Surrogate models coupled with machine learning to approximate complex physical phenomena involving aerodynamic and aerothermal simulations

Romain Dupuis

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par :

Institut National Polytechnique de Toulouse (Toulouse INP)

Discipline ou spécialité :

Dynamique des fluides

Présentée et soutenue par :

M. ROMAIN DUPUIS

le lundi 4 février 2019

Titre :

Surrogate Models Coupled with Machine Learning to Approximate Complex Physical Phenomena Involving Aerodynamic and Aerothermal Simulations

---

Ecole doctorale :

Mécanique, Energétique, Génie civil, Procédés (MEGeP)

Unité de recherche :

Centre Européen de Recherche et Formation Avancées en Calcul Scientifique (CERFACS)

Directeur(s) de Thèse :

M. JEAN-CHRISTOPHE JOUHAUD

M. PIERRE SAGAUT

Rapporteurs :

M. CHRISTOPHE CORRE, ECOLE CENTRALE DE LYON

Mme MARIA VITTORIA SALVETTI, UNIVERSITA DI PISA

Membre(s) du jury :

M. ANGELO IOLLO, UNIVERSITÉ DE BORDEAUX, Président

M. CLEMENT LOURIOU, AIRBUS FRANCE, Invité

M. JEAN-CHRISTOPHE JOUHAUD, CERFACS, Membre

Mme ANNE GAZAIX, IRT SAINT-EXUPERY, Invité

M. PIERRE SAGAUT, AIX-MARSEILLE UNIVERSITE, Membre

M. PIOTR BREITKOPF, UNIVERSITE DE COMPIEGNE, Membre

Contents

Remerciements

Je souhaite remercier en premier lieu Mme. Maria Vitoria Salvetti et M. Christophe Corre d'avoir accepté de rapporter ce travail de thèse, ainsi que M. Piotr Breitkopf et M. Angelo Iollo pour leur rôle d'examinateur dans ce jury de thèse. Je vous remercie aussi pour vos commentaires et remarques constructives adressés durant la soutenance.

Je n'aurais pas pu démarrer cette thèse sans Anne Gazaix qui m'a accueilli dans l'équipe du projet MDA-MDO à l'IRT Saint Exupéry. Ainsi, je tiens à l'en remercier très chaleureusement. De même, j'adresse mes remerciements les plus sincères à mon directeur de thèse, Jean-Christophe Jouhaud, qui a su me guider, me conseiller et se rendre disponible au quotidien lors de ces trois années, ainsi que mon co-directeur de thèse, Pierre Sagaut, pour ses précieux conseils et les différentes remarques constructives qu'il a pu apporter à mon travail. Ce duo m'a permis d'avancer sereinement durant ces trois années.

Lors de cette thèse, j'ai principalement partagé mon temps entre l'IRT et le CERFACS. J'aimerais remercier l'ensemble des personnes et des collègues que j'ai pu côtoyer au sein de l'IRT Saint-Exupéry. Je pense notamment à Pierre-Jean, mon co-bureau, pour avoir égaillé les journées de travail parfois longues, à Damien pour m'avoir aidé pendant de longues heures à maitriser les différents outils CFD et pour m'avoir très souvent accompagné durant mes sorties running, à Vincent Ambert le gars du sud toujours motivé pour une pause, à Vincent Gachelin et Nicolas, les compagnons du midi, à François, je garderai un très bon souvenir de notre conférence en Floride, à Benoit toujours en vacances et aux ONERA guys (Nathalie, Thierry et Rémi) pour les échanges très conviviaux.

Bien entendu, mes remerciements s'adressent aussi à l'équipe CFD du CERFACS qui m'a accueilli très facilement même si je n'étais pas forcement très présent. En particulier, merci à Pamphile et Robin pour les multiples discussions dans leur bureau ainsi qu'à Mathieu pour les nombreux moments à parler de tennis ou d'autres sujets. Merci aussi aux différents thésards que j'ai pu croiser durant ces trois années d'avoir rendu mes passages aux CERFACS toujours agréables : Valentin, Gauthier, Bastien, Maxime, et je suis sur d'en oublier plein d'autres. Bien entendu, toute ma gratitude va à l'équipe CSG d'une efficacité redoutable pour résoudre les problèmes rencontrés au quotidien sur les machines et toujours disponible pour nous aider. Enfin, je remercie Marc pour son aide précieuse dans la résolution de mes multiples problèmes avec elsA, je remercie Jean-François Boussuge de m'avoir accueilli dans l'équipe aéro, ainsi que l'équipe administrative, notamment Marie du coté CFD et Chantal, avec elles les formalités administratives étaient toujours d'une

simplicité enfantine.

Lors de ma thèse, j'ai eu l'opportunité de passer plusieurs mois au sein d'Airbus dans le département thermique. Tout d'abord, je voudrais très chaleureusement remercier Vincent Rebeyrotte qui a permis de rendre possible ce séjour à Airbus. Merci pour ton implication et ta motivation malgré tous les obstacles administratifs que pouvait représenter ma venue. Merci à Thomas Spoerry de m'avoir accueilli dans l'équipe. Je remercie aussi Bruno Esteve de s'être intéressé à mes travaux et d'y avoir participé. Naturellement, merci à Clément Louriou de m'avoir accompagné sur les aspects techniques et guidé à travers la complexité que pouvait représenter la réalité industrielle. Enfin, merci à mes collègues de quelques mois (Patrice, Yannick, Jérémy, Kevin et Madhu) pour avoir rendu mon passage à Airbus très plaisant.

Evidemment, j'adresse un énorme merci à l'équipe des semi-croustillants ! Vous avez rendu ces trois années à Toulouse vraiment superbes. Qui aurait cru que j'allais aimer les quizz (surtout les blind tests...) ? Merci à vous tous : Charlie avec tes lessives interminables, Thomas et Marie nos compagnons d'aventure exilés à Dreux (la tristitude ?), Aurélien pour ton grand cœur hébergeant deux chômeurs-SDF pendant un mois et demi, Thibault notre belge préféré, Xou avec ton petit coté québécois et mélomane, JB toujours motivé pour donner un coup de main, Théo et Elisa la touche hispano-italienne, Laure notre doyenne (merci pour les corrections d'anglais bien sûr !), Margaux un petit Time'up ?, Florent et Marine toujours partant pour une bonne ripaille et Tyseboob qui peut pas car il a patin. Merci aussi à Enzo, de Cranfield à Toulouse, c'est vraiment super de partager une bière ou une sortie running.

Merci à tous les Courdimanchois de rendre les nombreux retours sur Paris aussi agréables et attendus. Merci à Valentin, Kice, Chrystelle, Sylvain, Elise, Nicolas et Mathilde. Merci en particulier à Kevin d'avoir fait le déplacement jusqu'à Toulouse pour encourager un futur collègue docteur.

Je voudrais maintenant remercier toute ma famille. Merci à mes parents, Chantal et Michel, de m'avoir soutenu dans tous mes choix et mes étapes, c'est grâce à vous que j'en suis arrivé là. Merci à ma sœur et à Adnan, c'est toujours très agréable de se retrouver sur Toulouse, la plupart du temps autour d'un verre ! Merci aussi à, si j'ose dire, mes beaux-parents, Alain et Isabelle, pour leur présence et leur soutien.

Enfin, mes derniers mots vont bien entendu à Lucie. Merci de m'avoir accompagné partout en France (et en Europe maintenant !). Merci pour ton affection, ton soutien durant cette thèse et pendant ces onze années passées ensemble. Simplement merci de te tenir à mes côtés au quotidien.

General introduction

Global air passenger traffic demand has widely increased over the last decade and tended to double over twenty years [157]. Heavy constraints are imposed on the aviation, such as the reduction of pollutant emissions and noise levels, the increasing demand for safety and security, the overall cost reduction target, and the passenger demands for comfort [10]. Consequently, airlines have launched major fleet extensions, renewals, and modernization programs to meet clients' and authorities' requirements. Aircraft manufacturers have been put under intense pressure by this growing need of new aircraft and face major challenges in terms of manufacturing processes, supply chains, and engineering. In the context of a worldwide competition, innovations appear as mandatory in order to address these challenges, such as the improvement of existing technologies and technological breakthroughs.

As regards the aircraft design phase, innovative methodologies can become a significant asset. They represent a major objective for the aircraft industry, as they aspire to reduce the lead-time of the aircraft development in order to remain competitive, while also improving the overall quality of the aircraft, especially in terms of fuel consumption, noise and price. In particular, the numerical simulation has become a key element in the design process, complementing physical tests and judicious flight tests. It can provide a substantial assistance in the exploration of tradeoffs and it helps to choose the design path among various candidates.

Currently, the continuous computing growth is supporting the development of modern physical models, efficient optimization algorithms, and accurate numerical methods. However, the computational cost remains a significant obstacle to an intensive use, motivating the use of surrogate models. The latter approximate complex numerical simulations by learning their behavior based on information provided by a set of computations. Added to this, the sudden surge of interest in machine learning technologies offers attractive opportunities to improve surrogate model techniques in order to reduce design times and to increase the quality of the aircraft. This Ph.D. thesis is devoted to the development of new surrogate models capable of generating global approximations of complex aircraft simulations, in particular for aerodynamics and aerothermal analysis.

Introduction générale

Le trafic aérien civil a largement augmenté au cours de la dernière décennie et tend à presque doubler en vingt ans [157]. D'importantes contraintes sont imposées à l'aviation, telles que la réduction des émissions polluantes et des niveaux sonores, la demande croissante de sécurité et de sûreté, l'objectif global de réduction des coûts et les exigences de confort des passagers [10]. Par conséquent, les compagnies aériennes ont lancé d'importants programmes d'extension, de renouvellement et de modernisation de leur flotte afin de répondre aux besoins des clients et des autorités. Les constructeurs ont du répondre à ce besoin croissant de nouveaux avions et se sont confrontés à des défis majeurs en termes de processus de fabrication, de chaînes d'approvisionnement et d'ingénierie. Dans le contexte d'une concurrence mondiale, les innovations apparaissent comme obligatoires pour relever ces défis, tels que l'amélioration des technologies existantes et les innovations technologiques.

En ce qui concerne la phase de conception de l'avion, des méthodologies innovantes peuvent devenir un atout important. Ils représentent un objectif majeur pour l'industrie aéronautique, car ils visent à réduire les délais de développement des avions afin de rester compétitifs, tout en améliorant la qualité globale de l'appareil, notamment en termes de consommation de carburant, de bruit et de prix. En particulier, la simulation numérique est devenue un élément clé du processus de conception, complétant les essais physiques et les essais en vol. Elle peut fournir une aide substantielle dans l'exploration des compromis et elle aide à choisir certains concepts parmi divers candidats.

Actuellement, la croissance continue des ressources informatiques soutient le développement de modèles physiques modernes, d'algorithmes d'optimisation efficaces et de méthodes numériques précises. Cependant, le coût de calcul reste un obstacle important à une utilisation intensive, motivant l'utilisation de modèles de substitution. Ces derniers remplacent les simulations numériques complexes par des approximations mathématiques apprenant leur comportement à partir des informations fournies par un ensemble de calculs. De plus, l'intérêt soudain pour les technologies d'apprentissage automatique offre des possibilités intéressantes pour améliorer les techniques des modèles de substitution afin de réduire les temps de conception et d'augmenter la qualité de l'avion. Cette thèse de doctorat est consacrée au développement de nouveaux modèles de substitution capables de générer des approximations globales de simulations d'avions complexes, en particulier pour l'aérodynamique et l'analyse aérothermique.

Part I

Introduction

Table of Contents

CHAPTER 1

Context

Résumé

Le trafic aérien de passagers au niveau mondial a plus que doublé sur les vingt dernières années. Les compagnies aériennes ont lancé un large renouvellement de leurs flottes afin de réduire leurs coûts, mais aussi pour satisfaire aux contraintes grandissantes sur les émissions polluantes et sonores des avions. Ainsi, les constructeurs aéronautiques ont dû mettre en place des outils innovants dans les phases de conception afin de répondre pleinement aux demandes des compagnies, notamment en ayant largement recours à la simulation numérique.

Dans cette thèse, un intérêt particulier est porté à deux disciplines intervenant dans la conception : l'aérodynamique et l'aérothermie. Elles profitent grandement des apports de la simulation numérique. Pour l'aérodynamique, des codes de mécanique des fluides numérique peuvent prédire la traînée ou encore la charge sur la voiture pendant la mission. De même, la simulation est très utile pour obtenir le comportement thermique d'une pièce.

Les outils numériques haute-fidélité nécessaires à ces simulations ont généralement un coût de calcul très important, rendant leur usage intensif impossible, notamment pour décrire l'enveloppe de vol. C'est pourquoi des modèles de substitution (surrogate models en anglais) ont été développés pour approximer les codes de calculs coûteux, comme les méthodes de régression avec le Kriging ou les méthodes multi-fidélité. Parmi les différentes méthodes, on s'intéresse dans cette thèse aux modèles réduits par Décomposition Orthogonale aux valeurs Propres (Proper Orthogonal Decomposition (POD) en anglais) : celle-ci permet de prédire des champs complets en fonction de plusieurs paramètres d'entrée.

Les modèles de substitution peuvent rencontrer des problèmes lorsque le régime physique de l'écoulement varie significativement. Plusieurs pistes sont envisagées pour les améliorer. L'une d'entre elles est le recours à l'apprentissage automatique (machine learning en anglais) : il permet d'extraire des connaissances supplémentaires à partir des données de simulations. Certain modèles de substitution appartiennent à l'apprentissage automatique à proprement

parler, mais ils peuvent être complétés, voire couplés à diverses méthodes afin d'améliorer leur capacité prédictive.

Ce manuscrit apporte plusieurs contributions au domaine des modèles de substitutions, notamment :

- une revue des méthodes de substitution non-intrusives ;
- le développement de modèles de substitution locaux ;
- l'application de la méthode locale à un problème aérodynamique avion complet ;
- l'application de modèles de substitution à un problème aérothermique mât moteur.

1.1 Aircraft design

The aircraft design is a long-range work. It can generally be divided into four phases: the conceptual design, the project definition, the detailed design, and the final phase [122]. The first and last phases are usually carried out sequentially, while the project definition and detailed design require frequent exchanges to iterate over the tradeoff studies. One can note that the manpower loading varies significantly with respect to the phase.

The first phase – the conceptual design – generates the general aircraft specifications from the customer requirements and the market surveys. If the initial design is accepted, the project definition – the second step of the aircraft design – carries on with various extensive analyses: wind-tunnel testing, fluid analysis, structural analysis, definition of the systems, weight estimation, manufacturing planning, etc. In the detailed design phase, the peak of manpower is reached. The objective is to complete the detailed design of all the systems and to test them. As regards the manufacturing part, the assembly line is almost achieved. The final phase corresponds to the certification. Several aircraft are manufactured in order to undergo rigorous ground tests, flight tests, and certification campaigns, while the ramp-up to full production is carried out.

Each phase needs design tools with the required precision and an acceptable processing time. The best tradeoff between precision and time can be seen as fundamental in the aircraft design, as the preliminary design phase tests numerous configurations considering only major disciplines. Thus quick and low-fidelity tools are used at this stage, whereas the detailed design phase takes into account the interaction between several disciplines. Consequently it requires high-fidelity tools. In addition, real-time and accurate capabilities can be very useful to support flight test and certifications.

In this context of aircraft design, numerical simulation tools were quickly identified as very promising. They are well suited to the different phases described above, as they provide physical models with various fidelity levels, implementation complexity, and costs. Moreover, the increase of the computing resources through the development of High Performance Computing (HPC) promises a significant enhancement in aircraft design over the next decades [2].

1.2 Numerical simulations

In the past decades, simulation tools gained in maturity, reliability, and accuracy. The aircraft industry has capitalized upon its extensive experience and best practices. Thus, a full virtual aircraft capable of simulating flight tests now represents the final goal of the numerical simulation [2, 121]. All the interactions between the disciplines will be represented in a computer environment generating relevant data to support real flight tests and certification. This concept is considered by Airbus as one of the main long-term objectives for aircraft development [2]. The current use of two specific disciplines of interest in the thesis are detailed in an industrial context: aerodynamics and thermal engineering.

During a flight, an aircraft encounters various flow conditions, such as variations in air density, speed, engine thrust, or pitch. Figure 1.1 illustrates a typical mission profile of a long-range aircraft. Naturally, the external environment is completely different during takeoff, cruise conditions or climb. In addition, the design of the aircraft must be valid not only for a single mission, but also for different mission profiles depending mainly on the aircraft weight and its range. Besides, even if the aircraft follows routine flights, the air traffic control can modify its trajectory to face the air traffic demand or weather problems.

Ideally, an important amount of parameters representative of the flight missions must be taken into account during the aircraft design. Nevertheless, all the possible combinations are impossible to consider with the current high-fidelity tools given the dimension of the flight envelope and the computational cost of these tools.

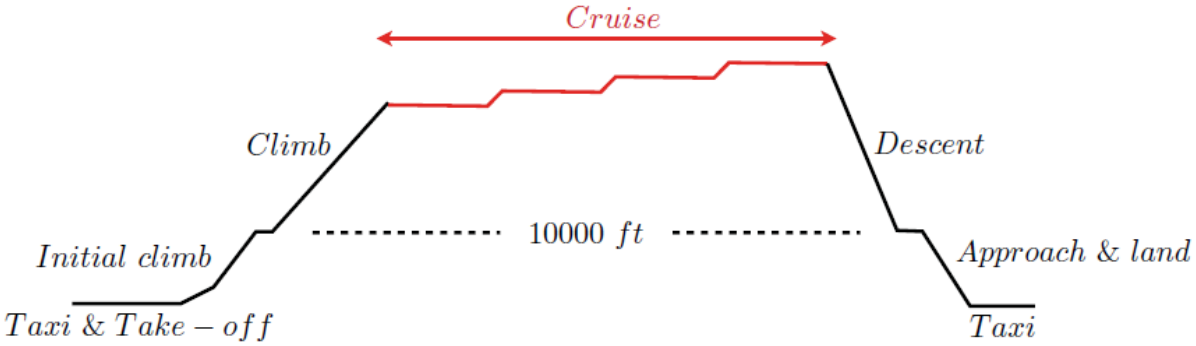


Figure 1.1: Typical mission profile for a long-range civil transport aircraft [79].

A classical solution consists in considering only a few specific flight conditions, for example the cruise condition where the aircraft spends the majority of the time. In the context of aerodynamic shape optimization, this strategy represents a first step to reduce the drag [139]. However, the optimal shapes obtained can be very sensitive to slight modifications of the design point. As a consequence, a single design target does not necessary reflect the real operation of an aircraft: the behavior of an aircraft and its performance can be completely different at off-design conditions. Therefore, it appears as mandatory to be able to explore the flight envelope to improve the representativity of a whole flight.

In the optimization literature, the multi-point approach has been developed to cope with this problem [170, 154, 113, 79]. It consists in simulating several flight conditions representative of a typical mission, leading to the rise of the computational cost. The latter remains manageable as the number of considered points is generally less than a dozen.

However, in the context of flight envelope exploration, the computational cost appears as intractable. For example, the search of critical cases for the load envelope can require more than $100,000$ simulations [193]. Similarly, a full thermal analysis of a specific component of all the aircraft flights can require thousands of computations. For these reasons, the data are generated mainly from wind tunnel testing, empirical, and hand-book methods. A solution that decreases the numerical cost consists in reducing the fidelity of the model. Instead of resolving high-fidelity equations, a simpler problem can be used instead; however several physical phenomena are not solved anymore. The tradeoff between accuracy and computational cost may become quickly unfavorable.

Another possible solution, investigated in this Ph.D. thesis, consists in replacing the expensive high-fidelity model with a surrogate model in order to reduce the computational costs generated by the predictions of the quantities of interest over the parameter space. This substitution is achieved naturally at the expense of model precision and the main priority is the tradeoff between the precision and the computational cost.

1.3 Aerodynamic design

The prediction of the airplane aerodynamic behavior plays a key role in the overall performance. Since the first use of computational fluid dynamics (CFD) for aerodynamic design in the 1960s, aerospace industry has been relying increasingly on numerical simulations. The latter were first deemed as a useful complement to wind-tunnel experiments and flight tests. Over the years, simulations have occupied a prominent place in all stages of the design of an aircraft [197], since the industry has identified CFD as capable of adding significant values to the development of new aircraft by shortening the design process, the time to commercialize being critical for aircraft manufacturers [106]. Figure 1.2 provides a schematic view of the use of CFD at Airbus in 2010. A frequent use is observed for critical parts, such as the wing and nozzle design, the aero-loads data, or the flutter prediction.

Hence, CFD and aerodynamic tools are employed in the daily development work of aircraft. Methods based on the Reynolds-averaged Navier-Stokes equation (RANS) represent the current standard to support the design of complex systems [2]. Moreover, the use of CFD has opened up further opportunities in multidisciplinary simulations. For instance, the aerodynamic loads have a wide impact on the aircraft structure. Conversely, the deformation of the structure modifies the aerodynamic performance of the aircraft. Thus, CFD simulations can be coupled to computational structural mechanics models to improve the understanding of aircraft behavior during various flight maneuvers.

The important computational cost of simulations on complex configurations remains an obstacle to an intensive use, mandatory for a detailed design analysis. Designers need and already use tools able to quickly approximate the skin pressure fields of the aircraft. These tools are mainly based on empirical analysis and surrogate models and they can demonstrate weaknesses. For this reason, surrogate models in the context of aerodynamics are of special interest in this Ph.D, in particular the capability of generating aero-loads data for various flight missions, which is of possible relevance to the industry.

Figure 1.2: Use of CFD at Airbus in 2010. Credits from Abbas-Bayoumi and Becker [2].

1.4 Thermal design

The aircraft design process requires a detailed knowledge of the thermal environment, as the temperatures are of great importance to aircraft engineering. For instance, they play a crucial role in structural deformation due to thermal loads, hot spots insulation, cooling system design, and fatigue [64]. Thermal engineers interact widely with other departments, in particular structural engineering, aerodynamics, equipment, and system design. Therefore, an accurate prediction of the temperature appears as a mandatory step towards the deployment of the modeling of the whole aircraft.

Methods that rely on thermal finite elements represent the main numerical modeling technique in the aircraft industry for the prediction of temperature fields. They have emerged in the 1990s to support the design of specific components, such as the nose fuselage. Then, they have been integrated into more global models including in particular fluids, structures, and systems of the aircraft. This family of methods has been used in the design of the Airbus A350XWB [64].

One of the main priorities of new civil aircraft concepts is to reduce the fuel consumption and greenhouse gas emissions. Increasing the size and the bypass ratio of aircraft engines appears as one of the best solutions. This type of engine architecture improves the thermodynamic cycles efficiency of the propulsion system and reduces pollutant emissions. However, the integration of high engine bypass ratios beyond 15 represents a complex case. The diameter of these engines is highly increased, which leads to a closer positioning to other parts of the airplane, in particular the wing, fuselage and tailplane. The integration is highly impacted and modifies the thermal environment. For instance high lift devices immersed in the engine jet are subjected to vibrations and thermal stresses that influence their fatigue sizing.

The pylon plays a key role in the integration of the engine. For these reasons, a particular

Figure 1.3: Pylon illustration.

interest will be paid to its design. The thermal response of the pylon is directly driven by the engine jet and the flight conditions, namely speed, altitude, and thrust. Thus, several parameters varying over a wide range have to be taken into account in the thermal simulation of the pylon. For example, during takeoff the engines run at full throttle and the aircraft is at low speed and altitude, while during the cruise phase the aircraft almost reaches its maximal speed and altitude with the engines providing low thrust. In conclusion, the pylon design represents an interesting application involving several disciplines of relevance to the industry. This study is particularly interested in thermal engineering and aerodynamics.

The thermal model over full flight missions requires to properly predict thermal inertia. Several inherent limitations due to the respect of industrial constraints are observed. First of all, the data exchange with other disciplines can lead to important computational costs. The input data of the thermal simulations take different forms such as geometry, external aerothermal fields, or engine conditions. These intensive physical interactions require an efficient workflow to exchange data, in particular for varying inputs. Moreover, overall process needs to be quick to evaluate in order to reach tractable computational cost.

Therefore, the data obtained from the different physical models can be simplified: mesh coarsening, boundary conditions simplifications, or neglected input parameters. However, setting the right fidelity target while ensuring that the model would stay predictive remains a tricky question. Moreover, the thermal model is usually validated against the flight test measurements. The specifications of the numerical models should reflect the real aircraft test data: the flight test mission may change compared to the mission used for the sizing, external conditions may differ, or some test components may show differences with the components of the model.

Accordingly, data-driven methods, such as surrogate models, can be considered: to alleviate the numerical burden, to explore a wide range of input parameters, and to adjust the model with flight tests. For instance, such methods build approximations at various levels of fidelity and can generate boundary conditions for the model over different flight missions. Moreover, they represent an interesting tool for collaborative engineering: heterogeneous codes can be avoided, confidentiality can be ensured, high-speed data exchanges are provided, and remote calculations

are allowed. Ideally, adaptive techniques compensate the weaknesses of the data-driven model or of the surrogate model by learning from data generated during the last steps of the design process.

1.5 Surrogate models

As suggested in previous sections, specific tools are needed to reduce the computational time in the design process in mono- and multidisciplinary environments. A common problem in science and engineering consists in solving partial differential equations. They can be discretized and solved with classical methods such as the finite difference or finite volume methods, leading to a high-dimensional input-output problem. It is a classical problem encountered in CFD, where the discretized system can have a large number of degrees of freedom.

The surrogate model substitutes the discretized physical model with a continuous mathematical approximation built from a sampling of simulations. The initial physical model usually has a high complexity and an important computational cost due to its dimension. The mathematical approximation can then be used to analyze the physical model at a moderate cost, as numerous evaluations of the low-cost models can be computed to fulfill different goals: the validation of the model, the prediction of the physical model behavior at unknown operating conditions, the optimization of the system, the sensitivity analysis, or the control of the system [116]. Obviously, the surrogate model does not approximate the physical model perfectly and one of the main difficulties is to handle the tradeoff between the precision of the surrogate model and an acceptable computational cost. A very classical classification divides the surrogate into three kinds [59]:

- **Data-fit surrogate models**: data-fit surrogate models are also called response surfaces or metamodels. They build an explicit non-physics-based relation between the input parameters and the output of the system. The term data-fit refers to the notions of interpolation and regression and usually means that the output is scalar. In other words, the data-fit model is classically built from an integrated value of a high-dimensional quantity of interest. From a statistical point of view, the regression aims at explaining the behavior of a random variable with respect to other random variables by analyzing the data. The relation between these variables can be parametric, such as a linear model, or non-parametric, such as Gaussian Process regression;
- **Reduced-order models (ROMs)**: this term refers mainly to Projection-Based ROMs. The central idea is to project the system of differential equations governing all the degrees of freedom onto a set of basis functions of smaller dimension. Projection-based methods have the advantage to retain some of the physics from the governing equations and to give rigorous error bounds and error estimation [192]. ROMs have been designed for modeling and controlling large-scale dynamical systems of large scale and various complexities, and involving various physical phenomena, such as heat transfer, fluid dynamics, or electronic circuits. They approximate the full-order models with low-dimensional models that are very quick to evaluate. ROMs have shown great capabilities in modeling non-parametric systems. However, research for parametric model reduction is still very active [15]. Different methods exist to compute the basis of the projection, such as the Krylov-subspace methods [85, 47] or the Proper Orthogonal Decomposition [44];
- **Multi-fidelity models**: the physical problems can be modeled at different levels of fidelity. For example, different approximation assumptions can be considered: simplification of the governing equations [98], different mesh refinements [127], or different convergence tolerances [72]. The multi-fidelity surrogate models handle these simulations with different accu-

racies and computational costs in order to achieve the approximation with the best trade-off [89, 164].

The choice between the different kinds of surrogate models is driven by the structure of the problem [15]. In the context of aerodynamic analysis, many data-fit models have been built from integrated values such as the drag or the lift. However, the development of efficient ROMs gives the possibility to improve the physical meaning of the surrogate model and to directly generate a full field of the quantity of interest, opening up further opportunities to build multidisciplinary simulations. Data-fit and ROM approaches are detailed in the following sections. As regards the multi-fidelity approaches, they are not considered in this thesis, as low-fidelity models are not necessary available or relevant.

1.5.1 Data-fit surrogate models of computer experiments

Data-fit surrogate models have been initially designed to assess and extend scalar values from results of physical experiments, usually with a polynomial approximation. As the name implies, data-fit surrogate models are non-physics-based models and are built from a purely mathematical point of view based on the available data.

They have been quickly generalized to computer experiments thanks to their capability to build quiete accurate and relatively inexpensive mathematical models of long-running computer codes, as stated in the paper on design and analysis of computer experiments by Sacks et al. [175]. As various applications require many evaluations of a computational model, such as predictions in a design space or optimization, data-fit surrogate models quickly became a popular method to approximate scalar values from complex and expensive problems. In particular, the engineering design of systems has provided an ideal context for the data-fit surrogate models and the engineering community has greatly benefited from their developments.

Over the last two decades, a unified formalism has been given to the data-fit surrogate models, mainly through the works of Jones et al. [108], Simpson et al. [178], Kleijnen [116], Forrester et al. [73], or Gorissen et al. [82]. Various interpolation and regression models have been developed to improve the accuracy of the surrogate models. In particular, the Kriging method, also known as Gaussian Process Regression, has been widely studied and extended. It was designed to interpolate highly nonlinear functions by computing sensitivities to the input parameters. Kriging also provides an estimation of the uncertainty of each prediction. Other data-fit surrogate model methods can also be cited, for example Support Vector Machines, Radial Basis Function, Polynomial Chaos, or Neural Networks. However, the intelligent selection of data-fit surrogate models remains an open question and depends on the nature of the problem [73, 195].

Data-fit surrogate models have been applied successfully on various engineering problems, mainly in aerospace and to a lesser extent in other fields such as water modeling [169]. Among the many applications for data-fit surrogate models, one can cite for instance Simpson et al. [179] who compare low-order polynomial regression and Kriging for the design of an aerospike nozzle. This pioneering work demonstrates the capability of Kriging for modeling deterministic computer experiments on the basis of a three-variable engineering problem that mixes CFD and a finite element model.

Moreover, optimization processes based on data-fit surrogate models have become popular for problems with a limited number of input parameters, usually less than a dozen. The work of Jones et al. [108] on the Efficient Global Optimization (EGO) gave an alternative to the classical optimization algorithm, usually based on quasi-Newton methods, very useful for aerodynamic optimization [37, 72, 187, 98, 102, 14].

1.5.2 Proper Orthogonal Decomposition and ROM

Proper Orthogonal Decomposition (POD) is an efficient technique for dimension reduction based on spectral decomposition for high-dimensional data set. A wide range of applications can be found in the literature, such as human face characterization [114], data compression [9] or optimal control [134]. The POD term was coined by Lumley [138] in 1967 for the extraction of coherent structures with his work on dominant turbulent eddy. The theory has also been developed in the context of approximation methods [34]. POD is also known as Karhunen-Loève Decomposition, Hotelling Analysis or Principal component analysis, depending on the field of application.

Within the context of fluid mechanics, POD was mainly used to analyze flow structures. Sirovich [180] employed this method to define a unified framework between turbulence theory, coherent structures and low-dimensional manifolds. Kirby et al. [115] observed the distribution of the different features in term of energy ratio from a sampling of a supersonic shear layer. They noticed that the main features of the flow were well captured by considering 80% of the energy, while small-scale structures required to consider the low energy terms of the decomposition in order to be captured. The capabilities of POD have also been successfully applied to image and flow reconstruction. Sirovitch and Everson [66] proposed to use the modal expansion of the POD in order to reconstruct a full signal from an incomplete data set. The method, called Gappy POD, was applied on a NACA 012 airfoil [25]. The POD bases are computed with 51 snapshots and an incomplete signal pressure, obtained with only 121 pressure values (against the 6369 values needed to build the full signal), is successfully reconstructed by Gappy POD. Three different methods can be identified to build a ROM with POD: projection, residual, and interpolation. They are briefly described in the following paragraphs.

1.5.2.1 Projection-based ROM

As mentioned in the previous section, a reduced set of equations can be obtained after the projection (Galerkin) on a reduced basis. The latter is built by computing snapshots at different time steps. In the case of fluid and aeroelastic simulations, such ROMs show very accurate results in the prediction of the transient response of complex systems, such as a cascade of airfoils [60] or a complete fighter jet [130]. These examples even capture complex patterns characterizing transonic regimes.

However, the ROM is built at given parameters (freestream conditions, modeling parameter, etc.) and fails to predict the transient response at different operating points, such as various Mach numbers or angles of attack, without reconstructing a new ROM for each change in the flow conditions, which leads to a dramatic increase of the computational cost. Interpolation methods have been tried on ROMs in order to take into account the dependence of the system matrices on the parameters and therefore reducing the computational cost of building the ROM [49]. Amsallem and Farhat [6] proposed a modification of the classical interpolation method, based on Grassman manifold interpolation, to adapt properly the POD basis. They show a successful application of an aeroelastic ROM for a varying freestream Mach number on F-16 and F-18/A fighters.

1.5.2.2 Method of residual minimization

A different method than the Galerkin projection employs the concept of residual minimization to compute the parameters of the ROM. The method of residual minimization of the Euler equations has been used to build ROM for the NACA0012 airfoil based on the DLR TAU code at subsonic and transonic conditions [214]. The results are compared with other methods, such as Kriging interpolation, and show that the residual minimization captures well the shock in the transonic case at

extrapolatory conditions. Full-order simulations can also be initialized by predictions coming from ROMs built by residual minimization. The application to a high-lift configuration demonstrates a significant CPU speed-up for converged simulations [151].

Different kinds of residual definitions can be considered. Alonso et al. [4] used RANS simulations to build the ROMs and solved a residual minimization based on the Euler equations in order to be independent of the numerical scheme. A specific treatment was used for the shocks and the method was applied on two-dimensional airfoils and a three-dimensional wing at low transonic Mach number [5]. The definition of the residual expression plays a crucial role in the behavior of the system. Amsallem et al. [8] demonstrated the link between the residual definition, the mesh, and the type of flow. It was shown that nonlinear CFD problems with shocks can be well managed by minimizing a specific residual.

1.5.2.3 Interpolated ROM

Rather than minimizing the residual or building the ROM by Galerkin projection, the POD coefficients can also be computed by interpolations or regressions. Such methods do not need any knowledge of the physical systems and do not modify the source code of the computational code. Transient aerodynamics with structural coupling has been investigated by combining artificial neural networks and ROM for a transport aircraft [184]. The aeroelastic behavior of the ROM shows a good agreement with the full-order model, except for cases located between training samples with very different aerodynamic behaviors. Fluid structure interaction shows other successful applications of interpolation in the process of ROM, where sparse grid interpolation and radial basis functions are used [204].

1.6 Intrusive versus non-intrusive

Projection-based ROMs and methods of residual minimization are called intrusive. The knowledge of the governing equations (or of the residual expressions) is needed in order to build the ROM. In most cases, this requirement means that the source code representing the physical system must be modified. Therefore, this approach cannot be pursued for commercial softwares for which the source code is not available and becomes very difficult for complex and legacy codes. These severe drawbacks motivate the use of non-intrusive methods. The latter only require data computed by the full-order model. Interpolated ROMs are very popular non-intrusive methods, especially for steady and parametric problems. Figure 1.4 presents a schematic view of the surrogate model classification, and lists intrusive and non-intrusive techniques.

There are no clear comparisons that evaluate the performance between projection-based and interpolated ROM, especially with Kriging models [15]. Peherstorfer [161] drew conclusions based on steady thermal design problem by comparing intrusive and non-intrusive approaches. Although they do not retain the structures of the physical model, interpolated ROMs show good results in providing predictions of the output not too far from the initial training points [49], while projection-based ROM are more competitive to predict "extrapolated" behaviors, where the predictions are distant from the full-order simulations used to build the ROM. In addition, the non-intrusive methods have the advantage of being very quick to implement. If only one quantity of interest has to be predicted, the interpolated ROM does not need to consider all conservative variables, unlike a classical projection-based ROM.

A comparative study has been performed between interpolation and projection methods on heat

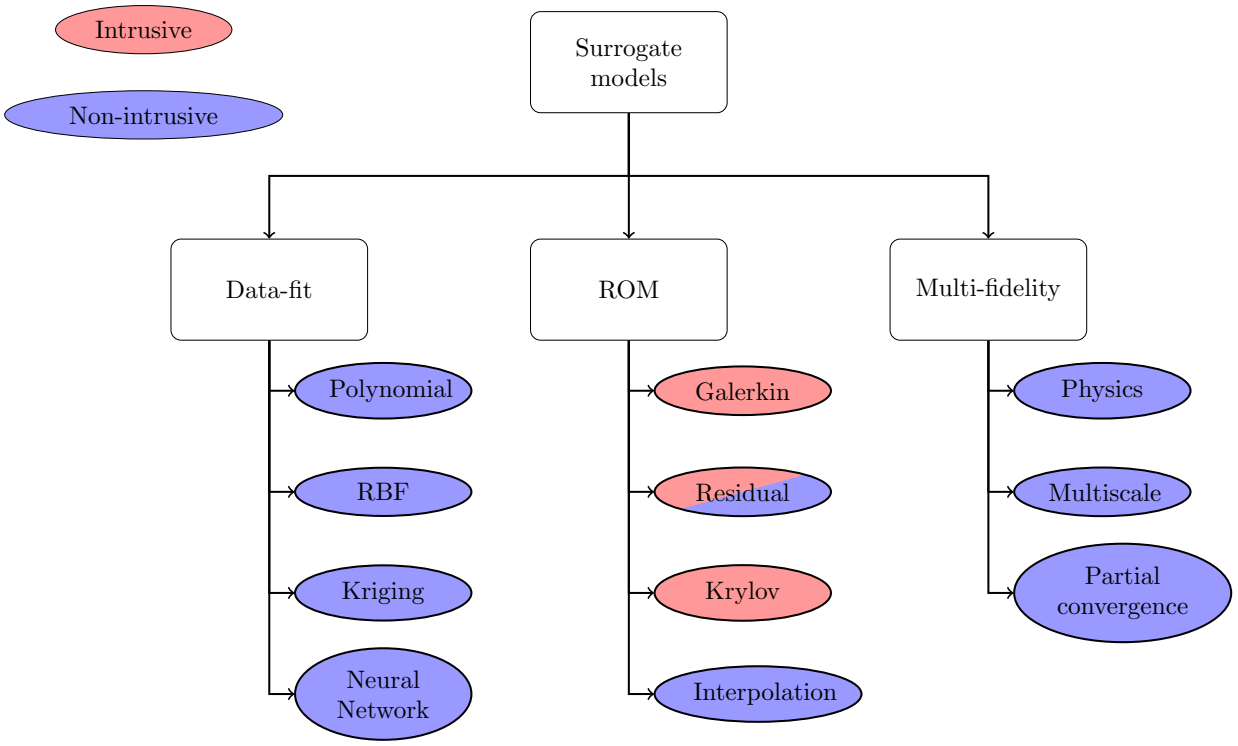


Figure 1.4: Classification of the different types of surrogate models.

transfer problems [201]. The authors highlight that the POD projection method outperforms the POD interpolation method for complex problems. On the other hand, a coupling between Artificial Neural Networks and POD has been investigated on analytical cases and steady incompressible flows: a similar accuracy has been achieved by neural networks and Galerkin strategies [94]. Finally, one can note that original methods combining projection-based and non-intrusive approaches have been developed. The operators of the ordinary differential equations, corresponding to the projection of the governing equations, are approximated from the full-order simulations by solving a least squares problem [162].

In the context of this thesis, the source code of the high fidelity simulator is not available. For this reason, only non-intrusive approaches are considered. In particular, interpolated POD models are investigated in order to build ROMs of steady nonlinear problems. The latter represent a great challenge for non-intrusive ROMs.

1.7 Modeling systems with various regimes

The prediction of solutions with bifurcations or a mixture of physical regimes can be the source of high discrepancies for non-intrusive ROMs, since the solution can change considerably as input parameter values change. This typical behavior is proved to be complex for surrogate models.

In the context of CFD, the varying inflow conditions usually lead to changing flow regimes, either subsonic or transonic. These changes have a wide impact on the evolution of the spatial distribution of the quantities of interest. The latter shows important nonlinear and discontinuous features related to the advection term of the governing equations, responsible of the nonlinear behavior, in particular the hyperbolic part of the equations.

Therefore, the generation of a very large number of high-fidelity simulations is required in order to capture all the variations of the flow. This ensures that the linear combination of the reduced basis functions is able to reproduce any shapes and that the regression model is well trained. Otherwise, the ROM fails to produce accurate responses with a restricted number of dominant modes for predictions in highly nonlinear regions and not directly in the neighborhood of the snapshots [141, 167, 51]. Two explanations can be given to this problem. First, discontinuous features dominate the POD modes and thus small errors in the multivariate regression step amplify POD modes associated with a physical regime that does not exist for the considered prediction. Indeed, low-energy perturbations can be masked although they could be representative of a part of the system's behavior [83]. Problems with bifurcations can have typical characteristics, such as aerodynamic flows with varying inflow conditions leading to either a subsonic or transonic regime. The second explanation involves a set of basis functions that is not representative enough.

The analysis performed by Li and Zhang [128] confirms these different insights. They highlight the poor capability of the interpolation ROM to properly predict the flow solution in extrapolation regions when high discontinuities are present. The evident mismatch between POD, a linear method, and the reconstruction of nonlinear responses is also emphasized. In addition to those comments, the mathematical properties of the POD and those of the equations can be compared. As underlined in Noack [155], the POD shows elliptic properties while aerodynamics is characterized by a hyperbolic part in the equations. This mismatch is the root cause of most of its weaknesses.

In that respect, an additional computationally intensive endeavor is required to properly train a model built on a training set mixing subsonic and transonic flows. It goes against the principle of the parametric ROM that aims at building surrogate models with a favorable tradeoff between computational cost and accuracy. For this reason, specific surrogate models dealing with various physical regimes must be developed, in particular for aerodynamics.

1.8 Learning physics from data with Machine Learning

The term "machine learning" refers to a class of algorithms capable of extracting relevant information from various data sources without being explicitly programmed. It is closely related to artificial intelligence and deep learning. These algorithms are built from different fields such as statistics, computer science, and information theory.

Over the last decade, the dramatic increase in large datasets and the capability to process them explains the growing popularity of machine learning and data science in general. Many technological applications that use machine learning can be found in everyday life, such as detection of spam emails, real-time translation, or facial recognition. Most machine learning methods are usually divided into three broad categories:

- **Supervised learning**: the goal is to find the relationship between some input features and labeled output. A function is inferred from the training data forming pairs of inputs and outputs. If the output data are continuous, the supervised learning method solves a regression task. Surrogate models belong to regression methods. On the contrary, if the outputs are discrete, the supervised learning method solves a classification problem;
- **Unsupervised learning**: the algorithm extracts hidden structures and patterns from unlabeled data. The training set is no more associated to outputs or targets. Unsupervised learning can describe several tasks, for instance dimensionality reduction, clustering, or generative modeling;

- **Reinforcement learning**: an agent is trained to interact with an environment by maximizing a numerical reward signal. Thus, this agent must discover the best action yielding the best reward using "trial and error". Direct instructions are not given to the agent. Moreover, a specific action may affect the reward over an extended time period, thus complicating the task of the agent. Typical examples of reinforcement learning examples include robot control, telecommunication or games, such as the board game Go.

Machine learning and artificial intelligence methods are starting to be adapted by the aerospace industry. For instance, the development of powerful tools for extracting knowledge from the vast amounts of data generated by large-scale aerospace simulations is listed in NASA's roadmap of their 2030 CFD vision [181].

In particular, the turbulence modeling community is starting to widely use machine learning tools in order to develop specific forms of turbulence models [188, 55], thanks to the growing databases of LES and DNS flow solutions. Therefore, this Ph.D. thesis aims at coupling surrogate models, in other words supervised learning, with other machine learning techniques to improve the relevance and performance of the surrogate models.

1.9 MDA-MDO project

The Multi-Disciplinary Analysis - Multi-Disciplinary Design Optimization (MDA-MDO) project was initiated at the beginning of 2015 at the Institute of Research Technology (IRT) Saint Exupéry. The IRT organization gathers academic partners, such as CERFACS and ISAE, and industrial partners, such as Airbus, Airbus Group Innovations, Altran Technologies, and Sogeti High Tec. The main objective is to develop MDO methodologies by taking into account industrial needs and constraints by building the multidiscplinary platform SPIRO and the GEMS library for supporting MDO capabilities [80]. MDA-MDO techniques are still at low Technology Readiness Level (TRL) and require maturation before injection into industrial processes. The MDO of an advanced aircraft configuration represents one of the challenges of NASA's CFD vision 2030 [181].

As mentioned previously, placing the engine closer to the other parts of the airplane makes pylon design critical. The MDA-MDO project considers this application as the main test case. The position of the nacelle is optimized by modifying the size and the shape of the pylon. The optimization aims at reducing a macro objective: the cash operating cost.

The novel techniques developed in this thesis are not integrated in the tools of the MDA-MDO project, mainly due to their low level of maturity. Nevertheless, they offer a contribution through a first insight into future research directions to leading-edge surrogates models. As pointed out in Viana et al. [195], further efforts are needed to broaden the use of surrogate models in MDO and non-MDO communities. In particular, new developments focused on practical applications can attract the attention of the industry, which is one of the main objectives of this thesis.

1.10 Contributions

The present research investigates the development of efficient non-intrusive surrogate models that generate global approximations of complex aircraft simulations. A particular importance is given to complex aerodynamic and aerothermal test cases. Several original contributions can be identified in this Ph.D. thesis:

- **Review of non-intrusive ROMs for fluid flow problems**: a limited survey of non-intrusive ROMs is carried out with a focus on methods capable of managing different flow regimes. Their advantages and drawbacks are illustrated;
- **Development of local surrogate models adapted to problems with multiple physical regimes**: machine learning algorithms, such as clustering and classification, have been used to build local surrogate models adapted to different regimes. This development can be seen as a generalization of mixture of experts and local reduced-order bases to non-intrusive ROMs. Moreover, this work has been extended to adaptive sampling in order to dedicate the computational budget to the more challenging areas of the parameter space;
- **Aerodynamic load predictions over a full aircraft**: the strategy of local surrogate models is applied to complex three-dimensional configurations. These studies give a first insight into the behavior of the developed method and its interpretation. Furthermore, a particular attention is paid to the statistical validation of the surrogate models;
- **Surrogate-based aerothermal model in a flight mission context**: a surrogate model replaces an expensive CFD code to generate the boundary conditions of a thermal solver. Thus, the thermal behavior of the aircraft pylon is simulated over the full flight mission by taking into account the variation of the external and engine conditions. Industrial tools have been employed in order to be representative of the industrial constraints.

1.11 Thesis outline

The thesis is divided into three parts. Part I introduces the subject, Part II presents different tools building non-intrusive ROMs and the Local Decomposition Method developed in this thesis, finally Part III shows the applications of these methods on various aerodynamic and aerothermal problems. Figure 1.5 sketches the relation between the different chapters, colored by part.

The introduction part encompasses Chapter 1 and Chapter 2. The latter is devoted to a review of state-of-the-art non-intrusive ROMs focused on different variants improving the accuracy of the ROMs in the context of fluid flows that encounter or not different physical regimes.

The second part is focused on the methodology. Chapter 3 presents different common tools to build ROMs, such as POD, Design of Experiments, various regression methods, and strategies of validation. Chapter 4 details the steps of the Local Decomposition Method, in particular: the machine learning tools, the use of sensors, the resampling strategy, and the assessment of the method on analytical and academical test cases.

Finally, the third part is dedicated to the application of the methods presented in the second part. The novel technique developed in this thesis is applied to aerodynamic test cases in Chapter 5, while classical ROMs are used to build an aerothermal application by linking a surrogate model with a thermal solver in Chapter 6.

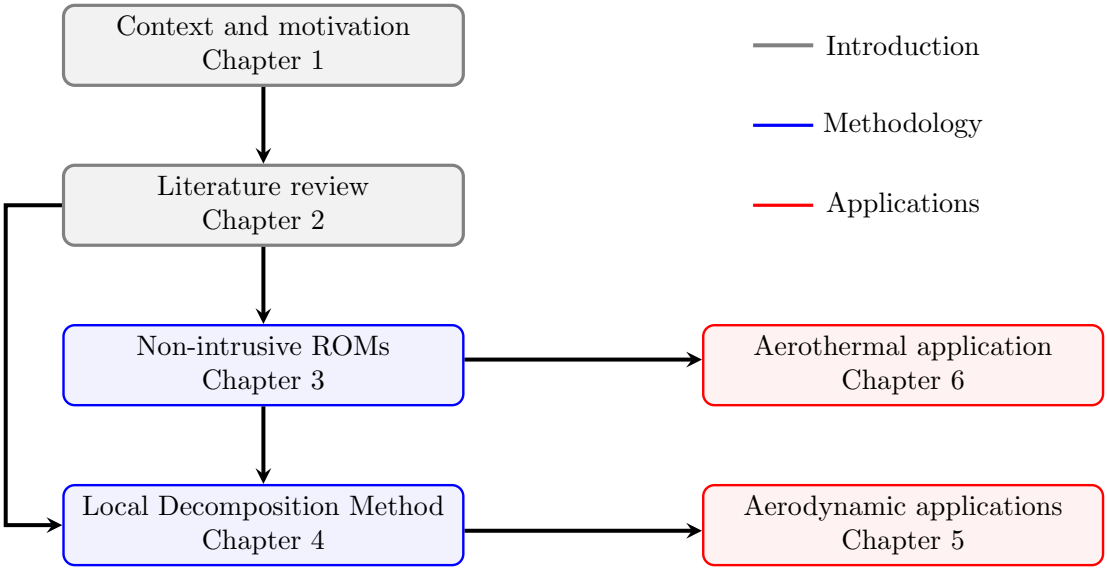


Figure 1.5: Reading guide.

CHAPTER 2

Literature review on non-intrusive ROMs

Résumé

Les capacités des modèles réduits à prédire des écoulements aérodynamiques de manière non-intrusive ont été démontrées au début des années 2000. La principale méthode consiste à déterminer les composantes de la décomposition aux valeurs propres, appelées coordonnées réduites, par des méthodes de régression. Cette approche a été utilisée pour de nombreuses applications et des améliorations ont été proposées comme : le réechantillonnage adaptatif de l'espace des paramètres, la réduction de l'espace d'entrée par analyse des sensibilités, la prise en compte de plusieurs niveaux de fidélité ou encore de sources d'informations différentes telles que le gradient.

Néanmoins, malgré ces améliorations, il a été observé que les modèles réduits peuvent rencontrer des difficultés dans des cas particuliers, notamment lorsqu'un système traverse des régimes physiques très différents. Plusieurs méthodes ont émergé dans la littérature scientifique pour résoudre ce problème, principalement pour les modèles de régressions statistiques mais aussi pour les modèles réduits.

Tout d'abord, l'approche zonale concerne uniquement les modèles réduits et découpe le domaine spatial en plusieurs sous espaces. Le modèle réduit est appliqué lorsque l'écoulement est moins complexe, ce qui permet d'utiliser le modèle haute-fidélité uniquement sur une région restreinte de l'espace, réduisant ainsi les coûts de calcul.

Une autre méthode, la décomposition de l'espace des paramètres, s'est largement développée dans le domaine de la régression. Elle tire profit de l'hétérogénéité des solutions afin de construire des modèles de substitution locaux au sens de l'espace des paramètres. Ce dernier est séparé en plusieurs blocs, généralement par de l'apprentissage automatique. L'objectif visé est de mieux capter la complexité des phénomènes physiques avec chacun des modèles locaux plus spécialisé et mieux entrainé pour prédire un type de solution en particulier.

Le dernière approche palie des faiblesses de la POD en employant des méthodes de réduction de dimension non-linéaires. Ces dernières extraient des variétés de faible dimension afin de construire des modèles réduits plus pertinents.

Enfin, il peut être observé que l'application de modèles réduits à des cas industriels reste rare dans le domaine de la mécanique des fluides. Néanmoins, plusieurs exemples peuvent être trouvés dans la littérature comme : la prédiction des charges aérodynamiques et du givrage autour de voilures, la quantification d'incertitudes dans le domaine des turbomachines via des calculs aux grandes échelles, la dispersion de polluants via des méthodes de Lattice Boltzmann ou encore le calcul aérothermique de véhicules hypersoniques.

Ainsi, les travaux de cette thèse se positionnent selon plusieurs axes par rapport à la littérature scientifique et l'état de l'art. Tout d'abord, on s'intéressera aux méthodes de décomposition de l'espace des paramètres afin de développer des solutions adaptées à des problèmes aérodynamiques. L'accent sera largement porté sur l'application avec la validation de la méthode sur des cas complexes. De plus, des modèles réduits plus classiques seront implémentés dans une chaine de calcul aérothermique afin de valider leur utilisation dans un contexte industriel.

Inherent computational costs of CFD simulations explain the significant use of ROMs to solve problems involving fluid flows. This section is focused on the development of non-intrusive ROMs and gives a wide overview of different strategies employed to deal with fluid flows, in particular aerodynamics which represents one of the main concern of this work. Aerodynamic analysis appears to be well suited to non-intrusive ROMs. In fact, high-fidelity models involve a mix of numerical schemes, algorithms and turbulence modeling: modifying such models is not an easy task and requires a profound knowledge of the source code. Therefore, preference is given to non-intrusive approaches, as illustrated by a rich literature on the subject.

The classical methods to build non-intrusive ROMs are first presented. Specific methods have been developed in response to difficulties of approximating flow simulations with a high computational cost, in particular resampling and multi-fidelity approaches. Details of the latter strategies are provided below. In addition, specific flows with very different flow regimes represent a great challenge for non-intrusive ROMs. Strategies based on local surrogate models, which have been developed in this work, are presented in detail. A wide range of industrial applications also illustrates methods of non-intrusive ROMs to solve complex problems. Finally, the methods developed in this thesis are put in the context of this literature review.

It is important to specify that usually non-intrusive ROMs, intrusive ROMs, and data-fit models are closely related. Therefore, even if this chapter is focused on non-intrusive ROMs, brief digressions may be made on regression models or intrusive ROMs in order to introduce interesting methods useful for non-intrusive ROMs.

The majority of the papers presented in this section is summarized in Table 2.1 and Table 2.2.

2.1 Classical non-intrusive ROMs

As presented in Chapter 1, POD is the leading dimension reduction tool to build ROMs. Therefore, the first solution envisaged to build non-intrusive ROMs interpolates directly reduced coordinates in the parameter space. One of the first application of this method can be found in Bui-Thanh et al. [24]. The authors have presented a POD method with interpolation in order to analyse the parametric variation of an inviscid steady flow about a NACA0012 airfoil for transonic conditions.

Predicting the pressure fields has shown a high degree of accuracy, although the number of training sample (231) is comparatively high for 2 parameters and a limited range of variation. This first result has paved the way for the development of non intrusive ROM methods and their application to various cases. Many airfoils, in particular RAE2822 and NACA0012, have been widely studied with POD and interpolation methods [22, 141, 167, 51], but in most of the cases only for a limited flow regime (subsonic or transonic) or a small range of variation.

Classical non-intrusive ROMs are also applied on unsteady flows. Walton et al. [199] have successfully combined two levels of POD and a Radial Basis Function (RBF) interpolation in order to build unsteady and non-intrusive parametric ROMs. Only a small loss of accuracy, compared to the full-order model (FOM), bas been measured for the oscillating ONERA M6 wing. However, the results have only been computed for a single frequency of the oscillation. It cannot be concluded that such a method can be generalized for various frequencies. Xiao et al. [204] have developed a non-intrusive ROM for dynamical fluid-structure interactions. This strategy has been based on a recursive RBF interpolation: at each time step, the interpolation is function of the physical quantities at the previous time step. Although applications have only been performed on two-dimensional test cases, it has been one of the first application of non-intrusive ROMs on fluid-structure interaction problem.

A recent study has applied an unsteady and parametric analysis on a one-dimensional model of a rocket combustor with four parameters [200]. The proposed method approximates the reduced coefficients by an artificial neural network on both time and parameters. The numerical results have shown that the ROM is very consistent with the full order model. Nevertheless, a very large number of 41.000 snapshots has been needed to train the model. It demonstrates that further research on more complex geometries and governing equations remains open, especially for transient problems.

A clear limitation of classical non-intrusive ROMs is highlighted in Zimmermann and Görtz [214]. The method has been compared to a more intrusive approach minimizing the residual. The POD-based interpolation method using Kriging has shown weaknesses to predict transonic flows in region of extrapolation, outside of the range of the training set. Qamar and Sanghi [166] have suggested to adress these weaknesses by enriching the training set with predictions from the ROM. The idea is to extrapolate the quantity of interest closely to the bound of the training set, as predictions are still relatively accurate. Thus, the predicted points can be added to the training set. POD modes and interpolation models are computed on this new data. This process is repeated until the parameter value of the initial extrapolation is reached. This strategy has been applied for supersonic flows and reduces errors by more than 50%. However, this method significantly increases the computational cost due to numerous computations of the POD basis and an intensive training of the interpolation model. Moreover, if the step size is badly chosen, the training set can be corrupted.

2.2 General improvement strategies

The parametric reconstruction of flows by ROM represents a major challenge if large operating conditions are considered and simulations take several thousand CPU-hours, be it through intrusive or non-intrusive methods. Therefore, several solutions have been proposed in order to optimally spend the computational budget in this challenging context and to improve the predictions. In this section, particular emphasis is given to resampling strategies and multi-fidelity approaches. It is important to note that the improvement strategies for non-intrusive ROMs are very broadly inspired by data-fit methods and intrusive methods. A parallel is arisen below for each strategy. Methods using general improvement strategies are marked in the "General Improvement" column

in Table 2.1.

2.2.1 Resampling strategies

Building a surrogate model, scalar or vectorial, is usually divided into several stages. Two of them play a critical role in managing the simulations. First, the DOE explores the parameter space by generating an initial sampling in order to learn the main features of the function to approximate. Then, the initial sampling can be progressively enriched during the resampling step in order to improve the model. In the context of global surrogate modeling, the resampling aims at finding an optimal selection of points by using a trade-off between exploration and exploitation of the model [46, 131]. On the one hand, the exploration of the initial DOE is continued in order to find regions of interest that have not yet been identified before. On the other hand the exploitation is focused on improving the knowledge on the interesting regions that have already been detected. Figure 2.1 illustrates the concept of resampling strategies.

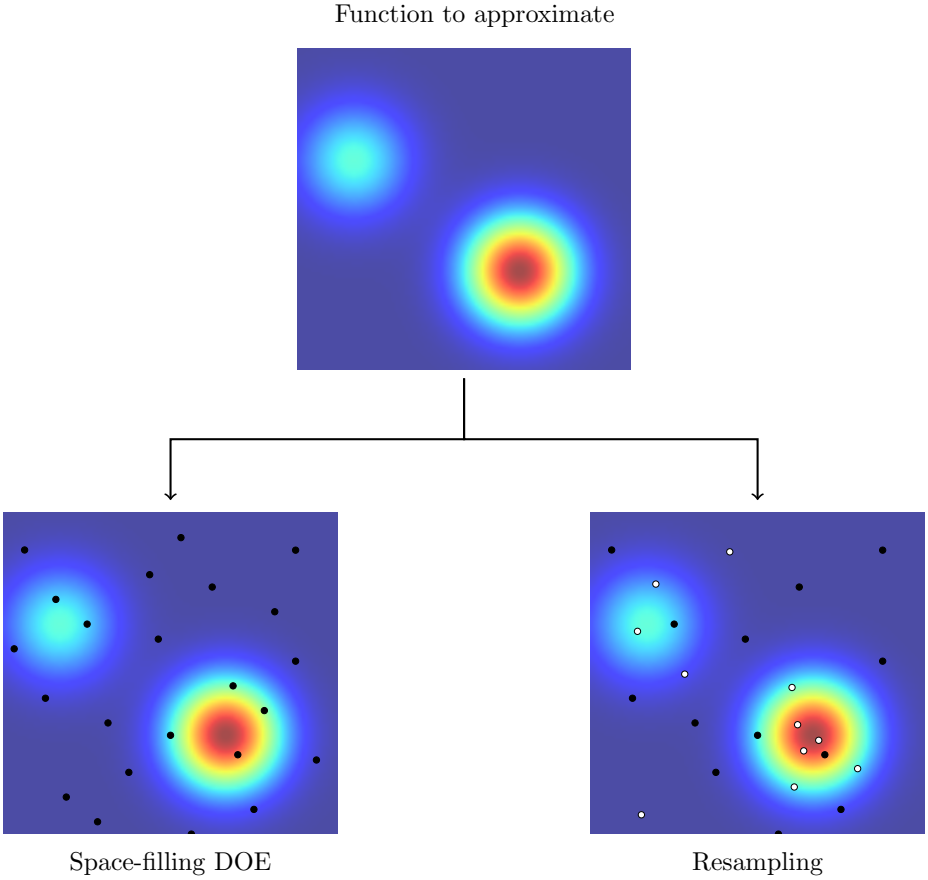


Figure 2.1: Illustration of space-filling DOE versus resampling. The samples generated by a classical space-filling strategy are in black while the additional samples query by the resampling step are in white.

The resampling algorithm aims to ensure the best choice between these two contradictory options. If the resampling strategy exclusively selects exploration samples, computational budget may be wasted in areas of low interest which are already well predicted and other areas may be neglected where the surrogate model shows poor predictive capabilities and evident weaknesses.

Conversely, a resampling strategy solely based on exploitation may left large regions of interest unsampled. The concept of exploration-exploitation is also encountered in optimization [35] where the exploitation is focused on finding the optimum values.

The examples described in this section follow generally the same idea than greedy approaches encountered in projection-based ROMs [159, 15]. Such methods seek, at every iteration, the location in the parameter space where the norm of a given error metric between ROMs and the full order models is maximal. It leads to an adaptive sampling decreasing directly the error of the model. Unfortunately, the derivation of an exact error metric appears intractable as it needs to compute the full model at many different parameter points, annihilating the interest of a ROM. Therefore, a posteriori error estimations are used instead of the exact metrics. Under certain assumptions, they can be derived from the partial differential equations representing the full order model. In the general case, a posteriori error bounds [84] or residual estimations [26] are employed.

The greedy approach takes advantage of error metrics given by projecting the governing equations. However by their very nature, the non-intrusive methods do not have access to the equations and must find another way to approximate the error of the model. For this reason, alternative methods have been proposed to substitute the estimation of the error metric.

The k-fold cross validations and the particular case of the leave-one-out cross validation (LOO) are very popular methods to provide an estimation of the modeling error at each point of the training set. Details are given on these methods in Chapter 3. Braconnier et al. [22] introduce a leave-one-out error estimation coupled with a quad-tree refinement approach on a RAE2822 airfoil. The objective is to refine the flight domain in the regions of large errors. The leave-one-out method gives an estimation of modeling errors at each point of the training set. Details are given in the following section. At first sight, the resampling strategy reaches a better accuracy than classical Latin Hypercube sampling. However the application is limited to two dimensions and the proposed approach is very exploitation-oriented, assuming the parameter space is already well covered. In addition, the quad-tree algorithm does not scale as the number of resampling points follows an exponential growth with dimensions, which is not applicable to industrial complex problems.

Other methods based on leave-one-out cross-validation have been developed. They can show great robustness to high-dimension problems. Kato and Funazaki [111] have weighted the estimated error of the leave-one-out with the smallest Euclidean distance to the sampling. Therefore, the exploitation information coming from the model is coupled with a geometric exploration criterion. This strategy has been applied for the optimization of a supersonic impulse turbine and shows that a very satisfactory Pareto front can be build with non-intrusive ROMs. A similar approach is found in Zhan et al. [210, 211] for aero-icing analysis. Instead of using the Euclidean distance, the authors propose a centroidal Voronoi tessellation where the density function of the mass centroid is substituted by the LOO error. Once again, the exploitation of the model is emphasized by assuming the exploration of the initial sample catches all significant features of the signal. The error-driven sampling clearly focuses on a particular type of ice formation and improves the accuracy in this specific challenging region.

Additional resampling strategies have been directly derived by using information given by the data-fit methods. In particular, Kriging shows a rich literature where the variance estimation given by the model plays a key role by substituting the error metric. The objective is to reduce the error predicted by Kriging model. Two popular criteria are identified in the literature: the Maximum Mean Squared Error (MMSE) and the Integrated Mean Squared Error (IMSE) [175, 28]. Mackman et al. [140] have compared the MMSE with traditional sampling strategies in the context of aerodynamic test cases, in particular the RAE2822 airfoil and the DLR-F12 aircraft. They have concluded that such resampling strategies perform better than classical space-filling

sampling, even optimized Latin Hypercube Sampling. To the best of the author's knowledge, such strategies have been seldom used in non-intrusive ROMs.

Roy et al. [174] is a rare example which couples MMSE with the leave-one-out cross validation. The training sample with the highest leave-one-out cross-validation error is identified as critical. A neighborhood is defined around this sample and the point with the maximum variance in this neighborhood is chosen as the resampling point. The exploration aspect of the resampling is neglected with this kind of strategy, which is however validated on analytical cases with various dimensions and shows very satisfactory results. The lack of exploration is counterbalanced by a initial sampling which is large enough.

In conclusion, resampling strategies have demonstrated significant capabilities to improve the accuracy of surrogate models for various application cases. They provide an optimum use of the available computational budget by increasing the accuracy in critical regions. In addition, they require only a slight modification of the global workflow of ROMs. The equations of the dimension reduction and of regression methods do not change.

2.2.2 Multi-fidelity approaches

Limiting the computational budget motivates the development of multi-fidelity strategies. In addition to expensive high fidelity data, some cheap and low fidelity data may be available. The notion of fidelity can be seen as the level of approximation used to be representative of physics. Low fidelity data can take various forms: partially converged simulations [72], coarser meshes [141], governing equations of lower fidelity [81], etc. Even if the accuracy of such data is decreased, they are generally computationally cheap and can be used to enhance surrogate models [112, 73] and to explore the parameter domain. Figure 2.2 illustrates the concept of multi-fidelity approaches. The accuracy of the surrogate model is clearly improved by merging two sources of data. However, the different sources of data requires to find specific relations between them. Therefore, a mathematical formalism capable of generating surrogate models merging various levels of fidelity have been developed with the objective of improving accuracy of the surrogate models.

Multi-fidelity methods are very popular in fluid mechanics and solid mechanics as illustrated in Fernández-Godino et al. [70]. More than 71% of a basis of 157 papers reviewed concern these fields. Two explanations may be suggested: fluid and solid mechanics are particularly well suited, by their very nature, to be approximated by surrogate models but they also have a wide range of modeling strategies with various fidelity levels. One can cite for example for fluid mechanics, from highest to lowest fidelity: Direct Numerical Simulation, Large Eddy Simulation, Detached Eddy Simulation, RANS, Euler equations, potential flow, and lifting line theory.

Various types of multi-fidelity strategies have been proposed in the literature of data-fit models. Three broad categories can be identified [206]:

- correction methods, also called bridge functions or scaling functions, are widely used to build multi-fidelity models. The discrepancy of low fidelity data are reduced by correction terms, mainly additive or multiplicative. These terms are derived by matching high fidelity and low fidelity data. Consequently, the training sets of low and high fidelity data are forced to coincide. In addition, one can cite the evofusion, developed by Forrester et al. [72], that is a very popular correction method building a Kriging model on the difference between the two fidelity levels;
- the co-Kriging is a multivariate extension of Kriging models building a relation between

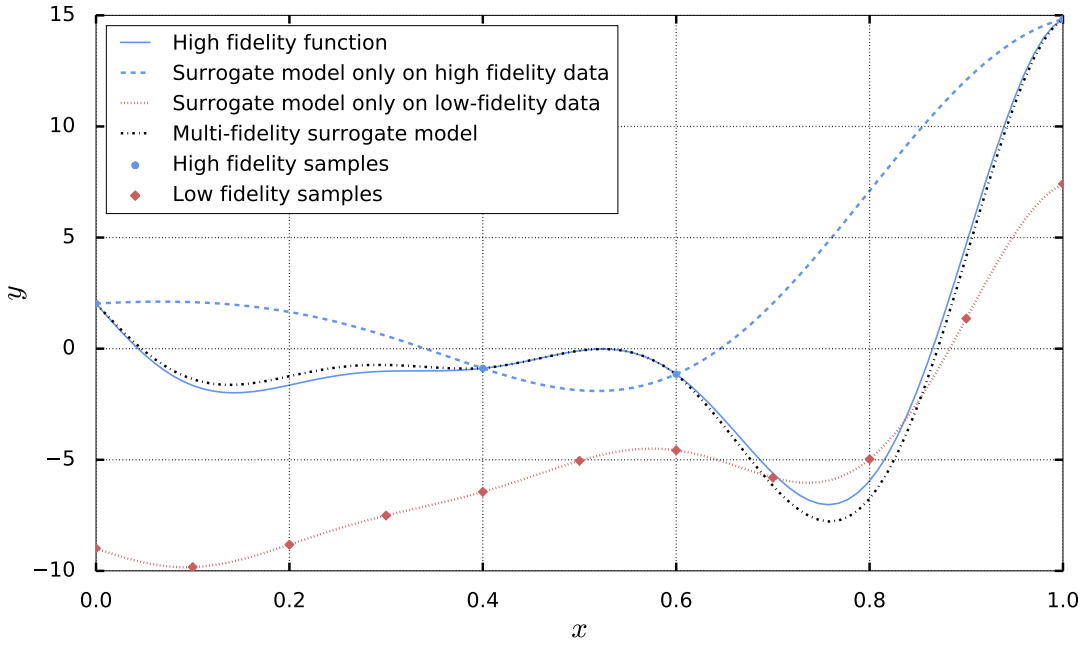


Figure 2.2: Multi-fidelity illustration. Inspired by Forrester et al. [73].

high fidelity and low fidelity data by using a cross correlation matrix, which quantifies the relationship between data from various levels of fidelity. The cross correlation matrix can be calculated in different ways, based on specific assumptions. For example, Zimmermann and Han [215] present a simple method by assuming the covariance for both low fidelity and high fidelity models is identical. This assumption appears very strong as it means the two models behave in the same way, independently of the level of fidelity. A second approach seeks to estimate a multiplicative factor between low and high fidelity [88]. Finally, the most prevalent form of multi-fidelity co-Kriging has been proposed by Kennedy and O'Hagan [112]. This model defines an autoregressive relation between different levels of fidelity based on a Markov property. Co-Kriging models show a very good predictive property, however their main drawback is the ill-conditioning of the covariance matrix due to large data sets generated by the observations at all fidelity levels. An iterative approach has been developed to even further reduce the complexity by building a sequence of Kriging models while preserving the predictive efficiency of the co-Kriging [126];

- the hierarchical Kriging modifies the classical Kriging by substituting regression functions by metamodels built on low fidelity data [89]. It demonstrates a good predictivty and important robustness. In addition, this approach provides interesting numerical properties as the size of the covariance matrix does not change compared to ordinary Kriging. The numerical implementation remains simple, since the equations are very similar.

Correction methods have been applied to aerodynamic problems, such as the prediction of aerodynamic coefficients of an RAE 2822 airfoil [90] or of a crew transfer vehicle with a wide flight envelope including subsonic, transonic and supersonic flows [185]. A correction method combining RANS and DNS has been proposed quite recently by Geraci et al. [81], where RANS are considered as low fidelity this time. The authors highlight that connecting RANS and DNS is not evident, as shows a low correlation factor. Hence, the multi-fidelity approach faces difficulties to build accurate response surfaces. Evofusion has also been applied on a global aerodynamic optimization

problem [72].

Communities of data-fit models and ROMs have developed and used very similar strategies. However, the strategies of multi-fidelity ROMs can encounter supplementary problems, in addition to managing the different sources of data. For example, the various sizes of mesh change the nature of the output and challenge the computation of the reduced basis. The Gappy POD [25] is a classical solution filling incomplete data or coarse meshes using a set of POD basis built from a fine mesh and solving a minimization problem.

Regarding aerodynamic ROMs, multi-fidelity strategies have been successfully applied on various cases from numerical toy functions to complex three dimensional cases. Toal [186] has applied a multi-fidelity approach on a NACA0012 airfoil at various flow conditions. The multi-fidelity strategy is based on the gappy POD method mixing results from a CFD code, viewed as high fidelity data, and a panel code, as low fidelity data. Interestingly, the drag coefficient is computed by integrating the full pressure signal from the ROM. The output is compared with predictions of data-fit methods, in particular classical Kriging and a multi-fidelity co-Kriging strategies, trained directly on integrated values given by simulations. The author has concluded that the Gappy POD increases significantly the accuracy of the predictions. More unexpectedly, a Kriging model built from reconstruction points of the gappy POD outperforms co-Kriging models built from the same multi-fidelity sampling and outperforms even more the Kriging model only built from the simulations on the fine mesh. This result suggests that a POD basis spanning a subspace, which sufficiently approximates the solution space, can accurately generate high fidelity data from only low fidelity data. In addition, using ROMs in the context of optimization problems is fully justified. ROM capabilities ensure to predict the full signal, containing important physical features, while data-fit methods are directly trained on integrated values which filter an important part of the physics. The trend is confirmed on a thermo-mechanical case of a gas turbine engine rotor, where the low fidelity model is simply a time truncation of the high fidelity model corresponding to a reduction of the computational cost of 36%. The final result shows a prediction error of less than 1%, compared to the full computation of the high fidelity model, and a computation cost reduced by 30%.

Malouin et al. [141] have applied the evofusion method by considering two different grids of the flow over the RAE2822 airfoil, a coarse one and a fine one. Although shocks on the suction side are better captured, this method generates very noisy flow solutions unlike the gappy POD, since once low fidelity data have been corrected by the error model, a nearest neighbor spatial interpolation is performed to predict the data on the fine grid.

Another application of multi-fidelity ROM on the RAE2822 has been investigated by Benamara et al. [14]. The authors have applied the gappy POD at two different levels of fidelity. The originality here comes from the multi-fidelity resampling proposed to improve the model with a strategy ensuring the trade-off between exploration and exploitation. Indeed, only the low fidelity model has been considered to explore the parameter space. Then, an error criterion has been derived to measure the error made by the gappy POD on the exploration samples at the low fidelity data. This criterion has been assumed to be a good approximation of the error at the high fidelity level, allowing to exploit the model by enriching the sampling where the error criterion is maximum. The validity of this assumption impacts inherently the quality of the resampling. The multi-fidelity method coupled with an adaptive resampling offers two interesting properties: a good stability of the prediction error with respect to the initial sampling and an overall improvement of predictivity capabilities compared to classical Kriging or ROMs coupled with LOO resampling.

2.2.3 Exploiting different types of data

In addition to data with various levels of fidelity, surrogate models can be enhanced by other type of data. The gradient information is the more common example. In particular, an adjoint solver can provide the gradient, as its computational cost does not generally scale with the number of design variables unlike other methods like finite differences or complex-step [170]. Chung and Alonso [37] have proposed to use co-kiriging by considering the gradient as an auxiliary variable in a supersonic business jet design optimization. This method is called Gradient-Enhanced Kriging and incorporates the gradient information in order to improve the accuracy of the predictions. The Gradient-Enhanced Kriging has been applied to various fields, in particular where the sensitivities are available and useful, such as the gradient-based optimizations. This explains the great interest of the method in aerodynamic design optimization [123] and aircraft mission analysis [129], where the superiority of Gradient-Enhanced Kriging in term of accuracy is emphasized.

Using indirect co-Kriging [132] has been introduced as an alternative to classical co-Kriging which do not require to modify the source code of the Kriging. The training set of the Kriging is augmented by adding training points from a first order Taylor approximation in each direction. Two drawbacks have been highlighted: a small step size (or a step size in the same order of magnitude as the training sample density) generate an ill-conditioned covariance matrix and a large number of input parameters can lead to important computational costs. Therefore, the step size must be defined carefully. A specific strategy building Gradient-Enhanced Kriging model for high dimensional problems have been developed and validated on analytical test cases [19].

Zimmerman [213] presents an original strategy to construct ROMs based on a gradient-enhanced interpolation of the POD coefficients. POD subspaces has been augmented by the gradient information. The predictions have shown a smaller error than the classical interpolation-based POD method. Nevertheless, the strategy has only been applied to an analytical model function. In the context of projection-based ROMs, a framework coupling POD with gradient information has been sucessfully demonstrated for an aeroelastic research wing in [32].

It is important to underline that the gradient information involves significant costs. For example, the computation of an adjoint solution of a CFD solver can be equivalent to the computational cost of a flow solution [170] or of several direct simulations. Therefore, this additional cost must be carefully measured in order to weight the computation of the gradients against a finer training sampling.

2.2.4 Input dimension reduction

Surrogate models and non-intrusive ROMs also encounter major issues while approximating systems with an important number of input design variables. The exploration of such large parameter space remains intractable due to the number of possible combinations growing exponentially, related to the notion of curse of dimensionality introduced by Bellman [13]. Moreover, numerical issues can arise by building surrogate models in high dimensions. For example, Kriging models estimate hyperparameters by maximazing the likelihood. This process has a significant computational cost as the covariance matrix must be computed and inverted for each combination of hyperparameters. Thus, the optimization problem cannot be solved for high-dimensional parameter spaces.

Therefore, several methods have been proposed to reduce the dimension of the input parameter space. The most popular strategy is the sensitivity analysis aiming at identifying and prioritizing the most influential input parameters. The non-influential parameters can be ignored by the model and are fixed to constant nominal values [100, 176]. The sensitivity analysis proposes different tools.

The screening seeks to identify the influential and non-influential inputs with a small number of model evaluations, usually about the number of inputs. This approach relies on several hypotheses on the model complexity and do not quantify the relative importance of each parameter. Usually, screening is first used to simplify the model before using more precise but more costly sensitivity analysis methods, such as variance-based quantitative methods. Sobol' indices rank the input parameter in term of influence by quantifying their contribution to the variance of the quantity of interest.

