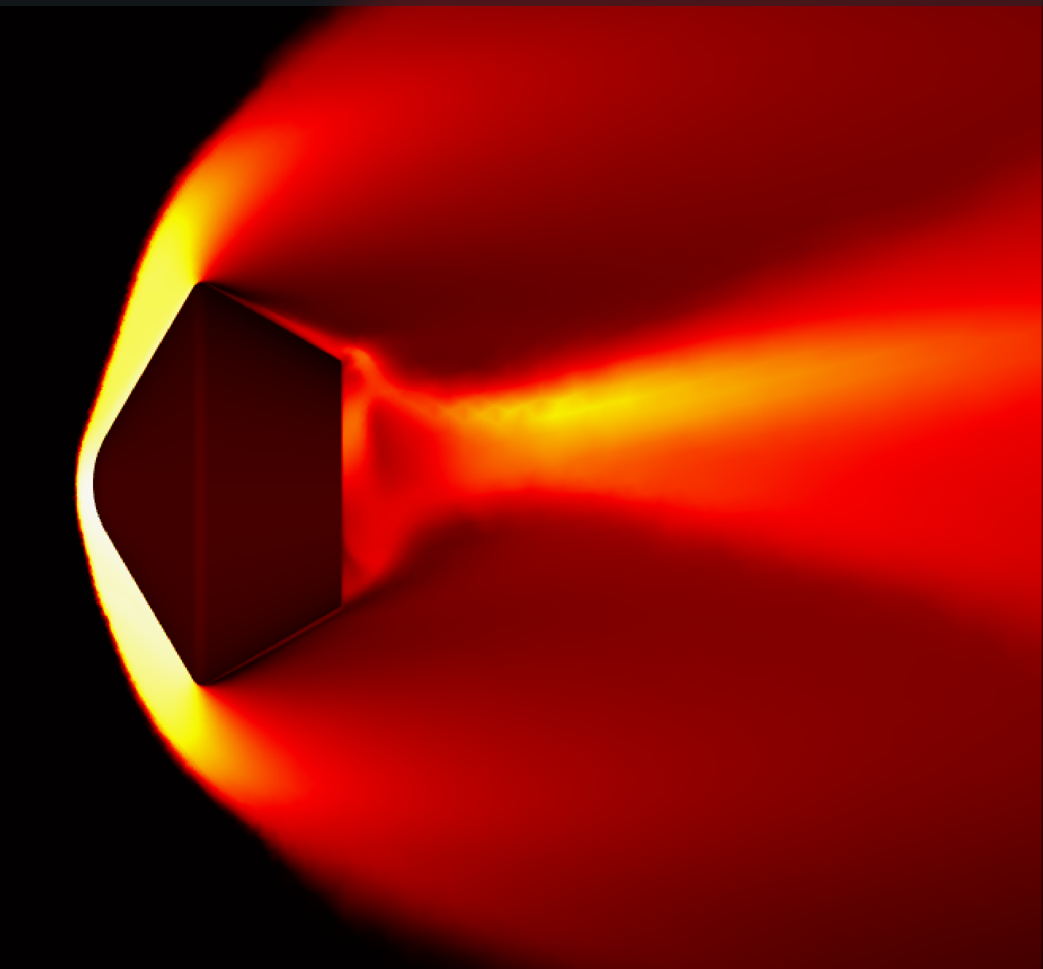
Gaussian Process regression for the prediction of aerodynamic performance

A study of multi-output surrogate modeling with optimal sampling for the development of hypersonic vehicles

Sebastian Fernandez Ruiz de las Cuevas



Gaussian Process regression for the prediction of aerodynamic performance

Thesis report

by

Sebastian Fernandez Ruiz de las Cuevas

to obtain the degree of Master of Science
at the Delft University of Technology
to be defended publicly on August 27, 2025 at 10:00

Thesis committee:

Chair:	Dr.ir. Alexander Herman van Zuijlen
Supervisors:	Dr. Mariasole Laureti
	Dr. Anh Khoa Doan
External examiner:	Dr. Botchu Jyoti
Place:	Faculty of Aerospace Engineering, Delft
Project Duration:	December, 2024 - August, 2025
Student number:	5947626

Faculty of Aerospace Engineering · Delft University of Technology

Delft
University of
Technology

Preface

This work was only possible thanks to the support from DLR Göttingen. I would like to thank the spacecraft department for their continuous support and being always available and welcoming. I especially thank Dr. Laureti and Dr. Doan for supervising this work and always providing valuable feedback. Also, thanks to Dr. Horchler and Dr. Ecker for hosting me at the DLR for both my internship and Masters thesis. And finally, I want to thank my parents for their constant support.

Contents

Nomenclature

List of Abbreviations

ANN	Artificial Neural Network
AoA	Angle of Attack
CA	Committee A
CB	Committee B
CNN	Computational Fluid Dynamics
CNN	Convolutional Neural Network
DLR	German Aerospace Center (Deutsches Zentrum für Luft- und Raumfahrt)
DTR	Decision Tree Regressor
ESA	European Space Agency
GBR	Gradient Boosting Regression
GU	Greedy Uncertainty
KL	Kullback-Leibler
LHC	Latin Hypercube
MAE	Mean Average Error
MC	Monte Carlo
MOGP	Multi Ouptput Gaussian Process
MSE	Mean Squared Error
RANS	Reynolds-Averaged Navier–Stokes
RBF	Radial Basis Function
RF	Random Forest
RLV	Reusable Launch Vehicles
RMSE	Root Mean Square Error
ROM	Reduced-Order Model
RQ	Rational Quadratic
SBO	Surrogate Based Optimization
SSTO	Single Stage To Orbit
SVM	Support Vector Machine
TPS	Thermal Protection System
VTOL	Vertical Take-off and Landing

List of Symbols

α	Angle of Attack
α_S	Stoichiometric coefficient of reactants
β	Ballistic Coefficient
β_S	Stoichiometric coefficient of products
β_{bf}	Body flap deflection angle
λ	Mean free path
μ	Mean value
ν	Smoothness parameter
ω_S	Chemical source term
ρ	Density
Σ	Covariance Matrix
σ	Standard deviation
$\underline{\underline{F}}^{Eu}$	Inviscid flux matrix in the Navier-Stoker equations
$\underline{\underline{F}}^{NS}$	Viscous flux matrix in the Navier-Stoker equations
$\underline{\underline{I}}$	Identiy matrix
$\underline{\underline{P}}$	Viscous stress tensor
$\underline{n}$	Normal vector
$\underline{Q}$	Source term
A	Area
a_r, b_r, c_r	Coefficients in the Arrhenius equation for the reaction rate
C_A	Axial force coefficient
C_D	Drag coefficient
C_L	Lift coefficient

C_m	Moment coefficient
C_N	Normal force coefficient
C_p	Pressure coefficient
D	Drag
Da	Damköhler number
E	Specific total energy
H	Height
K^{eq}	Equilibrium constant
k_r	Reaction rate
Kn	Knudsen number
L	Lift
l	Lenght scale
M	Mach number
M_S	Molar mass of species S
N	Number of training points
n_s	Molar concentration of species S
P	Number of outputs
Sc	Schmidt number
T	Temperature
t	Time
u	Velocity
v	Velocity
X_S	Molar fraction of species S

List of Figures

List of Tables

Part I

Introduction

1
Introduction and literature review

Reusable space systems are designed to recover the main components of the vehicle itself after a mission, therefore making space travel more sustainable and cost-efficient. With an ever-growing space industry looking to capitalize on new markets, such as satellite constellations and space tourism, reusable technologies have undergone substantial development in recent years. The most prominent example is Falcon 9, SpaceX's propulsive landing test vehicle. It has demonstrated the technological and economic maturity of reusable space systems with more than 419 successful launches. Through the European Space Access Strategy, the European Space Agency (ESA) has also made it clear that developing reusable vehicles is a priority for security and defense reasons. Although they have not successfully tested re-usability, there is a series of projects that aim to develop it, most notably Prometheus, Callisto, Themis, and Icarus, with Arian Next being the final goal of a reusable ”cost-efficient European rocket” [1].

Besides retro-propulsion systems, there exist other typologies of reusable space systems, most notably those that use atmospheric drag as the main braking mechanism. A prime example of this is NASA's Space Shuttle, which can be classified as a glider. Other types of vehicles include capsules, such as the Apollo command module, and lifting bodies, such as the X-38. Although different vehicles undergo different physical phenomena, they share some common scenarios, most notably the high heat loads and g-loads during entry. Due to the high velocity and rarefied flow present, complex gasdynamic and chemical phenomena need to be studied. In order to characterize this complex flow field, a series of experimental and numerical methodologies exist; however, they both come with their own set of drawbacks. Both methodologies take significant resources to operate and are time-intensive, which represents a serious bottleneck in the development pipeline of these vehicles. This work explores the use of surrogate models to quickly predict aerodynamic quantities, reducing the development time of reusable vehicles.

Surrogate models approximate complex relationships based on available data points. They have been extensively used in many areas such as finance [2], medicine [3] [4], and agriculture [5] [6]. Moreover, there are many instances of applications in the aerospace field that show promising results [7] [8]. The purpose of this work is to explore how these methods can aid the design of re-entry vehicles. This first part of the work is organized into an overview of vehicle architectures in section 1.1 and the physical phenomena present in their missions in section 1.2, an overview of existing experimental and numerical design methodologies in section 1.3, an explanation of surrogate models and their current state of the art in the aerospace field in section 1.4 and finally the research questions are established on part 1.5. Then, the theoretical background and methodology followed are presented in part II, the discussion of the results in part III, and the conclusions in part IV.

1.1. Re-usable space transportation systems

In the early years of rocket propulsion, technology was too heavy to pose the question of re-usability. Furthermore, the first rockets, such as the V-2 [9], were meant to deliver weapons, so their recovery was never a concern. However, after the war, research focus changed, and reusable vehicles started to be conceptually developed. Early examples include the Von Braun Ferry Rocket [10] and the General Dynamics Nexus [11]; however, like many other early projects, they were never built. Nonetheless, they helped develop the technology in this field. The first successful reusable launcher was the Space Shuttle [12]. Developed by NASA in the 1960s, this vehicle was composed of three main parts: the orbiter (also called space plane), two recoverable solid boosters, and an expendable fuel tank. During launch, the solid boosters would aid the orbiter to reach the desired altitude, and once they had consumed all the fuel, the orbiter would detach. The first flight was in 1981 with the Columbia orbiter, whereas the final mission was carried out by Endevour in 2011. Over the course of three decades, over 100 successful launches and landings were performed by 5 different orbiters. In the 1990s, the Delta Clipper (DC-XA) [13] was also developed, a vertical takeoff, vertical landing (VTVL) single-stage rocket that had 8 successful flights. For both cases, the motivation to develop reusable technologies was similar; they wanted to build a cost-effective method to get to space, and at that time, the NASA Reusable Launch Vehicle Technology Program determined that a fully-reusable, rocket-powered single-stage-to-orbit (SSTO) launch was the best solution. Another objective was to open the 'space frontier' to new space industries, improving economic competitiveness. In order to do this, NASA collaborated with private industries with the hope of incentivizing them to further develop RLV. However, in order to achieve this goal, a series of technical challenges had to be overcome. Shortly after the first test of the DC-XA, a paper [14] was published highlighting the main technical concerns. Among them were the weight of the cryogenic tank, the thermal protection system (TPS), the composite primary structure, and the propulsion system. Although these are still concerns for modern vehicles, most of these problems have been addressed by Falcon 9, which has proven the technological maturity of VTVL vehicles. SpaceX has developed a carbon fiber oxygen tank, reducing the weight significantly, along with the Raptor engine, which is the highest thrust-to-weight ratio engine to date. Furthermore, the TPS has been improved, allowing for a reliable reentry phase. Nonetheless, Falcon 9 and recently Starship are currently the only vehicles capable of performing retro-propulsive landing. Moreover, this typology of vehicles is not intended to return humans to Earth. For this purpose, capsules or lifting bodies remain the best choice. On a global scale, reusable technology is still far from mature. Although there are several promising projects, there are important challenges that have yet to be overcome.

One of the main problems is the high temperature that the reusable vehicles experience during atmospheric re-entry. For this typology of vehicles, the main braking mechanism is atmospheric drag, which causes the vehicle's kinetic energy to be dissipated into the surrounding air. The high velocities cause specific total energies of around 10 MJ/Kg for orbital velocities [15]. As a result, the shock layer around the vehicle can reach extreme temperatures. These conditions limit the vehicle's design parameters significantly, as not every material can handle these operating conditions. Classic re-entry vehicles, such as capsules, are very blunt in order to increase drag; however, they lose aerodynamic performance and control authority. The strong deceleration also causes the vehicle to experience high g-loads, which limits the design parameters further, as it needs to remain as light as possible for the launching phase, but strong enough to endure atmospheric re-entry. Generally, a steep re-entry angle results in high maximum aero-thermal loads, but less exposure time. Steep-angle trajectories are usually limited by the TPS and structural strength. On the other side, shallow re-entry angles are limited by the integral loads, since the maximum loads are lower but over a longer period. Furthermore, in order to achieve shallow angles, the vehicle needs to be able to generate sufficient lift and control capability in order to penetrate the denser atmosphere. However, if the lift is too high or the angle is too shallow, the vehicle might not be able to enter the thicker parts of the atmosphere and generate sufficient drag to slow down. This concept is known as "skipping" since it is analogous to skipping stones in water. The overall limitations can be depicted in a re-entry corridor diagram (Figure 1.1).

An important parameter to assess the re-entry nature is the ballistic coefficient β,

$$\beta = \frac{m}{c_D A}$$

where m is the vehicle mass, c_D is the drag coefficient, and A is the reference area used to calculate c_D. A lower β means a lower vehicle mass or higher drag/area, which generally results in less intense heating and deceleration. Overall, there is a need to balance several, often competing requirements, mainly: heating, deceleration, and landing precision and accuracy. To design a vehicle capable of performing a successful re-entry, it is absolutely imperative that all the present physical phenomena are understood. This makes it possible to quantify the aerothermal loads, through quantities such as lift coefficient, drag coefficient, and heating fluxes, among many others, that serve as design parameters for the TPS and structure. In the following section, the relevant physical phenomenon for re-entry vehicles is studied in order to understand the main design challenges. First, the most relevant gasdynamic challenges are explored in section 1.2 and the main design methodologies in section 1.3.

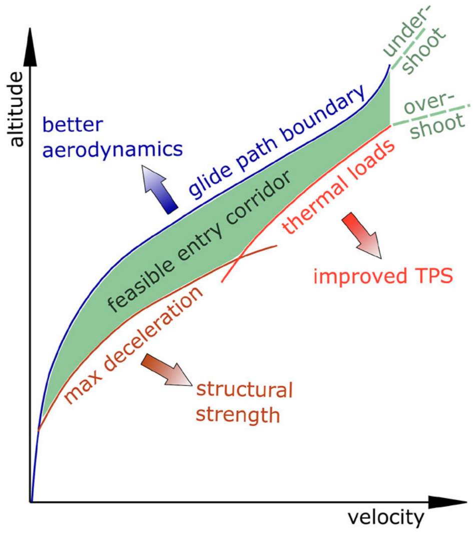


Figure 1.1: Illustration of entry corridor [15]

1.2. Gasdynamic challenges for re-entry vehicles

The typical flow field and physical phenomena encountered change for every vehicle architecture. Although some share similarities, this chapter goes into detail for each of them. A typical flow field around a capsule is shown in Fig. 1.2. One of the main elements observed is the bow shock, which dissipates most of the vehicle's kinetic energy ($95\% - 99\%$) into the surrounding air, while the remaining one is transferred as heat to the vehicle's surface. On the leeward side, a series of shock and expansion waves develops, whose exact structure depends on the angle of attack and the shape of the capsule. Furthermore, as is characteristic in hypersonic flow, there is a thin shock layer on the windward side. Moreover, there are secondary shock structures in the wake that can lead to unwanted heating.

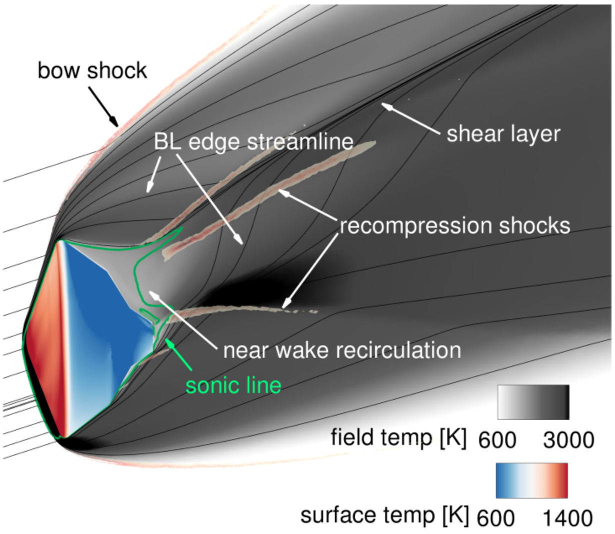


Figure 1.2: Typical flow field around re-entry capsule [15]

Gliders and lifting body configurations also experience a thin, high-temperature, shock layer, as is expected from the hypersonic regime. Moreover, there are phenomena that increase in significance with the Mach number. As the hypersonic flow is slowed down by viscous effects, the kinetic energy is dissipated into the surrounding air, causing a strong increase in the temperature. Considering a constant pressure in the wall normal direction, it follows from the ideal gas equation $\rho = p/RT$ that this results in a strong decrease in density. This can cause the boundary layer to grow, displacing viscous interactions. Furthermore, wing structures cause highly curved shock waves that generate entropy layers, which can be thicker than the boundary layer and can cover large parts of the vehicle. The phenomena can have significant effects on the flow field, most notably adverse cross-flow, earlier boundary layer separation, and altered laminar-turbulent transition points. Therefore, the flow properties close to the vehicle are not governed by viscous properties exclusively but depend greatly on the leading edge design and configuration.

Furthermore, as NASA's X-15A2 rocket-powered aircraft experiment showed, shock-shock interaction can severely damage the vehicle's structure. For example, an outer shock wave impinging on a local bow shock forms a supersonic jet downstream of the interaction, generating extreme loads. Shocks can also interact with the development of the boundary layer, creating strong local adverse pressure gradients, leading to the growth of the boundary layer and the separation of the flow zones.

A typical flow field for a retro-propulsive vehicle is shown in Figure 1.3. One of the main elements is the detached bow shock found upstream of the engine exhaust plume. This large plume forms a barrel shock and Mach disk, which shields the vehicle from the high-energy incoming free stream. At high altitude, the plumes from different engines can merge and interact with each other due to the low back pressure, which can lead to backflow of hot gases and therefore areas of high heating.

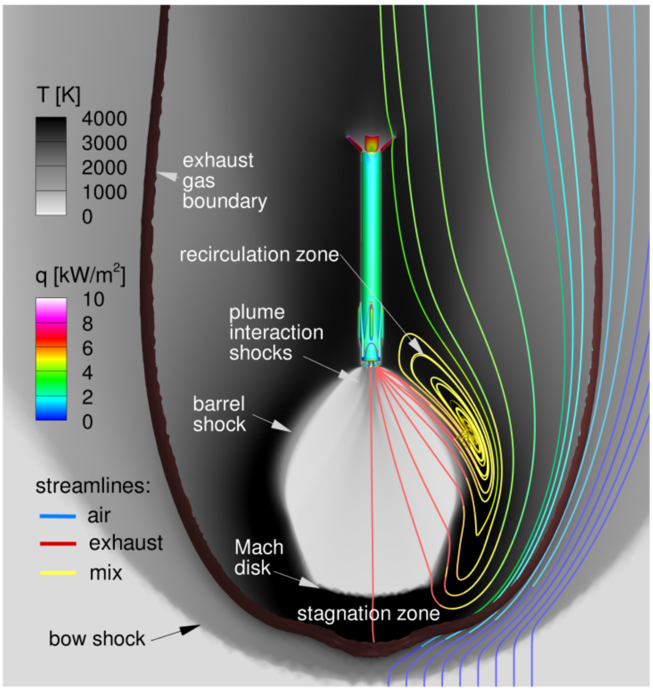


Figure 1.3: Typical flow field for retro propulsive rocket [15]

Another important phenomenon present in reentry missions is Aerothermochemistry. Vehicles entering the atmosphere are usually in the hypersonic regime, which is commonly defined as above $M = 5$. In this environment, a bow shock forms, which causes the kinetic energy of the atmospheric gas to be dissipated into thermal energy. This results in a steep increase of thermodynamic energy and collisional interactions between particles, which then results in a series of chemical and thermodynamic processes. Mainly, the kinetic energy excites the vibrational and rotational modes of molecules, which, eventually, causes inter-nuclear bonds to break and dissociation to occur. This excitement also causes collisional processes between molecules and atoms, which further excite electrons and cause the gas to radiate electromagnetic energy as electrons decay into lower energetic states.

The purpose of this section is not to perform a detailed analysis on the aerothermochemistry; for that, the following sources are provided [16] [17]. For now, it is sufficient to understand that there are several phenomena that complicate the modeling methods and make these simulations particularly resource-intensive.

1.3. Design methodologies

There are two main approaches when it comes to characterizing the flow around reentry vehicles: Experimental and numerical. Both come with a series of benefits and drawbacks. An ideal study would include both approaches in a way that they complement each other. However, it is often not practical or feasible to cover both areas, and compromises are often made. In order to better understand how to approach this problem, the following section gives a brief overview of both experimental and numerical approaches.

1.3.1. Experimental methods

Experimental tests are often performed in wind tunnels and are therefore referred to as ’ground-based’ tests. The main challenge with this approach is the ability to simulate the desired flight conditions. There is no single test facility capable of simulating all of the flight conditions a reentry vehicle encounters during a mission [18]. Therefore, specific facilities are used to study a particular phenomenon or condition. The effectiveness of simulating certain conditions can be evaluated through the use of similarity parameters, which are values that describe the flow state. For low-enthalpy flows (below 2MJ/Kg), the Mach and Reynolds numbers are the only relevant parameters. However, for high-enthalpy conditions, the Mach number becomes increasingly irrelevant due to thermochemical phenomena such as dissociation and ionization of the air that travels through strong shocks [19]. Therefore, other parameters are used to describe the flow, specifically the flow velocity and the Damkohler number, which relates the vehicle body length and the dissociation relaxation distance. There are cases in which matching all relevant criteria is not feasible; therefore, engineering judgments need to be applied to interpret the results in a correct way.

Several types of ground-based facilities achieve different flight conditions and rely on different mechanisms. A blowdown tunnel works by expanding flow coming from a high-pressure reservoir through a nozzle into a low-pressure section, which can be ambient pressure or lower if a vacuum system is in place. This provides a short test time but the ability to produce high Reynolds numbers as a result of high pressures. Another drawback is the fact that the air gets cooled down as it expands, and therefore, in order to avoid condensation, the operating temperature is often limited. This also means that Mach number similarity is achieved with a lower velocity flow, which means lower enthalpy and stagnation temperatures relative to flight conditions. This can be overcome by introducing an arc heater, but this introduces problems such as electric erosion. In order to achieve higher enthalpy flow, shock or expansion tunnels can be used. This facility typology uses shock waves in order to generate a region of high temperature and high pressure, without sacrificing the flow velocity [15].

