	
$Th = 0.05 \cdot (SLSThrust/4.4482198) \cdot 1e^{-4}$	
$FuelCost = CostMisBlockFuel \cdot (FuelPrice * 1 \cdot 10^3/FuelDensity)$	
$CockpitCrew = BH \cdot 2 \cdot (440 - 0.532 \cdot WG)$	
$CabinCrew = BH \cdot ceil(TotalNpax/50) \cdot LaborCost$	
$LaborFrame = ((1.26 + 1.774 \cdot Wf - 0.1071 \cdot Wf^2) \cdot Tt + ...$	
$\qquad\qquad + (1.614 + 0.7227 \cdot Wf + 0.1204 \cdot Wf^2)) \cdot LaborCost$	
$MatrlFrame = (12.39 + 29.8 \cdot Wf + 0.1806 \cdot Wf^2) \cdot Tt + ...$	
$\qquad\qquad + (15.2 + 97.33 \cdot Wf - 2.862 \cdot Wf^2)$	
$FrameMC = LaborFrame + MatrlFrame$	
$LaborEngine = Nengine \cdot (0.645 \cdot Tt + Th \cdot (0.566 \cdot Tt + 0.434)) \cdot LaborCost$	
$MatrlEngine = Nengine \cdot (25 \cdot Tt + Th \cdot (0.62 \cdot Tt + 0.38))$	

$$EngineMC = LaborEngine + MatrlEngine$$

$$LDGfees = (MTOW/1000) \cdot 7.5$$

$$NAVfees = (CostMisRange/185200) \cdot sqrt((MTOW/1000)/50) \cdot 77$$

$$CashOpCost = FuelCost + FrameMC + EngineMC + ...$$

$$CockpitCrew + CabinCrew + LDGfees + NAVfees$$

Bibliography

[ANR09]   French National Reasearch Agency ANR. Integrative design for complew system (id4cs). `http://www.irit.fr/id4cs/`, January 2009. COSINUS program, project ANR ID4CS ANR-09-COSI-005.

[AP96]   T.W. Athan and P.Y. Papalambros. Note on weighted criteria methods for compromise solutions in multi-objective optimization. *Engineering Optimization*, 27:155–176, 1996.

[Apo99]   G. Apostalakis. The distinction between aleatory and epistemic uncertainties is important: an example from the inclusion of aging effects in the psa. In *Proceedings of PSA 99, International Topical Meeting on Probabilistic Safety Assessment*, pages 135–142. American Nuclear Society, August 1999.

[App13]   W. Appel. *Probabilités pour les non-probabilistes*. H & K, june 2013.

[Aro11]   J.S. Arora. *Introduction to Optimum Design*. Elsevier Science, third edition, 2011.

[Ass08]   A. Assenza. A comparative study of various probability density estimation methods for data analysis. *International Journal of Computational Intelligence Systems*, 1(2):188–201, May 2008.

[B̈96]   T. Bäck. *Evolutionary algorithms in theory and practice*. Oxford University Press, Oxford, 1996.

[Bad05]   C. Badufle. *Conceptual aircraft design: toward multiobjective, robust and uncertain optimisation*. PhD thesis, Paul Sabatier University Toulouse III, 2005.

[Bau12]   V. Baudoui. *Optimisation robuste multiobjectifs par modèles de substitution*. PhD thesis, Paul Sabatier University Toulouse III, 2012.

[Bea04]   B. Beauzamy. *Méthodes Probabilistes pour l'étude des Phénomènes Réels*. Société de Calcul Mathématique S.A., February 2004.

[Bie06]   M. Bierlaire. *Introduction à l'optimisation différentiable*. PPUR Presses Polytechniques et Universités Romandes, 2006.

[BL97]   J. Birge and F. Louveaux. *Introduction to Stochastic Programming*. Springer-Verlag, 1997.

[BL00]   S.D. Balkin and D.K.J. Lin. A neural network approach to response surface methodology. *Communications in statistics. Theory and methods*, 29(9 and 10):2215–2227, 2000.

[BR00]   R.J. Balling and J. Richard. Pareto sets in decision-based design. *Journal of Engineering Valuation and Cost Analysis*, 3(2):189–198, 2000.