Bouhlel et al. [20] have developed a specific version of Kriging based on partial least squares in order to deal with high-dimensional parameter spaces, called Kriging Partial Least Square (KPLS). The method performs a change of variables by seeking the best multidimensional direction explaining the variability of the output. Thus, only a reduced number of new variables is chosen to build the Kriging, reducing the complexity of the hyperparameter optimization. This approach has been validated on analytical cases with up to 100 variables.

A strategy based on a relatively similar idea, called active subspace method, has been proposed by Constantine et al. [40]. The coordinates of the parameter space are rotated after analyzing the gradients, as the authors have observed that the main variations of the model do not occur necessary in input directions aligned with the coordinate system. The main steps of the methods are the followings: building a covariance matrix from the partial derivatives of the function, determining the main directions of the gradient based on a eigendecomposition of the matrix, and training the surrogate model on the principal gradient directions. The method has been advantageously applied on various test cases, in particular the uncertainty quantification of a complex scramjet with 7 parameters where each run takes about 9.500 CPU-hours [41].

However, as with gradient enhanced methods, active subspaces need the gradient information usually from an adjoint solver, increasing the overall computational cost. If an adjoint solver is not implemented, a finite difference approach becomes quickly intractable for high-dimensional parameter spaces. Alternatives have been proposed, such as using a linear approximation in order to compute the one important direction in the input space [41]. However, the validity of such an approximation must be validated carefully.

All methods presented in this section have been developed in the context of scalar quantities of interest. To the best of the author's knowledge, their application to ROMs is very rare in the literature. Margheri and Sagaut [143] have developed a workflow coupling POD, Kriging, ANOVA decomposition and Sobol' indices for the uncertainty quantification of urban flows. The quantification of uncertainty has also motivated the use of Sobol' indices in the study of the LS89 blade cascade proposed by Roy et al. [174] mixing POD, Kriging, and adaptive resampling based on the sensitivity.

2.3 A challenging modeling of flows with various regimes

As pointed out in the introduction, physical systems encountering different regimes can pose serious problems to surrogate models. For this reason, several methods have been developed in order to reduce the number of snapshots ensuring an acceptable accuracy for flows with various physical regimes. In particular, two local approaches are discussed in the following sections: the decomposition of the spatial domain and the decomposition of the parameter space. They are identified as "Local Method" in Table 2.1.

2.3.1 Zonal spatial decomposition

The main idea of zonal methods is to decompose the spatial domain into several blocks by coupling models with different levels of fidelity, for example FOMs and ROMs. Figure 2.3 shows an illustrative example of how a computational domain can be split. Usually, specific regions with shocks and high gradients are isolated. Different models are respectively applied on high gradient regions and on outer regions. In this example, a high-fidelity model covers the area Ω_{HF} around the airfoil while a low-fidely model represents the far field Ω_{LF}. An overlapping region Ω_O ensures the transition between the two models. Lucia et al. [137] demonstrates the ability of such methods on a two-dimensional flows with moving shock waves. Two different models are considered for the region of the flow field containing a moving shock: directly the FOM and a specific ROM for the shock region. The first approach works well, even if the computational cost is increased, while the second one encounters major issues due to the shock motion and its high intensity.

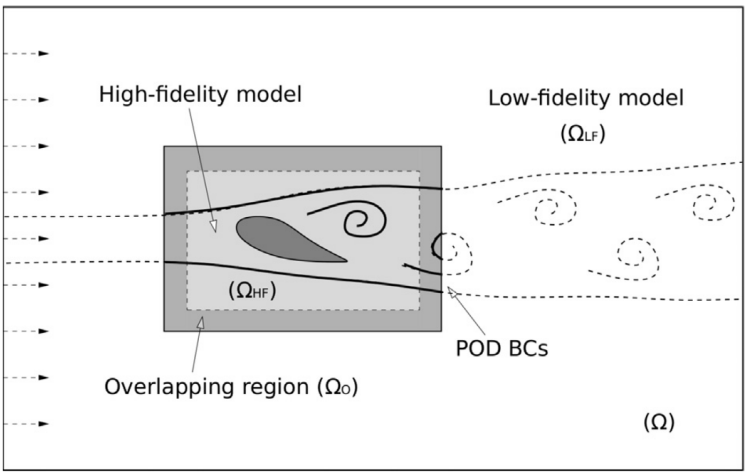


Figure 2.3: Zonal POD. Credits from Bergmann et al. [16]

The main issue here is the clear identification of each region: how to know for each mesh node if it belongs to high-fidelity or low-fidelity models? Different strategies have been proposed, mainly based on the detection of thresholds:

- Constantine and Iaccarino [39] identify regions with high parameter gradients. For each point of the domain, the gradient is computed and if it exceeds a given threshold, the point, considered of poor quality, is tagged as requiring a specific correction. The shock are well reconstructed with this strategy. However the method is only applied on a one-dimensional parameterized hyperbolic conservation law;

- similarities can be found with Lorente et al. [136]. The authors seek local regions of high gradients and correct them by separating the full signal into steep jumps and smooth profiles, which are used to train different ROM. Thus, the predictions at untried values mix predictions from the ROM trained on steep jumps and on smooth profiles. The final signal is reconstructed with a well captured shock. This approach needs to calibrate several parameters in order to ensure a proper separation between the jump and the smooth profiles. This represents a significant limit to the method;

- Bergmann et al. [16] split the computational domain with a leave-one-out strategy. The a priori estimated error is computed on all the domain. If the value exceeds a given threshold,

the point is associated with the high-fidelity region managed by the FOM. This region is extended by a rectangular shape in order to avoid irregular shapes. Obviously, the value given to the threshold plays a crucial role, as a too small value leads to a high fidelity region covering all the domain. It can be noted that an overlapping zone is also introduced, defined by a second threshold value performing an optimal transition between FOMs and ROMs. This method has been used to solve aerodynamic shape optimization of a car section and shows a reduction 5% of the drag coefficient;

- Iuliano and Domenico [102] conceive a zonal method in order to solve optimization problems. In that case, the high fidelity region is manually defined near the wall. The surrogate-based optimization coupled with the zonal CFD/POD model performs very well on the shape optimization of a RAE2822 airfoil, especially by finding a profile with a weak shock. This is explained by the fact that non-linearities occurring near the wall are fully captured by the high fidelity model, which was not the case for the ROM, but with a significant computational endeavor.

All the methods presented above are only applied on one-dimensional or two-dimensional problems. Nevertheless, the capabilities of zonal methods have been demonstrated on a three-dimensional test case, an urban street canyons [205]. Instead of using a mixed full/reduced order model, a set of local POD basis with independent numbers of POD modes is built on the computational domain. This strategy have resulted in a slight improvement of the non-intrusive ROM accuracy applied to a complex flow problem over widely varying range of scales. Finally, the management of boundary conditions between the different models appears as a critical step for zonal methods. The accuracy of the predictions must be reliable, as the ROM is used to provide the value at the boundary. Indeed, the FOM can be strongly affected by spurious boundary conditions leading to numerical instabilities or large discrepancies, possibly worse than a global ROM which has a lower computational cost.

2.3.2 Parameter space decomposition

The parameter space decomposition aims at combining different local models, in the parameter space, into a final one in order to improve the overall accuracy, as global models can be very limited compared to multiple models when different kinds of physical regimes are present. Local models are more specialized and trustworthy but only on smaller parts of the parameter space. In the context of scalar models, this type of approach is called mixture of experts and has been introduced in the 1990s to solve regression and classification problems [109, 208]. Mixtures of expert model are competitive for regression problems with non-stationary and piecewise continuous data. That is why it appears well suited to applications featuring discontinuities.

For example, Bettebghor et al. [17] have developed a specific strategy, based on a Gaussian mixture of experts, able to approximate discontinuous functions. Such features are very common in structural analysis due to the various combination of load cases. The methodology has been evaluated on two analytical functions and an engineering problems. It turns out that mixtures of experts dramatically improve accuracy of the predictions compared to the best global surrogate model. Recently, mixtures of experts have also been used to evaluation aircraft fuel burn over a complete mission [129]. The authors highlight the clear superiority of mixtures of experts compared to conventional surrogate models to approximate aerodynamic coefficients with complex shapes. Two aircraft configurations are tested for various mission profiles. Mixtures of experts are below 2% of error in the range prediction while the universal Kriging error ranges from 6% to 20%. The both examples presented here rely extensively on machine learning methods, in particular unsupervised

learning for the separation of the solutions and supervised learning, in the case of Liem et al. [129], in order to associate new predictions with the correct cluster.

The parameter space decomposition has recently emerged in ROM literature. The central idea is the same than for scalar problems: considering only a restriction of the full set of training snapshots in order to have local surrogates more adapted to the solution. The method can take different forms depending on the type of problem or surrogate model. One may argue that the capability of compression of POD is weakened by increasing the number of bases, and that the robustness of the method computing the reduced coordinates can decrease with the reduction of training samples, which are spread on different models. But on the contrary, for cases with various physical regimes, the local models enable a clear separation of the phenomena, which improves the surrogate models. The advantage is two-fold: the local POD basis is more representative and interpolation tools (or projected equations) gain in flexibility by being heterogeneous on the parameter space. As mentioned previously, the POD shows elliptic properties, thus these kind of local approaches can demonstrate similarities with the discontinuous Galerkin method to tackle hyperbolic problems.

As regards intrusive ROMs, the first attempts of parameter domain decomposition do not use machine learning. Cizmas et al. [38] propose a method in order to accelerate ROMs. The transient part is separated from the steady part. Two splitting methods are tested: they monitor the variation of the reduced coordinates and the ratio between CPU time and physical time. These two strategies identify successfully the transition between the two regimes by using a given threshold defined in advance. As each subset is associated with a specific regime, the first modes contain more energy and fewer modes are needed to reconstruct the ROM, leading to an increase of the speed-up by about 50%. Haasdonk et al. [86] introduce a recursive subdivision of the parameter domain based on quadtree. For each leaf element of the grid, if the error bound or the size of the basis exceed given thresholds, the element is refined into several subdomain. Then, the process is restarted until all the thresholds are not exceeded. The method has been tested on a two-dimensional case and the prediction errors are considerably reduced. However, such a method based on quadtree is very sensitive to the dimension of the parameter space.

Strategies relying on machine learning have been developed in order to build intrusive or non-intrusive ROMs. The core idea remains the same but methods have been adapted to ROMs: an unsupervised learning algorithm groups similar snapshots into clusters, local ROMs are built on each cluster, and a classifier assigns the input parameter of each prediction to the best suited local ROM. Replacing the global POD basis with several local POD bases may seem counterintuitive but this strategy has already been applied successfully to build intrusive parametric ROMs for fluid-structure-electric interaction problems [7] and aerodynamic problems [202]. Peherstorfer et al. [163] propose a sophisticated method by coupling parameter domain decomposition and discrete empirical interpolation method presented in the introduction. Thus reconstructing the nonlinear term of the projected equations, usually expensive to evaluate, can be approximated with different distributions of the interpolation points for each cluster of the parameter domain decomposition. It widely reduces the prediction error of the ROMs in the case of a simple reactive flow. Finally, to the best of the author's knowledge, the only method of non-intrusive local ROMs has been developed in the context of the aero-icing certification [210, 211].

The parameter space decomposition can be seen as a blessing of abstraction, countering the curse of dimensionality: if the local models are more adapted to physics, they need to learn less structures, compared to global approaches.

2.3.3 Nonlinear dimensionality reduction

POD assumes that flow solutions can be approximated by a linear decomposition, where the basis functions span an optimal subspace only among all linear decompositions. Even if POD is a linear method, it is applicable to nonlinear problems as no linear hypothesis is imposed on the process. The linearity lies in the fact that only linear subspaces are sought but the final ROM can show nonlinear features if they are present in the initial FOM [44]. Nevertheless, highly nonlinear features with strong discontinuities can still represent a major challenge for POD. For example, shocks in transonic flows are badly approximated. For this reason, more sophisticated methods based on curved manifolds have been applied to build non-linear ROM, also referred to as nonlinear dimensionality reduction.

Mifsud [150] introduces a nonlinear POD methodology, called kernel POD or kernel PCA. It extends POD using kernel methods. The original idea is to map the nonlinear data to a feature space of higher dimension, where data can be more easily analysed by linear methods, such as POD. Kernel POD are applied to transonic flows over the two-dimensional RAE2822 airfoil. However, predictions of highly nonlinear features are not signifcantly improved by the kernel POD approach for this test case compared to classical non intrusive POD-based ROMs.

Franz et al. [77, 76] also use a nonlinear dimension reduction method, called Isomap, which belongs to manifold learning approaches. The flow simulations form a manifold approximated isometrically by a low-dimension embedding space. Isomap is then associated with an interpolation method in order to build the full ROM. In addition, an original resampling strategy has been coupled with the manifold learning algorithm in order to space-fill the embedding space instead of the parameter space. The idea is to reduce the lack of information at minimum cost by analysing geometrical characteristics of data, obtained by a manifold method. The full process is called *Isomap+I* and is applied to a two-dimensional NACA64A010 airfoil and a three-dimensional XRF-1 fuselage-wing configuration. The predictions of transonic flows with shocks are improved compared to classical interpolation-based POD methods. Lastly, one can note that the *Isomap+I* can also be seen as a parameter space decomposition, since the back-mapping ensuring the conversion from the embedding space to the physical space uses only a combination of nearest neighbors on the estimated manifold to compute a prediction. Therefore, for each prediction, only the best-suited snapshots are considered.

2.4 Industrial applications

Industrial-like configurations are rarely investigated in the literature. Nevertheless, few example of complex applications can still be cited. They are marked as "Industrial cases" in Table 2.2. Fossati [74] applies non-intrusive ROMs to various aerodynamic cases of industrial relevance for viscous turbulent flows: a truncated wing, an aircraft configuration, and an helicopter rotor in hover. In addition to the encouraging results in term of ROM accuracy, a complete framework is implemented including DOE (Centroidal Voroni Tesselation), POD, Bayesian Kriging and an a priori error estimation with leave-one-out. This similar framework is also employed to solve a aero-icing problem on a configuration with fuselage, wing, pylon, and engine nacelle, highlighting the capability of such method to explore both aerodynamic and icing envelope [75].

Roy et al. [174] perform an uncertainty quantification analysis on a LS89 blade cascade. Even if the geometry remains simple, the authors use Large Eddy Simulation on a 20 million cells mesh. Such computational costs represent a real challenge but they are necessary to demonstrate capabilities of the non-intrusive ROMs to deal with complex cases. The influence of both the

turbulence intensity and the angle of attack on heat transfer coefficients are illustrated. It clearly seems the turbulence intensity contributes to 70% of the variance of the heat transfer coefficient.

In the context of the Digital-X project, DLR has developed various ROM methodologies applied to a configuration of XRF-1 aircraft, a 3D test case of a transonic wing-body transport aircraft [193]. For example, ROMs are integrated in a framework to compute aeroelastic loads at different flight conditions. The structural displacements are in good agreement between coupled high-fidelity models and ROMs, even if the prediction of the pressure distribution shows significant discrepancies on the suction surface, near the shock wave region. ROMs also help to determine critical load cases in the sizing process, to perform aero-data production and structural optimization. However, in the context of the MDO process, the ROM is only applied on low-fidelity data (linearized Navier Stokes equations).

Regarding supersonic flows, Mifsud et al. [152] couple POD and radial basis functions for a parametric study of weapon aerodynamics. They consider a three-dimensional flow around a fin-stabilized missile with three parameters: Mach number, incidence, and flare base radius. The results suggest that a reliable low-cost high-fidelity tool can be built: from a small number of simulations, the pressure and density contours are well predicted.

Chen et al. [36] propose an aerothermodynamic application of interpolation-based ROM for hypersonic vehicles. The geometry is based on the Lockheed F-104 Starfighter wing. Thermal and chemical non-equilibrium models are considered in the FOM. The mesh counts 819.000 nodes and 3 parameter vary: Mach number, angle of attack, and altitude. ROMs are validated by leave-one-out cross validation. Predicting the temperature requires an important number of samples, between 100 and 300, in order to reach satisfactory accuracy whatever methods used.

Margheri and Sagaut [142, 143] combine POD, Kriging, and sensitivity decomposition with the objective of quantifying the uncertainties of urban pollutant dispersion using Lattice Boltzmann Methods. The more challenging computation simulates a full urban area from two to five input variables. The computational grids count 6 millions of cells and a single simulation requires about 1.000 CPU hours.

2.5 Positioning the present research in the current state of the art

This section describes the positioning of the thesis work compared to the existing literature for non-intrusive ROMs. Table 2.1 and Table 2.2 summarize the different works presented in this section. Various information is displayed: authors, use of general improvement methods or local methods, fields of application, and general comments. In particular, the type of validation for the ROMs is indicated in column "Statistical validation" by the number of points in the validation set or if LOO strategies are used. Moreover, application cases are specified: analytical, academical, or industrial.

The improvement strategies based on resampling offer a very important advantage: increasing locally the quality of the model where physics is more challenging. This notion of local improvement appears very important in order to deal with flows showing different flow regimes which do not need the same level of accuracy. These methods appear promising for this thesis, as they distribute optimally the training samples. Thus, they will be investigated in this thesis.

Multi-fidelity approaches have also shown interesting capabilities to both ensure a great exploration of the parameter space with the low-fidelity data and improve the global accuracy with well-chosen high-fidelity data. However, an important assumption plays a key role in such approaches: the low-fidelity levels must be able to model a part of the physics. In the context of

this thesis, the validity of partially converged simulations or coarse meshes is difficult to assess. Moreover, approximating flow fields involving shock waves and complex three-dimensional geometries requires a high level of accuracy which appears hard to reach for simplified models. For these reasons, multi-fidelity methods have not been considered in details in this work.

As regards gradient-based methods, even if an adjoint solver is available in the CFD solver used in this thesis, the adjoint has not been used due to the signifcant increase in the computational cost for each simulation. Consequently, gradient-based methods are not explored. As regards active subspace methods, applying the change of variables to the reduced coordinates may be very challenging. These functions do not reflect any physical behaviour as they are derived from the computation of the POD modes. The assumption that each function has a main direction appears difficult to be validated. Furthermore, each reduced coordinate can have different main directions. Thus active subspace methods are not explored. At last, quantitative sensibility analysis methods, such as Sobol' indices, may help to improve the understanding of the ROMs and to simplify the training processes.

The major issue regarding non-intrusive ROMs applied to industrial cases relates to the computational cost of such simulations, as industrial applications usually imply complex phenomena with a high number of input parameters. Hence, an important number of samples is needed to reach a satisfactory accuracy. In addition, it leads to a lack in clear statistical validations in term of accuracy. To the best of the author's knowledge, statistical validations are very rarely performed in the literature, as Table 2.1 and Table 2.2 suggest. They indeed require an important supplementary computation budget which is usually not available. For this reason, the classic way to validate non-intrusive parametric ROMs is obtained by an a priori estimation error, without any high-fidelity computation, or by computing few simulations not to close from the training samples [141, 75, 76]. This trend is also observed in Table 2.1 and Table 2.2. Therefore, a specific emphasis will be put in the application of innovative non-intrusive ROMs on industrial-like applications coupled with statistical validations of the model.

Local methods, both domain and parameter space decomposition, appear very promising. Usually, the flow around an aircraft encounters discontinuities in both space and the parameter domain during a mission, as the flow can be either subsonic or transonic in function of the speed but also in function of the angle of attack. The shocks present in the transonic regime are responsible of strong discontinuities in space domain. In the context of intrusive ROMs, the parameter space decompositon has only be developed for analytical cases, integrated quantities [202], or simple academical cases [163]. The only development of non-intrusive local methods concerns the field of aero-icing [210, 211]. All these elements emphasize the possibility to develop new parameter space decomposition for the specific aerodynamic applications. Hence, investigating these methods are going to be central in this thesis with the development of the Local Decomposition Method (LDM), which couples machine learning algorithms with classical surrogate models in order to decompose the parameter space.

Authors	Year	General improvement	Local Method	Statistical validation	Fields
Bui-Thanh et al. [24]	2003			3 pts	Aerodynamics
Bui-Thanh et al. [25]	2004			2 pts	Aerodynamics
Mifsud [150]	2008			2 pts	Aerodynamics
Qamar and Sanghi [166]	2009			/	Aerodynamics
Mifsud et al. [152]	2010			2 pts	Aerodynamics
Zimmerman et al. [214]	2010			2 pts	Aerodynamics
Braconnier et al. [22]	2011	X		16 pts	Aerodynamics
Fossati and Habashi [75]	2013	X		4 pts	Aero-icing
Iuliano and Domenico [102]	2013		X	50 pts	Aerodynamics
Malouin et al. [141]	2013	X		4 pts	Aerodynamics
Walton et al. [199]	2013			1 freq	Aerodynamics
Zimmermann [213]	2013	X		/	Fluid dynamics
Kato and Funazaki [111]	2014			LOO	Turbomachinery
Toal [186]	2014	X		80 pts	Turbomachinery
Franz [76]	2015		X	10 pts	Aerodynamics
Qiu and Bai [167]	2015			10 pts	Aerodynamics
Chen et al. [36]	2015			LOO	Aerothermal
Fossati [74]	2015	X		LOO	Aerodynamics
Zhan et al. [210]	2015		X	LOO	Aero-icing
Zhan et al. [211]	2016		X	LOO + 4 pts	Aero-icing
Dolci and Arina [51]	2016			14 pts	Aerodynamics
Xiao et al. [204]	2016			/	FSI
Benamara et al. [14]	2016	X		LOO	Aerodynamics
Margheri and Sagaut [143]	2016			/	Fluid dynamics
Verveld et al. [193]	2016			LOO	Aerodynamics
Xiao et al. [205]	2017		X	/	Aerodynamics
Wang et al. [200]	2018			1 pt	Combustion
Roy et al. [174]	2018	X		LOO	Turbomachinery
Dupuis et al. [54]	2018		X	300 pts	Aerodynamics
Dupuis et al. [53]	2018		X	300 pts	Aerodynamics

Table 2.1: Summary of non intrusive ROMs, part I. (FSI: Fluid Structure Interaction)

Authors	Analytical cases	Academical cases	Industrial cases	Comments
Bui-Thanh et al. [24]		X		First application
Bui-Thanh et al. [25]		X		Gappy POD
Mifsud [150]		X		Kernel PCA
Qamar and Sanghi [166]	X		X	Extrapolation marching
Mifsud et al. [152]			X	Supersonic
Zimmerman et al. [214]	X			Residual minization
Braconnier et al. [22]	X	X		Resampling
Fossati and Habashi [75]		X	X	Bayesian Kriging
Iuliano and Domenico [102]		X		Zonal method
Malouin et al. [141]		X		Multi-fidelity
Walton et al. [199]		X		Unsteady
Zimmermann [213]	X			GEK
Kato and Funazaki [111]		X		Optimization
Toal [186]	X			Multi-fidelity
Franz [76]		X	X	Manifold learning
Qiu and Bai [167]		X		Variable geometry
Chen et al. [36]			X	Hypersonic
Fossati [74]		X	X	Bayesian Kriging
Zhan et al. [210]		X		Local method
Zhan et al. [211]			X	Local method
Dolci and Arina [51]		X	X	Optimization
Xiao et al. [204]		X	X	RBF
Benamara et al. [14]		X		Multi-fidelity/resampling
Margheri and Sagaut [143]	X	X	X	Uncertainty quantification
Verveld et al. [193]			X	Aeroelasticity
Xiao et al. [205]	X			Zonal method
Wang et al. [200]		X		Unsteady
Roy et al. [174]	X		X	Resampling
Dupuis et al. [54]	X	X		LDM
Dupuis et al. [53]			X	LDM

Table 2.2: Summary of non intrusive ROMs, part II.

Part II

Reduced model methods

Table of Contents

CHAPTER 3

Non-intrusive reduced-order model based on POD

Résumé

Ce chapitre fournit une introduction étendue de l'ensemble des méthodes nécessaires à la construction d'un modèle réduit. Cette dernière se décompose généralement en six étapes :

- **le plan d'expérience**, ou Design of Experiments (DOE) en anglais, qui détermine les points d'entrainement ;
- **le calcul haute-fidélité** de ces points d'entrainement ;
- **la réduction de dimension** qui calcule les bases propres et les coordonnées réduites via la POD ;
- **la régression** dans l'espace des paramètres des coordonnées réduites ;
- **le réechantillonnage adaptif** (optionnel) ;
- **les prédictions** avec le modèle réduit assemblé.

La section 3.2 détaille les éléments mathématiques à l'origine de la Décomposition Orthogonale aux valeurs Propres. Elle insiste sur sa formulation en dimension finie et son application sur un maillage.

La section 3.3 présente différentes méthodes de régression : la régression polynomiale, la régression à base de machine à vecteurs de support, les réseaux de neurones, les fonctions à base radiale et les processus gaussiens. Ces derniers sont identifiés comme la méthode la plus performante dans le cadre de la régression de coordonnées réduites pour des problèmes de mécanique des fluides. En effet, ils sont particulièrement performants lorsque le nombre de paramètres d'entrée et le nombre de points d'entrainement sont faibles. De plus, ils peuvent

apprendre des fonctions complexes et fournissent une estimation de la variance en chaque prédiction.

La section 3.4 décrit les plans d'expériences. Différents outils statistiques sont introduits afin de mesurer leur qualité puis un large panel de plan d'expériences est présenté : factoriel, maximin et minimax, hypercube latin ou encore séquences à faible discrépance. Ces dernières se sont montrées être les plus adaptées aux modèles réduits. Effectivement, elles possèdent à la fois des propriétés déterministes, séquentielles (le nombre de points peut être augmenté de manière séquentielle sans altérer les propriétés du plan d'expérience) et de remplissage de l'espace des paramètres.

Enfin, la section 3.5 définit des métriques de validation des modèles de substitution : coefficient de prédictivité, erreur quadratique moyenne ou encore erreur normalisée. Ces métriques peuvent être calculées par différentes stratégies. La plus simple consiste à séparer les points de calculs en une base d'entrainement servant à construire le modèle de substitution et une base de test permettant, comme son l'indique, de le tester. Une autre méthode très populaire, la validation croisée, construit plusieurs modèles de substitution en utilisant à chaque fois des bases d'entrainement et de test différentes et estime la qualité du modèle en faisant la moyenne de la métrique. Cette méthode présente comme principal avantage d'utiliser l'ensemble des points de calculs dans la construction et l'évaluation du modèle.

This chapter presents a general method to build non-intrusive reduced order models (ROMs) based upon POD. As mentioned in the introduction, non-intrusive methods do not need access to the system of equations modeling physics and do not modify the source code of the simulator. First, a clear framework for non-intrusive ROM is presented. Then, a broad review of their applications on fluid and aerodynamics problems illustrates their capabilities and weaknesses. Indeed, simple flows and geometries do not raise major issues. However the accuracy strongly decreases if non-intrusive ROMs encounter different flow conditions. Finally, the more classical components used in building non-intrusive reduced-order models are detailed, such as Dimension Reduction, Design of Experiments and Numerical Regression.

3.1 General definition of the problem

As regards symbol definitions, capital boldface characters are used for matrices and tensors, lower-case boldface characters are employed for vectors, and lower-case characters for scalars and numbers.

3.1.1 Notations

As explained in the introduction, the goal of ROMs is to substitute the CFD model in order to estimate a field of specific quantity of interest f by a numerical approximation $\widetilde{f}$. The quantity of interest, for example a pressure or a temperature, is written as a function of two inputs: the mesh coordinate $\boldsymbol{x} \in \mathbb{R}^3$ and a vector of the input parameters $\boldsymbol{\chi} \in \mathbb{R}^p$ which defines the simulations, where p is the number of input parameters:

$$\begin{aligned} f\colon \mathbb{R}^3 \times \mathbb{R}^p &\to \mathbb{R} \\ \boldsymbol{x} \quad \times \boldsymbol{\chi} &\mapsto f(\boldsymbol{x}, \boldsymbol{\chi}). \end{aligned} \tag{3.1}$$

The function f is evaluated on a mesh Ω of size d formed by a list of three-dimensional coordinates, such as $\Omega = [\boldsymbol{x_1}, \cdots, \boldsymbol{x_d}]^T \in \mathbb{R}^{d\times 3}$, where $\boldsymbol{x_i}$ indicates the coordinate of the i-th point of the mesh,

which can also be one or two dimensional with no loss of generality. Figure 3.1 illustrates computer experiments.

The components of the vector $\boldsymbol{\chi}$ correspond to specific attributes of the CFD simulations, such as the value of a boundary condition, the nature of the flow or a numerical parameter. Building a surrogate model is based on training samples (also called training set, training data or training sampling), generated from results of several CFD simulations. In other words, quantities of interest are computed on the mesh for n different training input parameters $\boldsymbol{\chi_t}$, forming the training set of input parameters:

$$\boldsymbol{X_t} = [\boldsymbol{\chi_{t_1}}, \cdots, \boldsymbol{\chi_{t_n}}]^T \in \mathbb{R}^{n \times p}. \tag{3.2}$$

The training snapshot matrix $\boldsymbol{S_t}$ regroups the quantities of interest on the mesh and is defined by:

$$\boldsymbol{S_t} = \begin{bmatrix} f(\boldsymbol{x_1}, \boldsymbol{\chi_{t_1}}) & \cdots & f(\boldsymbol{x_d}, \boldsymbol{\chi_{t_1}}) \\ \vdots & & \vdots \\ f(\boldsymbol{x_1}, \boldsymbol{\chi_{t_n}}) & \cdots & f(\boldsymbol{x_d}, \boldsymbol{\chi_{t_n}}) \end{bmatrix} \in \mathbb{R}^{n \times d}. \tag{3.3}$$

In the interests of simplifying notations, the mesh is ommitted as it remains constant in our applications. The vector-valued function $\boldsymbol{f}$ of the quantities of interest is defined by:

$$\begin{aligned} \boldsymbol{f} \colon \mathbb{R}^p &\to \mathbb{R}^d \\ \boldsymbol{\chi} &\mapsto \boldsymbol{f}(\boldsymbol{\chi}). \end{aligned} \tag{3.4}$$

The same simplification is applied on the training snapshot matrix, which can be rewritten in a shorter form:

$$\boldsymbol{S_t} = \begin{bmatrix} \boldsymbol{f}(\boldsymbol{\chi_{t_1}}) \\ \vdots \\ \boldsymbol{f}(\boldsymbol{\chi_{t_n}}) \end{bmatrix} \in \mathbb{R}^{n \times d}. \tag{3.5}$$

One introduce as well the fluctuating quantity $\boldsymbol{f'}$ where the mean snapshot has been subtracted :

$$\boldsymbol{f'}(\boldsymbol{\chi}) = \boldsymbol{f}(\boldsymbol{\chi}) - E[\boldsymbol{f}(\boldsymbol{\chi})] = \boldsymbol{f}(\boldsymbol{\chi}) - \frac{\sum_{i=1}^{n} \boldsymbol{f}(\boldsymbol{\chi}_i)}{n}, \tag{3.6}$$

and its corresponding matrix $\boldsymbol{S'_t}$:

$$\boldsymbol{S'_t} = \begin{bmatrix} \boldsymbol{f'}(\boldsymbol{\chi_{t_1}}) \\ \vdots \\ \boldsymbol{f'}(\boldsymbol{\chi_{t_n}}) \end{bmatrix} \in \mathbb{R}^{n \times d}. \tag{3.7}$$

Once trained, the surrogate model aims at computing the snapshot matrix of the predictions $\boldsymbol{S_p}$ for the set of prediction sample $\boldsymbol{X_p} = [\boldsymbol{\chi_{p_1}}, \cdots, \boldsymbol{\chi_{p_m}}]^T$:

$$\boldsymbol{S_p} = \begin{bmatrix} \widetilde{\boldsymbol{f}}(\boldsymbol{\chi_{t_1}}) \\ \vdots \\ \widetilde{\boldsymbol{f}}(\boldsymbol{\chi_{t_m}}) \end{bmatrix} \in \mathbb{R}^{m \times d}, \tag{3.8}$$

where m is the number of predictions (the size of the prediction set $\boldsymbol{X_p}$) and $\widetilde{\boldsymbol{f}}$ is the vector-valued function of the surrogate model evaluating the quantity of interest on the mesh Ω. It has the same definition 3.4 than $\boldsymbol{f}$.

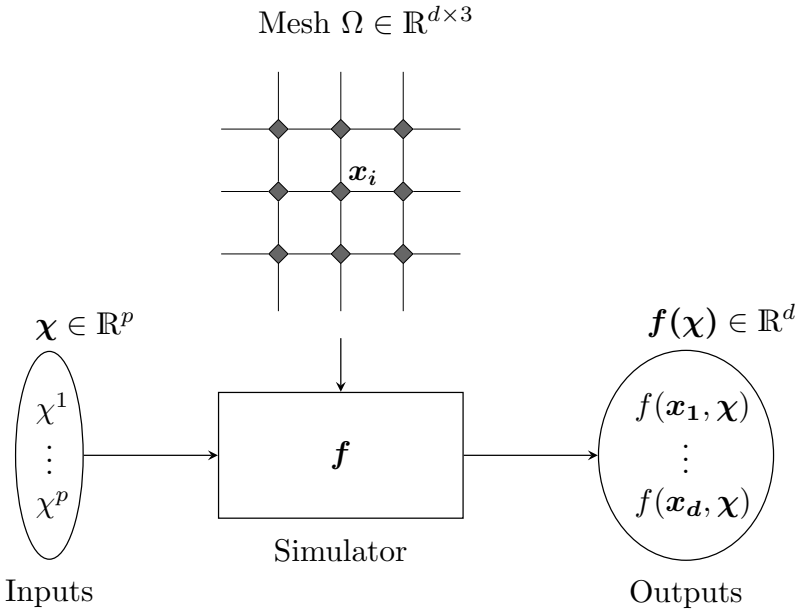


Figure 3.1: Computer experiments.

3.1.2 Global picture of the classical method

This section provides a global picture of the non-intrusive ROMs. Figure 3.2 depicts the workflow of the process as it has been widely employed in the literature, in particular for aerodynamics applications.

The first step of any ROM starts with the exploration of the domain in order to find regions of main interest, such as high gradients, peak values, or stable regions. The exploration is carried out by generating an initial sampling, also called initial Design of Experiments, composed of an a priori training set $\boldsymbol{X_t}$. It aims to provide a space-filling of the input space parameters which extracts the maximum amount of information. In other words, the DOE can be defined as a selection of samples in the parameter space, which are used to define physical or numerical simulations to extract meaningful information on the input-output relationship. The DOE aims at filling up the domain as evenly as possible. The notion of DOE is discussed extensively in a next section.

Computationally expensive high fidelity simulations are then computed at the DOE's samples, forming the training snapshot matrix $\boldsymbol{S_t}$. The dimension of the latter is then reduced by typical approaches determining an orthonormal basis $\boldsymbol{\phi}$. Then the large system can projected onto this reduced space to derive the reduced-order model. Therefore, the basis vectors $\boldsymbol{\phi}$ of size M, upon which the quantity of interest $\boldsymbol{f}$ can be expanded, is computed. They represent generally the main behaviors of the high-fidelity model and the projection of f provides the corresponding coefficient in the basis, called reduced coordinates a. In particular, the proper orthogonal decomposition (POD) has become a popular method of deriving the orthonormal basis.

Determining the reduced coordinate values for an untried set of input parameters represents one of the central issues of surrogate models. A classical and proven solution consists of using a regression method, such as Gaussian Process Regression (GPR) or Radial Basis Function (RBF). Finally, extra samples can be computed to improve the accuracy of the model. This step, called resampling is generally managed by an estimation of a quality criterion. Lastly the process ends by building the final form of the surrogate model $\widetilde{f}$.

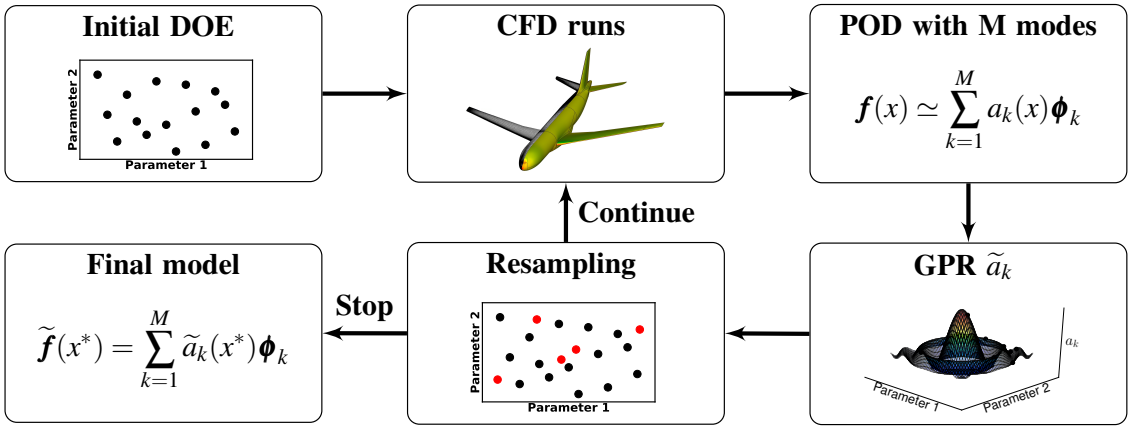


Figure 3.2: Flowchart of the POD/GPR method.

This non-intrusive ROM method is the common basis of the further developments presented in this thesis and serves as a reference for assessing different methods. The following sections give a review of how such methods have been applied, introduce the mathematical formalisms and provide details on each step of the process.

3.2 Dimension reduction by Proper Orthogonal Decomposition

POD is a linear method based on the identification and extraction of coherent structures (or main directions) for a given data set by solving an eigenvalue problem providing an eigen-basis. It can be shown this latter gives an optimum basis in term of energy representation [44]. This means for a given number of basis vectors, called modes, the POD contains more energy than any other basis. All the reduced-order process of POD results in the computation and the selection of a few number of these modes containing a very large amount of energy. The remaining modes are cut off. Each basis vector is associated with a scalar value, also called reduced coordinate, which depends on the parameter space.

Although POD is a linear procedure, there is no assumptions about the linearity of the system, which can explain its popularity in data analysis and model reduction. Moreover, two main versions of the POD exist: the continuous POD and the discrete POD. While the first version is associated with a theoretical framework, best suited to a broad analysis and the derivation of the main properties, the discrete version has been developed to perform computations and to be implemented in a practical manner. Indeed, the outputs of the computation codes are usually discretized in order to evaluate the partial differential equations (finite element methods, finite volume methods, etc...).

As a first step, the continuous problem and the main properties will be presented in the section 3.2.1. Then, the discrete POD and the Singular Value Decomposition (SVD) will be discussed in section 3.2.2. Finally, practical computational aspects are discussed.

3.2.1 Introduction to the continuous problem

In order to remain consistent with the abundant literature about POD, Equation 3.1 is rewritten, only for the POD section, by substituting the parameter χ by the time $t \in \mathcal{T}$, where $\mathcal{T}$ is an

interval of time. The function z is introduced such as:

$$\begin{aligned} z\colon \Omega \times \mathcal{T} &\to \mathbb{R} \\ \boldsymbol{x} \times t &\mapsto f(\boldsymbol{x}, t). \end{aligned} \tag{3.9}$$

All the following derivations remain valid for the input parameter $\boldsymbol{\chi}$, only some operators must be slightly adapted.

The central idea of POD is to reduce the information by expending the function z into K basis functions, such as:

$$z(\boldsymbol{x}, t) \approx \sum_{k=1}^{K} a_k(t)\phi_k(\boldsymbol{x}), \tag{3.10}$$

where a_k is the scalar reduced coordinate associated with the k-th POD mode ϕ_k. The limit of the decomposition, as K approaches the infinity, is z. However, this decomposition is not unique. Therefore, several specific conditions have been imposed to the basis functions in order to uniquely determine ϕ and a_k:

- The basis functions have been chosen as orthonormal. The definition of the inner product $(\cdot)$ introduces the domain $\mathcal{D}$, which can indifferently be the time interval $\mathcal{T}$ or the mesh Ω. In the same way, the variable s can be the time variable t associated with $\mathcal{T}$ or the space variable x associated with Ω. This imprecise definition is used in order to remain as general as possible. The choice for $\mathcal{D}$, between space and time, leads to two different computation methods of the POD basis, detailed in the section 3.2.2. By considering the inner product (.) of the square-integrable functions $\mathcal{L}^2(\mathcal{D})$, the orthonormality leads to:

$$(\phi_i(\boldsymbol{x}), \phi_j(\boldsymbol{x})) = \delta_{i,j}, \forall \boldsymbol{x} \in \mathbb{R}^3,\ \forall i, j \in [1, K]^2, \tag{3.11}$$

  where $\delta_{i,j}$ is the Kronecker delta symbol:

$$\delta_{i,j} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if otherwise} \end{cases}, \forall i, j \in [1, K]^2. \tag{3.12}$$

- The basis functions are seen as optimal in terms of the representativeness of the data, they are the most "probable" realizations of the data [44]. The interpretation of this property will lead to two equivalent approaches detailed in the following part.

A direct application of the inner product between the decomposition 3.10 and the basis functions ϕ_k gives directly the expression of the reduced coordinates a_k:

$$a_k(t) = (z(\boldsymbol{x}, t), \phi_k(\boldsymbol{x})), \forall k \in [1, K]. \tag{3.13}$$

3.2.1.1 Two equivalent representations

The POD has been presented in the literature in two different ways: as an approximation method [34] and as seeking the best correlated structures on average with the realizations [138]. The first way of expressing POD is written on the form of a minimization problem:

$$\begin{gathered} \min_{\phi_1, \cdots, \phi_K} \left\langle \left\| z(\boldsymbol{x}, t) - \sum_{k=1}^{K} a_k(t)\phi_k(\boldsymbol{x}) \right\|_2^2 \right\rangle, \\ \text{subject to } (\phi_i, \phi_j) = \delta_{i,j}, \end{gathered} \tag{3.14}$$

where $\|\cdot\|$ defines the norm on $\mathcal{L}^2(\mathcal{D})$ corresponding to the inner product $(\cdot,\cdot)$ and where the angle brackets $\langle\cdot\rangle$ denote the appropriate average for the problem under consideration. The objective is to find the orthonormal basis functions ϕ_i such as the projection of z into the subspace generated by the K basis functions is the best approximation, given by the minimal mean square error. A geometrical interpretation can also be given where the minimal mean distance between z and its projection into the subspace generated by the K basis functions is sought.

The second way of expressing POD aims at maximizing the mean projection of the function z into the orthonormal basis function $\phi_1, \cdots, \phi_K$, leading to the following maximization problem [67]:

$$\begin{aligned} &\max_{\phi_1,\cdots,\phi_K} \sum_{k=1}^{K} \left\langle |(z(\boldsymbol{x},t),\phi_k)(\boldsymbol{x})|^2 \right\rangle, \\ &\text{subject to } (\phi_i,\phi_j) = \delta_{i,j} \ \forall i,j \in [1,K]^2. \end{aligned} \tag{3.15}$$

The approximation 3.14 and the problem of best correlated structures 3.15 are shown to be similar, given rise to an eigenvalue problem [180, 92, 196]. For this reason, only derivations for the coherent structures approach ginven in Equation 3.15 are detailed in the following section. Interested readers can refer to [34, 44] for details on the approximation problem.

3.2.1.2 The Fredholm equation

The problem 3.15 is simplified by seeking only the basis function ϕ given the maximum projection of z:

$$\begin{aligned} &\max_{\phi} \left\langle |(z(\boldsymbol{x},t),\phi)|^2 \right\rangle, \\ &\text{subject to } (\phi,\phi) = 1. \end{aligned} \tag{3.16}$$

Solving this problem introduces the auto-correlation operator $\mathcal{R}$:

$$\begin{aligned} \mathcal{R}\colon \mathcal{L}^2 &\to \mathcal{L}^2 \\ \phi &\mapsto \mathcal{R}\phi, \end{aligned} \tag{3.17}$$

where

$$\begin{aligned} \mathcal{R}\phi\colon \mathbb{R} &\to \mathbb{R} \\ s &\mapsto \mathcal{R}\phi(s) = \int_D R(s,s')\phi(s')ds', \end{aligned} \tag{3.18}$$

with $R(s,s') = \langle z(s) \otimes z(s') \rangle$. It can be shown that $\mathcal{R}$ is a self-adjoint and non negative operator. Indeed, the calculation of $(\mathcal{R}\phi,\phi)$ gives:

$$\begin{aligned} (\mathcal{R}\phi,\phi) &= \left(\int_D R(s,s')\phi(s')ds', \phi(s) \right) \\ &= \int_D \int_D \langle z(s) \otimes z(s') \rangle \phi(s')ds'\phi(s)ds \\ &= \left\langle \int_D z(s)\phi(s)ds \int_D z(s')\phi(s')ds' \right\rangle \\ &= \left\langle |(z,\phi)^2| \right\rangle \geq 0, \end{aligned} \tag{3.19}$$

and a direct derivation shows that:

$$(\mathcal{R}\phi,\psi) = (\phi,\mathcal{R}\psi) \ \forall \phi,\psi \in \mathcal{L}(\mathcal{D})^2. \tag{3.20}$$

The commutativity of the spatial and temporal integration operators is assumed. According to the spectral theorem, a solution to the maximization problem 3.16 exist and must satisfy the following largest eigenvalue problem [44]:

$$\mathcal{R}\phi = \lambda\phi. \tag{3.21}$$

It can be written on the form of the Fredholm equation:

$$\int_D R(s,s')\phi(s')ds' = \lambda\phi(s). \tag{3.22}$$

As the operator $\mathcal{R}$ is linear, self-adjoint, and non-negative, a complete and countable orthonormal basis $\{\phi_i\}_{i=1}^{\infty}$ exist, associated with a sequence of nonnegative eigenvalues $\{\lambda_i\}_{i=1}^{\infty}$, such as [44]:

$$\mathcal{R}\phi_i = \lambda_i\phi_i, \text{ where } \; i = [1, \cdots, +\infty]. \tag{3.23}$$

The eigenvalues are ranked from highest to lowest:

$$\lambda_1 \geq \lambda_2 \geq \cdots \geq 0 \text{ with } \lim_{k\to\infty} \lambda_k = 0. \tag{3.24}$$

It has been shown that the first K eigenfunctions and eigenvalues are solutions to Equation 3.15 and 3.14 [196]. This results can be interpreted as follows: each basis function ϕ_i is solution of Equation 3.16 where the projection into the previous subspace ϕ_{i-1} has been subtracted. The rigorous resolution of the problem 3.23 requires specific mathematical tools. For this reason, the finite-dimensional case is introduced in following parts, easing the resolution of eigenvalue problems by employing classical tools from the linear algebra. Nevertheless, before introducing the finite-dimensional case, several important properties of the POD are emphasized.

3.2.2 Resolution for the finite-dimensional case

In practice, the output of the quantity of interest u is only known for a finite number of values which build the snapshots matrices defined in the section 3.1.1. The passage from the continuous case to the finite-dimensional helps to address the practical computation of the POD. For that purpose, the average operator $\langle\cdot\rangle$ and the inner product need to be defined in finite dimension. Two alternatives are generally presented for the choice of these operators, influencing directly the size of the eigenvalue problem and yielding to two methods: the direct method and the snapshot method.

3.2.2.1 Classical method

The integration domain $\mathcal{D}$ of the inner product is substituted by Ω and X by the mesh coordinate $\boldsymbol{x}$. Moreover, the statistical average is selected as a temporal average defined by:

$$\langle\cdot\rangle = \int_{\mathcal{T}} .\mathrm{d}t. \tag{3.25}$$

Thus, the Fredholm equation 3.22 can be written as:

$$\int_\Omega \frac{1}{T}\int_{\mathcal{T}} z(\boldsymbol{x},t)z(\boldsymbol{x'},t)\phi(\boldsymbol{x'})dtd\boldsymbol{x'} = \lambda\phi(\boldsymbol{x}). \tag{3.26}$$

The time and spatial integrals can be substituted by their discretizations such as:

$$\int_{\mathcal{T}} f(t)g(t)dt \approx \sum_{i=1}^{N_t} f(t_i)g(t_i), \tag{3.27}$$

$$\int_\Omega f(x)g(x)dx \approx \sum_{j=1}^{d} V_j f(x_j)g(x_j), \tag{3.28}$$

where V_j is the volume associated with the i-th cell of the mesh with d degrees of freedom. The discretization modifies an expression of the Fredholm equation for the finite-dimensional case:

$$\frac{1}{N_t}\sum_{i=1}^{N_t} z(\boldsymbol{x},t_i)\sum_{j=1}^{d} V_j z(x_j,t_i)\phi(x_j) = \lambda\phi(\boldsymbol{x}),\ \forall \boldsymbol{x}\in\Omega. \tag{3.29}$$

The matrix notation is introduced and the following eivenvalue problem is obtained:

$$\hat{\boldsymbol{Z}}\hat{\boldsymbol{Z}}^T\boldsymbol{\phi} = \lambda\boldsymbol{\phi}, \tag{3.30}$$

where $\hat{\boldsymbol{Z}} = (\boldsymbol{M}^{1/2})^T\boldsymbol{Z}$ and $\boldsymbol{M}$ is a diagonal matrix such as $M_{ii} = \frac{V_i}{N_t}$. In addition, the eigenproblem can be generalized for all the eigenvalues and not only the highest:

$$\hat{\boldsymbol{Z}}\hat{\boldsymbol{Z}}^T = \boldsymbol{\Phi}^T\boldsymbol{\lambda}\boldsymbol{\Phi}, \tag{3.31}$$

where

$$\boldsymbol{\Phi} = (\boldsymbol{\phi}^{(1)},\cdots,\boldsymbol{\phi}^{(n)})^T. \tag{3.32}$$

The size of the eigenvalue problem 3.31 is equal to d. In a typical industrial case, the discretisation of the spatial scheme involves millions of degrees of freedom. Therefore, a direct calculation results in a significant computational cost. Another method, the snapshot method, is presented in the following section. This approach proposes to solve an eigenvalue problem of size N_t. Therefore, if the time discretization is much smaller than the spatial discretization, an important amount of calculation time can be saved.

3.2.2.2 Snapshot method

The snapshot method has been introduced by Sirovich [180]. The central idea is to expand the spatial basis function ϕ as a linear combination of the snapshots:

$$\phi(x) = \sum_{k=1}^{N_t} a(t_k)z(\boldsymbol{x},t_k). \tag{3.33}$$

The Fredholm equation 3.22 can be rewritten as:

$$\int_\Omega R(\boldsymbol{x},\boldsymbol{x'})\phi(\boldsymbol{x'})d\boldsymbol{x'} = \lambda\phi(\boldsymbol{x}), \tag{3.34}$$

where the spatial correlation tensor R is expressed as follow with an ergodicity assumption:

$$\begin{aligned} R(\boldsymbol{x},\boldsymbol{x'}) &= \frac{1}{T}\int_T z(\boldsymbol{x},t)z(\boldsymbol{x'},t)dt \\ &= \frac{1}{N_t}\sum_{k=1}^{N_t} z(\boldsymbol{x},t_k)z(\boldsymbol{x'},t_k). \end{aligned} \tag{3.35}$$

Equation 3.33 and 3.35 are substituted in the Equation 3.34:

$$\frac{1}{N_t}\sum_{k=1}^{N_t}\sum_{l=1}^{N_t}\left(\int_\Omega z(\boldsymbol{x'},t_k)z(\boldsymbol{x'},t_l)d\boldsymbol{x'}\right)a(t_k)z(\boldsymbol{x},t_k) = \lambda\sum_{k=1}^{N_t} a(t_k)z(\boldsymbol{x},t_k). \tag{3.36}$$

As expressed in [44], Equation 3.36 provides a sufficient condition such as $a(t_k)$ is solution of the Fredholm equation:

$$\sum_{l=1}^{N_t}\left(\int_\Omega z(\boldsymbol{x'},t_k)z(\boldsymbol{x'},t_l)d\boldsymbol{x'}\right)a(t_k)z(\boldsymbol{x},t_k) = \lambda a(t_l),\ \forall l\in[1,\cdots,N_t]. \tag{3.37}$$

This equation can be interpreted as the symmetric of the classical POD [44], where the integration domain $\mathcal{D}$ of the inner product is substituted by Γ and X by the time t. Moreover, the statistical average is selected as a spatial average defined by:

$$\langle \cdot \rangle = \int_{\Omega} \cdot \, \mathrm{d}\boldsymbol{x}. \tag{3.38}$$

The final eigenvalue problem in finite dimension is allowed by the discretization of the integral and is written as:

$$\hat{\boldsymbol{Z}}^T \hat{\boldsymbol{Z}} \boldsymbol{A} = \lambda \boldsymbol{A}, \tag{3.39}$$

where $\boldsymbol{A} = [a(t_1) \ \cdots \ a(t_{N_t})]^T$ is the matrix of the reduced coordinates. Finally, the problem can be written in matrix form for all eigenvalues:

$$\hat{\boldsymbol{Z}}^T \hat{\boldsymbol{Z}} = \boldsymbol{A}^T \boldsymbol{\lambda} \boldsymbol{A}. \tag{3.40}$$

The last transformation ends up with an eigenvalue problem of size N_s, much more efficient in term of computational cost if $N_s << d$.

3.2.3 Singular value decomposition

The two eigenvalues problem given by Equation 3.31 and Equation 3.40 can be solved by a classical eigen decomposition. However, a very classical alternative method, called the singular values decomposition (SVD), is introduced in this section. A matrix $\boldsymbol{B} \in \mathbb{R}^{m \times n}$ can be uniquely decomposed as:

$$\boldsymbol{B} = \boldsymbol{U} \boldsymbol{\Sigma} \boldsymbol{V}^T, \tag{3.41}$$

where $\boldsymbol{U}$ and $\boldsymbol{V}$ are respectively $m \times m$ and $n \times n$ unitary matrix and $\boldsymbol{\Sigma}$ is a $m \times n$ diagonal matrix defined by:

$$\Sigma_{ii} = \sigma_i, \ \forall i \in [1, \cdots, \min(m, n)]. \tag{3.42}$$

The σ_i are called singular values. They are sorted in a descending order and are non-negative.

The SVD is closely related to the eigendecomposition of a matrix. The SVD is applied to $\hat{\boldsymbol{Z}}$. A straight derivation shows that the columns of $\boldsymbol{U}$ are the eigenvectors of $\hat{\boldsymbol{Z}} \hat{\boldsymbol{Z}}^T$:

$$\hat{\boldsymbol{Z}} \hat{\boldsymbol{Z}}^T = \boldsymbol{U} \boldsymbol{\Sigma} \boldsymbol{V}^T (\boldsymbol{U} \boldsymbol{\Sigma} \boldsymbol{V}^T)^T = \boldsymbol{U} \boldsymbol{\Sigma} \boldsymbol{V}^T \boldsymbol{V} \boldsymbol{\Sigma}^T \boldsymbol{U}^T = \boldsymbol{U} \boldsymbol{\Sigma} \boldsymbol{\Sigma}^T \boldsymbol{U}^T. \tag{3.43}$$

Similarly, the columns of $\boldsymbol{V}$ are eigenvectors of $\hat{\boldsymbol{Z}}^T \hat{\boldsymbol{Z}}$:

$$\hat{\boldsymbol{Z}}^T \hat{\boldsymbol{Z}} = \boldsymbol{V} \boldsymbol{\Sigma}^T \boldsymbol{U}^T \boldsymbol{U} \boldsymbol{\Sigma} \boldsymbol{V}^T = \boldsymbol{V} \boldsymbol{\Sigma}^T \boldsymbol{\Sigma} \boldsymbol{V}^T. \tag{3.44}$$

Equation 3.43 and Equation 3.44 can be identified respectively with the eigenvalue Equation 3.31 and Equation 3.40 as the matrices are real symmetric matrices. It gives for the classical method:

$$\begin{aligned} \boldsymbol{U} &= \boldsymbol{\phi}^T \\ \boldsymbol{\Sigma} \boldsymbol{\Sigma}^T &= \boldsymbol{\lambda}, \end{aligned} \tag{3.45}$$

and for the snapshot method:

$$\begin{aligned} \boldsymbol{V} &= \boldsymbol{A}^T \\ \boldsymbol{\Sigma}^T \boldsymbol{\Sigma} &= \boldsymbol{\lambda}. \end{aligned} \tag{3.46}$$

It can be noted the the eigenvalues for both $\hat{\boldsymbol{Z}} \hat{\boldsymbol{Z}}^T$ and $\hat{\boldsymbol{Z}}^T \hat{\boldsymbol{Z}}$ are equal to the square root of the singular values. Thus, a link between eigenvalue problems and SVD is established. Nevertheless, a closer relationship exists between SVD and POD as shown in the following section.

3.2.3.1 Link with the POD

Let introduce the inner product $(\cdot,\cdot)_{\boldsymbol{M}}$ in the finite dimensional case defined by:

$$(\boldsymbol{A},\boldsymbol{B})_{\boldsymbol{M}} = \boldsymbol{A}^T\boldsymbol{M}\boldsymbol{B}. \tag{3.47}$$

The associated norm is given by $\|A\|_{\boldsymbol{M}} = (\boldsymbol{A},\boldsymbol{A})_{\boldsymbol{M}} = \left\|(\boldsymbol{M}^T)^{1/2}\boldsymbol{A}\right\|_2$. The approximation problem 3.14 can be written in a finite dimension case [67]:

$$\begin{gathered}\min_{\phi_1,\cdots,\,\phi_K} \sum_{i=1}^{N_t}\left\|\boldsymbol{z}(t_i) - \sum_{k=1}^{K}(z_i,\boldsymbol{\phi_k})_{\boldsymbol{M}}\boldsymbol{\phi_k}\right\|_{\boldsymbol{M}}^2, \\ \text{subject to } (\phi_k,\phi_k)_{\boldsymbol{M}} = \delta_{i,j}.\end{gathered} \tag{3.48}$$

It is possible to reformulate 3.48 as a problem of the best representation of the matrix $\hat{\boldsymbol{Z}}$ in the sense of the Frobenius norm, defined by $\|\boldsymbol{U}\|_F^2 = \sum_{k=1}^{N}\|\boldsymbol{U_k}\|_2^2$:

$$\begin{gathered}\min_{\hat{\Phi}} \left\|\hat{\boldsymbol{Z}} - \hat{\Phi}\hat{\Phi}^T\hat{\boldsymbol{Z}}\right\|_F^2, \\ \text{subject to } \hat{\boldsymbol{\Phi}}^T\boldsymbol{\Phi} = \mathcal{I},\end{gathered} \tag{3.49}$$

where $\hat{\Phi} = (\boldsymbol{M}^{1/2})^T\Phi$. The solution of the problem 3.49 is given by the Eckart-Young theorem [58] which states that for any matrix $\boldsymbol{B} \in \mathbb{R}^{?\times?}$, the solution of the minimization problem:

$$\min_{rank(X)\leq k} \|\boldsymbol{B}-\boldsymbol{X}\|_F^2 = \|\boldsymbol{B}-\boldsymbol{B_k}\|_F \tag{3.50}$$

is given by $\boldsymbol{B_k}$, the truncated singular value decomposition of $\boldsymbol{B}$ defined by $\boldsymbol{B_k} = \boldsymbol{U_k}\boldsymbol{\Sigma_k}\boldsymbol{V_k}^T$. The theorem is applied to $\hat{\boldsymbol{Z}}$. The comparison of $\boldsymbol{U_k}\boldsymbol{\Sigma_k}\boldsymbol{V_k}^T$ and $\hat{\Phi}\hat{\Phi}^T\hat{\boldsymbol{Z}}$ provides a linear system computing the POD basis Φ:

$$(\boldsymbol{M}^{1/2})^T\Phi = \boldsymbol{U_K}. \tag{3.51}$$

Therefore, computing the truncated SVD of $\hat{\boldsymbol{Z}}$ gives a direct access to the values of the POD basis. A method to estimate the rank K given to $\hat{\boldsymbol{Z}}$ is discussed in the section 3.2.4.

3.2.3.2 Iterative SVD

In practice, all the snapshots are not necessarily available at the same time. Therefore, an incremental version of the POD basis computation has been developed [21], based on an iterative SVD [149]. The snapshot matrix $\boldsymbol{S}_{:k}$ is composed of the first k snapshots, such as $\boldsymbol{S}_{:k} = [S_1,\cdots,S_k]^T$. The associated SVD is given by $\boldsymbol{S}_{:k} = \boldsymbol{U_{:k}}\Sigma\boldsymbol{V_{:k}}^T$. The $k+1$-th snapshot is assumed to be available and is added to $\boldsymbol{S}_{:k}$ to build $\boldsymbol{S_{:k+1}} = [\boldsymbol{S_{:k}},\ \boldsymbol{s_{k+1}}]^T$. This iterative process is very useful for the error estimation step, based on cross-validation described in the section 3.5.3.

3.2.3.3 Computational aspects

Eigen-decomposition and singular value decomposition have their own advantages and drawbacks summarized in Table 3.1. In practice, SVD is used due to its sequential capability and better precision for smallest eigenvalues.

POD computation method	Characteristics
Eigen Decomposition	+ Fast computations of the eigen-vectors - Low precision on the calculation of the smallest eigen-values - No sequential computation (upgrading or downgrading) of the POD basis
Singular Value Decomposition	+ Better precision on the calculation of the smallest eigen-values + Capable of sequential computation (upgrading or downgrading) - Slower computation of the singular vectors

Table 3.1: Eigen Decomposition and Singular Value Decomposition characteristics [21]

.

3.2.4 Dimension reduction

One of the POD interests lies in approximating a vector-valued function with a limited number of basis functions while keeping a low approximation error, due to the fact that the dimensional space of the snapshots is usually sparse in a L_2 sense. This truncation error between the true function z and its expansion into the basis function at the M-th index is directly given by approximation Equation 3.14. This latter can be analytically computed, since Equation 3.10 is exact as K approaches the infinity:

$$\left\langle \left\| z(\boldsymbol{x},t) - \sum_{k=1}^{M} a_k(t)\phi_k(\boldsymbol{x}) \right\|_2 \right\rangle = \sum_{k=M+1}^{K} \lambda_k. \tag{3.52}$$

However, this expression is seldom used in practice. The main heuristic criterion is to consider the ratio of captured energy, which means finding a scalar M such as for a given amount of energy ratio ϵ:

$$\frac{\sum_{k=1}^{M} \lambda_k}{\sum_{k=1}^{K} \lambda_k} > \epsilon. \tag{3.53}$$

The term "ratio of captured energy" comes from the fact that the sum of eigenvalues is related to the sum of the squared mean of the reduced coordinates [44]. In the context of flow analysis, the POD is usually applied to velocity components, therefore the sum of the eigenvalues $\sum_{k=1}^{M} \lambda_k$ is interpreted as the mean energy captured by the first M POD modes. A classic ratio of energy found in literature is 0.99 [43, 180].

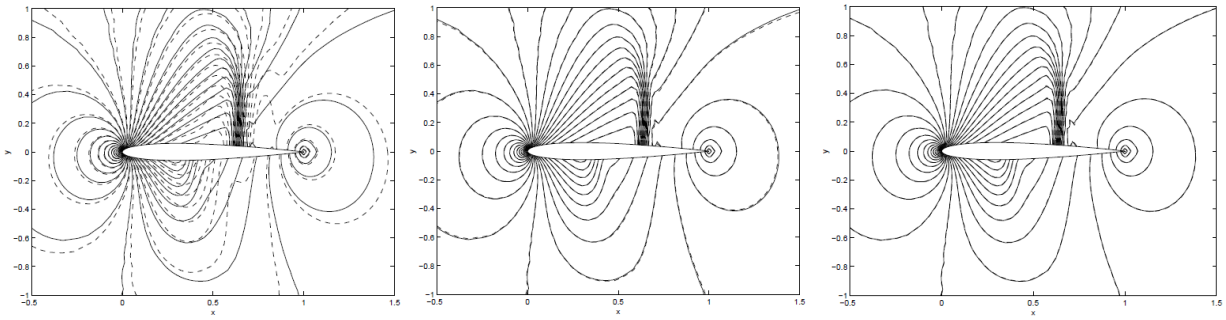
Cordier and Bergmann [43] have shown that a Computational Fluid Dynamics (CFD) simulation with a compression ratio of 90% is obtained by taking POD modes containing 99% of turbulent energy. In addition, it is noticeable that $\sum_{k=1}^{M} \lambda_k$ is easily computed with the matrix trace tr:

$$\sum_{k=1}^{M} \lambda_k = tr\left(\hat{\boldsymbol{Z}}^T \hat{\boldsymbol{Z}}\right). \tag{3.54}$$

An example of POD application in the field of aerodynamics is given by Bui-Thanh et al. [23]. Inviscid steady-state CFD is performed on a NACA 0012 airfoil for Mach numbers from 0.75 to

0.85 divided into 20 uniform intervals and angle of attack from 0° to 1.25° and divided into 10 uniform intervals) for a total of 231 simulations.

Figure 3.3 shows a pressure contour prediction for a Mach number of 0.812 and a angle of attack of 1.1°. The first five POD modes contain already a rough estimate of the pressure field while the first 35 modes (on the 231 modes available) represent very precisely the CFD simulation with a good dimension reduction.



(a) Five modes. (b) Thirty modes. (c) Thirty five modes.

Figure 3.3: Pressure field for Mach number = 0.812 and an angle of attack of 1.1 with different number of modes retained. The dotted line corresponds to the POD reconstruction and solid line to the CFD. Credit from Bui-Thanh et al. [23].

3.3 Determining the reduced coordinates at untried parameters

The previous sections have detailed theoretical and practical aspects of the POD. In particular, the methods to compute the basis vectors and reduced coordinates have been explained. However, the reduced-order model remains to be built in order to generate the prediction points $\boldsymbol{X_p}$. As explained in the introduction, the central idea of ROMs using POD is to represent the function $\boldsymbol{f}$ only with its projection onto the POD basis. By assuming M POD basis vectors are invariant with respect to the input parameters, the surrogate $\widetilde{f}$ is fully defined by approximating the reduced coordinates. There exist several methods to approximate these values.

The reduced coordinates a_k have been computed at a small number of training parameters whereas the analysis of the deterministic high-fidelity model for various input conditions requires a continuous evaluation over the input parameter space. Intrusive approaches have been quickly explained in the introduction and they fall out of the scope of this work. Therefore, the data-fit methods are well-suited for this kind of problem as they are non-intrusive. The final function $\boldsymbol{f}$ is fully approximated by building M data-fit models $\widetilde{a}$, one for each reduced coordinate, such as:

$$\boldsymbol{f}(\boldsymbol{\chi}) \simeq \overline{\boldsymbol{f}} + \sum_{k=1}^{M} \widetilde{a_k}(\boldsymbol{\chi})\ \boldsymbol{\phi}_k,\ \forall \boldsymbol{\chi} \in \mathbb{R}^p. \tag{3.55}$$

The approximations can be regression-based or interpolation-based. By following the same formalism than for the function $\boldsymbol{f}$ of the quantity of interest, the training reduced coordinate vector $\boldsymbol{a_k}$ is defined as:

$$\boldsymbol{a_k} = [a_k(\boldsymbol{\chi}_{t_1}), \cdots, a_k(\boldsymbol{\chi}_{t_n})]^T \in \mathbb{R}^n. \tag{3.56}$$

In this section, we discuss different data-fit methods used to generate non-intrusive ROMs. In this context, data-fit approaches aims at building an estimator $\widetilde{a_k}$ of reduced coordinates a_k,

which can be seen as a black box function. The most popular methods in the surrogate modeling literature are formed of polynomial regressions, Radial Basis Function, Support Vector Regression, and Gaussian Process Regression (GPR) [73]. A particular emphasis is given to this latter in a specific section.

3.3.1 Polynomial regression

The polynomial regression is usually considered to be the simplest method. The mathematical formalism of the one dimensional polynomial regression is presented. Let consider only in this part that each training sample is one dimensional, thus $p = 1$ and $\boldsymbol{X_t} \in \mathbb{R}^{n\times 1}$. The generalization for $p > 1$ can be founded in Fang et al. [69] for instance.

The classical one dimensional polynomial regression $\widetilde{a_k}$ of order m is written as follows:

$$\widetilde{a_k}(\boldsymbol{\chi}, \boldsymbol{w}) = w_0 + w_1.\boldsymbol{\chi} + w_2\boldsymbol{\chi}^2 + \ldots + w_m\boldsymbol{\chi}^m = \sum_{i=0}^{m} w_i.\boldsymbol{\chi}^i,\ \forall \boldsymbol{\chi} \in \mathbb{R}. \tag{3.57}$$

The vector of weights $\boldsymbol{w} = [w_0, w_1, \cdots, w_m]$ remains to be determined. The weights are computed by fitting the polynomial to the training data. The principle of least squares consist in minimizing an error function J, also called cost function, which is introduced as:

$$J(\boldsymbol{w}) = \frac{1}{2n}\sum_{i=1}^{n}[a_k(\boldsymbol{\chi}_{t_i}, \boldsymbol{w}) - \widetilde{a_k}(\boldsymbol{\chi}_{t_i}, \boldsymbol{w})]^2 = \|\boldsymbol{A} - \boldsymbol{w}\boldsymbol{V}\|, \tag{3.58}$$

where $\boldsymbol{V}$ is the Vandermonde matrix defined such as [73]:

$$\boldsymbol{V} = \begin{bmatrix} 1 & \boldsymbol{\chi}_{t_1} & \boldsymbol{\chi}_{t_1}^2 & \cdots & \boldsymbol{\chi}_{t_1}^m \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \boldsymbol{\chi}_{t_n} & \boldsymbol{\chi}_{t_n}^2 & \cdots & \boldsymbol{\chi}_{t_n}^m \end{bmatrix} \in \mathbb{R}^{n\times m}. \tag{3.59}$$

The expression of the cost function J represents the sum of squares of the errors between the predictions $\widetilde{a_k}$ for each training point $\boldsymbol{\chi_t}$ and the associated true values a_k. The value of $\boldsymbol{w^*}$ which minimizes the cost function is called the ordinary least squares estimator for $\boldsymbol{w}$ and solves the following problem:

$$\boldsymbol{w^*} = \min_{\boldsymbol{w}} J(\boldsymbol{w}). \tag{3.60}$$

The least squares problem can be solved analytically by setting the gradient of J to zero:

$$\nabla J(\boldsymbol{w^*}) = 0 \Leftrightarrow \boldsymbol{V}^T\boldsymbol{V}\boldsymbol{w} = \boldsymbol{V}^T\boldsymbol{A}. \tag{3.61}$$

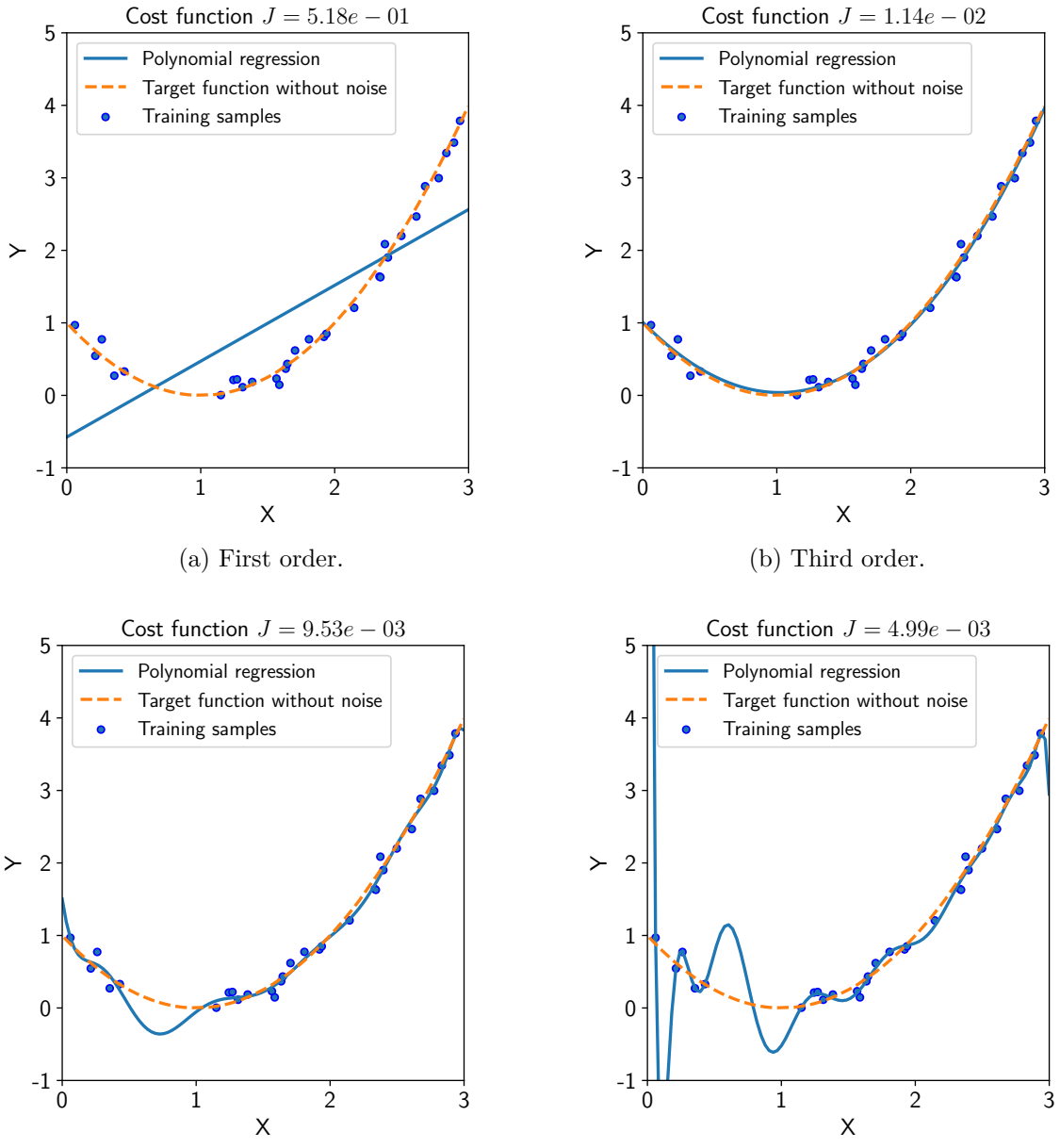
The inversion of the matrix $\boldsymbol{V}^T\boldsymbol{V}$ yields to:

$$\boldsymbol{w^*} = (\boldsymbol{V}^T\boldsymbol{V})^{-1}\boldsymbol{V}^T\boldsymbol{A}. \tag{3.62}$$

If the matrix inversion of $\boldsymbol{V}^T\boldsymbol{V}$ appears computationally very intensive, the minimization problem in Equation 3.60 can be solved numerically, for example with gradient descent algorithms.

However, the key parameter m, the order of the regression, is unknown and remains undetermined. It is usually fixed by the user. For $m = 1, 2$ and 3, linear, quadratic and cubic regressions are respectively obtained. These low-order polynomials are also referred to as Response Surface Methods [153] if they are built in some relatively small regions of the parameter space. One would think that more complex is the model (and more m is higher) smaller is the model error. Nevertheless,

from a given threshold level of m, increasing the order increases the error. This phenomenon, also called overfitting [124], explains that the relation between model complexity and model error is not strictly proportional. Figure 3.4 gives an example where the cost function J continuously decreases with the polynomial order m but for the high orders the polynomial regression differs strongly from the target solution. It can interpreted as an example of the bias-variance tradeoff [18]. For low polynomial orders, smooth and simple model are built with high bias and low variance while for high polynomial orders, complex models with high variance are generated. The latter are prone to overfitting as they are also learning the noise of the function. Therefore, a tradeoff in the complexity of the model must be found. In the example of Figure 3.4, a third order regression captures well the overall trend of the function even if small errors remain. The phenomenon of overfitting also exists for other regression methods.



(a) First order.

(b) Third order.

(c) Twelfth order.

(d) Fifteenth order.

Figure 3.4: Overfitting illustration

A classical solution overcoming overfitting consists of introducing a penalty term λ in the cost function J:

$$J(\boldsymbol{w}) = \frac{1}{2n}\sum_{i=1}^{n}[a_k(\boldsymbol{\chi}_{t_i}, \boldsymbol{w}) - \widetilde{a_k}(\boldsymbol{\chi}_{t_i}, \boldsymbol{w})]^2 + \lambda \left\|\boldsymbol{w}\right\|^2, \text{ with } \lambda > 0. \tag{3.63}$$

Such methods are called regularization or shrinkage. The particular case of polynomial regressions is called ridge regression and the use of penalization for neural networks is called weight decay [78, 18]. Choosing λ is not obvious but allows to control the overfitting. While specific values can reduce, or even suppress, overfitting, large values of λ lead to a very poor fit of the regression model as the term related to the training data becomes negligible compared to the penalization term.

Generalizing the multivariate case increases dramatically the number of polynomial basis of the expression. For example, a second-order polynomial model is expressed as follows for an input parameter $\boldsymbol{\chi}$ of size p:

$$\widetilde{a_k}(\boldsymbol{\chi}, \boldsymbol{w}) = w_0 + \sum_{i=1}^{p} w_i.\boldsymbol{\chi}_{t_i} + \sum_{i=1}^{p}\sum_{i=1}^{p} w_{ij}.\boldsymbol{\chi}_{t_i}\boldsymbol{\chi}_{t_j}, \ \forall \boldsymbol{\chi} \in \mathbb{R}^p, \tag{3.64}$$

where $\boldsymbol{w} = [w_0, \cdots, w_p, w_{1,1}, \cdots, w_{1p}, w_{2,1}, \cdots, w_{pp}]$. Increasing the order of the multivariate polynomial regression increases the complexity of the surrogate model and may lead to numerical issues, as several polynomial basis can be collinear. For this reason, it is preferred to use orthogonal polynomial models instead of the canonical basis, such as spline of wavelets to approximate periodic functions [69]. In addition, the data size required to fully define the linear system increases exponentially with m and the dimension of the size of the input variables p. For instance, fitting a problem of ten dimensions will require 11 samples for a linear regression and 286 samples for a cubic regression [3].

In conclusion, polynomial regressions have several major drawbacks: their incapacity of fitting high-dimensional problem and intrinsic features not suited for non-linear functions. Nevertheless, polynomial regressions can be used advantageously for noise filtering. Moreover, their easy interpretation, implementation, and mathematical manipulation (to compute derivatives for example) represent an important and popular advantage.

