POLITECNICO DI TORINO

Master's Degree in Aerospace Engineering

Master's Degree Thesis

Multi-fidelity Analysis and Optimization of a Cold-Gas Thruster

Supervisors

Prof. Andrea FERRERO

Ing. Leonardo STUMPO

Candidate

Federico VIGNANI

July 2024

Acknowledgements

Desidero esprimere la mia più sincera gratitudine al Professor Andrea Ferrero e all'Ingegner Leonardo Stumpo per avermi concesso l'opportunità di svolgere questa tesi di laurea magistrale. La loro guida e il loro supporto sono stati cruciali nei momenti più difficili della stesura dell'elaborato e grazie al loro costante incoraggiamento e alle preziose indicazioni sono riuscito a superare le difficoltà incontrate lungo il percorso.

Un ringraziamento speciale va inoltre a tutte le persone che mi hanno accompagnato durante questi anni al Politecnico, sia nella quotidianità della vita universitaria, sia al di fuori di essa. Il loro sostegno, la loro amicizia e i momenti condivisi hanno reso questo viaggio un'esperienza indimenticabile.

Summary

In the study of the response of complex engineering systems, especially in the aerospace field, computer experiments play a crucial role, enabling the simulation of the behaviour of such systems through the implementation of mathematical models, that can be extremely complex. For instance, optimization problems involving aerodynamic surfaces, turbomachinery components, as well as structural and thermo-fluid dynamic assessments that arise in various phases of aerospace design, require numerical simulations that are often extremely demanding in terms of simulation time and computational costs, when a high level of accuracy is sought. In these cases, the use of High Fidelity models, which aim for the most accurate representation possible of the phenomenon, often becomes infeasible. On the other hand, transitioning to Low Fidelity models allows for significant speed-up and reduction of computational burdens, but results in lower accuracy outputs. Multi-fidelity techniques enable the combination of the Low Fidelity models' ability to provide numerous low-cost insights with the High Fidelity models' capacity to ensure high result accuracy, allowing for a reduction in the required simulations. Moreover, they lead to the creation of surrogate models to use in outer loop applications, such as optimization tasks. A multi-fidelity modeling technique that has gained prominence in the aeronautical field in recent years is Co-Kriging. Rooted in geostatistics, Co-Kriging is a spatial interpolation technique for predicting the behaviour of a variable or function in untested locations.

This thesis implements a detailed study of the Co-Kriging technique for the development of a surrogate model to predict the thrust coefficient of a cold gas thruster, as the geometry of its exhaust nozzle varies. Cold gas thrusters are small propulsion system used for attitude control maneuvers of nanosatellites that rely on the simple expansion of pressurized gas in a nozzle, without combustion. For these systems, viscosity significantly impacts the performance in terms of achievable thrust and the boundary layer is found to occupy a significant portion of the exit area. For various geometrical configurations of the nozzle, the thrust coefficient is calculated using both a High Fidelity model, which involves solving the Navier-Stokes equations to study the flow, and a Low Fidelity one, where viscous effects are neglected,

requiring only the Euler equations. Additionally, the use of a 1D ideal nozzle model is considered as a potential Low Fidelity approach. Adaptive sampling techniques for the implementation of the Co-Kriging surrogate models are explored, testing three different algorithms, namely Maximum Variance, Expected Improvement and Information Gain, on benchmark single and two-variable functions. The last two algorithms are applied to the cold gas thruster case and the Information Gain criterion is found to outperform the other one. Moreover, leveraging the 1D ideal nozzle model as Low Fidelity, led to a reduction in overall computational cost while ensuring high global accuracy, despite the potential for minor local inaccuracies depending on the sampled points' locations. Once a Co-Kriging surrogate is obtained, a multi-objective optimization is carried on with the aim of maximizing C_F while minimizing the nozzle's mass. A Pareto front for the optimal points is obtained, allowing for the selection of the geometry that best meets the mission objectives during the cold gas thruster's design phase.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Computer-based models and their fidelity

In the framework of the study of complex engineering systems, with particular regard to fluid dynamics and aerodynamics problems, the challenge of selecting an experimental approach often arises, mainly due to the fact that testing facilities and highly expensive sensing equipment are required, not to mention the difficulty of varying geometry, boundary and initial conditions of the experiment. As a result, computer-based simulations have assumed a crucial role in the study of physical systems [1]. In order to simulate the behaviours of theoretical physical systems, engineers must develop mathematical models, which are deterministic and may often be highly complex, encompassing various levels of fidelity and a large number of design variables. These models usually involve sets of equations, that can be linear and nonlinear algebraic as well as ordinary and partial differential equations, depending on the considered phenomena [1]. That being said, the aim of reaching a comparable level of accuracy with respect to physical testing data entails computationally expensive analysis and simulation processes that are time-consuming and might be infeasible [2].