[BS07]   H.-G. Beyer and B. Sendhoff. Robust optimization - a comprehensive survey. *Computer Methods in Applied Mechanics and Engineering*, 196(33-34):3190–3218, July 2007.

[BTN98]   A. Ben-Tal and A. Nemirovski. Robust convex optimization. *Mathematics of Operations Research*, 23(4):769–805, November 1998.

[Bur99]   K Bury. *Statistical Distributions in Engineering*. Cambridge University Press, 1999.

[CC77]   A. Charnes and W. W. Cooper. Goal programming and multiple objectives optimisations. *European Journal of Operational Research*, 1:39–54, 1977.

[CC88]   P. Caperaa and B. V. Cutsem. *Méthodes et Modèles en Statistique Non Paramétrique: Exposé Fondamental*. Dunod, 1988.

[CC10]   C.A. Coello Coello. List of references on evolutionary multiobjective optimization. `http://www.lania.mx/~ccoello/EMOO/EMOObib.html`, february 2010.

[CCC99]   C.A. Coello Coello and A.D. Christiansen. Moses: A multiobjective optimization tool for engineering design. *Engineering Optimization*, 31(3):337–368, 1999.

[CCF55]   A. Charnes, W.W. Cooper, and R.O. Ferguson. Optimal estimation of executive compensation by linear programming. *Management Science*, 1(2):138–151, January 1955.

[CCS58]   A. Charnes, W. W. Cooper, and G. H. Symonds. Cost horizons and certainty equivalents: an approach to stochastic programming of heating oil. *Management science*, 4:235–263, 1958.

[CFS04]   A. Ciroth, G. Fleisher, and J. Steinbach. Uncertainty calculation in life cycle assessments, a combined model of simulation and approximation. *The Internatioanl Journal of Life cycle assessments*, 9(4):216–226, July 2004.

[Cor91]   G. Corradi. Quasi-newton methods for nonlinear equations and unconstrained optimisation problems. *International journal of conputer mathematics*, 38(1-2):71–89, 1991.

[Cre93]   N.A.C. Cressie. *Statistics for spatial data*. John Wiley & Sons, 1993.

[CS97]   V. Czitrom and P. Spagon. *Statistical Case Studies for Industrial Process Improvement*. American Statistical Association (ASA) and Society for Industrial and Applied Mathematics (SIAM), 1997. p. 155-167.

[CS02]   Y. Collette and P. Siarry. *Optimisation Multiobjectif*. Eyrolles, September 2002.

[Dar59]   C. Darwin. *The origin of species*. Dent Gordon, London, 1859.

[DD96]   I. Das and J. Dennis. Normal-boundary intersection: an alternative method for generating pareto optimal points in multicriteria optimization problems. Technical Report 96, Institute for Computer Applications in Science and Engineering NASA Langley Research Center, November 1996.

[DD97]   I. Das and J.E. Dennis. A closer look at drawbacks of minimizing weighted sums of objectives for pareto set generation in multicriteria optimization problems. *Structural Optimization*, 14:63–69, 1997.

[DDST98] Dimitri Mavris Daniel, Daniel A. Delaurentis, Danielle S. Soban, and Abstrac T. Probabilistic assessment of handling qualities characteristics in preliminary aircraft design. In *In 36th Aerospace Sciences Meeting & Exhibit, number AIAA*, pages 12–15, 1998.

[Deb01] K. Deb. *Multi-Objective Optimization using Evolutionary Algorithms*. John Wiley and Sons, 2001.

[Dev86] L. Devroye. *Non Uniform Random Variate Generation*. Springer-Verlag, 1986.

[DMC12] R. Duvigneau, M. Martinelli, and P. Chandrashekarappa. Uncertainty quantification for robust design. In *Multidisciplinary Design Optimization in Computational Mechanics ISTE*. Wiley, 2012.

[Doo05] J.A. Doornik. An improved ziggurat method to generate normal random samples. Technical report, Mimeo, Nuffield College, University of Oxford, 2005.

[DP86] D. Dubois and H. Prade. *Possibility Theory: An Approach to Computerized Processing of Uncertainty*. New York: Plenum Press, 1986.