3.3.2 Support vector regression

Support Vector Machine (SVM) has firstly be introduced as a supervised learning algorithm for classification problems [18]. In 1997, a regression version called Support Vector Regression (SVR) was proposed by Drucker et al. [52]. Excellent performances were obtained by this method in regression and time series prediction [182]. SVR defines a set of basis functions centered on the training data points and allows a training error. One of the main property is that only a part of the points are selected and used during the training phase, leading to a sparse model.

3.3.2.1 Basic form and problem formulation

The general form of SVR is given by a sum of basis function ϕ weighted by w_i and added to a bias μ:

$$\widetilde{a_k}(\boldsymbol{\chi}) = \mu + \sum_{i=1}^{n} w_i.\phi(\boldsymbol{\chi}, \boldsymbol{\chi}_i). \tag{3.65}$$

For the sake of simplicity, mathematical expressions are derived for the linear case, where $\phi(\boldsymbol{\chi}) = \boldsymbol{\chi}$, and given in a matrix format:

$$\widetilde{a_k}(\boldsymbol{\chi}) = \mu + \boldsymbol{w}^T.\boldsymbol{\chi}, \text{ with } \boldsymbol{w} = [w_1, \cdots, w_n]^T. \tag{3.66}$$

Just as the polynomial regression, the weight vector $\boldsymbol{w}$ and bias μ are unknown and have to be determined. SVR is built as an hyperplan given by the equation $\mu + \boldsymbol{w}^T\boldsymbol{\chi}$ and fitted to the data with a given tolerance margin ϵ. SVR is sought to generate predictions with good generalization properties: an approximation with, at most, a deviation of ϵ from the training data and with the minimum complexity [73]. The latter expression can be understood as a weight vector with the minimum norm, just as the penalization term of the polynomial regression. Indeed, the formulation of the linear SVR can be seen as a penalized polynomial regression where the quadratic error is replaced by an error function, given zero if the prediction is in an interval of size 2ϵ and centered around the training data. These characteristics can be written in a mathematical form as a minimization problem of the vector norm of $\boldsymbol{w}$ constrained by the training error contained in an ϵ interval [182]:

$$\begin{aligned} &\min \frac{1}{2} \|\boldsymbol{w}\|_2 , \\ &\text{subject to} \begin{cases} a_k(\boldsymbol{\chi}_i) - \boldsymbol{w}^T\boldsymbol{\chi}_i - \mu \leq \epsilon \\ \boldsymbol{w}^T\boldsymbol{\chi}_i + \mu - a_k(\boldsymbol{\chi}_i) \leq \epsilon \end{cases} . \end{aligned} \tag{3.67}$$

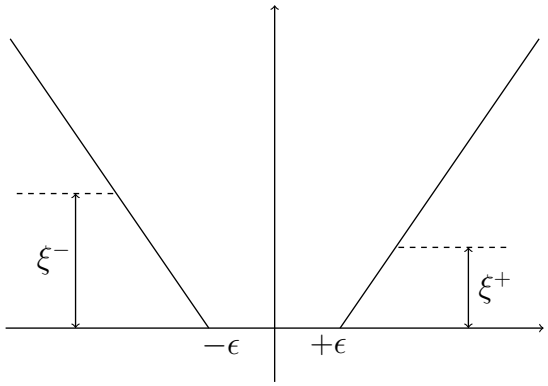
However the existence of the solution is not guaranteed for the problem 3.67. For instance, if ϵ is too small there is no solution. For this reason, a new optimization problem is created by introducing slack variables ξ^+ and ξ^- allowing violation of some constraints, as illustrated in Figure 3.5a. A positive constant C is introduced to define a trade-off between the flatness of the model and the tolerance of model error greater than ϵ. It can be seen as a penalization term and gives the new optimization problem:

$$\begin{aligned} &\min \frac{1}{2} \|\boldsymbol{w}\|_2 + C\sum_{i=1}^{n}(\xi_i^+ + \xi_i^-), \\ &\text{subject to} \begin{cases} a_k(\boldsymbol{\chi}_i) - \boldsymbol{w}^T\boldsymbol{\chi}_i - \mu \leq \epsilon + \xi_i^+ \\ \boldsymbol{w}^T\boldsymbol{\chi}_i + \mu - a_k(\boldsymbol{\chi}_i) \leq \epsilon + \xi_i^- \\ \xi_i^+, \xi_i^- \geq 0 \end{cases} . \end{aligned} \tag{3.68}$$

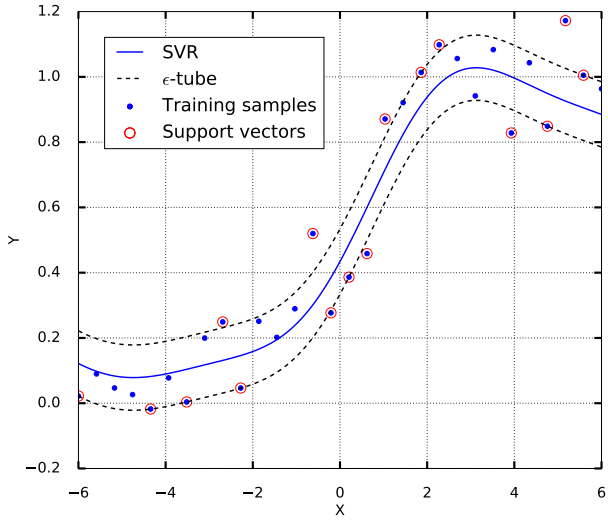
3.3.2.2 Dual problem

The corresponding Lagrangian L of Equation 3.68 is given by:

$$\begin{aligned} L =& \frac{1}{2} \|\boldsymbol{w}\|_2 + C\sum_{i=1}^{n}(\xi_i^+ + \xi_i^-) - C\sum_{i=1}^{n}(\eta_i^+\xi_i^+ + \eta_i^-\xi_i^-) \\ &- \sum_{i=1}^{n}\alpha_i^+(\epsilon + \xi_i^+ - a_i + \boldsymbol{w}^T\boldsymbol{\chi}_i + \mu) \\ &- \sum_{i=1}^{n}\alpha_i^-(\epsilon + \xi_i^- + a_i - \boldsymbol{w}^T\boldsymbol{\chi}_i - \mu), \end{aligned} \tag{3.69}$$



(a) Constraint function and slack variables.



(b) Support vectors and ϵ-tube.

Figure 3.5: Illustration of the SVR.

where α^+, α^-, η^+ and η^- are Lagrange multipliers. By differentiating with respect to w, μ, η^+ and η^-, the resolution of the saddle point of L leads to the dual problem formulation:

$$\max \; -\epsilon \sum_{i=1}^{n}(\alpha_i^+ + \alpha_i^-) + \sum_{i=1}^{n} y_i(\alpha_i^+ - \alpha_i^-) - \frac{1}{2}\sum_{i,j=1}^{n}(\alpha_i^+ - \alpha_i^-)(\alpha_j^+ - \alpha_j^-)\boldsymbol{\chi}_i^T\boldsymbol{\chi}_j$$
$$\text{subject to} \begin{cases} \sum_{i=1}^{n} a_i(\alpha_i^+ - \alpha_i^-) = 0 \\ 0 \leq \alpha_i^+ \leq C \\ 0 \leq \alpha_i^- \leq C \end{cases} . \tag{3.70}$$

In addition, the final expression of the SVR is given by:

$$\widetilde{\boldsymbol{a}}(\boldsymbol{\chi}) = \mu + \sum_{i=1}^{n}(\alpha_i^+ - \alpha_i^-).\boldsymbol{\chi}^T\boldsymbol{\chi}_i \tag{3.71}$$

The dual problem is solved by quadratic programming algorithms. Describing such algorithms is out of the scope of this chapter. Details on quadratic programming algorithms applied to SVM can be found in [182, 18].

Important comments can be made on dual problem in Equation 3.70 and on the final expression of the SVR given in Equation 3.71. First, one can note that SVR is defined only by a linear combination of training data $\boldsymbol{\chi}_i$ independently of the dimension of the input space. Then, applying Karush-Kush-Tucker (KKT) conditions gives extra information to ease understanding of the model. They state that at the solution of the dual problem, the product of Lagrange multiplier and constraint is equal to zero, such as:

$$\begin{aligned}(C - \alpha_i^+)\xi^+ &= 0 \\ (C - \alpha_i^-)\xi^- &= 0\end{aligned} \tag{3.72}$$

and

$$\begin{aligned}\alpha_i^+(\epsilon + \xi_i^+ - a_i + \boldsymbol{w}^T\boldsymbol{\chi}_i + \mu) &= 0 \\ \alpha_i^-(\epsilon + \xi_i^- + a_i - \boldsymbol{w}^T\boldsymbol{\chi}_i - \mu) &= 0.\end{aligned} \tag{3.73}$$

Firstly, Equation 3.72 shows that if the sample point is outside the ϵ-tube (which mean $\xi > 0$), then $\alpha = C$. Then if the sample is within the ϵ-tube, both ξ^+ and ξ^- are equal to zero, leading to $\alpha_+ = \alpha_- = C$. Introducing this result in Equation 3.71, it appears that the samples inside the ϵ-tube vanish of the prediction formulation and only the samples on or outside of the ϵ-tube remain. Thus SVR is only and fully defined by these samples, which are called support vectors, as illustrated in Figure 3.5b. This concept is central to the practical applicability of SVR, as the complexity of the model depends only on the number of support vectors. Indeed, once the model is trained, a significant proportion of the training data can be discarded. Only the support vectors must be retained to generate the predictions [18].

The expression of the coefficient μ is given by Equation 3.73. If the sample $\boldsymbol{\chi}_i$ lies on or outside the *epsilon*-tube, there are two possibilities: the sample is above the tube or below the tube, leading respectively to [73]:

$$\begin{aligned} &\alpha_i^+ = 0 \\ &\mu = a_i + \sum_{i=1}^{n} (\alpha_i^+ - \alpha_i^-) \boldsymbol{\chi}_i^T \boldsymbol{\chi}_p + \epsilon \end{aligned} \tag{3.74}$$

or,

$$\begin{aligned} &\alpha_i^- = 0 \\ &\mu = a_i + \sum_{i=1}^{n} (\alpha_i^+ - \alpha_i^-) \boldsymbol{\chi}_i^T \boldsymbol{\chi}_p - \epsilon \end{aligned} \tag{3.75}$$

Finally, the constants C and ϵ have to be chosen as they influence directly the shape of the model. A small constant C will lead to a flat prediction due to a greater importance attached to the minimization of w, while a large constant will lead to smaller slack variables and thus a model closer to the data. Figure 3.6 illustrates the behavior of the SVR with a non-linear kernel for different C. As regards the error ϵ, it quantifies the noise in the data and must be chosen with prior knowledge of the data or alternatively in a conservative manner in order to avoid large training errors. If the noise in the data is fully unknown, the error parameter ϵ may be substituted by the fraction of points lying outside the tube [73]. Methods that automatically adjust ϵ also exist [182].

3.3.2.3 Non linear feature and kernel trick

For the moment, only the linear case has been explained but small modifications can transform the formulation to be made non-linear. The main feature to perform this modification is that the data are used only through inner products. In order to capture more complicated landscapes, Equation 3.71 is extended by applying non-linear transformation ϕ to the input. Let consider $\boldsymbol{\chi}$ can be mapped from $\mathbb{R}^p$ to a feature space $\mathcal{H}$ by an application $\phi\colon \mathbb{R}^p \to \mathcal{H}$. Usually the dimension of $\mathcal{H}$ is larger than dimension of $\mathbb{R}^p$. However the inner product in high dimension leads to expensive computation cost. To tackle this issue, inner product is replaced by a function k called kernel thanks to Mercer Theorem, such as $k(\boldsymbol{\chi}, \boldsymbol{\chi}') = \phi(\boldsymbol{\chi})^T \phi(\boldsymbol{\chi}')$. This function k must satisfy several properties, in particular: being continuous, symmetric, and positive definite [73].

All equations from the linear case can be rewritten by mapping $\boldsymbol{\chi}$ to the feature $\mathcal{H}$. This is done simply by replacing the inner product $\phi(\boldsymbol{\chi}_i)^T \phi(\boldsymbol{\chi})$ by $k(\boldsymbol{\chi}_i, \boldsymbol{\chi})$. For example Equation 3.71 becomes:

$$\widetilde{\boldsymbol{f}}(\boldsymbol{\chi}) = \mu + \sum_{i=1}^{n} (\alpha_i^+ - \alpha_i^-) \phi(\boldsymbol{\chi}_i)^T \phi(\boldsymbol{\chi}) = \mu + \sum_{i=1}^{n} (\alpha_i^+ - \alpha_i^-)\, k(\boldsymbol{\chi}_i, \boldsymbol{\chi}). \tag{3.76}$$

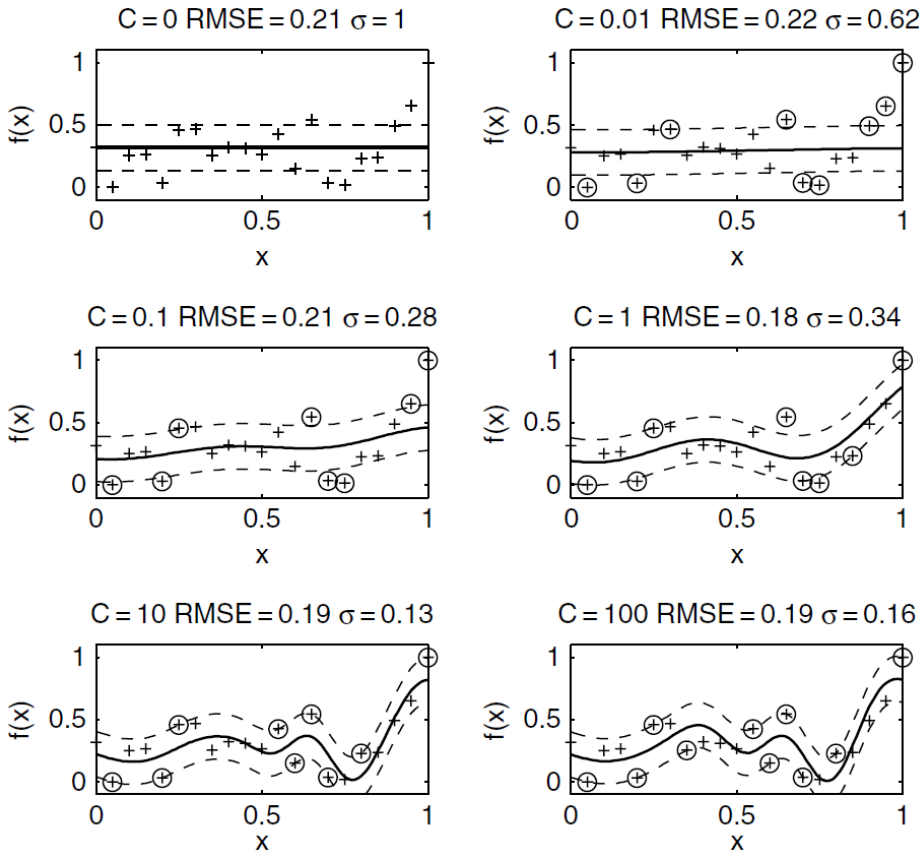


Figure 3.6: SVR predictions and corresponding RMSE for varying C. The support vectors are marked with circles and the dashed line corresponds to the ϵ tube. Credits from Forrester et al. [73].

The 'kernel trick' refers to the mapping of parameters to feature spaces of higher dimension in order to ease the computation on high dimensional data, as illustrated in Figure 3.7 for the context of classification. The following kernel functions are some examples which can be found in the literature:

$$\begin{aligned} &\text{Polynomial: } k(\boldsymbol{\chi}, \boldsymbol{\chi}') = (b + < \boldsymbol{\chi}, \boldsymbol{\chi}' >)^d \text{ with } b \in \mathbb{R} \text{ and } d \in \mathbb{R}^+ \\ &\text{Sigmoid: } k(\boldsymbol{\chi}, \boldsymbol{\chi}') = tanh(\gamma < \boldsymbol{\chi}, \boldsymbol{\chi}' > + b) \text{ with } b, \gamma \in \mathbb{R} \\ &\text{Radial Basis Function: } k(\boldsymbol{\chi}, \boldsymbol{\chi}') = exp(\tfrac{-|\boldsymbol{\chi} - \boldsymbol{\chi}'|^2}{\sigma^2}) \text{ with } \sigma \in \mathbb{R} \end{aligned} \tag{3.77}$$

3.3.2.4 Limitations

SVR shows mainly two limitations in the context of surrogate models: the absence of error estimation and the need to estimate a trade-off parameter. To address these issues, other methods have been developed, in particular the Relevant Vector Machine. This methods fixes a set of basis function in order to generate a sparser model and introduces a Bayesian formalism. Thus, the set of free parameters of the SVM is avoided and a probabilistic estimation of the error is given. However, this latter can take very small values for extrapolating regions outside of the training domain [18]. In addition, Relevant Vector Machine involvces the optimization of a nonconvex function while SVR requires only to solve a convex problem. Thus, the training times can be longer.

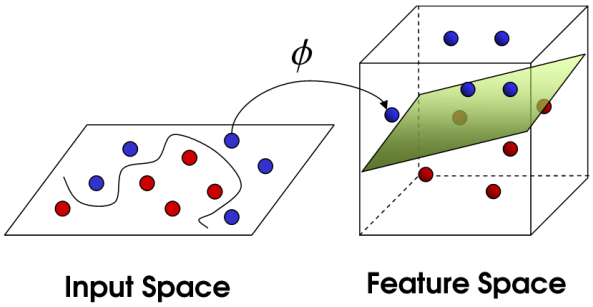


Figure 3.7: Kernel Trick Illustration. Credit from Zararsiz et al. [209]

3.3.3 Artificial Neural Networks

Artificial Neural Networks (ANN) are designed to emulate the way the human brain would solve a problem and is generally considered as a black box model due to its complexity. Its definition is an oriented and valuated graph, made up of a set of basic artificial neurons. Recently, ANN have gained popularity in the literature due to the massive use of Deep Learning. This specific method demonstrates a clear superiority in numerous application in pattern recognition and machine learning, compared to classical methods. However, Deep Learning requires a tremendous volume of data, which is not the case in the context of this thesis with few and very computationally expensive samples. Nevertheless, classical ANN remains a popular method of regression mainly thanks to the universal approximation theorem [97], ensuring that a neural network with a single hidden layer can represent a wide variety of continuous functions under specific assumptions. In addition, the development of the backward propagation of errors [125] allows to efficiently computes all the parameter of the network.

3.3.3.1 Basic neuron

The central idea of ANN for regression is to build a linear combinations of input parameter, and then to apply a nonlinear function on these combination in order to model the quantity of interest. This part presents the formalism of a neural network with a single hidden layer applies only to regression problems and not to pattern recognition. A neuron k can be characterized by three basic elements represented in Figure 3.8:

- the **synapses** connect the neuron k and each connection j is associated with a weight ω_{kj}
- the **adder** sums the input signals x_j with the respective weight of the synapses ω_{kj}
- the **activation function** φ_k transforms the combination of input signals added to the bias b_k, giving the output signal y_k

The bias b_k can be seen as a weight such as: $w_{k0} = b_k$ and $\chi^{(0)} = 1$. By introducing the input x_k of the neuron k, the final output y_k is written as follows:

$$y_k = \varphi\left(\sum_{j=0}^{p} \omega_{kj}\chi^{(j)}\right) = \varphi(x_k) \text{ with } x_k = \sum_{j=0}^{p} \omega_{kj}\chi^{(j)}. \tag{3.78}$$

The activation functions have different types and is usually chosen from the following expressions:

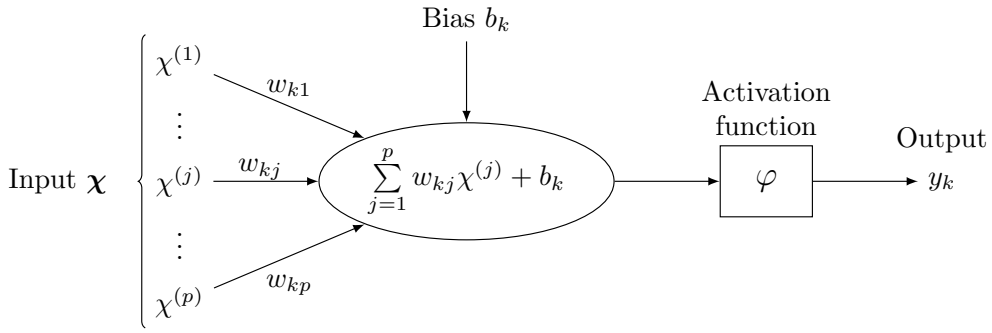


Figure 3.8: Basic Neuron.

- the identity function $\varphi(x) = x$
- the threshold function

$$\varphi(x) = \begin{cases} 1, & \text{if } x \geq \nu, \text{ with } \nu \in \mathbb{R}^+ \\ 0, & \text{otherwise} \end{cases} \tag{3.79}$$

- piecewise-linear function

$$\varphi(x) = \begin{cases} 1, & \text{if } x \geq +\frac{1}{2} \\ x, & \text{if } -\frac{1}{2} < x < +\frac{1}{2} \\ 0, & \text{otherwise} \end{cases} \tag{3.80}$$

- the sigmoid function

$$\varphi(x) = \frac{1}{1 + e^{-\nu x}}, \text{ with } \nu \in \mathbb{R}^+. \tag{3.81}$$

- the hyperbolic tangent

$$\varphi(x) = \frac{1 - e^{-2\nu}}{1 + e^{-2\nu}} \tag{3.82}$$

Sigmoid, linear function and hyperbolic tangent are particularly fitted for learning algorithms with back-propagation techniques (described in the follow part) as they are analytically differentiable. More recently, rectified linear functions (ReLUs), leaky rectified linear units (leaky ReLUs), and exponential linear units (ELUs) have become more common.

3.3.3.2 Multi-layer Neural Network

ANNs are characterized by their architecture: number of layers, type of connection (feedforward, feedback, etc...), and used activation functions. A layer is defined as a set of neurons which are not connected each other. The generalization from one layer to a feedforward neural network with multiple layers is obtained by connecting each layer in a sequential manner. In this case, the first layer takes as input a transformed linear combination of the input signal. Then the output of the first layer becomes one of the inputs to the nodes on the next layer and so on. This structure of the ANN involves three different kind of layers:

- the **input layer** performs an affine transformation of the input signals

- several **hidden layers** apply activation functions on the affine transformation of the inputs. The activation function is the same for each layer level.
- the **output layer** gives the outputs with a linear activation function

The layer index is written with a subscript, thus $w_{jk}^{(l)}$ corresponds to the weight connecting the j-th synapse with the k-th neuron in the l-th layer, $\varphi^{(l)}$ is the activation function of the k-th neuron in the l-th layer, $x_k^{(l)}$ is the input of the k-th neuron in the l-th layer and $y_k^{(l)}$ the associated output.

Neural network training with backpropagation A simple approach consist in solving a minimization problem for a sum-of-squares error function, just as for the polynomial regression and the SVR:

$$J(\boldsymbol{w}) = \frac{1}{2n}\sum_{l=1}^{n}[a_k(\boldsymbol{\chi}_{t_l}, \boldsymbol{w}) - \widetilde{a_k}(\boldsymbol{\chi}_{t_l}, \boldsymbol{w})]^2. \tag{3.83}$$

The gradient descent is a classical method to minimize the cost function J. In the context of neural networks, it is called backpropagation due to the structure in network which allows to use easily the chain rule for differentiation. It propagates the error between training and predicted data with a gradient computation from the output layer to the hidden and input layers. The algorithm is decomposed in two steps. The first pass, also called forward pass, corresponds to the evaluation of all the elements of the ANN and the computation of the output, but the weights remain unchanged. The second pass, also called backward pass, starts to the output layer and propagates recursively the error of the output layer through the hidden and input layers. During the process, the weights are updated to reduce this error following a gradient descent approach, such as if the algorithm is assumed to be at the s-th iterate, then the new values of the weight $w_{ij}^{(l)}$ is given by the gradient descent update at the $s+1$-th iterate:

$$w_{ij}^{(l)}\Big|_{r+1} = w_{ij}^{(l)}\Big|_{r} - \gamma\frac{\partial J}{\partial w_{ij}^{(l)}}, \tag{3.84}$$

where $\gamma > 0$ is known as the learning rate and must be fixed. Details for its parametrization can be found in Haykin [91]. Other methods than the gradient descent algorithm can be used in order to increase the convergence speed, to avoid local minimum and oscillation of the optimizer. One can cite for instance BFGS or Levenberg-Marquardt methods [91].

The expression of the partial derivative $\frac{\partial J}{\partial w_{ij}^{(l)}}$ in Equation 3.84 remains to determined. For simplicity, the layer index l is omitted. The chain rule can be applied and the partial derivatives are given by:

$$\frac{\partial J}{\partial w_{ij}} = \frac{\partial J}{\partial y_j}\frac{\partial y_j}{\partial x_j}\frac{\partial x_j}{\partial w_{ij}}. \tag{3.85}$$

The last term is derived by using Equation 3.78:

$$\frac{\partial x_j}{\partial w_{ij}} = y_j^{(l-1)}. \tag{3.86}$$

The first two terms form an error term δ_j very convenient to derive the gradient of the ANN and defines by:

$$\delta_j = -\frac{\partial J}{\partial y_j}\frac{\partial y_j}{\partial x_j}. \tag{3.87}$$

Two different cases must be distinguished: if j belongs to an hidden layer or to the output layer. The derivative of the cost function is well defined for the output layer:

$$\begin{aligned}\frac{\partial J}{\partial w_{ij}} &= -\delta_j y_i \\ &= -[a_k(\boldsymbol{\chi}_{t_j}, \boldsymbol{w}) - \widetilde{a_k}(\boldsymbol{\chi}_{t_j}, \boldsymbol{w})]\varphi'(x_j) y_i^{(l)}.\end{aligned} \tag{3.88}$$

However, the derivation of the partial derivative is different if j belongs to an hidden layer as the cost function J is not directly related to the output y_j. The idea is to estimate how the error generated by y_j is propagated into the next layer. The chain rule sums the contribution of y_j to all neurons of the next layer:

$$\begin{aligned}\frac{\partial J}{\partial y_j} &= \sum_{i \in (l+1)} \frac{\partial J}{\partial y_i} \frac{\partial y_i}{\partial x_i} \frac{\partial x_i}{\partial y_j} \\ &= \sum_{i \in (l+1)} -\delta_i w_{ji}.\end{aligned} \tag{3.89}$$

Hence, the substitution in Equation 3.85 yields:

$$\frac{\partial J}{\partial w_{kj}} = -\sum_{i \in (l+1)} \delta_i w_{ji} \varphi'(x_j) y_k^{(l-1)}. \tag{3.90}$$

Two-layers neural networks In the specific example of a two-layers neural network, there is only one hidden layer. Figure 3.9 illustrates this example for regression with an input of size p and m outputs. In the case of regression with only one output ($m = 1$), usually the activation function

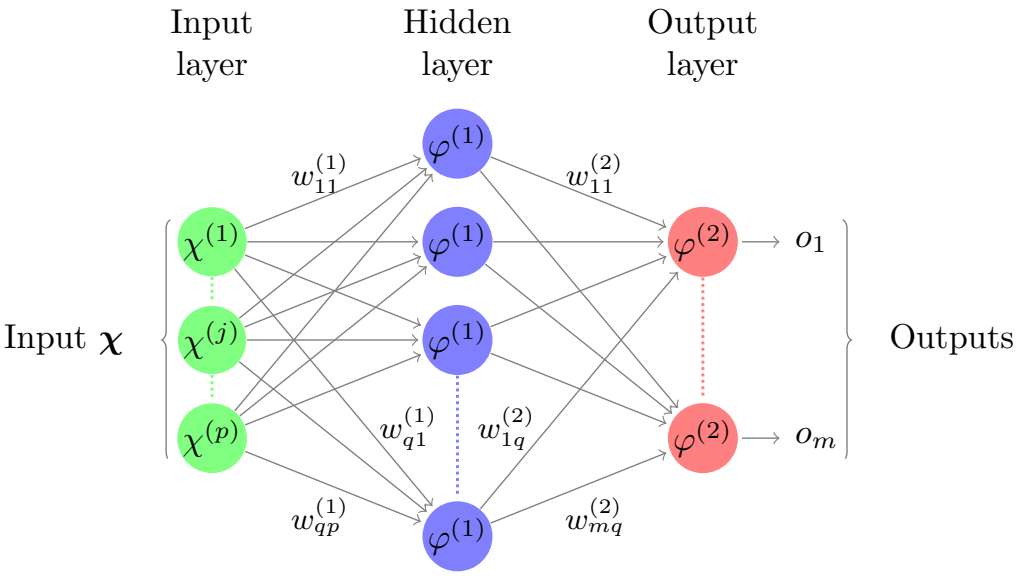


Figure 3.9: Two-layers artificial neural network. Bias parameters are omitted for clarity.

of the output layer is the identity function while the activation function of the hidden layer is a sigmoid σ. The expression for the final regression can be defined as:

$$\widetilde{a_k}(\boldsymbol{\chi}) = \sum_{i=0}^{q} w_{1i}^{(2)} \sigma \left(\sum_{j=0}^{p} w_{ij} \chi^{(j)} \right). \tag{3.91}$$

This expression has $q \times (p+1) + (m+1) \times q$ parameters which are determined by backpropagation.

3.3.3.3 Conclusion on the ANN

The main interests of ANN can be summed up as follows:

- the capacity to predict any continuous function with a multi-layer neural network ensured by the universal approximation theorem [97];
- a performant training method to estimate the weights of the network;
- a great variety of architectures, able to solve various problems: regression, classification, dynamical processes, etc. . . ;
- a possibility to predict multi-dimensional outputs;
- neural networks show a parsimonious structure: for a given function, they require less weights to fit than a linear model [91].

However, ANN remain a complex method. First of all, the architecture of the network must be chosen, such as the number of layers, the types of activation functions, the number of neurons per layer, etc. . . This choice determines the capacity of the neural network to predict complex functions but no general rules exist. In addition, other limitations can be cited:

- the weights and bias must be initialization ($r = 0$). For instance, they can picked from a uniform; distribution with zero mean and given variance;
- the learning must be stopped in order to avoid overfitting, for example with early stopping, and the learning rate γ must be well chosen;
- even with backpropagation, the optimization problem remains difficult to solve as it is non-convex and multimodal;
- the inputs have to scaled.

Usually in the context of data-fit method and surrogate models, the ANN are recommended for high-dimensional problems and when the size of the training sampling is important [178].

3.3.4 Radial basis functions

Radial Basis Function (RBF) method is firstly introduced by Powell [165] in order to approximate any smooth and continuous function as a combination of simple basis functions. Its structure is based on Artificial Neural Network and involves three layers:

- the input layer connects the input to the second layer;
- the hidden layer, also called RBF layer, applies a radial basis function (the activity function) to the input;
- the output layer only contains the identity as activation function and one weighted sum as propagation function.

This configuration leads to the following equation [73]:

$$\widetilde{a}(\boldsymbol{\chi}) = \sum_{k=1}^{n_c} w_i \psi(\|\boldsymbol{\chi} - \boldsymbol{c}_i\|) = \boldsymbol{w}^T \boldsymbol{\psi}, \tag{3.92}$$

where $\|.\|$ is the Euclidean distance, ψ the radial basis function, c_i the centre of i-th basis function, n_c the number of basis functions and $\boldsymbol{\psi}$ the vector containing the values of the basis functions evaluated at $\boldsymbol{\chi}$. The type of basis functions must be chosen and the following expressions are some exemples found in the literature:

$$\begin{aligned} &\text{Linear: } \phi(r) = r \\ &\text{Cubic: } \phi(r) = r^3 \\ &\text{thin plate spline: } \phi(r) = r^2 ln(r) \end{aligned} \tag{3.93}$$

However, parametric basis functions can also be used in order to gain more flexibilit by improving the generalization properties of Equation 3.92. This type of function introduces a new parameter σ, called hyperparameter, such as:

$$\begin{aligned} &\text{Gaussian: } \phi(r) = e^{-(r/\sigma)^2} \\ &\text{Multiquadric: } \phi(r) = \sqrt{\sigma^2 + r^2} \\ &\text{Inverse multiquadric: } \phi(r) = \frac{1}{\sqrt{\sigma^2 + r^2}} \\ &\text{Inverse quadratic: } \phi(r) = \frac{1}{\sigma^2 + r^2} \end{aligned} \tag{3.94}$$

The parameter $\boldsymbol{w}$ is estimated by imposing an interpolation condition, such as the number of basis n_c is equal to the number of samples n (for the uniqueness), leading to the following equation:

$$\Psi \boldsymbol{w} = \boldsymbol{y}. \tag{3.95}$$

If the centers of the basis functions are taken at the data input ($\boldsymbol{c}_j = \boldsymbol{\chi}_j$), Ψ denotes the Gram Matrix defined by $\Psi_{i,j} = \boldsymbol{\psi}(\|\boldsymbol{\chi}_i - \boldsymbol{\chi}_j\|)$. Thus, the weight $\boldsymbol{w}$ is directly computed by inverting the matrix Ψ, provided it exists:

$$\boldsymbol{w} = \Psi^{-1} \boldsymbol{y}. \tag{3.96}$$

Choosing the basis function has therefore an important effect because it directly impacts the inverse computation. For instance, the Gaussian radial function gives a symmetric positive definite Gram matrix efficiently inverted by Cholesky factorisation algorithm [73].

After the computation of the weight $\boldsymbol{w}$, the different parameters of the basis functions are estimated, based on a mimization of the model error. This latter can be performed by cross-validation for instance and the selected basis function parameters are those given the lower cross-validation error.

3.3.5 Regression by Gaussian Process

Gaussian Processes (GP) belong to supervised learning algorithms. They have been designed to solve two distinct types of problem: regression (Gaussian Process Regression) and classification (Gaussian Process Classification). The definition of classification and the use of GP in this context will be discussed in Chapter 4. This section focuses on the Gaussian Process Regression (GPR). It can be interpreted as a distribution over functions where the inference takes place in the space of functions. The presentation of this approach in the function-space formalism follows view of the book of Rasmussen and Williams [168].

The first use of GP dates back to the 1940's. They were studied to predict time series and stochastic processes [119, 203]. GPR is very similar to another method called Kriging. This latter was firstly introduced by the South African mining engineer Daniel G. Krige in mining exploration in the 1970's [120]. Based on a given number of boreholes, he estimated the spatial distribution of gold deposits at unobserved locations in a region of South Africa. His method focuses on spatial statistics and used spatial correlation and weighted average. The theory was mathematically formalized for geostatistics by the french Matheron [147] who gave the name of the so-called *Kriging* [45]. It was applied very early in design and analysis of computer experiments, introduced by Sacks et al. [175]. Then, Kriging has become a popular method to create surrogate models of costly simulations [73]. GPR and Kriging differ mainly on the derivation of the equations and the way of expressing the mean function. The term Kriging in this thesis could refer, somewhat imprecisely, to Gaussian Process and vice versa, in particular in the literature review.

3.3.5.1 Definition of the GP: prior

In the context of regression, GPR aims to build an estimator $\widetilde{a}$ of the reduced coordinates in order to predict values at prediction points $\boldsymbol{X}_p$, also called target points, given the training set $\{a(\boldsymbol{\chi_{t_1}}), \cdots, a(\boldsymbol{\chi_{t_n}})\}$. A GP can be defined as follow: "a collection of random variables, any Gaussian process finite number of which have a joint Gaussian distribution" [168]. In other words, any finite set of function values $(a(\boldsymbol{\chi_{p_1}}), \cdots, a(\boldsymbol{\chi_{p_m}}))$ follows a joint Gaussian distribution which can be conditioned on the training data. Before conditionning the GP on the data, a GP is completely defined by its mean function m and covariance function k, also called kernel:

$$\begin{aligned} m(\boldsymbol{\chi}) &= \mathbb{E}[\boldsymbol{\chi}], \\ \mathrm{Cov}(a(\boldsymbol{\chi}), a(\boldsymbol{\chi})') &= \mathbb{E}\left[(a(\boldsymbol{\chi}) - m(\boldsymbol{\chi}))(a(\boldsymbol{\chi}') - m(\boldsymbol{\chi}'))\right] = k(\boldsymbol{\chi}, \boldsymbol{\chi}'). \end{aligned} \tag{3.97}$$

Usually, the mean function is chosen as zero for GP, which represents one of the differences with the Kriging. In Ordinary Kriging, the mean function is constant while in Universal Kriging, the mean function is chosen with a linear shape $m(\boldsymbol{\chi}) = \phi(\boldsymbol{\chi})\theta$ [117]. Consequently, the GP is fully defined only by the kernel k as the mean function is assumed to be zero.

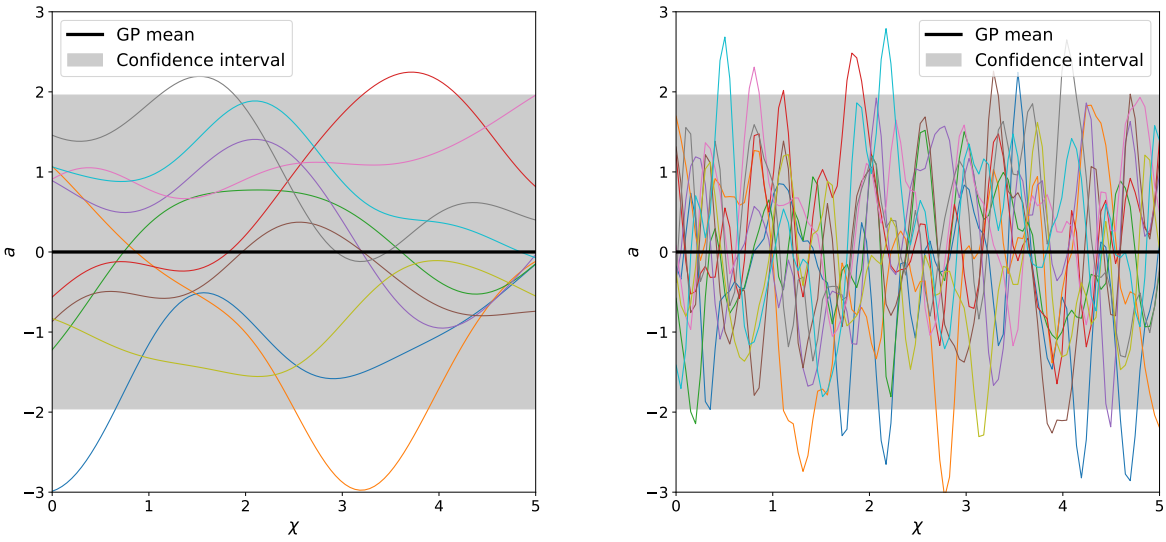
If the function $\widetilde{a}$ follows a Gaussian Process, it is noted such that:

$$\widetilde{a}(\boldsymbol{\chi}) \sim \mathcal{GP}\left(m(\boldsymbol{\chi}), k(\boldsymbol{\chi}, \boldsymbol{\chi}')\right). \tag{3.98}$$

Intuitively speaking, any value $\widetilde{a}(\boldsymbol{\chi})$ of the function $\widetilde{a}$ at a particular $\boldsymbol{\chi}$ is a component of a infinite multivariate Gaussian distribution. The training on an infinite distribution may appear computationally untracable. However, the fact that the GP is based on Gaussian distributions represents one of its strenghts. In particular, the marginalization property of Gaussians states that "the examination of a larger set of variables does not change the distribution of the smaller set" [168]. Thus, it means a set of observation does not influence the distribution of other smaller or larger sets. In the context of regression, an interpretation can be given to the two statistical moments of the distribution. The mean provides the final value of the prediction at the untried set of parameters, while the variance (or the standard deviation) estimates the possible range taken by the prediction which can also be seen as the mean-square error.

In our case, the function a is evaluated for a finite number of inputs. Therefore, the infinite vector can be considered as a high dimensional vector. The joint distribution of the reduced coordinates at training and unknown parameters is given by:

$$\begin{bmatrix} \boldsymbol{A}_t \\ \boldsymbol{A}_p \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu}_t \\ \boldsymbol{\mu}_p \end{bmatrix}, \begin{bmatrix} \boldsymbol{\Sigma}_{tt} & \boldsymbol{\Sigma}_{tp} \\ \boldsymbol{\Sigma}_{pt} & \boldsymbol{\Sigma}_{pp} \end{bmatrix}\right), \tag{3.99}$$



(a) Smooth prior

(b) Unsmooth prior

Figure 3.10: Samples from a GP prior. Each curve corresponds to a sample from a infinite multivariate Gaussian distribution. The thick black line represents the mean of the GP. The grey shade indicates a 95% confidence interval given by 1.96 standard deviation of the GP, coming from the cumulative distribution function of the normal distribution.

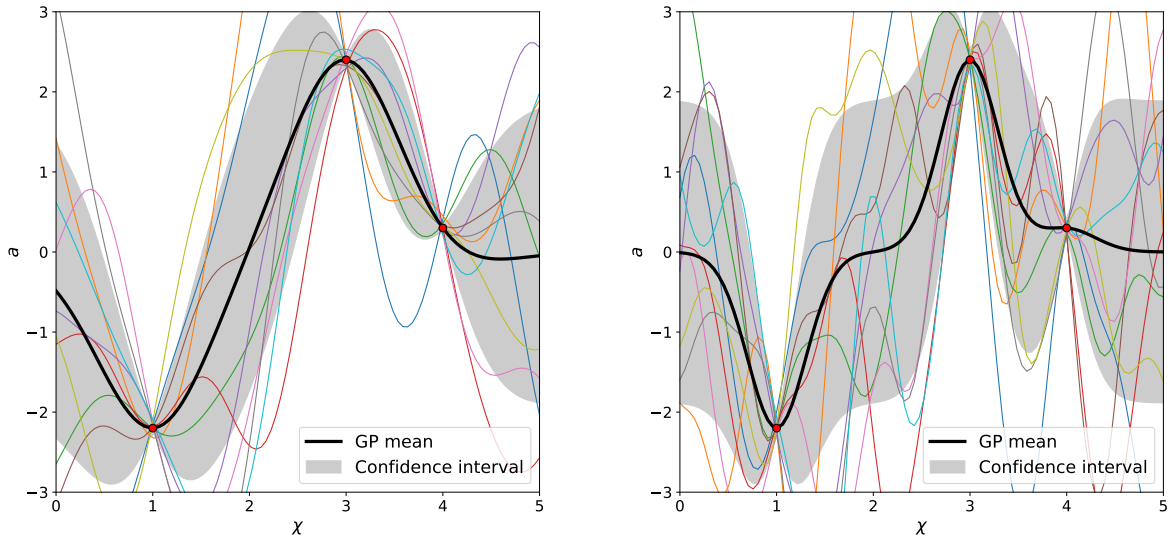
with $\boldsymbol{\Sigma_{tp}}$ the covariance matrix between $\boldsymbol{\chi}_t$ and $\boldsymbol{\chi}_p$ defined by:

$$\boldsymbol{\Sigma_{tp}} = \begin{bmatrix} k(\boldsymbol{\chi}_{t_1}, \boldsymbol{\chi}_{p_1}) & \cdots & k(\boldsymbol{\chi}_{t_1}, \boldsymbol{\chi}_{p_m}) \\ \vdots & & \vdots \\ k(\boldsymbol{\chi}_{t_n}, \boldsymbol{\chi}_{p_1}) & \cdots & k(\boldsymbol{\chi}_{t_n}, \boldsymbol{\chi}_{p_m}) \end{bmatrix} \in \mathbb{R}^{n \times m}. \tag{3.100}$$

All possible combination between training and prediction points are computed by the covariance function k, building each component of the covariance matrix $\boldsymbol{\Sigma_{tp}}$. The matrices $\boldsymbol{\Sigma_{tt}} \in \mathbb{R}^{n \times n}$ and $\boldsymbol{\Sigma_{pp}} \in \mathbb{R}^{m \times m}$ are defined in the same manner by substituting respectively the prediction index p by the training index t and vice versa in Equation 3.100.

In pratical manner, building a GPR is divided into three parts:

- First of all, an assumed shape is given to the sough function by choosing a specific kernel k. This shape can be interpreted in term of derivability, regularity, periodicity properties, etc. All functions satisfying this assumption belong to what is called the prior, just as in Bayesian statistical inference. Figure 3.10 illustrates two priors with different kernels providing specific shapes. The GPR model of this stage can be seen as fully unknown, represented by the confidence interval of Figure 3.10;
- Thereafter, information are added to the GP and the functions from the prior that disagree with the observations are rejected. The remaining functions are called the posterior. This process is discussed in the following section;
- Lastly, the covariance function is tuned by choosing the best parameters fitting the training data. Details are given on this step.



(a) Smooth prior conditioned on training data

(b) Unsmooth prior conditioned on training data

Figure 3.11: Samples from a GP prior conditioned on training data (the posterior). Mean and covariance functions have been altered by the conditioning. The uncertainties of the GP are zero on the training points. The GP mean of the unsmooth prior tends to the mean value of the prior (zero) for prediction points far from any training points.

3.3.5.2 Incorporing information to the GP: posterior

As mentioned above, the prior is restricted to functions which fit the training data. It might be thought that this approach is computationally expensive. Fortunately, this operation consists of computing the conditional propability. In the statistical vocabulary, it refers to conditioning the Gaussian distribution on the training data, which is also Gaussian. The final form of the new predictor is called a posterior distribution. It corresponds to the conditional distribution of $\boldsymbol{A}_p$ given $\boldsymbol{A}_t$ and is written as:

$$\boldsymbol{A}_p|\boldsymbol{A}_t \sim \mathcal{N}\left(\boldsymbol{\mu}_p + \boldsymbol{\Sigma}_{pt}\boldsymbol{\Sigma}_{tt}^{-1}(\boldsymbol{A}_t - \boldsymbol{\mu}_t),\ \boldsymbol{\Sigma}_{pp} - \boldsymbol{\Sigma}_{pt}\boldsymbol{\Sigma}_{tt}^{-1}\boldsymbol{\Sigma}_{tp}\right). \tag{3.101}$$

Conditioning the GP corrects the prior mean $\boldsymbol{\mu}_p$ and prior covariance matrix $\boldsymbol{\Sigma}_{pp}$ by adding information on the training data. Moreover, computing the inverse of the covariance matrix $\boldsymbol{\Sigma}_{tt}^{-1}$ is necessary to compute the posterior. Figure 3.11 highlights the difference between two posteriors conditioned on the same training data but from two different priors. The variance corrected by the training data has the interesting feature not to be dependent on the value of the output but only on the input parameters, as shown by Equation 3.101. The central issue of the GPR remains to be addressed, namely how to build the kernel function which highly influences the conditioned mean and covariance of the GP.

3.3.5.3 Covariance function

The covariance function is one of the main crutial ingredient of the GP by measuring the similarity between two points. Indeed, if the prediction at a test point appears as similar to the output at a training point, the predicted value will be highly influenced by the training output. This section discusses how is expressed the covariance function and gives details on kernel functions.

Definition As a reminder, the covariance function between two inputs is defined by a kernel function k in the context of GP:

$$\mathrm{Cov}[a(\boldsymbol{\chi}), a(\boldsymbol{\chi})] = k(\boldsymbol{\chi}, \boldsymbol{\chi}'). \tag{3.102}$$

The kernel function k can be defined by any positive definite function of two variables [118, 168]. However, this property is very general and can be difficult to prove. As a result, additional assumptions usually specify the kernel function. The most classical one is the stationarity assumption which states that the correlation depends only on the magnitude of the Euclidean distance between the two input parameters but not on the values themselves, such that kernel functions depends only on the difference $\boldsymbol{\chi} - \boldsymbol{\chi}'$. In practice, only a few well-known kernels are considered. Some of them are discussed in the following section.

Some examples of kernel functions A wide range of functions have been proposed to model the relation between the covariance and the input distance, such as the squared exponential, Matérn or periodic regression function [168]. They are usually defined by free parameters, called hyperparameters. They allow to control the shape given to the covariance function and their tunning plays a central role in the accuracy of the regression.

One of the most classical model of kernel functions is the Squared Exponential function, also called the Radial Basis Function. Its expression introduces two hyperparameters θ and σ_0:

$$k(\boldsymbol{\chi}, \boldsymbol{\chi}') = \sigma_0 \exp\left(\frac{-\|\boldsymbol{\chi} - \boldsymbol{\chi}'\|_2}{2\theta}\right) \quad \forall \boldsymbol{\chi}, \boldsymbol{\chi}' \in (\mathbb{R}^p)^2. \tag{3.103}$$

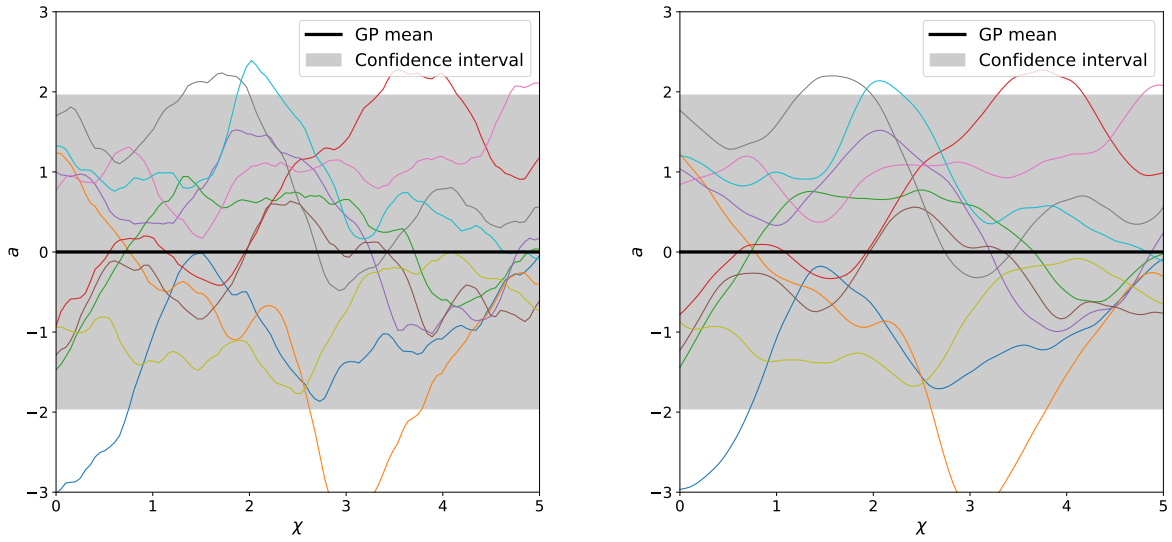
The scalar σ_0^2 is called the signal variance and control the amplitude. The length scale θ precises the smoothness of the function. On the one hand, a small value means the correlation is high between the inputs, as the model is very sensitive to the input parameter. On the other hand, a large value of θ denotes a model slowly varying with $\boldsymbol{\chi}$. The samples of Figure 3.10 are in fact drawn from two Square Exponential (SE) kernels, where the "smooth" priors correspond to a length scale of 1.0 and the "unsmooth" prior to a length scale of 0.1. The popularity of the Square Exponential kernel is explained by several properties: there are few hyperparameters to be fixed, they are easily interpretable, and the kernel is universal [56]. The universality property states that any continuous function can be represented by the kernel under specific conditions. In addition, the SE kernel gives mathematical properties to priors: smoothness, infinite differentiability and analytical derivability.

Another example of kernel function is the Matérn function:

$$k(\boldsymbol{\chi}, \boldsymbol{\chi}') = \sigma_0 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu}\|\boldsymbol{\chi} - \boldsymbol{\chi}'\|_2}{\theta}\right)^{\nu} K_\nu \left(\frac{\sqrt{2\nu}\|\boldsymbol{\chi} - \boldsymbol{\chi}'\|_2}{\theta}\right), \ \forall \boldsymbol{\chi}, \boldsymbol{\chi}' \in (\mathbb{R}^p)^2, \tag{3.104}$$

where Γ is the Gamma function, K_ν is the modified Bessel function, and σ_0, ν, and θ are hyperparameters. Just as for the Square Exponential, σ_0 controls the amplitude of correlation and θ the sensitivity to neighboring points. The hyperparameter ν plays a very interesting role and gives a large flexibility to the Matérn kernel by controling the differentiability, as illustrated in Figure 3.12. Indeed, the correlation function is m times differentiable if and only if $\nu \leq m$ [118]. One can note that if ν tends towards infinity, the Matérn kernel converges to the SE kernel.

Both kernels presented above are isotropic and raise a major issue. If the output is not equallly sensitive to all components of the input parameter, a kernel with a single hyperparameter does not distinguish the contribution of each components to the covariance. For this reason, it is preferable to employ multidimensional kernels, also called anisotropic kernels. The latter have different shapes



(a) Matérn prior for $(\theta, \nu) = (1.0,\ 1.5)$

(b) Matérn prior for $(\theta, \nu) = (1.0,\ 2.5)$

Figure 3.12: Illustrations of two Matérn priors with different ν associated with different differentiability.

for each component of the input parameter. A classical way to build multi-dimensional kernel consists of multiplying together the univariate kernel of each component of the input, such as:

$$k(\boldsymbol{\chi}, \boldsymbol{\chi}') = k\left(\boldsymbol{\chi}^{(1)}, \boldsymbol{\chi}'^{(1)}\right) \times \cdots \times k\left(\boldsymbol{\chi}^{(d)}, \boldsymbol{\chi}'^{(d)}\right). \tag{3.105}$$

Therefore, the anisotropic kernel increases the number of hyperparameters with the dimension of the input parameter. In the example of the Squared Exponential kernel, the isotropic kernel has 2 hyperparameters while the anisotropic kernel has $d + 1$ hyperparameters [118, 56]. Indeed, the product of univariate Square Exponential can be reduced such as:

$$k(\boldsymbol{\chi}, \boldsymbol{\chi}') = \sigma_0 \prod_{k=1}^{d} \exp\left(\frac{-\|\boldsymbol{\chi}^{(k)} - \boldsymbol{\chi}'^{(k)}\|_2}{2\theta_k}\right), \text{ where } \sigma_0 = \prod_{k=1}^{d} \sigma_k. \tag{3.106}$$

3.3.5.4 Determining the hyperparameters

The different examples of kernel have introduced hyperparameters which remain to be determined during the training phase of the GPR. This problem can be interpreted as a model selection. Two methods are widely used in the literature: the cross validation (CV) and the Maximum Likelihood Estimation (MLE) [168]. The hyperparameters estimations by both MLE and CV have been compared in Martin and Simpson [146]. The auhors conclude that the MLE performs better than the CV. This result is demonstrated on 6 different test problems.

Regarding MLE, the GP formalism provides a fully probabilistic framework and it will be very beneficial to determine the hyperparameters directly by finding the best prior that fit the training set, for a given shape of covariance function. However, maximizing the posterior probability $P(\boldsymbol{\theta}|\boldsymbol{A}_t, \boldsymbol{X}_t)$ appears as analytically intractable. Consequently, the marginal likelihood $p(\boldsymbol{A}_t|\boldsymbol{X}_t, \boldsymbol{\theta})$ is used to define a new maximization problem by invoking Baye's theorem:

$$p(\boldsymbol{\theta}|\boldsymbol{A}_t, \boldsymbol{X}_t) \propto p(\boldsymbol{A}_t|\boldsymbol{X}_t, \boldsymbol{\theta}) p(\boldsymbol{\theta}). \tag{3.107}$$

In practice, the minimization of the negative log of the marginal likelihood is considered. The expression of the latter can be analytically derived:

$$\log(p(\boldsymbol{A}_t|\boldsymbol{\theta})) = -\frac{1}{2}\boldsymbol{A}_t^T\boldsymbol{\Sigma}_{tt}^{-1}\boldsymbol{A}_t - \frac{1}{2}\log|\boldsymbol{\Sigma}_{tt}| - \frac{n}{2}\log(2\pi), \tag{3.108}$$

where $|\cdot|$ denotes the determinant operator. An interpretation can be given to the three terms of Equation 3.108: $-\frac{1}{2}\boldsymbol{A}_t^T\boldsymbol{\Sigma}_{tt}^{-1}\boldsymbol{A}_t$ is the only term depending on the training data $\boldsymbol{A}_t$ and acts as a data fit term, $-\frac{1}{2}\log|\boldsymbol{\Sigma}_{tt}|$ penalizes the complexity of the GP, and $-\frac{n}{2}\log(2\pi)$ is just a normalization term without influence on the training phase [168]. The minimization of the negative value of Equation 3.108 shows a non-convex property. Thus, it is intersting to use the gradient information in order to solve the minimization problem. The partial derivatives of the marginal likelihood with regards to the hyperparameters can be analytically derived:

$$\frac{\partial}{\partial\theta_j}\log\left(p(\boldsymbol{A}_t|\boldsymbol{\theta},\sigma)\right) = \frac{1}{2}\boldsymbol{A}_t^T\boldsymbol{\Sigma}_{tt}^{-1}\frac{\partial\boldsymbol{\Sigma}_{tt}}{\partial\theta_j}\boldsymbol{\Sigma}_{tt}^{-1}\boldsymbol{A}_t - \frac{1}{2}\mathrm{tr}(\boldsymbol{\Sigma}_{tt}^{-1}\frac{\partial\boldsymbol{\Sigma}_{tt}}{\partial\theta_j}) \;\forall j \in [0,p]. \tag{3.109}$$

The Limited-memory Broyden-Fletcher-Goldfarb-Shanno Bounded (L-BFGS-B) algorithm [29] is employed in this work to determine the hyperparameters. This popular quasi-Newton method handles simple bound constraints. However, the marginal likelihood encounters local minima which give different interpretation to the training data. Thus, the gradient-based optimization algorithm is often coupled with random restarts to avoid local maxima of bad quality. Similarly, a Particle Swarm approach coupled with local optimization algorithms has been proposed in order to solve the MLE [187].

The choice between CV and MLE is not evident. Martin and Simpson [145] and Rasmussen and Williams [168] do not have a definitive answer. Neither method show better results in term of accuracy and both need comparable computational ressources. Indeed, it has been demonstrated that CV avoids numerical issues of the MLE due to ill-conditioning of the covariance matrix. However, the CV can still performs worse. Thus, the main differences concern the implementation and the fact that the MLE assumes the data follow a Gaussian process while the estimate with the frequentist CV still holds, regardless of the assumption on the data.

3.3.5.5 Conclusion on the GPR

The GPR has proved to be an effective data-fit method. In terms of advantages, the GPR shows great capability to deal with nonlinear problems as long as the number of input parameters is small (less than about 50) [69]. In addition, the formalism of the GPR gives access to an estimation of the modeling error. The latter can be very usefull to derive an confidence interval of the surrogate model or to perform adaptive sampling. The possibility to define the covariance matrix with various mathematical properties offer a great flexibility, such as periodicity, monotony, or differentiability. Finally, the GPR and the Kriging provide additional versions capable of dealing with different levels of fidelity or to incorporate gradient information.

However, the GPR shows several disadvantages, in particular numerical issues. Indeed, the main source of complexity for the GP comes from the inversion of the covariance matrix $\boldsymbol{\Sigma}_{tt}$ which occurs in the Equations 3.101 and 3.108, associated respectively with the predictions of the GP and the selection of hyperparameters. Therefore, a GP has a computational complexity of $\mathcal{O}(n^3)$ and a memory requirement of $\mathcal{O}(n^2)$ for a training matrix of size n. In practice, a GP cannot be employed for a problems with more than 10^4 training points. Nevertheless, approximation methods have been developed to deal with this limitation, such as sparse GP [183] or stochastic variational inference [93]. In addition, the MLE can fail to converge. Thus, the selected hyperparameters are not adapted to the approximated function.

3.3.6 Comparison of surrogate model methods

Various kinds of surrogate models have led to several comparative studies [104, 179, 144]. However, there still is no clear consensus as they very largely depends on the problem, the type of data, the DOE, etc. Specific developments have combined different types of surrogate models. Viana et al. [194] propose to use a linear combination of different surrogate models. The latter are weighted in function of their cross-validation errors. All the model can be used or only a subset of the best ones. The final objective is to generate an automatic, robust, and accurate surrogate model. Table 3.2 gives an overview of advantages and limitations of data-fit methods presented in this chapter.

Surrogate model type	**Characteristics**
Polynomial Regression	+ Easy to use and low computational cost - Not appropriate for applications with more than 10 factors - Unsuitable for highly nonlinear problem
Support Vector Regression	+ Non-linear capability + Flexibility - Complexity: dynamic programming - Parameters to define: penalty, ϵ, kernel form
Neural Networks	+ Suitable for highly nonlinear problem + Well-suited for determinist application + Vectorial output - High number of sampled point - Complicated interpretation of the model
Radial Basis Function	+ Easy to implement + Mathematically close to ordinary Kriging and smaller computational cost + Flexibility in the choice of the basis function - Complicated interpretation of the model
Kriging	+ High flexibility and estimated error of the predictor + Number of Kriging versions (ordinary, universal, blind, cokriging, etc...) + Suitable for highly nonlinear problem - Complexity: multiple optimization algorithms - Very high computational cost to build the model

Table 3.2: Surrogate model characteristics

In general, Kriging appears to be a reference method for problems with nonlinear features, moderate numbers of input parameters (<10), and training samples [69, 178]. Indeed, the numerical complexity of the Kriging does not restrict its use in CFD as simulations are usually computationally intensive and the number of training samples limited. In addition, the Kriging provides an estimated error in its predictions. For these reasons, the Kriging is used as the data-fit method in the thesis.

3.4 Design of Experiments

Design Of Experiments (DOE) answers the following question: how to chose the input parameters of a set of experiments? In other words, how to distribute the training set $\{a(\boldsymbol{\chi_{t_1}}), \cdots, a(\boldsymbol{\chi_{t_n}})\}$. Fisher [71] introduced randomization tests and gave its first definition to the DOE in the context of real experimentations. Its application to numerical simulators has been generalized by Sacks et al. [175]. The analysis of DOE for deterministic simulators is of particular interest in this thesis. DOE has shown many applications, in particular for:

- **identifying the contribution** of each input to the output of the model and determining their interactions and their range of influence. This application is called screening [62] and is used at the beginning of the numerical or experimental procedure;
- **building a statistical or a surrogate model** by generating a set of training points. Usually, a classical strategy consists of first exploring the parameter space, this step is called the exploration, and then specific samples can complete the inital sampling by exploiting the model, also called the exploitation step. The latter can have different objectives. Regarding global surrogate-model, extra samples are added where the function seems complex and unknow in order to reduce the approximation error. Surrogate-based black-box optimization is another example and queries new points in specific areas of potential gain on the function to optimize [198]. Similar strategy can be adapted to estimate the probability to exceed a threshold [57] and to guarantee robustness [62];
- **computing the value of an integral** of an unknown function. This process requires to evaluate the function at specific values in the parameter space. In the absence of any information about the regularity of the function, the Monte Carlo method (random sampling) is frequently used. However, its convergence speed reaches $\mathcal{O}(\frac{1}{\sqrt{n}})$, where n is the number of points. Other kinds of DOE can accelerate the convergence speed under specific assumptions, such as quasi Monte Carlo methods [191].

In the context of the thesis, we are mainly interested in building surrogate models. These kinds of DOE can be classified into two categories [206]:

- **model-free designs** generate a priori distributions of samples based only on geometrical properties of the parameter space, such as the distance between training samples;
- **adaptive designs**, on the contrary, use information given by the evaluations of the function of interest and by the surrogate model in order to propose relevant new samples improving the model in a precise objective, such as finding the minimum of the function or enhancing the accuracy of the surrogate model.

These two designs are presented in this section after the introduction of several metrics quantifying some properties of a DOE.

3.4.1 Main properties of DOE

Before introducing examples of DOE, several properties quantifying the quality of DOE are defined in term of spatial distribution, iterativity, and stochasticity.

3.4.1.1 Spatial distribution

A good initial DOE must fullfil two fundamental requirements [177, 198]:

- **space-fill** or **uniformity**: the samples must be uniformly distributed over the parameter space. As the nature of the model is unknown, the design points should be evenly distributed throughout the parameter space in order to observe the response of the model at various conditions;
- **Non-collapse**: Each dimension must be explored as far as possible over the range of its variation. Therefore, two samples should not share a commun value for a given dimension.

DOE seeks trade-off between the two concepts of space-filling and non-collapsing. In high dimension, finding an optimal value for this trade-off is particulary complex [198]. Specific geometrical quantities can be defined to measure the space-filling of a design. For instance, the discrepancy and the minimal interpoint distance play a major role to build DOE.

The discrepancy measures the difference between theoretical uniformity of the distribution and DOE. A first local definition is introduced. Let asssume the parameter space is normalized in all directions. The vector $\boldsymbol{t}$ is used to define a local maximum bound of the parameter space, such as $[\boldsymbol{0}, \boldsymbol{t}]$ denotes the interval $[0, t_1] \times \cdots \times [0, t_s]$ forming a subset of the parameter space Γ. The local discrepancy is defined for a given subset at the origin of the pameter space and for a given sampling $\boldsymbol{X}$ [68]:

$$disc(\boldsymbol{X}, \boldsymbol{t}) = \frac{A(\boldsymbol{X}, \boldsymbol{t})}{n} - Vol(\boldsymbol{t}), \text{ where } Vol(\boldsymbol{t}) = \prod_{k=1}^{d} t^k. \tag{3.110}$$

The function A gives the number of samples located in the interval $[\boldsymbol{0}, \boldsymbol{t})$ and $Vol(\boldsymbol{t})$ denotes the volume of this interval. Equation 3.110 can be interpreted as follow: more uniform is the DOE in the subset defined by $\boldsymbol{t}$, closer to the hypervolume area $\prod_{k=1}^{d} t^k$ is the local density of samples A. In other words, low discrepancies are synonymous with well distributed sampling. Figure 3.13 illustrates an example of the local interval in two-dimension. In order to ensure that a design has a good uniformity property, a global metric covering all the parameter space is needed. Many useful measures of uniformity have been defined in the literature, mainly based on the L_p discrepancy D_p:

$$D_p(\boldsymbol{X}) = \left[\int_{[0,1[^s} disc^p(\boldsymbol{X}, \boldsymbol{t}) d\boldsymbol{t} \right]^{1/p}. \tag{3.111}$$

A popular discrepancy metric, called star or L_∞ discrepancy D_∞, takes $p = \infty$, such as:

$$D_\infty(\boldsymbol{X}) = \sup_{\boldsymbol{t} \in [0,1[^p} |disc(\boldsymbol{X}, \boldsymbol{t})|. \tag{3.112}$$

It can be interpreted as follows: if a region with a high local discrepancy value can be found for a given design $\boldsymbol{X}$ then the latter is described as non-uniform. Indeed, a uniform design must shows good property all over the parameter space. However, computing infinite discrepancy is intractable in practice for high-dimension cases due to the complexity of algorithms growing with the dimension of the problem [69]. Alternatives have been proposed, such as the L_2 discrepancy. However, Hickernell [95] highlights weaknesses of the L_p-discrepancy. The author has proposed the centered L_2-discrepency as an alternative. This modification is invariant under reflections of any plane $\chi^{(j)} = 0.5$ and is defined by:

$$D^2_{L_2\ center}(\boldsymbol{X}) = \left[\sum_{u \neq \emptyset} \int_{[0,1[^{card(\boldsymbol{u})}} \left| \frac{A(\boldsymbol{X}, \boldsymbol{t_u})}{n} - Vol(\boldsymbol{t_u}) \right|^p d\boldsymbol{t_u} \right]^{1/p}, \tag{3.113}$$

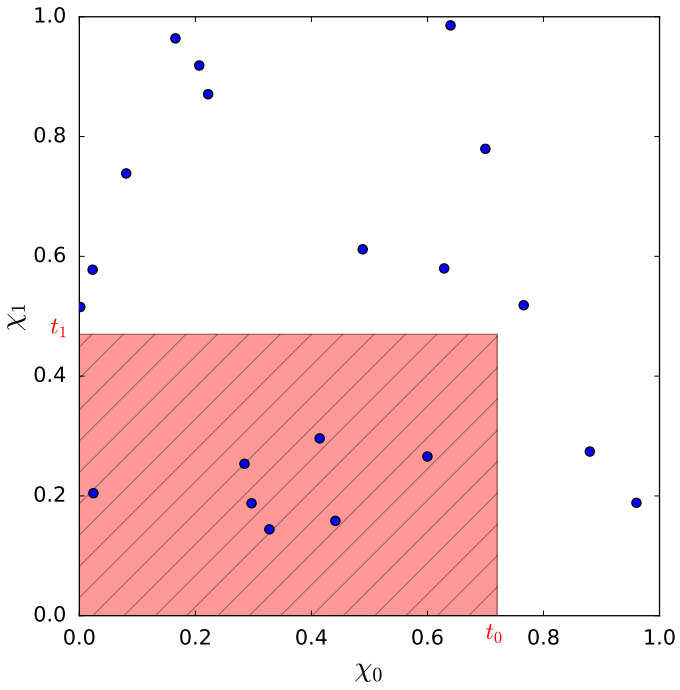


Figure 3.13: Illustration of the local discrepancy.

where u is a nonempty subset of the set of coordinate indice $\{1, \cdots, p\}$, $\boldsymbol{t_u}$ is the projection of $\boldsymbol{t}$ on $[0,1[^{card(\boldsymbol{u})}$. An analytical formula has been derived for the centered discrepancy:

$$\begin{aligned} D^2_{L_2\ center}(\boldsymbol{X}) = & \frac{13^p}{12} - \frac{2}{n}\sum_{k=1}^{n}\prod_{j=1}^{p}\left[1+\frac{1}{2}|\chi_k^{(j)}-0.5| - \frac{1}{2}|\chi_k^{(j)}-0.5|^2\right] \\ & + \frac{1}{n^2}\sum_{k=1}^{n}\sum_{j=1}^{n}\prod_{i=1}^{p}\left[1+\frac{1}{2}|\chi_k^{(i)}-0.5+\frac{1}{2}|\chi_k^{(i)}-0.5| - \frac{1}{2}|\chi_k^{(i)}-\chi_j^{(i)}|^2\right]. \end{aligned} \tag{3.114}$$

Other kinds of discrepancy exist, such as the wrap-around or the categorical discrepancy. Details of these discrepancies can be found in Fang et al. [69].

3.4.1.2 One-shot vs sequential

Independently of geometrical qualities, DOE can be either one-shot space-filling or sequential sampling [131]. The first property means that the parameter space is sampled with a given number of experiments defined a priori. The parameter space is covered in a single stage as uniformly as possible. Fixing a sample size in advance appears to be very difficult if the complexity and the properties of the simulator are unknown. The sequential sampling proposes an evolutionary and flexible approach, filling iteratively the parameter space based on the sampling at the previous iteration. The sequential method can also be divided in two paradigms: space-filling or goal-oriented. The sequential space-filling sampling has the same objective than the one-shot space-filling but the number of samples is not fixed in advance and can be increased on purpose, while the goal-oriented sampling, also called active learning, choses the next samples in function of a specific objective. Usually, the regions of special interest are focused, such as high uncertainty, minimum values, high gradients, etc...

In both one-shot and sequential space-filling methods, the evaluation of each sampling point by the simulator can be easily performed in parallel. All the evaluations are independent of the values taken by the quantity of interest, ensuring a full use of the available computation resources. However, it is not necessarily the case for adaptive sampling methods which need to have access to values taken by the quantity of interest at previous samples. Only a small batch of dozen simulations can be launched at each iteration. Hence, a particular attention should be given to the parallel capabilities of the active learning methods.

3.4.1.3 Stochastic vs deterministic

In this work, DOE is mainly used for a priori sampling plan, active learning and sensitivity analysis applied to surrogate models. Thus, this section focuses on space-filling sampling and resampling strategies. A model-free stance is taken, consequently model-dependant sampling, such as Box-Behnken or central composite design are not considered. Two main ways of filling techniques are described below:

- the **stochastic** sampling methods introduce a variability in results and are well-suited for data exhibiting random and unknown errors such as physical experiments. However, they also have drawbacks: these DOE can be difficult to reproduce and they have a non-zero probability to build a sampling of poor quality. However, an optimization algorithm can be coupled to the stochastic DOE to select the best distribution amongt all combinations. Monte-Carlo design, Latin Hypercube Sampling (LHS), or optimized LHS are examples of stochastic appoaches;
- the **deterministic** filling methods do not have a variability in the results. Their properties are known in advance and they are naturally very suitable for deterministic systems, such as numerical experiments. One can cite for instance full factorial, central composite, or low-discrepancy sequences.

One can note that a numerical seed ensures the reproductability of random processes. Thus, stochastic sampling can also be considered for DOE. Both types of designs are presented in the following sections.

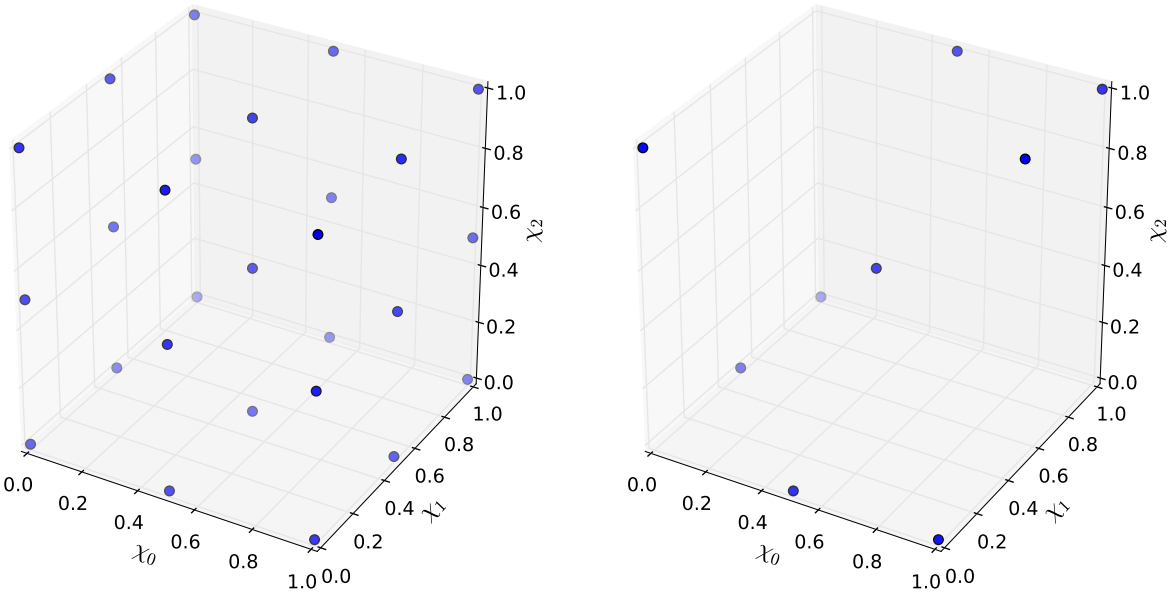
3.4.2 Model-free designs

The model-free designs, also called a priori or non-adaptive designs, fill the parameter space only based on geometrical aspects. This type of DOE does not query the values of the function of interest. Various designs are presented below.

3.4.2.1 Factorial

Full factorial DOE is the most basic and one of the most costly approach to sample the parameter space. The method considers all possible combinations at a given level by uniformly partitioning each parameter range. Its discrepancy tends to zero. However, the number of experiments increases exponentially with the dimension of the parameter space. If each dimension is partitioned in l levels and if n experiments are considered, this leads to get l^n points in the DOE. The approach becomes rapidly out of cost. The full factorial design shows very well uniformity properties but is collapsing at an important level.

An alternative to the full factorial DOE is the fractional factorial DOE. The number of samples is reduced by taking only a fraction of the full factorial DOE. A classical design takes a fraction $\frac{1}{l^p}$, such as the number of experiments is reduced to l^{n-p}. Figure 3.14 illustrates the two factorial approaches. The fractional factorial fills the parameter space with fewer samples than the full factorial. However, a bias is introduced in measuring the main effects as all the intereaction are not taken into account [178].



(a) Full factorial DOE

(b) Fractional factorial DOE

Figure 3.14: Comparison between full and fractional factorial DOE at 3 levels for a three dimensional parameter space.

3.4.2.2 Maximin and minimax design

Maximin and minimax designs have been introduced by Johnson in the 90's [107]. They are both based on very intuitive ideas that allow to cover very uniformly the parameter space.

Maximin design The sampling is assumed to be sparse, due to the size of the parameter space, and each sample is sought as the most distant to each other. It leads to a simple criterion maximizing the minimum distance among all possible pairs of samples [107]:

$$\max_{card(\boldsymbol{X_t})=n} \left(\min_{\boldsymbol{\chi},\boldsymbol{\chi}' \in \boldsymbol{X}_t^2} d(\boldsymbol{\chi},\boldsymbol{\chi}') \right).$$

This method aims at finding the sampling with the largest value of the smallest distance between neighboring points. Samplings with two or more samples that are too close to each other are rejected. The design is illustrated in Figure 3.15. It can be observed that design points tends to be pushed to the boundary. The criterion can be interpreted as follows: each design points is located at the center of a circle and the overlapping between circles must be minimized.

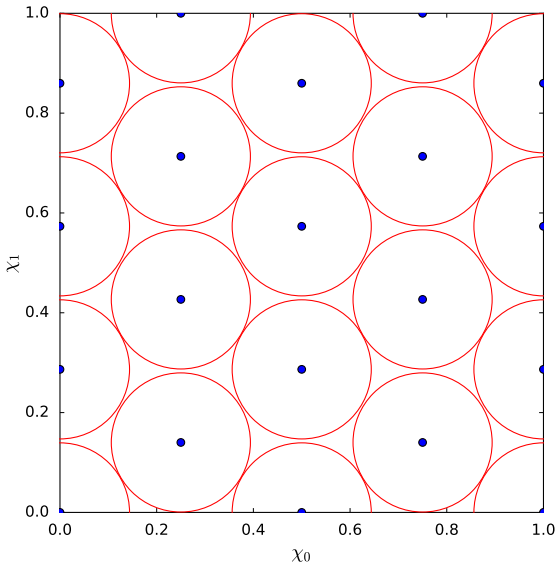


Figure 3.15: Maximin design. $D = 0.114$.