Overall, the primary limitation of experimental methods is their inability to replicate the full flight envelope, ranging from low to high Mach numbers, low to high enthalpy conditions, and varying flow regimes. As a result, multiple specialized tests are typically required, and engineering judgment is necessary to interpret and correlate the results. Additionally, issues such as geometric scaling, insufficient flow enthalpy, and short test durations constrain the extent to which ground-based facilities can accurately reproduce flight conditions. These limitations are particularly critical in the design of hypersonic vehicles, where many complex physical phenomena—such as strong shock interactions, chemical non-equilibrium, and turbulent transition—occur simultaneously.

1.3.2. Numerical methods

The other main approach to studying reentry vehicles is numerical methods, mainly Computational Fluid Dynamics (CFD). To apply this method, a consistent discretization scheme is used to transfer the governing continuous partial differential equations (PDEs) to a discrete system of algebraic equations. Consistency means that the discretization process can be reversed through a Taylor series expansion. In other words, as the discretization becomes infinitely small, the discrete equations should approximate the continuous PDEs. Then, a solver is used to approximate the solution of this system. This solver should be stable, in other words, provide lower numerical errors as the solution process goes forward. Numerical methods make it possible to study conditions impossible to replicate in ground-based testing; however, they come with their own limitations, as this section will address.

There is a series of governing equations that can describe the flow field depending on the operating conditions. The main parameter for this choice is the Knudsen number, which relates the mean free path of the gas λ to the characteristic length scale of the problem, L.

$$Kn = \frac{\lambda}{L}$$

The Boltzmann equation is a nonlinear integro-differential equation for the 6-dimensional probability density function of gas particles and momentum and is theoretically valid for any Knudsen number. However, since it models the evolution of the flow field through particle advection and binary collisions, they are usually employed in low-density, rarefied conditions. Other methods, such as the Bhatnagar-Gross-Krook (BGK) or Fokker-Planck, provide a more efficient solution for high-density flows. However, the most important equations for aerothermal analysis of space vehicles are the Navier-Stokes equations, valid for Knudsen numbers below an order of magnitude of 10^2.

Despite major advancements in computational power, there are still crucial challenges with CFD, such as the quantification of physical modeling errors and discretization errors. Furthermore, the added complexity of solving hypersonic flows introduces many limitations. For example, the high-temperature gradients require grid spacing in the order of 10^{-6}m. Furthermore, strong shocks can produce large errors if not properly aligned with the grid, for example, the carbuncle phenomenon. Moreover, if the thermodynamic and chemical relaxation rates are of a similar order of magnitude as fluid time scales, additional transport equations and chemical non-equilibrium models need to be introduced.

Overall, numerical methods face important limitations when applied to the complex aerothermal environment of atmospheric entry. These include uncertainties in physical models, namely for turbulence and non-equilibrium thermochemistry, which often lack experimental validation. Furthermore, high-fidelity CFD simulations are computationally expensive, making full trajectory and multi-parameter studies impractical. Lastly, no single numerical model can accurately describe all flow regimes. Therefore, similar to the experimental methods, engineering judgment is necessary to interpret the results.

1.3.3. Aerodynamic data bases

As explained in Section 1.2, reentry vehicles go through a series of different operating conditions characterized by different Mach numbers, atmospheric densities, and physical phenomena. Understanding how a vehicle behaves under these conditions is an integral part of the design phase of its various subsystems. A common practice is to run a series of numerical simulations and experimental tests to cover the various vehicle configurations and collect all this data in a single place, known as a database. This allows for a complete understanding of the vehicle's mission, which has an impact on other areas such as the structural, thermal, and guidance systems. An example is presented by Bibb K et al (2011) [20] where an aerodynamic database was created for the Orion Crew Module. The database provides force and moment coefficients for a given velocity, altitude, and vehicle configuration. The data used was mainly obtained through CFD tools, with over 1000 viscous, reacting gas chemistry simulations covering a range of Mach numbers from 2 to 37. The final database was used to design the vehicle trajectories, assess flight performance, and predict the landing ellipse during flight operations.

Another is the database presented by Marwege A, et al (2024) [21] consisting of experimental data for the landing burn of RETALT in the Vertical Free-Jet Facility Cologne at DLR in Cologne. The main objective of this work was to analyze the subsonic retropropulsion flow fields. The main focus was on characterizing the unsteady flow field through a series of high-speed schlieren videos and power spectral density.

Ecker et al (2024) [22] presents an aerothermal database for the descent phase of the CALLISTO vehicle. A series of CFD simulations was conducted to study the aerothermal loads present during the landing approach corridor for different configurations of the vehicle's aerodynamic surfaces. The purpose of this work is to gain a deeper understanding of the thermal loads to design a proper thermal protection system (TPS). In order to describe the landing phase, a large parameter space was needed (i.e., Mach number, altitude, and angle of attack). For this reason, the database mainly consists of 2D computations, but higher-fidelity 3D simulations were made on trajectory points where important phenomena take place, especially those involving 3D flow effects.

The creation of aerodynamic databases involves the use of both experimental and numerical data. As explained in the previous sections, these are time-intensive methods that represent the main bottleneck of the design process. Therefore, efforts have been made to use statistical and probabilistic approaches to generate these databases. The work in [23] compares the performance of most state-of-the-art machine learning models for the creation of an aerodynamic database for the Orion capsule [20]. They establish that these methods indeed provide an overall reduction in the time and expertise needed. [24] focuses on the use of multi-fidelity methods to fuse data from different sources. [25] uses structured covariance neural networks to perform a similar task. These implementations show that surrogate models are an effective and efficient way to generate aerodynamic databases, which allows for broader and more useful conclusions compared to traditional methods [26]. In the next section, surrogate models will be explained in more detail, and further applications will be presented.

1.4. Surrogate models

Surrogate models are approximations of more complex or higher-order relations. They map input data to an output without knowing the actual relation. This type of model, like most machine learning architectures, aims to generalize from a finite set of observed data. Surrogate models inherently have a lower accuracy when compared to the real relation since they are approximations based on a finite set of data points; however, they come with the advantage of reduced complexity and computational loads. The trade-off between accuracy and computational time depends on the goals of each individual application. These models have been widely used in a myriad of fields such as agriculture, finance, medicine, etc.

Often, the main task of a surrogate model is to perform regression, that is, to fit a function that describes the trend of the data. The simplest example is linear regression (e.g, least squares), where a straight line is used to approximate the target function [27]. This method, however, is often inadequate to describe the nonlinearities present in most distributions. Nonlinear methods can be used to fit more complex distributions, including polynomial regression, splines, radial basis functions, and many more. The following section provides an overview of these methods and how they are currently used in the aerospace field.

1.4.1. Applications in aerospace

Surrogate models have proven to be a useful tool in the aerospace field as they can replace expensive computations, mainly fluid flow analysis (i.e CFD) and optimization methods. A large field of this research focuses on aerodynamic shape optimization. Surrogate-Based Optimization (SBO) tries to reduce the number of evaluations of high-fidelity models, replacing them with either low-fidelity models or approximations of the high-fidelity models. The optimization of wings has been a popular problem to solve using these methods. This research has been active since the 80s. However, only thanks to the increased computational resources of recent years have these methods started to provide a viable solution. Some studies focus on 2D airfoils and use a Support Vector Machine as a surrogate model. While [28] uses a training set of 4000 points, [7] only uses 100, and both present methods that accurately predict the aerodynamic coefficients of the airfoil. However, these studies are limited to simple 2D geometries in subsonic and transonic configurations. The work in [29] compares 5 different machine learning algorithms for the prediction of the lift-to-drag ratio. The algorithms used are Random Forest (RF), Gradient Boosting Regression (GBR), Decision Tree Regressor (DTR), AdaBoost, and Linear Regression (LR). The study used the Coefficient of Determination R^2 and the MSE as indicators of performance. It was found that for different training set sizes, different algorithms performed the best. A large number of studies use kriging or Gaussian Process Regression for shape optimization in a wider range of test cases, including subsonic [30][31][32] and hypersonic 3D geometries optimization [33]. This method uses significantly fewer data points for the training process than the previously presented methods, as well as classic neural network applications [34].

Another set of studies focuses entirely on predicting the aerodynamic coefficients of more complex geometries. Yuxin Y et al. [35] explore the use of surrogate modeling for predicting a hypersonic lift body aerodynamic performance. The study compares a Gaussian process regression, polynomial regressions, and support vector regression to find that they perform similarly with large training sets, but with a training set size below 40% of the dataset, the GPR outperforms the rest. [26] uses a different configuration of a GPR to successfully predict various aerothermal quantities of a reentry vehicle. Other works implement the use of Artificial Neural Network (ANN) architectures to make these predictions. Convolutional Neural Networks have been widely used to predict flow field quantities from images. However, these are data-intensive algorithms that perform poorly on small datasets [36] [37] [38].

Another area of research focuses on predicting the flow field rather than the global performance metrics.

Most of these studies use a reduced order model (ROM) to simplify the complexity of the problem. Francés-Belda V. et al (2024) [39] present a new method using ß-variational autoencoder (ß-VAE) to reduce the dimension of the aerodynamic data to capture the most relevant features with fewer dimensions and then uses Gaussian process regression to predict the pressure distribution over a 3D wing. The performance indicators used include MAE, RMSE, and R2 score. This work is able to predict nonlinearities in the pressure field due to shock waves in a transonic regime. Similar studies include [40] and [35], which uses the proper orthogonal decomposition method as a ROM. Since the purpose of this work is to study the macro aerodynamic performance of a reentry vehicle (i.e, Cl, Cd, and Cm), ROMs will not be studied.

From the literature studied, it is clear that when it comes to small training set sizes, the GPR outperforms most methods. This result derives from the fact that you can inform a GP not only with the observations but also the prior distribution through the kernel function. Data efficiency is especially important in hypersonic aerodynamics, where simulations and experiments are particularly expensive. In fact, most works aiming to make predictions for hypersonic vehicles use a GPR. Moreover, as [26][41] [42] show, implementing a multi-output architecture can further reduce the amount of training points needed to make accurate predictions. Therefore, this is the method that will be explored in this work.

1.5. Research Formulation

Following the previous literature review, the following research questions and research objective can be derived.

Research Objective

To develop and assess the accuracy of a Gaussian process regression model that can predict the aerodynamic performance of reentry vehicles, namely capsules and gliders, by performing 3D CFD simulations using the DLR TAU code.

Research Question 1

How accurately can a Gaussian Process regression predict the aerodynamic performance of a reentry vehicle?

Research Question 2

How do the results change for different vehicle architectures? How much does the accuracy change? Do the covariance function and hyper-parameter need to be modified?

Research Question 3

To what degree can the number of simulations be reduced in order to build a representative aerodynamic database?

Part II

Methodology

In this part, the theoretical background for the Gaussian Process is described in Chapter 2. Namely, the mathematical framework is established for the single- and multi-output architectures, along with details of implementation. Then, the theory behind the TAU numerical solver is described in Chapter 3. Here, the numerical and physical assumptions are established alongside details regarding the setup used to run the simulations for the capsule and then for the glider.

2
Gaussian Process Regression

2.1. Theoretical background

Gaussian Process (GP) models are a type of surrogate model that has been widely used in machine learning applications. Their main use is non-linear regression, in other words, modeling observed data by a function that is a non-linear combination of the model parameters and depends on one or more independent variables. When coupled with a Bayesian framework, it can provide a powerful statistical model that can not only make predictions but also estimate their uncertainty, unlike other methods such as artificial neural networks (ANN). Another feature of this model is that it is non-parametric, which offers more flexibility when working with highly dimensional problems, as there is no risk of over-fitting. A GP is a generalization of multivariate Gaussian distributions; therefore, in order to understand it, we must first address the mathematical foundations on which it is built.

Multivariate Gaussian distribution is a generalization of univariate Gaussian distributions to higher-dimensional spaces. A vector $X = [X_1, ..., X_n]$ is said to have a multivariate Gaussian distribution if every linear combination of its components has a univariate normal distribution. Furthermore, its probability density function is given by:

$$p(x; \mu, \Sigma) = \frac{1}{(2\pi)^{n/2} \mid \Sigma \mid^{1/2}} \text{exp}(-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu))$$

Which can simply be written as $X \sim \mathcal{N}(\mu, \Sigma)$, where μ is the mean and Σ is the covariance matrix. The diagonal of Σ indicates the variance σ_i^2 of the i-th random variable. The off-diagonal elements σ_{ij} indicate the correlation between the i-th and the j-th random variable, which can assume values in the range $[-1, 1]$, where ± 1 indicates the strongest possible correlation and 0 indicates no correlation. To illustrate the effect of the covariance function, two different 2-dimensional examples with $\mu = 0$ are shown in Figure 2.1.

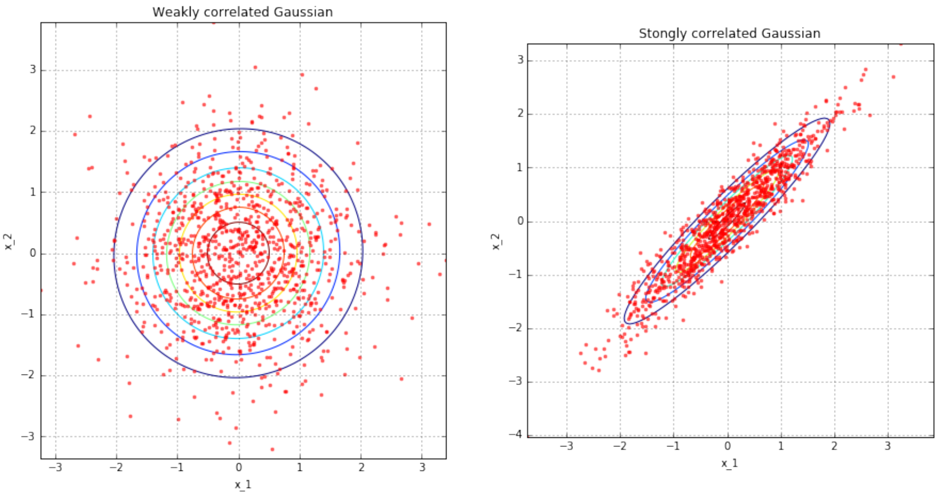


Figure 2.1: Two 2-dimensional Gaussian distributions. One the left a weakly correlated distribution with covariance matrix $\Sigma = \left[\begin{smallmatrix} 1 & 0.95 \\ 0.95 & 1 \end{smallmatrix}\right]$ and one the right a strongly correlated one with $\Sigma = \left[\begin{smallmatrix} 1 & 0.02 \\ 0.02 & 1 \end{smallmatrix}\right]$ [43]

The distribution on the left has a covariance matrix $\Sigma = \left[\begin{smallmatrix} 1 & 0.02 \\ 0.02 & 1 \end{smallmatrix}\right]$, meaning that x_1 and x_2 have a correlation of 0.02, which implies a weak correlation. On the other hand, the covariance matrix $\Sigma = \left[\begin{smallmatrix} 1 & 0.95 \\ 0.95 & 1 \end{smallmatrix}\right]$, implying that x_1 and x_2 have a strong corelation of 0.95.

In order to extend this concept to higher dimensions, we introduce another representation of the problem where the y-axis represents the value of a particular variable and the x-axis indicates the index of this variable, as shown in Figure 2.2 for the example of the 2-dimensional case. In this view, sampling from a highly correlated distribution will yield points that are close together, as seen in figure 2.3. This change of view is called the ”kernel trick”, and it effectively introduces a feature space where correlation between variables is translated as similar output values.

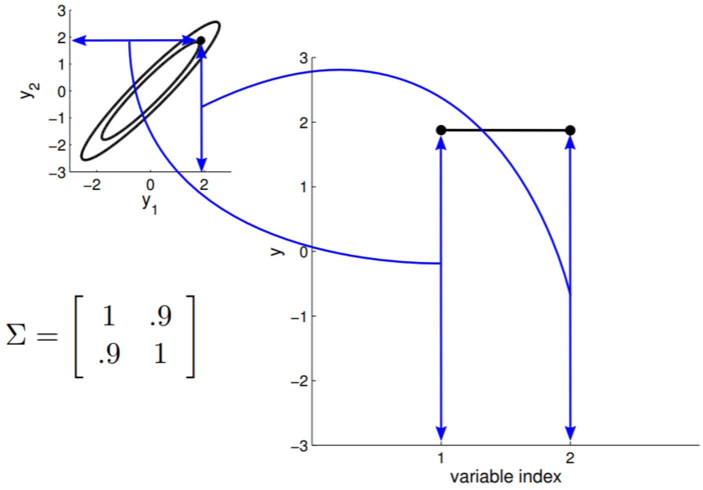


Figure 2.2: Illustration of the change of view introduced by the ”kernel trick” [44]

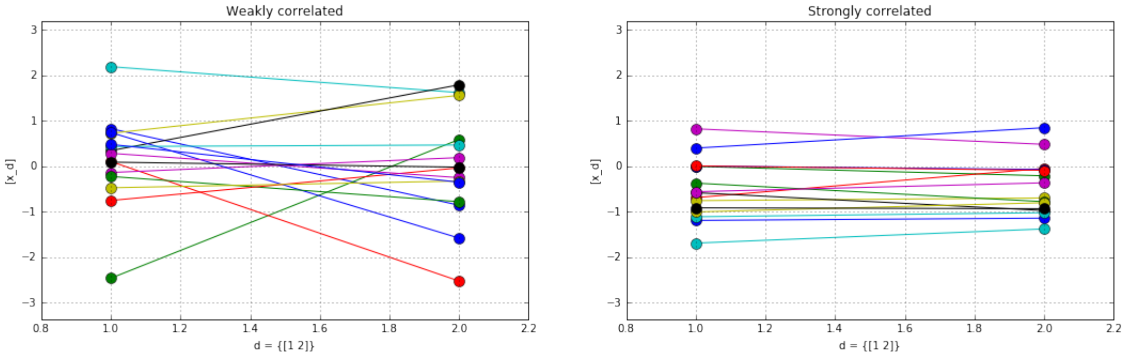


Figure 2.3: Weak and strong correlation in the feature space introduced by the ”kernel trick” [43]

We can then introduce more variables, each less correlated to the first one, as shown in Figure 2.4 (left) for 8 dimensions, where the correlation coefficients decrease progressively. Furthermore, the covariance between two consecutive points is always high. This means that if we sample from this distribution, we will obtain points that have a similar value if they are close together but very different values if they are far away. This process can naturally be extended to infinitely many dimensions as shown in Figure 2.4 (right).

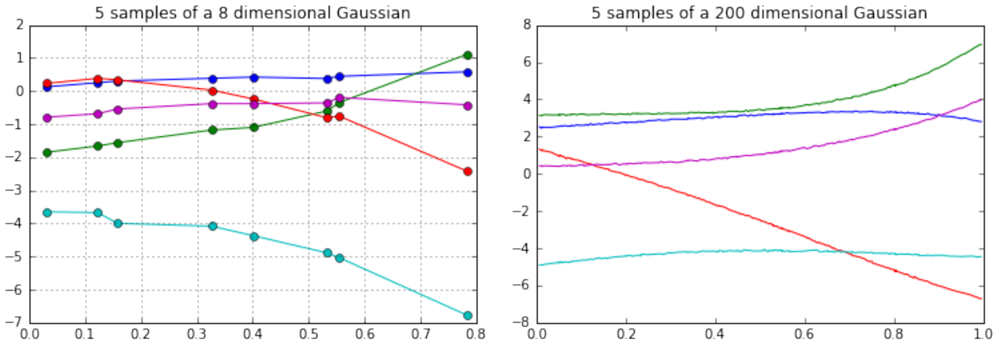


Figure 2.4: Arbitrary extension of the dimensions of a Gaussian distribution [43]

The process previously described alludes to why a GP can be described as an infinite-dimensional multivariate Gaussian distribution, from which functions can be sampled. It is worth noting that a GP is often called non-parametric, which is the same as having infinite parameters, as previously established. Instead of trying to find the right parameters to describe some data distribution, a GP uses a covariance function, also called a Kernel, and observations to fit some target distribution. In order to fully understand this method, a more mathematical definition is now given. Formally, a GP is defined as a collection of random variables, any finite number of which have a consistent Gaussian distribution. In the same way, a Gaussian distribution is defined by its mean and variance, a GP is specified by its mean function $m(x)$, and covariance function $k(x, x')$. In this way, we can write a GP as:

$$f(x) \sim \mathcal{GP}(m(x), k(x, x'))$$

This definition implies a series of properties that follow from its Gaussian nature. These properties later become fundamental when explaining how a GP can be used to make predictions and generalize data. Assuming that a random variable follows a multivariate Gaussian distribution, it is possible to partition its dimension into two sets: A, B. Then, the joint distribution can be written as:

$$p\,(\underbrace{f_1, f_2, ...,}_{f_A} \underbrace{f_{s+1}, f_{s+2}, ..., f_N}_{f_B}) \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{K})$$

Where,

$$\boldsymbol{\mu} = \begin{bmatrix} \boldsymbol{\mu}_A \\ \boldsymbol{\mu}_B \end{bmatrix}, \boldsymbol{K} = \begin{bmatrix} \boldsymbol{K}_{AA} & \boldsymbol{K}_{AB} \\ \boldsymbol{K}_{BA} & \boldsymbol{K}_{BB} \end{bmatrix}$$

Moreover, the marginal distribution can be written as:

$$p(\boldsymbol{f}_A, \boldsymbol{f}_B) \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{K})$$

$$p(\boldsymbol{f}_A) = \int_{\boldsymbol{f}_B} p(\boldsymbol{f}_A, \boldsymbol{f}_B) d\boldsymbol{f}_B = \mathcal{N}(\boldsymbol{\mu}_A, \boldsymbol{K}_{AA})$$

This important property means that even if a GP has infinitely many dimensions, we only need to represent a finite number of them, making it computationally possible to perform this operation. Therefore, given the available training data $\boldsymbol{f}$ and any (potentially infinite) number of test data $\boldsymbol{f}_*$ that follow a (potentially infinite) Gaussian distribution, $p(\boldsymbol{f}, \boldsymbol{f}_*) \sim \mathcal{N}(\boldsymbol{\mu}_\infty, \boldsymbol{K}_\infty)$. The marginalization property allows us to deal only with a finite mean and covariance function:

$$\boldsymbol{\mu}_\infty = \begin{bmatrix} \boldsymbol{\mu}_X \\ ... \end{bmatrix}, \boldsymbol{K}_\infty = \begin{bmatrix} \boldsymbol{K}_X & ... \\ ... & ... \end{bmatrix} \implies p(\boldsymbol{f}) = \mathcal{N}(\boldsymbol{\mu}_X, \boldsymbol{K}_X)$$

Where $\boldsymbol{X}$ is the training inputs and $\boldsymbol{K}_{XX}$ is the covariance matrix obtained by evaluating the covariance function at all given inputs. Therefore, assuming a mean of zero for simplicity, the GP will have the following joint Gaussian distribution:

$$\begin{bmatrix} \boldsymbol{f} \\ \boldsymbol{f}_* \end{bmatrix} \sim \mathcal{N}(0, \begin{bmatrix} \boldsymbol{K} & \boldsymbol{K}_* \\ \boldsymbol{K}_*^\top & \boldsymbol{K}_{*,*} \end{bmatrix})$$

Where $\boldsymbol{K}_*$ is the cross-variance matrix obtained by evaluating the function in pairs of training inputs $\boldsymbol{X}$ and test inputs $\boldsymbol{X}_*$, i.e

$$\boldsymbol{K}_* = k(\boldsymbol{X}, \boldsymbol{X}_*), \quad \boldsymbol{K}_{*,*} = k(\boldsymbol{X}_*, \boldsymbol{X}_*)$$

Where $k(.,.)$ is any given covariance function. Finally, in order to use a GP to make predictions, we make use of the conditional property, which follows directly from the Gaussian distribution properties. Assuming a training set $\boldsymbol{X} = [x_1, x_2, ..., x_N]$ and a GP prior $\boldsymbol{f} = [\boldsymbol{f}_1, .., \boldsymbol{f}_N]$ we can use it to condition the function outputs of a GP in order to perform an inference for the function value at any input location x_*. We do this using Bayes' theorem of conditional probability to get:

$$p(\boldsymbol{f}_* | \boldsymbol{f}_1, .., \boldsymbol{f}_N) = p(f(x_*)|f(x_1), ..., f(x_N)) \sim \mathcal{N}(\boldsymbol{K}_*^\top \boldsymbol{K}^{-1} \boldsymbol{f}, \boldsymbol{K}_{*,*} - \boldsymbol{K}_*^\top \boldsymbol{K}^{\boldsymbol{-1}} \boldsymbol{K}_{\boldsymbol{*}})$$

Where $\boldsymbol{\mu}_p = \boldsymbol{K}_*^\top \boldsymbol{K}^{-1} \boldsymbol{f}$ is called the predictive mean and $\boldsymbol{K}_p = \boldsymbol{K}_{*,*} - \boldsymbol{K}_*^\top \boldsymbol{K}^{\boldsymbol{-1}} \boldsymbol{K}_*$ is the predictive covariance. In this way, it is possible to extrapolate or interpolate new data based on the conditioning through observations and prior functions. The example shown in Figure 2.4 uses a prior defined by a covariance function that simply generates smooth functions as previously explained. This is known as the radial basis function (RBF) (also known as the squared exponential kernel) defined as:

$$k_{RBF} = \sigma^2 exp\left(-\frac{(x - x')^2}{2l^2}\right)$$

Where σ^2 is the output variance, which can be simply treated as a scaling factor. Further, the indices (x, x') simply indicate two different variables in the domain, with this particular covariance function, it can be seen that variables close together will return a value close to σ^2 and distant values will return a value close to zero. This essentially means that values close together will have a similar output value, effectively making smooth functions. Furthermore, l is the length scale which basically indicates the length of the fluctuations or the curves in the function as shown in Figure 2.5. σ and l are known as hyper-parameters, which are modified to fit the desired data.