[DPAM02] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algoritm: Nsga-ii. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.

[Dra95] D. Draper. Assessment and propagation of model uncertainty. *Journal of the Royal Statistical Society*, 57(1):45–97, 1995.

[EcP05] Electricité de France EADS EDF and European Aeronautic Defense & Space company Phimeca. Openturns, scientific library dedicated to the treatment of uncertainties. `http://www.openturns.org.`, 2005.

[Ehr05] M. Ehrgott. *Multicriteria Optimization*. Springer, second edition, 2005.

[EK05] B.R. Ellingwood and J. Kanda. *Structural Safety and its Quality Assurance*. Structural Engineering Institute of the American Society of Civil Engineers, 2005.

[Eur00] CITAC Eurachem. Quantifying uncertainty in analytical measurement. Second Edition, 2000.

[FEHP11] C. Forbes, M. Evans, N. Hasting, and B. Peacock. *Statistical Distribution*. Wiley, fourth edition, 2011.

[FF93] C.M. Fonseca and P.J. Fleming. Multiobjective genetic algortihms. In *Genetic algorithms for control systems engineering, IEE colloquium on*, May 1993.

[Fog95] D.B. Fogel. *Evolutionary computation: toward a new philosophy of machine intelligence*. Wiley- IEEE Press, 1995.

[FOO66] L.J. Fogel, A.J. Owens, and Wals O.M. *Artificial intelligence through simulated evolution*. Wiley, 1966.

[Geb95] A. Gebhardt. *Rapid Prototyping*. Hanser Gardner Publications, 1995. ISBN 9781569902813.

[GLK02] L.G. Green, H.-Z. Lin, and M.R. Khalessi. Probabilistic methods for uncertainty propagation applied to aircraft design. In *20th AIAA Applied Aerodynamics Conference 24-26 June 2002, AIAA 2002-3410, St. Louis, Missouri*, 2002.

[GNG11]   M. Guenov, M. Nunez, and A. Gondhalekar. Comparing design margins in robust and deterministic design optimization. In *Eurogen 2011: Evolutionary and Deterministic Methods for Design, Optimization and Control with Applications to Industrial and Societal Problems*. Cira Italian Aerospace Research Center and University of Trieste (associated to ECCOMAS and ERCOFTAC), September 2011.

[Gol89]   D.E. Goldberg. *Genetic algorithms in search optimization and machine learning*. Addition-Wesley, 1989.

[GS10]   M. Gilli and E. Schumann. A note on 'good starting values' in numerical optimisation. In *COMISEF working paper series*, number WPS-044. COMISEF, September 2010.

[HB96]   J.C. Helton and D.E. Burmaster. Treatment of aleatory and epistemic uncertainty. *Reliability Engineering and System Safety*, 54(2-3):91–262, 1996.

[Hen10]   R. Henrion. Chance constrained problems, August 2010. Pre-conf. PhD Workshop.

[HL92]   P. Hajela and C.-Y. Lin. Genetic search strategies in multicriterion optimal design. *Structural optimization*, 4(2):99–107, 1992.

[HMSW04]   W. Härdle, M. Müller, S. Sperlich, and A. Werwatz. *Nonparametric and Semiparametric Models*. Springer, New York, 2004.

[HNG94]   J. Horn, N. Nafpliotis, and D.E. Goldberg. A niched pareto genetic algorithm for multiobjective optimization. In *Proceedings of the first IEEE conference on evolutionary computation*, volume 1, pages 82–87, Piscataway, New Jersey, 1994. IEEE World Congress on computational intelligence, IEEE Service Center.

[Hol75]   J.H. Holland. *Adaptation in natural and artificial systems*. Ann Arbor: The University of Michigan Press, 1975.

[INR89]   Copyright INRIA. Scilab, open source software distributed under cecill license. Description on `http://www.scilab.org.`, 1989.

[JCS01]   R. Jin, W. Chen, and T. Simpson. Comparative studies of metamodeling techniques under multiple modeling criteria. *Structural and Multidisciplinary optimization*, 23(1):1–13, 2001.