Miximax design The minimax-distance design proposes a slightly different approach. Each point of the parameter space Γ must be "close" to at least one point of the sampling $\boldsymbol{X_t}$. The distance between a sample $\boldsymbol{\chi}$ and its training set $\boldsymbol{X_t}$ is introduced as the minimum between all the possible distance $d(\boldsymbol{\chi}, \boldsymbol{X_t}) = \min_{\boldsymbol{\chi}' \in \boldsymbol{X_t}} d(\boldsymbol{\chi}, \boldsymbol{\chi}')$. The minimax distance criterion can be defined as [107]:

$$\min_{card(\boldsymbol{X_t})=n} \left(\max_{\boldsymbol{\chi} \in \Gamma} d(\boldsymbol{\chi}, \boldsymbol{X_t}) \right).$$

One can note that no evidence can be found in the literature for the choice of the norm defining the distance (Manhattan, Euclidiean, etc.) [73].

3.4.2.3 Monte Carlo

Monte Carlo experiments compute samples randomly distributed in the parameter space. Despite the random process, Monte Carlo approaches have been widely used to solve deterministic problems, in particular in numerical integration and optimization [191]. In the context of DOE, it represents one of the simplest approach to perform sequential space-filling. Naturally, the probability to generate a non-uniform design is non-zero. Monte Carlo sampling has been widely studied in the literature and several convergence properties have been derived. However, they are mainly based on an important number of samples, which is not the case in the context of reduced-order models for expensive engineering simulations.

3.4.2.4 Latin Hypercube Sampling

LHS is an effective technique for space filling based on preservation of uniformity [148]. If the number of experiments is assumed to be n, each dimension is splitted into n equiprobable subintervals. The samples are randomly distributed in each subintervals such as each sample is the only one in each axis-aligned hyperplane containing it. Figure 3.17a and Figure 3.17b show two different LHS samplings in two dimensional case. The left sampling illustrates two of the main properties

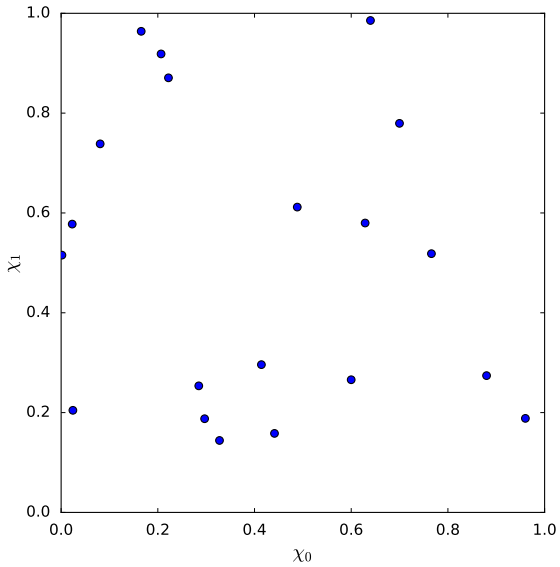
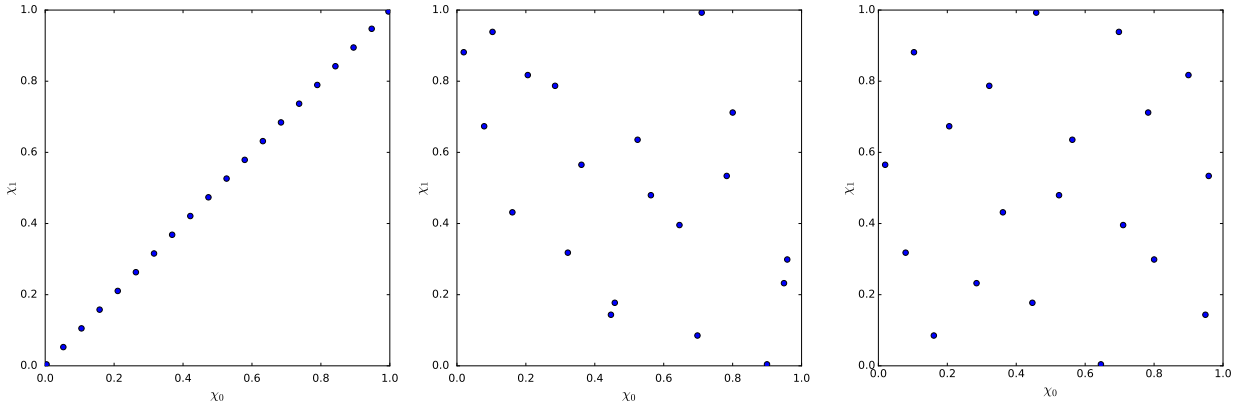


Figure 3.16: Monte Carlo sampling. Discrepancy of 0.127.

of LHS: the projection of the samples on each axis of the parameter space is close to a uniform distribution and the sampling is independant of the dimension of the parameter space. All samples have been placed along diagonal of design space and the LHS requirements are matched. However, the space-filling is very poor as areas far from diagonal remain unexplored. It highlights one of the drawback of LHS: the filling of the parameter space is not properly ensured, contrary to the non-collapsing property.



(a) Raw LHS. $D = 0.116$ (b) Random LHS. $D = 0.0713$ (c) Optimized LHS. $D_2 = 0.0352$

Figure 3.17: Examples of DOE based on LHS.

Therefore, enhancements of LHS have been proposed. The classical idea optimizes a criterion based on the uniformity of the sampling, such as discrepancy or minimax criterion. A given number of LHS sampling is generated and the design with the best value of the space filling criterion is chosen, ensuring a more robust LHS design. Figure 3.17c shows an example of optimized LHS. However, finding the global optimimum of such problems is very challenging. For instance, a maximin LHS belongs to NP-hard problem [68]. As mentioned in the introduction, as the LHS

sampling belongs to the one-shot category, additional samples cannot be added without breaking properties of the LHS.

3.4.2.5 Low discrepancy sequences

The low discrepancy sequences belong to Quasi-Monte Carlo methods. They have been designed in order to substitute the random sampling by points coming from a deterministic sequence with a more uniform distribution.

Halton sequence The low-discrepancy Halton sequence is the generalization in multidimension of the one dimensional Van der Corput sequence [48]. This later builds a sequence of N points by using the decomposition of the index into the base of a prime number p. Let consider χ_k, the k-th scalar sample of the Van der Corput sequence, k is expanded over the base of the prime number p as follow :

$$k = \sum_{i=0}^{R(k)} a_i(k).p^i, \tag{3.115}$$

where $a_i(k)$ is the i-th digit of k in the base p and $R(k)$ is the maximum index corresponding to a non-zero $a_i(k)$. Finally, the quasi-random sample χ_k is given by:

$$\chi_k = \sum_{i=0}^{R(n)} \frac{a_i(n)}{p^{i+1}}. \tag{3.116}$$

Halton sequence generalizes this Van der Corput sequence in dimensions d, leading to the sequence $\boldsymbol{\chi}_k = [\boldsymbol{\chi}_{p_1}^{(1)}(k), \cdots, \boldsymbol{\chi}_{p_i}^{(i)}(k), \cdots, \boldsymbol{\chi}_{p_d}^{(d)}(k)]$ where $\chi_{p_i}^{(i)}(k)$ comes from the Van der Corput sequence where k is decomposed into the base of the prime number p_i. For example, in dimension 2, the k-th sample corresponds to the Van der Corput sequence in base 2 for the first dimension and in base 3 for the second dimension.

Faure sequence Faure sequence belongs to the low discrepancy sequence as well. Firstly, a prime number p is selected such as it is the smallest prime number $p \leq d$ with d the dimension of the input domain. The first dimension of the sequence is obtained with the Van der Corput sequence using this prime number p[48]:

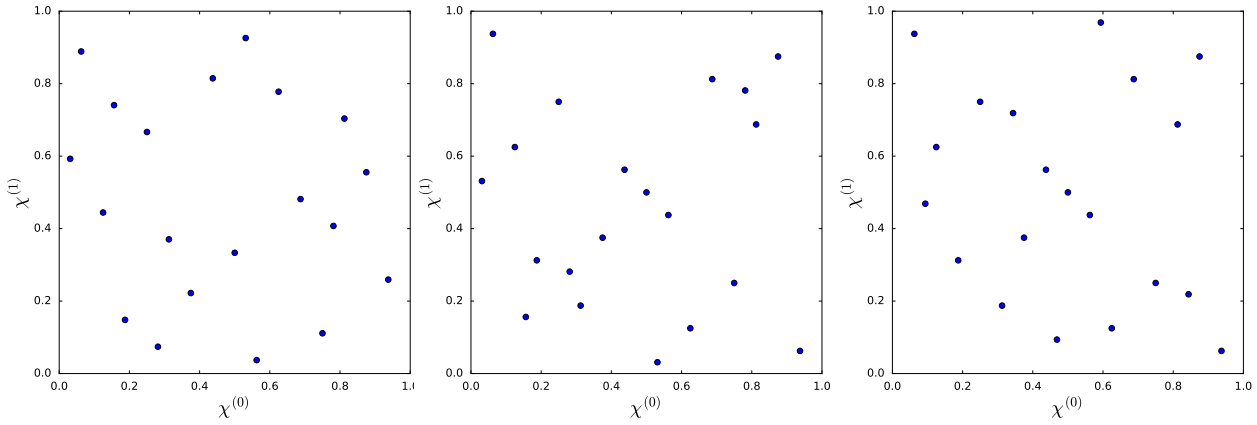
$$\boldsymbol{\chi}_k^{(1)} = \sum_{i=0}^{R(k)} \frac{a_i(k)}{p^{i+1}}. \tag{3.117}$$

The remaining components of $\boldsymbol{\chi}_k$ are built by recursion. The next $a_i^{(l)}(k)$ comes from a reordering of the previous $a_i^{(l-1)}(k)$ using binary coefficients modulo p :

$$a_i^{(l)}(k) = \sum_{j=i}^{R(k)} \binom{j}{i} a_i^{l-1}(k) \ mod \ p. \tag{3.118}$$

Once the new $a_i^{(l)}(k)$ has been computed, one can end up with the component l of the sample $\boldsymbol{\chi}_k$ by applying the Van der Corput sequence :

$$\boldsymbol{\chi}_k^{(l)} = \sum_{i=0}^{R(k)} \frac{a_i^{(l)}(k)}{p^{i+1}}. \tag{3.119}$$



(a) Halton sequence. $D = 0.0627$ (b) Faure sequence. $D = 0.0674$ (c) Sobol sequence. $D = 0.0479$

Figure 3.18: The first 20 samples of the two-dimensional Halton, Faure, and Sobol sequences.

Sobol sequence Sobol sequence is also a low-discrepancy sequence. First, a primitive polynom in the Galois field of two elements and of degree r is introduced to construct the k-th sample of the sequence [110]:

$$x^r + a_1 x^{r-1} + \ldots + a_{r-1} x + 1, \tag{3.120}$$

where a_i can be either 0 or 1. Previously an integer m has been chosen to define the upper bound 2^m for number of points in the sequence. Then arbitrary direction numbers v_j are chosen with $j \in [1, r]$ and $1 \leq \frac{v_j}{2^{m-j}} \leq 2^j - 1$ such as $\frac{v_j}{2^{m-j}}$ is an odd integer. This step is usually perform with quasi-random method in order to keep the deterministic property. The remaining directions are computed recursively:

$$v_j \doteq a_1 v_{j-1} \oplus a_2 v_{j-2} \oplus \ldots \oplus a_{r-1} v_{j-r+1} \oplus vj - r \oplus \frac{v_{j-r}}{2^r} \tag{3.121}$$

where $\oplus$ is the bit-by-bit exclusive-or operator. Finally, the l-th coordinate of the k-th sample is given by:

$$\boldsymbol{\chi}_k^{(l)} = \frac{\boldsymbol{\chi}_{k-1}^{(l)} 2^m \oplus v_c}{2^m}, \tag{3.122}$$

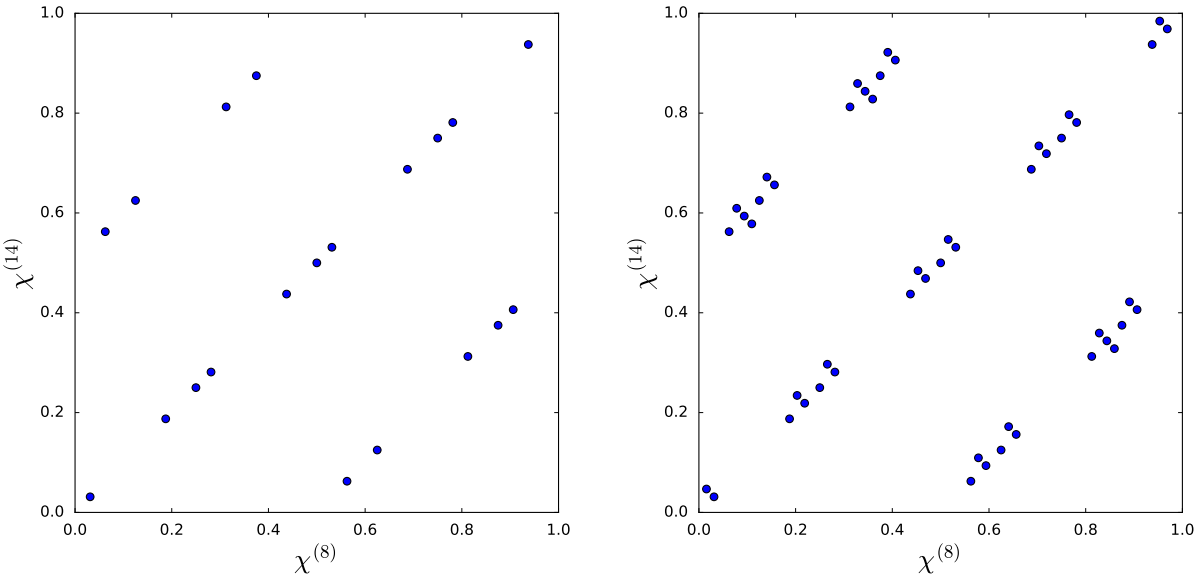
where $c \doteq min(j | b_j = 1)$ and b_j is the binary representation of l.

Main properties of the sequences The three low-discrepancy sequences presented above are the most communly used sequences for quasi-random methods. The DOE is performed in a deterministic and sequential manner by allowing to extend the number of points on purpose, which is not the case for stochastic methods where the number of points is given before the DOE generation. In addition, the infinite discrepancy of the sequences can be bounded by the following value [191]:

$$K \frac{(\log n)^s}{n} + \mathcal{O}\left(\frac{(\log n)^{s-1}}{n} \right). \tag{3.123}$$

The low discrepancy sequences are named after this property. Figure 3.18 illustrates the three sequences generating 20 samples. These examples indeed show low values for the discrepancy.

Nevertheless the low discrepancy sequences have several drawbacks. First of all, the upper bound of the discrepancy given by Equation 3.123 grows with the dimension. In addition, at a



(a) Projection with 20 samples

(b) Projection with 50 samples

Figure 3.19: High correlation in the 15-dimensional Sobol sequence. Subset of the 8-th and 14-th input dimensions.

given dimension, the bound increases between $n = 0$ and $n = e^s$. Therefore, the bound does not behave like the intuition stating that the discrepancy drops while the number of samples rises. For a parameter of size 7, the bound increases between 0 and 1096, which is significant if the experiments have an important computational cost. In addition, the low-discrepancy sequences shows another drawback in high dimensions. The uniformity of the sampling presents high irregularity if specific variables are considered. It means projections into low-dimensional subsets are highly correlated, as illustrated in Figure 3.19a for the 15-dimensional Sobol sequence. The subset given by the 8-th and 14-th dimensions shows a high correlation. The samples are clearly distributed only on three parallel lines, even if the number of samples is increases as for Figure 3.19b. This observation can be generalized for different types of sequences. For instance, Halton sequence is not recommanded to be used for dimensions higher than 6 or 7 due to internal cycle leading to poor space filling properties for low-dimensional projections [48].

A solution counterbalancing this problem randomizes the sequences. This approach is called scrambling [158] and helps to reduce the correlations between the different dimensions of the sequences. In addition, Faure [110] recommanded discarding the first (p^4-1) points to avoid clustered points around 0, even if it leads to the lost of some space-filing properties.

3.4.3 Adaptive designs

The adaptive designs, also called a posteriori designs, take advantage of the information provided by the surrogate model. Two types of designs can be distinguished: the model-based methods exploit the behavior of the model to find the optimal sampling for the specific model while response-based designs use only the outputs from the surrogate model to find additional samples.

3.4.3.1 Model-based

Model-based designs aims at improving the accuracy of the model by finding the design minimizing a metric associated with the uncertainty of the model. Let consider a GP model has been trained on a initial DOE $\boldsymbol{X_t}$ and the covariance matrix is accurately estimated. The model offers, by construction, the prediction variance σ, also called Mean Squared Error. This prediction variance can be seen as an estimation of the uncertainty given to the actual prediction. Various designs have been built from this quantity in order to reduce the uncertainty of the model.

The Integrated Mean Squared Error(IMSE) [175] measures the average accuracy of the surrogate model. The additional design points based on the IMSE criterion are determined as the points minimizing the integral of the Mean Squared Error over the parameter space:

$$\min_{\boldsymbol{\chi}\in\Gamma} \int_{\Gamma} MSE\left[\widetilde{a}\left(\boldsymbol{\chi}\right)\right] d\boldsymbol{\chi}. \tag{3.124}$$

The integral of the expression does not have an analytical form. It is usually approximated by Monte Carlo or trapezoidal methods. The IMSE criterion can be coupled with stochastic designs, such as the LHS, in order to optimize the space-filling.

Another criterion derived from the estimation of the variance is the maximum mean squared error (MMSE) [175]. It gives a measure of the risk to have large errors in the predictions. This criterion employs an approach similar to the minimax design. The new samples are chosen as the design minimizing the maximum of the mean squared error over the parameter space:

$$\min_{\boldsymbol{\chi}\in\Gamma} \left(\max_{\boldsymbol{\chi}'\in\Gamma} MSE\left[\widetilde{a}\left(\boldsymbol{\chi}'|\boldsymbol{X_t}\cup\{\boldsymbol{\chi}\}\right)\right] \right). \tag{3.125}$$

In addition to being computationally complex, the MMSE usually founds resampling point located near the boundary of the parameter space. Indeed, the variance of the GP model increases dramatically near the boundary of the parameter space if the model is in extrapolation. Other variance-based methods have been proposed, such as a combination of IMSE and MSE showing robust results and an improvement of the model accuracy. The method has been tested on a ensemble of 10 different functions [28].

Once the covariance matrix is fixed, the model-based design no longer depends on the value taken by the function of interest for new samples until the matrix is updated. Indeed, the new design points can be computed in one single batch, without updating the covariance matrix, or iteratively with small batchs. Thus, at each batch iteration the covariance matrix is updated, improving the relevance of the estimation variance. The size of the batch is a trade-off between the available time, the parallel computing capabilities, and the computational cost of the covariance matrix.

Other models may be used to build sampling criteria. For instance, a Bayesian approach of the Response Surface Method provides also an estimation variance. Therefore, optimal designs for the Response Surface Method can also be determined based on criteria close to the IMSE and MMSE.

In the context of the non-intrusive ROMs, a GP model is built on each of the M reduced coordinates. Hence, the variance-based criterion cannot be used directly as a new resampling point is found for each reduced coordinate. To the best of the author's knowledge, variance-based criterion have not been applied to ROMs.

3.4.3.2 Response-based designs.

Model-based designs are intrinsically linked to the a priori choice of the model. However, in the case of black-box functions nothing is known about the system and choosing a relevant model can be challenging. Thus, generic designs, called response-based, are proposed. They use only information from the output of the surrogate model to guide the sampling process. However, the idea is very close to the MSE criterion. A very common method uses a greedy approach by seeking the next sample $\boldsymbol{\chi}^*$ as the point maximizing an error criterion $\widetilde{e}$, such as:

$$\boldsymbol{\chi}^* = \arg\max_{\boldsymbol{\chi}\in\Gamma} \|\widetilde{e}(\boldsymbol{x},\boldsymbol{\chi})\| \,. \tag{3.126}$$

Jin et al. [105] proposes to build the error criterion by weighting the leave-one-out error e_{LOO} with the distance to the nearest training sample. The leave-one-out method is a simple method to estimate the quality of model. A surrogate model is built using all the training samples but one and the prediction of this surrogate is compared to the true value. This process is repeated for all training samples and an estimation of the error is obtained for each training sample. Details on the LOO are given in subsection 3.5.3. The weighting distance term allows to select a new samples not too close to the existing training samples. Thus, the final criterion is written as follows:

$$\boldsymbol{\chi}^* = \arg\max_{\boldsymbol{\chi}\in\Gamma} \left(\|e_{LOO}(\boldsymbol{x},\boldsymbol{\chi})\| \min_{\boldsymbol{\chi}'\in\boldsymbol{X_t}} d(\boldsymbol{\chi},\boldsymbol{\chi}') \right) . \tag{3.127}$$

3.4.3.3 Choosing sampling size and stopping criteria

Several rules of thumb have been developed in the surrogate model literature. The $10d$ rule [108, 133] is the the most known and most commonly used in the Kriging and GP community. The number of initial sample is simply fixed to 10 times the input dimension d. Other empirical rules have been established. The majority corresponds to a ratio of the input dimension or of the computational budget [131]. Naturally, these rules must be used as a general guideline and not as absolute rules.

An adaptive design starts usually with some samples of a space-filling DOE in order to explore the parameter space. The determination of the number of initial samples n_{init} affects the overall performance of the resampling. If n_{init} is too small, then the weaknesses of the surrogate model are not properly identified and the samples are distributed in regions of low interests. On the contrary, if n_{init} is too large, the part of adaptive sampling is low and, in the end, the parameter space is covered only by a classical space-filling method while the computational budget could have been spent more effectively. There are no rules to find this tradeoff. Usually, the choice is driven by a carefull analysis of the snapshots computed sequentially.

As regards the stopping criterion, the more obvious choice sets a maximum computational budget that cannot be exceeded. Other criteria based on the estimation of modeling error can be used. For instance, an absolute threshold can be estimated and compared to the leave-one-out error. In the same way, a relative reduction of the leave-one-out error can stop the process if the variation between several iterations is very small.

3.4.4 Summary

This section has presented various strategies of DOE. Figure 3.20 classifies them based on their belonging to a model-free or an adaptive designs. Generally, the a priori estimation of the number of

training samples appears complex, especially for aerodynamic surrogate models involving nonlinear phenomena. Thus, sequential mode-free designs are preferred in this work for the exploration phase, in particular low-discrepancy sequences such as Halton and Sobol sequences.

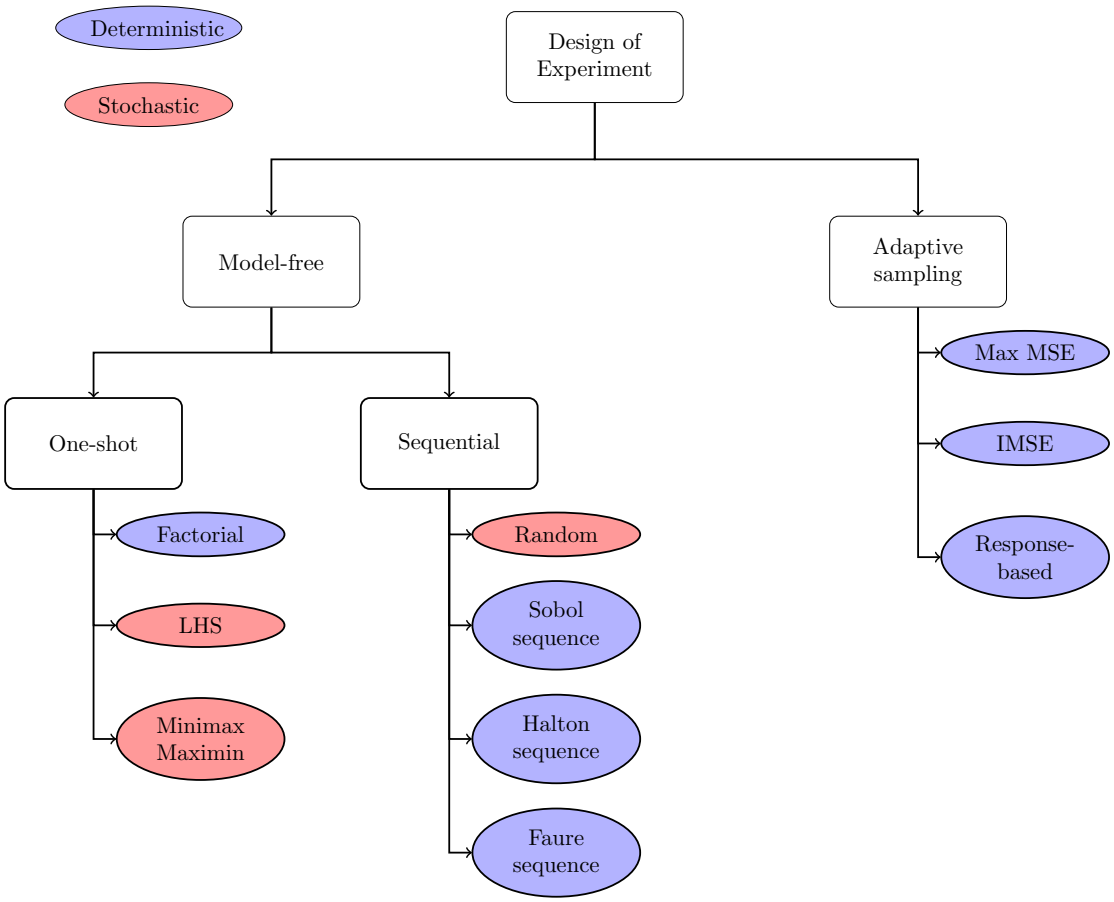


Figure 3.20: Classification of the DOE.

3.5 Validation of the model

Once the surrogate model is trained, its quality can be evaluated. Different methods have been developed. Some rely only training data while other methods require to compare the predictions with additional data. This section gives an overview of the different error metrics and discusses the most common approaches to validate a surrogate model.

3.5.1 Error metric

The so-called predictivity coefficient Q_2 provides a measure of the model quality, based on the ratio of the output variance which is explained by the metamodel. It can be interpreted as the classical coefficient of determination of a linear regression applied to a test sample [101]. The more the value

is close to 1, the higher variance is explained by the model. It is expressed as:

$$Q_2^{(i)} = 1 - \frac{\sum_{j=1}^{m} [f^{(i)}(\boldsymbol{\chi}_{p_j}) - \widetilde{f}^{(i)}(\boldsymbol{\chi}_{p_j})]^2}{\sum_{j=1}^{m} [\overline{f}^{(i)} - f^{(i)}(\boldsymbol{\chi}_{p_j})]^2}. \tag{3.128}$$

The value of the predictivity coefficient $Q2$ is strongly related to the value of the output variance. Thus, for very small variances the coefficient can tends to negative values, even if the predictions are close to the target values. On the contrary, if the variance of the quantity of interest is very large, the coefficient can show values close to 1 while the error is still important in term of different between the prediction and the target values. For these reason, three other metrics are introduced.

The so-called Root Mean Square Error (RMSE) and the Normalized Root Mean Square Error (NRMSE) are frequently used measure of the differences between the prediction values of a model and the target values. Their expressions are given by:

$$RMSE^{(i)} = \sqrt{\frac{1}{m} \sum_{j=1}^{m} [f^{(i)}(\boldsymbol{\chi}_{p_j}) - \widetilde{f}^{(i)}(\boldsymbol{\chi}_{p_j})]^2} \tag{3.129}$$

$$NRMSE^{(i)} = \frac{RMSE^{(i)}}{f_{max} - f_{min}}, \tag{3.130}$$

where f_{max} and f_{min} refer respectively to the scalar maximum and minimum values taken by the function to predict $\boldsymbol{f}$ for all the snapshots and over the mesh:

$$\begin{aligned} f_{max} &= \max(\|\boldsymbol{f}(\boldsymbol{\chi}_{p_1})\|_\infty, \cdots, \|\boldsymbol{f}(\boldsymbol{\chi}_{p_m})\|_\infty), \\ f_{min} &= \min(\|\boldsymbol{f}(\boldsymbol{\chi}_{p_1})\|_\infty, \cdots, \|\boldsymbol{f}(\boldsymbol{\chi}_{p_m})\|_\infty). \end{aligned} \tag{3.131}$$

These two metrics are related to the square root of the variance. Indeed, for an unbiased statistical estimator, the RMSE corresponds to the standard deviation.

One can note that the $Q_2^{(i)}$ and $NRMSE^{(i)}$ are computed at a given index i of the spatial domain Ω. The global measures are provided by averaging the quantities over this domain. The global quantities are referred to $\langle Q_2 \rangle_\Omega$ and $\langle NRMSE \rangle_\Omega$, with $\langle . \rangle_\Omega$ the spatial average operator. However, in order to provide also a statistical error analysis, the averaged normalized error E_i is introduced. It corresponds to the absolute error between the exact value and the prediction, normalized by the range of variation, at snapshot level j:

$$E_j = \frac{\langle |\boldsymbol{f}(\boldsymbol{\chi}_{p_j}) - \widetilde{\boldsymbol{f}}(\boldsymbol{\chi}_{p_j})| \rangle_\Gamma}{f_{max} - f_{min}}, \ \forall j \in [1, m]. \tag{3.132}$$

The statistical distribution of E_i is presented with a box plot formalism, illustrated in Figure 3.21. A box plot groups the data through different quantiles: the bottom and the top of the box represent respectively the value of the first and third quartiles Q1 and Q3, whereas the horizontal line inside the box is the median (second quartile) and the diamond the mean. The distance between the two quartiles Q1 and Q3 denotes the interquantile range of the data (IQR). The first vertical line indicates minimum value equal to Q1−1.5 IQR and the last vertical line gives the extreme value associated with Q3+1.5 IQR. Finally the outliers are plotted as crosses or dots.

Besides the direct assessment of the surrogate model using a wide validation test, an estimation of the error can be provided only with the training set. Cross validation strategies, such as the leave-one out error, or the variance given by the Gaussian Processes can estimate the a priori quality of each prediction.

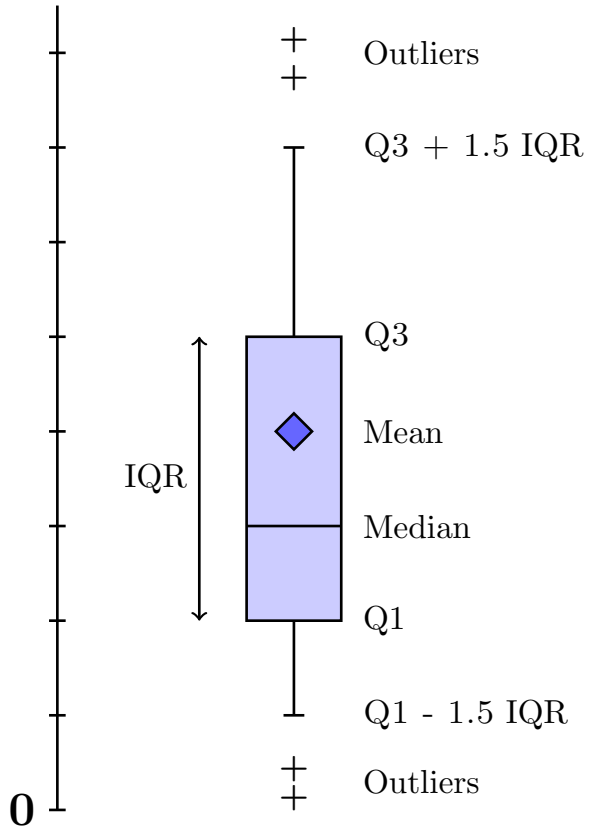


Figure 3.21: Illustration the box plot representation.

3.5.2 Split sampling

The split sampling method, also called hold-out approach, is the simplest way to evaluate the quality of a surrogate model. The data are divided into a training and a test set. The training set is used to build the surrogate model. Then the latter predicts the quantities of interest on the test set, given an unbiased estimation of the generalization error of the surrogate model. The generalization error measures the capacity of the model to predict values for unseen input data.

The split sampling approach has two main drawbacks. Firstly, the test set is not used to train the surrogate model, therefore the computational cost of the process is increased while the accuracy of the model is not improved. Then, the final estimation can exhibit a high variance if the points of the test set are not chosen uniformly in the parameter space. In many applications, however, the computational budget is very limited and the possible test sets are small. Consequently, the estimation of the predictive performance appears very noisy. An alternative consist in using as much as possible all the available data to train the model and to estimate the predictive performance.

3.5.3 Cross-validation

The cross-validation method, also called K-folds method, is a very popular approach to estimate the generalization error by using only the training set. Hence the surrogate model is trained with all the available data. The idea is to split the training set into k disjoint subsets of approximately equal size. A subset is chosen as the testing set and the remaining subsets form the training set. The model is trained on this training set and evaluated on the testing set. This process is repeated

k times and the final error is computed as the average of the error one each subset. Figure 3.22 gives an example of a 5-fold method. The model is trained on a group of 4 subsets and then is evaluated on the remaining test set. The procedure is repeated for the 5 folds and the generalization error is averaged.

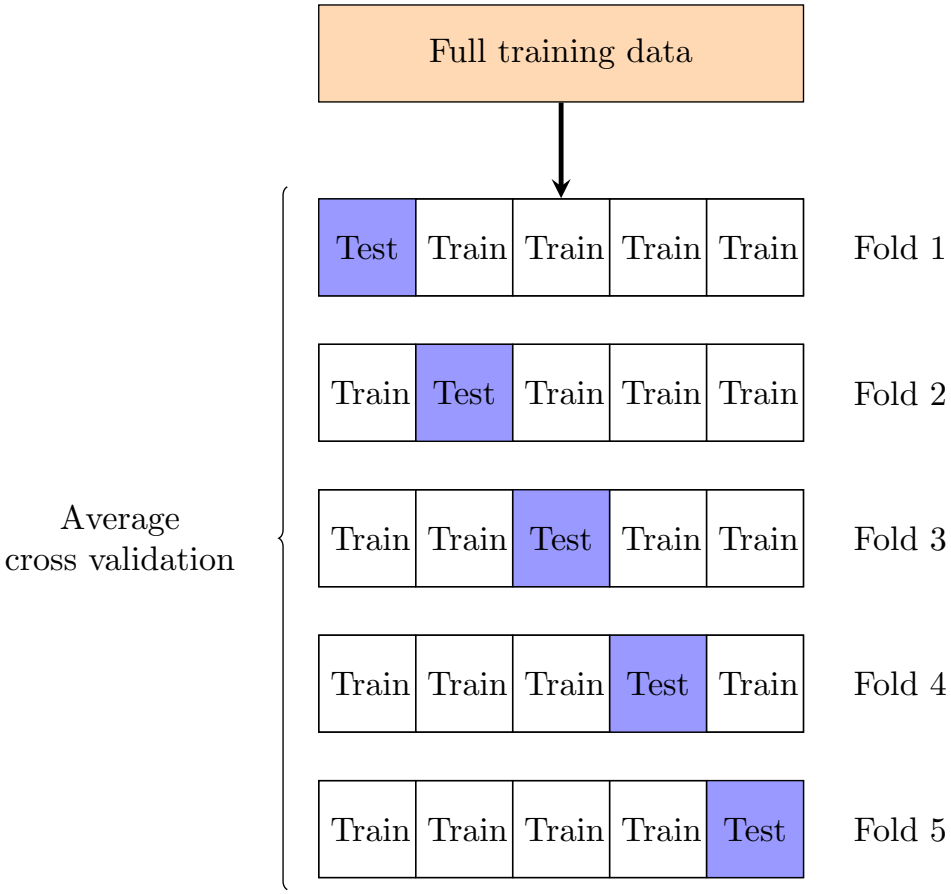


Figure 3.22: Illustration of a 5-fold cross validation.

A very specific case occurs when k is chosen equal to n. This case is called leave-one-out cross validation. Let consider $\widetilde{f}_{-i}$ corresponds to the surrogate model based on the training sample $\boldsymbol{X_t}$ excluding $\boldsymbol{\chi}_i$. The leave-one-out error $\boldsymbol{e_{LOO}}$ is written as:

$$e_{LOO} = \frac{1}{n}\sum_{i=1}^{n}\left\|\boldsymbol{f}(\boldsymbol{\chi}_i) - \widetilde{f}_{-i}(\boldsymbol{\chi}_i)\right\|^2. \tag{3.133}$$

From this expression, the leave-one-out predictivity coefficient Q_2 can be derived:

$$Q_2 = 1 - \frac{e_{loo}}{\widetilde{\sigma}^2}, \tag{3.134}$$

and the leave-one-out NRMSE:

$$NRMSE = \frac{e_{LOO}}{f_{max} - f_{min}}. \tag{3.135}$$

The leave-one-out method requires an additional computational endeavor. In the case of the ROMs, $n \times M$ models must be trained.

In conclusion, the parameter k plays a central role in the cross-validation method. The generalization error is predicted at low variance but high bias for high values of K, while the leave-one-out

method shows low bias but a higher variance [78]. Usually, leave-one-out method is widely used in the literature and typical values of K between 5 or 10 represent a good compromise for the K-fold method [78].

3.5.4 Estimated error by Gaussian Processes

The estimation of the error can be provided by expanding the Gaussian Process results into the POD decomposition. Taking the variance of Equation 3.55 yields:

$$\begin{aligned} Var(\widetilde{\boldsymbol{f}}(\boldsymbol{\chi})) &= Var\left(\sum_{k=1}^{M} \widetilde{a_k}(\boldsymbol{\chi})\boldsymbol{\phi}_k\right) \\ &= \sum_{k=1}^{M} Var\left(\widetilde{a_k}(\boldsymbol{\chi})\right)\boldsymbol{\phi}_k \circ \boldsymbol{\phi}_k \\ &= \sum_{k=1}^{M} \widetilde{\sigma_k}^2(\boldsymbol{\chi})\boldsymbol{\phi}_k \circ \boldsymbol{\phi}_k, \ \forall \boldsymbol{\chi} \in \mathbb{R}^p, \end{aligned} \tag{3.136}$$

where $\circ$ is the Hadamard product and $\widetilde{\sigma_k}$ the estimation by GP of the standard deviation of the k-th reduced coordinate, given in Equation 3.101. Thus, for each prediction, an estimation of the standard deviation $\boldsymbol{\sigma}_{\widetilde{\boldsymbol{f}}}$ is provided for the quantity of interest:

$$\boldsymbol{\sigma}_{\widetilde{\boldsymbol{f}}(\boldsymbol{\chi})} = \sqrt{\sum_{k=1}^{M} \widetilde{\sigma_k}^2(\boldsymbol{\chi})\boldsymbol{\phi}_k \circ \boldsymbol{\phi}_k}, \ \forall \boldsymbol{\chi} \in \mathbb{R}^p. \tag{3.137}$$

3.6 Summary

3.6.1 Technical choices

Building non-intrusive ROMs needs many various mathematical tools. Figure 3.2 summarized the different steps involved in the process of ROMs. The following particular methods have been chosen in this work based on the specific characteristics of aerodynamic and aerothermal problems:

- DOE: low-discrepancy sequences are preferred due to their sequential and correct space-filling properties. In particular, Halton sequences are used for training purpose and Sobol sequences for validation;
- dimension reduction: POD is a robust method widely used in the literature. It allows to reduce the information by expending the quantities of interest into well-chosen basis functions. Therefore, POD is used to reduce the dimension of the problem;
- regression: GPR is chosen as it handles problems with nonlinear features and can estimate the error for each prediction. Moreover, many performant packages have been developed to build GPR models [61];
- validation: LOO strategies give a first estimation of the model error using only the training set. Thus, they are used in the a-priori design and training of the surrogate model. Then, a significant test set is systematically built in order to validate the model.

This classical non-intrusive model is implemented and used for comparison purposes in Chapter 4 and Chapter 5 and is put in place in Chapter 6 to build the aerothermal surrogate model. Details on the numerical implementation can be found in the following part.

3.6.2 Numerical implementation

3.6.2.1 JPOD

Jack Proper Orthogonal Decomposition (JPOD) starts in 2007 at Cerfacs [21, 22]. The code was developed during 5 years where tests and validations have been performed through FP7 projects such as the European SimSAC and ALEF. These developments and results were used as a starting point for the work realized in this thesis: a full workflow capable of managing CFD simulations with some POD, DOE, and interpolation features.

In parallel of this work, the UQ team of Cerfacs has refactored the whole code and has added uncertainty quantification capabilities leading a new name: BATMAN, standing for Bayesian Analysis Tool for Modeling and uncertAinty quaNtification [173, 174].

3.6.2.2 scikit-learn library

Scikit-learn [160] is a free Python library dedicated to machine learning. It plays a central role in this thesis. First, scikit-learn is employed to generate the GPR models. The library shows good performances [61] compared to other libraries such as DACE [135] or DiceKriging [172].

In addition, scikit-learn offers also supervised and unsupervised learning capabilities, dimension reduction methods, and a very general API. In particular, the algorithms of GPC, DBSCAN, K-Means, and GMM from scikit-learn are used in the thesis.

3.6.2.3 Other tools

The Python library Numpy [190] has been used widely to manipulate matrices and to build numerical algorithm. The Matplotlib library [99] has been easily coupled with the Numpy arrays for the visualization of all the data. Finally, the Openturns library [12] has provided various DOE methods.

CHAPTER 4

Local surrogate models using Machine Learning

Résumé

La méthode de décomposition locale, appelée Local Decomposition Method (LDM) dans le manuscrit, propose d'adapter les modèles réduits détaillés dans le chapitre précédent à l'aérodynamique externe. L'objectif est de les rendre plus performants dans la prédiction d'écoulements transsoniques mélangeant plusieurs régimes physiques et présentant des chocs. La méthode reprend l'idée de la décomposition de l'espace des paramètres. Elle se compose en cinq étapes :

- **l'application d'un filtre** sur la base d'entrainement ;
- **le partitionnement des solutions** via un algorithme d'apprentissage non supervisé et la construction des modèles réduits locaux ;
- **la séparation de l'espace des paramètres** en plusieurs sous-ensembles ;
- **le rééchantillonnage** en tirant profit des particularités de chaque ensemble ;
- **La reconstruction d'un modèle global** en minimisant les erreurs aux interfaces.

Un senseur de choc de type Jameson est utilisé afin de mieux identifier les discontinuités. En effet, il permet d'améliorer les résultats de la seconde étape qui consiste à diviser la base d'entrainement en différents groupes homogènes. Les méthodes d'apprentissage non supervisé utilisées sont issues de l'apprentissage automatique et reposent principalement sur des mesures de distances entres les champs calculés. En calculant les distances directement sur le senseur de choc, le partitionnement est rendu plus pertinent du point de vue aérodynamique en se concentrant sur la présence ou non de discontinuités et de forts gradients. Une fois le partitionnement réalisé, les modèles réduits locaux sont entrainés sur chaque sous ensemble de la base d'entrainement.

Ensuite, l'espace des paramètres est séparé en plusieurs blocs avec un algorithme de classification issu aussi de l'apprentissage automatique. Ainsi, il est possible de prédire le bloc d'appartenance de chaque nouvelle prédiction et de lui associer le modèle réduit local correspondant. De plus, grâce à cette classification, des stratégies particulières de rééchantillonnage sont mises en place. En effet, il est possible d'analyser localement les propriétés de chaque sous ensemble et d'enrichir spécifiquement certains blocs de l'espace des paramètres avec de nouvelles simulations. Ainsi le budget de calcul est utilisé de manière optimale en privilégiant les régions de l'espace des paramètres qui semblent poser le plus de problèmes aux modèles réduits.

Enfin, l'ensemble des modèles réduits locaux sont assemblé afin de générer un modèle global capable de prédire des champs sur l'ensemble de l'espace des paramètres. La gestion des interfaces entre ces modèles est bien entendu un enjeu important de la méthode. Deux stratégies classiques peuvent être utilisées : soit chaque modèle contribue à la prédiction uniquement lorsqu'elle est localisée dans son sous ensemble soit la prédiction est formée par une pondération des modèles. Cette dernière approche a l'inconvénient de mélanger potentiellement plusieurs régimes physiques, ce qui peut être dommageable au modèle.

La méthode de décomposition locale a pu être validée sur deux cas académiques : l'équation de Burgers représentant le déplacement d'un choc et l'écoulement turbulent autour d'un profil bidimensionnel RAE2822. Elle a identifié deux régimes physiques associés à la présence ou non de chocs. Le partitionnement de ces deux régimes a été largement facilité par l'utilisation du senseur de choc qui a permis de séparer les écoulements de nature différente. Ainsi, les résultats ont été grandement améliorés, en particulier dans la région avec les forts gradients et les discontinuités.

The Local Decomposition Method (LDM), developed in this Ph.D. thesis, extends the classical POD/interpolation reduced-order modeling by splitting the parameter space, inspired by the parameter space decomposition presented in Chapter 2, in particular the dynamic local reduced-order modeling of Amsallem et al. [7] and Washabaugh et al. [202]. This section has been partially published in a journal paper [54].

LDM aims at improving the predictions of surrogate models for quantities of interest with very different behaviors within the input parameter space. Instead of a unique global POD basis, several local bases are determined using machine learning tools yielding to more flexible behaviors bringing out a precise delimitation of the physical regimes.

This method has mainly been developed in order to treat external aerodynamics with flow separations and a mix between the subsonic and transonic regimes. However, it could be envisioned to generalize the LDM to other problems characterized by bifurcating solutions. By this way, the LDM is expected to reduce the computation budget needed to build the training samples. In addition, separating the snapshots into several cluster can help to improve the physical understanding from numerical simulations.

4.1 Presentation of the Local Decomposition Method

The method proposed here creates local subspaces of the parameter space by combining a physical sensor with machine learning tools. The physical sensor is a central element of the method to achieve proper separations of physical regimes by computing specific features for all snapshots, in order to increase the quality of the clustering. Indeed, the conversion of the vector-valued output

into a vector of physical-based features gives the possibility to cluster the snapshots into subsets with the same physical behavior. Thus the POD vectors are more representative of the physics.

A specific shock sensor is used for problems mixing subsonic and transonic conditions. It measures the nonlinearities and sharp gradients of the flowfield. As the different phenomena are no more mixed in the POD basis, building a local reduced-order model on each of these subsets achieves a better consideration of the physical regimes. Clustering snapshots provides also a greater flexibility to the data fit model that can behave independently on each subgroup.

As regard the prediction of untried sets of parameters, the parameter space is divided in several domains by a classifier according to the clustering of the snapshots. Therefore each region of the parameter space is associated with a local reduced-order model and its respective subspace, allowing to classify the input parameter space to the right physical regime. Finally, resampling is carried out by identifying the subspaces with the highest entropy. Extra snapshots are added in these specific subspaces with the objective to minimize the redundancy of the sampling, thus increasing the accuracy of the surrogate model. Figure 4.1 sketches these first steps of the process. The

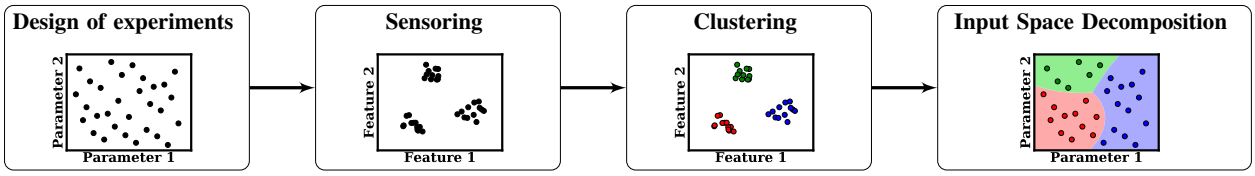


Figure 4.1: Sketch of the LDM.

specificity of the presented method includes, besides the introduction of a feature extraction with a shock sensor, a novel resampling strategy and the application to an aerodynamic case.

Figure 4.2 illustrates the whole method. All notations are introduced in the following sections which give details on the different steps of the LDM, namely: clustering, the parameter space decomposition, the proposed resampling strategy, and the recombination of local models into a global one.

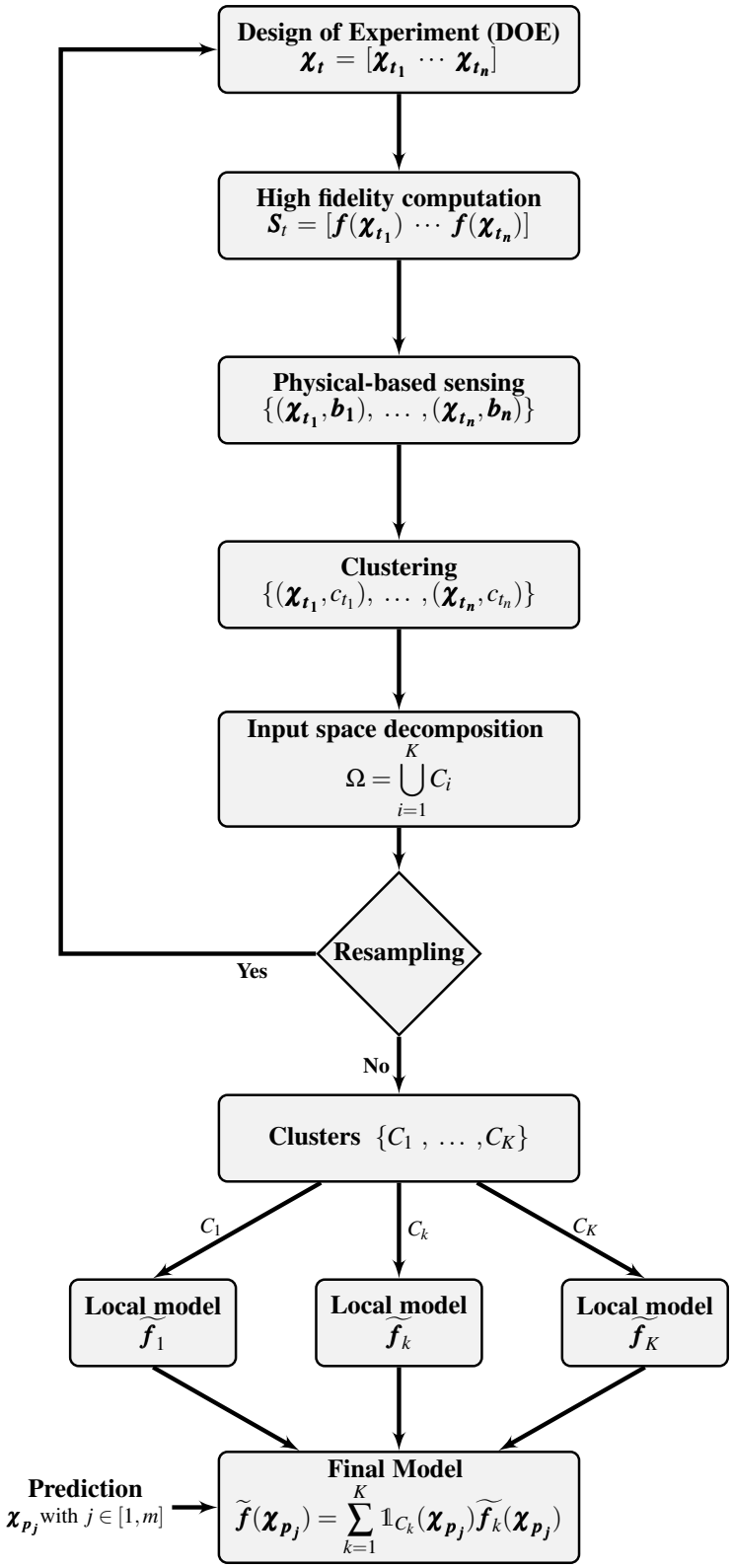


Figure 4.2: Flowchart of the LDM.

4.2 Separation of the snapshots

4.2.1 Physical-based shock sensor detecting flow regimes

Choosing quantity characterizing the physical regimes, on which the clustering is performed, is a question of central importance impacting the quality of the classification. Unsupervised learning algorithms are used directly on quantities of interest to build groups with patterns of small differences in ROMs [7, 202, 210]. In this thesis, the objective is to separate physical regimes. Applying the algorithms on the quantity of interest may lead to clustering error, as two fields of quantity of interest can have large differences even though they belong to the same physical regime. For this reason, this section proposes to enhance clustering with a filter. Based on a physical approach, a mathematical transformation converts quantities of interest into a sensor of the physical regime. The main goal is to sharply differentiate physical regimes to ease the clustering of snapshots. Somewhat similar methods can be found in Lorente et al. [136] or Constantine and Iaccarino [39], where the gradient is used to identify parts of the signal which can be improved by a specific correction.

External aerodynamics with subsonic and transonic regimes represents one of the main application in this thesis. These flows are characterized by shock waves. Thus, a straightforward idea is to consider a shock sensor that is able to detect large changes in the variation of the quantity of interest, such as the Jameson's Shock sensor [103]. The latter has been developed as a heuristic method for non-oscillatory shocks capturing. It is related to the second order derivative of the pressure. However, quantities of interest are not limited to pressure signals and a more general expression is introduced as:

$$\nu_i = \frac{|s_{i-1} - 2s_i + s_{i+1}|}{\epsilon_0 + |s_{i-1}| + 2|s_i| + |s_{i+1}|}, \text{ with } i \text{ such that } \boldsymbol{x_i} \in \Omega \backslash \partial\Omega, \tag{4.1}$$

where $\nu_i \in \mathbb{R}^{d'}$ is the generalized sensor, d' the dimension of $\Omega\backslash\partial\Omega$, ϵ_0 a constant avoiding division by 0, s_{i-1}, s_i, and s_{i+1} the quantities of interest at respectively node $i-1$, i and $i+1$ in the direction of the external flow. Other shock sensors could be used [156].

4.2.2 Clustering methods

Once a filter has been applied on data, structures must be discovered in order to separate the snapshots in function of the different flow shapes. The clustering identifies inherent groups in the input data based on similarity measures. It has been applied in various fields of application, such as image segmentation or data compression [18]. Several types of clustering may be distinguished, such as centroid models, probability distribution models, density models, hierarchical clustering, etc...

Clustering in high-dimension represents a very challenging problem related to the curse of dimensionality [13]. Many different transformations have been developed to find embedded non-linear manifolds within high-dimensional space, such as Locally linear embedding or Isomap methods [18]. However, a classical Principal Component Analysis (PCA) has been applied on sensor data, mainly due to its simplicity and robustness compared to more complex nonlinear dimensionality reduction methods. The percentage of variance is set to 99% and resulting subspaces from this transformation are grouped in matrix $\boldsymbol{B}$ such as:

$$\boldsymbol{B} = [\boldsymbol{b}_1 \ \cdots \ \boldsymbol{b}_n]^T, \ \boldsymbol{b_i} \in \mathbb{R}^{M'}, \tag{4.2}$$

where M' is the dimension of the cut-off in the PCA associated with 99% variance.

Three fundamental clustering methods are described in this section: K-means, Gaussian Mixture Model (GMM), and Density-Based Spatial Clustering of Applications with Noise (DBSCAN). Interesting readers can refer to Bishop [18] and Friedman et al. [78] for a more in-depth presentation of other methods.

If there are K clusters, a training set is built such as:

$$\{(\boldsymbol{\chi}_{t_1}, c_{t_1}),\ \dots\ , (\boldsymbol{\chi}_{t_n}, c_{t_n})\} \text{ with } c_{t_i} \in [1, \cdots, K],\ \forall i \in [1,\ n], \tag{4.3}$$

where c_{t_i} is the target variable of the cluster. This training set is used to train a supervised learning algorithm, described in the following section, in order to link the input parameter of each quantity $\boldsymbol{B}_i$ with a class of physical regime.

4.2.2.1 K-Means

The K-means algorithm is a very popular, simple, and easy to implement method. If the data $\boldsymbol{B}$ are splitted into K clusters, each cluster k is defined by a center $\boldsymbol{\mu_k} \in \mathbb{R}^{M'}$ where $k \in \{1, \cdots, K\}$. The squared Euclidean distance is chosen as a similarity metric. The goal of the K-means method is to found clusters minimizing the distortion measure J defined by the squares of the distance of each point to the closest cluster center $\boldsymbol{\mu_k}$:

$$J = \sum_{i=1}^{N} \sum_{k=1}^{K} r_{ik} \left\| \boldsymbol{b_i} - \boldsymbol{\mu_k} \right\|^2, \tag{4.4}$$

where $r_{ik} \in \{0, 1\}$ is the binary indicator variable. It is equal to 1 if $\boldsymbol{b_i}$ belongs to the cluster k and is equal to 0 otherwise. The function J can also be seen as an inertia where $\boldsymbol{\mu_k}$ corresponds to the center of mass of the cluster k.

The K-means algorithm consists in several steps. First, centroids are initialized with random values or more sophisticated methods. For instance, the K-means++ chooses the first center at random, then the probability of choosing the next center is weighted by the distance to the first center. It is repeated until k centers have been chosen. Then, a two-step iterative process, referred to as Lloyd's algorithm, is usually employed to find the minimum of J:

- The **assignment step** determines the new values given to the binary indicator variables r_{ik}. If the centers are considered as fixed, the distortion measure J is a linear function of r_{ik}. Consequently, terms of the sum can be minimized independently for each point in the data set, such as the new r_{ik} is chosen as:

$$r_{ik} = \begin{cases} 1, & \text{if } k = \underset{k' \in \{1, \cdots, K\}}{\arg\min} \left\| \boldsymbol{b_i} - \boldsymbol{\mu_{k'}} \right\|^2 \\ 0, & \text{otherwise} \end{cases}. \tag{4.5}$$

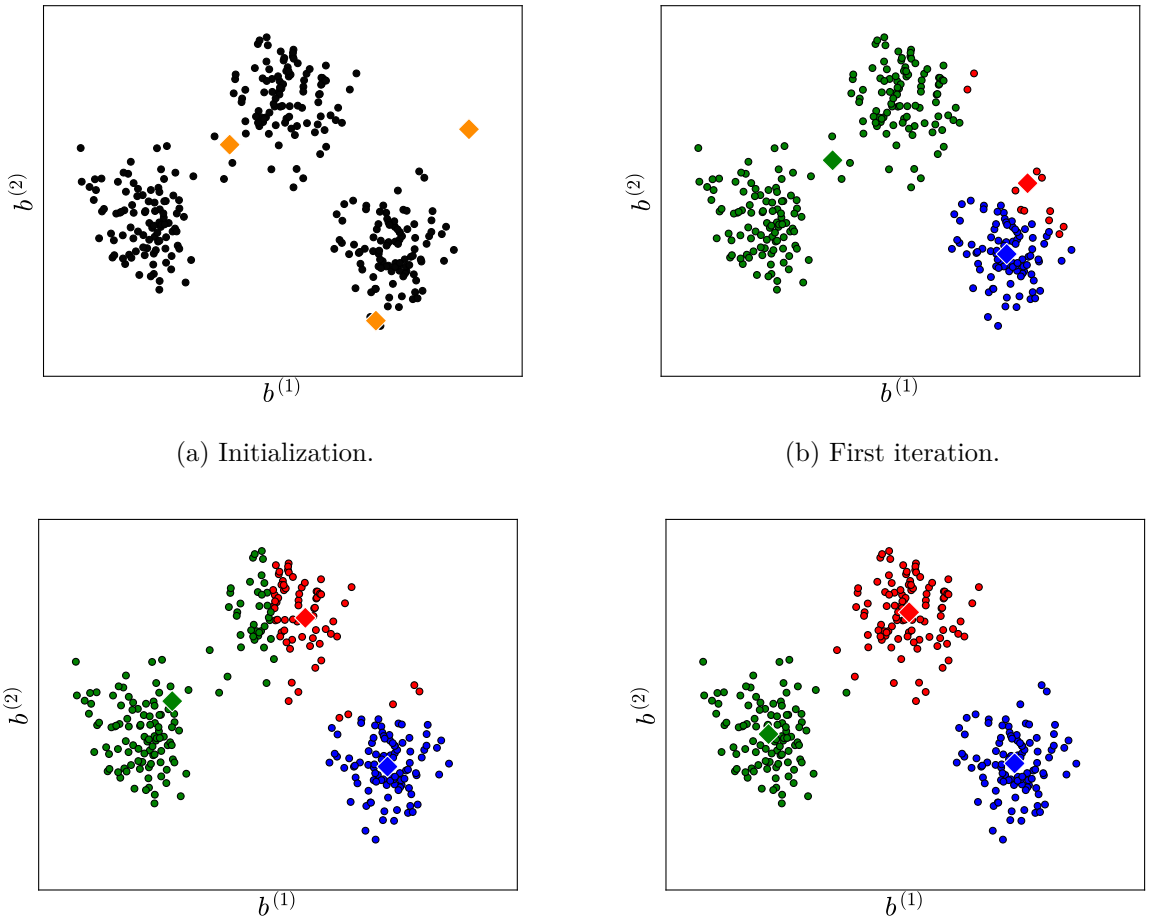
  The expression can be interpreted easily: each point is assigned to its closest centroid.

- The **update step** minimizes the value of J which is a quadratic function of $\boldsymbol{\mu_k}$ with the r_{ik} held fixed. If the derivative of J is set to zero, the new centroids are expressed as:

$$\boldsymbol{\mu_k} = \frac{\sum_{k=1}^{K} r_{ik} \boldsymbol{b_i}}{\sum_{k=1}^{K} r_{ik}}. \tag{4.6}$$

Once again, an interpretation can be given to the above expression: the new centroid $\boldsymbol{\mu_k}$ is simply the mean of all the data assigned to the cluster k.

The two-step process is stopped when the difference between old and new centroids is below a given threshold. Even if the K-means algorithm always converges [18], the function J is generally non-convex and minimums may be local. For this reason, the K-means algorithm needs to be run with different initializations and the best local minimum is selected.



(a) Initialization.

(b) First iteration.

(c) Second iteration.

(d) Third iteration.

Figure 4.3: K-means illustration.

Figure 4.3 illustrates the K-means algorithm. Initial data are generated from isotropic Gaussians. First, centroids are initialized randomly. Then, the algorithm starts to process. As the initial data are well separated, the algorithm quickly converges to the true clusters after only 3 iterations.

A variety of other algorithms has been built from the K-means method. One of the most popular is called fuzzy K-means. Instead of having each data point assigned to one single cluster, the data points belong to all clusters to varying degrees. The classical K-Means problem is referred as hard clustering, whereas fuzzy K-means is called soft clustering. In addition, other metrics can be chosen to improve the clustering with groups of different sizes. These methods are called K-medoids clustering [78].

4.2.2.2 Gaussian Mixture Model

Let us assume the training set $\boldsymbol{B}$, characterizing the physical regimes, comes from K clusters $C_1, \cdots, C_K$. Each cluster C_k follows a probability distribution of parameter $\boldsymbol{\theta_k}$ and proportion w_k, regrouped for all the clusters in a mixture parameter $\boldsymbol{\Phi} = [w_1, \cdots, w_q, \boldsymbol{\theta_1}, \cdots, \boldsymbol{\theta_K}]$. GMM consists of modeling $\boldsymbol{B}$ with a mixture distribution of multivariate normal distributions g. Each one is associated with the cluster C_k such that the probability density function of $\boldsymbol{b_i}$ is given by:

$$p(\boldsymbol{b_i}|\boldsymbol{\Phi}) = \sum_{k=1}^{K} w_k \, g(\boldsymbol{B_i}|\boldsymbol{\theta_k}), \ \forall i \in [1, \cdots, n], \tag{4.7}$$

where the mixture weight w_k represents the probability that the observation comes from a Gaussian distribution defined by θ_k giving mean and covariance of the multivariate normal distribution g. A K-dimensional binary random latent variable $\boldsymbol{z} \in \mathbb{R}^K$ is introduced such as a particular component z_k of $\boldsymbol{z}$ is equal to 1 and all other elements are equal to 0, meaning that z_k satisfies $z_k \in \{0, 1\}$ and $\sum_{k=1}^{K} z_k = 1$. In particular, the k-component is 1 if the training sample $\boldsymbol{b_i}$ is generated from the k-th Gaussian such as the marginal distribution over $\boldsymbol{z}$ is defined as $p(z_k = 1) = w_k$.

These mixture parameters regrouped in Φ are estimated iteratively using an Expectation Maximization algorithm (EM algorithm) [50]. The expectation of the posterior probability γ_k of belonging to the cluster k can be expressed with Bayes' theorem (E-step):

$$\gamma_k(\boldsymbol{B_i}) = p(C_k|\boldsymbol{B}_i) = \frac{p(\boldsymbol{B}_i|C_k)\, p(C_k)}{P(\boldsymbol{B}_i)} = \frac{w_k \, g(\boldsymbol{B}_i|\theta_k)}{\sum\limits_{l=1}^{q} w_l \, g(\boldsymbol{B}_i|\theta_l)}, \ \forall k \in [1, \cdots, q], \ \forall i \in [1, \cdots, n]. \tag{4.8}$$

Then, the mixture parameters can be re-estimated (M-step):

$$\mu_k = \frac{\sum\limits_{i=1}^{N} \gamma_k(\boldsymbol{B}_i).\boldsymbol{B}_i}{\sum\limits_{i=1}^{N} \gamma_k(\boldsymbol{B}_i)}, \ \forall k \in [1, \cdots, K], \tag{4.9}$$

$$\Sigma_k = \frac{\sum\limits_{i=1}^{N} \gamma_k(\boldsymbol{B}_i).(\boldsymbol{B}_i - \mu_k).(\boldsymbol{B}_i - \mu_k)^T}{\sum\limits_{i=1}^{N} \gamma_k(\boldsymbol{B}_i)}, \ \forall k \in [1, \cdots, K], \tag{4.10}$$

$$w_k = \frac{1}{n} \sum_{i=1}^{N} \gamma_k(\boldsymbol{B}_i), \ \forall k \in [1, \cdots, K]. \tag{4.11}$$

These two steps are iterated until the converge of the log likelihood:

$$ln\left(p(\boldsymbol{B}|\boldsymbol{\Phi})\right) = \sum_{i=1}^{n} ln\left(p(\boldsymbol{b_i}|\boldsymbol{\Phi})\right). \tag{4.12}$$

The cluster of each quantity $\boldsymbol{B}_i$ is determined using the previous probability expression. A hard splitting is applied such as the target variable c_{t_i} for each training input $\boldsymbol{\chi_{t_i}}$ is given by:

$$c_{t_i} = \left\{ k \mid \max_{k \in [0,K]} p(C_k|\boldsymbol{B}_i) \right\}, \ \forall i \in [1, \cdots, n]. \tag{4.13}$$

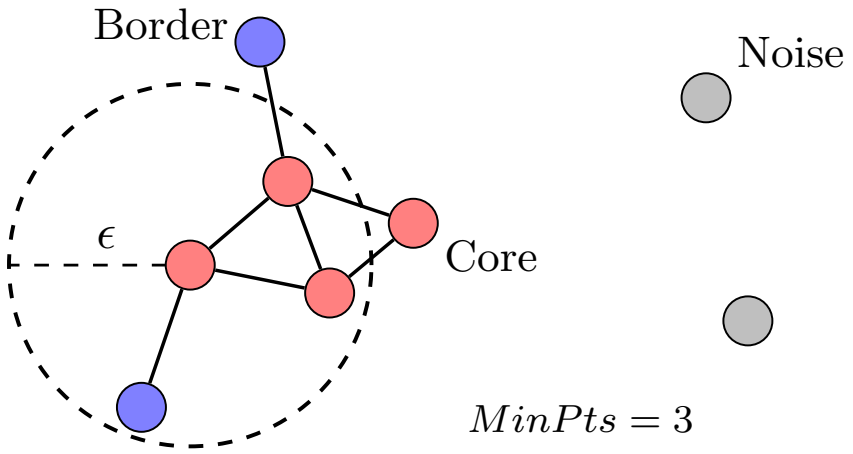


Figure 4.4: Example of application of the DBSCAN. The core points are represented in red, the border points in blue and the outliers in grey.

4.2.2.3 DBSCAN

The DBSCAN algorithm belongs to density-based clustering and has been proposed by Ester et al. [65] in 1996. Instead of assuming the number of clusters K, such methods define clusters as regions with a high density of data points separated by low density regions associated with outliers.

Let D be a database of points. For DBSCAN, the definition of the density is related to the ϵ-neighborhood N_ϵ:

$$N_\epsilon(\boldsymbol{b_i}) = \{\boldsymbol{b} \in D | d(\boldsymbol{b}, \boldsymbol{b_i}) < \epsilon)\}, \tag{4.14}$$

where d is the distance between the points, usually the Euclidean distance, and ϵ is a threshold below which the point $\boldsymbol{b}$ belongs to the cluster formed by $\boldsymbol{b_i}$.

If the cardinal of $N_\epsilon(\boldsymbol{b_i})$ is greater than or equal to a free parameter called $minPts$ then $\boldsymbol{b_i}$ is considered to be a *core-point*. The parameter $minPts$ controls the size of the smallest group of points which can be seen as a cluster. If a point $\boldsymbol{b_j}$ is located in the ϵ-neighborhood of a core-point but does not meet the $minPts$ criterion, then $\boldsymbol{b_j}$ is directly reachable from $\boldsymbol{b_i}$. The point $\boldsymbol{b_j}$ is called *directly density-reachable*. An extension of this definition can be created: a point $\boldsymbol{b_j}$ is density reachable from a point $\boldsymbol{b_i}$ if a chain of points can be found such that $\boldsymbol{b_i}$ and $\boldsymbol{b_j}$ are the extremities of the chain and that the $(k+1)$-th element of the chain is density-reachable from the k-th element (the 0-th element is $\boldsymbol{b_i}$ and the last is $\boldsymbol{b_j}$). They form the borders of the clusters. All the other points are considered as outliers by the algorithm.

Therefore, a cluster C must fulfill the following conditions:

- if $\boldsymbol{b_i}$ belongs to the cluster C and $\boldsymbol{b_j}$ is density-reachable from $\boldsymbol{b_i}$, then $\boldsymbol{b_i}$ belongs to C
- for all the points $\boldsymbol{b_i}$ and $\boldsymbol{b_j}$ belonging to C, there is a point $\boldsymbol{b_k}$ such that $\boldsymbol{b_i}$ and $\boldsymbol{b_j}$ are density-reachable from $\boldsymbol{b_k}$

Figure 4.4 shows an example of a cluster identified with the DBSCAN algorithm. All the core points have a ϵ-neighborhood of 3 or 4. Two outliers have been identified as they do not reach any core or border point.

A simple view of the DBSCAN algorithm is formulated by the following steps:

1. Fix the distance ϵ, the minimum number of points $minPts$ and declare all the points of the database D as unlabelled.
2. Select randomly an unlabelled point.
3. Label the point as core, border, or noise.
4. If the point is core point, create a new cluster with all the points that are density-connected to that core point. Mark all the points of the cluster as labelled.
5. If points of D are not labelled, return to step 2.
6. Border points reaching several core points are assigned to the cluster with the closest core point.
7. Return the set of cluster $\{C_1, \cdots, C_K\}$, with K the number of cluster that has been found.

The average run time complexity of the algorithm is in $\mathcal{O}(n.logn)$. More information on the detailed algorithm can be found in Ester et al. [65]. The determination of the parameters ϵ and $minPts$ represents a complex step of the algorithm. On the one hand, if the ϵ is too small or the $minPts$ is too large, the number of core points may be insufficient to detect the clusters. On the other hand, if ϵ is too large and $minPts$ small, the algorithm may identify only a unique cluster.

4.2.2.4 Choice of the numerical parameters

The unsupervised learning algorithm needs to set various numerical parameters in order to extract information from data. For instance, the number of clusters K must be fixed for K-means and GMM algorithms, while the DBSCAN requires the values of ϵ and $minPts$. Various methods have been developed to determine these parameters. They are mainly heuristic and graphical. It is important to note these methods should be used as a guideline to estimate the numerical parameters and not as an absolute rule.

Silhouette analysis Two main properties define the quality of the clustering: the tightness and the separation. Ideally, a cluster must be tight, meaning the points in a same cluster are close together. Tightness can be measured for instance by the average distance between the points in the same cluster and the centroid of the cluster. The term separation means the points of a given cluster should be far from points in other clusters.

Silhouette analysis [171] refers to a graphical method of interpretation and validation of consistency within clusters. It quantifies if clusters meet the requirements of tightness and separation. For a given sample $\boldsymbol{b_i}$ belonging to the cluster C_k, the value $a(\boldsymbol{b_i})$ computes the average distance between $\boldsymbol{b_i}$ and all the other points in the same cluster C_k. This definition corresponds to the average dissimilarity of the cluster:

$$a(\boldsymbol{b_i}) = \frac{1}{card(C_k) - 1} \sum_{\boldsymbol{b} \in C_k, \boldsymbol{b} \neq \boldsymbol{b_i}} d(\boldsymbol{b_i}, \boldsymbol{b}). \tag{4.15}$$

The second quantity $c(\boldsymbol{b_i})$ selects the smallest value of $a(\boldsymbol{b_i})$ if the cluster C_k is substituted by a different one:

$$c(\boldsymbol{b_i}) = \min_{l \neq k} \frac{1}{card(C_l)} \sum_{\boldsymbol{b} \in C_l} d(\boldsymbol{b_i}, \boldsymbol{b}). \tag{4.16}$$

Finally, the combination of the two terms a and c defines the Silhouette coefficient s as follows:

$$s(\boldsymbol{b_i}) = \begin{cases} 0, \text{ if } card(C_k) = 1 \\ \frac{c(\boldsymbol{b_i}) - a(\boldsymbol{b_i})}{max(a(\boldsymbol{b_i}), c(\boldsymbol{b_i}))}, \text{ otherwise} \end{cases}. \tag{4.17}$$

The value of s has a range of $[-1, 1]$. A Silhouette coefficient close to 1 indicates that the sample is far away from the neighboring clusters and that its cluster is homogenous. A negative values can be interpreted as a wrong assignment of the sample to the cluster, while a value near 0 can suggest a sample located between several clusters near the decision boundary. The Silhouette coefficient is computed for all the samples and the resulting distribution is plotted by clusters. The higher the values are, the more consistent the cluster is.

Cross-validation The cross-validation method, described in subsection 3.5.3, can also be used to determine the number of clusters. For instance, the leave-one-out predictivity coefficient can be computed for different numbers of clusters and the case with the highest value is selected as the optimal number of clusters. The predictivity coefficients can also be computed separately on each cluster to estimate the accuracy of each local models.

Elbow method The Elbow method is a graphical and heuristic method determining the numerical parameters of a clustering algorithm by identifying a break in slope of a specific quantity. This method has been developed for the DBSCAN algorithm in order to estimate the optimal value of ϵ. It is called the k-dist graph and is proposed in Ester et al. [65]. The core points are assumed to have their k nearest neighbors at roughly the same distance, while the k nearest neighbors of the noise points are at farther distances. By plotting k-distances for all the points in ascending order, a slope discontinuity can be identified, such as illustrated in Figure 4.5. The value of the k-th distance corresponding to this threshold is chosen as the ϵ associated with the thinnest cluster in the database and able to detect noise points. A typical value of 4 is proposed for k in the literature [65].

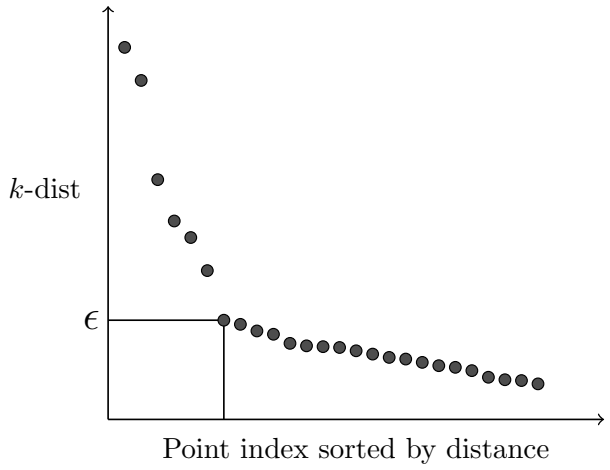


Figure 4.5: Typical shape of a sorted k-dist graph.

A similar method can estimate the number of clusters in a set of data. The total within-cluster sum of square (WSS), also called total intra-cluster variation, substitutes the k-distance of the

Elbow method for DBSCAN. This quantity measures the compactness of clusters. Let assume a K-means algorithm splits the data set into K clusters of centroid $\boldsymbol{\mu_i}$. The WSS is defined by:

$$WSS = \sum_{k=1}^{K} \sum_{\boldsymbol{b_i} \in C_k} d(\boldsymbol{b_i}, \boldsymbol{\mu_k}). \tag{4.18}$$

The total within-cluster sum of square is computed for different values of K, for instance from 1 to 10, depending usually of the size of the data set. Then the Elbow method plots the curve of WSS versus the number of clusters K. The break in slope gives an estimation of the appropriate number of clusters K.

4.2.2.5 Conclusion on clustering

The three algorithms presented above show respective advantages and drawbacks. For example, the K-means method is simple to implement and to interpret and it has a computational efficiency in high-dimension. However, the Euclidean distance represents one of its main limitation as all directions are assumed to have the same weight in the computation of the similarity. Therefore, k-means does not work well for clusters with different sizes, different densities, or for non-spherical clusters. In addition, the algorithm is sensible to initialization, rescaling of the data and outliers.

The GMM algorithm demonstrates a greater flexibility in the shape of clusters thanks to the covariance modelling. Furthermore, computing the hyperparameters can be run in parallel. Nevertheless, the GMM method shares a drawback common to several algorithms with a user-defined number of clusters. In the case of a uniformly distributed data set, these methods detect clusters accordingly to this a priori number of clusters, even if there are no meaningful clusters present in the data set. For this reason, a careful assessment of the number of clusters, as suggested in subsubsection 4.2.2.4, is a compulsory step prior to the training of the local surrogate models in order to evaluate the validity of clustering analysis.

The main advantage of the DBSCAN is that the algorithm does not need to define a number of clusters K. Thus, the number of clusters is determined by the algorithm, provided that the numerical parameters are accurately estimated. In addition, the method can find clusters of varying shapes and is numerically very efficient with a complexity in $\mathcal{O}\left(n.log(n)\right)$. However, the algorithm is not fitted for clusters with different densities, where methods such as K-means may be more efficient. Furthermore, high-dimensional data can be very challenging for DBSCAN, as the sphere of radius ϵ may not contain any data points due to the curse of dimensionality. Finally, the determination of ϵ and $minPts$ may be challenging if the k-dist does not work and if the type of data is not well understood.