The RBF function is classified as a universal kernel since it is able to approximate an arbitrary continuous target function on any compact subset of the input space [45]. Further, every function in its prior has

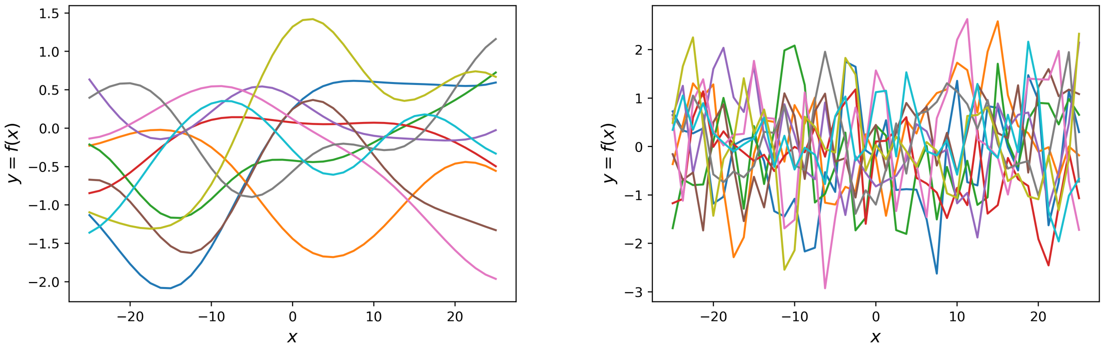


Figure 2.5: Samples from GP with RBF kernel with two different length scales, on top $l = 10$ and on the bottom $l = 1$

infinitely many derivatives and only has two parameters [46], making it a powerful and simple way to approximate most distributions. Other than the RBF, there are a series of commonly used kernels that have different properties, among them is the Rational Quadratic Kernel (RQ):

$$k_{RQ} = \sigma^2 exp\left(1 + \frac{(x - x')^2}{2\alpha l^2}\right)^{-\alpha}$$

Where α is called the scale mixture parameter, which determines the relative weight of large-scale and small-scale variations. It can be seen that if $\alpha \to \infty$, this kernel is identical to the RBF. The RQ kernel is equivalent to adding together many RBFs with different length scales. Like the RBF, this covariance function is able to approximate smooth functions, since a discontinuity or a 'kink' would imply a sudden change of the parameter l. Another kernel that stems from the RBF is the Matern kernel:

$$k_M = \frac{1}{\Gamma(\nu)2^{\nu-1}}\left(\frac{\sqrt{2\nu}}{l}(x - x')\right)^{\nu} K_\nu\left(\frac{\sqrt{2\nu}}{l}(x - x')\right)$$

Where $K_\nu(\cdot)$ is a modified Bessel function and $\Gamma(\cdot)$ is the gamma function with ν as a parameter which controls the smoothness of the resulting function, as $\nu \to \infty$ the kernel becomes equivalent to the RBF kernel. For more information about these covariance functions, see [47], Chapter 4. Moreover, there are many other kernels that describe different types of data structures, such as periodicity, linear trends, etc. These functions can be combined through multiplication and addition to describe complex, higher-dimensional distributions. Multiplying kernels is essentially applying the AND operator, where the resulting function will have a high value only if both of the components do as well. On the other hand, adding kernels can be thought of as applying the OR operator.

The choice of kernel is relevant since it determines almost all the generalization properties of a GP model. This choice depends on the specific data that is being approximated, and there exists no rigorous method for choosing them. Some works present methods for automatically choosing a kernel to best fit the provided data. For example, Zhao, S. et al (2024) [48] present an automatic kernel search algorithm along with a hyper-parameter optimization strategy. Yuxin Y, et al (2023) [35] present a beam search approach for automatic kernel construction for the prediction of the aerodynamic performance of lifting bodies.

2.2. Multi-output Gaussian Process

GPs are usually defined as having a single output; however, multiple outputs can be considered as well. Furthermore, the relation between these outputs can be modeled, which is of special interest when predicting multiple quantities that are linked by physical relations, like the drag, lift, and pitching momentum. This is often referred to as "multi-task" GP, or simply multi-output GPs (MOGPs). For a vector-valued prediction, mapping this unknown relation becomes a multi-dimensional function $\boldsymbol{f} : \mathbb{R}^D \mapsto \mathbb{R}^M$, where D is the number of inputs, and M is the number of outputs in the case of $M > 1$. Correspondingly, the GP has a mean function $\boldsymbol{\mu} : \mathbb{R}^D \mapsto \mathbb{R}^M$. Analogously, the covariance matrix becomes a block structured matrix $\boldsymbol{K} \in \mathbb{S}_+^{MN \times MN}$, where N is the number of samples. The same producer for the scalar-output framework described in section 2.1 can be used to obtain a posterior predictive mean and covariance. However, in this multi-output setting, the posterior prediction can potentially be improved if the correlation between the output target dimensions is incorporated into the structure of the GP kernel.

A popular and simple way to implement the multi-output architecture is through a method called linear model of coregionalization (LMC) [49]. In LMC a multi-output function $\boldsymbol{f}(\cdot)$ is constructed from a linear transformation $W \in \mathbb{R}^{P \times L}$ of L independent functions $g_l(\cdot)$ as:

$$\boldsymbol{f}(\boldsymbol{x}) = W\boldsymbol{g}(\boldsymbol{x})$$

Where $\boldsymbol{g}(\boldsymbol{x}) = \{g_l(\boldsymbol{x})\}_{l=1}^L$ and $g_l(\cdot) \sim \mathcal{GP}(0, k_l(\cdot, \cdot'))$ and with $\boldsymbol{f}(\boldsymbol{x}) \in \mathbb{R}^P$ and $\boldsymbol{g}(\boldsymbol{x}) \in \mathbb{R}^L$. This definition implies that the covariance becomes:

$$k(\{\boldsymbol{x}, p\}, \{\boldsymbol{x}', p'\}) = \sum_{l=1}^{L} W_{pl} k_l(\boldsymbol{x}, \boldsymbol{x}') W_{p'l}$$

.

One of the main drawbacks of GP is the computational cost, which scales with the number of training points N as $\mathcal{O}(N^3)$ over time. For MOGPs, there is the additional cost of calculating the covariance between all outputs P, which results in an $\mathcal{O}(N^3P^3)$ scaling. However, there have been developments in inducing variable approximations [50] that have significantly reduced the computational cost of MOGP.

The use of MOGP is a strategy that provides an additional level of data efficiency by leveraging the correlation between outputs. Essentially, data from one output contributes to the learning of others through the shared covariance structure, enabling more efficient use of available data and improved generalization. In the case of an aerodynamic database, the C_L, C_D, C_m are all related at a physical level as they are all governed by the same underlying flow phenomena. The usage of this method will therefore be explored to determine its efficacy in the particular context of this thesis.

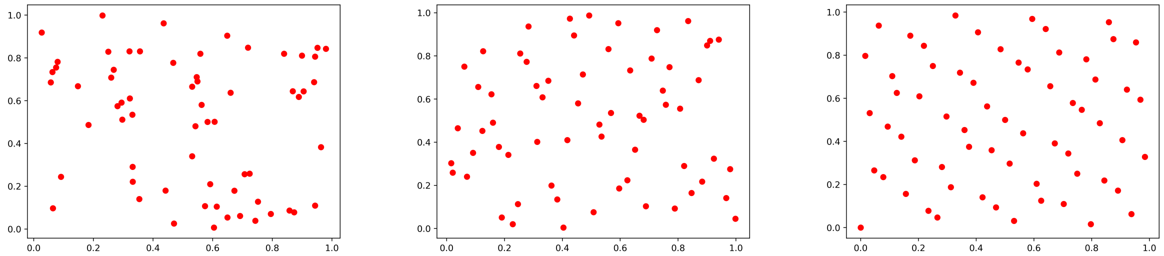
2.3. Active Sampling

The data used to train a surrogate model directly affects its accuracy, efficiency, and generalization capabilities. Poor sampling can lead to models that fail to capture key behaviors of the underlying system, while optimal sampling improves model fidelity with fewer data points [51]. Consequently, the method used to select this data plays a critical role in the development process—nearly as important as the choice of the surrogate model itself.

Monte Carlo sampling is a simple, unbiased approach in which data points are selected purely at random. While it is inexpensive and easy to implement, it is distribution-agnostic and often results in uneven coverage of the input space, potentially leaving large gaps or clusters. To address this, quasi-random sampling methods have been developed to promote more uniform coverage. Examples include Latin Hypercube (LHC) sampling and Sobol sequences, which aim to generate samples with low discrepancy—that is, samples that fill the space as uniformly as possible [52] as seen on Figure 2.6.

Low-discrepancy sampling reduces the variation between the empirical distribution of sampled points and the ideal uniform distribution. This enhanced coverage improves the surrogate model's ability to approximate the system across the entire domain, often leading to faster convergence and better predictive performance with fewer samples than standard random sampling [53].

In contrast, active sampling strategies select points based on their estimated value in improving model accuracy. These methods have been widely shown to outperform random sampling [54] [55], especially when data is limited or expensive to obtain. Unlike the Monte Carlo approach, active sampling queries the current state of the model to guide the sampling decisions. Common criteria include uncertainty sampling, where the points are chosen based on the associated confidence. This is particularly useful for GP since every prediction comes with an uncertainty as well [56]. Another popular approach is committee-based sampling [57], where a series of different surrogate models, called a committee, make predictions for the



(a) Random

(b) LHC

(c) Sobol

Figure 2.6: Distributions drawn from different sampling methods

same data set, and the point where the predictions differ the most is chosen as the most informative one. This method is based on the assumption that the point where several methods provide a very different prediction is the one where there is the highest uncertainty and therefore needs to be sampled. One drawback is that already trained models are required with a certain level of accuracy since their predictions will determine the next sample.

One of the main objectives of this thesis is to build a database as efficiently as possible; therefore, exploring active sampling techniques is of great interest since they aim to maximize information gain. By choosing the most informative points as opposed to random ones, the surrogate model can more quickly learn trends and make generalizations. Furthermore, this is of particular interest in the field of hypersonic aerodynamics, where data points are obtained through expensive CFD or experimental methods.

2.4. Python implementation

In order to implement these methods for creating surrogate models, Python was used, where a series of libraries exist that make the development of GPs more straightforward. Firstly, scikit-learn was used to initially implement a GP since it allows quick modification of the kernel, which was useful in the initial exploration phase of the project. Further, it comes with powerful optimization algorithms to find the best hyperparameters; by default, a Limited-memory BFGS (L-BFGS) optimizer is used. In order to implement the multi-output architecture, the library GPflow was used, which already comes with an implementation for the LMC.

3

CFD model and simulation setup

3.1. CFD model

To obtain the data required for training and eventually validating the GP, as well as to construct the initial database, a series of CFD simulations were conducted. To obtain data that is an accurate representation of the problem being studied, an appropriate solver needs to be used. To achieve this, the DLR-TAU code was used, a second-order finite-volume CFD solver capable of performing viscous and inviscid flow simulations for complex geometries covering regimes from low subsonic up to hypersonic flow. Despite considering reentry vehicles which are normally in the rarefied regime (Kn>0.1), this study only looks at the lower altitudes of the trajectory, as this is where the aerodynamic forces play the most significant role. TAU includes a comprehensive range of Reynolds-averaged Navier–Stokes (RANS) turbulence models. By default, the Spalart-Allmaras one-equation model is used, which is known to be numerically robust and able to cover a wide range of applications. Further, the model provides a good compromise between accuracy and numerical efficiency and is particularly suited for flows with strong shocks [58]. Another consideration to be made when solving hypersonic problems is the chemically reacting high-enthalpy flow. In order to model this, the Gupta reaction mechanism for air flow was implemented [59]. Considering a mixture of compressible ideal reacting gases, the Navier-Stokes equations in their integral form (following notation from [60]) can be written as:

$$\frac{\partial}{\partial t}\int_V \underline{U} dV + \int_S \underline{\underline{F}}^{Eu}\underline{n} dS = \int_S \underline{\underline{F}}^{NS}\underline{n} dS + \int_V \underline{Q} dV$$

Where the vector of the conservative variables for thermal equilibrium is:

$$\underline{U} = (\rho_S, \rho\underline{u}^T, \rho E)^T$$

The matrix of inviscid (Euler) fluxes is:

$$\underline{\underline{F}}^{Eu} = \begin{bmatrix} \rho_S\underline{u}^T \\ \rho\underline{u}\ \underline{u}^T \\ \rho E\underline{u}^T \end{bmatrix} + \begin{bmatrix} 0 \\ p\underline{\underline{I}} \\ p\underline{u}^T \end{bmatrix}$$

And the matrix of viscous (Navier-Stokes) fluxes is:

$$\underline{\underline{F}}^{NS} = \begin{bmatrix} (\frac{\mu}{Sc})\underline{\nabla}^T\frac{\rho_S}{\rho} \\ \underline{\underline{P}} \\ \kappa\underline{\nabla}^T T + (\frac{\mu}{Sc})\sum_S h_S\underline{\nabla}^T\frac{\rho_S}{\rho} + (\underline{\underline{P}}\ \underline{u})^T \end{bmatrix}$$

The diffusion flux, $\rho_S u_S^d$, for species S is modeled using Fick's law using an averaged diffusion coefficient D for all species using the viscosity μ and the Schmidt number Sc:

$$\rho_S u_S^d = -\rho D \nabla(\frac{\rho_S}{\rho}) = -(\frac{\mu}{Sc}) \nabla \frac{\rho_S}{\rho}$$

The viscous stress tensor $\underline{\underline{P}}$ is modeled using the Boussinesq approximation:

$$\underline{\underline{P}} = \mu[\underline{\nabla}\ \underline{u}^T + (\underline{\nabla}\ \underline{u}^T)^T] - \frac{2}{3}\mu(\underline{\nabla}^T \underline{u})\underline{\underline{I}}$$

Lastly, the source vector $\underline{Q}$ only includes the chemical sources from the reactions ω_S

$$\underline{Q} = \begin{bmatrix} \omega_S \\ 0 \\ 0 \end{bmatrix}$$

In order to model the chemical reactions taking place, the transport equations describing the evolution of the individual species, S, need to be solved, and the chemical source term, which determines the rate of production and destruction of a species, needs to be computed. Chemical sources ω_s, are created by a series of chemical reactions involving the species X_S and the stoichiometric coefficients α_S and β_S. A chemical source term ω_S can be calculated using the law of mass action by summation over all present reactions r:

$$\omega_S = M_S \sum_r (\beta_S^r - \alpha_S^r)[k_r^f \prod_S (n_S)^{\alpha_S^r} - k_r^b \prod_S (n_S)^{\beta_S^r}], \text{ where } n_S = \frac{\rho_S}{M_S}$$

Where the forward reaction rate is obtained from the modified Arrhenius law:

$$k_r^f = (a_r^f)(\frac{T}{1K})\text{exp}(-\frac{c_r^f}{T})$$

The backwards reaction rate is obtained from the equilibrium constant:

$$k_r^b = \frac{k_r^f}{K_r^{eq}}$$

Where the equilibrium constant is computed from the partition functions of the involved species:

$$K_r^{eq} = \prod_S (\frac{Q_S}{VN_A})^{\beta_S^r - \alpha_S^r}$$

Regarding the numerical solution procedure, the solver uses an edge-based dual-cell approach based on a vertex-centered scheme. The primary grid consists of tetrahedra, prisms, hexahedra, and pyramids. This grid is used to store the flow quantities. To implement the final volume discretization, a dual-grid cell is constructed around each vertex of the primary grid, as shown in Figure 3.1. We see the dual cell (in grey) of volume V_i and surface S_i constructed around point P_L. This hybrid structured/unstructured grid facilitates the analysis of complex geometries and is optimized for the application of massively parallel HPC systems.

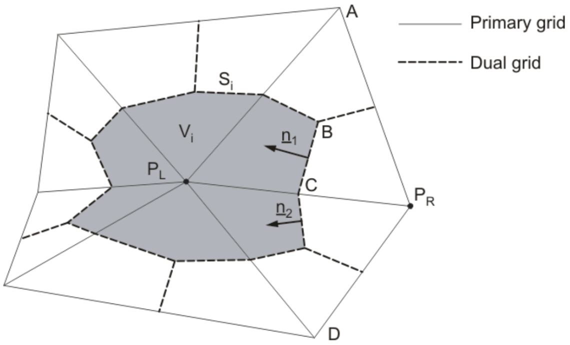


Figure 3.1: Schematic of dual cell structure in the TAU code [60]

3.2. Simulation setup

The vehicles chosen for this work are the Stardust capsule [61] and the X-38 CRV lifting body. There exists ample literature characterizing the aerodynamic performance of these vehicles, which is important for the validation of the CFD simulations and eventually for the surrogate model. Moreover, the geometries of these vehicles are widely available, which simplifies the development process. Furthermore, by considering a glider and a capsule, we can study very different architectures of reentry vehicles and can therefore analyze how the surrogate model behaves for different operating conditions and physical phenomena, crucial to assess the robustness.

3.2.1. Geometries and meshing

The geometry for the capsule was obtained from [62] and is shown in Figure 3.2. These measurements were used to create a 3D model of the vehicle using CAD, which was later used to generate a mesh (figure 3.3) using the Centaur software. Since the vehicle is symmetrical, only half of it was used for the CFD simulations in order to reduce the computational time. Furthermore, for the field around the vehicle, the area where the bow shock was expected was refined (figure 3.4). This finer area was made large enough so that the shock would be captured for different angles of attack. Since the vehicle is symmetric and we are not considering roll angles, it was possible to simulate only half of the geometry to reduce computational time. The final mesh has 621,467 cells.

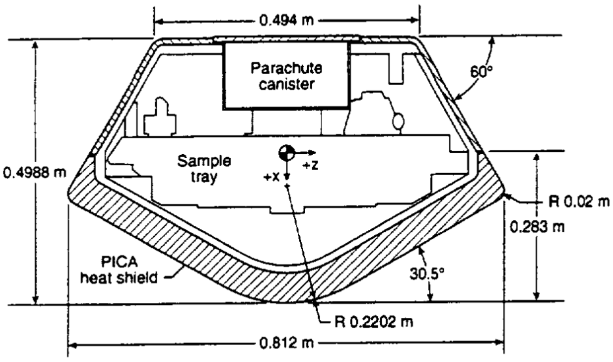


Figure 3.2: Stardust geometry [62]

The geometry for the X-38 was provided by the DLR in Göttingen, as they had previously worked on the original vehicle in the 90s. However, the provided CAD model did not include the flaps, which had to be modeled using reference pictures and manually inserted into the final geometry. This model was then meshed (figure 3.6) using Centaur. The flap deflection angles were chosen in a way to cover all possible operational conditions. 3 flap angles are considered: 0, 10, and 20 degrees. Similarly to the capsule, the

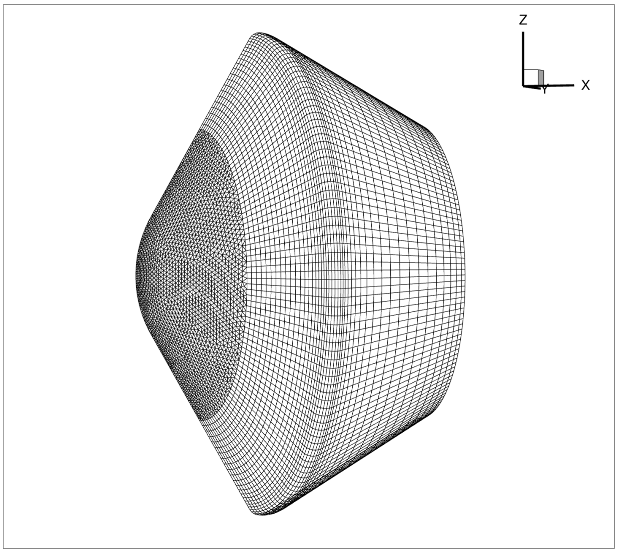


Figure 3.3: Meshed surface of the Stardust capsule

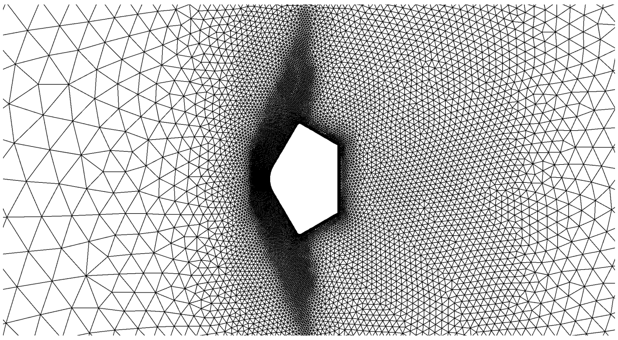


Figure 3.4: Mesh around the field of the Stardust capsule

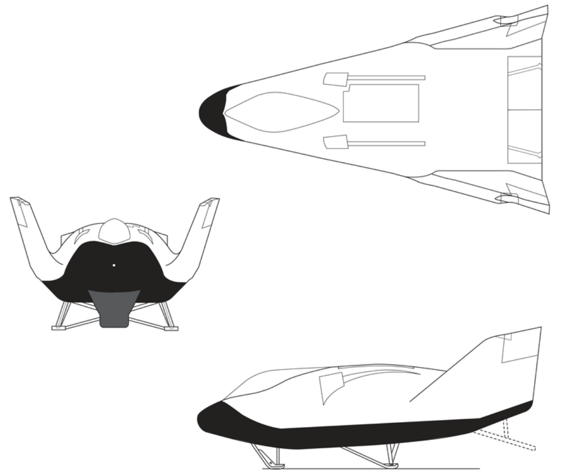


Figure 3.5: Overview of the X-38 vehicle

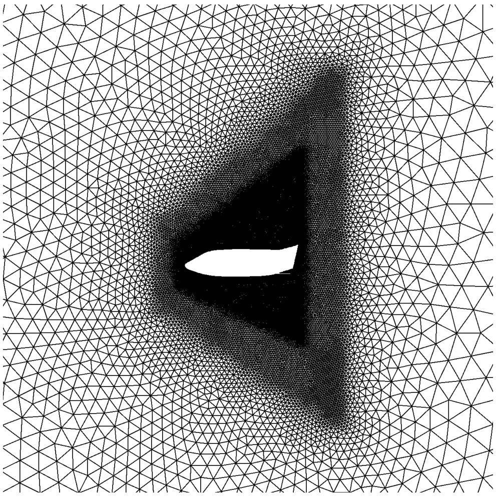


Figure 3.6: Mesh of the field around the X-38

vehicle is symmetric, and no roll angle is considered; therefore, only half of the model is simulated. This also means symmetry in the field of motion. Moreover, the field mesh was refined to capture the bow shock at different angles of attack as seen in Figure 3.6. The final mesh has 1,286,291 cells.