[JDC03]   R. Jin, X. Du, and W. Chen. The use of metamodeling techniques for optimization under uncertainty. *Structural and Multidisciplinary optimization*, 25:99–116, 2003.

[JKB95]   Norman L. Johnson, Samuel Kotz, and N. Balakrishnan. *Continuous univariate distributions. Vol. 2*. Wiley Series in Probability and Mathematical Statistics: Applied Probability and Statistics. John Wiley & Sons Inc., New York, second edition, 1995. A Wiley-Interscience Publication.

[Joh04]   O. Johnson. *Information theory and the central limit theorem*. Imperial College Press, 2004.

[Jur07]   F. Jurecka. *Robust design optimization based on metamodeling techniques*. PhD thesis, Technische Universitat Munchen, 2007.

[KCS06]   A Konak, D.W. Coit, and A.E. Smith. Multi-objective optimization using genetic algorithms: A tutorial. *Reliability Engineering and System Safety (In special issue - Genetic Algorithms and Reliability)*, 91(9):992–1007, 2006.

[KD00]   Z.A. Karian and E. Dudewicz. *Fitting Statistical Distribution: The Generalized Lambda Distribution and Generalized Bootstrap Methods*. Chapman and Hall/CRC Hardcover, May 2000.

[Kim01]   H. Kim. *Statistical Modelling of Simulation Errors and their Reduction via Response surface Techniques*. PhD thesis, Faculty of the Virginia Polytechnic Institute and State University, 2001.

[Kri51]   D.G. Krige. A statistical approach to some mine valuation and alliead problems on the witwatersrand. Master's thesis, University of the Witwatersrand, South Africa, 1951.

[Liu09]   B. Liu. *Theory and Practice of Uncertainty Programming*. Springer, third edition, 2009. http://orsc.edu.cn/liu/up.pdf.

[LS08]   J. Laurenceau and P. Sagaut. Building efficient response surfaces of aerodynamic functions with kriging and cokriging. *AIAA Journal*, 46(2):498–507, 2008.

[LSW04]   J.F. Lehman, T.J Santner, and Notz W.I. Designing computer experiments to determine robust control variables. *Statistica Sinica*, 14:571–590, 2004.

[Lun05]   J. Lundin. On quantification of error and uncertainty in two-zone models used in fire safety design. *Journal of Fire Sciences*, 23(4):329–354, 2005.

[MA04]   R.T. Marler and J.S. Arora. Survey of multi-objective optimization methods for engineering. *Structural and Multidisciplinary Optimization*, 26:369–395, 2004.

[ME98]   W.Q. Meeker and L.A. Escobar. *Statistical Methods for Reliability Data*. John Wiley and Sons, 1998.

[Mes96]   A. Messac. Physical programming: Effective optimization for computational design. *AIAA Journal*, 34(1):149–158, 1996.

[Mie99]   K.M. Miettinen. *Non linear multiobjective optimization*. Boston: Kluwer Academic Publishers, 1999.

[MM02]   A. Messac and C.A. Mattson. Generating well-distributed sets of pareto points for engineering design using physical programming. *Optimization and Engineering*, 3:431–450, December 2002.

[MMU99]   Michael D. McKay, John D. Morrison, and Stephen C. Upton. Evaluating prediction uncertainty in simulation models. *Computer Physics Communications*, 117:44–51, 1999.

[MSC$^+$99]   A. Messac, G.J. Sundararaj, Wei Chen, Wei Chen, Atul Sahai, and Atul Sahai. Physical programming for robust design. In *40th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference and Exhibit, AIAA-99-1206, American Institute of Aeronautics and Astronautics*, 1999.

[MT00]   G. Marsaglia and W.W. Tsang. A simple method for generating gamma variables. *ACM Transactions on Mathematical Software*, 26(3):363–372, Septembre 2000.

[MW65]   L. B. Miller and H. Wagner. Chance-constrained programming with joint constraints. *Operations Research*, 13:930–945, 1965.

[NA03]   T. Nilsen and T. Aven. Models and model uncertainty in the context of risk analysis. *Reliability Engineering and System Safety*, 79(3), March 2003.