In conclusion, choosing the clustering method depends largely on the distribution, the size, and the structure of the data set. Preliminary studies can help to guide the choice of numerical parameters.

4.3 Parameter space decomposition via classification

The decomposition of the input space into subspaces where a single physical regime drives the flow can be interpreted as a supervised classification problem. The clustering algorithm coupled to a shock sensor separates the snapshots into groups of similar shapes. The remaining task is to associate any input of the prediction set to the corresponding cluster. This step is called classification.

4.3.1 Overview of classification methods

Let assume the clustering step from the previous section associates the samples $\boldsymbol{\chi_{t_i}}$ to K clusters. The belonging to a cluster C_k is written through the corresponding variable c_i, such as $c_i = k$. The resulting training set $\{\boldsymbol{\chi_{t_i}}, c_{t_i}\}_{i\in[1,n]}$ is used to buid the classifier. Target variable of the clusters are arranged in a vector $\boldsymbol{c_t} = [c_{t_1}, \cdots, c_{t_n}]$ for the training points and in a vector $\boldsymbol{c_p} = [c_{p_1}, \cdots, c_{p_m}]$ for the test set.

Many regression algorithms have been presented and some of them have classification counterparts, such as ANN, SVM, GP, etc. The algorithms are modified to predict a categorical variable (a class) instead of real numbers. In this section, details are given on three different methods: k-nearest neighbors, Naive Bayes, and Gaussian Process Classification.

4.3.2 k-nearest neighbors

The k-nearest neighbors is one of the most simple algorithms in machine learning. The central idea is that only the k nearest neighbors of a prediction point $\boldsymbol{\chi_p}$ are used to estimate its value. The algorithm is memory-based, it does not require to be trained, and needs only 3 inputs: the training set, the number of neighbors k, and a distance metric used to find the k nearest points [78].

Let assume a classical Euclidean distance is chosen as distance metric. The query point $\boldsymbol{\chi_{p_j}}$ is associated with the most frequent cluster among the k nearest training samples to that query point, using simply a majority vote. Ties are broken at random. The value of k impacts directly the results of the algorithm. A low value gives better representations of thin and complex clusters and is more adapted to small data sets, while a large k is less sensible to noise by smoothing the decision boundary. The choice of k can be seen as a bias-variance tradeoff.

Despite its simplicity, the k-nearest neighbors algorithm has been successful used in a large range of classification problems, such as handwritten digits or satellite image scenes. In particular, the method is particularly relevant for irregular and complex decision boundaries [78].

4.3.3 Naive Bayes classifier

Naive Bayes classifiers are one of the most efficient and simplest algorithms for classification and data mining. Despite a very strong conditional independence assumption, rarely true for real applications, these classifiers show very good performance. Zhang [212] gives theoretical reasons explaining why naive Bayes still works even for strong dependencies between the input.

According to Bayes rule, the probability that the prediction $\boldsymbol{\chi_{p_j}}$ belongs to the cluster C_k is given by:

$$p(C_k|\boldsymbol{\chi_{p_j}}) = \frac{P(C_k)P(\boldsymbol{\chi_{p_j}}|C_k)}{P(\boldsymbol{\chi_{p_j}})}. \tag{4.19}$$

The product rule expands the right term of the numerator:

$$P(\boldsymbol{\chi_{p_j}}|C_k) = P\left(\boldsymbol{\chi}_{\boldsymbol{p_j}}^{(1)}|\boldsymbol{\chi}_{\boldsymbol{p_j}}^{(2)}, \cdots, \boldsymbol{\chi}_{\boldsymbol{p_j}}^{(d)}, C_k\right) P\left(\boldsymbol{\chi}_{\boldsymbol{p_j}}^{(2)}|\boldsymbol{\chi}_{\boldsymbol{p_j}}^{(3)}, \cdots, \boldsymbol{\chi}_{\boldsymbol{p_j}}^{(d)}, C_k\right) \times \cdots \times P\left(\boldsymbol{\chi}_{\boldsymbol{p_j}}^{(d)}|C_k\right). \tag{4.20}$$

The naive conditional independance assumption of the method states that:

$$P\left(\boldsymbol{\chi}_{\boldsymbol{p_j}}^{(i)}|\boldsymbol{\chi}_{\boldsymbol{p_j}}^{(i+1)}, \cdots, \boldsymbol{\chi}_{\boldsymbol{p_j}}^{(d)}, C_k\right) = P\left(\boldsymbol{\chi}_{\boldsymbol{p_j}}^{(i)}|C_k\right) \ \forall i \in [1, \cdots, d-1] \tag{4.21}$$

Moreover, the denominator of Equation 4.19 does not depend on the variable C_k and the input training data are fixed. Therefore, the probability to have C_k is expressed as:

$$p(C_k|\boldsymbol{\chi_{p_j}}) \propto P(C_k) \prod_{i=1}^{d} P\left(\boldsymbol{\chi}_{\boldsymbol{p_j}}^{(i)}|C_k\right). \tag{4.22}$$

The final value of the variable c associated with the input $\boldsymbol{\chi_{p_j}}$ is given by using a Maximum A Posteriori (MAP) estimation:

$$c_{p_j} = \underset{k\in[1,K]}{\arg\max}\, P(C_k) \prod_{i=1}^{d} P\left(\boldsymbol{\chi}_{\boldsymbol{p_j}}^{(i)}|C_k\right). \tag{4.23}$$

Usually, the class given by the previous equation is reliable but the precise value taken by the probability $p(C_k|\boldsymbol{\chi_{p_j}})$ should not be considered. Now it remains to determine the distribution associated with $P(\boldsymbol{\chi}_{\boldsymbol{p_j}}^{(i)}|C_k)$. Different assumptions can be made, as different kinds of naive Bayes classifiers exist. For instance, Gaussian naive Bayes assumes the distribution $P(\boldsymbol{\chi}_{\boldsymbol{p_j}}^{(i)}|C_k)$ follows a Gaussian. Means and variances are estimated using maximum likelihood computed on the training set. Other naive Bayes classifiers exist, such as multinomial or Bernoulli naive Bayes classifier.

4.3.4 Gaussian process classification

The Gaussian Process Classification (GPC) adapts the GPR to deal with classification problem. The principal steps of the method are outlined in this section only for two-class problems with $C_0 = +1$ and $C_1 = -1$. Interested readers can refer to Rasmussen and Williams [168] and Bishop [18] for the straightforward generalization to K classes.

As the nature of the classification outputs is discrete, it clearly differs from the regression problems outlined previously with GPR. The main idea is to transform the output of a Gaussian process defined on the real axis into a probability lying in the interval $[0, 1]$ using a nonlinear activation function. A latent function l defined on the input parameter space is introduced and we denote the training and testing latent vector by:

$$\boldsymbol{l_t} = [l_{t_1}, \cdots, l_{t_n}] \text{ and } \boldsymbol{l_p} = [l_{p_1}, \cdots, l_{p_m}]. \tag{4.24}$$

This function aims to provide a more convenient and tractable formulation of the model [168] and helps to define the posterior probability:

$$p(c_{p_j} = C_0|l_{p_j}) = \sigma(l_{p_j}),\ \forall j \in [1, \cdots, m], \tag{4.25}$$

with σ the nonlinear activation function defined by the sigmoid function:

$$\sigma(x) = \frac{1}{1 + e^{-x}}. \tag{4.26}$$

If l_{p_j} is positive and large in amplitude, the input $\boldsymbol{\chi}_{p_j}$ belongs to the class C_0 with large probability, while if l_{p_j} is negative and large in amplitude, the input $\boldsymbol{\chi}p_j$ belongs to the cluster C_1 with large probability.

A Gaussian process prior with a zero mean and a covariance matrix $\boldsymbol{\Sigma}_l$ is placed on the joint distribution of the latent function l:

$$\begin{bmatrix} \boldsymbol{l} \\ l_{p_j} \end{bmatrix} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_l). \tag{4.27}$$

The posterior of the latent variable is extended with the sum rule, product rule and Baye's theorem:

$$p(\boldsymbol{l}|\boldsymbol{X_t}, \boldsymbol{c_t}) = \frac{p(\boldsymbol{c_t}|\boldsymbol{l})p(\boldsymbol{l}|\boldsymbol{X_t})}{p(\boldsymbol{c_t}|\boldsymbol{X_t})}. \tag{4.28}$$

This expression is non-Gaussian due to $p(\boldsymbol{l}|\boldsymbol{X_t})$ and is used to build the probability of the latent value at a test point:

$$p(l_{p_j}|\boldsymbol{X_t}, \boldsymbol{c_t}) = \int p(l_{p_j}|\boldsymbol{X_t}, \boldsymbol{l})p(\boldsymbol{l}|\boldsymbol{X_t}, \boldsymbol{c_t})\, d\boldsymbol{l}. \tag{4.29}$$

Finally, the probabilistic prediction can be directly computed:

$$\begin{aligned} p(c_{p_j} = 1|\boldsymbol{X_t}, \boldsymbol{c_t}) &= \int p(c_{p_j} = 1|l_{p_j})p(l_{p_j}|\boldsymbol{X_t}, \boldsymbol{c_t})\, dl_{p_j} \\ &= \int \sigma(l_{p_j})p(l_{p_j}|\boldsymbol{X_t}, \boldsymbol{c_t})\, dl_{p_j}. \end{aligned} \tag{4.30}$$

One can note that $p(c_{p_j} = C_1|\boldsymbol{X_t}, \boldsymbol{c_t}) = 1 - p(c_{p_j} = C_0|\boldsymbol{X_t}, \boldsymbol{c_t})$. The integral expressed in Equation 4.30 is analytically intractable due to the non-Gaussian likelihood of $p(l_{p_j}|\boldsymbol{X_t}, \boldsymbol{c_t})$ [168]. It requires specific approximations, such as variational inference, expectation propagation or Laplace approximation [18].

The Laplace's method approximates $p(\boldsymbol{l}|\boldsymbol{X_t})$ by a Gaussian $q(\boldsymbol{l}|\boldsymbol{X_t})$ thanks to a second order Taylor expansion of $log\,(p(\boldsymbol{l}|\boldsymbol{X_t}))$ around the posterior maximum such as:

$$q(\boldsymbol{l}|\boldsymbol{X_t}) \sim \mathcal{N}\left(\widehat{\boldsymbol{l}}, \boldsymbol{A}^{-1}\right), \tag{4.31}$$

where $\widehat{\boldsymbol{l}} = \arg\max_{\boldsymbol{l}} p(\boldsymbol{l}|\boldsymbol{X_t})$ and $\boldsymbol{A}^{-1} = -\nabla\nabla log\,(q(\boldsymbol{l}|\boldsymbol{X_t})\,|_{\widehat{\boldsymbol{l}}}$.

The values of $\widehat{\boldsymbol{l}}$ is determined by maximizing the logarithm of the posterior given by Equation 4.28. Setting the gradient to zero leads to a non-linear equation which can be solved iteratively with a Newton's method. The value of $\nabla log\, p(\boldsymbol{c_t}|\widehat{\boldsymbol{l}})$ depends on the functional forms given to the likelihood. Classical solutions are the logistic and the cumulative Gaussian [168]. Details can be found in Rasmussen and Williams [168]. The probability of the latent value at a test point can be rewritten with the Gaussian approximation:

$$q(l_{p_j}|\boldsymbol{X_t}, \boldsymbol{c_t}) \sim \mathcal{N}\left(\boldsymbol{\mu_{p_j}}, \boldsymbol{\sigma_{p_j}}\right). \tag{4.32}$$

Finally, the probabilistic prediction is given by:

$$p(c_{p_j} = 1|\boldsymbol{X_t}, \boldsymbol{c_t}) = \int \sigma(l_{p_j})\mathcal{N}\left(l_{p_j}|\boldsymbol{\mu_{p_j}}, \boldsymbol{\sigma_{p_j}}\right)\, dl_{p_j}. \tag{4.33}$$

Hyperparameters of the covariance matrix $\boldsymbol{\Sigma}_l$ need to be determined in order to compute the probabilistic prediction. As for the GPR, the log marginal likelihood can be maximized. It also requires to use the Laplace approximation due to non-Gaussian terms. Derivating the likelihood exhibits terms which are implicitly functions of the hyperparameters. Nevertheless, specific derivations, developped in Rasmussen and Williams [168], evaluate the derivative of the likelihood with regards to hyperparameters. Thus a gradient-based optimizer can be used to select the hyperparameters.

4.3.5 Conclusion on classification algorithms

The k-nearest neighbors method is very sensitive to the choice of k. However, the optimal value can change as the number of training samples may increase during the training, for instance with resampling. Thus this approach is not adapted to ROMs. As regards Naive Bayes classifiers, the algorithm shows very good performance. However, the probability of belonging to a cluster may not be reliable in term of absolute value. This information can be useful to weight local models into a global one. Finally, GPs offer several advantages, such as: learning the kernel without cross validation, a fully probabilistic predictions, and an integrated feature selection. This method is chosen as the reference classification algorithm.

4.4 An entropy-based resampling

The original strategy proposed in this section takes advantage of the input space separation into several subsets. Indeed, the input space decomposition has provided subgroups of smaller size among which some subspaces of interest can be selected to perform the resampling on very specific regions.

A criterion based on the compressibility of the information is used to ensure the selection of the subspaces. It comes from an analysis of POD eigenvalues in order to identify relevant structures. As explained in the last chapter, POD eigenvalues represent the relative information contained in the different modes. The global entropy H measures the redundancy of information and is introduced as [44]:

$$H = -\frac{1}{\log(n)} \sum_{i=1}^{n} p_i \log(p_i) \text{ with } p_i = \frac{\lambda_i}{\sum_{j=1}^{n} \lambda_j}. \tag{4.34}$$

If the entropy goes to zero, there is only one nonzero singular value. The data are compressed into a unique mode. On the other hand, the entropy is equal to one if the information is distributed among the modes, meaning that no compression is possible. Between these two extreme values, the entropy increases with the number of fundamental modes. The active resampling of the LDM assumes that the entropy and the structures of the system with discontinuities or high gradients are directly correlated. Thus, the probability to find new modes with a non-negligible amount of energy is expected to be greater for clusters with high values of entropy.

Let consider a batch of n' simulations is performed. The objective is to distribute the computational budget on each cluster. The entropy can be measured locally on each cluster k, such as:

$$H_k = -\frac{1}{\log(|C_k|)} \sum_{i=1}^{n_k} p_i \log(p_i) \text{ with } p_i = \frac{\lambda_i}{\sum_{j \in C_k} \lambda_j}, \tag{4.35}$$

where n_k is the number of training samples belonging to C_k. A simple approach consisting in taking a ratio of entropy gives the number of new simulations n'_k associated with the cluster k:

$$n'_k = \frac{H_k}{\sum_{i=1}^{K} H_i} n'. \tag{4.36}$$

Figure 4.6 sketches the process of resampling. In this example, after the analysis of the entropy, blue and green clusters benefit from the majority of the computational budget while the red cluster

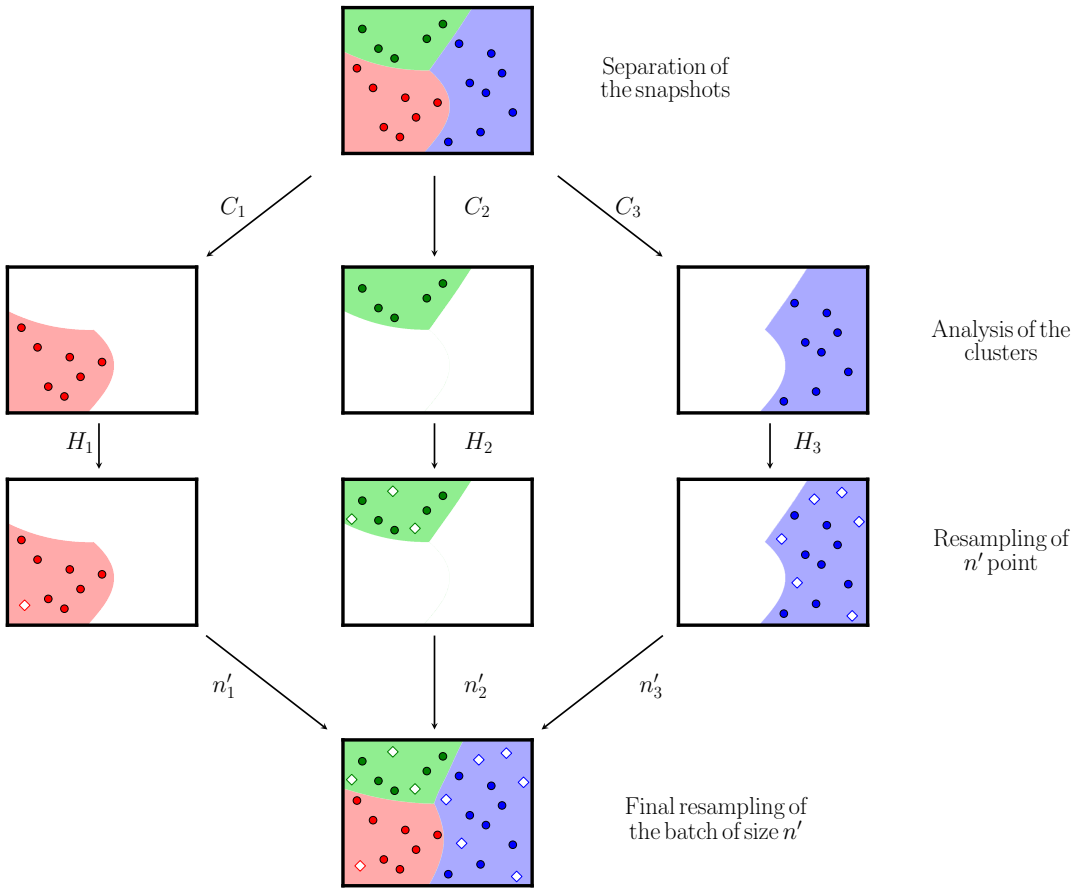


Figure 4.6: Sketch of the LDM resampling process. New snapshots are marked with white diamands.

is considered as already well explored and does not require extra snapshots as its entropy is very low.

This resampling strategy can be coupled with a stopping criterion adapted to the local properties of the LDM. Instead of computing the predictivity coefficient Q_2 by leave-one-out for all the snapshots, the Q_2 is computed and analysed severally on each cluster. Thus, this criterion allows to stop the resampling for clusters of high quality if a minimum number of snapshots have been computed. The resampling can foster the specific clusters with lower quality estimation and a significant amount of entropy.

This approach ensures to spend the computational budget in the more challenging regions of the input parameter space. Nevertheless, determining the value of the threshold is not evident. It may depend on other clusters or on the initial samples. A classical value from experience is between 0.80 and 0.99.

4.5 Recombination in a global model

The final recombination step consists of assembling the local reduced-order models in a single composite global model. It represents a challenging and crucial problem, as the classification near the decision boundary might be prone to errors due the classifier. In addition reduced coordinates can be extrapolated in this region. Thus, a special attention is paid to the interface between

clusters. Several solutions may be envisaged.

4.5.1 Hard split, soft split, and interface model

The simplest solution lets the hard or soft clustering managing the interface. Starting from the K clusters, a simple weighted sum is calculated using a 'hard' split:

$$\widetilde{\boldsymbol{f}}(\boldsymbol{\chi}_{p_j}) = \sum_{k=1}^{K} \mathbb{1}_{C_k}(\boldsymbol{\chi}_{p_j}) \widetilde{\boldsymbol{f}_k}(\boldsymbol{\chi}_{p_j}),\ \forall j \in [1, \cdots, m], \tag{4.37}$$

where

$$\mathbb{1}_{C_k}(\boldsymbol{\chi}) = \begin{cases} 1 \text{ if } k = \underset{i \in [1,K]}{\arg\max}\, P(\boldsymbol{\chi} \in C_i) \\ 0 \text{ else} \end{cases}, \tag{4.38}$$

and $\widetilde{\boldsymbol{f}_k}$ refers to the local POD/GPR model built on the k-th cluster. This sum provides a prediction of all the input space, leading to a global model which is not differentiable at the interface. The differentiable predictions require to use differentiable weighting functions, which is not the case for $\mathbb{1}$. The soft clustering [17] substitutes $\mathbb{1}_{C_k}(\boldsymbol{\chi})$ directly by $P(\boldsymbol{\chi} \in C_k)$, which is provided by the classification algorithm.

Another solution overcoming the interface problem generates a specific model at the interface either with all the training data or with only a restriction. Zhan et al. [210] propose to enclosed the clusters by freezing the classifier and sampling the decision boundary. This strategy is very efficient for two-dimensional parameter spaces where the boundary decision is a line. However, in higher dimensions, the decision boundary is represented by a hyperplan. Its sampling at a low computational budget becomes very challenging. In addition, a clear definition of the boundary decision is not always available.

Choosing the best strategy is guided by several observations. First of all, the soft clustering mixes several physical regimes. It can lead potentially to nonphysical predictions and can amplify extrapolations of the reduced coordinates near boundary decisions, one of the main weakness of GPR. Then, building a model at the interface will adds complexity to the final surrogate model as extra parameters must be introduced to identify the points in the neighborhood. Moreover, even if a global model at the interface might be a possible solution it remains sensible to the definition of the region overlapping the boundary decision. Thus, the solution of a local interface model cannot be retained and a classical hard-split approach is prefered.

4.5.2 Data enrichment near the boundary

In addition to the hard-split approach, a specific strategy improving predictions near decision boundaries has been developed. Indeed, the growing number of clusters increases directly the proportion of interface in the domain, raising problems of extrapolation in particular with the Gaussian Processes.

The central idea is to enrich each local model at the boundary with existing simulations. For each cluster, several snapshots located near the boundary do not belong to the cluster but can be useful to improve predictions near the interface. A least square problem is solved in order to determine values of reduced coordinates for points close to but outside the cluster. The strategy is divided into three steps repeted for all the clusters:

1. To find the r closest points to the cluster k but not belonging to it. This set of points is written $\boldsymbol{X_r}$. The simplest method to find these points is to compute the distance between the center of the cluster and the other snapshots in the sensor space.
2. To determine reduced coordinates for the r closest points solving the least square problem for the k-th cluster:
$$\min_{a_j(\boldsymbol{\chi})} \left\| \sum_{j=1}^{M_k} a_j(\boldsymbol{\chi}) \boldsymbol{\phi_j} - \boldsymbol{f}(\boldsymbol{\chi}) \right\|, \boldsymbol{\chi} \in \boldsymbol{X_r}, \tag{4.39}$$
where M_k denotes the number of POD modes for the k-th cluster.
3. To add these new reduced coordinates to the training set of the cluster k.

The optimal value of r can be determined by analyzing projection errors given in Equation 4.39. Snasphots with important errors must not be added to the local training set as they can significantly decrease accuracy of the local model.

This strategy can be seens as regression methods overlapping other subspaces. Nevertheless, POD modes are only computed with snapshots belonging to the cluster to avoid contamination of other physical regimes.

4.6 Validation of the method on academic test cases

The developed LDM strategy along with the adaptive sampling is first applied to analytical and two-dimensional aerodynamic test cases. The latter help to understand the functioning of the method and to validate the interest of building local ROMs. This first step prepares for more complex applications, in particular three-dimensional cases.

4.6.1 One-Dimensional Burgers' equation

The Burgers's problem [27] has been chosen to illustrate the capability of the LDM to clearly separate solutions with and without discontinuities. It underlines a moving discontinuity through the nonlinear convection-diffusion equation:

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = \nu \frac{\partial^2 u}{\partial x^2}, \ x \in [0, 2\pi], \ t \in [0, 2], \tag{4.40}$$

with ν the fluid viscosity equal to 0.1. A one-dimensional spatial domain between 0 and 2π is considered with a periodic boundary condition such that:

$$u(0, t) = u(2\pi, t). \tag{4.41}$$

The problem can be solved using the Cole-Hopf transformation [96] and the following initial solution:

$$u(x, 0) = -\frac{2\nu}{\phi}\frac{\partial \phi}{\partial x}, \tag{4.42}$$

$$\phi = e^{\frac{-x^2}{4\nu}} + e^{\frac{-(x-2\pi)^2}{4\nu}}. \tag{4.43}$$

The analytical solution is given by:

$$u(x, t) = -\frac{2\nu}{\phi}\frac{\partial \phi}{\partial x}, \tag{4.44}$$

$$\phi = e^{\frac{-(x-4t)^2}{4\nu(t+1)}} + e^{\frac{-(x-4t-2\pi)^2}{4\nu(t+1)}}. \tag{4.45}$$

For this particular problem, the discontinuity is convected outside the domain when the variable t is greater than 0.89. Thus, the challenge of the LDM is to correctly identify this threshold between smooth and discontinuous solutions.

Even if the Burgers' equation is only an analytical case, building an accurate surrogate model is very challenging. An illustration of the solutions for different values of t is shown in Figure 4.7, highlighting various shapes of the solution.

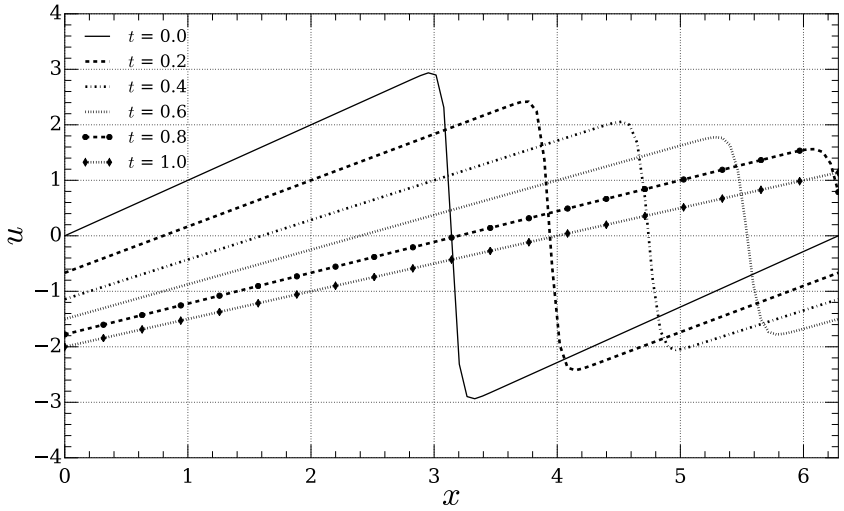


Figure 4.7: Solution of the Burgers' equation for different values of t.

4.6.1.1 Settings of the surrogate model

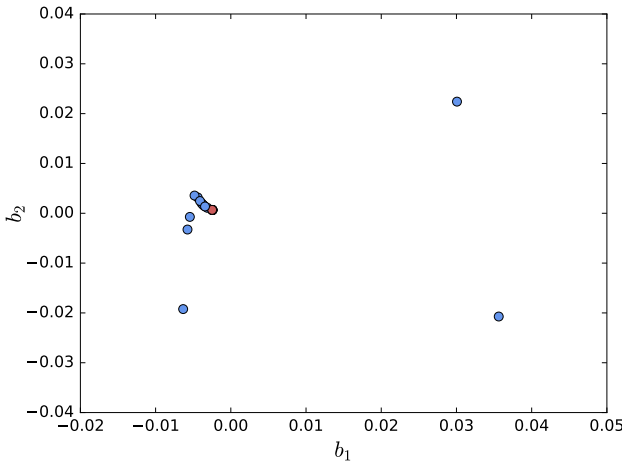
First, a global surrogate model is built without fulfilling the rule of thumb $10d$ [108], as an initial sampling of 10 provides only 5 samples with a discontinuity. For this reason, a set of 20 analytical solutions has been used to generate the training snapshots. The POD energy ratio is set to 0.99%. An Halton sequence builds the whole training set of the classical method.

Regarding the LDM, the first 10 samples are generated from a Halton sequence exploring uniformly the parameter space while the last 10 snapshots are dedicated to the resampling process, described in the previous section, in order to focus on nonlinear low-dimensional subsets. Solutions of the Burgers' equation are assumed to be characterized by two different types. Therefore, the number of clusters is set to two.

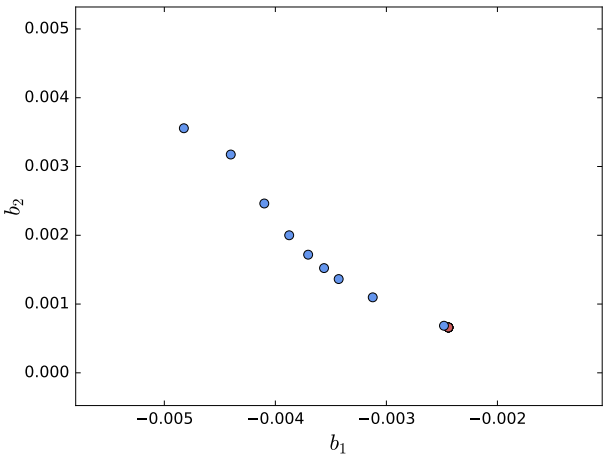
4.6.1.2 Separation of the snapshots

Different clustering algorithms are considered. As the a-priori number of clusters is known to be 2, a classical K-means algorithm is used. However, the first 10 samples are clustered into a group of 9 points and a group of a single point. Such a clustering result is not acceptable to generate relevant resampling and accurate local models. Therefore, a k-graph distance is drawn for the 10 samples of the training set, as illustrated in Figure 4.8c. A clear separation appears between a first group of 5 points at a distance of about 10^{-7} and the other 5 points between 10^{-2} and 10^{-1}. Thus, a DBSCAN algorithm is applied on the samples with ϵ sets to 10^{-6} and $minPts = 2$.

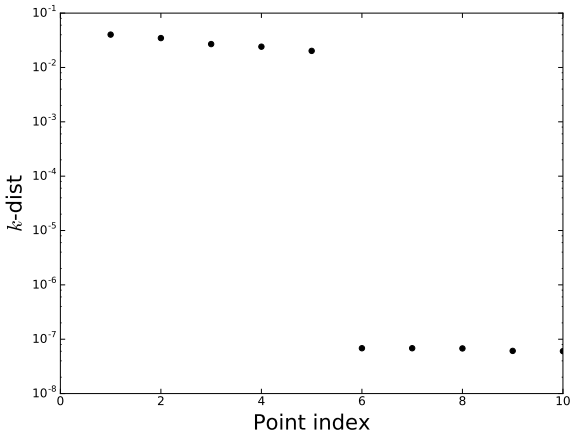
Two clusters are identified by the algorithm: one very tight and one very sparse associated with noise by the algorithm. Then, the resampling is performed by computing 10 new samples distributed between the two clusters. The final clustering of the samples with the DBSCAN algorithm shows that snapshots with no discontinuity form a very tight cluster. Indeed, Figure 4.8a represents the distribution of the samples in the space of the sensor. The blue dots correspond to the cluster with discontinuity while the red dots correspond to the case where the discontinuity is convected outside the domain. Even a zoom on the region of the red dots in Figure 4.8b is not sufficient to clearly separate the snapshots which are contained in a circle of radius $\epsilon = 10^{-6}$. Analyzing the Silhouette



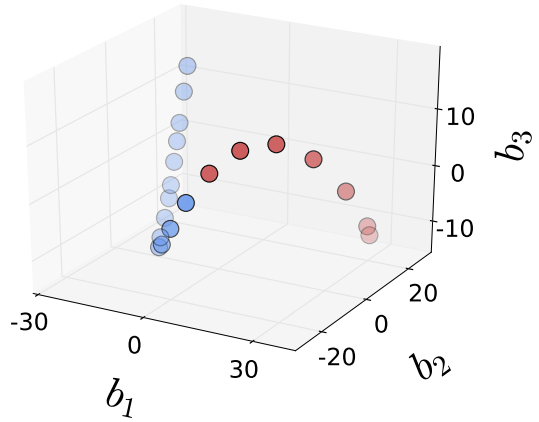
(a) Distribution of the samples with Jameson sensor. DBSCAN clustering.



(b) Zoom on the clustered region. DBSCAN clustering.



(c) k-graph.



(d) Distribution of the samples with no sensor. K-Means clustering.

Figure 4.8: Illustration of the different clusterings for the Burgers' equation.

coefficients indicates that the cluster associated with the smooth snapshots is very coherent with a mean value aroud 0.97 while the cluster corresponding to the discontinuous snapshots shows a mean coefficient of -0.1. This very low value is explained by very scattered snapshots, in particular those with the strongest discontinuities.

The clear clustering of the snapshots with no-discontinuity is ensured by the use of the Jameson's shock sensor. Indeed, Figure 4.8d exhibits the distribution of the first modes of the 20 snapshots without a sensor. Even if a one-dimensional manifold is identified, a clear separation between

the continuous and discontinuous terms is not permitted. The k-graph does not highlight any break of slope. All the k-dist values vary very little between 10^{-2} and 10^{-1}. Moreover, the initial training set with 10 samples appears very sparse and an attempt to separate these samples without the shock sensor leads to faillure of the clustering: very discontinuous snapshots are mixed with smooth ones. Therefore, these results validate the coupling between Jameson shock sensor and clustering algorithms in order to separate the different physical regimes.

Obviously, the one-dimensional classification of the well-separated snapshots does not pose a problem. Figure 4.9 illustrates the classification of the solutions performed by the algorithm. The decision boundary is located around $t = 0.93$, very close to threshold value $t = 0.89$ corresponding to the time when the discontinuity is convected outside the domain.

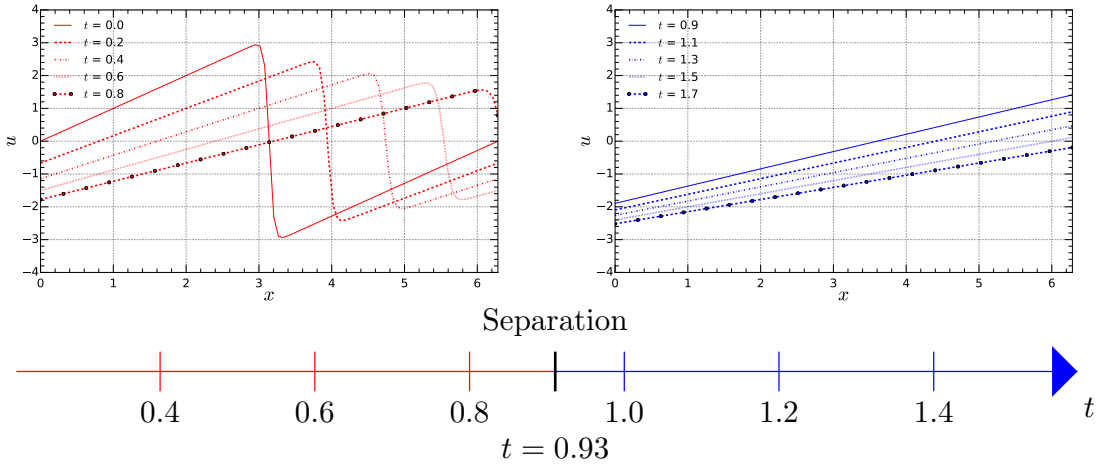


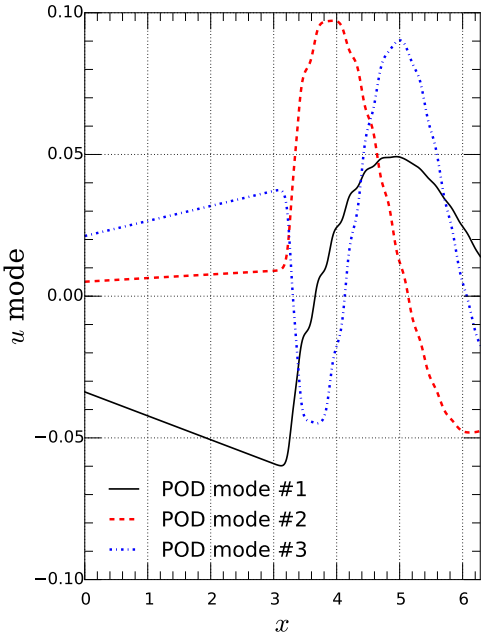
Figure 4.9: Separation of the Burgers' equation solutions.

4.6.1.3 Analysis of the POD modes and LOO errors

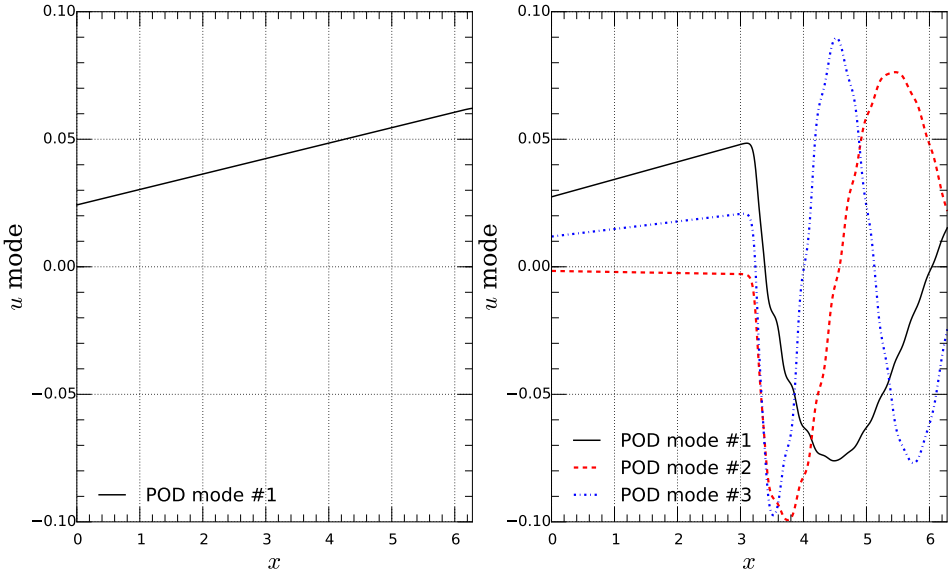
Looking at POD modes highlights the behavior of the classical method and the proposed LDM. Figure 4.10 displays the first three POD modes. Only discontinuous features are extracted for the classical method, as illustrated in Figure 4.10a. Thus, it can be assumed that the recombination of a linear signal from only discontinuous shapes gives rise to severe problems, especially in term of model accuracy. On the other hand, the LDM accurately isolates smooth and discontinuous features. Figure 4.10b emphasizes this separation. The smooth subset encompasses only a single POD mode with a linear shape, whereas the three first modes of the discontinuous subset have sharp shapes.

Table 4.1 gives the entropy and the number of required modes for the different methods with regards to the number of snapshots. It shows that the LDM has increased the entropy of the training set from 0.65 to 0.83, as not negligible new features have been found with the resampling process. The latter has demonstrated its efficiency by computing all the 10 extra snapshots in the time interval corresponding to the discontinuity still present in the computational domain. Moreover, the very low value of the entropy for the smooth subset of the LDM illustrates the redundancy of the snapshots for this specific type of solution.

A leave-one-out cross validation has been performed on the global and LDM models. The line graph in Figure 4.11 shows the leave-one-out predictivity coefficient Q_2 for the two kinds of models against the number of training samples. It is clear from the chart that the snapshots from the



(a) Classical method.



(b) LDM, smooth (left) and discontinuous subspaces (right).

Figure 4.10: POD modes of the Burgers' solution.

cluster #1 are sufficient to build an accurate model, as proven by the Q_2 which remained constant at 0.99 and an entropy of 2.10^{-3}. Thus, this cluster do not need to be enriched by additional snapshots and the full computational budget is devoted to cluster #2. The predictivity coefficient of the cluster #2 goes up noticeably from about 0.35 before the resampling to 0.875 with 20 training samples. The cluster associated with the discontinuous snapshots surpasses even the global method with an estimated Q_2 of 0.78. This a-priori analysis of the surrogate model validates the relevance of the LDM.

Method	Number of snapshots	Number of modes	Entropy
Classical	20	10	0.65
LDM (nonlinear subset)	15	13	0.83
LDM (linear subset)	5	1	2.10^{-3}

Table 4.1: Required modes of the different methods for the Burgers' problem.

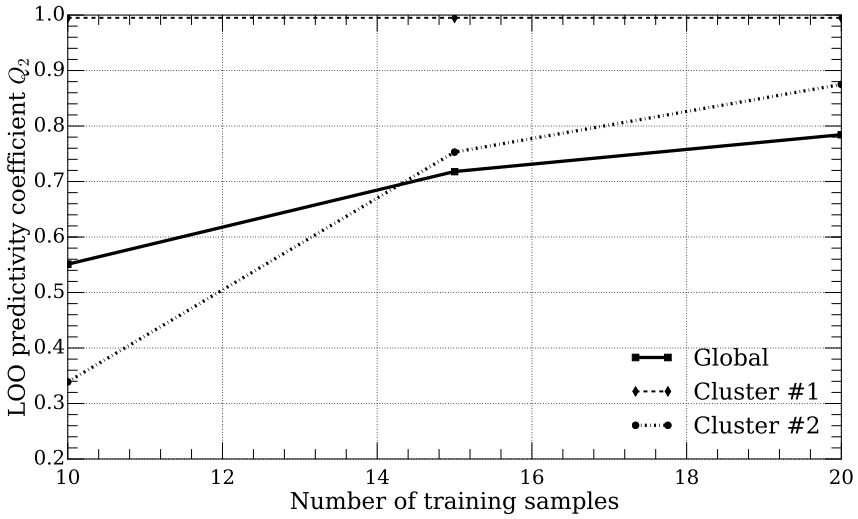


Figure 4.11: Evolution of the leave-one-out predictivity coefficient versus the number of training samples for the Burgers' equation. The global model and the LDM model (cluster #1 and #2) are presented.

4.6.1.4 Accuracy of the surrogate models

In this section, a more detailed look is taken at the analysis of the model accuracy. A test set has been built from 1000 snapshots of a Sobol sequence in order to reach a sufficient statistical validation of the model. Three different phases have been considered:

- The full parameter space containing all the samples of the testing set.
- The discontinuous part which encompasses snapshots with a value of t below 0.89, when the wave is still within the domain.
- The smooth part is defined as the complementary of the discontinuous domain: the wave has been convected outside the domain.

In Figure 4.12, the normalized error between classical method and LDM is presented for comparison. It exhibits that the LDM achieves better accuracy, as illustrated by the dramatic drop of the error for all the statistical characteristics of the box plot (quartiles, median, mean, percentile and outliers) and for all the considered phases. A striking finding is that the normalized error of the LDM tends almost to zero for the smooth type of solution. Moreover, it is clear that the discontinuous parts of the solutions represent the largest contribution to the error for both methods, justifying the local approach and the resampling process.

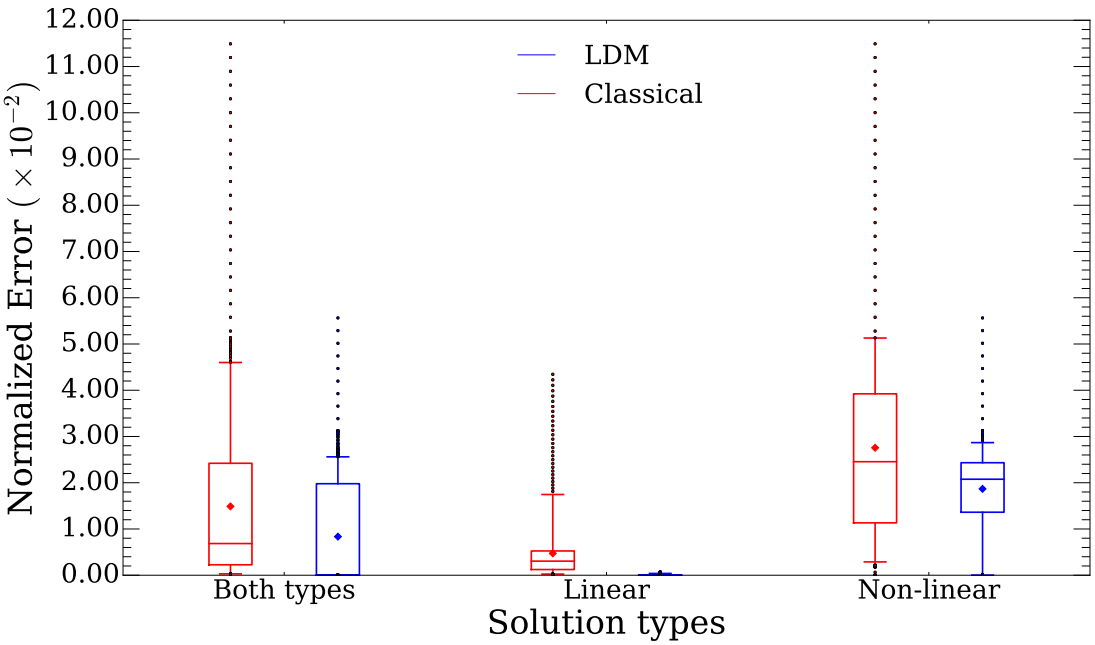


Figure 4.12: Accuracy of the surrogate model for the Burgers' problem.

4.6.2 Two-Dimensional RAE2822 Transonic Airfoil

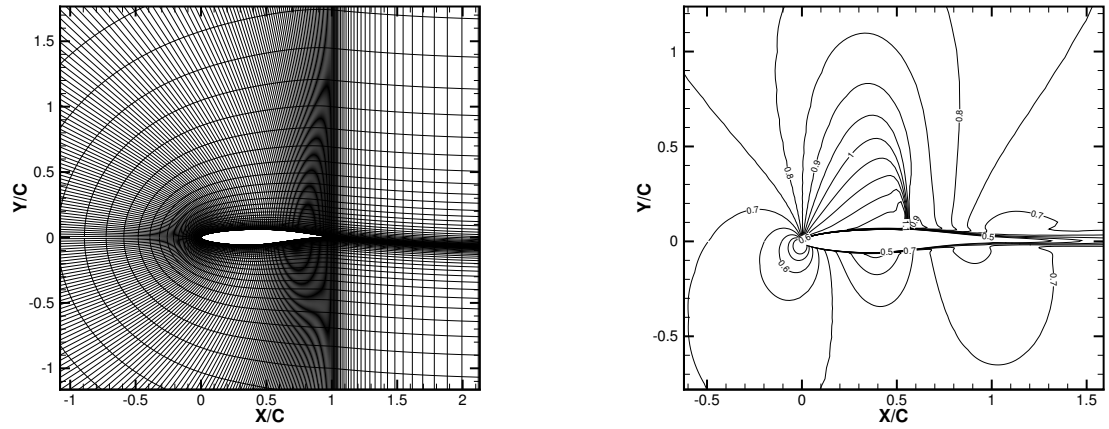
The turbulent flow around a RAE2822 airfoil has been widely studied in the literature both numerically and experimentally [42, 87]. The feature of prime interest of this test case is that the inflow conditions govern the flow regime, leading to the appearance of shock waves on the suction side. The detection and the clear separation of these regimes represent the main challenge for the model, demonstrating the capability of the LDM to deal with real and complex physics compared with the classical method.

4.6.2.1 Computational configuration

The high-fidelity computations are carried out using the cell-centered finite-volume solver *elsA*-ONERA [31], whose the co-owners are Airbus, Safran, and ONERA. It solves the compressible Reynolds Average Navier Stokes (RANS) equations on structured and hybrid grids. From the numerical point of view, the classical second order central scheme of Jameson, Schmidt and Turkel [103] is used for the space discretization. The time integration is performed with the backward Euler implicit scheme: the algebraic system is linearized with the LU-SSOR implicit method [207]. The turbulence modeling is ensured by the model of Spalart and Allmaras. A $2D$ mesh containing $23,010$ points is used, as illustrated in Figure 4.13a. We have validated this test case on a well-known regime flow at Mach 0.734 and a angle of attack of 2.79 [42]. Illustrations of the computational grid and of the Mach number contours are given in Figure 4.13b.

4.6.2.2 Input parameter space, quantities of interest and settings of the surrogate model

Three freestream parameters are considered as varying for this application: the flight speed, the angle of attack α, and the altitude h. Their variations are resumed in Table 4.2. These dimensional parameters are nondimensionalized and are expressed respectively as the Mach Number M, angle



(a) View of the computational grid.

(b) Mach number contours (M=0.734 and α=2.79).

Figure 4.13: Flow around the RAE2822 airfoil. The chord of the airfoil is written C, X refers to the horizontal coordinate, and Y refers to the vertical.

Freestream variable	Amplitude of variation
Flight speed (m.s^{-1})	88.5 - 269
Angle of attack (°)	0.5 - 3.0
Altitude (m)	1000 - 11,000

Table 4.2: Freestream conditions for the RAE2822.

of attack α (no change) and Reynolds number. One can note that the latter is impacted by the altitude variations but ensured to be maintained in a given interval in order to have a sufficiently resolved boundary layer. The atmosphere is modeled by the International Standard Atmosphere of the International Civil Aviation Organization [1]. It assumes that the air is a perfect gas and that the atmosphere can be divided into layers with a linear distribution of temperature against the altitude. The temperature T and the density ρ can be directly expressed in function of the altitude:

$$T = T_0 - L\,h, \tag{4.46}$$

$$\rho = \frac{p_0(1 - \frac{L\,h}{T_0})^{\frac{g}{rL}}}{r(T_0 - L\,h)}, \tag{4.47}$$

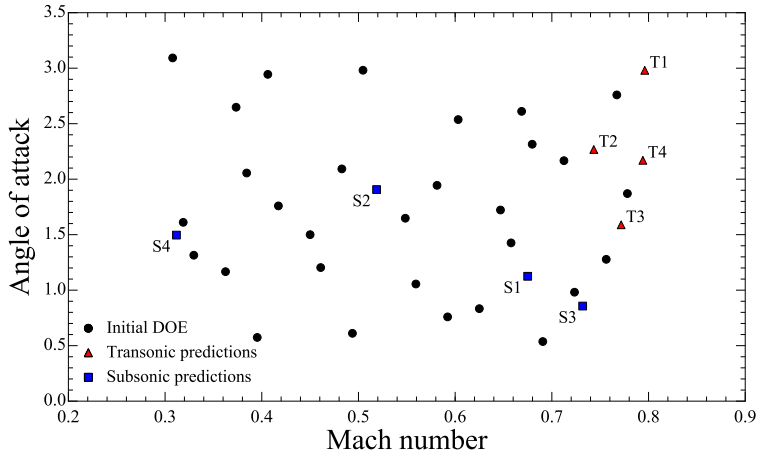
with p_0 and T_0 the pressure and temperature at sea level, L the temperature lapse rate, r the specific gas constant of air, and g the acceleration due to the gravity. The quantities of interest of the simulation are the pressure coefficient C_p and the friction coefficient C_f on the suction side defined by:

$$C_p = \frac{p - p_\infty}{\frac{1}{2}\rho_\infty U_\infty^2}, \tag{4.48}$$

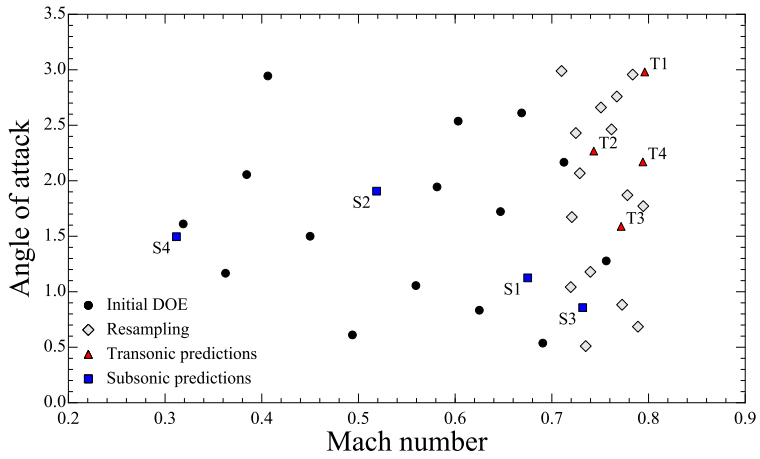
$$C_f = \frac{\tau_w}{\frac{1}{2}\rho_\infty U_\infty^2}, \tag{4.49}$$

where p is the static pressure, τ_w the wall shear stress, p_∞, ρ_∞, and U_∞ are respectively the static pressure, the density and the velocity in the freestream.

As regards DOE, 30 samples form the training set, following the rule of thumb $10d$ [133]. The samples of the global model are generated only from a Halton sequence while the sampling of the LDM is divided into two parts. An initial DOE mixing subsonic and transonic snapshots explores uniformly the parameter space with an Halton sequence of 15 samples. The last 15 samples follow the resampling process described in the previous section where a Q_2 threshold has been fixed to 0.99 as stopping criterion. The flow is assumed to be driven by two different flow regimes. Therefore, the number of clusters is set to two for the clustering step. Finally, a test set has been built from 300 snapshots of a Sobol sequence for the statistical validation of the LDM.



(a) Classical method: Halton sequence.



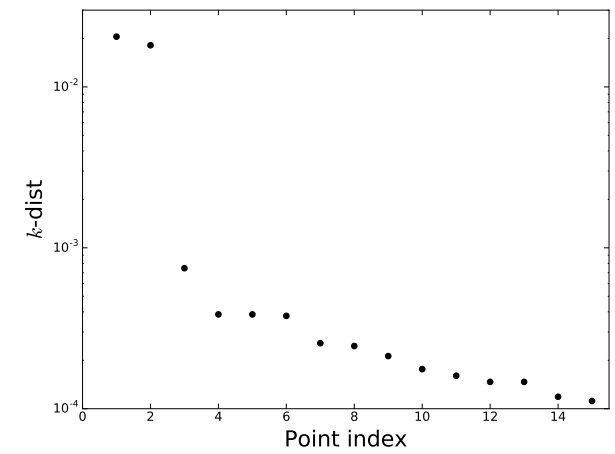
(b) LDM: 15 samples from an Halton sequence and 15 samples for the resampling process (grey diamonds).

Figure 4.14: Training set of 30 samples for the classical method and the LDM for the RAE2822. The eight illustrative predictions are also identified (with blue squares and red triangles).

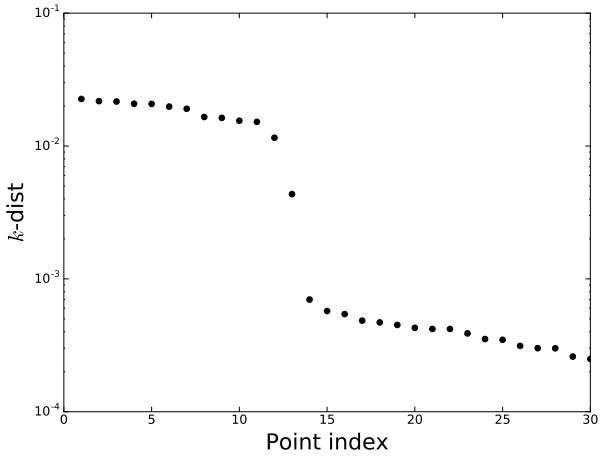
4.6.2.3 Separation of the snapshots and resampling of the parameter space

An analysis of the first 15 training samples exhibits a data structure with 2 clear clusters, as illustrated with the repartition of the k-distance in Figure 4.15a, where a typical distance of 10^{-3} can be measured. This value is set for the DBSCAN algorithm which separates successfully the 15 snapshots into two distinct groups as depicted in Figure 4.15c. One can note that the GMM algorithm achieves the same result. Only two points are well separated from the cluster of the

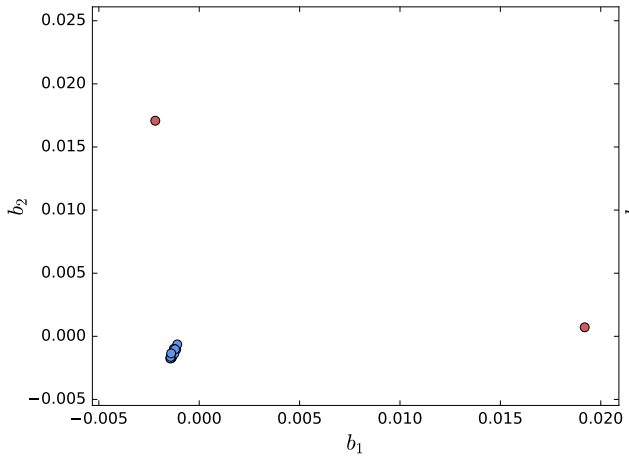
subsonic snapshots, highlighting the fact that the transonic region might be very narrow.



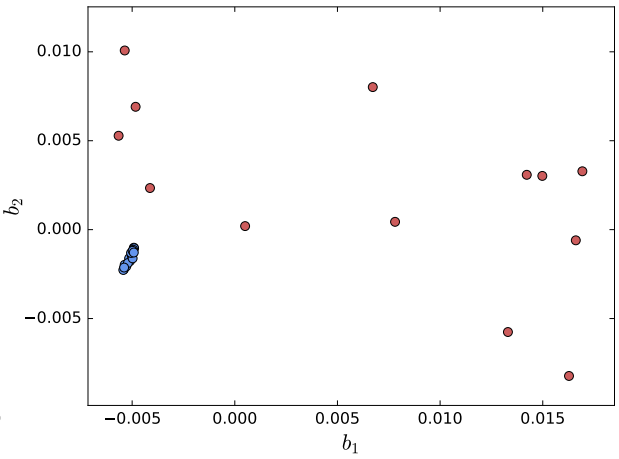
(a) k-graph for 15 samples.



(b) k-graph for 30 samples.



(c) Distribution of the samples for 15 samples in the shock sensor space.



(d) Distribution of the samples for 30 samples in the shock sensor space.

Figure 4.15: Illustration of the clustering for the RAE2822.

Figure 4.16 depicts the variation of the LOO predictivity coefficient with the number of training samples. A very high value of the Q_2 is observed for the cluster #1. This value remains above 0.99, hence this cluster is interpreted as having well predictivity capacities and all the computational budget is dedicated to the cluster #2 with a much lower Q_2 of 0.29 for 15 training samples. The evolution of the cluster #2 indicates that the model accuracy is improved for this cluster, justifying the use of the computational budget on this region. Flows with high disconitnuities are expected to be predicted more accurately, as the predictivity coefficient rises from about 0.29 to 0.68. Even if the latter value seems still a little bit low, the LDM should improve the global predicitivy capacities of the surrogate compared with the global model.

The two training sets end up with very different shapes. The global model explores uniformly the full input space with the Halton sequence, whereas the resampling process focuses on a small region located around high Mach number values, as illustrated in Figure 4.14 for the C_p. Furthermore, the resampling process helps to improve the clustering of the data, as illustrated by the clear

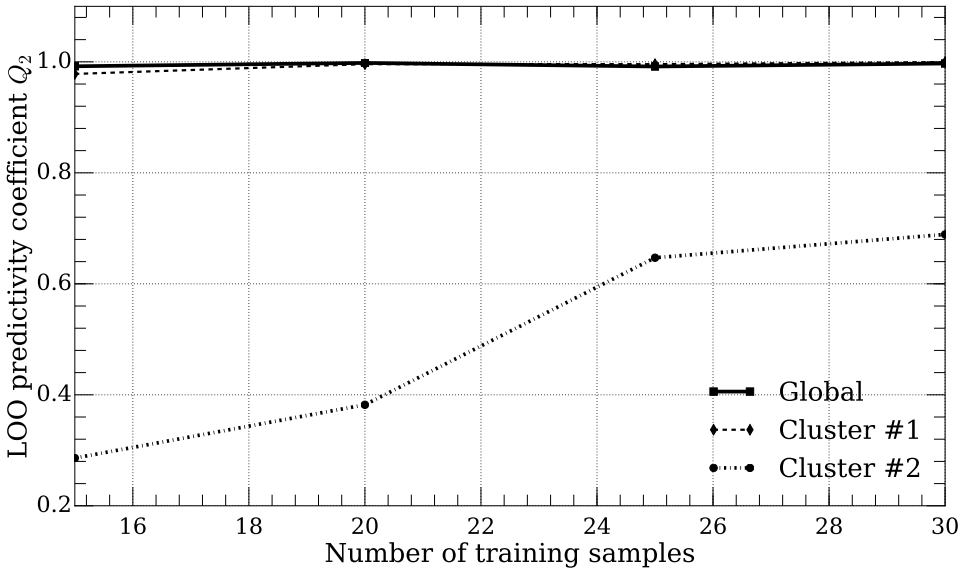


Figure 4.16: Evolution of the LOO Q_2 for the global model and the LDM.

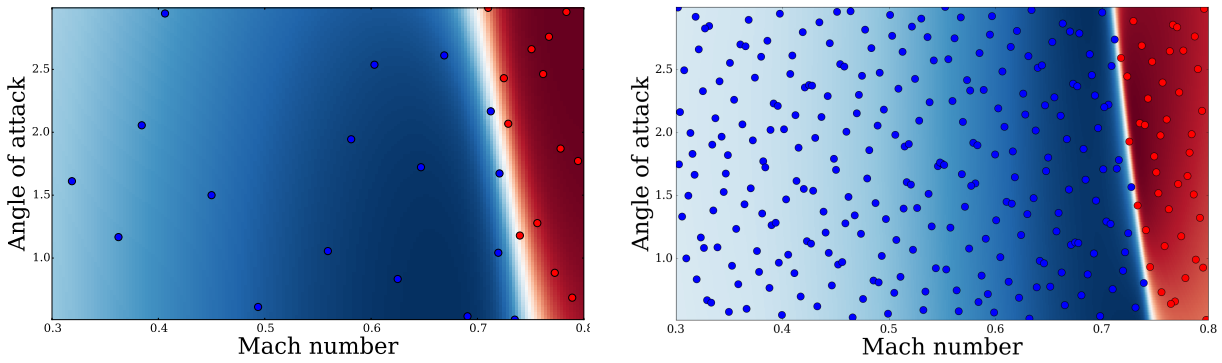
break in slope of the k-dist in the Figure 4.15b. Figure 4.15d shows the distribution of the samples in the shock sensor space and highlights the fact that the resampling focuses on snapshots with discontinuous shapes, represented by the red scattered points. This trend is confirmed by the number of training samples with a low k-dist value. Thus, the resampling step is very performant for this application, since the sensor space has been widely expanded with spread extra snapshots.

The supervised algorithm decomposes the input space parameter in two clear subsets, as illustrated in Figure 4.17a. This separation is confirmed on the testing set of 300 samples in Figure 4.17b. These two clusters can be interpreted as the subsonic and the transonic regions. It can be observed that the boundary is mainly influenced by the Mach number but also slightly by the angle of attack. Thus, the resampling process has increased the density of samples in the transonic regime, improving the accuracy of the model where predictions are more challenging. One can note that the boundary region is well defined by a thin region of probability between around 50% and represented in white.

4.6.2.4 Analysis of POD modes

The model can also be analysed from a dimension reduction point of view. Table 4.3 summarizes the POD information in terms of entropy, number of snapshots and number of modes. For a given POD energy ratio of 0.99%, the classical method reduces the dimension of the 30 snapshots with 7 modes and an entropy of 0.37, whereas the LDM identifies clearly a POD basis with a low entropy and a POD basis with a large one. The highest nonlinear cluster shows an entropy values of 0.63, 70% bigger than the linear cluster and the classical method. It means the data of the subsonic POD basis and of the classical method can be highly compressed. Conversely, the transonic POD basis requires 9 modes for 13 snapshots. Thus, the active resampling provides extra non-redundant information to the LDM compared with the 20 training samples from the Halton sequence of the classical method.

The shapes of POD modes can also help to understand the behavior of the different models. As regards the classical method, only discontinuous features emerge clearly from the global POD



(a) Training set (b) Test set

Figure 4.17: RAE2822 input space decomposition. Each color corresponds to a cluster, the blue one is the subsonic regime and the red one is the transonic regime. The density of color gives the probability to belong to the cluster.

Method	Number of snapshots	Number of modes	Entropy
Classical	30	7	0.37
LDM (transonic regime)	13	9	0.63
LDM (subsonic regime)	17	6	0.36

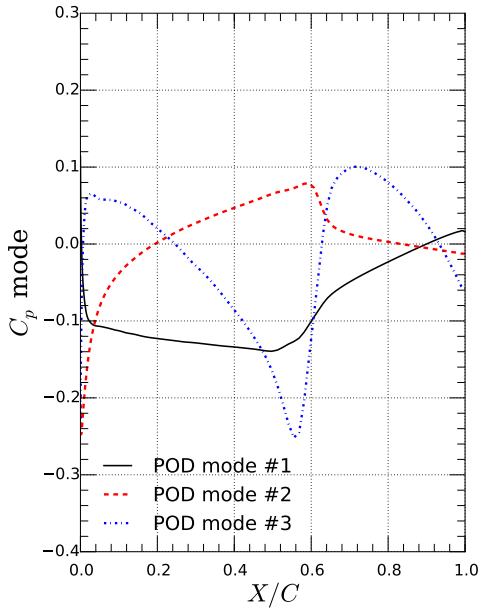
Table 4.3: Required modes of the different methods for the RAE2822.

as illustrated in Figure 4.18a, where POD modes shown only dominant discontinuities and no-moving shocks. Therefore all the reduced coordinates associated with subsonic snapshots must exactly cancel out the discontinuities of the modes. For this reason, the prediction of the reduced coordinates for snapshots in the subsonic region can be sensitive to interpolation errors leading to the appearance of "residual" shocks. Moreover, few snapshots are effectively located in transonic region for the classical model. The discontinuous features may be badly amplified by the regression models as the training set is very sparse, leading also to important error for transonic predictions. Thus, the clear separation of regimes in the POD domain represents a major asset inherent in the LDM. Figure 4.18b shows the first three modes of the two POD bases. High nonlinearities arise for the transonic regime whereas the subsonic region highlights similarities with the modes of the classical method but they are smoother.

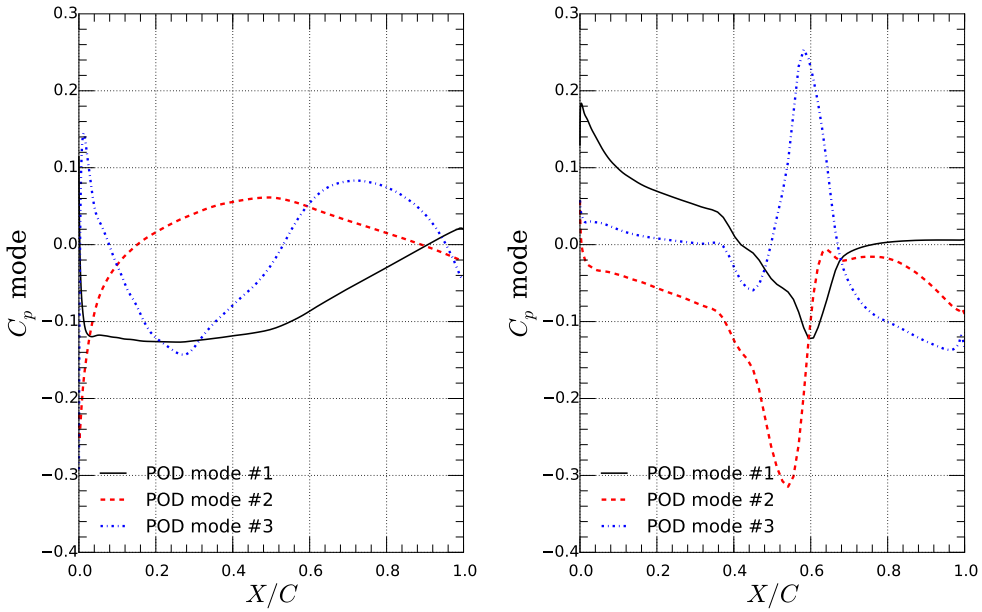
4.6.2.5 Accuracy of the model

A more detailed look is given to the statistical validation of the model and the analysis of the accuracy. Figures 4.19 and 4.20 display the comparison of the normalized error in term of C_p and C_f for both methods between the predictions and the test set. Results are presented with a box plot formalism and three different phases has been considered:

- the full domain contains all the samples of the testing set;
- the subsonic regime is only composed of the testing samples identified as subsonic, in blue dots in Figure 4.17b;
- the transonic regime encompasses the other snapshots, shown as red dots in Figure 4.17b.



(a) POD modes for the classical method.



(b) POD modes of the C_p for the LDM, subsonic (left) and transonic (right).

Figure 4.18: POD modes of the RAE2822 simulations.

A significant improvement of accuracy is induced by the LDM for the C_p. As regards the full domain, the normalized error decreases dramatically for all the statistical characteristics of the box plot. In particular, the extreme value of the LDM reaches the same level as the 95% error of the classical method, illustrating a large reduction of the model variability. The box plots for subsonic and transonic regimes provide a closer look at the repartition of the error. It clearly appears that the LDM improves the predictions at transonic regime, explained by the increase of the sample density. On the other hand, the subsonic regime is very slightly impacted compared with the

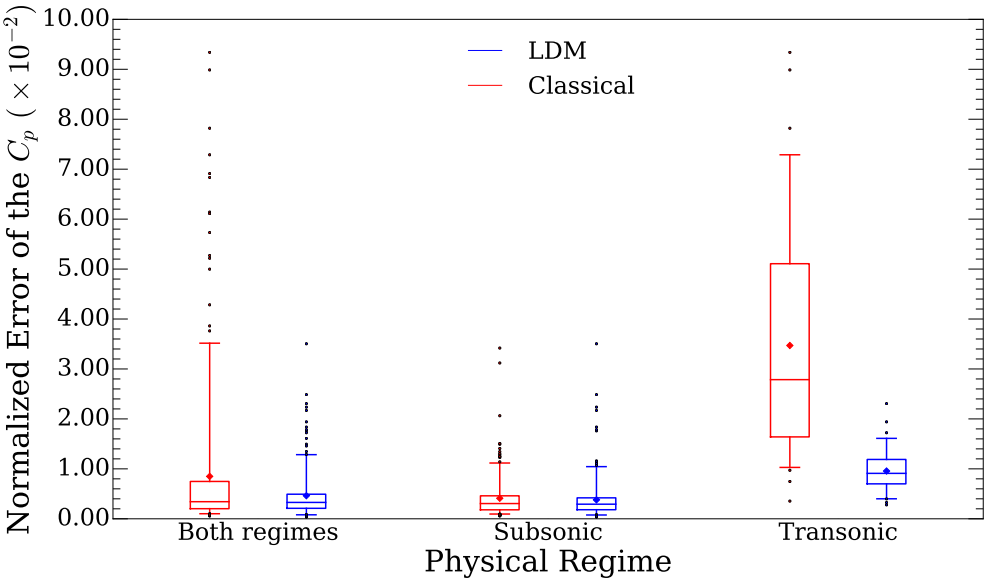


Figure 4.19: Accuracy of the surrogate model in terms of C_p.

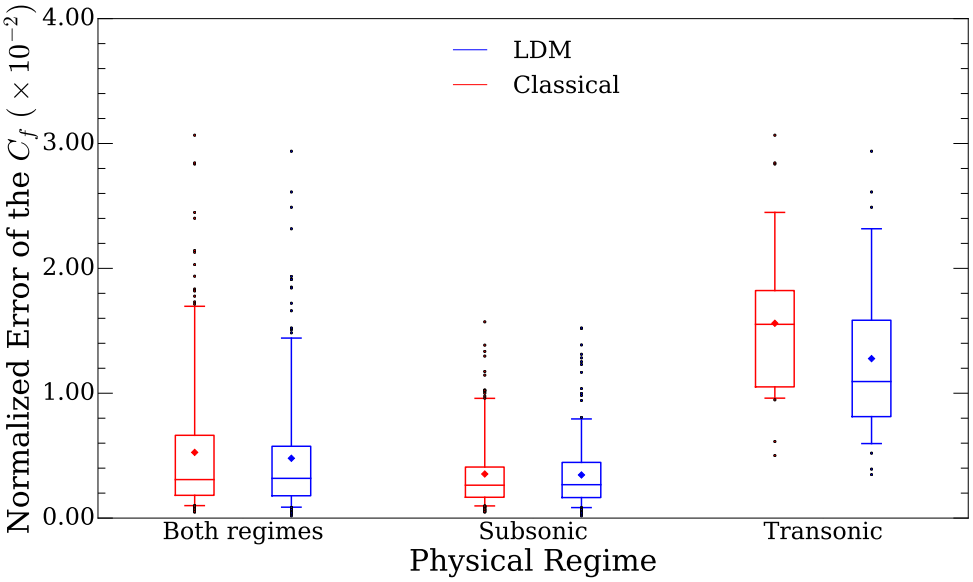


Figure 4.20: Accuracy of the surrogate model in terms of C_f.

classical method, although samples has been removed, highlighting the fact that extra samples are not necessary in this region of the parameter space.

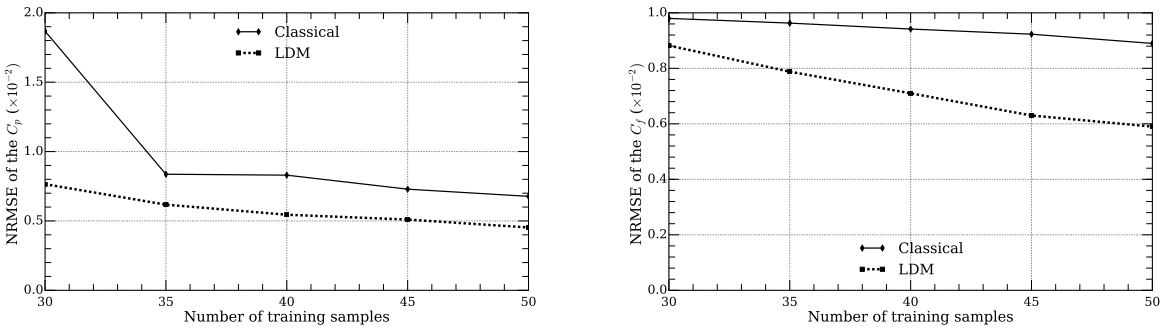
Table 4.4 provides a more global view of the error with the spatial average of the Q_2 and NRMSE. Whatever the quantity measuring the error, the same trend is observed for the two quantities of interest: the LDM significantly improves the accuracy of the predictions compared with the classical method. There are, however, legitimate doubts as to the value of the $\langle Q_2 \rangle_\Omega$ for the C_f which is below zero for the transonic regime. Several explications can be given. First of all, the $\langle Q_2 \rangle_\Omega$ is spatially averaged and the value of C_f can be very close to zero after the shock. Thus, some values of Q_2 falls far below zero in this region, impacting directly the average of the predictivity

coefficient. Then, the skin friction coefficient is also more challenging to predict due to its higher dependence to the altitude and higher nonlinearity. Finally, the variance can be locally very low and close to zero, leading to negative predictivity coefficients.

Regime	Method	$\langle Q_2 \rangle_\Omega$ of C_p	$\langle NRMSE \rangle_\Omega$ of C_p	$\langle Q_2 \rangle_\Omega$ of C_f	$\langle NRMSE \rangle_\Omega$ of C_f
Both	Classical	0.897	1.93×10^{-2}	0.837	1.15×10^{-2}
	LDM	0.989	7.48×10^{-3}	0.908	9.53×10^{-3}
Subsonic	Classical	0.984	7.46×10^{-3}	0.912	6.61×10^{-3}
	LDM	0.990	6.31×10^{-3}	0.957	6.72×10^{-3}
Transonic	Classical	0.0974	4.64×10^{-2}	-0.303	2.54×10^{-2}
	LDM	0.901	9.79×10^{-3}	-0.0223	1.72×10^{-2}

Table 4.4: Summary of the results in term of Q_2 and NRMSE for the RAE2822.

The evolution of the NRMSE with the number of training samples is shown in Figure 4.21 for both pressure and skin friction coefficients. One may note that the classical method with 50 training samples has broadly the same error than the LDM strategy with 30 samples for both signals. In the end, this observation could be interpreted as a saving of the computational budget of more than 60%. Moreover, the error of the LDM seems to continuously decrease when the number of training samples increases. In particular, the gap between models increases for the C_f. Thus a local approach becomes increasingly relevant in this case, as the resampling process allows to discover new typical shapes in the transonic region, explaining the evolution of the gap.



(a) Pressure coefficient.

(b) Friction coefficient.

Figure 4.21: Evolution of the NRMSE with the number of training samples.

4.6.2.6 Specific predictions

The statements of the POD analysis and the statistical validation are highlighted by looking at several C_p profiles comparing classical and LDM methods. Eight predictions are computed for an illustrative purpose and are grouped by physical regime (S for subsonic and T for transonic). Their coordinates in the parameter space are summarized in Table 4.5 and their repartition is illustrated in Figure 4.14.

One can observe that for the subsonic regime in Figure 4.22, the classical method induces residual shocks, certainly due to errors of prediction on the reduced coordinates coupled with sharpe POD modes not adapted to subsonic flows. On the contrary, residual shocks are filtered by the LDM as

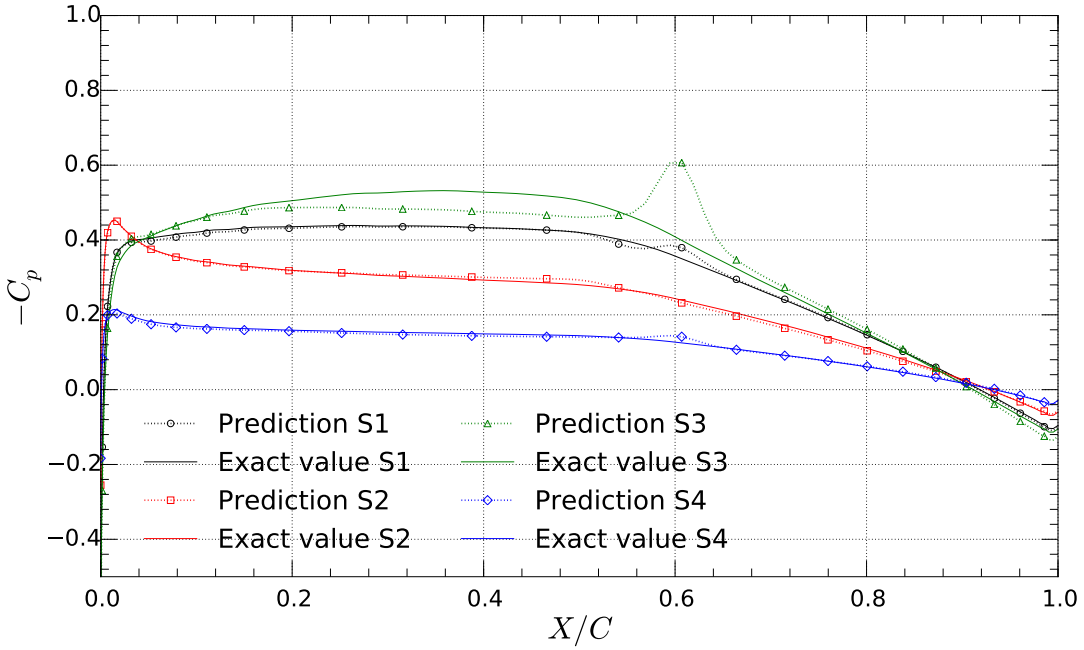
Predictions	Mach Number	Angle of attack (°)	Altitude (m)
S1	0.675	1.125	8500
S2	0.519	1.906	1625
S3	0.722	1.047	1937
S4	0.312	1.496	5453
T1	0.796	2.980	6391
T2	0.743	2.267	10,102
T3	0.772	1.587	3168
T4	0.794	2.170	3617

Table 4.5: Coordinates in the parameter space of the eight illustratives predictions.