3.2.2. Numerical setup

The vehicle surfaces are modeled as isothermal walls with a constant temperature of 600K. As a thermal study of the vehicles is not the main focus of this work, only one wall temperature was considered. The reference values to define each trajectory point are given by a pressure value, Mach number, and temperature. An upwind method is used for the inviscid flux discretization along the AUSMDV flux vector splitting scheme.

Part III

Discussion of results

In this part, the results of the CFD simulations and the GP are discussed for each vehicle. The main focus when analyzing the CFD simulation is to validate the results by comparing our data with other published works; this way, we can be sure that our setup provides accurate results. Moreover, we then examine the capability of the GP to predict new points, emphasizing accuracy and uncertainty quantification. Firstly, we look at the capsule in Chapter 4 and then the glider in Chapter 5.

4 Capsule

4.1. CFD results

In order to validate the simulation pipeline, we compared our results to existing literature. A study by NASA [62] performed a CFD campaign for the reentry phase of the Stardust capsule. They report the aerodynamic coefficients at different trajectory points, shown in Table 4.1, for angles of attack (AoA) $0°, 5°$ and $10°$. In order to compare our simulation results, we used the same reference geometry and solver settings. In their study, NASA uses the Langley Aerothermodynamics Upwind Relaxation Algorithm (LAURA), which implements a comparable physical model and numerical setup to TAU. LAURA is an upwind-biased, point-implicit relaxation algorithm code for numerically solving the Navier-Stokes equations for three-dimensional, viscous hypersonic flows [61]. In order to remain in the continuum regime, the highest trajectory point considered was at 50.63 km. A comparison of the results obtained with both solvers is shown in Tables 4.2 and 4.3 using the relative error (ReE) as a performance parameter. Note that for 0 degrees, no lift or momentum is generated since the capsule is symmetric.

Table 4.1: Stardust trajectory points

Altitude (Km)	Mach
83.7	24.7
68.96	17.1
58.7	12.2
50.63	24.7
46.54	17.1
45.75	12.2
44.44	10.5
43.24	8.5
41.6	7.15

Table 4.2: Result comparison for AoA = $0°$

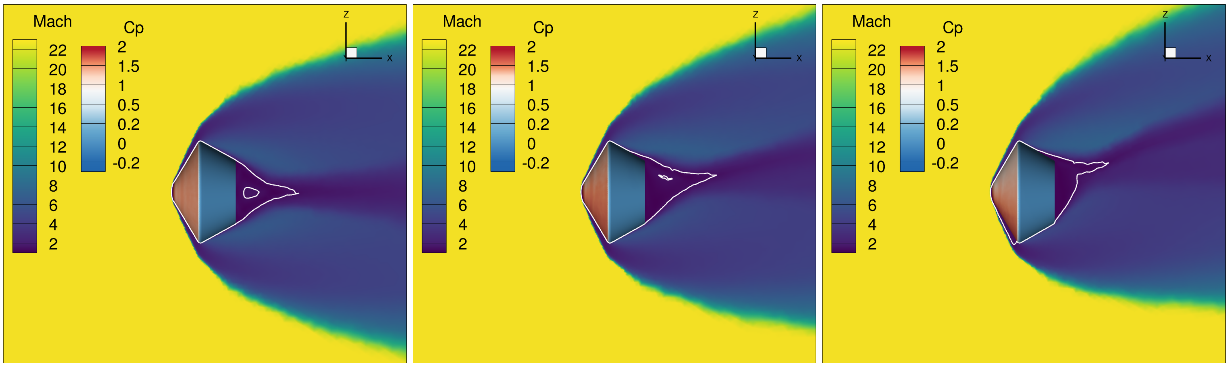
Altitude (Km)	C_D (NASA)	C_D (TAU)	ReE (%)
50.63	1.482	1.430	3.503
46.54	1.494	1.451	2.863
45.75	1.499	1.468	2.078
44.44	1.515	1.493	1.464
43.24	1.51	1.490	1.322
41.6	1.506	1.482	1.586

Table 4.3: Result comparison for AoA = 5°

Altitude (Km)	C_D (NASA)	C_D (TAU)	ReE (%)	C_L (NASA)	C_L (TAU)	ReE (%)
50.63	1.471	1.418	3.557	0.043	0.046	5.857
46.54	1.481	1.438	2.853	0.0445	0.045	1.794
45.75	1.489	1.451	2.528	0.045	0.042	5.211
44.44	1.498	1.461	2.501	0.041	0.032	20.910
43.24	1.496	1.457	2.624	0.030	0.025	18.802
41.6	1.477	1.453	1.595	0.023	0.021	8.922

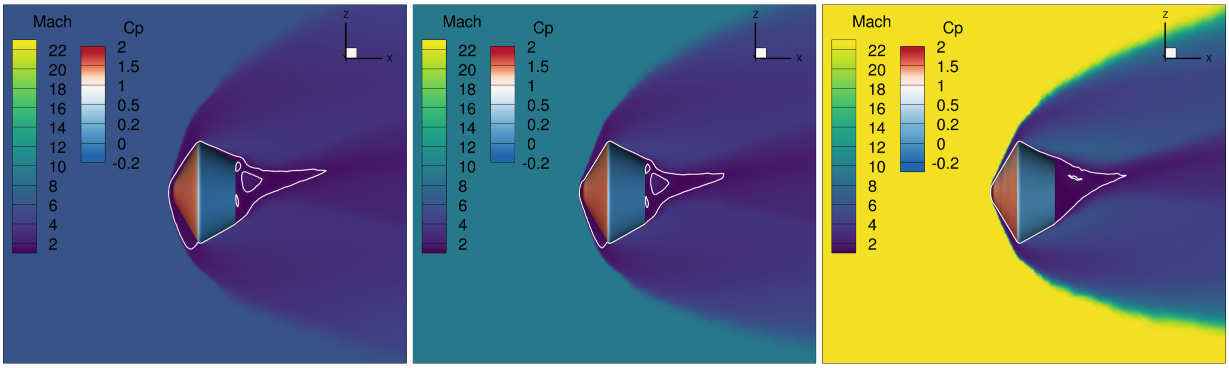
The only cases with large errors are the lowest three trajectory points when considering the lift coefficient. However, for Mach below 12, the authors of [62] assume chemical equilibrium, while this work models the chemical effects; therefore, discrepancies for this regime are expected. Moreover, for blunt reentry bodies, the lift is more sensitive to small differences in the flow field, geometry, meshing, and boundary conditions. This is because the lift force arises primarily from asymmetries in the pressure distribution over the body, especially between the windward and leeward sides [63]. In contrast, the drag force is mainly influenced by symmetric pressure loads, such as those on the forebody and base, making it relatively less sensitive to small-scale flow discrepancies. Overall, we can see that the results are very similar and conclude that our simulation was set up correctly.

We further examine the flow field to determine whether the expected physical phenomena are present. Figure 4.1 shows the effect of the angle of attack (AoA) at the highest point of the trajectory. A strong bow shock is observed, generated by the blunt shape of the capsule, along with a thin shock layer on the windward side, both of which become increasingly skewed as the AoA increases. Additionally, the sonic line (shown in white) shifts toward the shoulder region, consistent with observations in [15, 62]. This phenomenon correlates with the notable decrease in the lift force coefficient for Mach numbers below 12, as shown in Table 4.3. Furthermore, the region of highest pressure coefficient (C_p) becomes more concentrated and exhibits steeper gradients at higher AoA. The stagnation C_p consistently remains around 1.9, as expected for this type of vehicle [64]. Regarding the effect of Mach number (Figure 4.2), the shock stand-off distance decreases with increasing flow speed, while the overall pressure distribution and range remain largely unchanged.



(a) $\alpha = 0^{\circ}$ **(b)** $\alpha = 5^{\circ}$ **(c)** $\alpha = 10^{\circ}$

Figure 4.1: Effect of the AoA (α) on the flow field around the Stardust capsule at $M = 24.7$. Sonic line shown in white



(a) M=7.15 **(b)** M=10.5 **(c)** M=24.7

Figure 4.2: Freestream Mach number effect on the flow field around the Stardust capsule at $\alpha = 5°$

When visualizing the velocity divergence in figure 4.3, we can further observe a system of expansion and recompression waves on the leeward side. The velocity divergence allows us to characterize the waves present in the field. When this quantity is negative, there is a compression wave (a shock), and when it is positive, there is an expansion. Regions of hot gas form between these secondary shock structures and heat the wake, as seen in Figure 4.4, where we further see the wake and stagnation temperature increasing with the Mach number. Overall, the highest temperatures are found in the stagnation area. This phenomenon was also observed for the Mars Pathfinder mission [65] and the Viking Lander [66]. Figure 4.5 further shows areas of recirculation in the base through streamlines, which are not symmetrical as expected for an AoA of $10°$.

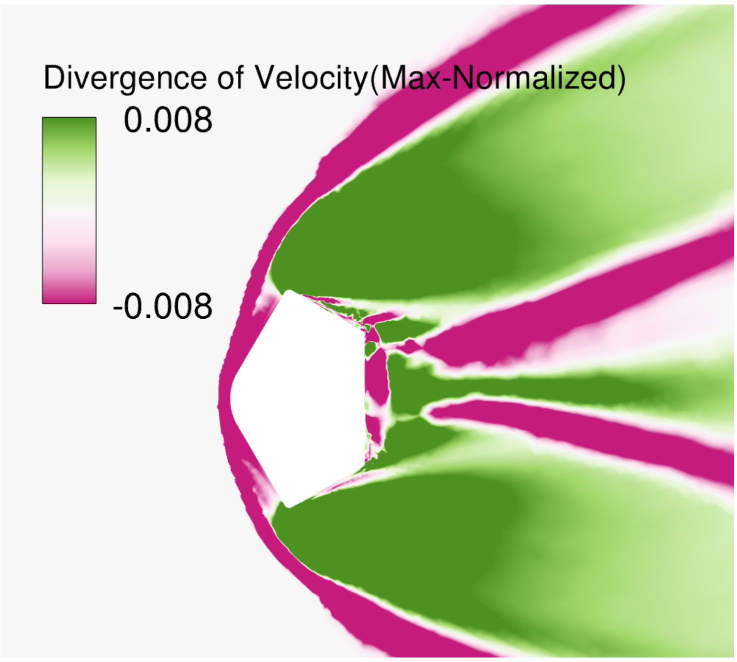
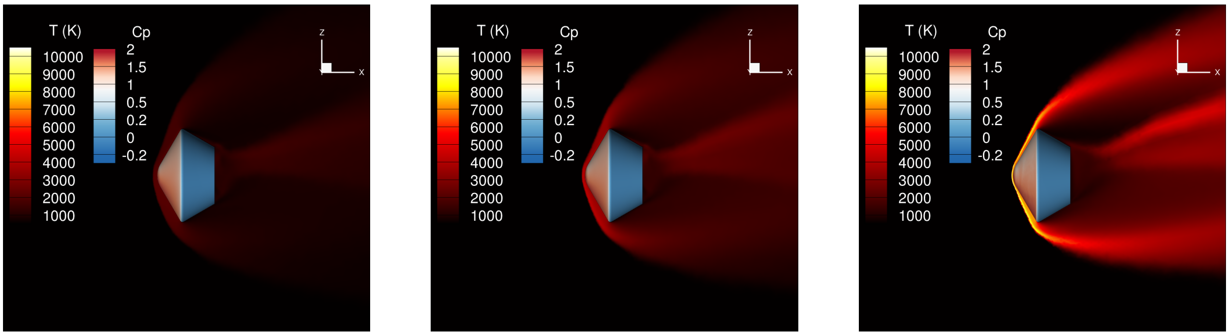


Figure 4.3: Divergence of the velocity field



(a) M=7.15

(b) M=10.5

(c) M=24.7

Figure 4.4: Effect of Mach number on the temperature field

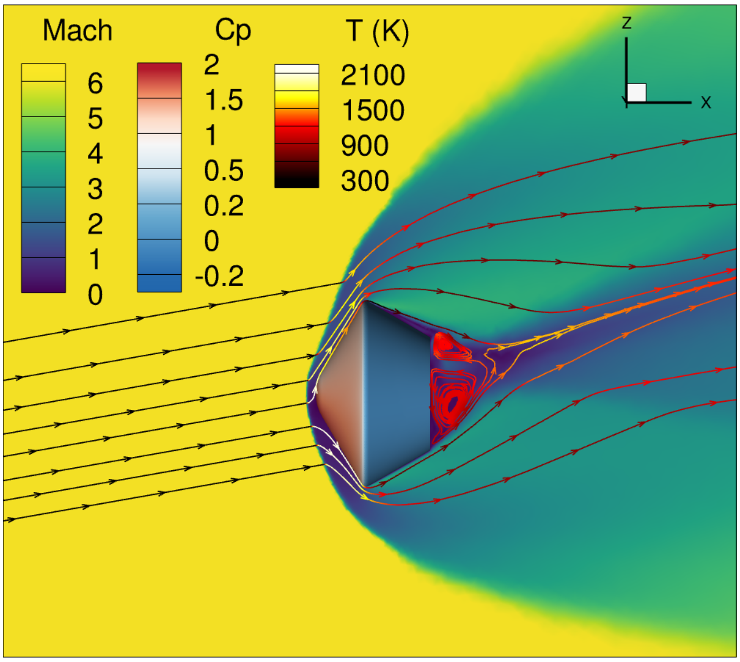


Figure 4.5: Field Mach number, surface pressure coefficient and streamlines colored by temperature for the Stardust capsule at $10°$ at M=7.15

4.2. GP results

4.2.1. Single output

This section presents the application of a GP pipeline described in Section 2 to predict the aerodynamic coefficients of the Stardust capsule. The capsule was the first test case for the implementation and validation of the Gaussian Process. This was done since this vehicle doesn't have any control surfaces; thus, an altitude h, velocity v, and angle of attack α are enough to describe a database point. The input data for the GP is therefore a 3-dimensional array $x_i = [h_i, v_i, \alpha_i]$. The objective is to construct an aerodynamic database with the least amount of points possible. In [26], a GP is developed to predict the aerodynamic performance of a reentry vehicle based on the same 3 parameters. In that study, the training data set is gradually increased, specifically the number of points is N = 10, 20, 30, 40, and 45, where 5 points are kept for validation. It was found that for $N \geq 30$, no significant improvements were found; therefore, it was determined to use 32 points as an initial baseline for the database. The distribution of these points was obtained using a Latin-hypercube (LHC) sampling scheme. The initial database is shown in 4.6. Using the TAU code, simulations were run for 32 points in order to obtain the aerodynamic coefficients, namely C_D, C_L and C_m.

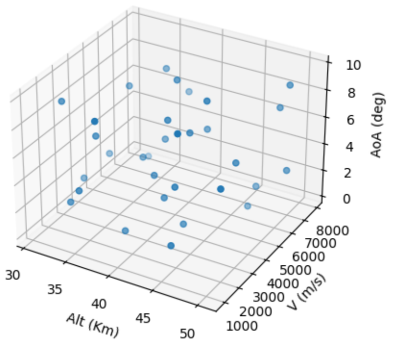


Figure 4.6: Initial database generated with LHC sampling technique

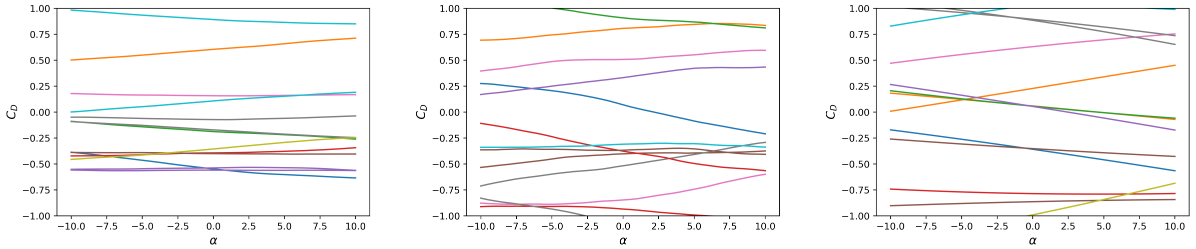
A Gaussian Process (GP) surrogate model was first trained to predict the drag coefficient. Prior to training, the input parameters were normalized, a standard preprocessing practice when using GPs (and surrogate models in general). This normalization is essential for improving the stability and efficiency of hyperparameter optimization during model fitting [67, 68].

In practice, input variables often span different ranges or units, which can distort the GP's perception of the geometry of the input space. Without normalization, the kernel (or covariance) function may assign inappropriate length scales to input dimensions, leading to skewed modeling assumptions, slower convergence of the marginal likelihood optimizer, and the risk of converging to poor local minima [47, 69]. Normalizing the inputs ensures that all features are treated on a consistent scale, allowing the optimizer to explore the hyperparameter space more effectively and improving the overall predictive performance of the surrogate model.

There was no rigorous process to choose the kernel, as developing or implementing an automatic kernel discovery tool was determined to be outside the scope of this work. Therefore, several combinations of kernels were manually tested until satisfactory results were obtained. In order to measure the performance of the model, a K-fold cross-validation metric was employed, a statistical technique used to evaluate the performance and generalization ability of a model on unseen data. It consists of splitting the data into K equal parts (folds), using one of them for validation and the remaining K-1 folds as a training set. In this case, a 4-fold cross validation was implemented, which yielded a 0.37% mean relative error (MRE). The kernel function that provided this result is the following:

$$k = RBF(\boldsymbol{l_{RBF}}) + RQ(\alpha = 0.521, l = 1e+05) + Matern(l = 249, \nu = 1.5) \tag{4.1}$$

Where $\boldsymbol{l_{RBF}}$ is an array whose entries correspond to the input dimensions' length scales ($\boldsymbol{l_{RBF}} = [l_h, l_v, l_\alpha]$), while the other two kernels have the same hyperparameters for each dimension. For comparison, if the covariance function was composed only of the Matern kernel (similar to the implementation in [26]), the MRE is 0.702%. Similarly, only including the RQ kernel, the MRE is 0.539%, and only including the RBF yields 0.873%. Overall, all kernels provide accurate predictions and the MRE is always kept under 1%. In fact, all kernels are variations of the RBF, and when looking at sample functions drawn from each case, the result is very similar, as seen in Figure 4.7 for the dimension input corresponding to the AoA. All three kernels produce smooth functions suitable for predicting drag curves. One reason that the combination of the three kernels performs better than any individual one might be the fact that, in this context, the addition serves as a OR operator, where the value of the resulting kernel is high if any of the constituting functions is high as well, possibly making the kernel more sensitive and able to capture more nuances.



(a) Kernel 4.1

(b) Matern

(c) RBF

Figure 4.7: Sample of functions using RBF derivative kernels as priors

When considering a kernel from a different family, the results are very different. For example, when choosing a periodic kernel, also called ExpSinSquared, defined as $k = \exp(-\frac{2\sin^2(\pi(x-x')/p)}{l})$ (where p is a preiodiciy parameter), the MRE jumps to around 50%. This is because we are trying to approximate non-periodic data with a periodic function, as seen from the function's samples in Figure 4.8.

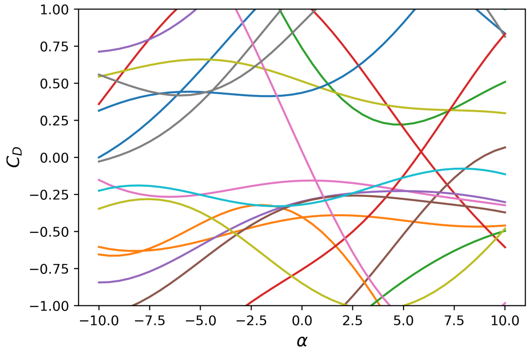


Figure 4.8: Samples using the ExpSineSquared covariance function as prior

With the normalization inverted, the optimized length scales are shown in Table 4.4. It can be seen that the length scales for the altitude and the velocity are similar to the range of the training data, whereas for the angle of attack, the length scale is smaller than the range of the training data. This result is also found in [26] and indicates that the aerodynamic coefficients are more sensitive to the angle of attack than to the other two inputs (altitude and velocity). This is consistent with the physics being modeled.

Table 4.4: Optimized length scales of the RBF kernel for drag prediction

l_h (km)	l_v (m/s)	l_α (deg)
75.6	1330	8

When it comes to predicting the lift coefficient, a different GP was trained. Obtaining an optimal kernel in this case was a more challenging task, and the final performance is below that of the drag coefficient prediction. Performing a 4-fold cross validation, an MRE of 12.3% was obtained. One reason causing this could be that lift is more sensitive to flow asymmetries and will therefore see larger changes than the drag for small changes in AoA and trajectory points, overall making it a more difficult distribution to model. Similar results were found in [70] and [29]. The kernel that provided the best results for the lift prediction is defined as follows:

$$k = RBF(\boldsymbol{l_{RBF}}) + RQ(\alpha = 0.313, l = 3.56e+05) + Matern(l = 1.33e+0.3, \nu = 1) \tag{4.2}$$

The optimized values for the length scales of the RBF component are shown in Table 4.5. Using the same reasoning as before, we observe that the lift has higher sensitivity to the AoA and the velocity compared to the drag, which is again consistent with the physical model of a blunt body.