[NAT07] NATO, North atlantic Treaty Organisation and RTO, Research & Technology Organisation. *Computational Uncertainty in Military Vehicle Design*, December 2007.

[NIS12] NIST/SEMATECH. e-handbook of statistical methods, 2012. http://www.itl.nist.gov/div898/handbook/.

[NW06] J. Nocedal and S. Wright. *Numerical Optimization*. Springer Series in Operations Research and Financial Enginering, second edition, 2006.

[ODR+02] W.L. Oberkampf, S.M. Deland, B.M. Rutherford, K.V. Diegert, and K.F. Alvin. Error and uncertainty in modelling and simulation. *Elsevier*, Reliability Engineering and System Safety:333–357, 2002.

[Osy78] A. Osyczka. An approach to multicriterion optimization problems for engineering design. *Computer Methods in Applied Mechanics and Engineering*, 15(3):309–333, September 1978.

[Pha89] M.S. Phadke. *Quality engineering using robust design*. Prentice Hall, 1989.

[PNT04] M.M. Putko, P.A. Newmann, and A.C. Taylor. Employing sensitivity derivatives for robust optimization under uncertainty in cfd. In *9th ASCE Specialty conference on probabilistic mechanics and structural reliability*, 2004.

[Pow08] J.L. Powell. Semiparametric estimation. *The New Palgrave Dictionary of Economics*, 2008.

[Pré70] A. Prékopa. On probabilistic constrained programming. In Princeton University Press, editor, *Proceedings of the Princeton Symposium on Mathematical Programming*, volume 33, pages 113–118, 1970.

[Pro09] European Union Project. Collaborative and robust engineering using simulation capability enabling next design optimisation (crescendo). Description on `http://www.crescendo-fp7.eu/`, May 2009.

[Rac01] R. Rackwitz. Reliability analysis - a review and some perspectives. *Structural Safety*, 23(4):365–395, 2001.

[Rao87] S.S. Rao. Game theory approach for multiobjective structural optimization. *Computers and Structures*, 25(1):119–127, 1987.

[Rec71] I. Rechenberg. *Evolutionsstrategie - Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*. PhD thesis, Stuttgart, 1971.

[Ril11] M.E. Riley. *Quantification of Model-Form, Predictive, and Parametric Uncertainty in Simulation-Based Design*. PhD thesis, Wright State University, 2011.

[RO10] C.J. Roy and W.L. Oberkampf. A complete framework for verification, validation, and uncertainty quantification in scientific computing. In *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*. AIAA Aerospace Sciences Meeting, January 2010.

[RR88] J.R. Rao and N. Roy. Fuzzy set theoretic approach of assigning weights to objectives in multicriteria desicion making. *International Journal of Systems Science*, 20:1381–1386, 1988.

[Sap06] G. Saporta. *Probabilités Analyse des Données et Statistique*. Technip, first edition, 2006.

[Sch81]   H.-P. Schwefel. *Numerical optimization of computer models*. Wiley, 1981.

[Sch85]   J.D. Schaffer. Multiple objective optimization with vector evaluated genetic algorithms. In *Genetic algorithms and their applications: proceedings of the first international conference on genetic algorithms*, pages 98–100, 1985.

[SD94]   N. Srinivas and K. Deb. Multiobjective optimization using nondominated sorting in genetic algorithms. *Evolutionary Computation*, 2(3):221–248, 1994.

[Sil86]   B.W. Silverman. *Density Estimation for Statistics and Data Analysis*. Chapman and Hall, 1986.

[SL02]   R. Sarker and K.-H. Liang. A new multiobjective evolutionary algorithm. *European Journal of Operational Reasearch*, 140(1):12–23, 2002.

[Sor80]   H.W. Sorenson. *Parameter estimation: principles and problems*, volume 9 of *Control and systems theory*. Dekker, M., 1980.

[SP$^+$09] L. P. Swiler, T.L. Paez, , R.L. Mayes, and M.S. Eldred. Epistemic uncertainty in the calculation of margins. In *50th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, number AIAA-2009-2249, 4-7 May 2009.

[SS04]   A.J. Smola and B. Schölkopf. A tutorial on support vector regression. *Statistics and Computing*, 14(3):199–222, 2004.