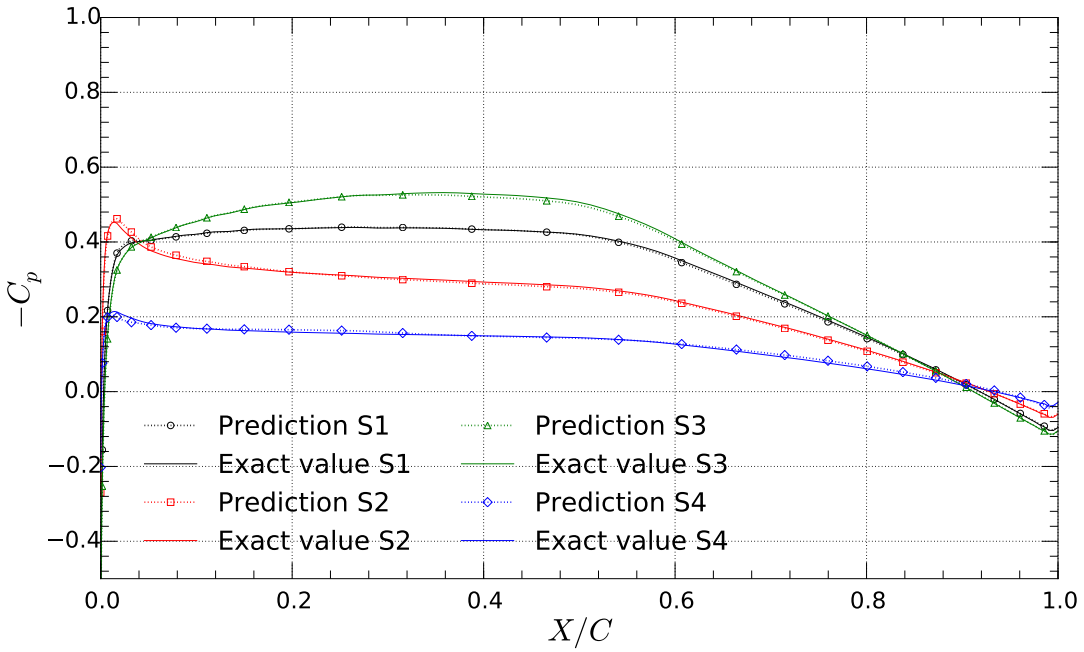
lower discontinuities are present in the training snapshots building the POD basis. It leads to final predictions less sensitive to errors on the reduced coordinates.

As regards the transonic regime for C_p profiles illustrated in Figure 4.23, the LDM greatly enhances the accuracy of the predictions. In particular, the modeling of the shock waves gives an insight into the behavior of the two models. Significant discrepancies in term of shock displacement and shock amplitude occur for the classical POD/GPR method. The combination of angle of attack and Mach number influences directly the localization of the steady shock wave and its amplitude, but this behavior can only be accurately caught if the training set contains a sufficient amount of snapshots with shocks, which is not the case for the classical POD/GPR methods.

The same trend is observed for C_f profiles. The predictions given by the LDM for the subsonic profiles in Figure 4.24 are almost superimposed with the CFD solutions. On the contrary, residual discontinuities are observed for the classical model. In particular, the prediction S_3 shows a high level of discrepancy. Indeed, as illustrated in Figure 4.14a this specific point is close to transonic snapshots of training sets which influence the interpolation of the reduced coordinates. The LDM avoids the issue thanks to the separation between subsonic and transonic snapshots.

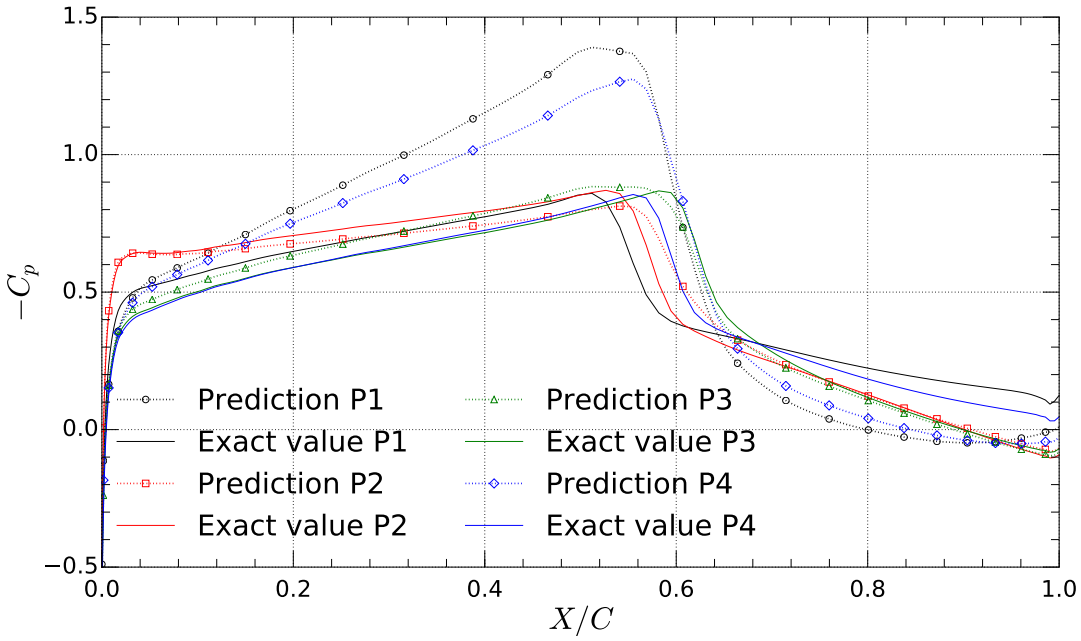


(a) Classical model

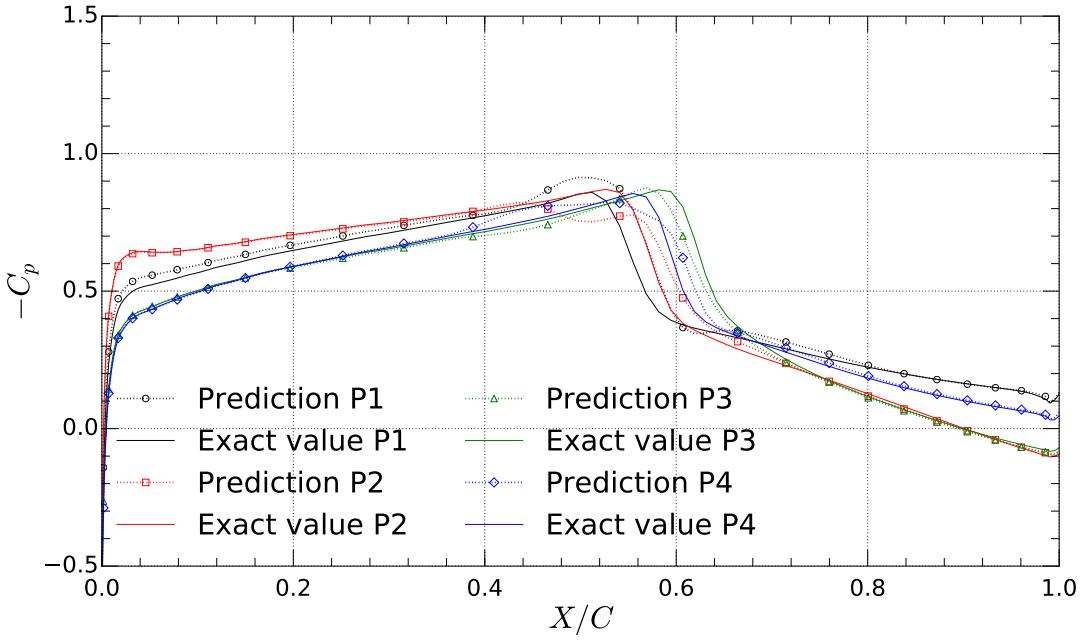


(b) LDM model

Figure 4.22: C_p profiles in the subsonic regime.

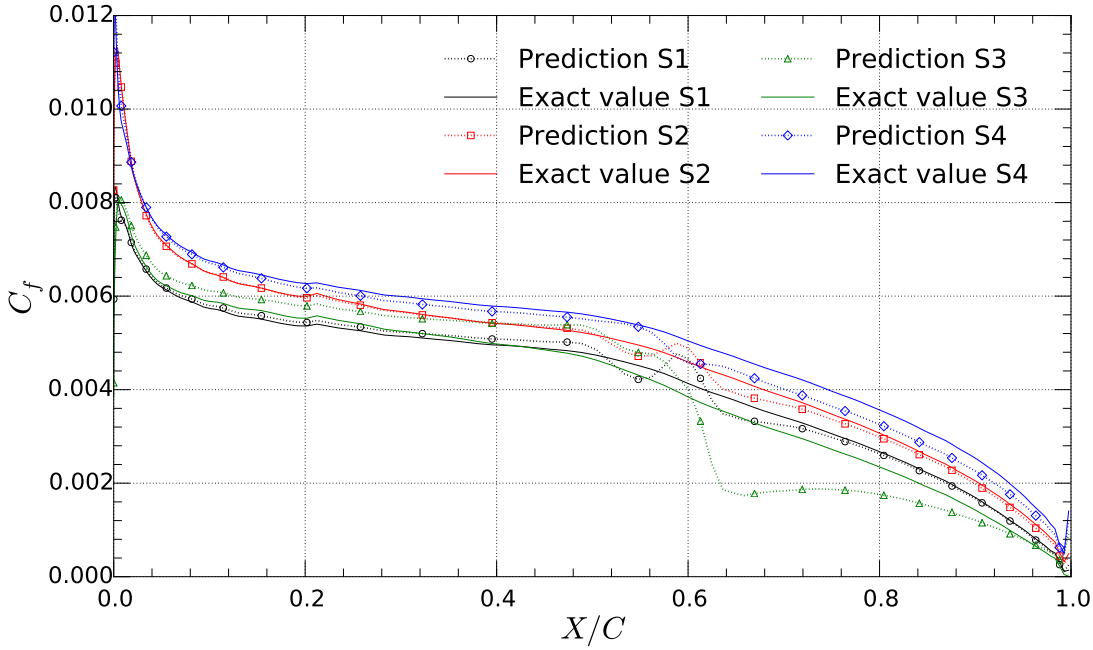


(a) Classical model

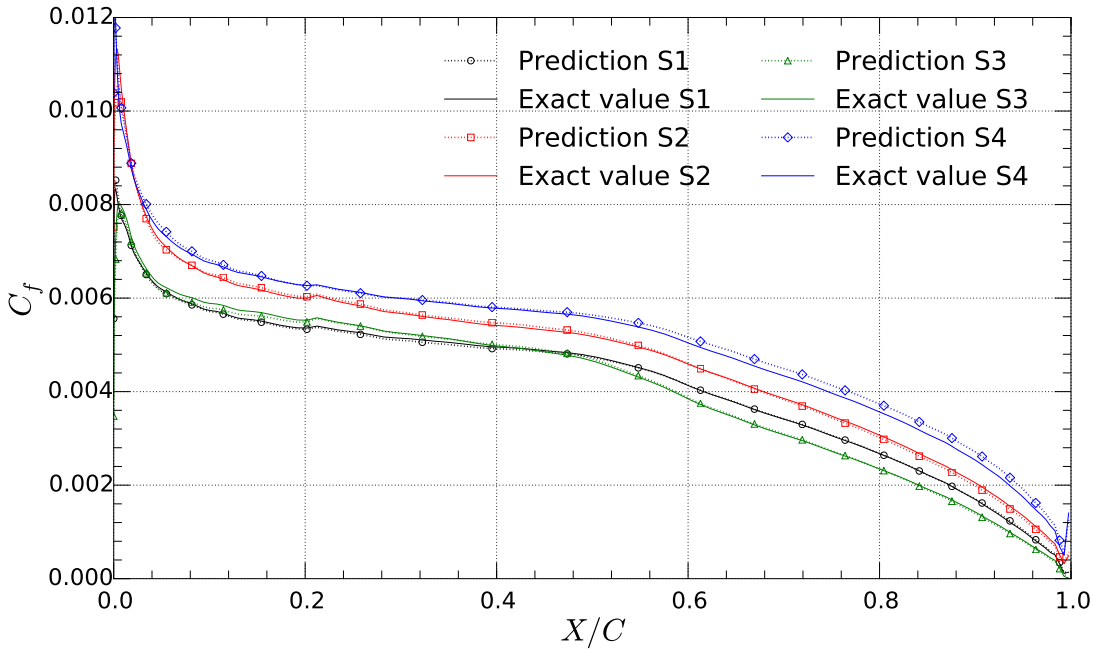


(b) LDM model

Figure 4.23: C_p profiles in the transonic regime.

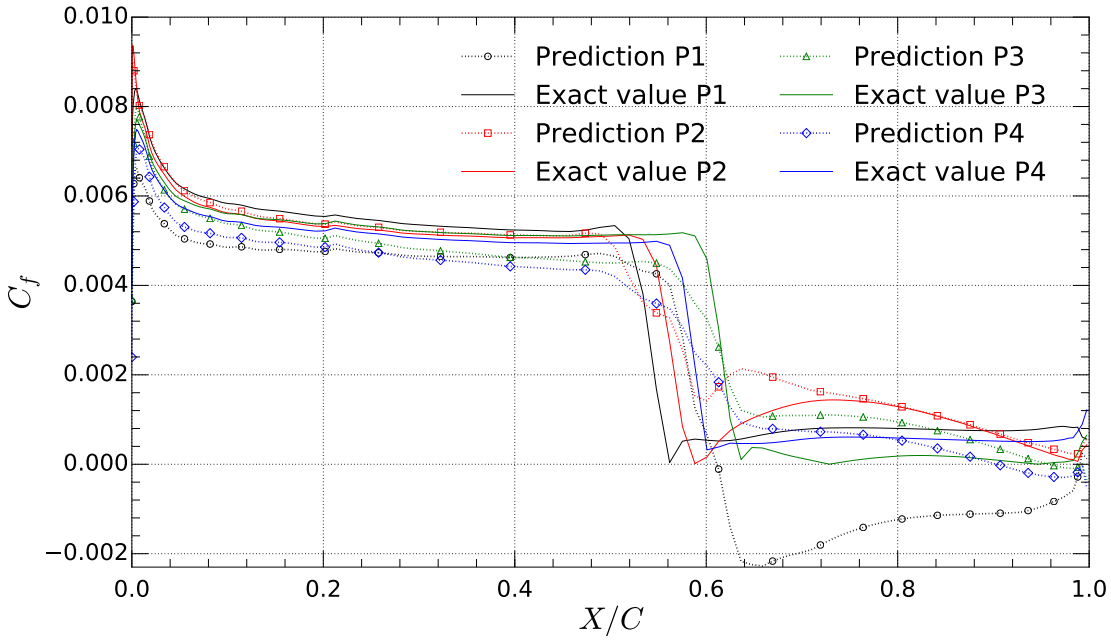


(a) Classical model

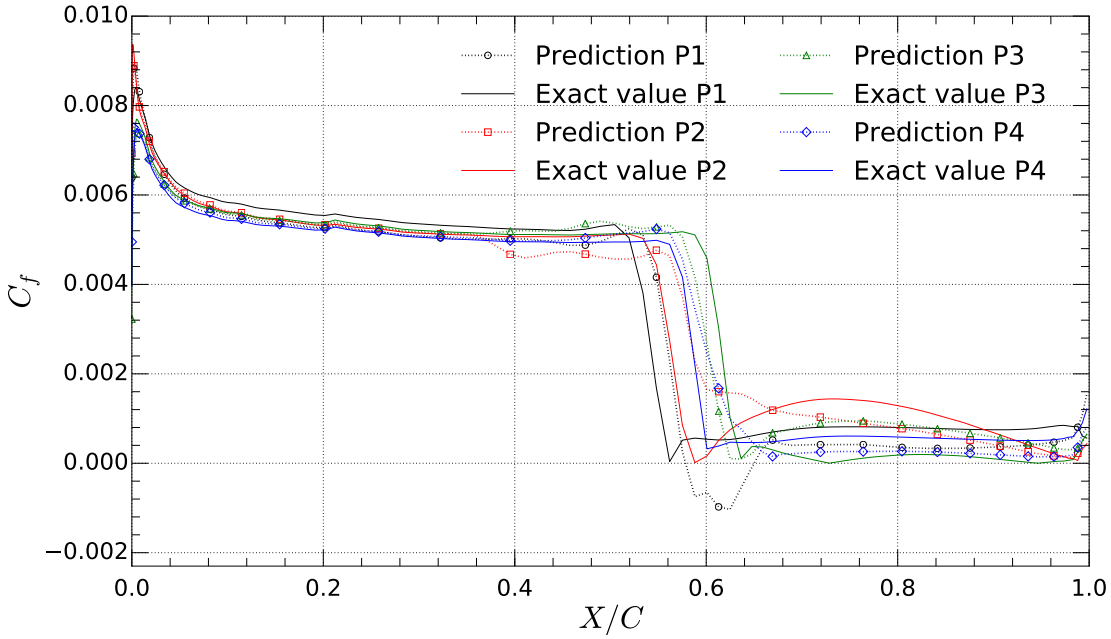


(b) LDM model

Figure 4.24: C_f profiles in the subsonic regime.



(a) Classical model



(b) LDM model

Figure 4.25: C_f profiles in the transonic regime.

4.7 Summary

The LDM approach has been developed to be more suited to hyperbolic properties than the global POD expansion. In particular, the use of a Jameson's shock sensor enhances the physical regime recognition and the resampling strategy adds automatically extra information to the subspaces with the highest discontinuous structures. The LDM has demonstrated its capabilities to build nonintrusive reduced order models capable of coping with aerodynamics problems at multiple operating conditions, which involve hyperbolic equations and different physical regimes leading to discontinuous features.

In particular, the method has been assessed on an analytical moving shock problem and on the simulation of a turbulent flow around the transonic RAE2822 airfoil. Results reveal a significant improvement of the model accuracy, especially in regions of high gradients and discontinuities.

The next chapter focuses on applying the LDM to three-dimensional geometries. The behavior of the method is deeply analyzed in order to identify best practice guidelines settings, such as a-priori analysis of data or best settings for the numerical parameters.

Part III

Applications to industrial test cases

Table of Contents

CHAPTER 5

Aerodynamic data predictions of transonic aircraft configurations

Résumé

Ce chapitre s'intéresse à la mise en œuvre de la méthode de décomposition locale sur deux cas d'applications proches des problématiques industrielles : une configuration simplifiée d'avion AS28G et la configuration XRF-1, proche d'une véritable géométrie avion et servant d'interface pour faciliter les échanges entre la recherche et l'industrie.

La complexité de ces deux applications met en exergue les capacités offertes par la LDM pour amélioration les prédictions mais aussi la difficulté à correctement paramétrer le modèle, notamment dans le choix du nombre de sous-ensemble. En effet, ce choix impacte directement la distribution des modèles locaux, leur comportement, ainsi que le rééchantillonnage. Néanmoins, il peut être montré que les outils statistiques présentés dans les chapitres précédents, comme par exemple la validation croisée, guident efficacement le choix des paramètres numériques.

Ainsi, la LDM se montre très efficace sur l'AS28G, en particulier avec deux clusters. L'analyse des modes POD illustre la séparation nette entre les régimes subsoniques et transsoniques. Cela se retrouve directement en terme de prédiction : l'erreur évaluée sur la base de test est largement réduite et l'observation de quelques prédictions réparties dans l'espace des paramètres illustre les capacités du modèle à reproduire des structures complexes de choc, ce que n'arrive pas à réaliser l'approche globale classique.

L'application sur le XRF-1 a permis une analyse fine du comportement des différents modèles selon le nombre de sous-ensembles. Quelle que soit la configuration, une large région localisée pour des faibles nombres de Mach est systématiquement identifiée. Sa taille varie peu en fonction des paramètres numériques ou de nombre d'échantillons d'entrainement. De plus, l'étude de la POD révèle une faible compressibilité, ce qui signifie que les points d'entrainement sont amplement redondants entre eux. Les autres sous-ensembles ont permis de donner plusieurs interprétations différentes des simulations faites à un nombre de Mach élevé. Globalement, les modèles de décomposition locale se sont montrés performants. En particulier, le cas avec deux

sous-ensembles a systématiquement été plus performant que le modèle global, peu importe le nombre de points d'apprentissage.

The goal of this chapter is to bridge the gap between academic methodologies and real-word applications. In particular, the work aims at validating the LDM strategy on aerodynamic problems involving three-dimensional geometries, large parameter spaces, and representative configurations. The predictions of deep transonic flows are specifically investigated on these applications. Two different cases are studied: a simplified AS28G configuration and an industrial-like XRF1 configuration. Strengths and weaknesses of the LDM are compared with the classical approach. These studies help to improve the understanding of the method and to evaluate its efficiency.

5.1 AS28G

The AS28G aircraft configuration is quite representative of an industrial configuration [33, 189]. The geometry includes an half-fuselage, a wing, a pylon and a nacelle, as illustrated in Figure 5.1. For this reason, the configuration is of special interest to the Local Decomposition Method (LDM) in order to assess its capabilities to deal with complex and industrial configurations. For all figures, the term extrados refers to the suction side and the term intrados refers to the pressure side. This section has been presented in a conference paper [53].

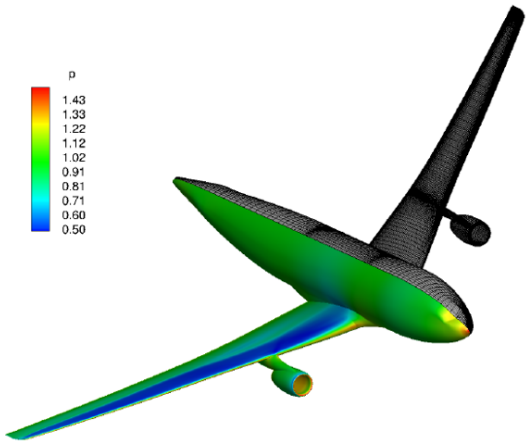


Figure 5.1: Non-dimensionalized pressure distribution on AS28G and view of the computational grid.

5.1.1 Presentation of the case

5.1.1.1 Simulation setting

The high-fidelity computations are also carried out using the cell-centered finite-volume solver elsA [30]. From the numerical point of view, the classical second order central scheme of Jameson, Schmidt and Turkel [103] is used for the space discretization. The time integration is performed with the backward Euler implicit scheme: the algebraic system is linearized with the LU-SSOR implicit method [207]. The turbulence modeling is ensured by the two equations k-ω turbulence model. The grid contains 56 blocks with a total of 3.5 million grid points and a multigrid acceleration method is applied with two level: a fine and a coarse mesh. Moreover, only a half aircraft is simulated as

a symmetric boundary condition is used. The other boundaries of the computational domain are defined by the reference conditions of the upstream flow. Adiabatic wall are also considered.

The feature of prime interest of this test case is that the inflow conditions can lead or not to the appearance of shock waves. Detecting and separating these regimes (with or without shock) represent the main challenges for the model. The pressure, which is the quantity of interest, is predicted over the entire wall surface. Its behavior can be completely different on the wing or on the fuselage.

5.1.1.2 Parameter space

Three parameters are considered: Mach number, angle of attack, and Reynolds number. Table 5.1 sums up their variation which is representative of parametric studies in aeronautical industry. A testing set of 300 snapshots from a Sobol sequence has been built in order to evaluate surrogate models: global (classical) and local (LDM) methods.

Freestream variable	Amplitude of variation
Mach number	0.4 - 0.85
Angle of attack (°)	0.5 - 3.5
Reynolds number	10^6 - 10^7

Table 5.1: Freestream conditions.

5.1.2 Construction of the surrogate model

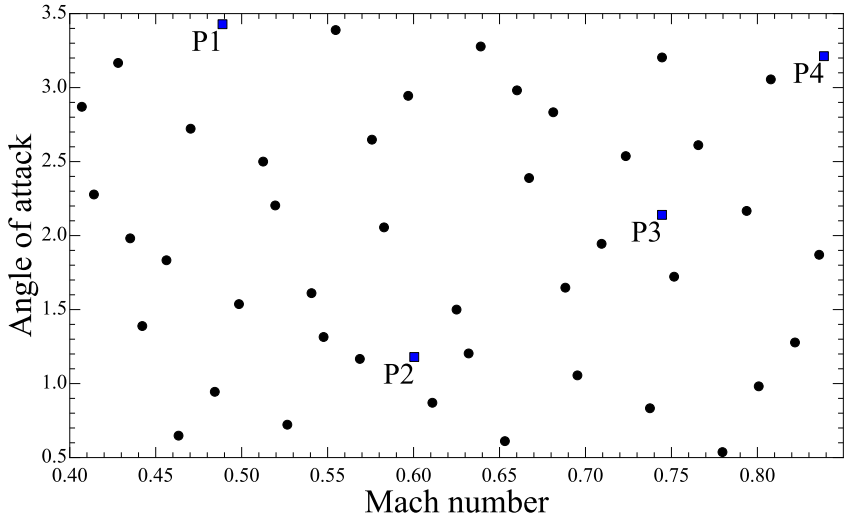
5.1.2.1 Sampling

The DOE of the classical method has been built with 40 samples of an Halton sequence, as illustrated in Figure 5.2a. The number of training samples does not deliberately follow the rule of thumb $10d$. Indeed, 10 supplementary snapshots have been added in order to better explore the transonic region which is bounded by a Mach number value of 0.85 while keeping a limited number of snapshots to assess the capability of the model close to real limitations of the computational budget. The LDM is built with an initial DOE of 20 samples in order to roughly detect the main features and to perform the first clustering. Then, 20 other samples are added during the resampling step by a batch processing of 5 computations. This resampling is performed with two different configurations:2 and 3 clusters.

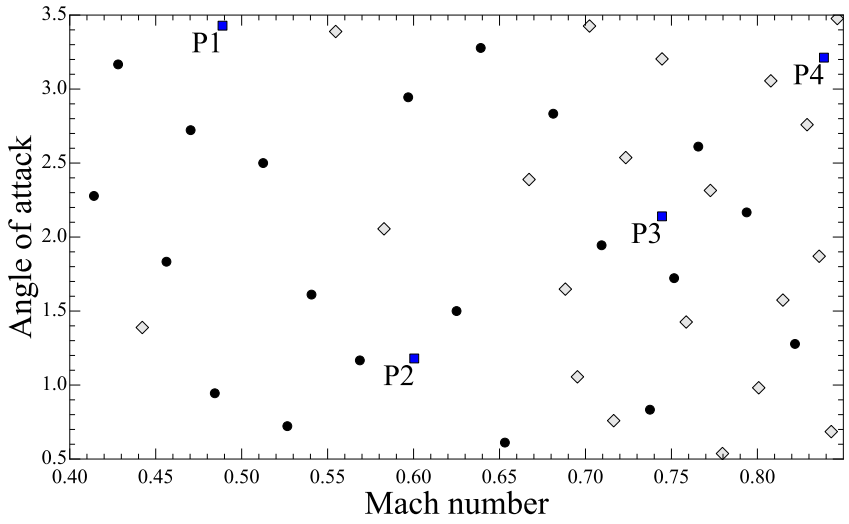
The clustering algorithm is fixed during all the resampling process in order to avoid significant changes in the shape of the cluster, as it ensures a certain coherence for the resampling. Changing the method could lead to uneven or inconsistent sampling distributions. For reasons of clarity, the DOE is represented in a two dimensional way, since the Reynolds number has much less influence than the Mach number and the angle of attack.

Figure 5.2 compares the sampling of the classical method with 40 samples from the Halton sequence and the sampling coming from the LDM with 2 clusters (20 Halton and 20 resampling points). While the Halton sequence covers uniformly the parameter space, the region at high Mach number is targeted by the resampling step of the LDM. Indeed, Figure 5.2b exhibits a resampling with 17 extra snapshots having a Mach number above 0.65, whereas only 3 snapshots have been distributed in the region of low Mach number below 0.65 with the classical method. This high-

density sampling near the transonic regime is expected to improve the surrogate model in more challenging parts of the parameter space. The sampling of the 3 clusters configuration is not shown, as the sampling for the two configurations is very similar (only two points differ).



(a) Classical method



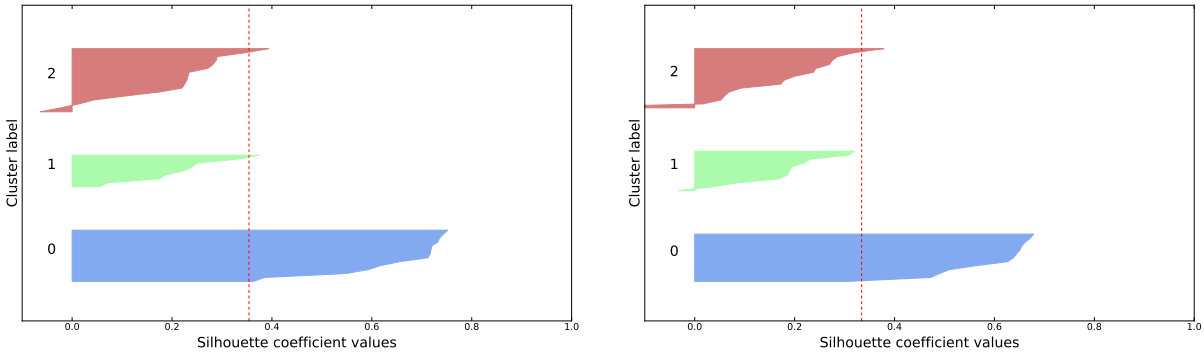
(b) LDM with 2 clusters

Figure 5.2: Sampling distribution. The black circles represent the initial DOE, the grey diamonds represent the resampling, and the blue squares represent four illustrative predictions P1, P2, P3 and P4.

5.1.2.2 Clustering and classification

The analysis of the k-graph highlights a break in slope less pronounced than for the validation cases. For this reason, the selection of the clustering strategy is studied on a case-by-case basis. Three algorithms are tested: K-means, GMM and DBSCAN. In particular, the impact of the number of clusters is evaluated for two and three clusters.

The case with two clusters gives very close results. Parameters of DBSCAN are chosen according to the graph of the k-distances. GMM and DBSCAN algorithms compute the same clusters, while the K-means algorithm finds a slightly smaller transonic cluster. Silhouette coefficient values are similar such as LOO errors. Thus, GMM is arbitrarily chosen as the clustering algorithm for two clusters. The configuration with 3 clusters gives also very similar results, except for DBSCAN which does not identify a significant third cluster. Figure 5.3 gives the distribution of silhouette coefficients for K-means and GMM. Only slight differences are observed. The K-Means algorithm is preferred as clusters #2 and #3 seem slightly more consistent with fewer negative values. Finally, it can be observed that the average Silhouette score is 20% higher for 2 clusters. Thus, the two clusters configuration can be expected to be more relevant.

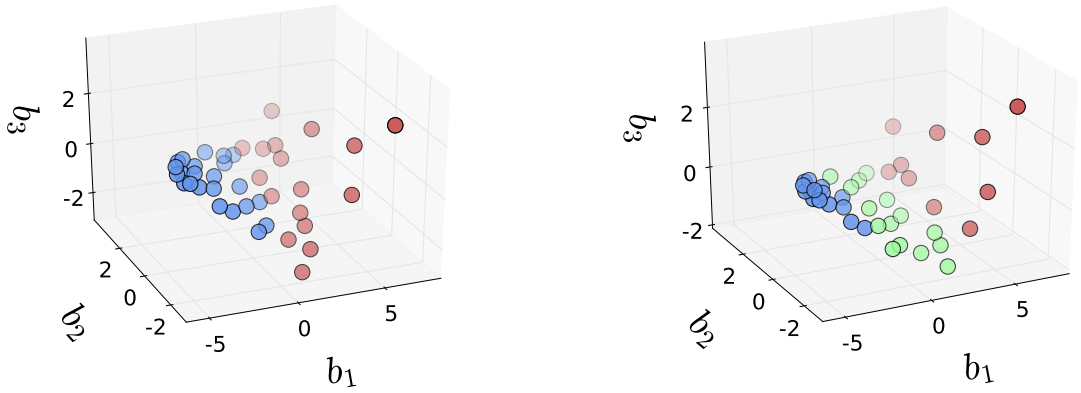


(a) K-means algorithm.

(b) GMM algorithm.

Figure 5.3: Silhouette coefficients.

In Figure 5.4, the distribution of samples for the Jameson's sensor is illustrated for the first three components of the PCA at the end of the sampling process. As regards the case with two clusters, a clear separation is observed in Figure 5.4a with a change of slope in the distribution of the sensor. The clustering with three clusters given by the K-means algorithm identifies a large transition region as shown in Figure 5.4b with the green points.



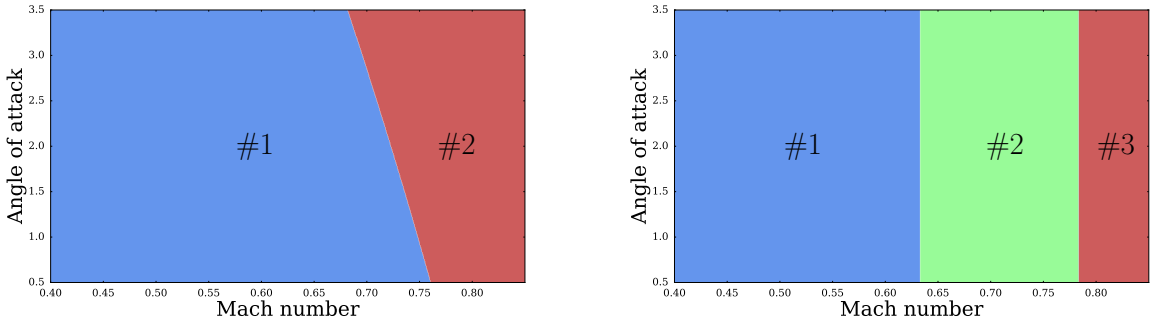
(a) Two clusters configuration.

(b) Three clusters configuration.

Figure 5.4: Clustering of the shock sensor.

Naturally, these results lead to different separation of the parameter space as illustrated in Fig-

ure 5.5, where the background color gives the cluster with the maximum probability. The cluster with the lower Mach number values in blue is called cluster #1. The red cluster corresponding to the highest values of Mach numbers is called cluster #2 for the two clusters configuration, and cluster #3 for the three clusters configuration. Finally, the intermediate cluster in green is named cluster #2.



(a) Two clusters configuration.

(b) Three clusters configuration.

Figure 5.5: Parameter space separation.

The first finding is that the size and the shape of the cluster #1 are widely impacted by the choice of the clustering strategy:

- **two clusters configuration**: the angle of attack influences the shape of the boundary decision which goes above Mach 0.75;
- **three clusters configuration**: the decision boundary of the cluster #1 is independent of the angle of attack and does not exceed a Mach number of 0.65.

Moreover, by comparing Figure 5.2b and 5.5, it may be noted for the two clusters configuration that several snapshots belonging to the cluster #1 have been added near the decision boundary during the resampling. It can be explained by a moving decision boundary. Indeed, new snapshots update the classification of the parameter space every 5 computations. Thus, specific snapshots have been estimated as being in the cluster #2 but then they may change the cluster as the resampling progresses. Obviously, this problem also occurs for the three clusters configuration.

5.1.2.3 POD analysis

This section analyses the dimension reduction step. Table 5.2 shows the main parameters for the training set. First of all, it appears the clusters with the strongest non-linearities (the last cluster of each configuration) exhibits a much higher entropy of 0.73 and 0.74 than the global approach with only 0.51. The same trend was found for the RAE2822 case. Thus, it suggests once again specific modes have been discovered by exploring the clusters during the resampling step. Moreover, the first cluster has data with a low entropy of respectively 0.41 and 0.40 for two clusters and three clusters, meaning data in this region contain a high redundancy. Finally, as might be anticipated, the cluster #2 of the three clusters configuration has an intermediate trend with an entropy of 0.64 between the low values around 0.40 and the high values around 0.73, but still higher than the entropy of the global model. That confirms clearly its role of buffer region between the cluster of continuous responses and the cluster with high discontinuous responses.

Number of cluster	Cluster index	Snapshots	Modes	Entropy
1	-	40	18	0.51
2	#1	24	11	0.41
	#2	19	16	0.73
3	#1	16	6	0.40
	#2	15	12	0.64
	#3	10	9	0.74

Table 5.2: Dimension reduction behavior for the training set.

POD has also been performed on the testing set for comparison purposes. The data are summarized in Table 5.3, bearing in mind the data come from a Sobol sequence of 300 snapshots without resampling (for both global and LDM). Similar trends are observed for the entropy, except all the values decrease as the number of snapshots by cluster widely rises. Thus it validates the clustering and parameter space decomposition obtained with the training samples.

Number of cluster	Cluster index	Snapshots	Modes	Entropy
1	-	300	79	0.44
2	#1	194	16	0.25
	#2	106	73	0.63
3	#1	137	11	0.25
	#2	90	29	0.43
	#3	73	56	0.66

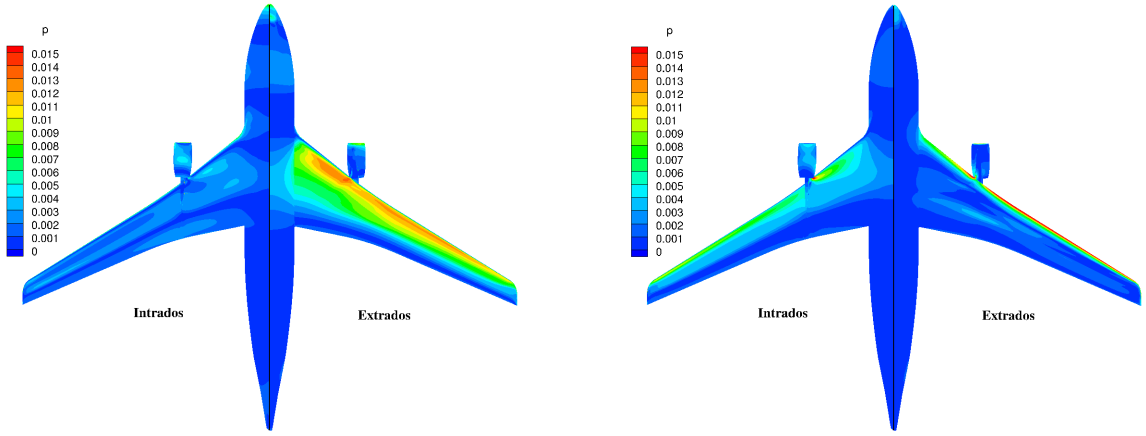
Table 5.3: Dimension reduction behavior for the testing set.

The first two POD modes are specifically analysed and compared. They are displayed in Figure 5.6 for the global approach and in Figure 5.7 and 5.8 for respectively the two and three clusters configurations. The mode shapes appear as very similar between the global approach and the clusters #1 of the two configurations. In particular, the first mode shows a large dominant structure localized on the suction side. Due to these similarities, the other clusters are expected to bring a finer representation of complex flows as they capture more significant modes than the global model.

As regards the two clusters configuration, the first mode of the second cluster highlights a strong amplitude between the trailing edge and the recompression location on the suction side. This shape shows similarities with the second mode of the cluster #3. Furthermore, the first mode of this latter exhibits a particular behavior, with a high amplitude near the wing tip. These modes play a central role in the reconstruction of discontinuities by surrogate models.

Finally, the modes of the intermediate cluster can be interpreted as an amplification of those of the first cluster. Indeed, the first mode shows a strong amplitude near the leading edge while the second mode highlights very large structures on both sides of the wing.

In conclusion, the POD analysis highlighted capability of LDM to extract representative modes of specific flow shapes. Resampling enhances the training set by adding extra simulations in regions with complex flows. Thus, these news features creates POD models able to provide finer representations of complex flows.



(a) Mode No.1 (b) Mode No.2

Figure 5.6: First and second POD modes of the global model.

5.1.2.4 A priori error

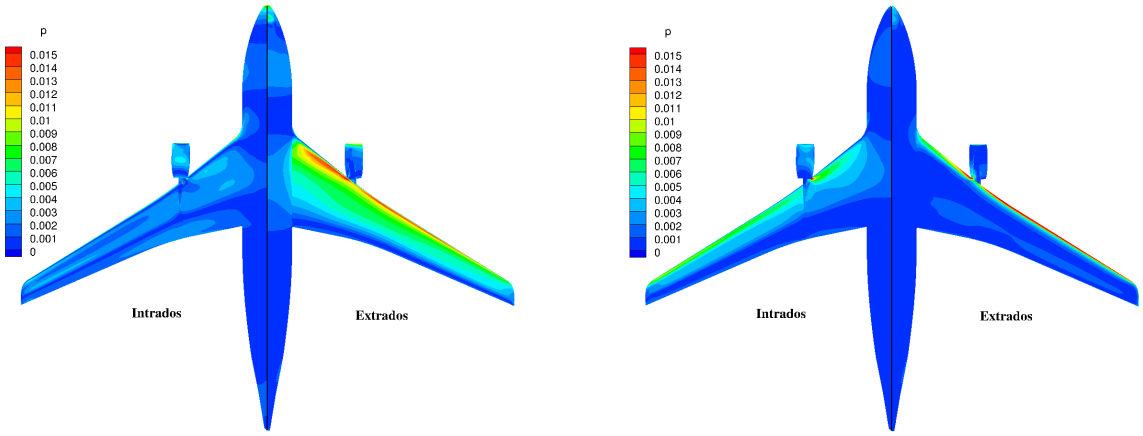
During the sampling process, the LOO predictivity coefficient Q_2 has been computed for both methods: on all the snapshots for the global approach and locally for the LDM with the two configurations. The results are illustrated in Figure 5.9. The cluster #1 and the global method show high values of Q_2 with a smooth evolution. Indeed, few snapshots have been added to cluster #1 during the resampling, thus it was expected that the Q_2 remains stable. Analyzing the three clusters configuration in Figure 5.9b reveals an important variation of the LOO error for the cluster #3 between 20 and 30 training samples with a negative value of Q_2. Then, the values tend to stabilize. Hence, these results express a high degree of uncertainty regarding the quality of the surrogate model for the transonic part and a high reliability of the predictions in the subsonic part for the LDM.

It may be thought that the predictivity coefficient for the global method is overestimated in regards to the LDM with the two clusters configuration in Figure 5.9a and the three clusters configuration in Figure 5.9b. Indeed, this observation emphasizes the main weakness of the a priori validation of the surrogate by a LOO approach which is its high dependence on the training sampling. As the resampling step focuses mainly on challenging regions, the LOO Q_2 necessarily drops but leads to more representative values of Q_2 compared to the global method where the samples cover evenly the parameter space.

In the light of these results, the two clusters configuration seems to be more reliable with lower variations of the Q_2. However, it would be interesting to evaluate the behavior of the LDM in the 3 clusters configuration but with 50 training samples in order to properly train the local models on each subsets.

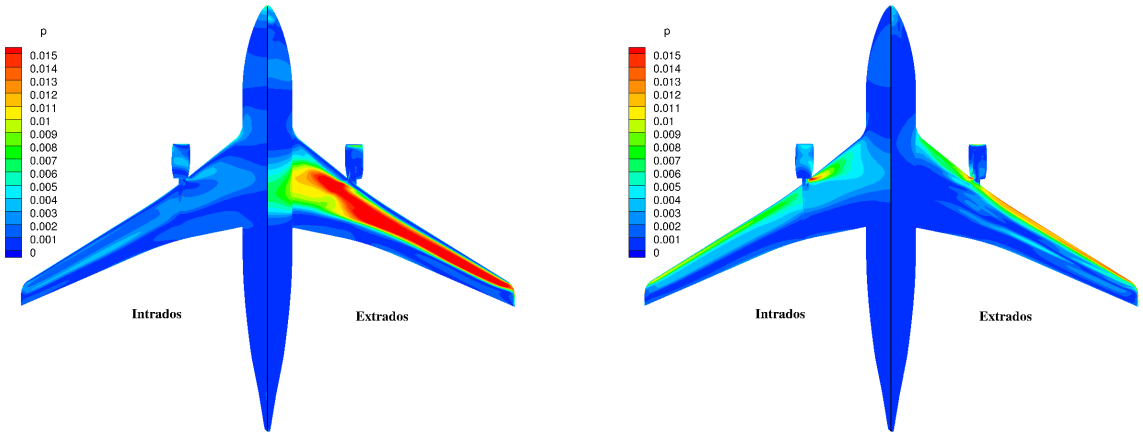
5.1.3 Global accuracy

In this section, a more detailed look is given to the analysis of the model accuracy. Figure 5.10 and 5.11 show the spatial distribution of the RMSE and the predictivity coefficient Q_2 averaged on all the predictions. In both cases, two comments may be made: the suction side of the wing appears as the more challenging part and the LDM increases the accuracy of the surrogate model for both configurations. Nevertheless, the location of the maximum error differs with the error



(a) Mode No.1 (b) Mode No.2

Cluster #1 for the two clusters configuration



(c) Mode No.1 (d) Mode No.2

Cluster #2

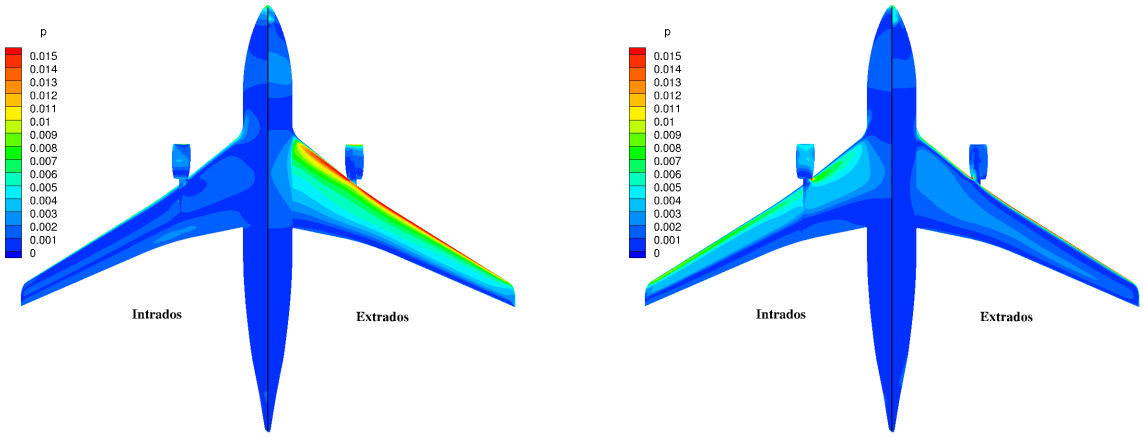
Figure 5.7: First and second POD modes of the clusters for the LDM.

metric.

Looking at the RMSE displayed in Figure 5.10, the main errors are located broadly between the leading edge and 70% of chord on the suction side, corresponding to the region of strong recompression gradients. This confirms that surrogate models have trouble predicting accurately the location and the amplitude of the sharp gradients and shock waves. Nevertheless, these troubles are limited with the LDM approach for two and three clusters.

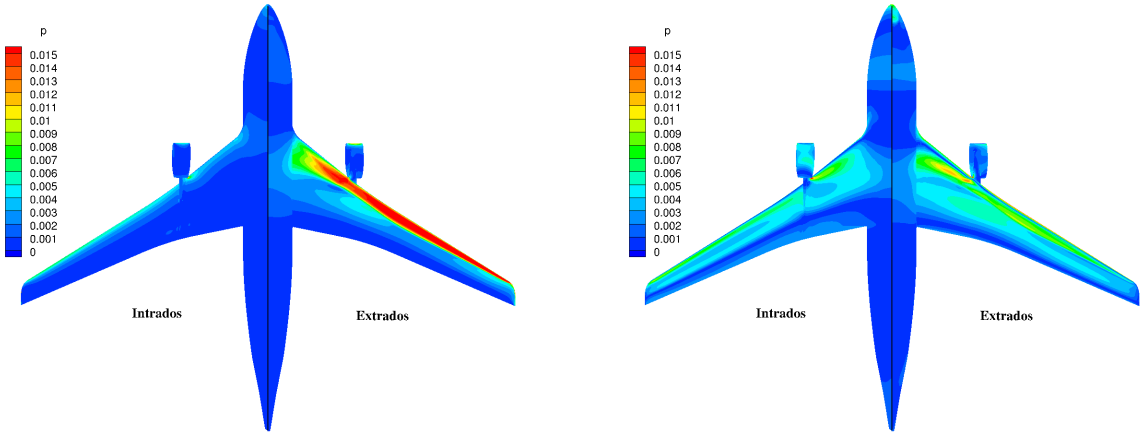
As regards the predictivity coefficient Q_2 illustrated in Figure 5.11, a different location of the error is highlighted mainly after the recompression on the suction side of the wing and at the rear of the fuselage on the upper part. The main explanation for this difference is that the predictivity coefficient weights the error values by the inverse of the variance. Thus, the regions of low quality near the trailing edge and the wing tip couple non-negligible errors with low variances. Nevertheless, the LDM also achieves better performances: the lowest values of the predictivity coeffcient are around 0.5 while the global model shows negative values.

The superiority of the LDM, observed with the spatial distribution of the error, is confirmed by looking at the the statistical distribution of the absolute error in term of pressure coefficient C_p, averaged on the mesh. Results are presented in Figure 5.12 with a box plot formalism. The



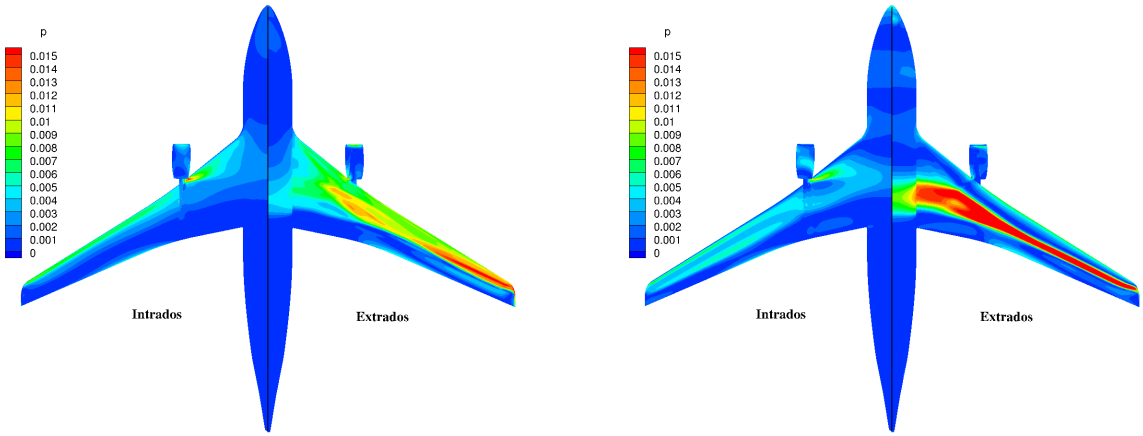
(a) Mode No.1 (b) Mode No.2

Cluster #1



(c) Mode No.1 (d) Mode No.2

Cluster #2

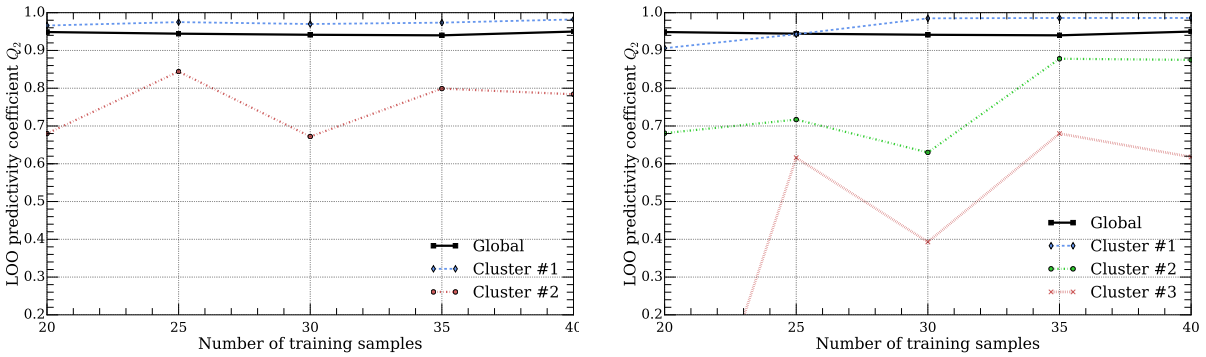


(e) Mode No.1 (f) Mode No.2

Cluster #3

Figure 5.8: First and second POD modes of the clusters for the LDM.

absolute error is associated to each testing snapshot and then statistical quantities such as mean, median or quantiles are computed. Three different phases have been considered:



(a) Two clusters configuration.

(b) Three clusters configuration.

Figure 5.9: Leave One Out Error.

- the full domain contains all the samples of the testing set;
- the subsonic regime is only composed of the testing samples belonging to the cluster #1;
- the transonic regime encompasses the other snapshots.

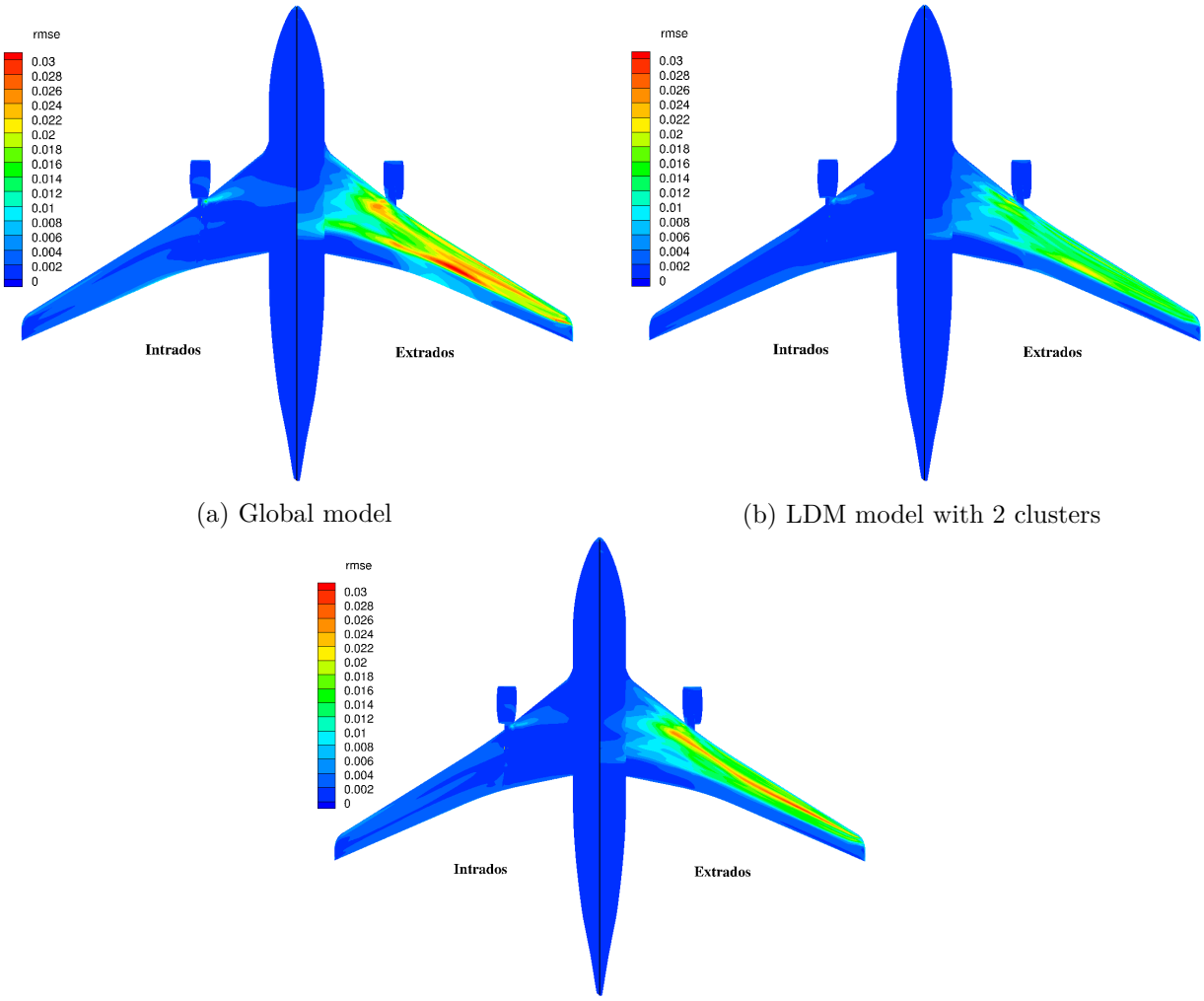
A significant improvement in the accuracy is induced by the LDM, in particular with the two clusters configuration. As regards the full domain, the normalized error decreases dramatically for all the statistical characteristics of the box plot. In particular, the mean error of the LDM falls by more than 30% compared to the classical method. Moreover, the amplitude of the box plots decreases for all phases, illustrating a large reduction of the model variability.

Box plots for subsonic and transonic regimes provide a closer look at the error distribution. It clearly appears that the LDM improves the predictions at transonic regime, explained by the increase of the sample density. The error for the subsonic part decreases as well, although samples have been removed, compared with the global model. However, one notices that the three-clusters configuration is globally less accurate than the two-clusters configuration for all the phases and all the statistics of the box plot. Two possibilities may explain this deviation. First, as the number of clusters increases, the number of interfaces also rises. However, the latter are very challenging for the GPR models due to the extrapolation. Secondly, the number of training samples is also reduced on each cluster and it has a negative impact on the training of the GPR, in particular for hyperparameters.

All these results confirm the relevance of the LDM for the two and three clusters configurations. As it was suggested in section 5.1.2.3, the greatest variety of the local POD modes is more representative of the flow for both physical regimes, reducing the sensibility to interpolation errors and increasing the accuracy of the surrogate model.

5.1.4 Specific predictions

This section illustrates specific predictions of the LDM and the global model at 4 locations in the parameter space: the points $P1$, $P2$, $P3$, and $P4$ are identified in the Figure 5.2. The pressure coefficient is computed at 7 different spanwise sections η on the wing: 0.173, 0.289, 0.416, 0.478, 0.570, 0.780, and 0.912. The results for the 4 predicted points are illustrated in Figure 5.13, 5.14, 5.15, and 5.16 for respectively the points $P1$, $P2$, $P3$, and $P4$. The chord of the wing is written c and X refers to the horizontal coordinate.



(a) Global model

(b) LDM model with 2 clusters

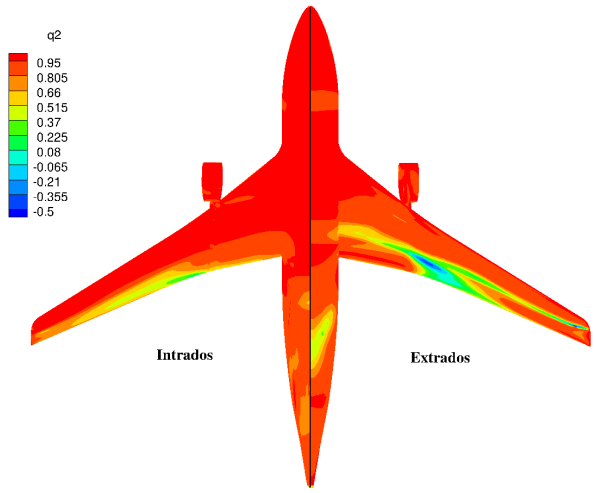
(c) LDM model with 3 clusters

Figure 5.10: Averaged spatial distribution of the RMSE.

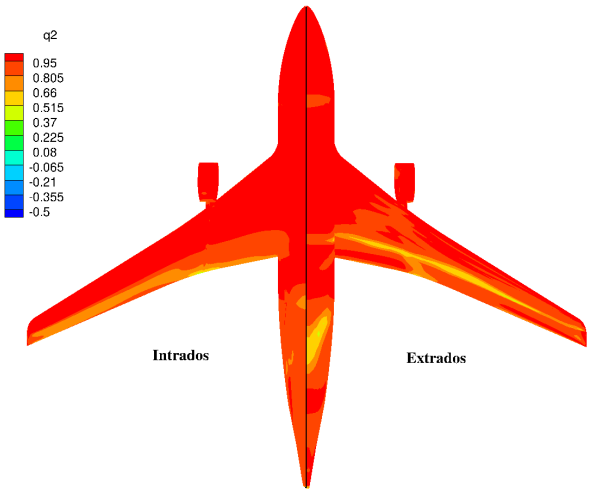
Several characteristic features are observed. As regards the two points P1 and P2, in the subsonic regime, the predictions and the CFD are in close agreement for both models. Therefore, the POD modes representative of transonic flows do not impact the predictions for the global model. In addition, even if the number of snapshots is reduced in the subsonic regime, the quality of the predictions does not decrease for the LDM. It can be noticed that the global method has slight discrepancies with the CFD on the suction side.

The predictions at P3 and P4 are more challenging for the surrogate model. The accuracy drops dramatically with the presence of the shock. In particular, the behavior of the shock is poorly predicted by the global method in terms of intensity and location. The extrapolation of the point P4 by the global method, due to the DOE, exacerbates the error around the shock. However, it can be noted that the rest of the pressure signal shows smaller errors. As regards the LDM, these two transonic points are very well predicted at all spanwise sections. Only small discrepancies are observed between CFD and surrogate model.

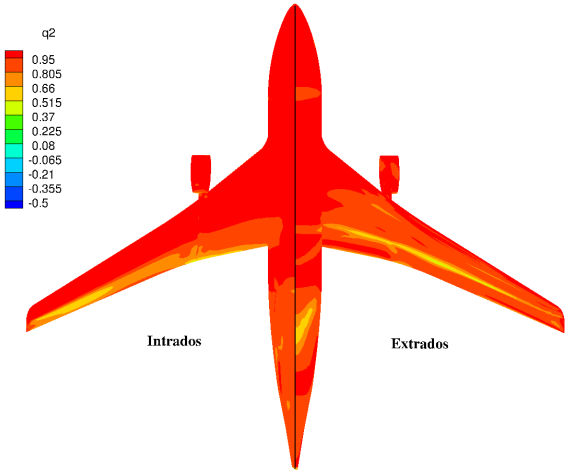
These four specific predictions are an excellent example illustrating the performance of the LDM to deal with external aerodynamics. Shocks are well predicted in terms of amplitude and position for transonic flows and subsonic flows do not encountered spurious resulting shocks.



(a) Global model



(b) LDM model with 2 clusters



(c) LDM model with 3 clusters

Figure 5.11: Averaged spatial distribution of the Q_2.

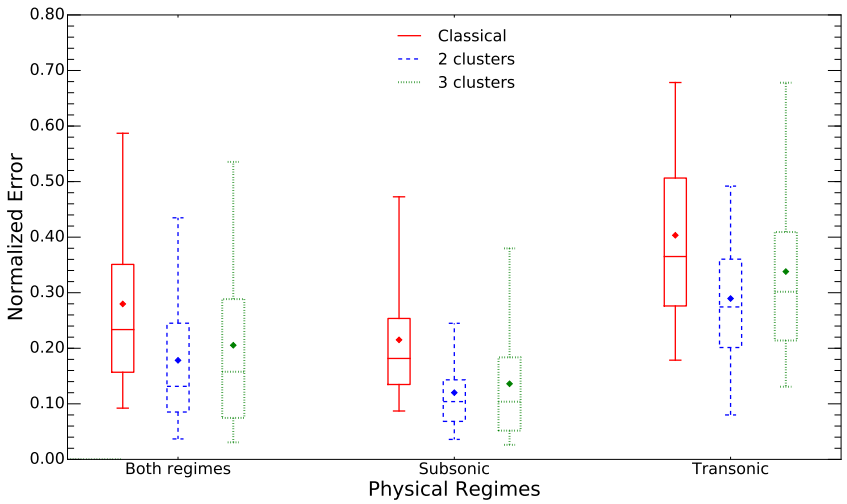


Figure 5.12: Distribution of the normalized error along the snapshot.

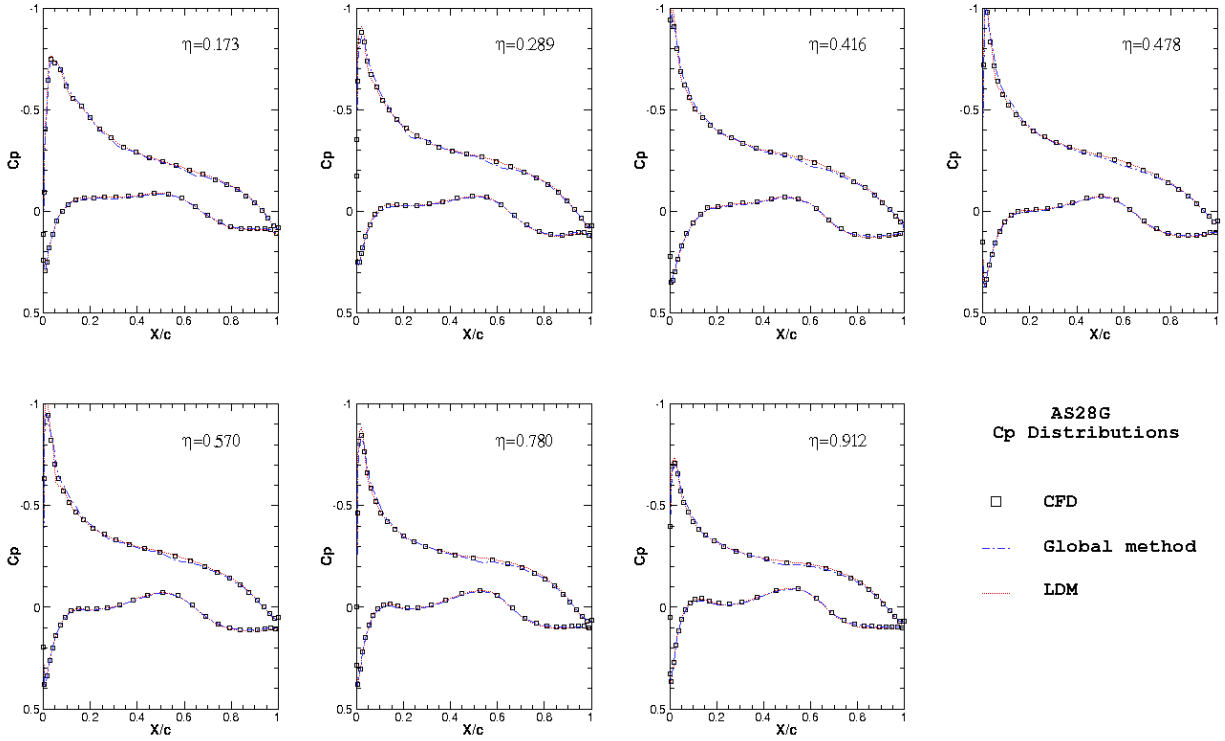


Figure 5.13: Prediction P1 - Mach = 0.489, $\alpha = 3.43$, and Re = 8.5×10^6.

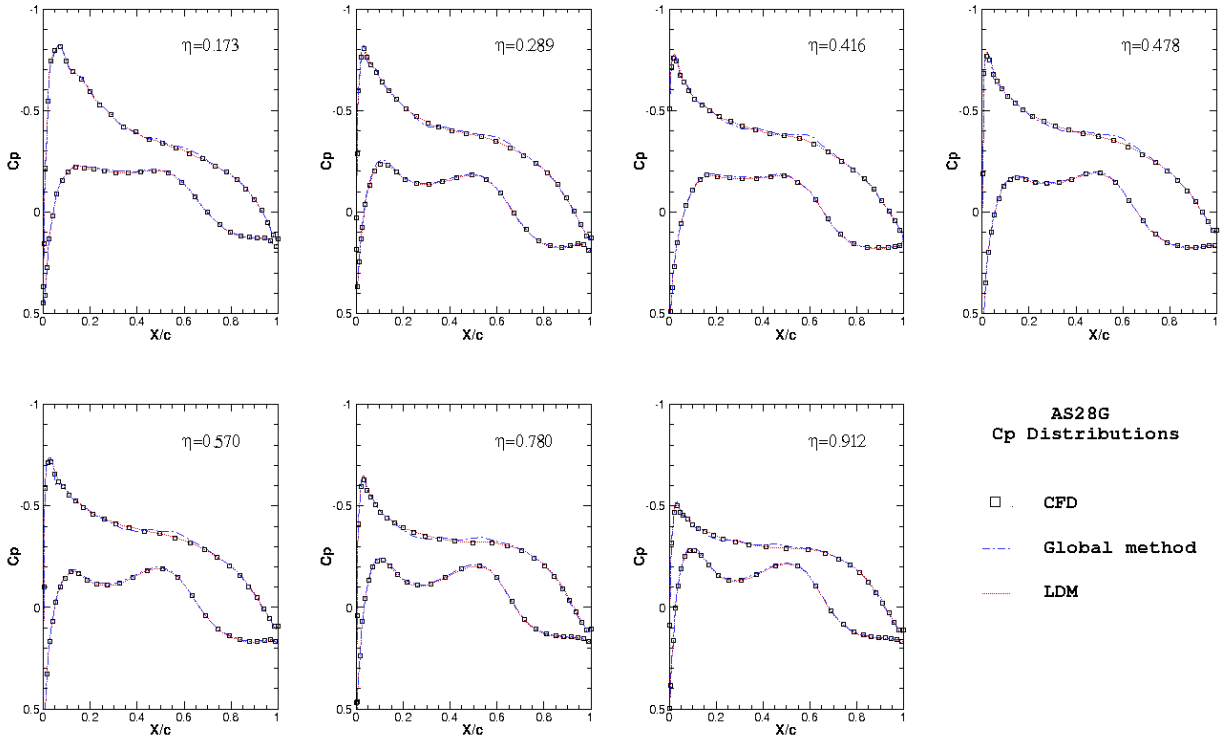


Figure 5.14: Prediction P2 - Mach = 0.60, $\alpha = 1.18$, and Re = 6.26×10^6.

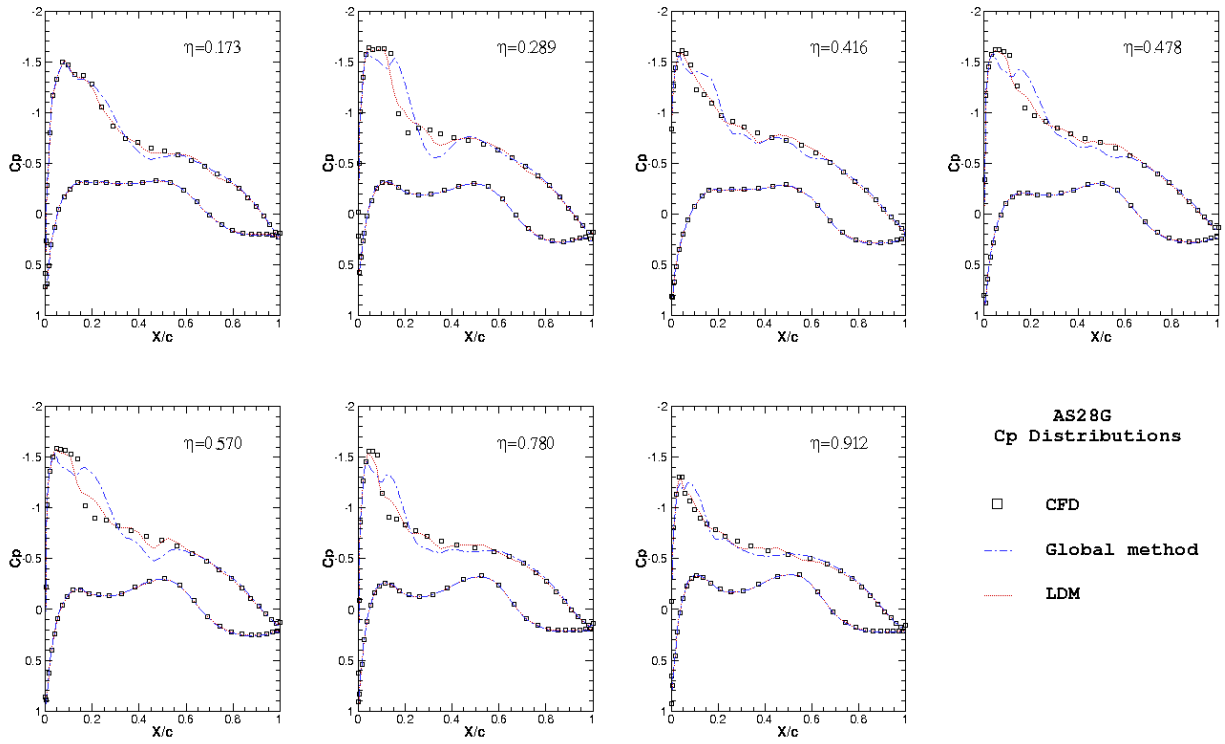


Figure 5.15: Prediction P3 - Mach = 0.744, $\alpha = 2.14$, and Re = 2.8×10^6.

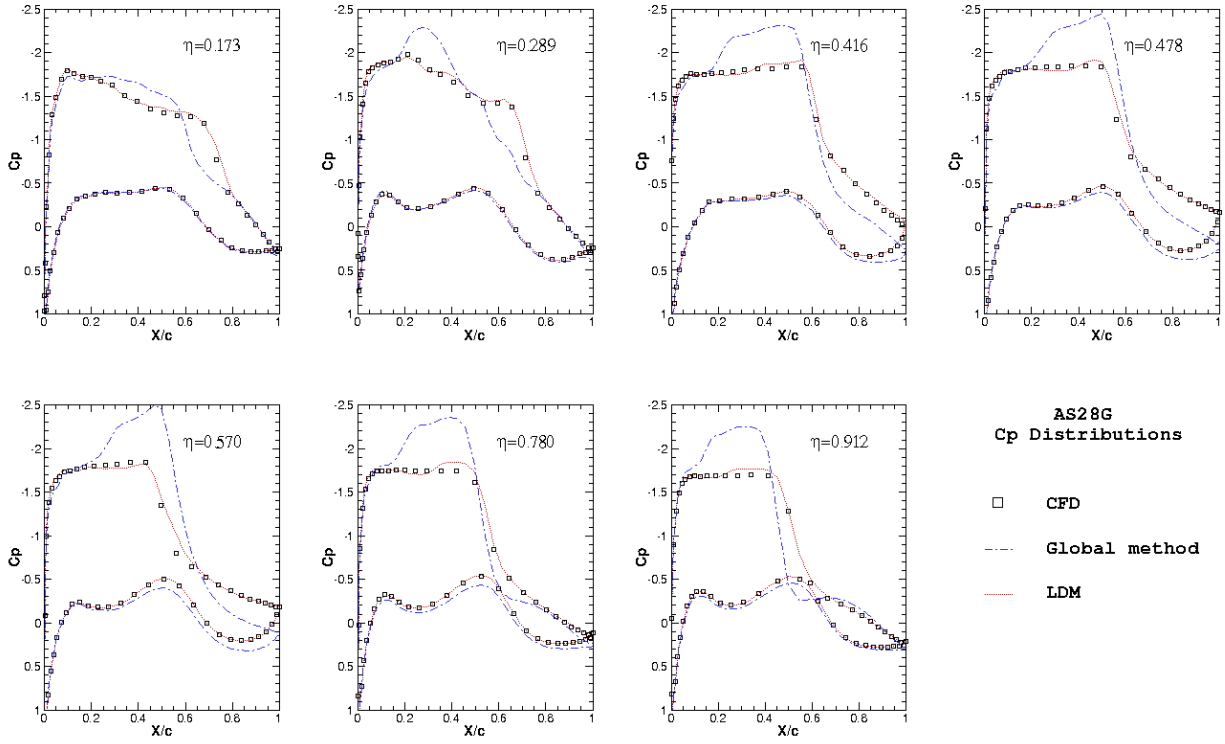


Figure 5.16: Prediction P4 - Mach = 0.839, $\alpha = 3.21$, and Re = 4.34×10^6.

5.2 XRF-1

The Airbus large transport aircraft of the eXternal Research Forum, also called the XRF-1, has been developed to facilitate exchanges between academia and industry. It is used in this section in order to assess the LDM developed in the thesis by taking into account industrial constraints and input data. The XRF-1 is a wing-body configuration, representative of modern civil transport aircraft. It is designed to carry about 350 passengers with a maximum range of 8000 nautical miles [11]. This reference configuration has already been used as a use case for the DLR project Digital-X [121] and for the MDA-MDO project [80].

The XRF1 presents a greater complexity than the AS28G configuration mainly due to number of input variables, leading to large parameter space, coupled with an important computational cost. Therefore, a deep study is dedicated to the sensibility of the parameter space decomposition against the number of training samples for 2, 3, and 4 clusters. The primary goal is to understand the different clustering results and to interpret them. This study is focused on the wing in order to analyse deeply the regions encountered different physical regimes.