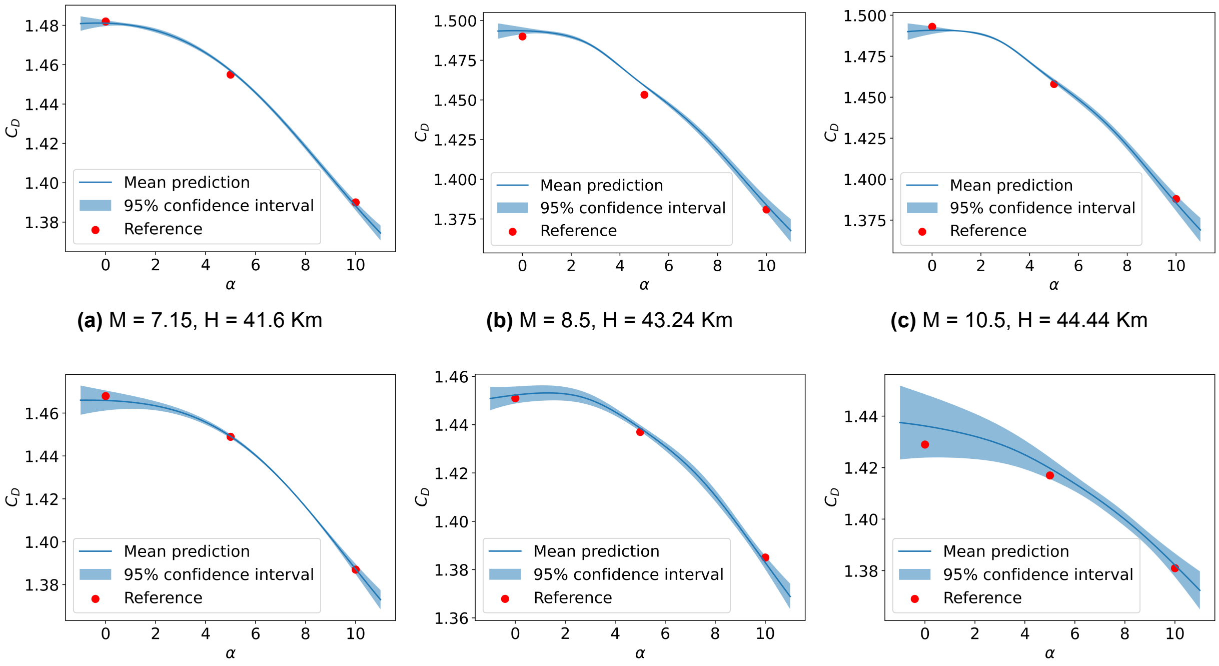
Table 4.5: Optimized length scales of the RBF kernel for lift prediction

l_h (km)	l_v (m/s)	l_α (deg)
86	843	1.7

In order to further validate the model, emphasis is now placed on more realistic and probable data points. To do this, the GP was tested on the actual trajectory points provided by [62]. Until now the performance was evaluated on the database generated with the LHS, which effectively covers the parameter domain, but fails to assess if the points are realistic for a reentry mission (e.g. high mach, low altitude). Measuring the error on more physically realistic points might give a more insightful perspective of how this tool could be used together with a trajectory planning tool, as most of the points considered will be within a realistic trajectory. The results, along with the relative error for all trajectory points, are found in Appendix A and plotted in Figures 4.9 and 4.10. We observe that when considering only trajectory points, the prediction for the drag remains very accurate, while the prediction for the lift improves greatly. In this test, the drag prediction presents an MRE of 0.2% and the lift of 2.29%. Therefore, we can conclude that non-physical trajectories, for example, high velocity low altitude conditions, make the problem harder to predict and generalize for the GP. Further, a normal reentry trajectory presents a smooth decrease in both velocity and height, which is easier to model.

The performance of the method can further be analyzed by looking at the confidence interval of the predictions. Overall, we see that most test points fall within the 95% confidence interval and are generally near the mean prediction, and as expected, the uncertainty grows in the extremes where we perform out-of-distribution predictions. For the first 3 trajectory points (Figures 4.9a, 4.9b, 4.9c) the drag prediction uncertainty is relatively small compared to the last 3 (Figures 4.9d, 4.9e, 4.9f), where it grows notably, specially for the last one. This result tells us that the GP is less certain of the prediction for higher altitude and velocity trajectory points. An explanation could be that the database was constructed in a quasi-random manner, under-representing this region. Further, at higher altitudes the aerodynamic forces are smaller, which might increase the difficulty of modeling them.

For the lift curve prediction shown in 4.10, it can also be seen that most points fall within the 95% confidence interval and are close to the mean prediction. Unlike the drag, the uncertainty stays constant for all trajectory points. Moreover, the mean prediction is not a smooth curve but has some kinks, especially for the lower points in figure 4.10a and 4.10b. This might be a case of overfitting, where the algorithm prefers to fit the observed points, rather than generalize the overall trend, which is still a problem for Bayesian inference methods like GPs [47]. In this case, using a more generic kernel with fewer hyperparameters could provide smoother predictions.



(a) M = 7.15, H = 41.6 Km

(b) M = 8.5, H = 43.24 Km

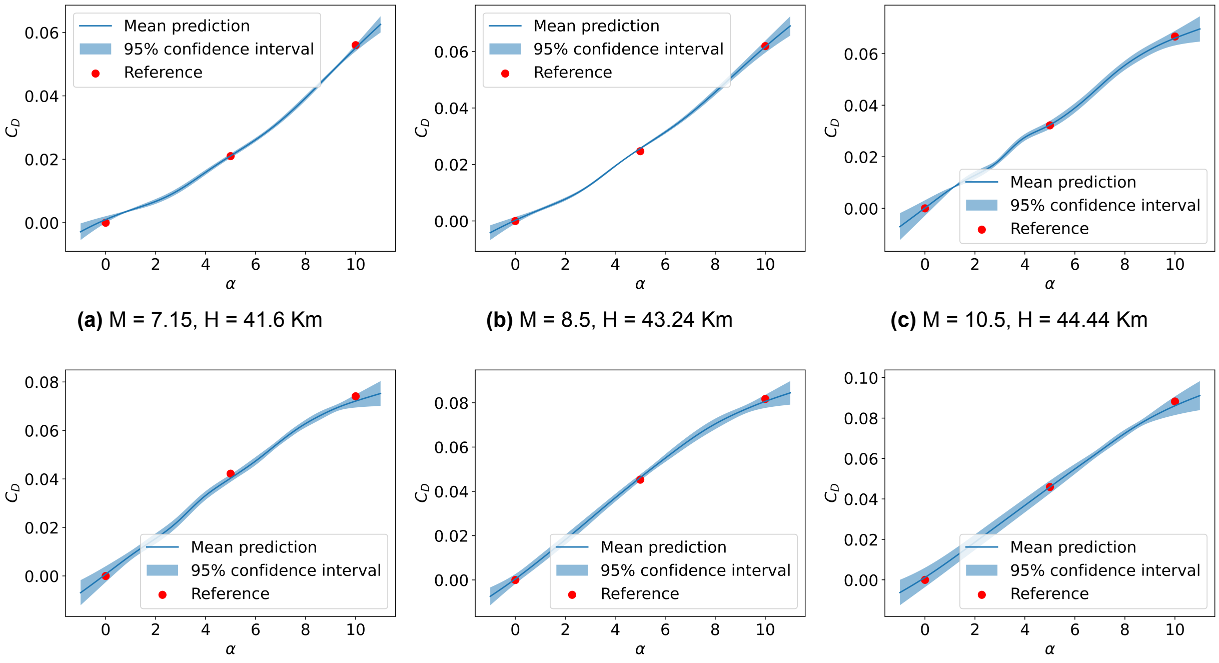
(c) M = 10.5, H = 44.44 Km

(d) M = 12.2, H = 45.75 Km

(e) M = 17.1, H = 46.54 Km

(f) M = 24.7, H = 50.63 Km

Figure 4.9: Drag polar predictions for 6 validation trajectory points



(a) M = 7.15, H = 41.6 Km

(b) M = 8.5, H = 43.24 Km

(c) M = 10.5, H = 44.44 Km

(d) M = 12.2, H = 45.75 Km

(e) M = 17.1, H = 46.54 Km

(f) M = 24.7, H = 50.63 Km

Figure 4.10: Lift polar predictions for 6 validation trajectory points

4.2.2. Active sampling

The initial database was constructed using the Latin Hypercube Sampling (LHS) method to provide a rapid and space-filling coverage of the input parameter space. However, while LHS is a quasi-random sampling scheme that ensures uniform marginal distributions, it has certain limitations—most notably its model-agnostic nature and its inability to incrementally refine an existing sample distribution without compromising space-filling properties. In this section, we explore methods for sampling optimal points to expand our database. Mainly, we explore the use of committee-based sampling and uncertainty sampling.

To implement the committee-based sampling, two separate committees were constructed, each composed of two Gaussian Process (GP) models with different kernels (summarized in table 4.6). To quantify model disagreement in the committee-based approach, metrics such as entropy and Kullback-Leibler (KL) divergence are commonly used [71]. In this case, the element-wise KL divergence was used, defined as:

$$\text{kl\_div}(x, y) = \begin{cases} x\, log(x/y) - x + y & x > 0, y > 0 \\ y & x = 0, y \geq 0 \\ \infty & \text{otherwise} \end{cases}$$

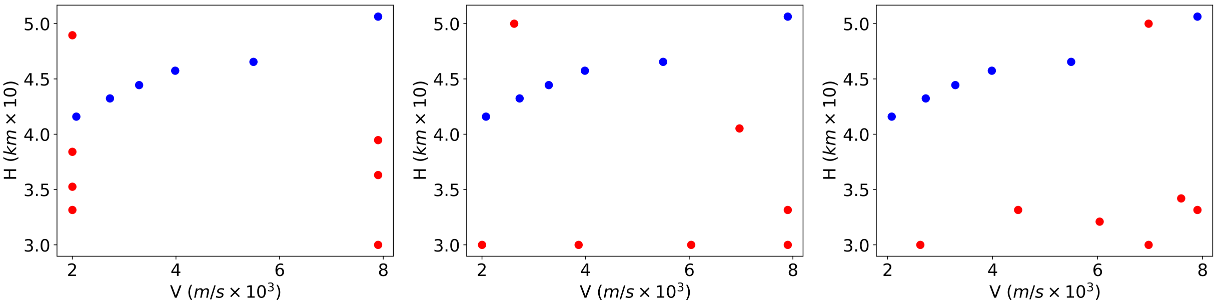
Table 4.6: Committee definition

Committee	Kernel 1	Kernel 2
A	RQ	Matern
B	RBF	RQ

In parallel, we employ a second strategy, leveraging the inherent uncertainty estimates provided by Gaussian Processes. In this greedy uncertainty (GU) method, the next sampling point is simply chosen as the one with the highest predicted variance. This method is straightforward to implement and does not require multiple models.

We therefore compare the performance of these methods to determine the best one. As an initial database for all methods, the last 6 trajectory points from section 4.1 are used. This was done to simulate a case with an extremely small database. In order to assess the performance, the same GP was trained with the points determined by the 3 different active sampling strategies, and its uncertainty distribution is used as a performance indicator. This GP was kept constant in order to evaluate the performance of the point distribution only. Since this was the initial test for any active sampling method, the angle of attack was fixed to 5 degrees, and only the drag coefficient was studied in order to simplify the problem and understand the performance of each method more clearly. This procedure was very time-consuming as we had to wait for the results of the previous simulation to determine the next sampling point; therefore, after 8 iterations, the process was stopped. The final databases provided by the different methods are shown in Figure 4.11.

Generally, all three methods chose points near the extremes; however, committee B does this to a lesser degree, going further inside the domain and being the only one to choose a point within the trajectory. It is also worth noting that while the committee A and the greedy uncertainty method seem to choose entirely points within the limit of the domain, the GU method prioritizes extremes on the velocity domain (Figure 4.11a), while committee A does so for the altitude domain (Figure 4.11b). The mean uncertainty for each method at different steps is shown in Figure 4.12. Overall, we see that the committee B and the GU approach provide similar results, outperforming committee A. Overall, the GU method provides a more constant performance, constantly decreasing the uncertainty.



(a) Greedy uncertainty method **(b)** Committee A **(c)** Committee B

Figure 4.11: Databases generated with 3 different methods. In blue the initial database, in red the added points for each method

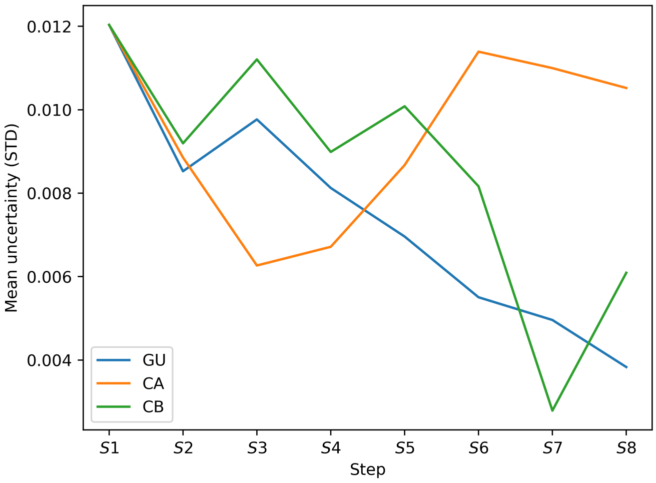


Figure 4.12: Mean uncertainty evolution of successive steps for different sampling methods

Furthermore, we look at the distribution of the uncertainty of a GP trained with the different databases obtained, shown in Figures 4.13 - 4.15. Firstly, it is interesting to note that the choice of points greatly influences the distribution of the uncertainty. As expected, the uncertainty is the lowest on the points belonging to the database. We can see that when the sampling method covers a wide range of the vertical axis (Figure 4.11a), the uncertainty spreads horizontally (Figure 4.13). Analogously, when the database covers much of the horizontal axis (Figure 4.11c), the uncertainty spreads vertically (Figure 4.15). However, comparing the database of committee A and committee B, one would expect similar performances, as the point distribution is similar overall; however, as previously stated, committee B far outperforms committee A. We can see that for committee A, the uncertainty spreads more evenly in all directions from each data point, but with less intensity, leaving large gaps of high uncertainty overall, performing worse.

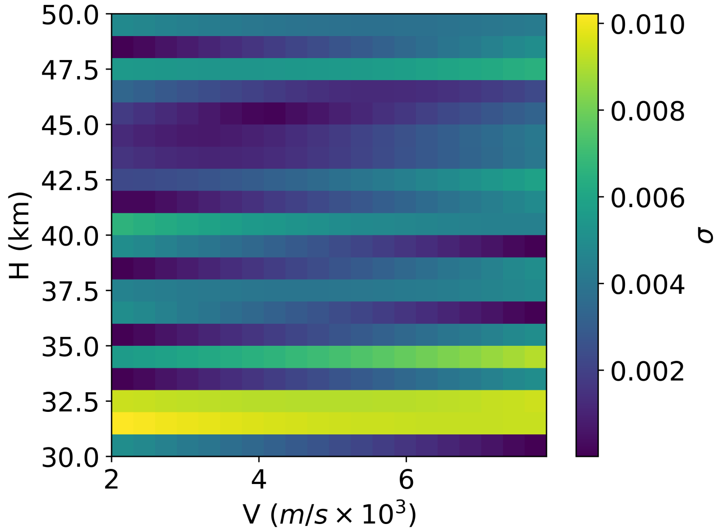


Figure 4.13: Uncertainty distribution for data base generated with the GU method

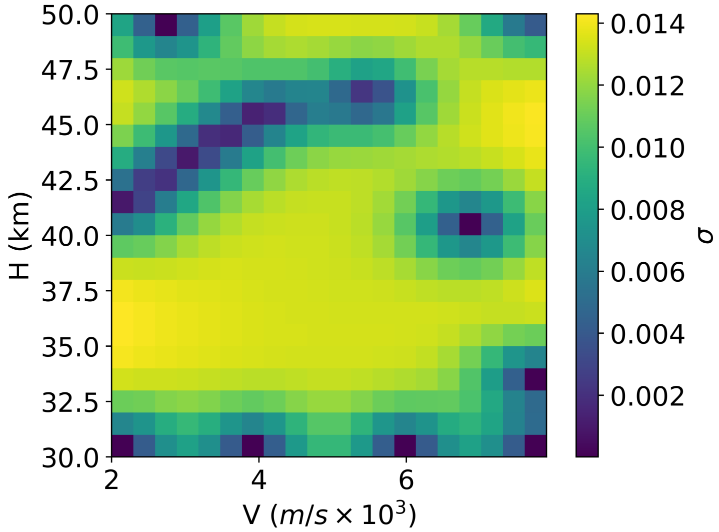


Figure 4.14: Uncertainty distribution for data base generated with the committee A

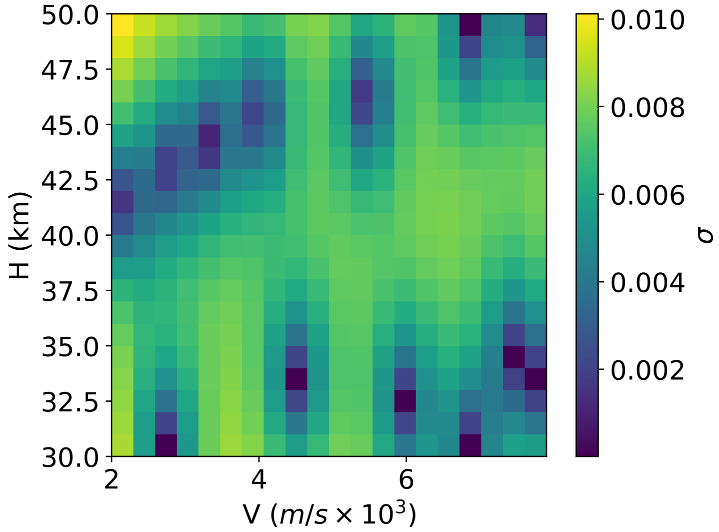


Figure 4.15: Uncertainty distribution for data base generated with the committee B

In general, we see that the 3 different methods provide different databases, which further generate different uncertainty distributions. The greedy uncertainty method, which chooses points on the extremes of the velocity domain as most informative, seems to reduce the mean uncertainty the most, and provides a more even uncertainty distribution.

4.2.3. Multi-output GP

In this section, we explore the implementation of the multi-output GP trained for the capsule. The idea behind this method, as stated in section 2.2, is to learn the relationship between the predicted outputs to further strengthen the accuracy of the GP. For this test, the linear coregionalization method (LCM) was implemented using the GPflow Python library [42]. LCM, as opposed to other MOGP implementations, allows each output to have its own hyperparameters, which theoretically can enable simulation predictions for vastly different distributions. Since the purpose of this section is only to explore the possible benefits of an MOGP compared to a single-output GP, parameters like the inducing variables will not be explored. Furthermore, the library used to implement the multi-output architecture was different from the one that produced the results of the single-output GP in section 4.2.1. Therefore, direct comparison to those results is not entirely possible. The main difference is that GPflow allows for less control when defining the kernel function, while scikit-learn provides a wider range of functions and parameters to choose from. This causes a single-output GP trained with scikit-learn to outperform one trained with GPflow. The purpose of this section is to analyze how the performance of a GP changes when increasing the number of outputs, which was tested by initially predicting only the C_D, then predicting the C_L and C_D contemporarily, and finally predicting C_M as well. The method was trained with the database shown in Figure 4.6 and validated using the trajectory points previously used to validate the single-output GP.

We first look at the drag prediction, where results are shown in Table 4.7. It can be seen that for all angles of attack, increasing the number of outputs from 1 (C_D) to 2 (C_D and C_L) provided an improvement in the accuracy of the drag prediction, showing that it is beneficial to predict 2 quantities simultaneously. This proved that the LCM is able to learn the relation between the lift and drag and improve the predictions. However, when adding the C_m to the outputs of the GP, little improvement was found for angles of attack 0 and 5 degrees, and a decrease in accuracy was found for 10 degrees. This was an unexpected result as the momentum coefficient is clearly physically related to the other two quantities and should provide more information about the underlying equations behind the aerodynamic performance of the capsule. This test was repeated several times with different settings and kernels, but the results were always similar; therefore, the results seem to show that the C_M is not always beneficial for the drag prediction. A similar result was found when an MOGP was trained to predict C_D, C_L, and L/D. In this case, the third output didn't provide any improvement for the prediction. However, this is expected since it is a linear combination of the other outputs and is therefore not informative. The addition of the C_m should be informative, therefore a possible explanation is that the complexity of the relation between C_m and the other two coefficients is not well described by the available database.

Furthermore, the performance of the lift prediction shown in Table 4.8 does not show a clear trend. For 5 degrees, the performance between 1 and 2 outputs is almost identical, worsening slightly, and improving with the third output. For 10 degrees, the trend is similar to the drag prediction; 2 outputs are beneficial, but 3 are not. However, unlike previously, the final result is worse than the single output result.

MRe (%) of C_D prediction

	1 Output	2 Outputs	3 Outputs
$0°$	0.40746825262376	0.317470041081502	0.312670809563351
$5°$	0.171002683479302	0.160746422877557	0.150537862896209
$10°$	0.71066743182376	0.528243808866083	0.542288067771018

Table 4.7: Mean relative error for the prediction of the drag coefficient for different angles of attack and number of outputs of the GP

MRe (%) of C_L prediction

	1 Output	2 Outputs	3 Outputs
$5°$	7.61444934477191	7.67928081189179	5.47788479291089
$10°$	3.37910623176539	2.80417209596946	3.82705219815986

Table 4.8: Mean relative error for the prediction of the lift coefficient for different angles of attack and number of outputs of the GP

From a physical standpoint, all three aerodynamic coefficients are clearly related. However, this interconnection is not clearly translated to an MOGP architecture. When comparing the performance between 1 and 2 outputs, there is a slight improvement for all cases except one. However, when adding the third output, there is no clear performance trend. We might conclude that the LMC is capable of learning the relation between two outputs, but the third proves to be a more challenging task. Indeed, we are increasing the dimensionality of the problem while keeping the database constant.

This section presented all results related to the Stardust capsule. The CFD simulations captured the expected physical phenomena and showed strong agreement with results from existing literature, providing confidence in their physical accuracy. Establishing the validity of the CFD data was a crucial step, as this dataset forms the basis for training the Gaussian Process surrogate model. Consequently, the underlying distribution of the data must accurately reflect the true physical behavior of the system to ensure the surrogate model is both realistic and reliable.

The single-output GP provided high-accuracy predictions and was able to accurately describe lift and drag distributions with low uncertainty. Moreover, it was found that the uncertainty sampling method outperformed both committee-based sampling schemes, proving to be an efficient method to expand an existing database and quickly reduce the uncertainty. Lastly, the MOGP seems to be beneficial when predicting two outputs; however, the third output is not always beneficial. The knowledge gained in this section will now be used for a different test case. The X-38 Glider will now be explored in the next section.

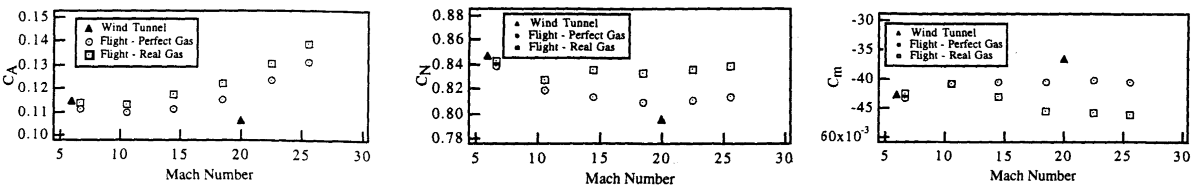
5

Glider

5.1. CFD Results

Following the same procedure adopted for the capsule, we look at existing data in the literature to validate our CFD simulations for the X-38 reentry vehicle. The flow conditions for the simulation were chosen from the original vehicle trajectory. This data was provided by DLR Göttingen and is shown in Appendix B. In order to stay within the continuous regime, the highest trajectory point was at 70.16 Km.