[SWMH89]   J. Sacks, W.J. Welch, T.J. Mitchell, and Wynn H.P. Design and analysis of computer experiments. *Statistical Science*, 4(4):409–423, 1989.

[TC11]   R.M. Torres-Calderon. *Multi-objective environmentally-friendly departure procedures for civil aircraft: modelling, optimization and robustness assessment*. PhD thesis, Paul Sabatier University Toulouse III, 2011.

[TCT00]   G. Taguchi, S. Chowdhury, and S. Taguchi. *Robust Engineering: Learn How to Boost Quality While reducing Cost and Time Market*. McGraw Hill Professional, 2000.

[TNB02]   D.Q. Tao, V.T.V. Nguyen, and A. Bourque. On selection of probability distributions for representing extreme precipitations in southern quebec. In *Annual Conference of the Canadian Society for Civil Engineering*, June 2002.

[Tsy08]   A.B. Tsybakov. *Introduction to Nonparametric Estimation*. Springer, first edition, 2008.

[VGS97]   V. Vapnik, S. Golowinch, and A. Smola. Support vector method for function approximation, regression estimation and signal processing. *Advances in Neural Information Processing System*, 9:281–287, 1997.

[VLC86]   G.H. Vatistas, S. Lin, and Kwok C.K. Reverse flow radius in vortex chambers. *AIAA Journal*, 24(11):1872–1873, 1986.

[Vos00]   D. Vose. *Risk Analysis, A quantitative Guide*. John Wiley and Sons, January 2000. ISBN:978-0471997658.

[War98]   C.A. Ward. Technology tracking for the high-speed reasearch program. Technical report, SAE Technical Paper 985549, 1998.

[Web07]   A. Weber. Study and employment of the monte carlo simulation for the robust design of space structures. In *CEAS European Air and Space Conference*. CEAS European Air and Space Conference, 2007.

[Yan08]   X.S. Yang. *Introduction to Mathematical Optimization: From Linear Programming to Metaheuristics.* Cambridge International Science Publishing, January 2008.

[YH95]   K.P. Yoon and C.-L. Hwang. *Multiple Attribute Decision Making: An Introduction.* Sage Publications, 1995.

[Zdr93]   Z. Zdrahal. Second order measures for uncertainty processing. In *IJCAI'93 Proceeding of the 13th International Joint Conference on Artificial Intelligence*, volume 1, pages 626–631. Morgan Kaufmann Publishers Inc., 1993.

[ZGD95]   K. Zhao, K. Glover, and J.C. Doyle. *Robust and Optimal Control.* Prentice Hall, 1995.

[Zio09]   E. Zio. *Computational Methods for Reliability and Risk Analysis*, volume 14 of *Series on Quality, Reliability and Engineering Statistic.* World Scientific Publishing Co. Ptc. Ltd., 2009.

[ZT99]   E. Ziztler and L. Thiele. Multiobjective evolutionay algorithms: a comparative case study and the strength pareto approach. *IEEE Transactions on Evolutionary Computation*, 3(4):257–271, 1999.

AUTHOR: Jessie BIRMAN

TITLE: Uncertainty quantification and propagation in Conceptual Aircraft Design: from deterministic optimisation to chance constrained optimisation

SUPERVISORS: Jean-Baptiste HIRIART-URRUTY and Sophie JAN

PLACE AND DATE OF THE DEFENSE: Université de Toulouse TOULOUSE (FRANCE), 19/12/2013

ABSTRACT: Conceptual aircraft sizing is the first step in the development project of a passenger transport aircraft. Classically, in this phase, a lot of aircraft configurations are compared after having been globally sized thanks to a deterministic, multidisciplinary and constrained optimisation problem. The purpose is to determine the main characteristics of the airplane according to a set of Top Level Requirements. At preliminary stage, designers have to deal with limited knowledge and high uncertainty when solving this problem. Managing that uncertainty is a major issue: assessing its impact on the design in the early stage allows to save time and cost.