5.2.1 Presentation of the case

5.2.1.1 Geometry and mesh

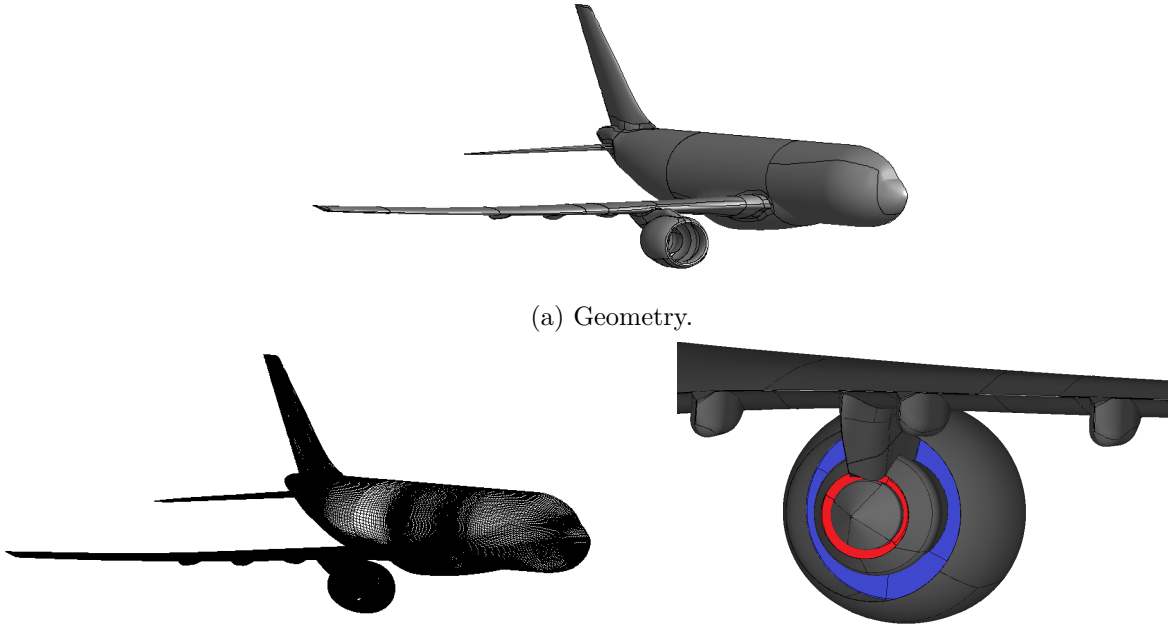
A computer-aided design visualization of the XRF-1 configuration is shown in Figure 5.17a and highlights the relative complexity of the geometry. Indeed, various systems are represented such as vertical and horizontal tail planes or the flap track fearings. This geometry is used for all flight phases. High lift devices are not considered with a view to simplification. Regarding the mesh, the XRF-1 configuration contains more than $120,000,000$ nodes for 324 blocks. The size of the domain is about 50 chords. A specific wall treatment has been applied in order to have a dimensionless wall distance in the order of magnitude of the unity. Injection conditions are imposed on the engine exit face and on the fan exit face, depicted in Figure 5.17c respectively in red and blue. An illustration of the surface mesh is given in Figure 5.17b. Moreover, only a half aircraft is simulated as a symmetric boundary condition is used. The other boundaries of the computational domain are defined by the reference conditions of the upstream flow. Adiabatic wall are also considered.

5.2.1.2 Solver settings

The simulations are also carried out using the solver elsA. From the numerical point of view, the space is discretized by a first order Roe's scheme to improve the stability of the computations. The time integration is performed with the backward Euler implicit scheme: the algebraic system is linearized with the LU-SSOR implicit method. The turbulence modeling is ensured by the model of Spalart and Allmaras and a multigrid strategy on three mesh levels is employed.

5.2.1.3 Parameter space

The generation of the XRF-1 model involves seven different parameters: Mach number, angle of attack, Reynolds number, two total pressures at the exit of the core, and two total temperatures at the exit of the fan. The stagnations quantities are used to defined injection boundary conditions using a characteristic relation and the Newton method. Table 5.4 sums up their variations which are representative of parametric studies in aeronautical industry. The engine parameters are nondimensionalized.



(a) Geometry.

(b) Mesh.

(c) Zoom on engine boundary conditions.

Figure 5.17: XRF1.

A testing set of 190 snapshots from a Sobol sequence has been built, corresponding to roughly 400,000 CPU hours. The parameter space is truncated in the region of Mach numbers and high angles of attack. These points are not representative to a real flight mission and they encounter very important unsteadinesses.

Freestream variable	Amplitude of variation
Mach number	0.1 - 0.87
Angle of attack (°)	0.0 - 6.0
Reynolds number	2.0×10^6 - 9.5×10^7
T_t core	2.4 - 3.2
P_t core	1.0 - 1.5
T_t fan	1.0 - 1.3
P_t fan	1.0 - 1.6

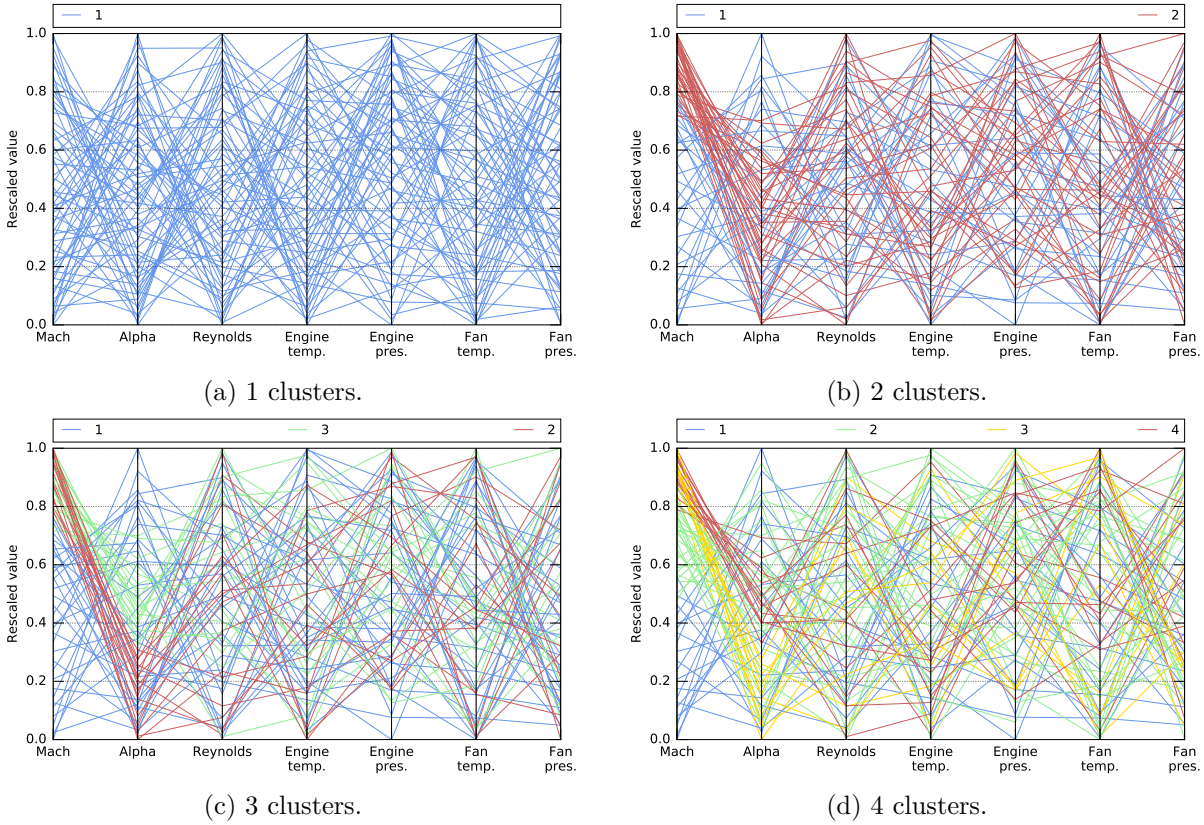
Table 5.4: Freestream conditions.

5.2.2 Building surrogate models

5.2.2.1 Sampling

Final surrogate models are built from 70 training samples, following the rule of thumb $10d$. Figure 5.18 depicts the distribution of the training samples in a parallel coordinate formalism for the global approach and the LDM with 2, 3, and 4 clusters. It includes the resampling for the LDM with 20 initial simulations and the rest for the resampling. The most striking feature is the correlation between the Mach number and the clustering. By looking at Figure 5.18b, 5.18c, and 5.18d, a clear clustering is observed on the Mach number axis, which is not the case for the other axes

with the exception of the angle of attack. Indeed, in some cases certain clusters can also be noticed on the angle of attack axis. As a consequence, the following analyse pays special attention to the Mach number and the angle of attack.



(a) 1 clusters.

(b) 2 clusters.

(c) 3 clusters.

(d) 4 clusters.

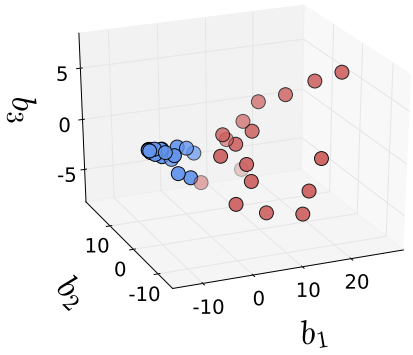
Figure 5.18: Distribution of the training samples in a parallel coordinate formalism. The belonging to the different cluster is represented in color.

The resampling process behaves here in a similar fashion to the RAE2822 and the AS28G. Comparing Figure 5.18a, 5.18b, 5.18c, and 5.18d highlights that the resampling focuses on regions of high Mach numbers. Indeed, the sampling of the LDM, irrespective of the number of clusters, is much more dense above a rescaled Mach number of 0.6 while, on the contrary, the sampling below a value of 0.6 is very sparse compared to the classical method. An indirect effect can also be observed: as the high angles of attack have been limited to Mach numbers below 0.4 to avoid unsteadiness, the resampling process avoids these specific values of angles of attack.

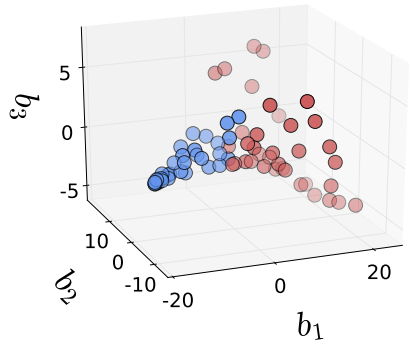
5.2.2.2 Clustering and classification

As the purpose of this section is to analyse and to compare the behavior of the LDM for different numbers of clusters, the K-means algorithm appears to be a suitable choice. Indeed, it ensures quite homogeneous clusters, even for an important number of clusters. The DBSCAN method is limited to 2 clusters by fixing the numerical parameter with the Elbow method while the GMM was less stable on the AS28G configuration for a model with 3 clusters. Moreover, the average Silhouette coefficient values are close for all configurations: between 0.41 and 0.43. Therefore, the K-means algorithm is preferred to perform the clustering.

Figure 5.19 depicts the clustering of the shock sensor for different numbers of clusters and

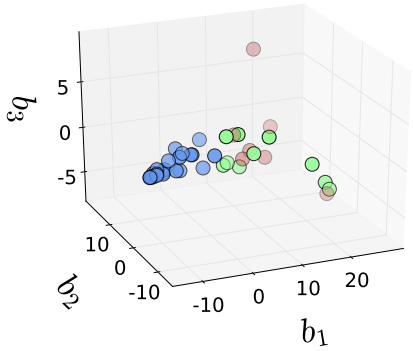


(a) 40 training samples.

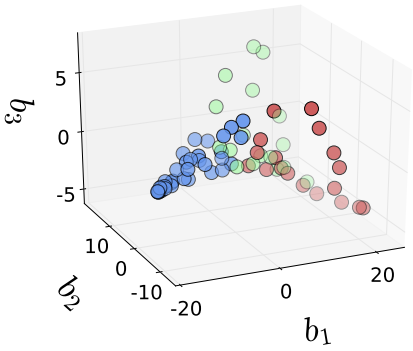


(b) 70 training samples

2 clusters

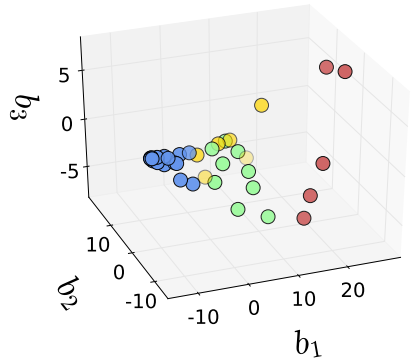


(c) 40 training samples for 3 clusters

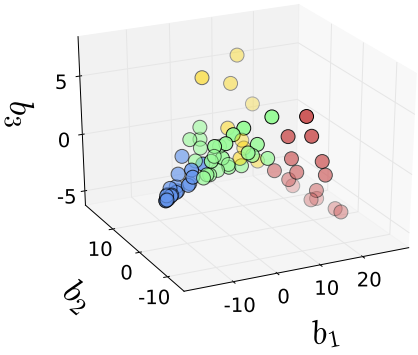


(d) 70 training samples for 3 clusters

3 clusters



(e) 40 training samples for 4 clusters



(f) 70 training samples for 4 clusters

4 clusters

Figure 5.19: Clustering with different numbers of clusters and training samples.

two numbers of training samples. A clear pattern is observed with 40 training samples: some clusters are concentrated in a specific region (the blue dots) while the others are spread across a manifold. Thus, the clustering of the spread snapshots gives quite sparse clusters. Nevertheless, the resampling corrects the problem by adding information in these clusters, as shown by the

clustering with 70 training samples. Figure 5.20 illustrates the classification of the parameter space for different numbers of clusters and training samples. Its analysis aims to improve the understanding of the method. The case with 4 clusters and 30 training samples is not represented as some clusters have insufficient size. First, it can be noticed that the classification remains quite stable for 2 and 4 clusters. On the contrary, the decomposition of the parameter space for 3 clusters changes significantly as the number of training samples increases. In particular, the green and red clusters associated to high Mach numbers constantly change. Clear size and location are not identified.

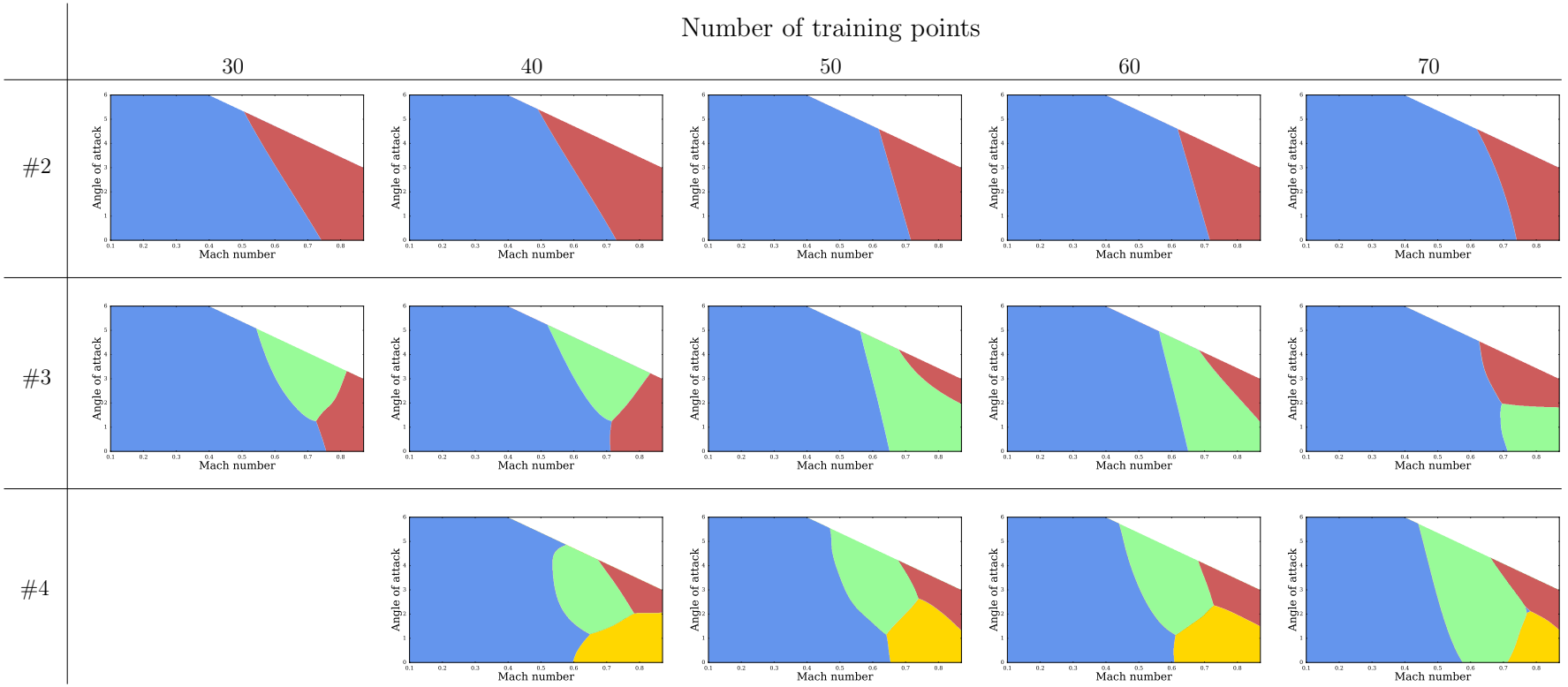
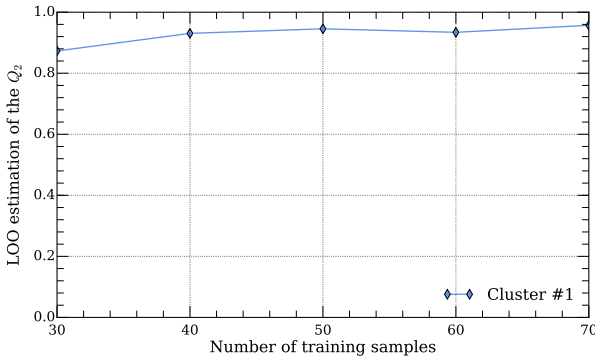


Figure 5.20: Evolution of the parameter space decomposition.

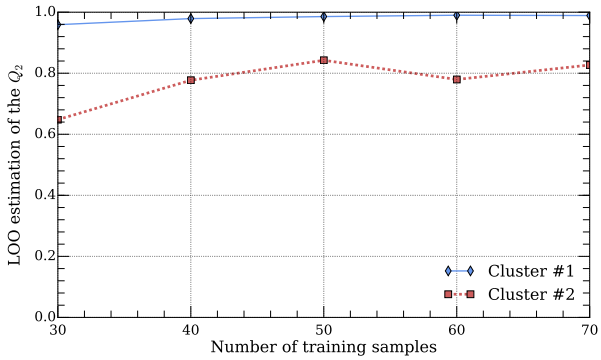
As regards the particular classifications with 70 training samples, several observations are proposed:

- a region of low Mach number, represented in blue, is clearly founded for all the models. Only the location of the boundary changes;
- the classification given by the configuration with 2 clusters is consistent with the RAE2822 and the AS28G, as a small cluster of high Mach numbers is associated to a large cluster of smaller Mach numbers, while a linear dependency to the angle of attack is observed;
- configurations with 3 and 4 clusters show similarities. The high Mach numbers are split along the values of the angle of attack;
- the configuration with 4 clusters reduces the size of the cluster #1 in favor of an intermediate region, exactly as the AS28G with 3 clusters.

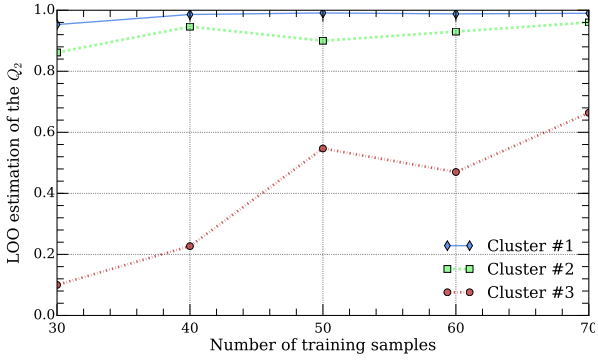
These results highlights the challenge of correctly setting the method, as various interpretations can be proposed. A complex application requires a detailed analysis of the models in order to select the best one.



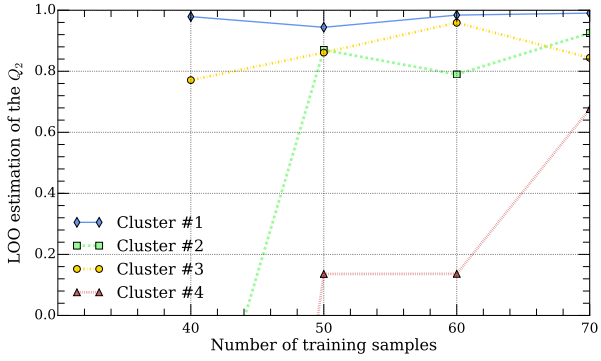
(a) 1 cluster



(b) 2 clusters



(c) 3 clusters



(d) 4 cluster

Figure 5.21: Evolution of the leave-one out error.

Therefore, cross-validation errors can help to obtain a reliable model and can guide in the choice of the best number of clusters. Figure 5.21 provides the LOO error for different models. The models with 3 and 4 clusters show weaknesses when they are trained with a low number of training samples. In particular, some local models have negative or almost null values of Q_2 with 30 and 40 training samples. This result is expected, as the number of training samples is not enough to sufficiently

train each local models. It would not be appropriate to consider these models with such LOO results.

5.2.2.3 POD analysis

The analysis of the POD is only performed for 70 training samples. The results are summarized in Table 5.5. Overall, they are similar to those of the other test cases. The cluster associated to low Mach numbers shows a low entropy compared to the other clusters. However, a specific element can be underlined: the clusters #2 and #3 of the model with 3 clusters and the clusters #3 and #4 of the model with 4 clusters are of the same order of magnitude with a high entropy. It might be thought that the Mach number is the only parameter responsible for the generation of entropy, as these clusters differ mainly by the angle of attack.

Number of cluster	Cluster index	Snapshots	Modes	Entropy
1	#1	70	32	0.481
2	#1	38	22	0.441
	#2	32	28	0.699
3	#1	35	20	0.444
	#2	17	15	0.7397
	#3	18	16	0.7287
4	#1	21	12	0.466
	#2	26	20	0.542
	#3	10	9	0.750
	#4	15	14	0.766

Table 5.5: Dimension reduction behavior.

A more detailed analysis of the POD modes is now proposed. Figure 5.23 represents the first POD mode on the suction side for different LDM models with 70 training samples. The cluster denoted by #1 is represented by the parameter space decomposition in the left column. Its associated mode depicts a typical shape that is found for models with 1, 2, and 3 clusters: a negative amplitude is observed on the majority of the wing with the exception of a narrow band near the trailing edge generating a non negligible gradient. The latter is not present for the LDM with 4 clusters. Indeed, its cluster #1 is smaller and does not contain snapshots with a Mach number above 0.6, explaining the smoother evolution of the mode along the chord. The mode of the cluster #1 can be interpreted as the recompression occurring for subsonic flows. Models with 3 and 4 clusters show similarities between respectively the clusters #2 and #3 and the clusters #3 and #4. Consequently, their POD modes share common shapes. In contrast with the cluster #1, the red clusters are characterized by a strong discontinuity associated to positive values, while the clusters at low angle of attack and high Mach numbers emphasize only a narrow peak values located about the discontinuity observed for the other modes.

The second POD modes are presented in Figure 5.24. Only the clusters #1 show different shapes, all the other clusters are very similar and look like the clusters #1 of the the first POD mode. Interestingly, the case with 4 clusters differentiates itself from the other models with a peak positive value on nearly all the suction side counterbalancing its smooth first mode.

The POD analysis of the AS28G has provided a clear delimitation between subsonic and transonic flows with only 2 clusters. It seems that the application of the LDM to the XRF1 configuration

may require more clusters. Indeed, the clear separation between smooth and discontinuous modes is an expected result of the LDM and it encourages in particular for further study of models with more than 2 clusters.

5.2.3 Numerical results

The accuracy of the LDM is presented in term of the normalized error. Figure 5.22 depicts the evolution of the error with the number of training samples. First of all, the main remark to be made regards the model with 2 clusters. The latter outperforms the global model whatever the number of training samples. This observation is particularly true for 30, 40, and 50 training samples.

Moreover, as predicted with the LOO error, the LDM with 3 and 4 clusters shows a poor accuracy with 30 training samples (and 40 training samples for the model with 4 clusters). Beyond this threshold, they reach an acceptable performance level. Nevertheless, when the number of training samples increases, the global method reduces the gap with the LDM. At 60 and 70 training samples, all the methods show almost a similar accuracy. Two possible interpretations are proposed: (i) a sufficient number of training samples is reached and can overcome limitations of the global model, as regression models are well trained, (ii) the truly challenging points are too few in the test sets, since the parameter space is evenly filled.

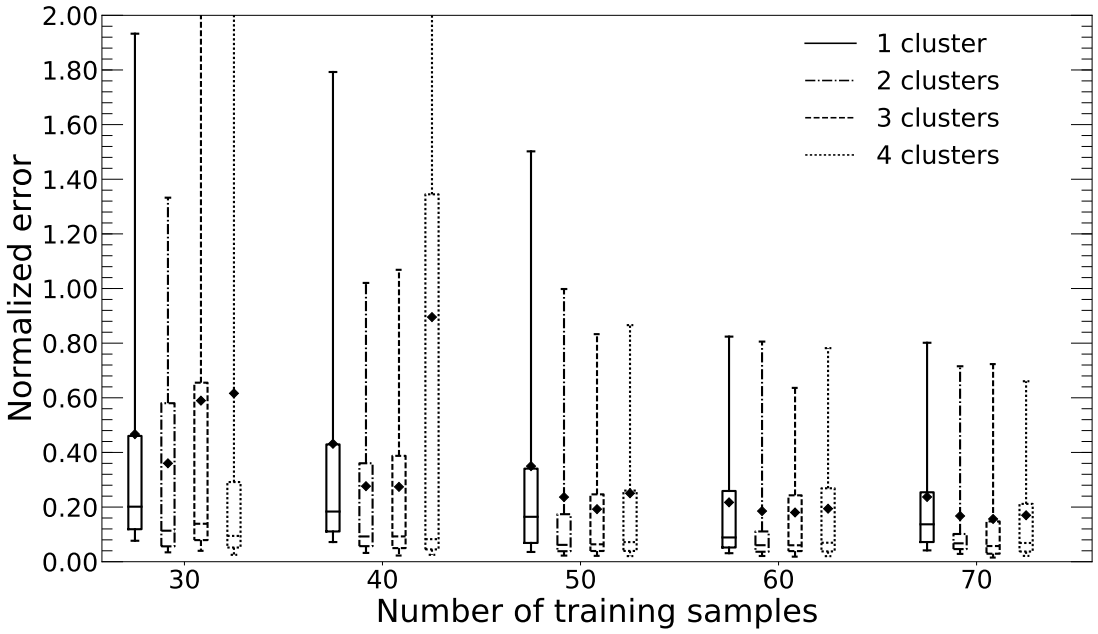


Figure 5.22: Box plots of the normalized error for different numbers of training samples.

Once again, similarities are observed between all the test cases of this thesis. The pressure predictions of the RAE2822, the AS28G, and the XRF-1 have been widely enhanced by the LDM with 2 clusters. This improvement is mitigated with 3 clusters for the AS28G. However the conclusion for the XRF-1 is more complex as the accuracy widely increases once a threshold is reached in term of training samples for a model with 3 and 4 clusters.

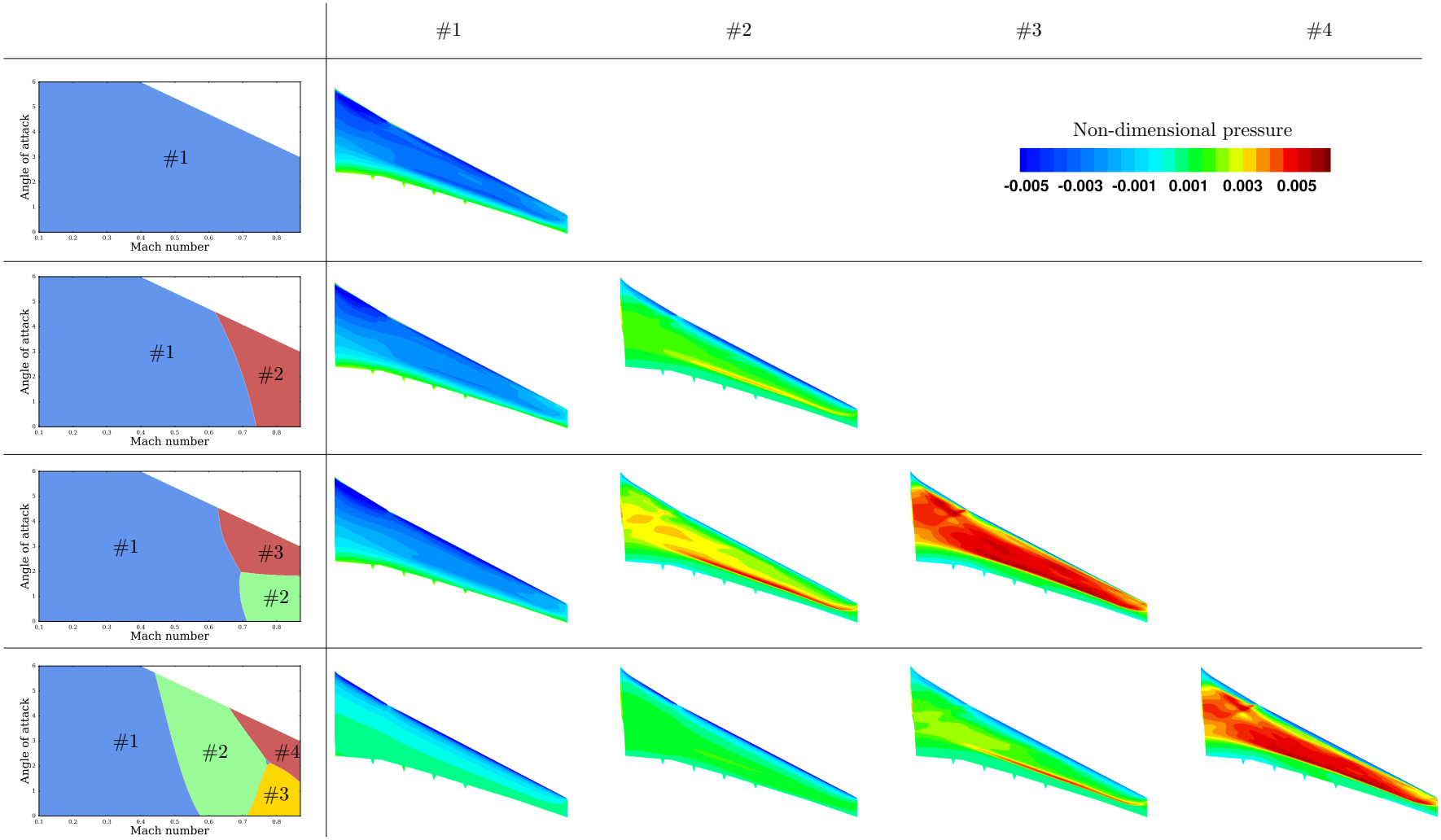


Figure 5.23: First mode on each cluster at 70 training samples.

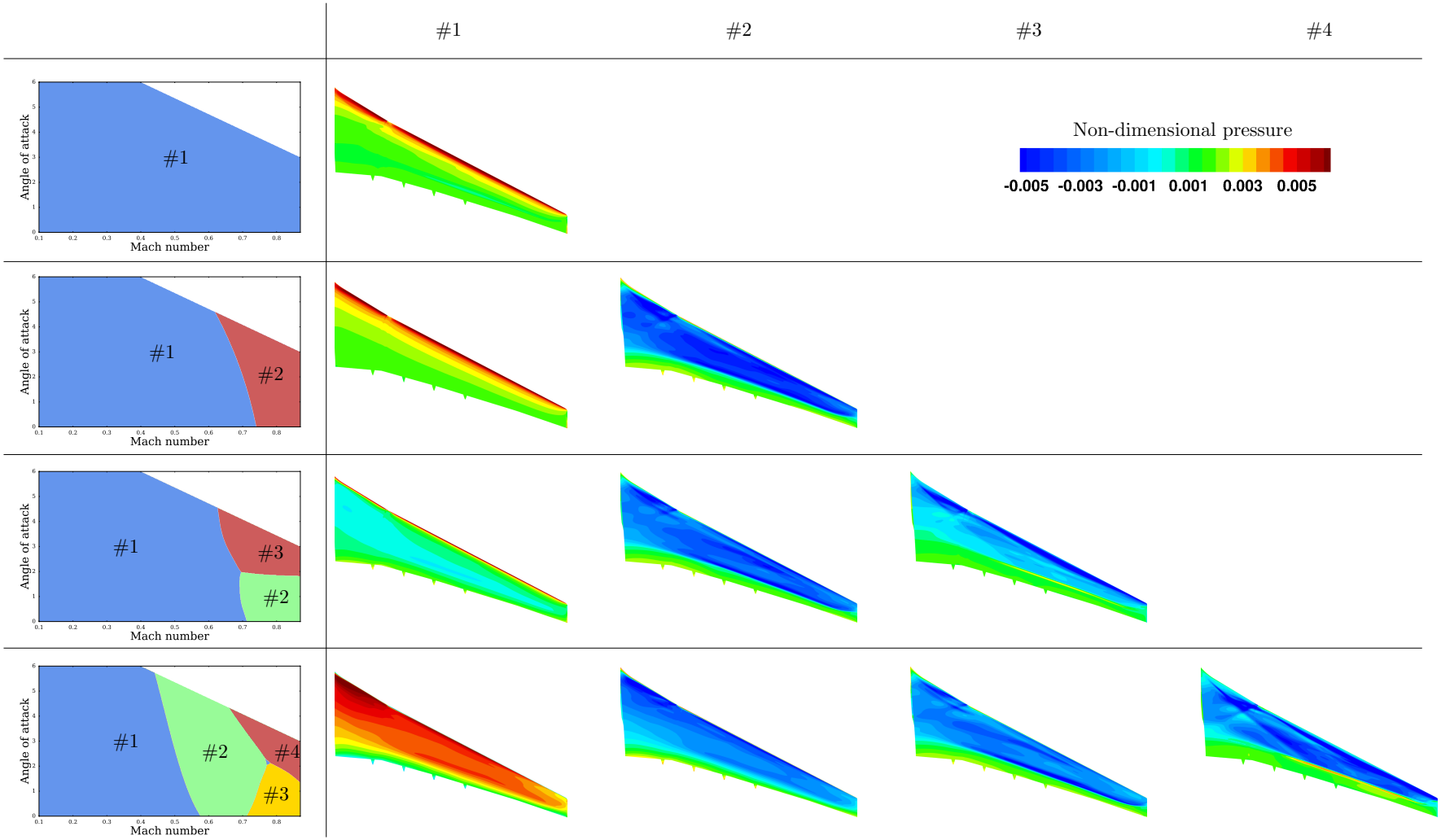


Figure 5.24: Second mode on each cluster at 70 training samples.

5.3 Conclusion and main findings

The LDM was applied and assessed with simulations of a turbulent flow around the AS28G half fuselage configuration and the industrial-like XRF1 configuration. Observations revealed a consistent separation of the parameter space with continuous regions and a clear identification of the subsonic and transonic regime. Moreover, a significant improvement of accuracy compared to the classical method was reached for almost all the cases. These results were expected as the LDM was designed to split subsonic and transonic flows. In particular, the relevance of the resampling step was highlighted, as subdomains associated to high gradients and discontinuities were targeted.

Nevertheless, the LDM emerged as a sophisticated method which may require to tune several numerical parameters. Its application on the AS28G was straightforward with impressive results with only 2 clusters. However, several tools must be deployed to ensure the quality of the result, in particular the analysis of the clustering (elbow method, Silhouette analysis...) and the accuracy with an estimation by leave-one out error. These tools are particularly important for complex applications such as the XRF-1.

Finally, these two aerodynamic test cases emphasized the complexity to validate surrogate models. For XRF-1, more than 150 simulations were performed with a significant CPU cost. However, measuring the real accuracy of each model appeared very challenging, mainly due to: the size of the parameter space, the computational costs, and the vast amount of data, as each simulation represents thousands of scalars. Finding a relevant metric measuring represents a major challenge, justifying the use of various quantities such as predictivity coefficient, root mean square error, or relative error, under different forms: mean, spatial distribution, box plot, etc.

CHAPTER 6

Aerothermal simulations using reduced-order models

Résumé

Une grande partie du travail de thèse s'est concentrée sur le développement de modèles réduits pour l'aérodynamique. Dans ce chapitre, on s'intéresse à l'application et l'intégration de modèles réduits dans un processus industriel de calcul aérothermique, en particulier pour le dimensionnement du mât moteur. L'objectif est de prédire le comportement transitoire du mât moteur pendant une mission caractéristique englobant les phases de décollage, de montée, de croisière, de descente et d'atterrissage. Pour cela, des conditions limites de convection forcée sont imposées par un modèle fluide stationnaire au modèle thermique transitoire. Habituellement, ces conditions limites, qui évoluent en fonction des conditions de vol, sont générées par un modèle semi-empirique ou par un modèle de mécanique des fluides très couteux en termes de calcul. Le remplacement par un modèle réduit permet de réduire le coût de calcul global du processus, de fournir un outil collaboratif aux différentes équipes d'ingénierie et d'avoir une méthode construite autour des données de simulations.

L'intégration de deux modèles réduits sur le cas XRF-1 a donné de très bons résultats thermiques. Le comportement transitoire fourni par le processus utilisant les modèles réduits est très proche du comportement obtenu par des calculs directs de mécanique des fluides numérique. En particulier, les phases de décollage, de montée, de descente et d'atterrissage ont été particulièrement bien prédites pour les trois éléments du mât moteur que l'on étudie. Seuls certains points de croisière se sont montrés plus difficile à prédire, principalement à cause de points de fonctionnement très proches des bornes de la base d'apprentissage.

Les outils d'analyse amont se sont aussi montrés très performants. Ils ont permis de déterminer un nombre suffisant de points d'entrainement afin d'atteindre une bonne prédiction et aussi d'identifier clairement les régions de l'espace où les prédictions des modèles réduits pouvaient être de qualité moyenne. En particulier, les points critiques au niveau de la croisière ont été anticipés par ces outils. Cela permet d'envisager plusieurs améliorations directes du

processus aérothermiques, comme par l'exemple l'utilisation d'un réechantillonage adaptatif permettant d'améliorer la base d'entrainement initiale selon la mission. De plus, les panneaux latéraux du mât moteur ont montré un comportement très particulier. En effet, les flux primaires et secondaires viennent se mélanger au niveau des panneaux. Ainsi, une frontière nette se dessine entre les parois chaudes immergées dans l'écoulement primaire et les parois plus froides immergées dans l'écoulement secondaire. Ce type de phénomène se prête particulièrement bien à la méthode de décomposition locale. Il serait intéressant de poursuivre les travaux afin d'étudier la validité de cette méthode sur des cas très différents de l'aérodynamique.

This chapter evaluates the performance of reduced-order models for aerothermal simulations. The application under study focuses on pylon sizing, for which the transient thermal behavior is required over various flight missions. Engine fan and core flows drive an important part of heat exchange and their interactions with the external flow change with operating conditions, making predictions of the thermal behavior even more complex. Modelling such phenomena usually involves CFD simulations or semi-empirical models. This chapter aims to demonstrate the possibility of integrating surrogate models into industrial calculation processes. The proposed approach presents several advantages: to reduce the computational budget compared to direct CFD, to offer a data-driven approach, and finally to provide a more collaborative tool for different engineering teams.

6.1 Pylon sizing over a flight mission

Pylon structures are highly impacted by engine exhaust. The transient prediction of their thermal behavior needs several elements:

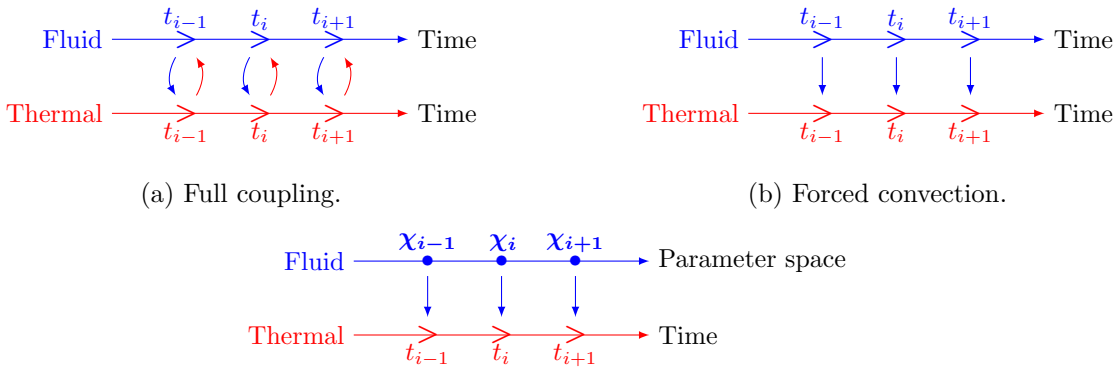
- a clear definition of the flight mission in terms of time and operating conditions;
- a thermal model taking into account material thermal inertia, conductive, inner convection, and radiative heat transfers;
- a fluid model accounting external convective boundary conditions.

This last point represents the central element of this chapter, as the fluid model can be substituted by ROMs.

6.1.1 Mapping strategy

This section presents the strategy adopted to perform aerothermal simulations for limited computational capacity. Heat transfers between fluid and solid are assumed to be driven only by forced convections, reducing significantly the computational cost compared to a fully dynamic coupling illustrated in Figure 6.1a, where the fluid model can be CFD, surrogate models, or semi-empirical models. This assumption allows us to consider only a one way coupling from the fluid to the solid without a feedback of the structure on the fluid, as illustrated in Figure 6.1b. In other words, the fluid is assumed to be independent of the wall temperature. One can note that the assumption of forced convection is verified at high speed but can be questioned at low speed.

Moreover, as demonstrated in Errera and Baqué [63], the ratio between diffusion time in the solid and in the fluid is proportional to the Reynolds number. For this reason, the flow field is simulated at specific time steps in steady state conditions. Hence, the computational time is reduced by considering a sequence of steady states for the fluid, as illustrated in Figure 6.1c.



(a) Full coupling.

(b) Forced convection.

(c) Quasi dynamic and forced convection assumptions.

Figure 6.1: Evolution of the thermal mapping strategy.

Despite these assumption, aerothermal simulations over a mission remain intractable since typical sizing missions can require between dozens or hundreds of steady simulations. Approximations are usually used to face this computational problem. The main purpose of this chapter is to build an aerothermal process based on surrogate models and to understand the impact of these models.

6.1.2 Definition of a mission

An example of a representative flight mission of long range aircraft is chosen to assess the performance of surrogate models in aerothermal mapping. As for XRF-1 aerodynamic case, seven input parameters are considered. The mission is discretized by 26 time steps of varying duration and they are grouped into different flight phases: ground, take-off, climb, cruise, descent and landing. These flight phases are broadly separated by Mach number. Table 6.1 summarizes all the 26 points.

This classical mission present interesting features. The different phases are analyzed by separating external conditions (Mach, angle of attack, and Reynolds) and engine conditions. First, the ground points are computed at a Mach number of 0.1 in this thesis. The fluid solver elsA is designed for compressible flows, thus this value avoids the use of low Mach preconditioners (even if they are available), while engine conditions remain representative of the ground. Then, during climb and descent, the aircraft encounters a wide range of external conditions. However engine conditions differ significantly as climb needs a significant proportion of maximal thrust while engines idle for descent. Finally, cruise is by far the longest phase but it also shows the lowest variability, as only fan conditions change significantly.

6.1.3 Thermal and fluid models

Thermal modeling is performed by the solver NX® Thermal developed by Siemens. It combines a high order finite volume method with a finite element mesh. This engineering software simulates the three modes of heat transfer: conduction, convection, and radiation.

The XRF-1 thermal model used for this study contains 13,865 elements, where the vast majority is two-dimensional, and is composed of 4 main elements: pylon primary, secondary structures, engine mounts, and engine casing. Pylon primary and engine mounts are mainly driven by conduction and natural convection, while the engine casing generates an important heat flux by conduction

Time (s)	Phase	Mach	Angle of attack	Reynolds	T_t core	P_t core	T_t fan	P_t fan
0	Ground	0.10	0.0	1.74×10^6	3.10	1.06	1.01	1.05
1,800		0.10	0.0	1.74×10^6	3.10	1.06	1.01	1.05
2,400		0.10	0.0	1.74×10^6	3.10	1.06	1.01	1.05
2,420	Take-off	0.15	0.5	3.49×10^6	3.13	1.26	1.07	1.46
2,440		0.20	4.0	4.66×10^6	2.82	1.25	1.07	1.44
2,460		0.25	5.5	5.83×10^6	2.81	1.23	1.06	1.42
2,490		0.30	5.5	6.90×10^6	2.80	1.23	1.09	1.42
2,520		0.35	5.5	7.94×10^6	2.73	1.23	1.09	1.41
2,540		0.39	5.5	8.68×10^6	2.73	1.17	1.08	1.41
2,550	Climb	0.39	4.5	8.63×10^6	2.70	1.19	1.06	1.29
2,760		0.45	4.5	7.94×10^6	2.78	1.37	1.10	1.37
3,114		0.65	3.0	8.44×10^6	2.85	1.45	1.06	1.21
3,800		0.80	3.0	7.32×10^6	2.90	1.23	1.09	1.12
3,720	Cruise	0.85	2.5	7.76×10^6	2.85	1.15	1.07	1.07
27,200		0.85	2.5	6.65×10^6	2.88	1.15	1.11	1.18
58,192		0.85	2.5	5.51×10^6	2.93	1.18	1.15	1.52
58,200	Descent	0.83	2.0	5.38×10^6	2.81	1.14	1.07	1.04
58,567		0.70	1.5	8.02×10^6	2.83	1.15	1.05	1.10
59,000		0.45	3.0	7.94×10^6	3.10	1.15	1.01	1.06
59,350		0.39	3.0	8.62×10^6	3.06	1.15	1.01	1.03
59,377	Landing	0.39	4.0	8.68×10^6	2.91	1.03	1.02	1.02
59,460		0.20	4.0	4.66×10^6	2.85	1.05	1.02	1.04
59,520		0.10	2.0	2.33×10^6	2.72	1.14	1.02	1.27
60,600	Ground	0.10	0.0	1.74×10^6	3.10	1.06	1.01	1.05
62,400		0.10	0.0	1.74×10^6	3.10	1.06	1.01	1.05
62,500		0.10	0.0	1.74×10^6	3.10	1.06	1.01	1.05

Table 6.1: Discretization of a sizing mission.

and radiation, as its temperature is set to core temperature. Secondary structures are the only parts in contact with external and engine flows. Consequently, they are specifically studied, in particular the three components marked in Figure 6.2:

- the exhaust which includes the plug, the lower skin of the aft pylon fairing and the inner nozzle. This part encounters flow at high temperature coming from the core;
- the outer nozzle which is colored in red in Figure 6.2. The heat transfer is mostly driven by fan flow;
- the lateral panels which are represented in yellow in Figure 6.2. Predicting an accurate heat flux is very challenging as the mix between fan and core flow occurs on these panels.

Regarding fluid modeling, adiabatic simulations of the XRF-1 from section 5.2 are reused and new isothermal simulations are performed, as presented in the following section. Details on solver settings can be found in section 5.2.1.2.

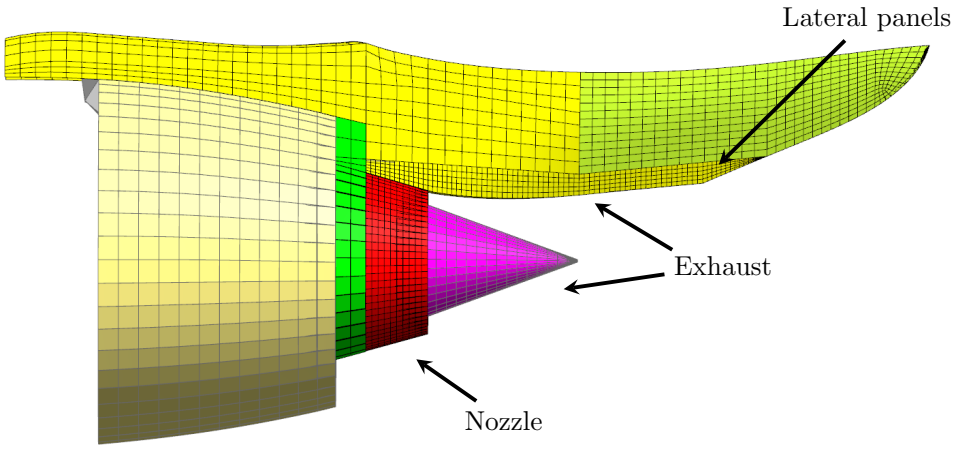


Figure 6.2: Illustration of the NX® thermal finite element model.

6.1.4 Boundary conditions at the interface

Quantities at the boundary of the solid are determined with the Newton's law of cooling, such as heat flux density $\phi_{f\to s}$ from the fluid to the solid is given by:

$$\phi_{f\to s}(\boldsymbol{x}) = h(\boldsymbol{x})\left(T_{ref}(\boldsymbol{x}) - T_s(\boldsymbol{x})\right), \forall \boldsymbol{x} \in \Omega_s, \tag{6.1}$$

where h is the heat transfer coefficient, $\boldsymbol{x}$ is located on the wall, Ω_s is the thermal mesh, T_s and T_{ref} are respectively the wall temperature and the reference temperature.

The thermal model requires to know the value of T_{ref} and h to be well defined. Their computation is made of two consecutive steps:

1. A simulation with adiabatic walls is performed in order to determine the field of reference temperature T_{ref} associated to a zero heat flux density.
2. A second simulation is performed with an isothermal wall condition. The field of wall temperature T_{isoth} is obtained by subtracting a ratio r of the infinite temperature T_∞ to the field of the reference temperature. Thus, the heat transfer coefficient can be computed:

$$h(\boldsymbol{x}) = \frac{\phi_{f\to s}(\boldsymbol{x})}{(T_{ref}(\boldsymbol{x}) - T_{isoth}(\boldsymbol{x}))} \tag{6.2}$$

$$= \frac{\phi_{f\to s}(\boldsymbol{x})}{rT_\infty}. \tag{6.3}$$

This process is repeated for all the snapshots of the training and testing sets.

6.1.5 Data exchange between different models with Padge

Padge (Parametric And Differentiated Geometrical Engine) is an in-house differentiated geometric kernel developed at Airbus, mainly used for shape parametrization, mesh deformation, and optimization. One of its main features allows to connect the surface mesh with the computer-aided design (CAD). This connection, called CAD to mesh link, facilitates the data transfer from a mesh to another one where the CAD plays the role of a neutral interface. In the context of this thesis,

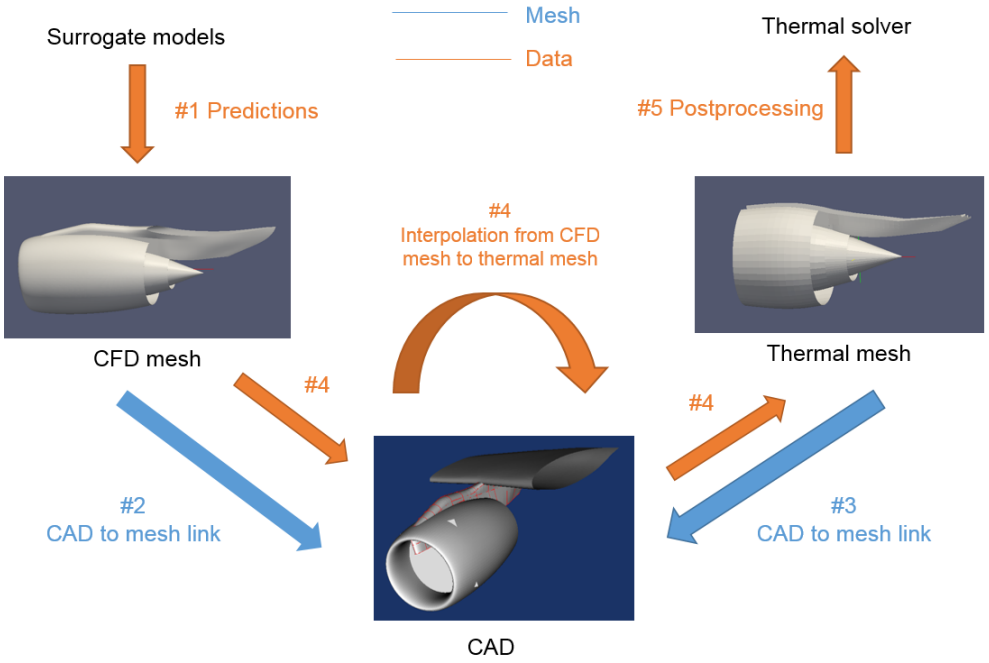


Figure 6.3: Padge workflow.

Padge has been used for the first time to perform coupling between CFD, surrogate models, and thermal model. Various scripts have been developed to achieve a fully functional process.

The surfaces $S_1, \cdots, S_n$ of the CAD are represented by Non-uniform rational B-splines (NURBS) defined by two independent parameters u and v. Each mesh point $\boldsymbol{x_i}$ is expressed in the space (u, v) of the closer surface, such as its new coordinates u_i and v_i minimize the following expression:

$$\min_{k\in[1,n]} \min_{u,v\in[0,1]^2} \|\boldsymbol{S_k}(u,v) - \boldsymbol{x_i}\|, \forall \boldsymbol{x_i} \in \Omega_s. \tag{6.4}$$

Once fluid and thermal meshes are linked to the CAD, data from the finer mesh can be interpolated on the coarser mesh. Indeed, all the mesh points of both meshes now have coordinates (u, v) in a given surface of the CAD. Figure 6.3 summarizes the different steps of the whole process:

1. The surrogate model predicts data on the CFD mesh.
2. The CFD mesh is linked to the CAD.
3. The thermal mesh is linked to the CAD.
4. For all the surface $S_1, \cdots, S_n$, data on the coarser mesh are determined by spline interpolations in the space (u, v).
5. Data on the thermal mesh are processed: converting nodes data to cell data and writing files readable by the thermal solver.

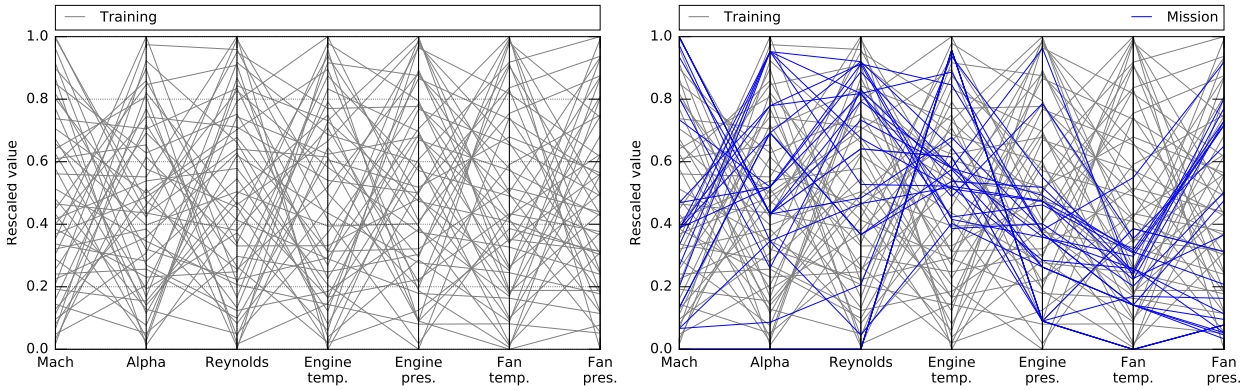
Errors are generated during this process. Padge projects nodes data to the CAD, while CFD and thermal solvers are cell-centered. Consequently, converting cell data to node data and node data to cell data contributes to the error. Other sources of error are identified in the process with the CAD to mesh link, in particular projection errors and mismatching with the surfaces can occur. But the most significant error is observed during the fourth step of the process with the interpolation, as CFD and thermal meshes usually have very different densities.

6.2 Generation of surrogate models

As explained in the previous section, the boundary conditions of the thermal model require two parameters: the reference temperature T_{ref} and the heat transfer coefficient h. Thus, two independent surrogate models are trained on these quantities to substitute CFD.

6.2.1 Sampling

The two quantities are highly dependent because the computation of heat transfer is based on isothermal wall conditions determined from a reference temperature field. This constraint imposes that the two surrogate models follow the same DOE to minimize the computational budget.



(a) Training set.

(b) Training set with the mission.

Figure 6.4: Parallel coordinate plots of the training set with 50 samples.

A classical low-discrepancy Halton sequence has been chosen to explore the parameter space. Figure 6.4a shows the distribution of the training set for seven input parameters with a parallel coordinate plot. The samples seem to be evenly distributed over each parameter. Figure 6.4b gives an example of mission chosen for the pylon sizing. The largest portion of mission time steps is well distributed between upper and lower bounds of the parameter space. Nevertheless, several critical points can be identified: the ground point is located at the lower bound of 4 parameters (Mach number, angle of attack, Reynolds, and fan temperature) and the cruise points reach the upper bound of the Mach number.

Predicting points near the bounds of the training set is a well-known weakness of the POD/GP method, motivating a slight modification of the training set. It may be expected that ROMs face problems to predict such a challenging point without extra information adjusting reduced coordinates near the bounds. Thus, a specific point typical of the ground is added to the training set, since all flight missions have a ground phase and that significant time is spent in this phase. Finding a point representing this phase is feasible since it is characterized by very specific conditions: low Mach number, low angle of attack, zero altitude, low fan and core pressures, and low fan and core temperature. However, finding a point representing the second critical phase, the cruise, is more challenging, as this phase is described by various values of the input parameters: Mach number, altitude, and engine conditions may change according to the mission. For this reason, the training set cannot be specifically enriched for the cruise phase.

6.2.2 Validation

Surrogate models are assessed in two different ways: with a-priori methods and by comparison with a testing set. These results aim to identify strengths and weaknesses of surrogate models to estimate their impacts on thermal process. Quantities of interest are computed in dimensionless form.

6.2.2.1 A-priori error

The aerothermal mapping can greatly benefit from a-priori methods, since they do not need testing sets. For instance, they can be used to validate partially the model or to propagate uncertainties through the whole process. Figure 6.5 shows the evolution of the leave-one-out predictivity coefficient against the number of training samples for the two surrogate models. They are expected to be more reliable than the aerodynamic case, since shocks and high discontinuities should be absent. The predictivity coefficient continuously rises and seems to reach a plateau for 50 training samples. Thus, only the results with 50 training samples are analyzed.

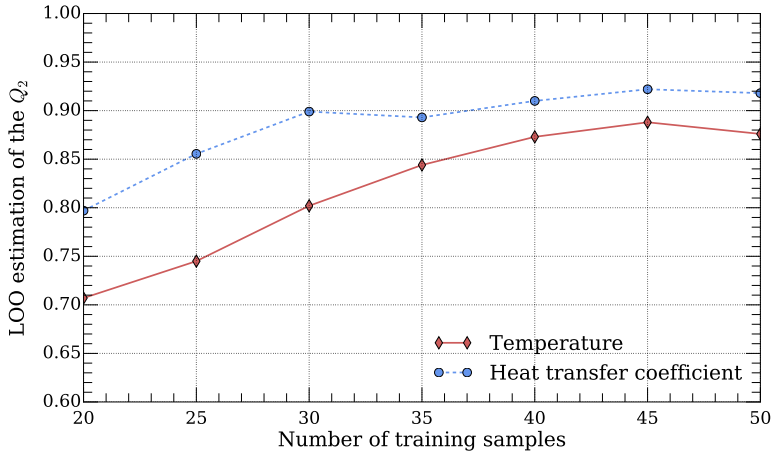
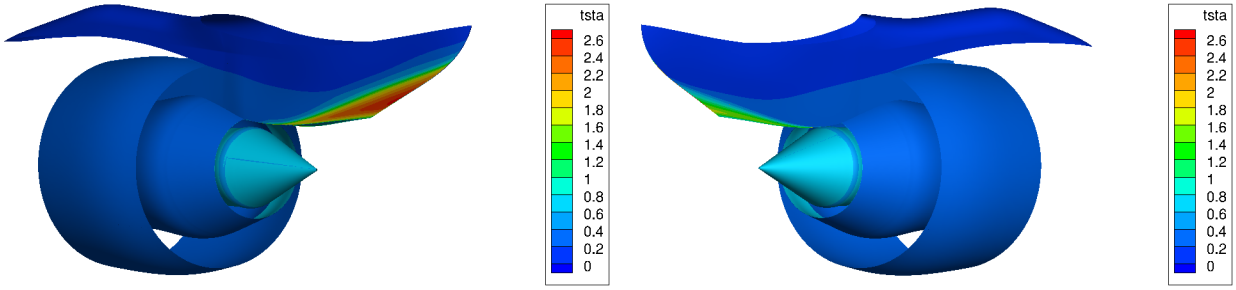


Figure 6.5: Leave-one-out estimation of spatially averaged errors for temperature and heat transfer coefficient.

The spatial distribution of the mean variance of the Gaussian Process, given by Equation 3.137, is illustrated in Figure 6.6 for temperature and in Figure 6.7 for heat transfer, computed as dimensionless quantities. The maximum variance is observed on the lateral panels for both temperature and heat transfer coefficient. A physical interpretation can be given to this result: engine core, fan and external flows mix in this region and the exact location of the interface between the three flows depends mainly on local properties driven by boundary conditions and operating conditions. One can note that the maximum variance reaches a specific value of about 2.6, corresponding broadly to the difference in temperature between core and fan. Moreover, a dissymmetry of the flow is clearly noticed by comparing right and left views, as the variance of the left panel is much higher for both reference temperature and heat flux coefficient. Several elements may explain this dissymmetry: the geometry of the pylon, the decrease of the chord length, and the impact of the fuselage.

As regards temperature for the other parts, only the plug and the inner nozzle show a significant variance, but it must be weighed as these components also encounter the highest temperatures: about 3.0 for core flow while fan flow is about 1.2. Finally, the lower panel facing the plug, not

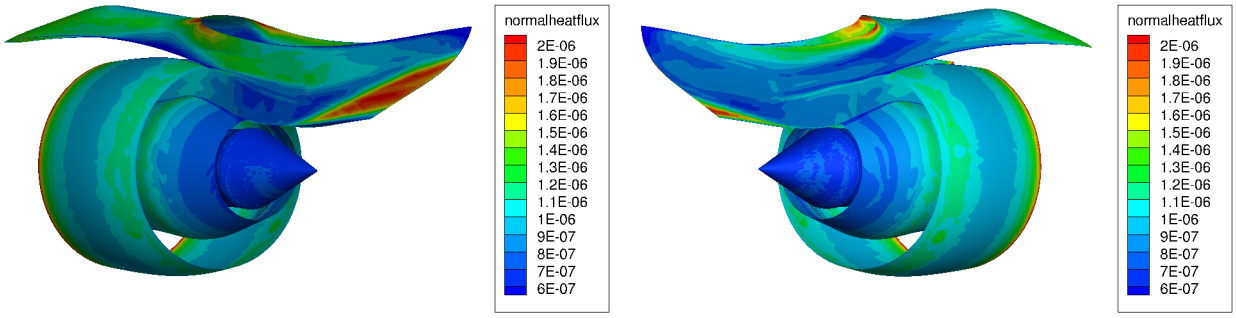
visible on the figures, has an intermediate behavior. The variance is similar to the plug and inner nozzle in its center and shows higher values for nodes close to the external panels.



(a) Left view. (b) Right view.

Figure 6.6: Estimation of the mean error provided by Gaussian Process Regression for reference temperature.

As regards heat transfer coefficient, the distribution of variance differs from the temperature, except for the lateral panels at the rear. In particular, high values are located on the outer nozzle, the nacelle, and the upper part of the pylon, while the plug presents low variance. This result is expected since heat transfers depend on local flow properties and that reference temperature is mainly driven by the temperature of the flow, in particular fan and core flows. Indeed, core flow depends only on total pressure and temperature of the core and its structure remains similar for various engine conditions, explaining low variance for the plug.



(a) Left view. (b) Right view.

Figure 6.7: Estimation of the mean error provided by Gaussian Process Regression for heat transfer coefficient.

Therefore, the a-priori analysis suggests that predicted errors can be quantified as follows:

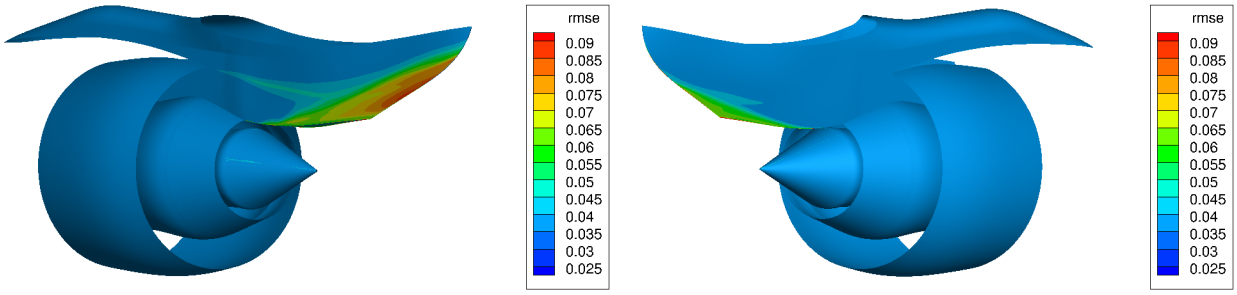
- on the lateral panels: surrogate models seem to show significant weaknesses for both heat transfer coefficient and temperature. This results is expected since two flows with very different properties mix in this region. The issue is straightforward for the surrogate models: to predict precisely the interface between the two flows. This is particularly important for reference temperature field, as the engine core flow can reach more than 750°C while the engine fan flow is below 100°C. Conversely, the heat transfer coefficient has much smaller amplitude between extreme values. Thus, the surrogate model of the reference temperature represents the critical part of the aerothermal process for the lateral panels;

- on regions in contact only with engine fan flow: errors seem more mitigated in this part. Important variance is observed for heat flux while significant variance are located on the nozzle, relatively to typical dimensionless temperature of the fan around 1.1;
- on regions in contact only with engine core flow: errors are foreseen to remain low, since variance is minimum for heat flux and relatively low for reference temperature. The components are immersed in engine core flow which has high energy. It is quite possible that this flow is less sensitive to external conditions and engine fan conditions. Only its own parameters, engine core total pressure and total temperature, may influence the heat transfer, explaining the relative low variance.

6.2.2.2 Global accuracy

Surrogate models are validated with a testing set of 160 snapshots from a Sobol sequence, as for the aerodynamic part of XRF-1 case. The results are expressed as normalized root mean square error (NRMSE).

The distribution of NRMSE, shown in Figure 6.8 and Figure 6.9, partially confirms the expected error given by GP in Figure 6.6 and Figure 6.7. The typical high error located on the lateral panels is also found for NRMSE with the exact same shape. Nevertheless, a noticeable NRMSE is observed on all the geometry for the temperature, which is not the case for the estimated variance. Discrepancies are also observed for heat transfer coefficients with an error overestimated by the surrogate model, in particular for the nozzle.

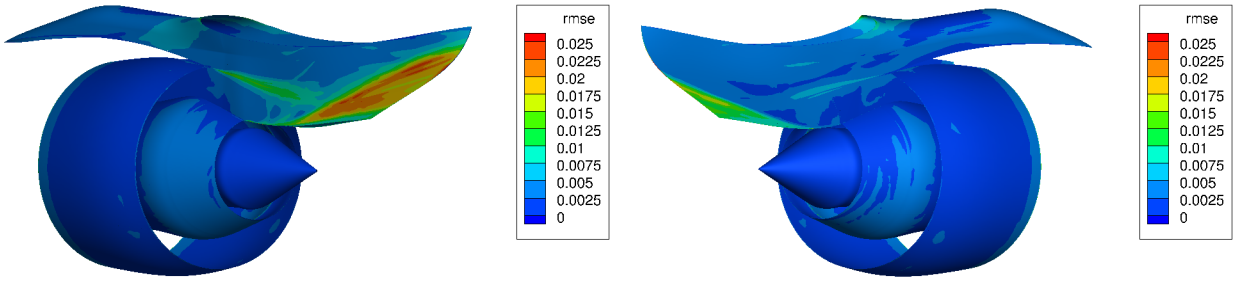


(a) Left view. (b) Right view.

Figure 6.8: Distribution of NRMSE for temperature.

Nevertheless, the surrogate model accuracy appears satisfactory as errors are mainly limited to a specific region which is expected to be very challenging: the lateral panels. Thermal behaviors of other critical components, such as the plug, the nozzle, or the aft pylon fearing, are well predicted by surrogate models for the testing set, as the mean NRMSE is below 5% for the reference temperature and below 1% for the heat flux coefficient.

The statistical assessment in this section validates the use of surrogate models in the aerothermal process. The mean NRMSE for all the components, except the lateral panels, is approximately lower by a factor of three than the maximum value reached by lateral panels. These acceptable values motivates the next step: evaluating the accuracy of the surrogate model within the aerothermal process for a representative sizing mission.



(a) Left view. (b) Right view.

Figure 6.9: Distribution of NRMSE for heat flux coefficient.

6.3 Applying surrogate models to the aerothermal mapping

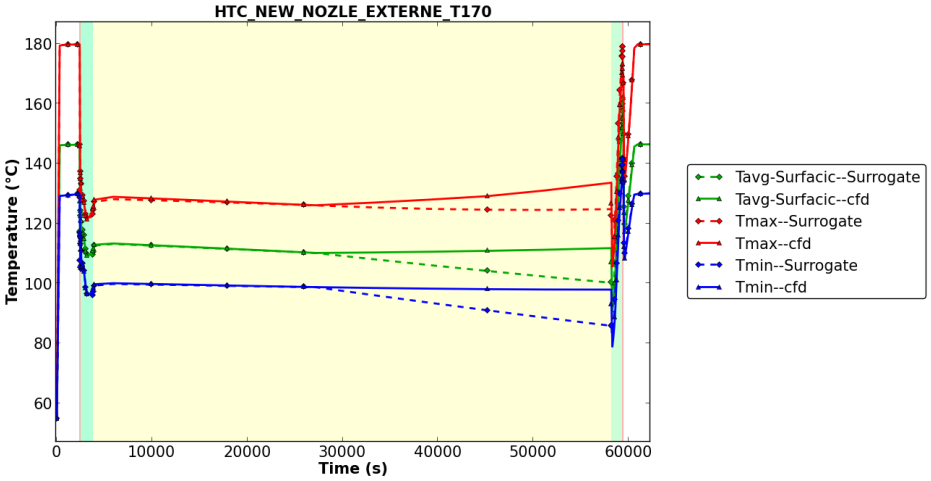
This section presents the aerothermal mapping for three components: nozzle, lateral panels, and exhaust. All the points of the simulations are predicted with surrogate models and they are also computed with CFD. Comparisons between direct CFD and ROMs focus on temporal evolution of maximum, mean, and minimum temperatures. In particular, the maximum temperature represents an essential criterion for the pylon design as it can motivate the choice of materials and location of thermal protection.

Surrogate models are expected to be challenged during take-off, climb, descent, and landing, as all input parameters change drastically: any hot spot must be predicted in term of absolute value and location as well as the overall shape of fields. The flight phases are represented in color for the different figures: ground in white, take-off and landing in red, climb and descent in green, and cruise in yellow.

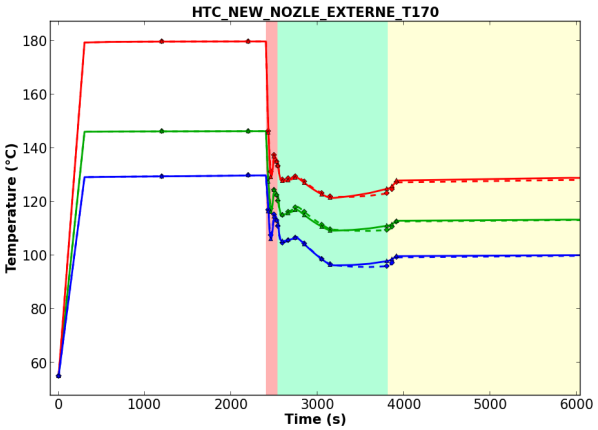
6.3.1 Nozzle

Figure 6.10a shows variations of temperature against the time with an overall view, a zoom on first time steps, and a zoom on last time steps. The high discrepancy between CFD and the surrogate model at the last point of the cruise around $50,000$ seconds is the most striking finding. The temperature is underestimated by more than 10°C for the three statistics with the exact same trend. This error is quiet significant as the nozzle reaches a maximum temperature of about 180°C during the flight.

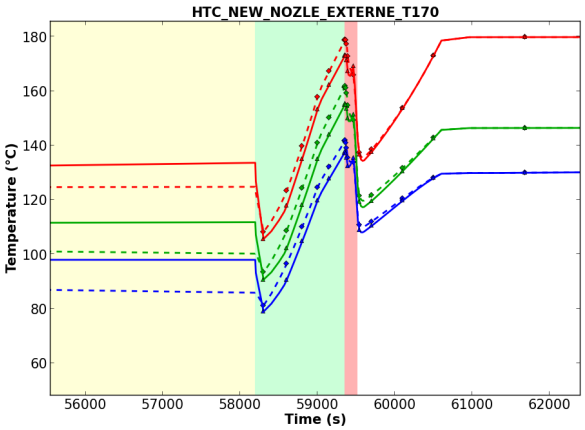
The spatial field at the end of cruise ($51,211$ seconds), represented in Figure 6.11, confirms the underestimation of the temperature over the full nozzle. Nevertheless, the shape is well predicted, as hot and cold spots are located in the same regions between CFD and surrogate models even if the whole field is shifted by 10°C. Fluid boundary conditions of CFD and surrogate model show that heat transfer coefficients are almost identical while an uniform gap of 15°C is observed for the reference temperature. This difference is difficult to explain as both surrogate models use the same DOE. Table 6.1 and Figure 6.4 highlight that the last cruise point at $58,192$ seconds is associated to a fan pressure close to the upper bound. Even if several training points are computed for high fan pressures, it is very likely that GPR models generate significant errors due to extrapolation conditions. In particular, sensibilities to the fan pressure for the reference temperature may be incorrect to extrapolate reduced coordinates. The amplitude of the variance field for this specific cruise point is indeed almost twice as much as the mean value, highlighting that prediction errors are expected for these input parameters.



(a) Overall behavior.



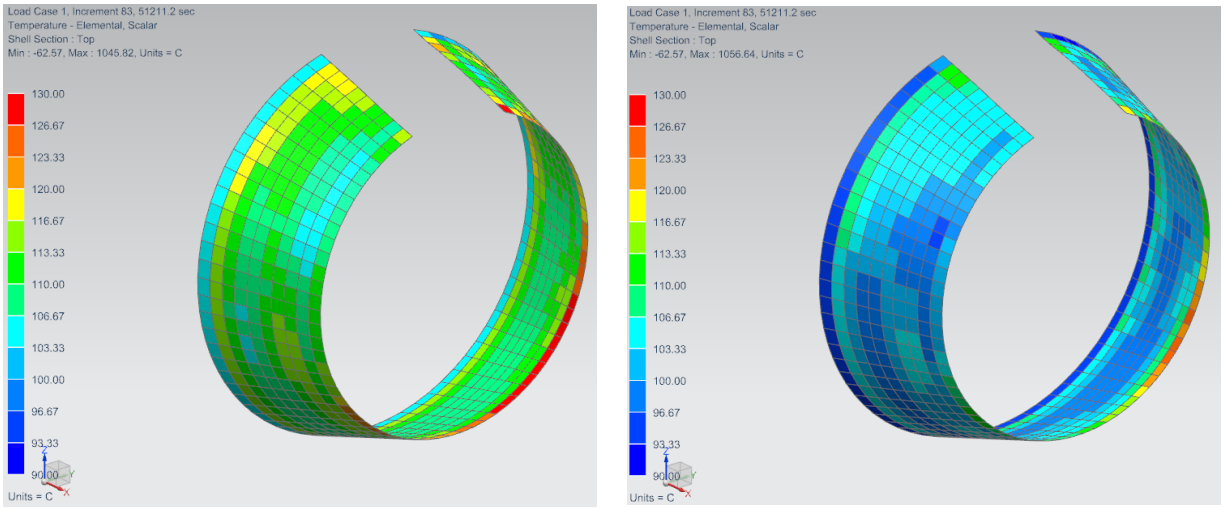
(b) Zoom on take-off and climb.



(c) Zoom on descent and landing.

Figure 6.10: Transient thermal analysis of the outer nozzle.

Results in other phases are in very close agreement. Zooms on take-off, climb, descent, and landing, as seen in Figure 6.10b and Figure 6.10c, show CFD and surrogate model curves of temperature are almost superimposed. Only a slight difference of 4°C is observed between CFD and surrogate models during descent for the three statistics suggesting that the shape is well predicted but shifted. The analysis of reference temperature and heat transfer coefficient indicates that discrepancies partially cancel out: on the one hand the reference temperature is slightly overestimated and on the other hand the heat transfer coefficient is slightly underestimated. This gap does not appear for take-off and climb while there are also significant variations in external and engine conditions. The common characteristic of all points in descent and landing is the low pressure fan compared to points in take take-off and climb. Indeed, several mission points are close to the lower bound of the fan pressure as depicted in Figure 6.4. This confirms that several points of the mission challenge the surrogate model, in particular near the upper and lower bounds defining the training set.



(a) CFD (b) Surrogate

Figure 6.11: Field on outer nozzle at time step $t = 51,211$ seconds.

6.3.2 Lateral panels

Figure 6.12 compares the transient thermal behavior between CFD and surrogate models for the lateral panels. Although these components are expected to be very challenging, significant discrepancies are observed only at the beginning and the end of cruise. The maximum temperature is overestimated by about 40°C around times steps $t = 1,000$ seconds, while the mean temperature is well predicted. This suggests that only hot spots are incorrect.