A NASA study [72] conducted a series of CFD simulations to validate wind tunnel results performed for the X-38; Their results for the aerodynamic coefficients are presented in Figure 5.1. The study does not provide exact numerical results, so we have to approximate the values from these figures. Before comparing the results, some differences in the setup must be addressed. Firstly, the authors of [72] use the 'Rev 3.1' geometry [73], which corresponds to a slightly smaller version of the vehicle. To match the dimensions, we scaled our model down by 3%. However, it is not clear if other changes are present, since only the length and wingspan of the vehicle are provided. Furthermore, the NASA study uses a flap deflection of $\beta_{bf} = 25°$, while we use $\beta_{bf} = 20°$. Since the purpose of this comparison was only to validate our results, only 1 of the 6 flight cases was simulated, corresponding to the lowest trajectory point (M = 6.68, Alt = 40 Km) where aerodynamic forces are more dominant due to the denser atmosphere. A comparison of the results is shown in Table 5.1. In general, the relative error is between 3.28% and 14%, which, taking into account the differences in the setup, allows us to conclude that our simulations are representative of the problem and provide physical results. Unlike the capsule in Chapter 4, here we observe the highest error in the axial force coefficient. This makes sense as we are no longer dealing with a blunt body but with a reentry glider configuration. In this case, the lift becomes more stable and less sensitive to small changes in geometry and angle of attack.



(a) Axial force coefficient **(b)** Normal force coefficient **(c)** Moment coefficient

Figure 5.1: Comparison of experimental and numerical results for the aerodynamic coefficient of the X-38 from [72]

Furthermore, looking at the flow field, we see the expected phenomena for winged reentry vehicles. In Figure 5.2, we can see the complete flow field for an angle of attack of $45°$ and flap deflection of $20°$, where we see the characteristic strong bow shock. The shock structure matches experimental Schlieren imaging from [73] shown in Figure 5.3. It is worth noting that this image was obtained for M=6; in fact, we see a larger shock stand-off distance. It is also noted that the models have slightly different geometries; mainly, the experimental model seems to be slimmer. However, the overall shock structures are very

Table 5.1: Result comparison for X-38 at M = 6.68, Alt = 40 Km

	TAU	NASA	ReE (%)
C_A	0.1288	0.113	13.98
C_N	0.7615	0.84	9.34
C_m	-0.412	-0.426	3.28

similar. The streamlines show flow separation on the upper surface, which is expected at an angle of attack of $45°$. This separation region also coincides with an area of negative C_p, which likely indicates an adverse pressure gradient. Figure 5.4 further shows how the flow field changes with the AoA, where it can be seen that the area of negative C_p grows in size and intensity as α increases. A similar result was found in [72]. The location of the highest C_p corresponds to the expected stagnation point in the nose and belly of the vehicle, and we observe negative values at the top of the vehicle, especially at the interface between the wings and the body. Furthermore, as the Mach number increases (Figure 5.5), the stagnation region becomes more concentrated and the gradients become stronger. In addition, it can be seen that the shock stand-off distance decreases with increasing flow velocity.

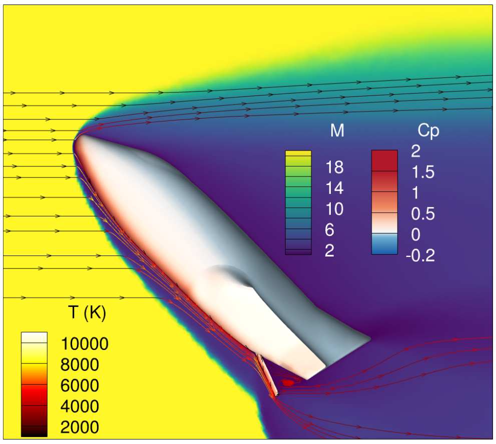


Figure 5.2: Flow field around X-38 at $\alpha = 45°$, $\beta_{bf} = 20°$, $M = 20.87$, $H = 70.15$Km

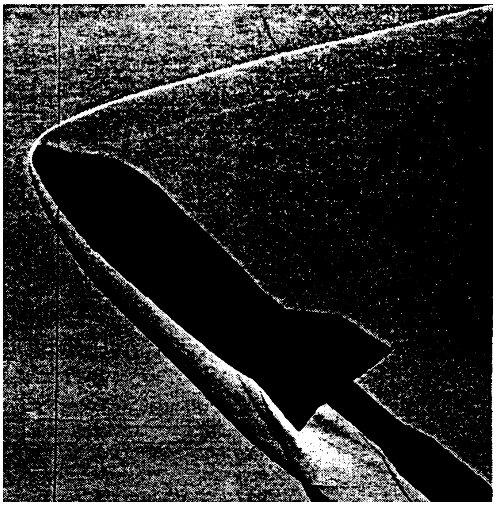
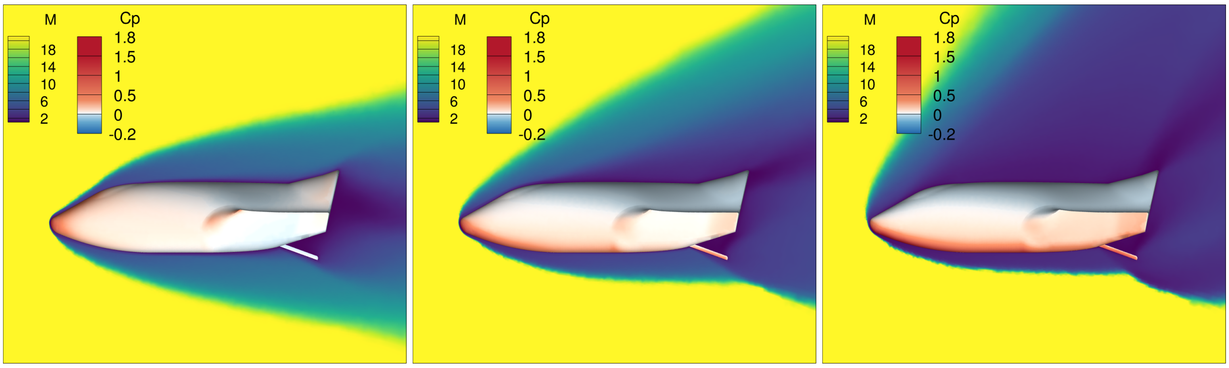
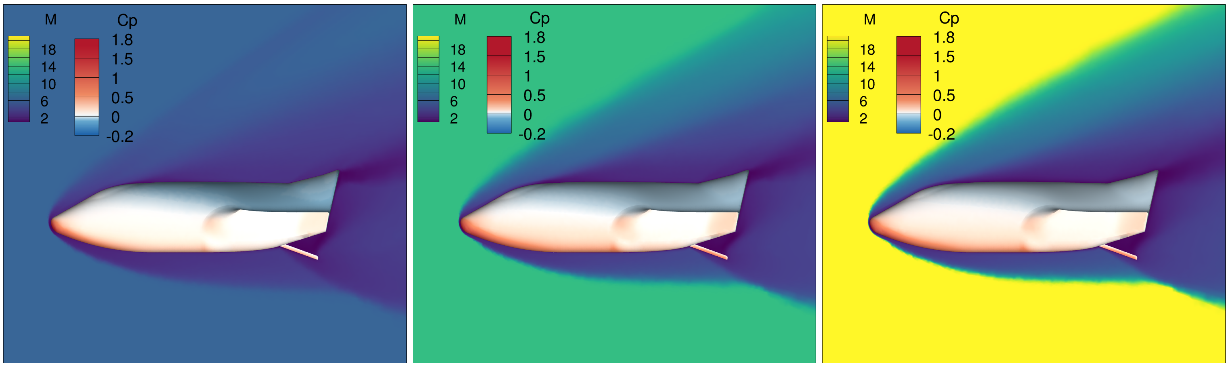


Figure 5.3: Experimental Schlieren imaging for M=6, $\alpha = 40°, \beta_{bf} = 25°$ [73]



(a) $\alpha = 0°$ **(b)** $\alpha = 22.5°$ **(c)** $\alpha = 45°$

Figure 5.4: Effect of the AoA (α) on the flow field around the Stardust capsule at $M = 20.9, H = 70.157\ Km, \beta_{bf} = 20°$



(a) M=7.15 **(b)** M=10.5 **(c)** M=24.7

Figure 5.5: Freestream Mach number effect on the flow field around the X38 at $\alpha = 22.5°, \beta_{bf} = 20°$

The full temperature field is shown in Figure 5.6, where two regions with high temperatures are identified. Firstly, at the stagnation point in the nose of the vehicle, this is due to the energy transfer of the incoming flow. This can also be seen through the colored streamlines in Figure 5.7. Moreover, we see high temperatures in the wake, which are most likely caused by the recirculation of hot gases in that region. We also observe high temperatures near the flap, most likely caused by a shock wave. An area of recirculation is observed in the flap as shown in Figure 5.8, where the streamlines also show the high temperatures near this control surface.

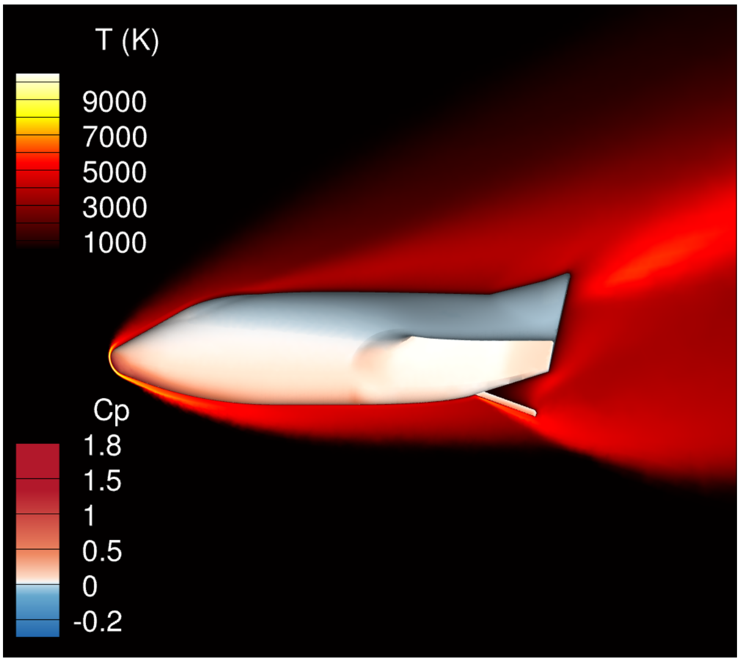


Figure 5.6: Temperature field around X-38 at $\alpha = 22.5°$, $\beta_{bf} = 20°$, $M = 20.87$, $H = 70.15$Km

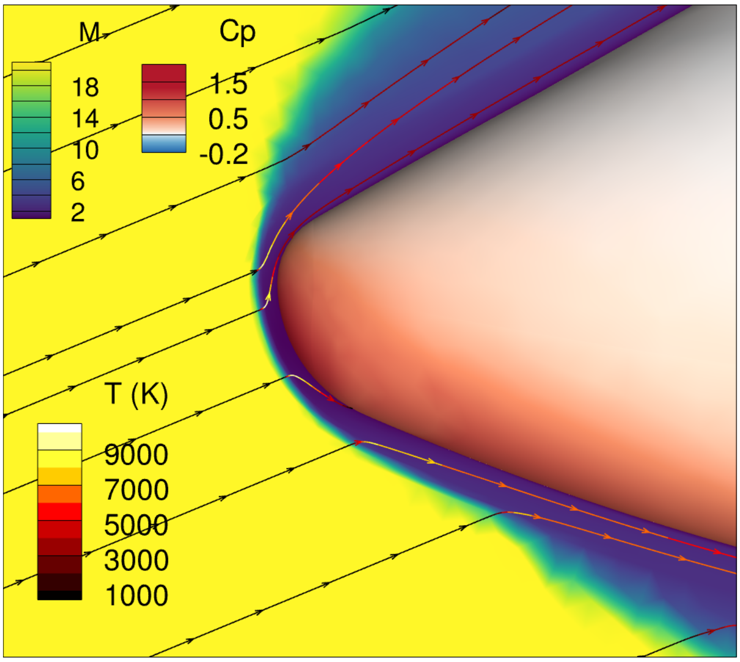


Figure 5.7: Flow field near the nose of the X-38

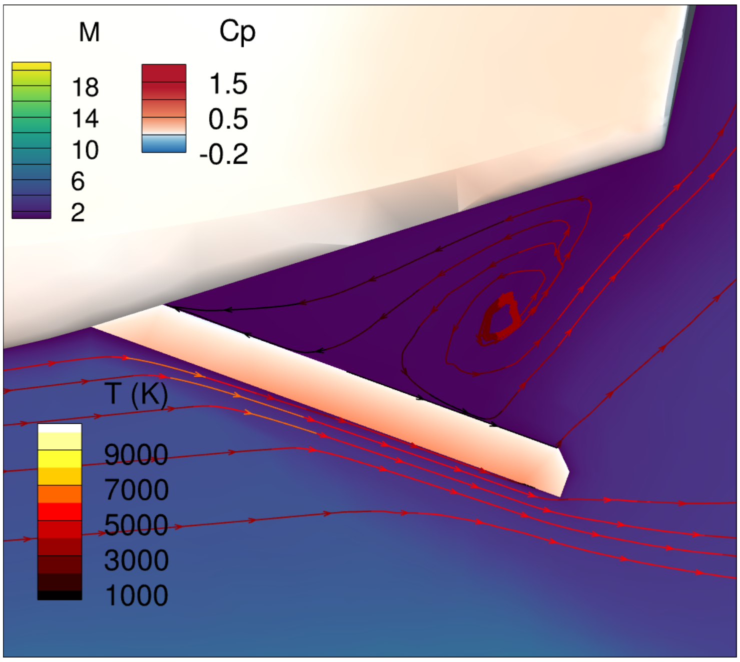


Figure 5.8: Flow field near the flap of the X-38 with an angle deflection of $\beta_{bf} = 20°$

5.2. GP results

The purpose of this study was to analyze how the previously used GP architecture would behave for a different vehicle type, with more dimensions. In this case, every data point is described by a Mach number M, an altitude H, an angle of attack α, and a flap deflection β_{bf}. The database was limited to 36 points, corresponding to 4 different trajectory points for 3 angles of attack and 3 flap deflections. We first look at a single-output GP and compare the results to those of the capsule. One main difference from the capsule is the method used for constructing the initial database. Like previously stated, the flap angle deflections had to be chosen beforehand, so no optimized sampling strategy could be applied. However, the range of values was covered evenly, like the LHS scheme would do. Similarly, for the angle of attack, only 0, 22.5, and 45 degrees were considered. Furthermore, based on previous knowledge, we knew that the aerodynamic coefficients would not change much at the higher points of the trajectory; therefore, more points were chosen in the high-density regime. The range of the input values are as follows $Alt \in [43.479, 70.157]Km, M \in [6.2, 20.88], \alpha \in [0°, 45°]$, $\beta_{bf} \in [0°, 20°]$.

5.2.1. Single output GP

Similarly to the capsule, we first look at the performance of individual single-output GPs and their performance for the different quantities of interest. The two main differences for this case are the type of vehicle and, most importantly, the dimensionality, since we now consider the additional parameter of flap angle deflection β_{bf}. An interesting result is that the kernel used for the capsule remains a good choice. However, it was found that removing the Matern function provided better results for all predictions. Furthermore, unlike the capsule, the same kernel was able to provide good results for all 3 aerodynamic coefficients. The kernel is defined as follows:

$$k = RBF(\boldsymbol{l_{RBF}}) + RationalQuadratic(\alpha = 0.152, l = 111)$$

Using the same 4-fold cross-validation as before, a mean relative error of 6.14% was obtained for the C_D, 13.51% for the C_L, and 8.22% for the C_M. It is worth noting that in this case, the performance of all individual GPs was worse than for the capsule. However, this was an expected result as the size of the database was kept constant while the dimensionality of the problem was increased. Nonetheless, the accuracy is kept at an acceptable level. Moreover, we can look at the value of the optimized length scales for the RBF kernel in Table 5.2. It is interesting to observe that a similar range of parameters is obtained for both the capsule and the glider. In this case, the length scale of the height and velocity is slightly smaller, indicating more sensitivity to these parameters. The sensitivity to the angle of attack seems to be similar. Moreover, the length scale of the body flap angle is similar to the range of values used for the training, indicating a similar sensitivity to the height or flow speed. Showing that, as expected, the flap has a noticeable effect on the aerodynamic coefficients.

Table 5.2: Optimized length scales of the RBF kernel for all coefficients prediction

l_h (km)	l_v (m/s)	l_α (deg)	β_{bf}
65	414	1.1	10.7

Visualizing the uncertainty is a more challenging task in this case since it is a 4-dimensional problem. However, a useful way to do so is through the aerodynamic polars, where we can visualize the 95% confidence interval of the prediction over a range of α values. For the trajectory point defined by $H = 56.905$ Km, $M = 13.451$, with a flap deflection of $\beta_{bf} = 10°$, the predicted drag curve is shown in Figure 5.9. We can immediately see that there are 3 points with zero uncertainty, which correspond to 3 existing training points. As expected, the uncertainty reaches a maximum between these known points. For a contrasting example, the predicted drag polar for a case with flap deflection $\beta_{bf} = 15°$ is shown in Figure 5.10 for which there are no corresponding training points. As expected, the uncertainty never goes to zero, but it is important to note that it is kept constant and does not grow uncontrollably. For the case with flap deflection $\beta_{bf} = 10°$, the mean uncertainty along the curve is 0.0097σ, while for the $\beta_{bf} = 15°$ case it is 0.0141σ. Similarly, for the lift curve, the mean uncertainty for the $\beta_{bf} = 10°$ case is 0.0242σ, and 0.041σ for the $\beta_{bf} = 15°$ case. This overall small difference in performance, even for ranges with no existing training points, shows that the GP can make good out-of-distribution predictions. The lift polar prediction for the $\beta_{bf} = 10°$ case is shown in Figure 5.11, where we see the same trend as for the drag, where the uncertainty reaches zero for the training points. While the scale makes the uncertainty look smaller, the previous numerical values show that the performance for the lift prediction is worse, a trend that is constantly observed in this work.

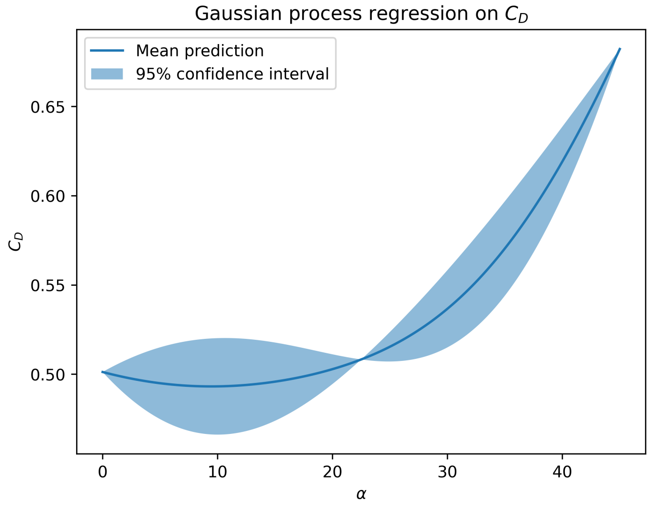


Figure 5.9: Predicted drag curve for $\beta_{bf} = 10°$, $H = 56.905$ Km, $M = 13.451$

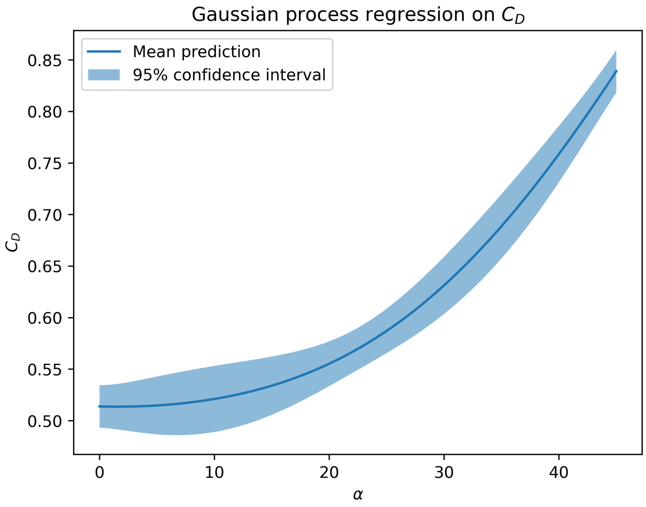


Figure 5.10: Predicted drag curve for $\beta_{bf} = 15°$, $H = 56.905$ Km, $M = 13.451$

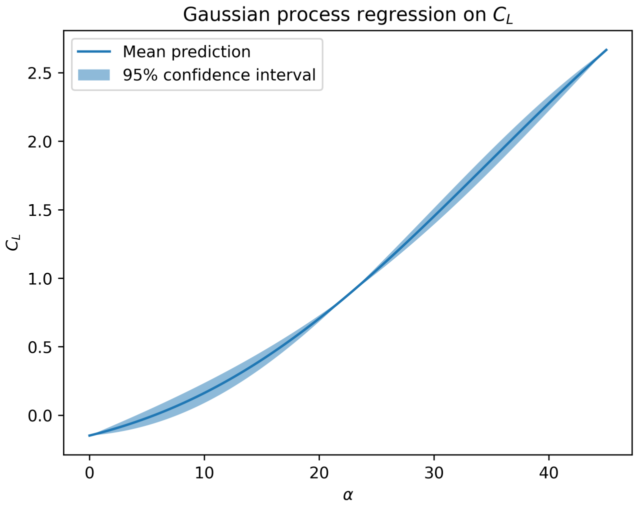


Figure 5.11: Predicted lift curve for $\beta_{bf} = 10°$, $H = 56.905$ Km, $M = 13.451$

As discussed in section 4.2.3, the greedy uncertainty (GU) method was the best-performing active sampling strategy; therefore, it is the one adopted for the glider. However, in this case, only one additional point was sampled. For a GP trained with the original database, the uncertainty over all the domain was of 0.0169σ for the drag, and of 0.04752σ for the lift. After adding an additional point with the GU method, this changed to 0.0153σ and 0.0434σ, respectively, representing an improvement in accuracy of 9.5% for the drag and 8.7% for the lift. However, when dealing with aerodynamic databases, an important point was addressed in [23]. The authors argue that considering the performance of a model on a complete range of values might not be representative of the true performance, since possible future observations will follow a certain distribution, in this case, a reentry trajectory. Therefore, the high uncertainties seen in extreme values (i.e, high Mach low altitude) might skew the analysis. This had been considered to some extent in Section 4.2.1 when we observed that the GP performed better when only considering points within the trajectory. To assess this, we now look at the accuracy of predictions for points within a real trajectory, shown in Figure 5.12, which was provided by the DLR Göttingen. With the original database, the uncertainty in this region is 0.0139σ for the drag and 0.0405σ for the lift. Therefore, compared to the entire domain, predictions within a trajectory are already 17.8% and 14.8% more accurate for the drag and lift, respectively.

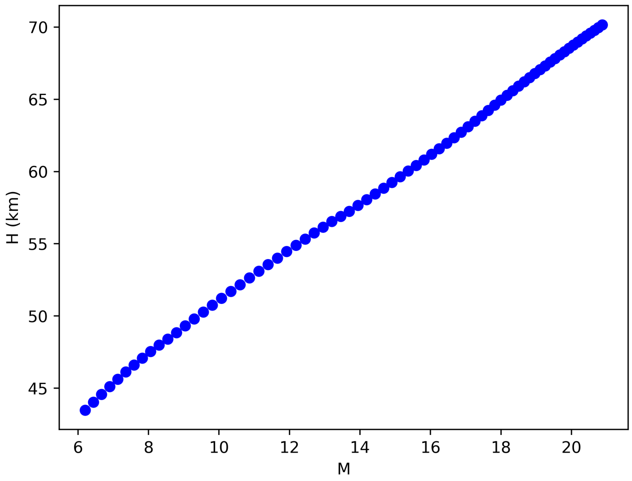


Figure 5.12: Trajectory points of the X-38

When adding the additional data point through active sampling, the uncertainty within the trajectory decreases by 2.9% and 2.5%, respectively, compared to the decrease of 9.5% and 8.7% in the global domain. Therefore, we can see that this type of active sampling benefits the global performance more than the one for possible future observations. This makes sense when considering that the GU method often chooses points in the extremes of the domain. Therefore, in the future, a sampling strategy that is biased towards realistic trajectory points might further improve the data efficiency. In fact, there is no point in improving the uncertainty of predictions that will likely not be considered for the development of a reentry vehicle.