This PhD thesis introduces a new methodology to solve the aircraft design optimisation affected by uncertainty. First of all, the main source of uncertainty involved at this stage is identified as predictive model uncertainty, which is part of epistemic uncertainty. This uncertainty is quantified within a probabilistic framework. For that purpose, based on the Beta distribution, we create a new generic distribution function able to assume a wide range of distribution shapes: it is called Beta-Mystique distribution. Second of all, we realise uncertainty propagation studies with Monte Carlo and moment propagation methods, in order to analyse the robustness of aircraft configuration according to a set of uncertainties. Finally, a chance constrained optimisation is solved to produce a robust aircraft configuration. Two strategies are considered: the use of Surrogate models to approximate the probabilities and the resolution of the optimisation problem thanks to the moment propagation method.

KEYWORDS: Conceptual aircraft design, robust optimisation, chance constrained optimisation, probabilistic constraints, uncertainty

ADMINISTRATIVE FIELD: Applied Mathematics (Optimisation, Aeronautics)

UNIVERSITY ADDRESS: Future Projet Office, Airbus Opération S.A.S., 316 route de Bayonne, 31060 Toulouse cedex 9 and Institut de Mathématiques de Toulouse, Paul Sabatier University, 118 route de Narbonne, 31062 Toulouse Cedex 9

AUTEUR: Jessie BIRMAN

TITRE: Quantification et propagation d'incertitude dans les phases amont de projets de conception d'avions : de l'optimisation déterministe à l'optimisation sous contraintes probabilistes

DIRECTEUR DE THESE: Jean-Baptiste HIRIART-URRUTY et Sophie JAN

LIEU ET DATES DE SOUTENANCE: Université de Toulouse, TOULOUSE (FRANCE), 19/12/2013

RESUME: Le *Conceptual Design* est la première étape d'un projet d'avion de transport de passagers. Classiquement, au cours de ce processus, un grand nombre de configurations possibles sont comparées après avoir été dimensionnées sur la base d'un processus d'optimisation déterministe, multidisciplinaire sous contraintes. L'objectif est de définir les paramètres principaux de l'avion qui répondent à un cahier des charges donné de haut niveau. A ce stade du projet, les ingénieurs doivent résoudre le problème en ayant très peu de connaissances sur le produit final et donc beaucoup d'incertitude. La gestion de l'incertitude est un point crucial : réussir à comprendre au plus tôt l'impact qu'elle aura sur la configuration et les performances de l'avion peut permettre de choisir les configurations qui présentent le meilleur rapport bénéfice sur risque ainsi que de réduire le temps de conception ultérieur et donc les coûts.

Cette thèse introduit une nouvelle méthodologie pour la résolution d'un problème d'optimisation de configuration avion affecté par de l'incertitude. Dans un premier temps, la source principale d'incertitude présente à ce stade du projet est identifiée comme étant de l'incertitude de prédiction des modèles de simulation. Cette incertitude est de type épistémique. Elle est quantifiée à l'aide d'outils probabilistes. Pour ce faire, et en nous inspirant de la loi Béta, nous avons créé une nouvelle loi de probabilité générique, capable de s'ajuster à des distributions de formes très différentes, intitulée distribution Beta-Mystique. Dans un second temps, nous réalisons des études de propagation d'incertitudes à l'aide des méthodes de Monte Carlo et de propagation des moments, afin d'analyser la robustesse d'une configuration avion par rapport à une quantité d'incertitude donnée. Enfin, une optimisation sous contraintes probabilistes est résolue afin de générer des configurations avions robustes. Deux stratégies sont mises en place : l'approximation des contraintes probabilistes à l'aide de surfaces de réponses et la résolution du problème à l'aide de la méthode de propagation des moments.

MOTS CLES: Dimensionnement préliminaire avion, optimisation robuste, contraintes probabilistes, incertitudes

DISCIPLINE ADMINISTRATIVE: Mathématiques Appliquées (Optimisation, Aéronautique)

ADRESSE DE L'UFR OU DU LABORATOIRE: Bureau des Avants-projet, Airbus Opération S.A.S., 316 route de Bayonne, 31060 Toulouse cedex 9 et Institut de Mathématiques de Toulouse, Université Paul Sabatier, 118 route de Narbonne, 31062 Toulouse Cedex 9