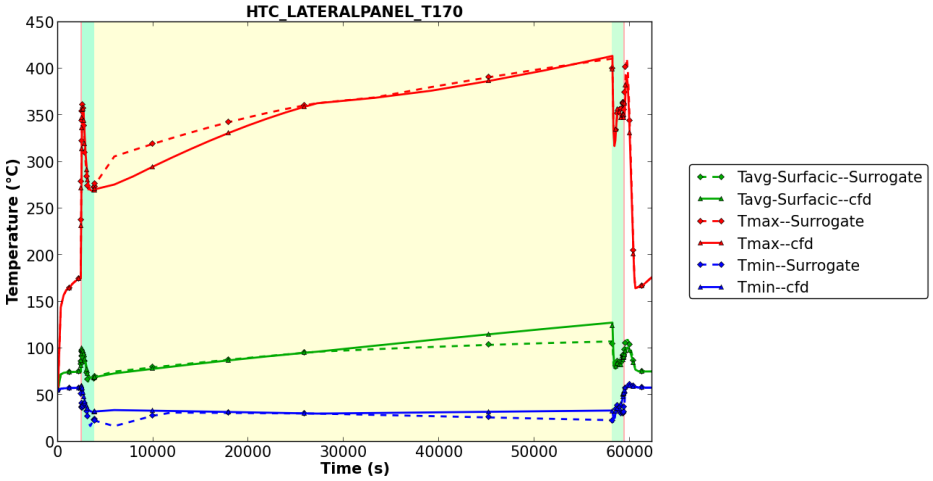
The spatial field, illustrated in Figure 6.13, confirms this intuition as only few cells are overestimated. Once again, the source of the error comes from the surrogate model generating the reference temperature. A plausible explanation is that POD modes representing hot spots on the panels have been wrongly amplified by the associated reduced coordinates. The only possible correction is to enhance the training samples near cruise points.

The error on the mean temperature at the end of the cruise (around $t = 58,000$ seconds) raises a very interesting point, as it corresponds to the same critical point than for the nozzle. This gap of 20°C is expected and can be explained by a small misplacement of the interface between hot flow from the core and cold flow from the fan, due to extrapolating conditions for the fan total pressure. Nevertheless, this is the only significant discrepancy for the mean temperature, since all other points are well predicted. For instance, Figure 6.14 illustrates the prediction of reference temperature at the beginning of the descent (58,567 seconds). Differences between CFD and surrogate model fields are barely visible. The same result can be observed for the heat transfer coefficient.

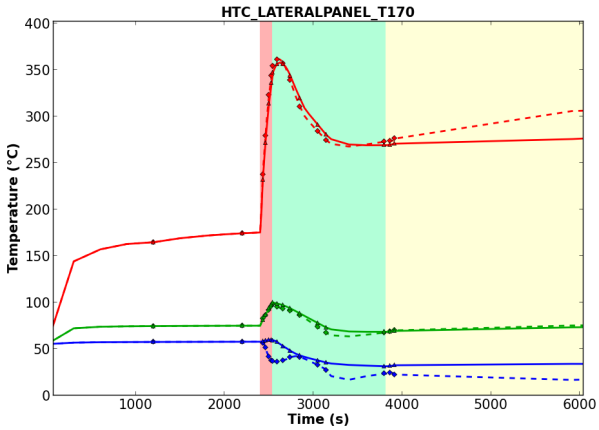
ROMs demonstrated their capabilities to substitute CFD models even for complex flow patterns encountered on lateral panels. The interface between core flow and fan flow seems to be quite well predicted. The important uncertainties of the surrogate models have a low impact of the statistics of interest: maximal, mean, minimal temperature.

6.3.3 Exhaust

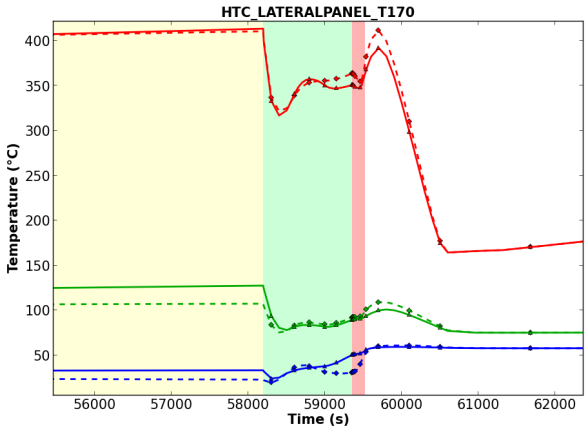
Figure 6.15 illustrates the transient behavior for the full mission. Even if the exhaust shows a great temperature range, from 200°C to 750°C, ROMs achieve impressive results for the exhaust. Only



(a) Overall behavior.

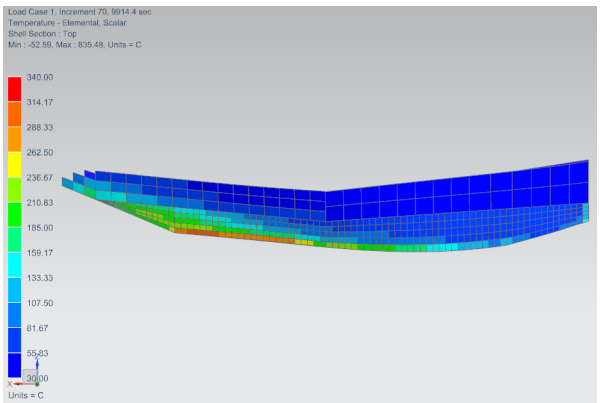


(b) Zoom on take-off and climb.

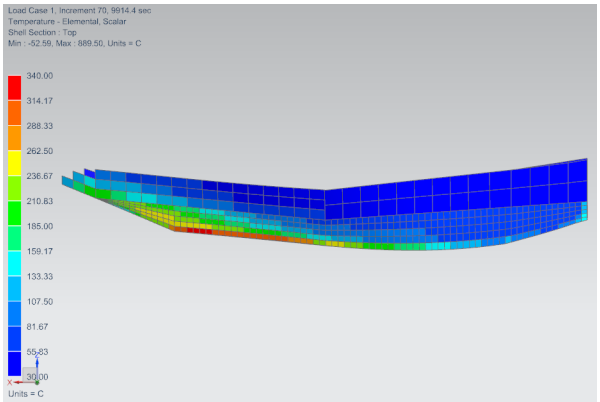


(c) Zoom on descent and landing.

Figure 6.12: Thermal analysis of the lateral panels.

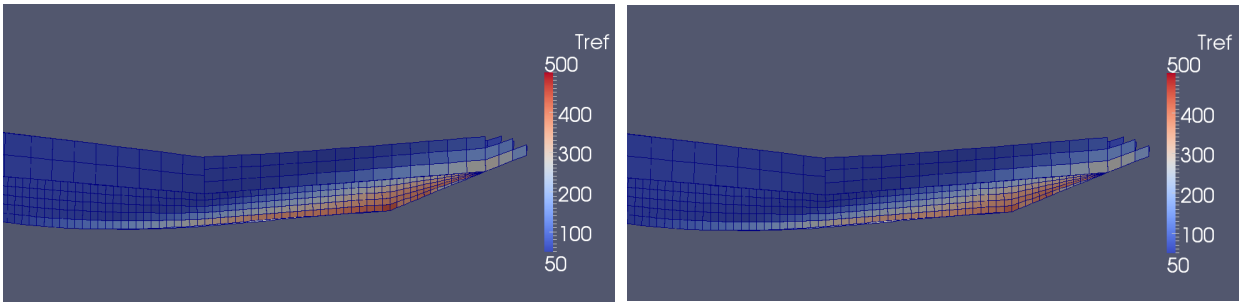


(a) CFD



(b) Surrogate

Figure 6.13: Instantaneous temperature on right panels at time step $t = 9,914$ seconds.



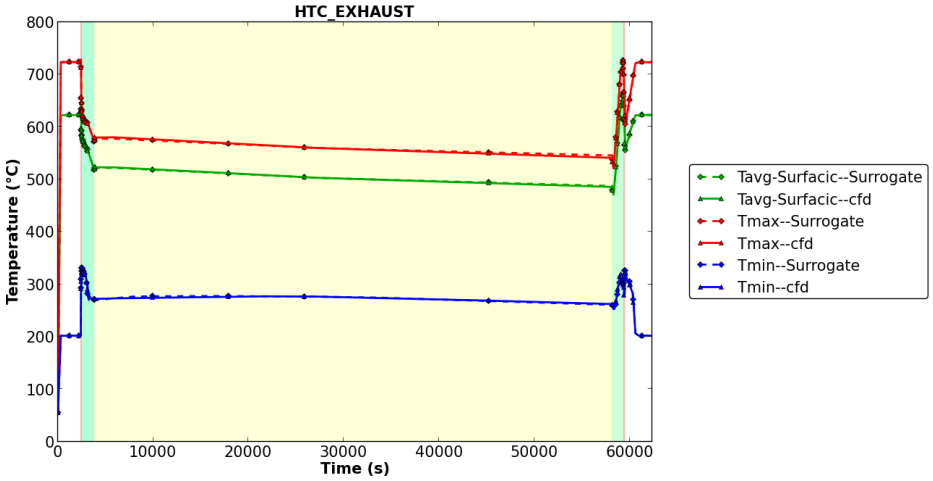
(a) CFD (b) Surrogate

Figure 6.14: Reference temperature on left panels at time step $t = 58,567$ seconds.

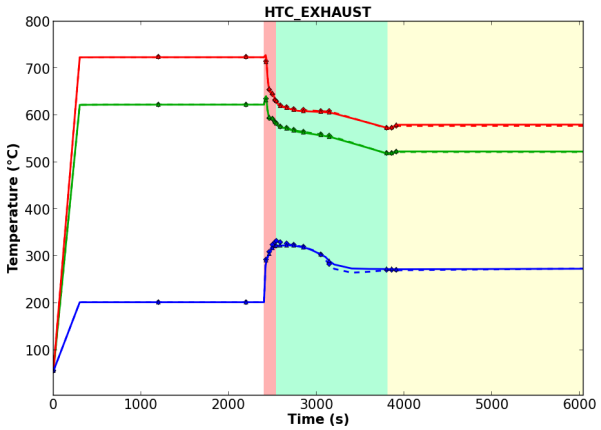
small discrepancies are observed between CFD and surrogate models, even for the cruise: the three statistics are in close agreement. Therefore, it is very likely that the overall shape of the flow and extreme values are well predicted for both heat transfer coefficient and reference temperature.

This result can be explained as the flow around the exhaust is mainly, or even only, driven by engine core temperature and pressure. In addition, the flow is very uniform and its shape shows small variability. In these conditions, ROMs can be expected to be very accurate, as POD modes can be highly representative of the flow.

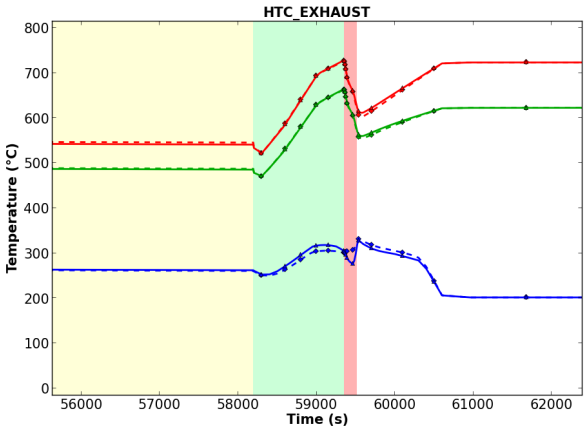
Different results are obtained for exhaust, nozzle, and lateral panels. These gaps highlight the various factors that influence thermal fields: the sensibilities are not uniform. However, global ROMs have regression models which cover the entire domain. Thus, building spatial local models may generate more relevant ROMs, closely related to the different boundary conditions and their parameters. The definition of these local models can be guided by the a-priori error and the analysis of the spatial sensibilities.



(a) Overall behavior.



(b) Zoom on take-off and climb.



(c) Zoom on descent and landing.

Figure 6.15: Thermal analysis of the exhaust.

6.4 Conclusion and main findings

Applying ROMs to an industrial aerothermal process yielded interesting results. First of all, a-priori analyses were identified as very useful tools to evaluate the robustness and the accuracy of surrogate models. Leave-one-out cross validations determined that 50 samples were sufficient to reach satisfactory accuracy, while the estimation of variance, given by GPR, provided important information on the sensibilities of surrogate models and their potential weaknesses.

Thermal simulations were performed twice: with surrogate models and with expensive CFD. Satisfactory results were obtained, as overall trends were in agreement for both methods. In particular, flight phases involving widely varying conditions, such as climb or descent, were very well predicted with very few errors. The thermal behavior was different on each component, as they were driven by different boundary conditions and input parameters. Main errors were only observed when influential parameters reached bounds of the training set.

Thus, several improvements can be suggested:

- critical points were predicted by the regression models. Thus, classical resampling strategies, such as IMSE, could be used to improve the accuracy and the quality of the DOE. As climb and descent were very well predicted, reducing the number of training samples in these phases could be considered in order to allocate computational budget to phases identified a-priori as more challenging;
- uncertainties provided by GPR could be integrated to the aerothermal process in order to get confidence interval on the transient response. Nevertheless, such an integration needs further investigations. For instance, reference temperatures are necessarily bounded by core or fan temperatures, while applying the high variances on the lateral panel can lead to unrealistic temperature (negative or far beyond melting point of the thermal protection). Such physical considerations should be taken into account to avoid irrelevant values;
- a clear discontinuity was noticed on lateral panels due to the interface between engine fan flow and engine core flow. This typical behavior can be related to the LDM developed in Chapter 4. First investigations concluded that snapshots could be separated into different clusters but a great deal of work remains to be done to develop relevant sensors and to analyze influences of each variable on the parameter space decomposition;
- the training set was one of the main limitation of ROMs. A strong assumption was made on engine parameters, since all the temperatures and pressures were uncorrelated. It may lead to divergent or unphysical simulations. The engine parameters lie on a manifold driven by several latent variables, such as the thrust. Thus, improving surrogate models for aerothermal applications may require to consider engine models or more physical relationships. The dimension of the parameter space will be also reduced.

Finally, 50 samples training a model predicting only 26 mission points may seem excessive. However, surrogate models are able to predict any kind of missions (long range, short range, various cruise level, different engine conditions, etc.), as long as they are inside the training set. Moreover, this first work on aerothermal simulations was very conservative. Models can be enhanced with above suggestions, reducing potentially the number of training samples.

Part IV

Conclusion

CHAPTER 7

Conclusion and perspectives

Surrogate models represent a valuable resource to support aircraft design analysis. The high-fidelity numerical tools are indeed usually computationally expensive in terms of CPU time and surrogate models can reduce the time to deliver the results to the design team. This thesis has put the emphasis on the applicability of surrogate models in an industrial environment for two specific disciplines: the prediction of aerodynamic loads over the aircraft skin and the aerothermal engineering of the engine/pylon interaction. As regards aerodynamics, the aero-load data within the flight envelope can be computed with surrogate models at low computational cost or with high-fidelity computational fluid dynamic at high computational cost. However, the complex phenomena encountered in aerodynamics during the flight pose problems for surrogate models, as they can generate high discrepancies in certain areas of the flight envelope. The Local Decomposition Method (LDM) is proposed to alleviate these issues by building local models instead of a global one. For its part, the aerothermal analysis encompasses two intersecting fields: fluid mechanics and thermal physics. The problems under study are usually transient and the coupling between the two disciplines becomes intractable for complex geometries. A classical solution consists in enabling only one way coupling from the discipline with the strongest influence onto the second discipline. Even with this simplification, computational cost is still very high and the fidelity of the fluid model is usually reduced. Thus, implementing a fluid surrogate model, built on high-fidelity simulations, in an aerothermal process could improve accuracy while maintaining the computational cost to an acceptable level.

Outcomes

An abundant literature deals with surrogate models for fluid problems. Nevertheless, the fields of application in this thesis impose two strong constraints to the surrogate model: to be pluggable into industrial tools and fully substitute numerical simulations. In other words, the surrogate model must be able to generate fields with thousands of degrees of freedom for each prediction. Amongst all the existing methods, non-intrusive reduced order models appear as a relevant solution. They generally use a Proper Orthogonal Decomposition approach to extract dominant shapes, called

modes, within the whole training set and recombine them to build predictions. However, a review of the literature demonstrates that exploring the flight envelope of an aircraft, associated to various changes in the input parameters, raises major problems for these models, since a mixture of different physical regimes is found in the modes. If the interpolation models ensuring the recombination of the modes are not sufficiently trained, significant prediction errors can be observed due to the amplification of spurious physical regimes.

The Local Decomposition Method (LDM) developed in this thesis mitigates some of the limitations encountered by the global reduced order model. It proposes to separate the dominant shapes of the solutions in different training sets in order to avoid mixing different physical regimes. Such separations aim to specialize the local models for a specific regime. The partition of the training set is ensured by two machine learning methods: the clustering and classification algorithms, which represent the central element of the method and are coupled with a physical sensor to ease the detection of the physical regimes. Moreover, an adaptive resampling strategy was proposed. It takes advantage of the partitioning of the training set by adding extra computations to the more challenging areas of the parameter space. Statistical metrics are computed on each subset in order to identifiy the local models at risk of being less accurate. This strategy successfully expand the database with simulations associated to high gradients and discontinuities. One of the major limitations of the LDM relates to the correct setting of the model, which is why different tools were introduced to guide the choice of the numerical parameters, such as cross-validation or coherence of the clusters. Finally, the efficiency of the LDM was assessed on two academic test cases: the one dimensional Burgers' equation and the turbulent flow around the two-dimensional airfoil RAE2822. Two clusters were clearly identified for both cases, associated to the presence or not of discontinuities. The LDM was shown to be much more accurate and robust than the global reduced order models. For instance, amongst the various error metrics employed, the mean relative error was reduced by at least 30%. These encouraging results have motivated the application of the method developed in this thesis to more realistic problems.

The LDM was studied on two complex aircraft configurations: the AS28G and the XRF-1. The goal was to bridge the gap between academic research and industrial needs. While the first application involves only 3 input parameters, the flight envelope of the XRF-1 is described by 7 parameters. Thus, the LDM was assessed with two different levels of complexity. A particular attention was paid to the influence of the number of clusters. This analysis provided insight into how the training set is interpreted in terms of flow regime, since a unique clustering of the training set is rarely achieved. For the simplest case, the AS28G, the model with two clusters gave the most impressive results by separating the parameter space into a subsonic and a transonic subset. However, using three clusters also provided interesting conclusions where the third cluster plays an intermediary role between the subsonic and the deep transonic regions. As regards the XRF-1 application, determining the optimum number of clusters was really challenging. The analysis of the results resulted in several guidelines: (i) a small number of clusters must be preferred with few training samples compared to the size of the input parameter, (ii) the local leave-one-out error provides a reliable estimation of the cluster quality, and (iii) resampling in specific clusters does not necessary stabilize their shape and their size. Finally, the application of the LDM on aircraft configurations demonstrated that the methodology can be successfully generalized to complex three-dimensional cases. In conclusion, the LDM was shown to improve the accuracy of the surrogate model for various different numerical settings and conditions.

The aerothermal application focused on pylon sizing and on integrating surrogate models into an industrial process involing a CFD software and a thermal software. The objective was to compare capabilities of surrogate models with a reference case built with CFD simulations. The fluid model, either CFD or surrogate models, predicted the boundary conditions of the thermal

model for forced convection over a full flight mission. Surrogate models provided a-priori methods able to analyze the overall quality of the model but also each prediction. These tools gave relevant information guiding the aerothermal process. Moreover, the transient thermal behaviors predicted using surrogate models and CFD were in close agreement. Only local discrepancies were observed but they were largely anticipated by the a-priori tools. Therefore, this work gave a first insight into the possibility of using data-driven approaches in aerothermal applications.

Programming several scripts also represents an important part of the thesis outcomes. An existing software building surrogate model was enhanced by implementing new elements, such as designs of experiments, regression methods, or resampling strategies. The LDM was also integrated into this software, which contributed to adding new statistical and visualization tools. Furthermore, scripts based on the Padge library were developed to enable information exchange between the surrogate models, the fluid model, and the thermal model.

Perspectives

The Local Decomposition Method At the current status of the LDM, several improvements can be suggested. Certain aspects of the method were chosen and implemented for their robustness and simplicity, which is why more sophisticated tools can be proposed. Besides, multiple strategies developed in the literature and not considered in the thesis can greatly enhance the LDM. Progresses can be made on the following aspects:

1. Fully automated process: The application of the LDM on various test cases highlights that the results can be very sensitive to the choice of some numerical parameters. For instance, the number of clusters can lead to different interpretations of the data. Several statistical tools were introduced but human and technical expertise is still needed to validate the setting of the surrogate model. For this reason, developing a fully automated process would represent a clear added value to the method. Ideally, the process should also be able to decide to use the LDM or a global surrogate model by analyzing the training set.

2. Interface management: The LDM introduces interfaces between the local models by clustering the parameter space. The quality of the prediction drops in these border regions mainly due to the regression method which behaves poorly in extrapolation. The solutions proposed in the thesis only modify the choice of the local model or enrich the border of the clusters with approximations from the adjacent clusters. A possible solution, called fuzzy clustering, would consist in modifying the clustering algorithm by allowing a percentage of overlapping between the clusters.

3. Extension of the application: Our approach focuses on aerodynamic problems, thus motivating the use of a shock sensor. However, the LDM could be generalized to various other problems. For instance, partial differential equations with bifurcating solutions might benefit from a local approach. Machine learning algorithms could be used to identify the different stationary points of the system as clusters, guided by the sudden change in the outputs. Other applications can be imagined, as long as the substituted system shows heterogeneities in the solution related to variations in the input parameter space.

4. Resampling: Although the additional snapshots on each cluster are usually computed based on the continuation of a low discrepancy sequence, they are designed to fulfill the initial design space and not clusters with very specific shapes. Thus, resampling methods taking into account the existing snapshots should be preferred. Moreover, seeking the new points in

the space of the shock sensor and not in the initial design space could be envisioned in order to find new features of the solution which have not yet been discovered.

Thermal mapping The aerothermal application presented in this thesis is based on a global reduced-order model. Important work remains to be achieved to analyze deeply all the results and to enhance surrogate models. Several potential sources of improvement can be proposed. As regards the case of the pylon design, the sensitivities to the different input variables are located at several locations. Thus, a spatial decomposition of the computational domain, presented in the literature review, could improve the accuracy and the robustness of the surrogate model. This is explained in part by the structure of the problems: the input parameters are directly related to the boundary conditions and the latter are spatially well separated regarding engine conditions. Moreover, discontinuities encountered on lateral panels can be related to the LDM. Further developments could improve predictions on this component by separating different scenarii. For instance, two cases can be identified a-priori: when all panels are at a uniform temperature close to the engine fan temperature or when the core flow reach first panels creating a discontinuity.

Interactions between the fluid and the thermal solver could also be more deeply analyzed. The surrogate model could be used to compute sensitivities and to estimate the errors of the model. Thus, the uncertainties given by the fluid surrogate model could be propagated through the thermal model. The analysis of the whole aerothermal process could provide very interesting insight to improve the sampling or accuracy of the surrogate model. Finally, the use of surrogate models for the thermal mapping could be extended to other applications than the sizing of the pylon. For instance, flight test analysis could benefit from these methods, since they are very usually carried out in the different conditions as initial simulations due to weather, flight control, measurement uncertainties, or other external imperatives. Therefore, surrogate models could help to readjust the aerothermal simulations through new boundary conditions and by generating confidence intervals.

Data fusion and machine learning The LDM demonstrates that surrogate modelling, in particular reduced order models, can benefit from an extensive use of machine learning techniques. One of the next challenges is to develop a tool able to build surrogate models by exploiting data of various types. Ideally, such a tool might integrate multifidelity models, gradient information, or even wind-tunnel and flight test data. Matured strategies have already been developed to manage one specific type of data, for instance gradient-enhanced Kriging or multi-fidelity surrogate models. However coupling all these heterogeneous data together remains a challenge and machine learning methods can propose promising solutions. They could be used to learn various features from all the available data, which could guide the generation of the most relevant surrogate model. The LDM has contributed to improving the use of machine learning in surrogate modelling but future research work must still be carried out to fully integrate these two disciplines.

Bibliography

[1] Standard atmosphere first edition. Corrigendum 1, ISO International Standard 2533-1975, 1978. Geneva, Switzerland. *Cited in page 126*

[2] A. Abbas-Bayoumi and K. Becker. An industrial view on numerical simulation for aircraft aerodynamic design. *Journal of Mathematics in Industry*, 1(1):10, 2011. doi: 10.1186/2190-5983-1-10. *Cited in pages 12, 13, 14, and 15*

[3] Adams, Bauman, Bohnhoff, Dalbey, Ebeida, Eddy, Eldred, Hough, Hu, Jakeman, Stephens, Swiler, Vigil, and Wildey. Dakota, a multilevel parallel object-oriented framework for design optimization, parameter estimation, uncertainty quantification, and sensitivity analysis: Version 6.0 user's manual. Technical report, Sandia National Laboratories, July 2014. *Cited in page 64*

[4] D. Alonso, J. Vega, and A. Velazquez. Reduced-order model for viscous aerodynamic flow past an airfoil. *AIAA journal*, 48(9):1946–1958, 2010. *Cited in page 20*

[5] D. Alonso, J. M. Vega, Á. Velázquez, and V. de Pablo. Reduced-order modeling of three-dimensional external aerodynamic flows. *Journal of Aerospace Engineering*, 25(4):588–599, 2011. *Cited in page 20*

[6] D. Amsallem and C. Farhat. Interpolation method for adapting reduced-order models and application to aeroelasticity. *AIAA journal*, 46(7):1803–1813, 2008. *Cited in page 19*

[7] D. Amsallem, M. J. Zahr, and C. Farhat. Nonlinear model order reduction based on local reduced-order bases. *International Journal for Numerical Methods in Engineering*, 92(10): 891–916, 2012. ISSN 1097-0207. doi: 10.1002/nme.4371. *Cited in pages 39, 102, and 105*

[8] D. Amsallem, C. Farhat, and M. Zahr. On the robustness of residual minimization for constructing pod-based reduced-order cfd models. In *21st AIAA Computational Fluid Dynamics Conference*, San Diego, CA, June 2013. AIAA. *Cited in page 20*

[9] C. A. Andrews, J. M. Davies, and G. R. Schwarz. Adaptive data compression. In *Proc. IEEE*, volume 55, March 1967. *Cited in page 19*

[10] E. Baalbergen, J. Kos, C. Louriou, C. Campguilhem, and J. Barron. Streamlining cross-organisation product design in aeronautics. *Proceedings of the Institution of Mechanical*

Engineers, Part G: Journal of Aerospace Engineering, 231(12):2192–2202, 2017. doi: 10.1177/0954410017716480. *Cited in pages 3 and 5*

[11] T. Bach, T. Führer, C. Willberg, and S. Dähne. Automated sizing of a composite wing for the usage within a multidisciplinary design process. *Aircraft Engineering and Aerospace Technology: An International Journal*, 88(2):303–310, 2016. *Cited in page 160*

[12] M. Baudin, A. Dutfoy, B. Iooss, and A.-L. Popelin. Openturns: An industrial software for uncertainty quantification in simulation. *Handbook of Uncertainty Quantification*, pages 2001–2038, 2017. *Cited in page 99*

[13] R. Bellman. Adaptive control processes - a guided tour. *Princeton University Press*, 1961. *Cited in pages 35 and 105*

[14] T. Benamara, P. Breitkopf, I. Lepot, and C. Sainvitu. Adaptive infill sampling criterion for multi-fidelity optimization based on gappy-pod. *Structural and Multidisciplinary Optimization*, 54(4):843–855, 2016. ISSN 1615-1488. doi: 10.1007/s00158-016-1440-3. *Cited in pages 18, 34, 43, and 44*

[15] P. Benner, S. Gugercin, and K. Willcox. A survey of projection-based model reduction methods for parametric dynamical systems. *SIAM Review*, 57(4):483–531, 2015. doi: 10.1137/130932715. *Cited in pages 17, 18, 20, and 31*

[16] M. Bergmann, A. Ferrero, A. Iollo, E. Lombardi, A. Scardigli, and H. Telib. A zonal galerkin-free pod model for incompressible flows. *Journal of Computational Physics*, 352:301–325, 2018. *Cited in page 37*

[17] D. Bettebghor, N. Bartoli, S. Grihon, J. Morlier, and M. Samuelides. Surrogate modeling approximation using a mixture of experts based on em joint estimation. *Structural and Multidisciplinary Optimization*, 43(2):243 – 259, february 2011. *Cited in pages 38 and 118*

[18] C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006. ISBN 0387310738. *Cited in pages 63, 64, 66, 67, 68, 105, 106, 107, 114, and 115*

[19] M. A. Bouhlel and J. R. R. A. Martins. Gradient-enhanced kriging for high-dimensional problems. *Engineering with Computers*, February 2018. ISSN 1435-5663. doi: 10.1007/s00366-018-0590-x. *Cited in page 35*

[20] M. A. Bouhlel, N. Bartoli, A. Otsmane, and J. Morlier. Improving kriging surrogates of high-dimensional design models by partial least squares dimension reduction. *Structural and Multidisciplinary Optimization*, 53(5):935–952, 2016. *Cited in page 36*

[21] T. Braconnier and M. Ferrier. Decomposition (jpod) for steady aerodynamic model - part i: Theory. *Internal Report*, February 2009. *Cited in pages 59, 60, and 99*

[22] T. Braconnier, M. Ferrier, J.-C. Jouhaud, M. Montagnac, and P. Sagaut. Towards an adaptive pod/svd surrogate model for aeronautic design. *Computers and Fluids*, 40, January 2011. doi: 10.1016/j.compfluid.2010.09.002. *Cited in pages 29, 31, 43, 44, and 99*

[23] T. Bui-Thanh, M. Damodaran, and K. Willcox. Proper orthogonal decomposition extensions for parametric applications in compressible aerodynamics. *Math. Cmmput. Model.*, 33, 2003. doi: 10.1016/S0895-7177(00)00240-5. *Cited in pages 60 and 61*

[24] T. Bui-Thanh, M. Damodaran, and K. Willcox. Proper orthogonal decomposition extensions for parametric applications in transonic aerodynamics. In *15th AIAA Computational Fluid Dynamics Conference, AIAA Paper 2003-4213*, Orlando, FL, June 2003. *Cited in pages 28, 43, and 44*

[25] T. Bui-Thanh, M. Damodaran, and K. E. Willcox. Aerodynamic data reconstruction and inverse design using proper orthogonal decomposition. *AIAA journal*, 42(8):1505–1516, 2004. *Cited in pages 19, 34, 43, and 44*

[26] T. Bui-Thanh, K. Willcox, and O. Ghattas. Model reduction for large-scale systems with high-dimensional parametric input space. *SIAM Journal on Scientific Computing*, 30(6): 3270–3288, 2008. *Cited in page 31*

[27] J. M. Burgers. Mathematical model illustrating the theory of turbulence. *Advances in Applied Mechanics*, 1:171–199, 1948. *Cited in page 119*

[28] E. Burnaev and M. Panov. Adaptive design of experiments based on gaussian processes. In *International Symposium on Statistical Learning and Data Sciences*, pages 116–125. Springer, 2015. *Cited in pages 31 and 92*

[29] R. H. Byrd, P. Lu, J. Nocedal, and C. Zhu. A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, 16, 09 1995. doi: 10.1137/0916069. *Cited in page 80*

[30] L. Cambier and J. Veuillot. Status of the elsa cfd software for flow simulation and multidisciplinary applications. In *46th AIAA Aerospace Science Meeting and Exhibit*, Reno, USA, January 2008. *Cited in page 146*

[31] L. Cambier, S. Heib, and S. Plot. The onera elsa cfd software: input from research and feedback from industry. *Mechanics & Industry*, 14(3):159–174, 2013. *Cited in page 125*

[32] K. Carlberg and C. Farhat. A low-cost, goal-oriented 'compact proper orthogonal decomposition'basis for model reduction of static systems. *International Journal for Numerical Methods in Engineering*, 86(3):381–402, 2011. *Cited in page 35*

[33] E. Chaput, L. Barrera, C. Gacherieu, and L. P. Tourette. Navier-stokes analysis for engine airframe integration. In *ICAS 20th Congress of the International Council of the Aronautical Sciences, ICAS-96-4.7.3*, Sorrento, Napoly, Italy, September 1996. *Cited in page 146*

[34] A. Chatterjee. An introduction to the proper orthogonal decomposition. *Current science*, 78 (7), April 2000. *Cited in pages 19, 54, and 55*

[35] J. Chen, B. Xin, Z. Peng, L. Dou, and J. Zhang. Optimal contraction theorem for exploration-exploitation tradeoff in search and optimization. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 39(3):680–691, May 2009. ISSN 1083-4427. doi: 10.1109/TSMCA.2009.2012436. *Cited in page 31*

[36] X. Chen, L. Liu, T. Long, and Z. Yue. A reduced order aerothermodynamic modeling framework for hypersonic vehicles based on surrogate and pod. *Chinese Journal of Aeronautics*, 28(5):1328–1342, 2015. *Cited in pages 41, 43, and 44*

[37] H.-S. Chung and J. Alonso. Using gradients to construct cokriging approximation models for high-dimensional design optimization problems. In *40th AIAA Aerospace Sciences Meeting & Exhibit*, page 317, Reston, Virginia, January 2002. *Cited in pages 18 and 35*

[38] P. G. Cizmas, B. R. Richardson, T. A. Brenner, T. J. O'Brien, and R. W. Breault. Acceleration techniques for reduced-order models based on proper orthogonal decomposition. *Journal of Computational Physics*, 227(16):7791–7812, 2008. *Cited in page 39*

[39] P. G. Constantine and G. Iaccarino. Reduced order models for parameterized hyperbolic conservations laws with shock reconstruction. *Center for Turbulence Research Annual Research Briefs*, pages 115–126, 2012. *Cited in pages 37 and 105*

[40] P. G. Constantine, E. Dow, and Q. Wang. Active subspace methods in theory and practice: applications to kriging surfaces. *SIAM Journal on Scientific Computing*, 36(4):A1500–A1524, 2014. *Cited in page 36*

[41] P. G. Constantine, M. Emory, J. Larsson, and G. Iaccarino. Exploiting active subspaces to quantify uncertainty in the numerical simulation of the hyshot ii scramjet. *Journal of Computational Physics*, 302:1–20, 2015. *Cited in page 36*

[42] P. Cook, M. Firmin, M. McDonald, and R. A. Establishment. *Aerofoil RAE 2822: Pressure Distributions, and Boundary Layer and Wake Measurements*. Technical memorandum / Royal Aircraft Establishment. RAE, 1977. *Cited in page 125*

[43] L. Cordier and M. Bergmann. Two typical applications for pod: Coherent structures eduction and reduced order modelling. In *Lecture series 2002-04, 2003-03 and 2008-01 on post-processing of experimental and numerical data*, page 60. von Karman Institute for Fluid Dynamics, 2008. *Cited in page 60*

[44] L. Cordier, J. Delville, and J. Bonnet. Review of some fundamentals of data processing: Proper orthogonal decomposition. In *Handbook of Experimental Fluid Mechanics*. Spring-Verlag, 2007. *Cited in pages 17, 40, 53, 54, 55, 56, 57, 58, 60, and 116*

[45] N. Cressie. The origins of kriging. *Mathematical Geology*, 22(3):239–252, Apr 1990. ISSN 1573-8868. doi: 10.1007/BF00889887. *Cited in page 75*

[46] K. Crombecq, D. Gorissen, D. Deschrijver, and T. Dhaene. A novel hybrid sequential design strategy for global surrogate modeling of computer experiments. *SIAM Journal on Scientific Computing*, 33(4):1948–1974, 2011. *Cited in page 30*

[47] L. Daniel, O. C. Siong, L. S. Chay, K. H. Lee, and J. White. A multiparameter moment-matching model-reduction approach for generating geometrically parameterized interconnect performance models. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 23(5):678–693, 2004. *Cited in page 17*

[48] *MACROS : Generic Tool for Design of Experiments*. Datadvance, 2013. *Cited in pages 89 and 91*

[49] J. Degroote, J. Vierendeels, and K. Willcox. Interpolation among reduced-order matrices to obtain parameterized models for design, optimization and probabilistic analysis. *International Journal for Numerical Methods in Fluids*, 63(2):207–230, 2010. *Cited in pages 19 and 20*

[50] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39 (1):1–38, 1977. ISSN 00359246. *Cited in page 108*

[51] V. Dolci and R. Arina. Proper orthogonal decomposition as surrogate model for aerodynamic optimization. *International Journal of Aerospace Engineering*, 2016, 2016. *Cited in pages 22, 29, 43, and 44*

[52] H. Drucker, C. J. C. Burges, L. Kaufman, A. J. Smola, and V. Vapnik. *Support Vector Regression Machines*. MIT Press, 1997. *Cited in page 64*

[53] R. Dupuis, J.-C. Jouhaud, and P. Sagaut. Aerodynamic data predictions for transonic flows via a machine-learning-based surrogate model. In *AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, AIAA SciTech Forum*, Kissimmee, Florida, January 2018. AIAA 2018-1905. *Cited in pages 43, 44, and 146*

[54] R. Dupuis, J.-C. Jouhaud, and P. Sagaut. Surrogate modeling of aerodynamic simulations for multiple operating conditions using machine learning. *AIAA Journal*, 56(9):3622–3635, September 2018. ISSN 0001-1452. doi: 10.2514/1.J056405. *Cited in pages 43, 44, and 102*

[55] K. Duraisamy, G. Iaccarino, and H. Xiao. Turbulence modeling in the age of data. *arXiv preprint arXiv:1804.00183*, 2018. *Cited in page 23*

[56] D. Duvenaud. *Automatic model construction with Gaussian processes*. PhD thesis, University of Cambridge, 2014. *Cited in pages 78 and 79*

[57] B. Echard, N. Gayton, M. Lemaire, and N. Relun. A combined importance sampling and kriging reliability method for small failure probabilities with time-demanding numerical models. *Reliability Engineering & System Safety*, 111:232–240, 2013. *Cited in page 82*

[58] C. Eckart and G. Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218, Sep 1936. ISSN 1860-0980. doi: 10.1007/BF02288367. *Cited in page 59*

[59] M. S. Eldred, A. A. Giunta, S. S. Collis, N. A. Alexandrov, and R. M. Lewis. Second-order corrections for surrogate-based optimization with model hierarchies. In *11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, September 2004. doi: 10.2514/6.2004-4457. AIAA 2004-4457. *Cited in page 17*

[60] B. Epureanu. A parametric analysis of reduced order models of viscous flows in turbomachinery. *Journal of Fluids and Structures*, 17(7):971 – 982, 2003. ISSN 0889-9746. doi: https://doi.org/10.1016/S0889-9746(03)00044-6. *Cited in page 19*

[61] C. B. Erickson, B. E. Ankenman, and S. M. Sanchez. Comparison of gaussian process modeling software. *European Journal of Operational Research*, 266(1):179 – 192, 2018. ISSN 0377-2217. doi: https://doi.org/10.1016/j.ejor.2017.10.002. *Cited in pages 98 and 99*

[62] L. Eriksson, E. Johansson, N. Kettaneh-Wold, C. Wikström, and S. Wold. Design of experiments. *Principles and Applications, Learn ways AB, Stockholm*, 2000. *Cited in page 82*

[63] M.-P. Errera and B. Baqué. A quasi-dynamic procedure for coupled thermal simulations. *International Journal for Numerical Methods in Fluids*, 72(11):1183–1206, 2013. ISSN 1097-0363. doi: 10.1002/fld.3782. *Cited in page 174*

[64] B. Estebe and J. L. Montes. Thermal simulation in aircraft development - a global approach, innovation into perspective. In *6th European Conference for Aeronautic and Space Sciences*, Kracow, Poland, June 2013. EUCASS. *Cited in page 15*

[65] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, pages 226–231. AAAI Press, 1996. *Cited in pages 109, 110, and 111*

[66] R. Everson and L. Sirovich. Karhunen–loeve procedure for gappy data. *JOSA A*, 12(8): 1657–1664, 1995. *Cited in page 19*

[67] M. Fahl. *Trust-Region methods for flow control based on Reduced Order Modeling*. PhD thesis, Trier university, 2000. *Cited in pages 55 and 59*

[68] K.-T. Fang, C.-X. Ma, and P. Winker. Centered l_2-discrepancy of random sampling and latin hypercube design, and construction of uniform designs. *Mathematics of Computation*, 71(237):275–296, 2002. *Cited in pages 83 and 88*

[69] K.-T. Fang, R. Li, and A. Sudjianto. *Design and modeling for computer experiments*. Chapman and Hall/CRC, 2005. *Cited in pages 62, 64, 80, 81, 83, and 84*

[70] M. G. Fernández-Godino, C. Park, N.-H. Kim, and R. T. Haftka. Review of multi-fidelity models. *arXiv preprint arXiv:1609.07196*, 2016. *Cited in page 32*

[71] R. A. Fisher. *The design of experiments*. Oliver And Boyd; Edinburgh; London, 1937. *Cited in page 82*

[72] A. I. J. Forrester, N. W. Bressloff, and A. J. Keane. Optimization using surrogate models and partially converged computational fluid dynamics simulations. In *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, volume 462, pages 2177–2204. The Royal Society, 2006. *Cited in pages 17, 18, 32, and 34*

[73] A. I. J. Forrester, S. András, and A. J. Keane. *Engineering Design via Surrogate Modelling: a practical guide*. Wiley, 2008. *Cited in pages 18, 32, 33, 62, 65, 67, 68, 74, 75, and 87*

[74] M. Fossati. Evaluation of aerodynamic loads via reduced-order methodology. *AIAA Journal*, 53(8):2389–2405, 2015. doi: 10.2514/1.J053755. *Cited in pages 40, 43, and 44*

[75] M. Fossati and W. G. Habashi. Multiparameter analysis of aero-icing problems using proper orthogonal decomposition and multidimensional interpolation. *AIAA Journal*, 51(4):946–960, 2013. *Cited in pages 40, 42, 43, and 44*

[76] T. Franz. *Reduced-order modeling for steady transonic flows via manifold learning*. PhD thesis, Feb 2015. *Cited in pages 40, 42, 43, and 44*

[77] T. Franz, R. Zimmermann, S. Görtz, and N. Karcher. Interpolation-based reduced-order modelling for steady transonic flows via manifold learning. *International Journal of Computational Fluid Dynamics*, 28(3-4):106–121, 2014. doi: 10.1080/10618562.2014.918695. *Cited in page 40*

[78] J. Friedman, T. Hastie, and R. Tibshirani. *The elements of statistical learning*, volume 1. Springer series in statistics. *Cited in pages 64, 98, 106, 107, and 113*

[79] F. Gallard. *Aircraft shape optimization for mission performance*. PhD thesis, INP Toulouse, 2014. *Cited in page 13*

[80] A. Gazaix, F. Gallard, V. Gachelin, T. Druot, S. Grihon, V. Ambert, D. Guénot, R. Lafage, C. Vanaret, B. Pauwels, et al. Towards the industrialization of new mdo methodologies and tools for aircraft design. In *18th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, page 3149, 2017. *Cited in pages 23 and 160*

[81] G. Geraci, G. Iaccarino, F. Montomoli, et al. Multi-fidelity uncertainty quantification using rans and dns. In *Center for Turbulence Research, Summmer Program*, 2016. *Cited in pages 32 and 33*

[82] D. Gorissen, I. Couckuyt, P. Demeester, T. Dhaene, and K. Crombecq. A surrogate modeling and adaptive sampling toolbox for computer based design. *Journal of Machine Learning Research*, 11(Jul):2051–2055, 2010. *Cited in page 18*

[83] M. D. Graham and I. G. Kevrekidis. Alternative approaches to the karhunen-loeve decomposition for model reduction and data analysis. *Computers & chemical engineering*, 20(5): 495–506, 1996. *Cited in page 22*

[84] M. A. Grepl and A. T. Patera. A posteriori error bounds for reduced-basis approximations of parametrized parabolic partial differential equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 39(1):157–181, 2005. *Cited in page 31*

[85] E. J. Grimme. *Krylov projection methods for model reduction*. PhD thesis, University of Illinois at Urbana-Champaign, 1997. *Cited in page 17*

[86] B. Haasdonk, M. Dihlmann, and M. Ohlberger. A training set and multiple bases generation approach for parameterized model reduction based on adaptive grids in parameter space. *Mathematical and Computer Modelling of Dynamical Systems*, 17(4):423–442, 2011. *Cited in page 39*

[87] W. Haase. *EUROVAL, an European initiative on validation of CFD codes: results of the EC/BRITE-EURAM project EUROVAL, 1990-1992*. Notes on numerical fluid mechanics. Vieweg, 1993. ISBN 9783528076429. *Cited in page 125*

[88] Z. Han, R. Zimmerman, and S. Görtz. Alternative cokriging method for variable-fidelity surrogate modeling. *AIAA journal*, 50(5):1205–1210, 2012. *Cited in page 33*

[89] Z.-H. Han and S. Görtz. Hierarchical kriging model for variable-fidelity surrogate modeling. *AIAA journal*, 50(9):1885–1896, 2012. *Cited in pages 18 and 33*

[90] Z.-H. Han, S. Görtz, and R. Hain. A variable-fidelity modeling method for aero-loads prediction. In *New results in numerical and experimental fluid mechanics vii*, pages 17–25. Springer, 2010. *Cited in page 33*

[91] S. Haykin. *Neural Networks: A Comprehensive Foundation*. Prentice Hall PTR, Upper Saddle River, NJ, USA, second edition, 1998. ISBN 0132733501. *Cited in pages 71 and 73*

[92] T. Henri and J.-P. Yvon. Convergence estimates of pod-galerkin methods for parabolic problems. *System Modeling and Optimization*, pages 295–306, 2005. *Cited in page 55*

[93] J. Hensman, N. Fusi, and N. D. Lawrence. Gaussian processes for big data. In *Uncertainty in Artificial Intelligence*, page 282, 2013. *Cited in page 80*

[94] J. Hesthaven and S. Ubbiali. Non-intrusive reduced order modeling of nonlinear problems using neural networks. *Journal of Computational Physics*, 363:55 – 78, 2018. ISSN 0021-9991. doi: https://doi.org/10.1016/j.jcp.2018.02.037. *Cited in page 21*

[95] F. Hickernell. A generalized discrepancy and quadrature error bound. *Mathematics of Computation of the American Mathematical Society*, 67(221):299–322, 1998. *Cited in page 83*

[96] E. Hopf. The partial differential equation $ut + uux = \mu xx$. *Communications on Pure and Applied Mathematics*, 3(3):201–230, 1950. ISSN 1097-0312. doi: 10.1002/cpa.3160030302. *Cited in page 119*

[97] K. Hornik. Approximation capabilities of multilayer feedforward networks. *Neural networks*, 4(2):251–257, 1991. *Cited in pages 69 and 73*

[98] L. Huang, Z. Gao, and D. Zhang. Research on multi-fidelity aerodynamic optimization methods. *Chinese Journal of Aeronautics*, 26(2):279–286, 2013. *Cited in pages 17 and 18*

[99] J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing In Science & Engineering*, 9(3):90–95, 2007. *Cited in page 99*

[100] B. Iooss and P. Lemaître. A review on global sensitivity analysis methods. In *Uncertainty management in simulation-optimization of complex systems*, pages 101–122. Springer, 2015. *Cited in page 35*

[101] B. Iooss, L. Boussouf, V. Feuillard, and A. Marrel. Numerical studies of the metamodel fitting and validation processes. *International Journal On Advances in Systems and Measurements*, 3:11–21, 2010. *Cited in page 94*

[102] E. Iuliano and Q. Domenico. Proper orthogonal decomposition, surrogate modelling and evolutionary optimization in aerodynamic design. *Computers & Fluids*, 84:327–350, June 2013. ISSN 00457930. doi: 10.1016/j.compfluid.2013.06.007. *Cited in pages 18, 38, 43, and 44*

[103] A. Jameson, W. Schmidt, and E. Turkel. Numerical solutions of the euler equations by finite volume methods using runge-kutta time-stepping schemes. In *AIAA 14th Fluid and Plasma Dynamic Conference*, volume 81. AIAA Paper, 1981. *Cited in pages 105, 125, and 146*

[104] R. Jin, W. Chen, and T. Simpson. Comparative studies of metamodelling techniques under multiple modelling criteria. *Structural and Multidisciplinary Optimization*, 23(1):1–13, December 2001. ISSN 1615-147X. doi: 10.1007/s00158-001-0160-4. *Cited in page 81*

[105] R. Jin, W. Chen, and A. Sudjianto. On sequential sampling for global metamodeling in engineering design. In *ASME 2002 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, pages 539–548. American Society of Mechanical Engineers, 2002. *Cited in page 93*

[106] F. T. Johnson, E. N. Tinoco, and N. J. Yu. Thirty years of development and application of cfd at boeing commercial airplanes, seattle. *Computers & Fluids*, 34(10):1115 – 1151, 2005. ISSN 0045-7930. doi: https://doi.org/10.1016/j.compfluid.2004.06.005. *Cited in page 14*

[107] M. E. Johnson, L. M. Moore, and D. Ylvisaker. Minimax and maximin distance designs. *Journal of statistical planning and inference*, 26(2):131–148, 1990. *Cited in pages 86 and 87*

[108] D. R. Jones, M. Schonlau, and W. J. Welch. Efficient global optimization of expensive black-box functions. *Journal of Global optimization*, 13(4):455–492, 1998. *Cited in pages 18, 93, and 120*

[109] M. I. Jordan and R. A. Jacobs. Hierarchical mixtures of experts and the em algorithm. *Neural computation*, 6(2):181–214, 1994. *Cited in page 38*

[110] M. H. Kalos and P. A. Whitlock. *Monte Carlo methods. 2nd revised and enlarged ed.* John Wiley & Sons, 2008. *Cited in pages 90 and 91*

[111] H. Kato and K.-i. Funazaki. Pod-driven adaptive sampling for efficient surrogate modeling and its application to supersonic turbine optimization. In *ASME Turbo Expo 2014: Turbine Technical Conference and Exposition*, volume 2, pages V02BT45A023–V02BT45A023. American Society of Mechanical Engineers, 2014. *Cited in pages 31, 43, and 44*

[112] M. C. Kennedy and A. O'Hagan. Predicting the output from a complex computer code when fast approximations are available. *Biometrika*, 87(1):1–13, 2000. *Cited in pages 32 and 33*

[113] G. K. W. Kenway and J. R. R. A. Martins. Multipoint high-fidelity aerostructural optimization of a transport aircraft configuration. *Journal of Aircraft*, 51:144–160, 2014. doi: 10.2514/1.C032150. *Cited in page 13*

[114] M. Kirby and L. Sirovich. Application of the karhunen-loeve procedure for the characterization of human faces. In *IEE T. Pattern Anal*, 1990. *Cited in page 19*

[115] M. Kirby, J. P. Boris, and L. Sirovich. A proper orthogonal decomposition of a simulated supersonic shear layer. *International Journal for Numerical Methods in Fluids*, 10(4):411–428, 1990. doi: 10.1002/fld.1650100405. *Cited in page 19*

[116] J. P. Kleijnen. *Design and analysis of simulation experiments*, volume 230 of *International Series in Operations Research & Management Science*. Springer, 2008. *Cited in pages 17 and 18*

[117] J. P. Kleijnen. Regression and kriging metamodels with their experimental designs in simulation: a review. *European Journal of Operational Research*, 256(1):1–16, 2017. *Cited in page 75*

[118] J. Koehler and A. Owen. Computer experiments. *Handbook of statistics*, 13:261–308, 1996. *Cited in pages 78 and 79*

[119] A. N. Kolmogorov. Interpolation and extrapolation of stationary random sequences. *Izv. Akad. Nauk SSSR, Ser. Mat.*, 5(1):3-14, 1941. *Cited in page 75*

[120] D. Krige. *A statistical approach to some mine valuation and allied problems on the Witwatersrand*. PhD thesis, University of the Witwatersrand, 1951. *Cited in page 75*

[121] N. Kroll, M. Abu-Zurayk, D. Dimitrov, T. Franz, T. Führer, T. Gerhold, S. Görtz, R. Heinrich, C. Ilic, J. Jepsen, J. Jägersküpper, M. Kruse, A. Krumbein, S. Langer, D. Liu, R. Liepelt, L. Reimer, M. Ritter, A. Schwöppe, J. Scherer, F. Spiering, R. Thormann, V. Togiti, D. Vollmer, and J.-H. Wendisch. Dlr project digital-x: towards virtual aircraft design and flight testing based on high-fidelity methods. *CEAS Aeronautical Journal*, 7(1):3–27, Mar 2016. ISSN 1869-5590. doi: 10.1007/s13272-015-0179-7. *Cited in pages 13 and 160*

[122] A. K. Kundu. *Aircraft Design*. Cambridge Aerospace Series. Cambridge University Press, 2010. doi: 10.1017/CBO9780511844652. *Cited in page 12*

[123] P. Laurenceau, J.; Sagaut. Building efficient response surfaces of aerodynamic functions with kriging and cokriging. *AIAA Journal*, 46, 02 2008. *Cited in page 35*

[124] S. Lawrence, C. L. Giles, and A. C. Tsoi. Lessons in neural network training: Overfitting may be harder than expected. In *Fourteenth National Conference on Artificial Intelligence*, 1997. *Cited in page 63*

[125] Y. Le Cun. Learning process in an asymmetric threshold network. In *Disordered systems and biological organization*, pages 233–240. Springer, 1986. *Cited in page 69*

[126] L. Le Gratiet. *Multi-fidelity Gaussian process regression for computer experiments*. PhD thesis, Université Paris-Diderot-Paris VII, 2013. *Cited in page 33*

[127] R. M. Lewis and S. G. Nash. Model problems for the multigrid optimization of systems governed by differential equations. *SIAM Journal on Scientific Computing*, 26(6):1811–1837, 2005. *Cited in page 17*

[128] J. Li and W. Zhang. The performance of proper orthogonal decomposition in discontinuous flows. *Theoretical and Applied Mechanics Letters*, 6(5):236 – 243, 2016. ISSN 2095-0349. doi: https://doi.org/10.1016/j.taml.2016.08.008. *Cited in page 22*

[129] R. P. Liem, C. A. Mader, and J. R. R. A. Martins. Surrogate models and mixtures of experts in aerodynamic perfomance prediction for aircraft mission analysis. *Aerospace Science and Technology*, 43, June 2015. *Cited in pages 35, 38, and 39*

[130] T. Lieu, C. Farhat, and M. Lesoinne. Reduced-order fluid/structure modeling of a complete aircraft configuration. *Computer methods in applied mechanics and engineering*, 195(41-43): 5730–5742, 2006. *Cited in page 19*

[131] H. Liu, Y.-S. Ong, and J. Cai. A survey of adaptive sampling for global metamodeling in support of simulation-based complex engineering design. *Structural and Multidisciplinary Optimization*, pages 1–24. *Cited in pages 30, 84, and 93*

[132] W. Liu and S. Batill. Gradient-enhanced response surface approximations using kriging models. In *9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, page 5456. *Cited in page 35*

[133] J. L. Loeppky, J. Sacks, and W. J. Welch. Choosing the sample size of a computer experiment: A practical guide. *Technometrics*, 51(4):366–376, 2009. doi: 10.1198/TECH.2009.08040. *Cited in pages 93 and 127*

[134] E. Lombardi, M. Bergmann, S. Camarri, and A. Iollo. Low-order models: optimal sampling and linearized control strategies. *Journal Européen des Systèmes Automatisés*, 2011. *Cited in page 19*

[135] S. N. Lophaven and H. B. Nielsen. *DACE: a Matlab kriging toolbox*, volume 2. Citeseer, 2002. *Cited in page 99*

[136] L. Lorente, J. Vega, and A. Velazquez. Generation of aerodynamic databases using high-order singular value decomposition. *Journal of Aircraft*, 45(5):1779–1788, 2008. *Cited in pages 37 and 105*

[137] D. J. Lucia, P. I. King, and P. S. Beran. Reduced order modeling of a two-dimensional flow with moving shocks. *Computers & Fluids*, 32(7):917–938, 2003. doi: 10.1016/S0045-7930(02)00035-X. *Cited in page 37*

[138] J. L. Lumley. The structure of inhomogeneous turbulence. In *Atmospheric Turbulence and Wave Propagation*. A.M. Yaglom, V.I. Tatarski, 1967. *Cited in pages 19 and 54*

[139] Z. Lyu, G. K. W. Kenway, and J. R. R. A. Martins. Aerodynamic shape optimization investigations of the common research model wing benchmark. *AIAA Journal*, 53(4):968–985, 2015. doi: 10.2514/1.J053318. *Cited in page 13*

[140] T. Mackman, C. Allen, M. Ghoreyshi, and K. Badcock. Comparison of adaptive sampling methods for generation of surrogate aerodynamic models. *AIAA journal*, 51(4):797–808, 2013. *Cited in page 31*

[141] B. Malouin, J.-Y. Trépanier, and M. Gariépy. Interpolation of transonic flows using a proper orthogonal decomposition method. *International Journal of Aerospace Engineering*, 2013, 2013. *Cited in pages 22, 29, 32, 34, 42, 43, and 44*

[142] L. Margheri. *Etude de l'impact des incertitudes dans l'évaluation du risque NRBC provoqué en zone urbaine*. PhD thesis, UPMC, 2015. *Cited in page 41*

[143] L. Margheri and P. Sagaut. A hybrid anchored-anova pod/kriging method for uncertainty quantification in unsteady high-fidelity cfd simulations. *Journal of Computational Physics*, 324:137 – 173, 2016. ISSN 0021-9991. doi: http://dx.doi.org/10.1016/j.jcp.2016.07.036. *Cited in pages 36, 41, 43, and 44*

[144] A. Marrel, B. Iooss, F. Van Dorpe, and E. Volkova. An efficient methodology for modeling complex computer codes with gaussian processes. *Computational Statistics & Data Analysis*, 52(10):4731–4744, 2008. *Cited in page 81*

[145] J. D. Martin and T. W. Simpson. A study on the use of kriging models to approximate deterministic computer models. In *Proceedings of ASME 2003 Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, September 2003. *Cited in page 80*

[146] J. D. Martin and T. W. Simpson. Use of kriging models to approximate deterministic computer models. *AIAA Journal*, 43, 04 2005. doi: 10.2514/1.8650. *Cited in page 79*

[147] G. Matheron. *Les variables régionalisées et leur estimation.* ed. Masson, 1965. *Cited in page 75*

[148] M. D. McKay, R. J. Beckman, and W. J. Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21 (2):239–245, 1979. ISSN 00401706. *Cited in page 87*

[149] J. Melenchón, I. Iriondo, and L. Meler. Simultaneous and causal appearance learning and tracking. *ELCVIA Electronic Letters on Computer Vision and Image Analysis*, 5(3):44–54, 2005. *Cited in page 59*

[150] M. Mifsud. *Reduced-order modelling for high-speed aerial weapon aerodynamics.* PhD thesis, Cranfield University, 2008. *Cited in pages 40, 43, and 44*

[151] M. Mifsud, R. Zimmermann, and S. Görtz. Speeding-up the computation of high-lift aerodynamics using a residual-based reduced-order model. *CEAS Aeronautical Journal*, 6(1):3–16, 2015. *Cited in page 20*

[152] M. J. Mifsud, S. T. Shaw, and D. G. MacManus. A high-fidelity low-cost aerodynamic model using proper orthogonal decomposition. *International Journal for Numerical Methods in Fluids*, 63(4):468–494, 2010. ISSN 1097-0363. doi: 10.1002/fld.2085. *Cited in pages 41, 43, and 44*

[153] R. H. Myers, D. C. Montgomery, and C. M. Anderson-Cook. *Response surface methodology: process and product optimization using designed experiments.* John Wiley & Sons, Inc., New York, NY, USA, 2009. ISBN 0471581003. *Cited in page 62*

[154] M. Nemec, D. W. Zingg, and T. H. Pulliam. Multipoint and multi-objective aerodynamic shape optimization. *AIAA Journal*, 42(6):1057–1065, June 2004. ISSN 0001-1452. doi: 10.2514/1.10415. *Cited in page 13*

[155] B. R. Noack. From snapshots to modal expansions - bridging low residuals and pure frequencies. *Journal of Fluid Mechanics*, 802:1–4, 2016. doi: 10.1017/jfm.2016.416. *Cited in page 22*

[156] M. Oliveira, P. Lu, X. Liu, and C. Liu. Universal high order subroutine with new shock detector for shock boundary layer interaction. In *47th AIAA aerospace sciences meeting including the New Horizons forum and aerospace exposition*, page 1139. *Cited in page 105*

[157] I. C. A. Organization. ICAO long-term traffic forecasts passenger and cargo, July 2016. *Cited in pages 3 and 5*

[158] A. B. Owen. Scrambling sobol'and niederreiter–xing points. *Journal of complexity*, 14(4): 466–489, 1998. *Cited in page 91*

[159] A. Paul-Dubois-Taine and D. Amsallem. An adaptive and efficient greedy procedure for the optimal training of parametric reduced-order models. *International Journal for Numerical Methods in Engineering*, 102(5):1262–1292. doi: 10.1002/nme.4759. *Cited in page 31*

[160] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011. *Cited in page 99*

[161] B. Peherstorfer. *Model Order Reduction of Parametrized Systems with Sparse Grid Learning Techniques*. Dissertation, Department of Informatics, Technische Universität München, Oct. 2013. *Cited in page 20*

[162] B. Peherstorfer and K. Willcox. Data-driven operator inference for nonintrusive projection-based model reduction. *Computer Methods in Applied Mechanics and Engineering*, 306: 196–215, July 2016. ISSN 00457825. doi: 10.1016/j.cma.2016.03.025. *Cited in page 21*

[163] B. Peherstorfer, D. Butnaru, K. Willcox, and H.-J. Bungartz. Localized discrete empirical interpolation method. *SIAM Journal on Scientific Computing*, 36(1):A168–A192, 2014. *Cited in pages 39 and 42*

[164] B. Peherstorfer, K. Willcox, and M. Gunzburger. Survey of multifidelity methods in uncertainty propagation, inference, and optimization. *arXiv preprint arXiv:1806.10761*, 2018. *Cited in page 18*

[165] M. J. D. Powell. Algorithms for approximation. chapter Radial Basis Functions for Multivariable Interpolation: A Review, pages 143–167. Clarendon Press, New York, NY, USA, 1987. ISBN 0-19-853612-7. *Cited in page 73*

[166] A. Qamar and S. Sanghi. Steady supersonic flow-field predictions using proper orthogonal decomposition technique. *Computers & Fluids*, 38(6):1218–1231, 2009. *Cited in pages 29, 43, and 44*

[167] Y. Qiu and J. Bai. Stationary flow fields prediction of variable physical domain based on proper orthogonal decomposition and kriging surrogate model. *Chinese Journal of Aeronautics*, 28(1):44 – 56, 2015. ISSN 1000-9361. doi: https://doi.org/10.1016/j.cja.2014.12.017. *Cited in pages 22, 29, 43, and 44*

[168] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2005. ISBN 0-262-18253-X. *Cited in pages 74, 75, 78, 79, 80, 114, and 115*

[169] S. Razavi, B. A. Tolson, and D. H. Burn. Review of surrogate modeling in water resources. *Water Resources Research*, 48(7), 2012. *Cited in page 18*

[170] J. J. Reuther, A. Jameson, J. J. Alonso, M. J. Rimlinger, and D. Saunders. Constrained multipoint aerodynamic shape optimization using an adjoint formulation and parallel computers, part 1. *Journal of aircraft*, 36(1):51–60, 1999. ISSN 0021-8669. doi: 10.2514/2.2413. *Cited in pages 13 and 35*

[171] P. J. Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987. *Cited in page 110*

[172] O. Roustant, D. Ginsbourger, and Y. Deville. Dicekriging, diceoptim: Two r packages for the analysis of computer experiments by kriging-based metamodeling and optimization. *Journal of statistical software*, 51(1):1–55, 2012. *Cited in page 99*

[173] P. T. Roy, S. Ricci, R. Dupuis, R. Campet, J.-C. Jouhaud, and C. Fournier. Batman: Statistical analysis for expensive computer codes made easy. *The Journal of Open Source Software*, 3(21):493. doi: https://doi.org/10.21105/joss.00493. *Cited in page 99*

[174] P. T. Roy, L. M. Segui, J.-C. Jouhaud, and L. Gicquel. Resampling strategies to improve surrogate model-based uncertainty quantification: Application to les of ls89. *International Journal for Numerical Methods in Fluids*, 87(12):607–627, 2018. doi: 10.1002/fld.4504. *Cited in pages 32, 36, 40, 43, 44, and 99*

[175] J. Sacks, W. J. Welch, T. J. Mitchell, and H. P. Wynn. Design and analysis of computer experiments. *Statistical Science*, 4(4), November 1989. *Cited in pages 18, 31, 75, 82, and 92*

[176] A. Saltelli, M. Ratto, T. Andres, F. Campolongo, J. Cariboni, D. Gatelli, M. Saisana, and S. Tarantola. *Global Sensitivity Analysis. The Primer*. John Wiley & Sons, Ltd, 2008. *Cited in page 35*

[177] T. J. Santner, B. J. Williams, and W. I. Notz. *The design and analysis of computer experiments*. Springer Science & Business Media, 2013. *Cited in page 83*

[178] T. Simpson, P. J.D., K. P.N., and J. Allen. Metamodels for computer-based engineering design: Survey and recommendations. *Engineering with Computers*, 17, July 2001. *Cited in pages 18, 73, 81, and 86*

[179] T. W. Simpson, J. J. Korte, T. Mauer, and F. Mistree. Comparison of response surface and kriging models for multidisciplinary design optimization. *7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis & Optimization, AIAA-98-4755*, 1:381–391, 1998. *Cited in pages 18 and 81*

[180] L. Sirovich. Turbulence and the dynamics of coherent structures part i: Coherent structures. *Quarterly of applied mathematics*, XLV(3), 1987. *Cited in pages 19, 55, 57, and 60*

[181] J. P. Slotnick, A. Khodadoust, J. J. Alonso, D. L. Darmofal, W. D. Gropp, E. A. Lurie, and D. J. Mavriplis. Cfd vision 2030 study: A path to revolutionary computational aerosciences, 2014. *Cited in page 23*

[182] A. J. Smola and B. Scholkopf. A tutorial on support vector regression. *Statistics and Computing*, 14(3):199–222, aug 2004. ISSN 0960-3174. doi: 10.1023/B:STCO.0000035301.49549.88. *Cited in pages 64, 65, 66, and 67*

[183] E. Snelson and Z. Ghahramani. Sparse gaussian processes using pseudo-inputs. In *Advances in neural information processing systems*, pages 1257–1264, 2006. *Cited in page 80*

[184] K. Sommerwerk, B. Michels, K. Lindhorst, M. Haupt, and P. Horst. Application of efficient surrogate modeling to aeroelastic analyses of an aircraft wing. *Aerospace Science and Technology*, 55:314–323, 2016. *Cited in page 20*

[185] C. Tang, K. Gee, and S. Lawrence. Generation of aerodynamic data using a design of experiment and data fusion approach. In *43rd AIAA Aerospace Sciences Meeting and Exhibit*, page 1137. *Cited in page 33*

[186] D. J. Toal. On the potential of a multi-fidelity g-pod based approach for optimization and uncertainty quantification. In *ASME Turbo Expo: Turbine Technical Conference and Exposition*. American Society of Mechanical Engineers, June 2014. *Cited in pages 34, 43, and 44*

[187] D. J. J. Toal and A. J. Keane. Efficient multipoint aerodynamic design optimization via cokriging. *Journal of Aircraft*, 48(5):1685–1695, Sep 2011. ISSN 0021-8669. doi: 10.2514/1.C031342. *Cited in pages 18 and 80*

[188] B. D. Tracey, K. Duraisamy, and J. J. Alonso. A machine learning strategy to assist turbulence model development. In *53rd AIAA Aerospace Sciences Meeting*, page 1287, 2015. *Cited in page 23*

[189] H. van der Ven and F. J. Brandsma. Navier stokes computations of 3d transonic flow for a wing/fuselage configuration. Technical Report AD/AG-26 TP-127, GARTEUR, 2001. *Cited in page 146*

[190] S. van der Walt, S. Colbert, and G. Varoquaux. The numpy array: A structure for efficient numerical computation. *Computing in Science Engineering*, 13(2):22–30, March 2011. *Cited in page 99*

[191] S. Varet. *Développement de méthodes statistiques pour la prédiction d'un gabarit de signature infrarouge*. PhD thesis, Université Paul Sabatier-Toulouse III, 2010. *Cited in pages 82, 87, and 90*

[192] K. Veroy and A. T. Patera. Certified real-time solution of the parametrized steady incompressible navier-stokes equations: rigorous reduced-basis a posteriori error bounds. *International Journal for Numerical Methods in Fluids*, 47(8-9):773–788, 2005. ISSN 1097-0363. doi: 10.1002/fld.867. *Cited in page 17*

[193] M. J. Verveld, T. Kier, N. Karcher, T. Franz, M. Abu-Zurayk, M. Ripepi, and S. Görtz. Reduced order models for aerodynamic applications, loads and mdo. In *DLRK 2016*, September 2016. *Cited in pages 14, 41, 43, and 44*

[194] F. A. Viana, R. T. Haftka, and V. Steffen. Multiple surrogates: how cross-validation errors can help us to obtain the best predictor. *Structural and Multidisciplinary Optimization*, 39 (4):439–457, 2009. *Cited in page 81*

[195] F. A. Viana, T. W. Simpson, V. Balabanov, and V. Toropov. Special section on multidisciplinary design optimization: metamodeling in multidisciplinary design optimization: how far have we really come? *AIAA Journal*, 52(4):670–690, 2014. *Cited in pages 18 and 23*

[196] S. Volkwein. Proper orthogonal decomposition: Theory and reduced-order modelling. In *Lecture Notes*. University of Konstanz, 2013. *Cited in pages 55 and 56*

[197] J. Vos, A. Rizzi, D. Darracq, and E. Hirschel. Navier-Stokes solvers in european aircraft design. *Progress in Aerospace Sciences*, 38(8):601 – 697, 2002. ISSN 0376-0421. doi: https://doi.org/10.1016/S0376-0421(02)00050-7. *Cited in page 14*

[198] K. K. Vu, C. D'Ambrosio, Y. Hamadi, and L. Liberti. Surrogate-based methods for black-box optimization. *International Transactions in Operational Research*, 24(3):393–424, 2017. *Cited in pages 82 and 83*

[199] S. Walton, O. Hassan, and K. Morgan. Reduced order modelling for unsteady fluid flow using proper orthogonal decomposition and radial basis functions. *Applied Mathematical Modelling*, 37(20-21):8930–8945, 2013. *Cited in pages 29, 43, and 44*

[200] Q. Wang, J. S. Hesthaven, and D. Ray. Non-intrusive reduced order modeling of unsteady flows using artificial neural networks with application to a combustion problem. *Journal of Computational Physics*, 2018. *Cited in pages 29, 43, and 44*

[201] Y. Wang, B. Yu, Z. Cao, W. Zou, and G. Yu. A comparative study of pod interpolation and pod projection methods for fast and accurate prediction of heat transfer problems. *International Journal of Heat and Mass Transfer*, 55(17):4827 – 4836, 2012. ISSN 0017-9310. doi: https://doi.org/10.1016/j.ijheatmasstransfer.2012.04.053. *Cited in page 21*

[202] K. Washabaugh, D. Amsallem, M. Zahr, and C. Farhat. Nonlinear model reduction for cfd problems using local reduced-order bases. *42nd AIAA Fluid Dynamics Conference and Exhibit*, June 2012. doi: 10.2514/6.2012-2686. *Cited in pages 39, 42, 102, and 105*

[203] N. Wiener. Extrapolation, interpolation, and smoothing of stationary time series: with engineering applications. *MIT Press*, 1949. *Cited in page 75*

[204] D. Xiao, P. Yang, F. Fang, J. Xiang, C. C. Pain, and I. M. Navon. Non-intrusive reduced order modelling of fluid-structure interactions. *Computer Methods in Applied Mechanics and Engineering*, 303:35–54, May 2016. ISSN 0045-7825. doi: 10.1016/j.cma.2015.12.029. *Cited in pages 20, 29, 43, and 44*

[205] D. Xiao, F. Fang, C. Pain, I. Navon, and C. Heaney. Domain decomposition non-intrusive reduced order modelling of non-linear flow dynamics. *Comput Methods Appl Mech Eng., submitted*, 2017. *Cited in pages 38, 43, and 44*

[206] R. Yondo, E. Andrés, and E. Valero. A review on design of experiments and surrogate models in aircraft real-time and many-query aerodynamic analyses. *Progress in Aerospace Sciences*, 96:23 – 61, 2018. ISSN 0376-0421. doi: https://doi.org/10.1016/j.paerosci.2017.11.003. *Cited in pages 32 and 82*

[207] S. Yoon and A. Jameson. Lower-upper symmetric-gauss-seidel method for the euler and navier-stokes equations. *AIAA journal*, 26(9):1025–1026, 1988. *Cited in pages 125 and 146*

[208] S. E. Yuksel, J. N. Wilson, and P. D. Gader. Twenty years of mixture of experts. *IEEE transactions on neural networks and learning systems*, 23(8):1177–1193, 2012. *Cited in page 38*

[209] G. Zararsiz, F. Elmali, and A. Ozturk. Bagging support vector machines for leukemia classification. *International Journal of Computer Science Issues (IJCSI)*, 9(6):355, 2012. *Cited in page 69*

[210] Z. Zhan, W. Habashi, and M. Fossati. Local reduced order modeling and iterative sampling for parametric analyses of aero-icing problems. *AIAA Journal*, 53(8):2174 – 2185, August 2015. ISSN 0001-1452. doi: 10.2514/1.J053654. *Cited in pages 31, 39, 42, 43, 44, 105, and 118*

[211] Z. Zhan, W. Habashi, and M. Fossati. Real-time regional jet comprehensive aeroicing analysis via reduced-order modeling. *AIAA Journal*, 54(12):3787 – 3802, 2016. doi: 10.2514/1.J055013. *Cited in pages 31, 39, 42, 43, and 44*

[212] H. Zhang. The optimality of naïve bayes. In *In FLAIRS2004 conference*, 2004. *Cited in page 113*

[213] R. Zimmermann. Gradient-enhanced surrogate modeling based on proper orthogonal decomposition. *Journal of Computational and Applied Mathematics*, 237(1):403–418, 2013. *Cited in pages 35, 43, and 44*

[214] R. Zimmermann and S. Görtz. Non-linear reduced order models for steady aerodynamics. *Procedia Computer Science*, 1(1):165–174, 2010. *Cited in pages 19, 29, 43, and 44*

[215] R. Zimmermann and Z.-H. Han. Simplified cross-correlation estimation for multi-fidelity surrogate cokriging models. *Advances and Applications in Mathematical Sciences*, 7(2):181–202, December 2010. *Cited in page 33*

Résumé – Les simulations numériques représentent un élément central du processus de conception d'un avion complétant les tests physiques et essais en vol. Elles peuvent notamment bénéficier de méthodes innovantes, telle que l'intelligence artificielle qui se diffuse largement dans l'aviation. Simuler une mission de vol complète pour plusieurs disciplines pose d'importants problèmes à cause des coûts de calcul et des conditions d'opérations changeantes. De plus, des phénomènes complexes peuvent se produire. Par exemple, des chocs peuvent apparaître sur l'aile pour l'aérodynamique alors que le mélange entre les écoulements du moteur et de l'air extérieur impacte fortement l'aérothermie autour de la nacelle et du mât. Des modèles de substitution peuvent être utilisés pour remplacer les simulations haute-fidélité par des approximations mathématiques afin de réduire le coût de calcul et de fournir une méthode construite autour des données de simulations.

Deux développements sont proposés dans cette thèse : des modèles de substitution utilisant l'apprentissage automatique pour approximer des calculs aérodynamiques et l'intégration de modèles de substitution classiques dans un processus aérothermique industriel. La première approche sépare les solutions en sous-ensembles selon leurs formes grâce à de l'apprentissage automatique. En outre, une méthode de reéchantillonnage complète la base d'entrainement en ajoutant de l'information dans des sous-ensembles spécifiques. Le deuxième développement se concentre sur le dimensionnement du mât moteur en remplaçant les simulations aérothermiques par des modèles de substitution. Ces deux développements sont appliqués sur des configurations avions afin de combler l'écart entre méthode académique et industrielle. On peut noter que des améliorations significatives en termes de coût et de précision ont été atteintes.

Mots clés : Modèle de substitution, Apprentissage automatique, Modèle réduit, Méthode locale, Aérodynamique, Aérothermie.

Abstract – Numerical simulations provide a key element in aircraft design process, complementing physical tests and flight tests. They could take advantage of innovative methods, such as artificial intelligence technologies spreading in aviation. Simulating the flight mission for various disciplines pose important problems due to significant computational cost coupled to varying operating conditions. Moreover, complex physical phenomena can occur. For instance, the aerodynamic field on the wing takes different shapes and can encounter shocks, while aerothermal simulations around nacelle and pylon are sensitive to the interaction between engine flows and external flows. Surrogate models are used to substitute expensive high-fidelity simulations by mathematical approximations in order to reduce overall computation cost and to provide a data-driven approach.

In this thesis, we propose two developments: (i) machine learning-based surrogate models capable of approximating aerodynamic experiments and (ii) integrating more classical surrogate models into industrial aerothermal process. The first approach mitigates aerodynamic issues by separating solutions with very different shapes into several subsets using machine learning algorithms. Moreover, a resampling technique takes advantage of the subdomain decomposition by adding extra information in relevant regions. The second development focuses on pylon sizing by building surrogate models substituting aerothermal simulations. The two approaches are applied to aircraft configurations in order to bridge the gap between academic methods and real-world applications. Significant improvements are highlighted in terms of accuracy and cost gains.

Keywords: Surrogate model, Machine Learning, Reduced-order model, Local method, Aerodynamics, Aerothermal analysis.