5.2.2. Multi-output GP

The purpose of this section is to study the influence of the number of outputs in an MOGP on its accuracy in predicting the aerodynamic coefficients of the X-38. Similarly to the single output case, it was found that the same kernel for the capsule also provided good results for the glider. The covariance function used to conduct these tests is defined using GPflow as follows:

$$k = gpf.kernels.Matern52() + gpf.kernels.RationalQuadratic()$$

To perform the test, each aerodynamic coefficient was predicted using 1, 2, and 3 outputs, and the error was measured on a validation set that included new trajectory points shown in table 5.3. These points were chosen as they lay between the points used for the initial database. The results for the MOGP test are shown in Table 5.4. Overall, we observe the same trend as for the capsule; in general, increasing the number of outputs improves the accuracy. However, for different coefficients, there seem to be more informative outputs than others. In this case, including the moment coefficient seems to help the drag prediction more than the lift. From a physical perspective, this makes sense. Flow separation is captured by the C_m, which for winged bodies like the X-38 affects the C_D more severely than the C_L. We can therefore conclude that not all outputs contribute equally, and there are more informative outputs than others. This finding is supported by similar studies [67] [74] [75].

Table 5.3: Validation data set for MOGP for Glider

Altitude (Km)	M	α	β_{bf}
68.757	20.055	22.5	0
62.344	16.666	22.5	0
52.639	10.863	22.5	0
46.124	7.358	22.5	0

	MRe (%) of MOGP prediction		
	1 Output	2 Outputs	3 Outputs
C_D	2.61256267851122	2.52133663740309	2.18673091768778
C_L	0.930495866281368	0.778855460804079	0.765704503745365

Table 5.4: Mean relative error for the prediction of the lift coefficient and drag coefficients

Part IV

Conclusions and recommendations

6 Conclusion

The purpose of this thesis was to study the use of Gaussian Process regression to predict the aerodynamic performance of hypersonic re-entry vehicles. Reentry aerothermodynamics presents a unique challenge due to the highly nonlinear, multi-regime nature of the flow field, the cost of high-fidelity simulations, and the need for reliable force predictions across a wide parameter space. The primary goal of this work was to develop and evaluate surrogate-based frameworks that reduce computational cost while preserving prediction accuracy in this complex setting. In order to develop this framework, a series of CFD simulations were performed on a capsule and a glider to build an initial database used to train a series of GPs used to predict different aerodynamic coefficients, and their performances were compared. The possibility of expanding databases optimally was addressed by exploring active sampling techniques, mainly uncertainty sampling and committee-based sampling. Finally, the use of multi-output GPs was studied as a possibility to further increase the accuracy of the predictions without adding additional training points.

6.1. Research Questions

The research questions posed in Chapter 1 are repeated below for convenience, which will now be addressed in the full context of the work.

Research Question 1

How accurately can a Gaussian Process regression predict the aerodynamic performance of a reentry vehicle?

Research Question 2

How do the results change for different vehicle architectures? How much does the accuracy change? Does the covariance function and hyper-parameter need to be modified?

Research Question 3

To what degree can the number of simulations can be reduced in order to build a representative aerodynamic database?

The first research question refers to the level of accuracy of these methods. The single-output GPs seem to provide excellent predictions. For a 3-dimensional GP trained with a database generated using LHS, 4-fold cross-validation gives an MRE of 0.37% for the drag and 12% for the lift predictions. However, accuracy is greatly improved when considering only points within a reentry trajectory. Other methods, such as Gaussian Networks [25], are able to predict the aerodynamics of reentry capsules with similar accuracy, but use databases 2 orders of magnitude larger. When considering a 4-dimensional GP trained with a non-optimized database, the MRE of the drag increased to 6.14%, and the lift to 13.51%. Showing that GPs are not extremely sensitive to the dimensionality of the problem. However, in existing literature, GPs are rarely used as single output predictors; variations are often used, such as multi-fidelity [76] [77], multi-output

[26], or paired with reduced order methods [35]. This motivated the further exploration of a multi-output Gaussian process (MOGP). It was found that for most cases, increasing the number of variables also increased the accuracy of the prediction. This is expected from the Linear Model of Coregionalization (LMC) method used, which allows the GP to learn from the relation between the outputs. Unlike most surrogate models, an MOGP presents 3 sources of information: observations, kernel, and the relation between outputs, making it extremely data efficient, as the previous results have shown. This allows for accurate predictions without having to resort to simplified physics or reduced order models [78], which are inherently intrusive. However, in the tests conducted, this improvement in accuracy was not so significant, and the trend of performance was not always clear, meaning that for some cases, adding more outputs was not beneficial. Moreover, it was found that there were more informative outputs than others. This makes sense from a physical perspective, as there are quantities more strongly correlated than others. Only one MOGP architecture was studied through the LMC method. However, there are other, more complex methods that might provide different results. Another advantage of MOGPs is the reduced computational time, compared to training individual GPs for each output [42]. This criterion was not tested in this study, but it is a possible area of study for future work.

Furthermore, the capability of efficiently expanding existing databases was explored through different sampling strategies. It was found that the ability of GPs to inherently provide an uncertainty attached to every prediction could be used to guide the sampling process. By simply sampling the point with the highest uncertainty, the mean uncertainty of the GP was reduced by more than half by adding just 8 additional points.

The second research question regards the applicability of the model to different scenarios. Firstly, for the single-output GPs, it was found that the kernel needed to be modified when predicting different aerodynamic coefficients. However, when comparing the two different vehicle types, it was found that the same kernel could provide good results for both. Still, the best results were obtained when hyperparameters were optimized for each application. It was found that the task of manually tweaking the kernel until obtaining good results was very time-consuming. It is believed that applying automatic kernel searches such as [35] [79] [48] [80] would be beneficial for future work. Furthermore, it was interesting to notice that the optimized hyperparameters for the kernels used in both vehicles showed very similar trends and ranges. This could indicate that the surrogate model is able to learn the underlying physics of reentry vehicles regardless of the architecture being studied. Furthermore, the results in the uncertainties of the predictions seem to be consistent with the physics behind the simulations. Namely, we see that in the capsule, the uncertainty for the lift prediction was significantly higher than for the drag, consistent with the physical model of a blunt reentry body, where lift is entirely a result of flow asymmetry, therefore more unstable. For the Glider, on the other hand, both predictions show a more similar range of uncertainty, consistent with the more stable lift and drag produced by winged reentry configurations. Therefore, we can conclude that analyzing the optimized hyperparameters and uncertainties of a GP is a powerful method to further learn about the underlying physics of the problem at hand.

The third research question addresses the degree to which a database can be reduced and still achieve accurate predictions. This idea was mainly explored through the use of active sampling strategies. Although this study mainly explored how different strategies affected the distribution of the uncertainty in a simplified 2D case, it also showed that the uncertainty sampling is a powerful method to quickly reduce the uncertainty of the model. Mainly, just by adding 8 additional points, the uncertainty of the prediction dropped by 66%. Moreover, even when analyzing the results of the GPs trained with quasi-random sampling distributions, highly accurate predictions are achieved with very few simulations. In the case of the capsule, with 32 points, the predictions of aerodynamic coefficients for points within a re-entry trajectory present an MRE of under 1%. Even when adding an additional dimension to the parameter space, data-efficient methods like the MOGP keep the accuracy comparable, even when keeping the training set at the same size. As seen in the case of the glider, where a database of only 36 points is enough to train an MOGP that presents an MRE of under 3%. For context, works such as [81] perform more than two orders of magnitude more CFD simulations to characterize the thermal loads of a reentry vehicle with a similar parameter space. Nonetheless, in order to fully answer this last research question, future work would be required. Mainly, a more rigorous definition of what constitutes an accurate simulation and a study varying the number of data points to precisely establish when predictions start to become accurate. The next and final chapter addresses further improvements and recommendations that were identified for this work.

7 Recommendations

Building on the findings of this work, the following recommendations are identified for further research and applications.

Multi-Fidelity

Another method that tries to use data as efficiently as possible is the multi-fidelity architecture for Gaussian Processes. This allows the integration of multiple sources of data while taking into account their level of accuracy. In this way, low-fidelity CFD simulation can be integrated alongside higher fidelity ones, further reducing the overall design process. It is believed that if integrated alongside an active sampling and MOGP architecture, the overall method could become more data-efficient.

Active Sampling

A more rigorous test regarding the active sampling strategies would greatly help quantify their benefit over other methods. This would be especially helpful to answer Research Question 3. Therefore, it is recommended to conduct a systematic study where two parallel databases (with the same parameter space) are generated with different strategy and their convergence in generating an accurate prediction is studied, where the accuracy of the prediction is further established through a more robust definition.

Furthermore, implementing an active sampling method that favors points within a realistic trajectory corridor might be beneficial to the efficiency of the data used. Although sampling points in the extremes of the domain improve the global accuracy, in the interest of data efficiency, we also need to consider what the most likely trajectory points are for a reentry vehicle.

Systematic Study of Output Dependencies

The observed benefit of including C_m as a secondary output highlights the importance of understanding cross-output correlations. However, the results presented here do not allow for strong conclusions, and further testing is required. A natural next step would be to repeat the experiments conducted in this work (training GPs with different combinations of outputs) but using a sufficiently large database so that changes in performance can be attributed solely to the number and type of outputs.

It is likely that the database used in this study was too small to accurately assess the effect of the number of outputs. In fact, no strong trend was observed, and results fluctuated greatly. The intention here was to simulate an extremely data-scarce scenario to explore the limits of a GP. Nevertheless, to properly study the interconnections between outputs for a reentry vehicle, additional data points should be included until clear trends emerge and more definitive conclusions can be drawn.

Furthermore, a broader investigation spanning different vehicle typologies and flow regimes would provide stronger guidance for multi-output surrogate design. This would involve applying the same testing methodology to multiple vehicle configurations and considering a range of domains (e.g., subsonic, supersonic, hypersonic). In the present work, constraints led to differing tests and validation criteria for the capsule and the glider. For a more accurate quantification of geometric effects, a more consistent and rigorous testing campaign should be implemented. Moreover, varying the flow regime would also change the underlying physics of the problem, providing an opportunity to test the hypothesis that the benefit of including certain outputs is closely tied to the governing physical phenomena.

Moreover, an aspect that was not addressed in this work was the study of the optimized hyperparameters of the multi-output architecture. It was previously found that for the single-output cases, this was a powerful method to learn about the physics behind the model. Analyzing their value ranges could be a useful indication of whether the model is approximating a physical solution or perhaps overfitting. Given that the results of the MOGP were not very conclusive, this could be addressed in the future.

Uncertainty distribution

The data used to train surrogate models rarely comes without uncertainty by itself. This is especially true when using CFD simulation, which inherently comes with both numerical and physical uncertainties. For example, previous works [22] show that surface heat flux values for a retro-propulsive launcher change greatly depending on the turbulence model used. Ideally, the surrogate model would be able to integrate this uncertainty into the predictions made. This way, a more complete and robust assessment of the problem can be made. The multi-fidelity architecture previously described could be a way to account for this inherent uncertainty.

References

[1] Antoine Patureau de Mirand et al. “Ariane Next, a vision for the next generation of Ariane Launchers”. In: *Acta Astronautica* 170 (2020), pp. 735–749. DOI: https://doi.org/10.1016/j.actaastro.2020.02.003. URL: https://www.sciencedirect.com/science/article/pii/S0094576520300631.

[2] Yi Yin et al. “Modified cross sample entropy and surrogate data analysis method for financial time series”. In: *Physica A: Statistical Mechanics and its Applications* 433 (2015), pp. 17–25. DOI: https://doi.org/10.1016/j.physa.2015.03.055. URL: https://www.sciencedirect.com/science/article/pii/S0378437115003179.

[3] Elçin Içten et al. “Process control of a dropwise additive manufacturing system for pharmaceuticals using polynomial chaos expansion based surrogate model”. In: *Computers Chemical Engineering* 83 (2015). 24th European Symposium on Computer Aided Process Engineering(ESCAPE), pp. 221–231. DOI: https://doi.org/10.1016/j.compchemeng.2015.07.014. URL: https://www.sciencedirect.com/science/article/pii/S0098135415002495.

[4] Kaatje Bollaerts et al. “Thyroid cancer incidence around the Belgian nuclear sites: Surrogate exposure modelling”. In: *Cancer Epidemiology* 39.1 (2015), pp. 48–54. DOI: https://doi.org/10.1016/j.canep.2014.10.011. URL: https://www.sciencedirect.com/science/article/pii/S1877782114001891.

[5] Xin Wu et al. “Optimizing conjunctive use of surface water and groundwater for irrigation to address human-nature water conflicts: A surrogate modeling approach”. In: *Agricultural Water Management* 163 (2016), pp. 380–392. DOI: https://doi.org/10.1016/j.agwat.2015.08.022. URL: https://www.sciencedirect.com/science/article/pii/S037837741530086X.

[6] J.C. Pierson et al. “Evaluating the effectiveness of overstory cover as a surrogate for bird community diversity and population trends”. In: *Ecological Indicators* 61 (2016), pp. 790–798. DOI: https://doi.org/10.1016/j.ecolind.2015.10.031. URL: https://www.sciencedirect.com/science/article/pii/S1470160X15005828.

[7] Carlos Paulete-Perianes et al. “Surrogate modelling for aerodynamic coefficients prediction in aeronautical configurations”. In: *8TH EUROPEAN CONFERENCE FOR AERONAUTICS AND SPACE SCIENCES (EUCASS)* (2019).

[8] D. Viúdez-Moreiras et al. “Surrogate modeling for the main landing gear doors of an airbus passenger aircraft”. In: *Aerospace Science and Technology* 68 (2017), pp. 135–148. DOI: https://doi.org/10.1016/j.ast.2017.04.021. URL: https://www.sciencedirect.com/science/article/pii/S1270963816311063.

[9] ”The Editors of Encyclopaedia Britannica”. “”V-2 rocket””. In: *”Encyclopedia Britannica”* (”2025”). URL: https://www.britannica.com/technology/V-2-rocket.

[10] ”Mark Wade”. “” Von Braun 1952””. In: *”Encyclopedia Astronautica”* (”2019”). URL: http://www.astronautix.com/v/vonbraun1952.html.

[11] ”Kraft Arnold Ehricke”. “”NEXUS concept of a large reusable earth launch vehicle””. In: *”UAH Archives”* (”1963”).

[12] ”NASA”. “”The Space Shuttle””. In: *”NASA Encyclopedia”* (”2023”). URL: https://www.nasa.gov/reference/the-space-shuttle/.

[13] ”Andrew J. Butrica”. “”The Spaceship that Came in From the Cold War””. In: *”Ad Astra 2001 (Volume 13 Number 2)”* (”2001”). URL: https://nss.org/the-spaceship-that-came-in-from-the-cold-war-the-untold-story-of-the-dc-x/.

[14] Delma C Freeman et al. “Reusable launch vehicle technology program11Paper IAF.-96-V.4.01 presented at the 47th International Astronautical Congress, Beijing, China, 7–11 October 1996.” In: *Acta Astronautica* 41.11 (1997), pp. 777–790. DOI: https://doi.org/10.1016/S0094-5765(97)00197-5. URL: https://www.sciencedirect.com/science/article/pii/S0094576597001975.

[15] Sebastian Karl et al. “Sustainable space technologies—Strategies toward a predictive aerothermal design of re-useable space transportation systems”. In: *Review of Scientific Instruments* 95.2 (Feb. 2024), p. 021101. DOI: 10.1063/5.0177075. eprint: https://pubs.aip.org/aip/rsi/article-pdf/doi/10.1063/5.0177075/19591312/021101_1_5.0177075.pdf. URL: https://doi.org/10.1063/5.0177075.

[16] ”Sanford Gordon et al. “”Computer program for calculation of complex chemical equilibrium compositions and applications. Part 1: Analysis””. In: *”NASA Reference Publication”* (”1994”).

[17] V. Emelyanov et al. “Simulation of hypersonic flows with equilibrium chemical reactions on graphics processor units”. In: *Acta Astronautica* 163 (2019). Space Flight Safety-2018, pp. 259–271. DOI: https://doi.org/10.1016/j.actaastro.2019.01.010. URL: https://www.sciencedirect.com/science/article/pii/S0094576518320435.

[18] J. C. Martino et al. *Apollo Wind Tunnel Testing Program – Historical Development of General Configurations and Wind Tunnel Model Test Program*. NASA Technical Note D□3748 NASA□TN□D□3748. Document ID 19670002631. NASA Langley Research Center, 1966.

[19] Sangdi Gu et al. “Capabilities and limitations of existing hypersonic facilities”. In: *Progress in Aerospace Sciences* 113 (2020), p. 100607. DOI: https://doi.org/10.1016/j.paerosci.2020.100607. URL: https://www.sciencedirect.com/science/article/pii/S0376042120300191.

[20] Karen Bibb et al. “Development of the Orion Crew Module Static Aerodynamic Database, Part I: Hypersonic”. In: *29th AIAA Applied Aerodynamics Conference*. DOI: 10.2514/6.2011-3506. eprint: https://arc.aiaa.org/doi/pdf/10.2514/6.2011-3506. URL: https://arc.aiaa.org/doi/abs/10.2514/6.2011-3506.

[21] Ansgar Marwage et al. “Aerodynamic characteristics of the retro propulsion landing burn of vertically landing launchers”. In: *Experiments in Fluids* (2024).

[22] Tobias Ecker et al. “Aerothermal characterization of the CALLISTO vehicle during descent”. In: *CEAS space Journal* (2024).

[23] Thomas J. Wignall et al. “Comparisons of Performance Metrics and Machine Learning Methods on an Entry Descent and Landing Database”. In: *AIAA SCITECH 2023 Forum*. DOI: 10.2514/6.2023-1183. eprint: https://arc.aiaa.org/doi/pdf/10.2514/6.2023-1183. URL: https://arc.aiaa.org/doi/abs/10.2514/6.2023-1183.

[24] James B. Scoggins et al. “Multihierarchy Gaussian Process Models for Probabilistic Aerodynamic Databases using Uncertain Nominal and Off-Nominal Configuration Data”. In: *AIAA SCITECH 2023 Forum*. DOI: 10.2514/6.2023-1185. eprint: https://arc.aiaa.org/doi/pdf/10.2514/6.2023-1185. URL: https://arc.aiaa.org/doi/abs/10.2514/6.2023-1185.

[25] Tenavi Nakamura-Zimmerer et al. “Structured Covariance Gaussian Networks for Orion Crew Module Aerodynamic Uncertainty Quantification”. In: *AIAA SCITECH 2023 Forum*. DOI: 10.2514/6.2023-1184. eprint: https://arc.aiaa.org/doi/pdf/10.2514/6.2023-1184. URL: https://arc.aiaa.org/doi/abs/10.2514/6.2023-1184.

[26] Jens A. Rataczak et al. “Surrogate Models for Hypersonic Aerothermodynamics and Aerodynamics using Gaussian Process Regression”. In: *AIAA SCITECH 2024 Forum*. DOI: 10.2514/6.2024-0461. eprint: https://arc.aiaa.org/doi/pdf/10.2514/6.2024-0461. URL: https://arc.aiaa.org/doi/abs/10.2514/6.2024-0461.

[27] Syed Naeem Ahmed. “9 - Essential statistics for data analysis”. In: *Physics and Engineering of Radiation Detection (Second Edition)*. Ed. by Syed Naeem Ahmed. Second Edition. Elsevier, 2015, pp. 541–593. DOI: https://doi.org/10.1016/B978-0-12-801363-2.00009-7. URL: https://www.sciencedirect.com/science/article/pii/B9780128013632000097.

[28] E. Andrés et al. “Efficient aerodynamic design through evolutionary programming and support vector regression algorithms”. In: *Expert Systems with Applications* 39.12 (2012), pp. 10700–10708. DOI: https://doi.org/10.1016/j.eswa.2012.02.197. URL: https://www.sciencedirect.com/science/article/pii/S0957417412004708.

[29] Amir Teimourian et al. “Airfoil aerodynamic performance prediction using machine learning and surrogate modeling”. In: *Heliyon* 10.8 (2024), e29377. DOI: https://doi.org/10.1016/j.heliyon.2024.e29377. URL: https://www.sciencedirect.com/science/article/pii/S2405844024054082.

[30] J.-C. Jouhaud et al. “A surrogate-model based multidisciplinary shape optimization method with application to a 2D subsonic airfoil”. In: *Computers Fluids* 36.3 (2007), pp. 520–529. DOI: https://doi.org/10.1016/j.compfluid.2006.04.001. URL: https://www.sciencedirect.com/science/article/pii/S0045793006000582.

[31] X. Liu et al. “Modeling Multiresponse Surfaces for Airfoil Design with Multiple-Output-Gaussian-Process Regression”. In: *Journal of Aircraft* 51.3 (2014), pp. 740–747. DOI: 10.2514/1.C032465. eprint: https://doi.org/10.2514/1.C032465. URL: https://doi.org/10.2514/1.C032465.

[32] Shinkyu Jeong et al. “Efficient Optimization Design Method Using Kriging Model”. In: *Journal of Aircraft* 42.2 (2005), pp. 413–420. DOI: 10.2514/1.6386. eprint: https://doi.org/10.2514/1.6386. URL: https://doi.org/10.2514/1.6386.

[33] Fei Liu et al. “Surrogate-based aerodynamic shape optimization of hypersonic flows considering transonic performance”. In: *Aerospace Science and Technology* 93 (2019), p. 105345. DOI: https://doi.org/10.1016/j.ast.2019.105345. URL: https://www.sciencedirect.com/science/article/pii/S1270963819308983.

[34] Oscar Ulises Espinosa Barcenas et al. “Surrogate Aerodynamic Wing Modeling Based on a Multilayer Perceptron”. In: *Aerospace* 10.2 (2023). DOI: 10.3390/aerospace10020149. URL: https://www.mdpi.com/2226-4310/10/2/149.

[35] Yuxin Yang et al. “Fast flow field prediction of three-dimensional hypersonic vehicles using an improved Gaussian process regression algorithm”. In: *Physics of Fluids* 36.1 (Jan. 2024), p. 016129. DOI: 10.1063/5.0183291. eprint: https://pubs.aip.org/aip/pof/article-pdf/doi/10.1063/5.0183291/18504954/016129_1_5.0183291.pdf. URL: https://doi.org/10.1063/5.0183291.

[36] Xinyu Hui et al. “Fast pressure distribution prediction of airfoils using deep learning”. In: *Aerospace Science and Technology* 105 (2020), p. 105949. DOI: https://doi.org/10.1016/j.ast.2020.105949. URL: https://www.sciencedirect.com/science/article/pii/S1270963820306313.

[37] Janghoon Seo et al. “Establishment of CNN and Encoder–Decoder Models for the Prediction of Characteristics of Flow and Heat Transfer around NACA Sections”. In: *Energies* 15.23 (2022). DOI: 10.3390/en15239204. URL: https://www.mdpi.com/1996-1073/15/23/9204.

[38] Xu (□□) Wang et al. “Unsteady aerodynamic prediction for iced airfoil based on multi-task learning”. In: *Physics of Fluids* 34.8 (Aug. 2022), p. 087117. DOI: 10.1063/5.0101991. eprint: https://pubs.aip.org/aip/pof/article-pdf/doi/10.1063/5.0101991/16589830/087117_1_online.pdf. URL: https://doi.org/10.1063/5.0101991.

[39] Víctor Francés-Belda et al. “Toward aerodynamic surrogate modeling based on β-variational autoencoders”. In: *Physics of Fluids* 36.11 (Nov. 2024), p. 117139. DOI: 10.1063/5.0232644. eprint: https://pubs.aip.org/aip/pof/article-pdf/doi/10.1063/5.0232644/20242978/117139_1_5.0232644.pdf. URL: https://doi.org/10.1063/5.0232644.

[40] Alberto Solera-Rico et al. *β-Variational autoencoders and transformers for reduced-order modelling of fluid flows*. 2023. arXiv: 2304.03571 [physics.flu-dyn]. URL: https://arxiv.org/abs/2304.03571.

[41] Mauricio A. Álvarez et al. “Computationally Efficient Convolved Multiple Output Gaussian Processes”. In: *Journal of Machine Learning Research* 12.41 (2011), pp. 1459–1500. URL: http://jmlr.org/papers/v12/alvarez11a.html.

[42] Mark van der Wilk et al. *A Framework for Interdomain and Multioutput Gaussian Processes*. 2020. arXiv: 2003.01115 [stat.ML]. URL: https://arxiv.org/abs/2003.01115.

[43] Andreas Damianou. *Gaussian process introductory tutorial in Python*. http://adamian.github.io/talks/Damianou_GP_tutorial.html. URL: http://adamian.github.io/talks/Damianou_GP_tutorial.html.

[44] Yuge Shi. “Gaussian Processes, not quite for dummies”. In: *The Gradient* (2019).

[45] Charles A. Micchelli et al. “Universal Kernels”. In: *Journal of Machine Learning Research* 7.95 (2006), pp. 2651–2667. URL: http://jmlr.org/papers/v7/micchelli06a.html.

[46] ”David Kristjanson Duvenaud”. “”Automatic Model Construction with Gaussian Processes””. PhD thesis. ”University of Cambridge”, ”2014”.

[47] Carl Ramussen et al. *Gaussian Processes for Machine Learning*. The MIT Presss, 2006.

[48] Shifan Zhao et al. *Efficient Two-Stage Gaussian Process Regression Via Automatic Kernel Search and Subsampling*. 2024. arXiv: 2405.13785 [cs.LG]. URL: https://arxiv.org/abs/2405.13785.

[49] Hachem Kadri et al. *Operator-valued Kernels for Learning from Functional Response Data*. 2016. arXiv: 1510.08231 [cs.LG]. URL: https://arxiv.org/abs/1510.08231.

[50] Michalis Titsias. “Variational Learning of Inducing Variables in Sparse Gaussian Processes”. In: *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics*. Ed. by David van Dyk et al. Vol. 5. Proceedings of Machine Learning Research. Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA: PMLR, 16–18 Apr 2009, pp. 567–574. URL: https://proceedings.mlr.press/v5/titsias09a.html.

[51] Ruichen Jin et al. “Comparative studies of metamodeling techniques under multiple modeling criteria”. In: *8th Symposium on Multidisciplinary Analysis and Optimization*. DOI: 10.2514/6.2000-4801. eprint: https://arc.aiaa.org/doi/pdf/10.2514/6.2000-4801. URL: https://arc.aiaa.org/doi/abs/10.2514/6.2000-4801.

[52] C. Devon Lin et al. *Latin Hypercubes and Space-filling Designs*. 2022. arXiv: 2203.06334 [stat.ME]. URL: https://arxiv.org/abs/2203.06334.

[53] Jian Cheng et al. *Computational Investigation of Low-Discrepancy Sequences in Simulation Algorithms for Bayesian Networks*. 2013. arXiv: 1301.3841 [cs.AI]. URL: https://arxiv.org/abs/1301.3841.

[54] David Cohn et al. “Improving Generalization with Active Learning”. In: *Mach. Learn.* 15.2 (May 1994), pp. 201–221. DOI: 10.1023/A:1022673506211. URL: https://doi.org/10.1023/A:1022673506211.

[55] D. A. Cohn et al. *Active Learning with Statistical Models*. 1996. arXiv: cs/9603104 [cs.AI]. URL: https://arxiv.org/abs/cs/9603104.

[56] Hongri Cong et al. “A Novel Gaussian Process Surrogate Model with Expected Prediction Error for Optimization under Constraints”. In: *Mathematics* 12.7 (2024). DOI: 10.3390/math12071115. URL: https://www.mdpi.com/2227-7390/12/7/1115.

[57] ”Yoav Freund et al. “Selective Sampling Using the Query by Committee Algorithm”. ”English (US)”. In: *”Machine Learning”* ”28”.”2-3” (”1997”), ”133–168”. DOI: "10.1023/a:1007330508534".

[58] Mariasole Laureti et al. “Aerothermal analysis of Themis T3: influence of nozzle cluster geometry on the base heating and heat loads on the grid fins”. In: *3rd International Conference on Flight Vehicles, Aerothermodynamics and Re-entry FAR 2025*. Mai 2025. URL: https://elib.dlr.de/213810/.

[59] Roop N. Gupta et al. *A Review of Reaction Rates and Thermodynamic and Transport Properties for an 11-Species Air Model for Chemical and Thermal Nonequilibrium Calculations to 30,000 K*. NASA Reference Publication NASA RP-1232. Washington, D.C.: NASA, 1990. URL: https://ntrs.nasa.gov/api/citations/19900017748/downloads/19900017748.pdf.

[60] Sebastian Karl. “Numerical Investigation of a Generic Scramjet Configuration”. PhD thesis. Technische Universität Dresden, Dresden, 2011.

[61] David Olynick et al. “Aerothermodynamics of the Stardust Sample Return Capsule”. In: *Journal of Spacecraft and Rockets* 36.3 (1999), pp. 442–462. DOI: 10.2514/2.3466. eprint: https://doi.org/10.2514/2.3466. URL: https://doi.org/10.2514/2.3466.

[62] R. Mitcheltree et al. “Aerodynamics of stardust sample return capsule”. In: *15th Applied Aerodynamics Conference* (1997). DOI: 10.2514/6.1997-2304.

[63] Thomas Gawehn et al. “Impact of Shape Change on Capsule Aerodynamics and Shock-wave Boundary Layer Interaction”. In: *2nd International Conference on Flight Vehicles, Aerothermodynamics and Re-entry Missions & Engineering (FAR)*. June 2022. URL: https://elib.dlr.de/187173/.

[64] Hirotaka Otsu. “Aerodynamic Characteristics of Re-Entry Capsules with Hyperbolic Contours”. en. In: *Aerospace* 8.10 (Oct. 2021), p. 287. DOI: 10.3390/aerospace8100287. URL: http://dx.doi.org/10.3390/aerospace8100287.

[65] Peter A. Gnoffo et al. “Prediction and Validation of Mars Pathfinder Hypersonic Aerodynamic Database”. In: *Journal of Spacecraft and Rockets* 36.3 (1999), pp. 367–373. DOI: 10.2514/2.3455. eprint: https://doi.org/10.2514/2.3455. URL: https://doi.org/10.2514/2.3455.

[66] Karl Edquist. “Computations of Viking Lander Capsule Hypersonic Aerodynamics with Comparisons to Ground and Flight Data”. In: *AIAA Atmospheric Flight Mechanics Conference and Exhibit*. DOI: 10.2514/6.2006-6137. eprint: https://arc.aiaa.org/doi/pdf/10.2514/6.2006-6137. URL: https://arc.aiaa.org/doi/abs/10.2514/6.2006-6137.

[67] Alexander Forrester et al. *Engineering design via surrogate modelling: a practical guide*. John Wiley & Sons, 2008.

[68] C.M. Bishop. *Pattern recognition and machine learning*. Vol. 4. Springer New York, 2006. URL: http://scholar.google.com/scholar.bib?q=info:jYxggZ6Ag1YJ:scholar.google.com/&output=citation&hl=en&as_sdt=0,5&as_vis=1&ct=citation&cd=0.

[69] Nestor V. Queipo et al. “Surrogate-based analysis and optimization”. In: *Progress in Aerospace Sciences* 41.1 (2005), pp. 1–28. DOI: https://doi.org/10.1016/j.paerosci.2005.02.001. URL: https://www.sciencedirect.com/science/article/pii/S0376042105000102.

[70] Romain Dupuis et al. “Surrogate Modeling of Aerodynamic Simulations for Multiple Operating Conditions Using Machine Learning”. In: *AIAA Journal* 56.9 (Sept. 2018), pp. 3622–3635. DOI: 10.2514/1.j056405. URL: http://dx.doi.org/10.2514/1.J056405.

[71] Hideitsu Hino et al. *Active Learning by Query by Committee with Robust Divergences*. 2022. arXiv: 2211.10013 [stat.ML]. URL: https://arxiv.org/abs/2211.10013.

[72] Mark Loomis et al. “Aeroheating and aerodynamic CFD validation and prediction for the X-38 program”. In: *32nd Thermophysics Conference*. DOI: 10.2514/6.1997-2478. eprint: https://arc.aiaa.org/doi/pdf/10.2514/6.1997-2478. URL: https://arc.aiaa.org/doi/abs/10.2514/6.1997-2478.

[73] Charles Campbell et al. “An overview of X-38 hypersonic aerothermodynamic wind tunnel data and comparison with numerical results”. In: *32nd Thermophysics Conference*. DOI: 10.2514/6.1997-2475. eprint: https://arc.aiaa.org/doi/pdf/10.2514/6.1997-2475. URL: https://arc.aiaa.org/doi/abs/10.2514/6.1997-2475.

[74] Jaehyun Lim et al. “Multi-output Infinite Horizon Gaussian Processes”. In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. Xi’an, China: IEEE Press, 2021, pp. 1542–1549. DOI: 10.1109/ICRA48506.2021.9561031. URL: https://doi.org/10.1109/ICRA48506.2021.9561031.

[75] Benjamin Peherstorfer et al. “Survey of Multifidelity Methods in Uncertainty Propagation, Inference, and Optimization”. In: *SIAM Review* 60.3 (2018), pp. 550–591. DOI: 10.1137/16M1082469. eprint: https://doi.org/10.1137/16M1082469. URL: https://doi.org/10.1137/16M1082469.

[76] Javier Nieto-Centenero et al. “FUSING AERODYNAMIC DATA USING MULTI-FIDELITY GAUSSIAN PROCESS REGRESSION”. In: *15th ECCOMAS Thematic Conference on Evolutionary and Deterministic Methods for Design, Optimization and Control* (2023).

[77] Loïc Brevault et al. “Overview of Gaussian process based multi-fidelity techniques with variable relationship between fidelities, application to aerospace systems”. In: *Aerospace Science and Technology* 107 (2020), p. 106339. DOI: `https://doi.org/10.1016/j.ast.2020.106339`. URL: `https://www.sciencedirect.com/science/article/pii/S127096382031021X`.

[78] Steven Snyder et al. “AeroFusion: Data Fusion and Uncertainty Quantification for Entry Vehicles”. In: *AIAA SCITECH 2023 Forum*. DOI: `10.2514/6.2023-1182`. eprint: `https://arc.aiaa.org/doi/pdf/10.2514/6.2023-1182`. URL: `https://arc.aiaa.org/doi/abs/10.2514/6.2023-1182`.

[79] Yuxin (□□□) Yang et al. “Improved automatic kernel construction for Gaussian process regression in small sample learning for predicting lift body aerodynamic performance”. In: *Physics of Fluids* 35.6 (June 2023), p. 066108. DOI: `10.1063/5.0153970`. eprint: `https://pubs.aip.org/aip/pof/article-pdf/doi/10.1063/5.0153970/17936895/066108\_1\_5.0153970.pdf`. URL: `https://doi.org/10.1063/5.0153970`.

[80] Huang Zhang et al. *A Practitioner’s Guide to Automatic Kernel Search for Gaussian Processes in Battery Applications*. 2025. arXiv: `2505.01674 [eess.SY]`. URL: `https://arxiv.org/abs/2505.01674`.

[81] Tobias Ecker et al. “Aerothermal characterization of the CALLISTO vehicle during descent”. In: *9 th EUROPEAN CONFERENCE FOR AERONAUTICS AND AEROSPACE SCIENCES (EUCASS)*. June 2022. URL: `https://elib.dlr.de/187034/`.

A

Capsule GP results

Table A.1: Result comparison for AoA = 0°

Altitude (Km)	C_D (TAU)	C_D (GP)	ReE (%)
50.63	1.429693321	1.43697847	0.509560277
46.54	1.451131553	1.45601983	0.336859678
45.75	1.468041281	1.46541639	0.178802267
44.44	1.492813759	1.49359469	0.052312689
43.24	1.490036426	1.498326	0.556333671
41.6	1.482112617	1.48167965	0.029212844

Table A.2: Result comparison for AoA = 5°

Altitude (Km)	C_D (TAU)	C_D (GP)	ReE (%)
50.63	1.417086969	1.41974626	0.187659001
46.54	1.436833228	1.43857666	0.121338504
45.75	1.449417087	1.45234744	0.202174602
44.44	1.457785307	1.45992471	0.146757084
43.24	1.453348494	1.45892993	0.384039778
41.6	1.455408882	1.45378201	0.111781087

Table A.3: Result comparison for AoA = 10°

Altitude (Km)	C_D (TAU)	C_D (GP)	ReE (%)
50.63	1.381167864	1.38217678	0.073048029
46.54	1.384621853	1.38294611	0.121025354
45.75	1.386727219	1.38407047	0.191584138
44.44	1.387982487	1.3855542	0.174950861
43.24	1.38075287	1.38390539	0.228318881
41.6	1.390032929	1.39015785	0.008986874

Table A.4: Result comparison for AoA = $5°$

Altitude (Km)	C_L (TAU)	C_L (GP)	ReE (%)
50.63	0.0459629099134	0.04574092	0.482976194976031
46.54	0.045308525746	0.04620134	1.97052152834353
45.75	0.0421999655346	0.04029293	4.51904524195977
44.44	0.0322213888082	0.0323524	0.406596973767517
43.24	0.024651580186	0.02537679	2.94183905667783
41.6	0.0206746583752	0.02061708	0.278497347598592

Table A.5: Result comparison for AoA = $10°$

Altitude (Km)	C_L (TAU)	C_L (GP)	ReE (%)
50.63	0.0881779269708	0.08553221	3.00042999613285
46.54	0.0816590789104	0.08006123	1.95673148867285
45.75	0.0740585461176	0.07203163	2.73691049022404
44.44	0.0666683198774	0.06601016	0.987215334975189
43.24	0.0618522611328	0.06024231	2.60289778144626
41.6	0.0562099263336	0.05305037	5.62099354987297

B

X-38 Trajectory data

Table B.1: X-38 Trajectory data

t (s)	H(km)	Pdyn (kPa)	Flux SP (kW/m2)	Mach	Vrel (m/s)	AoA (°)	rho (kg/m3)	T (K)	p (Pa)
720.000	70.157	1.556	386.383	20.875	6195.976	45.000	8.11E-05	219.179	5.1005E+00
725.000	69.973	1.582	386.278	20.764	6169.253	45.000	8.31E-05	219.617	5.24E+00
730.000	69.783	1.609	386.172	20.652	6142.083	45.000	8.53E-05	220.070	5.39E+00
735.000	69.589	1.638	386.067	20.537	6114.446	45.000	8.76E-05	220.539	5.55E+00
740.000	69.389	1.667	385.966	20.420	6086.325	45.000	9.00E-05	221.023	5.71E+00
745.000	69.184	1.698	385.868	20.301	6057.699	45.000	9.26E-05	221.530	5.89E+00
750.000	68.973	1.731	385.776	20.179	6028.545	45.000	9.52E-05	222.047	6.07E+00
755.000	68.757	1.764	385.687	20.055	5998.840	45.000	9.81E-05	222.594	6.26637E+00
760.000	68.534	1.800	385.600	19.929	5968.558	45.000	1.01E-04	223.156	6.47E+00
765.000	68.305	1.837	385.519	19.800	5937.673	45.000	1.04E-04	223.749	6.69E+00
770.000	68.069	1.875	385.452	19.668	5906.155	45.000	1.08E-04	224.358	6.93E+00
775.000	67.827	1.916	385.386	19.533	5873.972	45.000	1.11E-04	224.998	7.17E+00
780.000	67.577	1.959	385.308	19.395	5841.092	45.000	1.15E-04	225.655	7.44E+00
785.000	67.320	2.004	385.249	19.255	5807.478	45.000	1.19E-04	226.329	7.72E+00
790.000	67.056	2.052	385.183	19.110	5773.092	45.000	1.23E-04	227.047	8.02E+00
795.000	66.783	2.102	385.119	18.963	5737.891	45.000	1.28E-04	227.784	8.35E+00
800.000	66.502	2.155	385.054	18.812	5701.832	45.000	1.33E-04	228.550	8.70E+00
805.000	66.211	2.211	384.986	18.658	5664.864	45.000	1.38E-04	229.349	9.07E+00
810.000	65.910	2.270	384.916	18.499	5626.934	45.000	1.43E-04	230.179	9.48E+00
815.000	65.600	2.333	384.848	18.337	5587.984	45.000	1.49E-04	231.041	9.91E+00
820.000	65.278	2.400	384.770	18.170	5547.949	45.000	1.56E-04	231.948	1.04E+01
825.000	64.944	2.472	384.678	17.999	5506.763	45.000	1.63E-04	232.882	1.09E+01
830.000	64.597	2.548	384.593	17.823	5464.345	45.000	1.71E-04	233.867	1.15E+01
835.000	64.236	2.630	384.497	17.641	5420.611	45.000	1.79E-04	234.900	1.21E+01
840.000	63.861	2.718	384.354	17.455	5375.467	45.000	1.88E-04	235.970	1.27E+01
845.000	63.483	2.807	383.871	17.263	5328.827	45.000	1.98E-04	237.056	1.35E+01
850.000	63.104	2.897	382.956	17.068	5280.671	45.000	2.08E-04	238.148	1.42E+01
855.000	62.724	2.987	381.602	16.869	5230.992	45.000	2.18E-04	239.240	1.50E+01
860.000	62.344	3.078	379.800	16.666	5179.787	45.000	2.29E-04	240.328	1.58299E+01
865.000	61.962	3.168	377.542	16.459	5127.055	45.000	2.41E-04	241.423	1.67E+01
870.000	61.580	3.259	374.824	16.248	5072.802	45.000	2.53E-04	242.512	1.76E+01
875.000	61.196	3.348	371.645	16.033	5017.035	45.000	2.66E-04	243.607	1.86E+01
880.000	60.811	3.437	368.004	15.815	4959.768	45.000	2.79E-04	244.706	1.96E+01
885.000	60.424	3.526	363.912	15.593	4901.016	45.000	2.94E-04	245.793	2.07E+01
890.000	60.035	3.612	359.373	15.367	4840.802	45.000	3.08E-04	246.882	2.19E+01
895.000	59.644	3.698	354.393	15.138	4779.151	45.000	3.24E-04	247.972	2.31E+01
900.000	59.251	3.782	348.999	14.906	4716.088	45.000	3.40E-04	249.047	2.43E+01
905.000	58.855	3.864	343.194	14.670	4651.645	45.000	3.57E-04	250.136	2.56E+01
910.000	58.457	3.944	337.009	14.431	4585.854	45.000	3.75E-04	251.225	2.71E+01
915.000	58.056	4.023	330.452	14.190	4518.748	45.000	3.94E-04	252.311	2.85E+01
920.000	57.652	4.099	323.545	13.944	4450.369	45.000	4.14E-04	253.411	3.01E+01
925.000	57.254	4.168	316.121	13.697	4380.750	45.000	4.34E-04	254.498	3.17E+01
930.000	56.905	4.208	307.491	13.451	4310.145	45.000	4.53E-04	255.448	3.3224E+01
935.000	56.541	4.252	298.964	13.203	4238.925	45.000	4.73E-04	256.446	3.48E+01
940.000	56.154	4.303	290.636	12.952	4166.981	45.000	4.96E-04	257.514	3.66E+01
945.000	55.750	4.359	282.382	12.698	4094.208	45.000	5.20E-04	258.633	3.86E+01
950.000	55.331	4.416	274.106	12.442	4020.546	45.000	5.46E-04	259.791	4.08E+01
955.000	54.901	4.474	265.756	12.183	3945.972	45.000	5.75E-04	261.002	4.31E+01
960.000	54.462	4.531	257.310	11.921	3870.492	45.000	6.05E-04	262.261	4.55E+01
965.000	54.015	4.587	248.768	11.658	3794.125	45.000	6.37E-04	263.523	4.82E+01
970.000	53.562	4.640	240.134	11.393	3716.898	45.000	6.72E-04	264.787	5.11E+01
975.000	53.103	4.692	231.421	11.128	3638.849	45.000	7.09E-04	266.026	5.41E+01
980.000	52.639	4.740	222.639	10.863	3560.023	45.000	7.48E-04	267.212	5.73821E+01
985.000	52.172	4.785	213.806	10.598	3480.474	45.000	7.90E-04	268.316	6.09E+01
990.000	51.702	4.826	204.940	10.334	3400.264	45.000	8.35E-04	269.332	6.46E+01
995.000	51.229	4.863	196.066	10.072	3319.460	45.000	8.83E-04	270.223	6.85E+01
1000.000	50.753	4.896	187.196	9.812	3238.106	45.000	9.34E-04	270.971	7.26E+01
1005.000	50.278	4.922	178.329	9.554	3156.203	45.000	9.88E-04	271.544	7.70E+01
1010.000	49.802	4.944	169.520	9.297	3073.843	45.000	1.05E-03	271.949	8.17E+01
1015.000	49.326	4.962	160.802	9.044	2991.101	45.000	1.11E-03	272.136	8.67E+01
1020.000	48.850	4.976	152.213	8.793	2908.041	45.000	1.18E-03	272.124	9.19E+01
1025.000	48.408	4.965	143.463	8.545	2824.736	45.000	1.24E-03	271.901	9.7154E+01
1030.000	47.992	4.934	134.736	8.300	2741.745	45.000	1.31E-03	271.508	1.02E+02
1035.000	47.552	4.917	126.521	8.059	2659.217	45.000	1.39E-03	270.868	1.08E+02
1040.000	47.091	4.908	118.712	7.823	2576.952	45.000	1.48E-03	269.979	1.15E+02
1045.000	46.614	4.905	111.229	7.590	2494.818	45.000	1.58E-03	268.832	1.22E+02
1050.000	46.124	4.903	104.013	7.358	2412.755	45.000	1.68E-03	267.485	1.2937E+02
1055.000	45.623	4.901	97.044	7.128	2330.758	45.000	1.80E-03	265.996	1.38E+02
1060.000	45.109	4.898	90.309	6.898	2248.859	45.000	1.94E-03	264.446	1.47E+02
1065.000	44.582	4.892	83.816	6.667	2167.105	45.000	2.08E-03	262.890	1.57E+02
1070.000	44.039	4.886	77.577	6.435	2085.532	45.000	2.25E-03	261.350	1.69E+02
1075.000	43.479	4.880	71.594	6.202	2004.168	45.000	2.43E-03	259.812	1.81237E+02