As previously mentioned, computer-based models can be distinguished by different levels of fidelity: for clarity, as stated by Fernandez-Godino [3], "fidelity in modeling refers to the level of detail and accuracy provided by a predictive model or simulation". High Fidelity models (HFMs), which can be described as complex high-dimensional systems that are designed to provide highly accurate results in predictions and simulations, are often impractical to develop and utilize if the computational resources are limited [3]. Although in recent years the availability of fast computing platforms have increased, if the evaluation of the High Fidelity objective function requires the numerical solution of partial differential equations

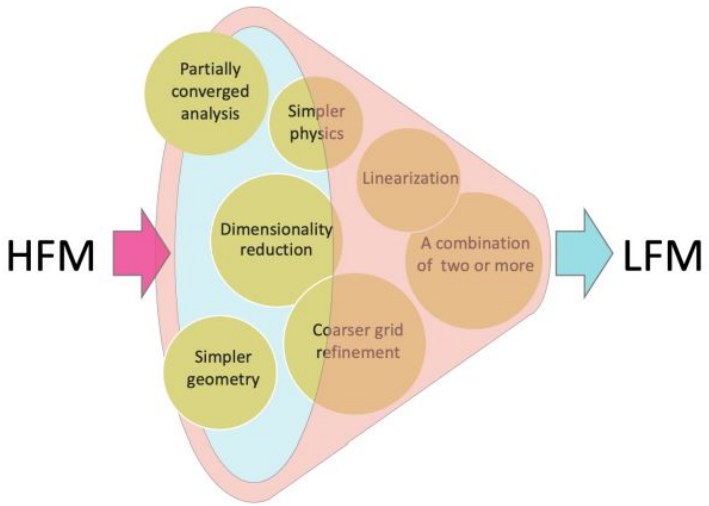

Figure 1.1: Connection between high-fidelity and low-fidelity models is commonly attributed to one or more of the following factors: dimensionality reduction, grid coarsening, linearization, partial convergence, reduced geometry complexity, and simplified physics [3]

(PDE), i.e. Navier-Stokes equations, even a single evaluation might take hours on a powerful computer or even more on a desktop architecture [4]. Moreover, complex models with higher fidelity tend to be characterized by a large number of design variables, i.e. the input parameters of the model, therefore one can verify what Forrester [5] calls the *curse of dimensionality*: the greater the number of design variables, the larger the number of samples needed to evaluate the objective function in order to ensure a certain degree of accuracy in the model prediction. Thus, an exponential relation exists between the number of samples required and the number of variables, making High Fidelity simulations too expensive in terms of computational cost and time [5].
Compared to High Fidelity simulations, which might often be feasible only for a few measuring locations, Low Fidelity models (LFMs) result in a simpler and more affordable outcome; obviously, the advantages in terms of computational and time burden come at the expense of the model's prediction accuracy. As shown in Figure 1.1, Low Fidelity models can generally be derived from High Fidelity models through dimensionality reduction, linearization, use of simplified physical models, simulations performed on coarser grids or partially converged results [3]. It is easy to infer that a model can be defined as High or Low Fidelity only in relation to another: for instance, a three-dimensional simulation holds a higher fidelity than a mere analytical function evaluation, but can be considered less accurate with respect to experimental measurements [3].

In general, it is conceivable that Low Fidelity simulations can be executed within reasonable time frames and with modest computational overheads, within the capabilities of a standard desktop architecture.

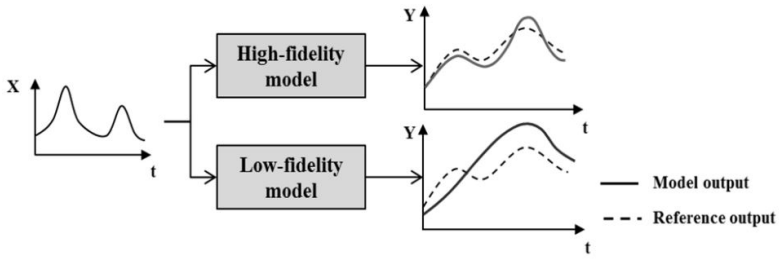

Figure 1.2: Fidelity of models [6]

For instance, considering a specific objective function that describes the response of the Low Fidelity model to a set of inputs, it would be feasible to perform a substantial number of evaluations of said function across various locations in the domain of interest, without significant difficulty. Despite this allows a rapid acquisition of a broad spectrum of information regarding the model output, the major drawback of employing a Low Fidelity model is that the results lack accuracy and deviate from the true behaviour of the objective function or, in any case, that described by a higher-fidelity model. On the other hand, evaluations of High Fidelity objective functions are constrained by computational resource availability, typically yielding precise yet limited information at only a few locations within the domain of interest. Ideally, one would like to have access to a model capable of retaining the accuracy that characterizes High Fidelity models while circumventing the issue of unsustainable computational burden, thereby combining the merits of HFMs and LFMs and overcoming their drawbacks. This goal can be achieved by suitably combining data obtained through simulations of different fidelities, giving rise to a variable-fidelity model, more commonly known as a multi-fidelity model (MFM) [3]. Several works and studies since the early 2000s have demonstrated how the combined use of different fidelities enables the speeding of simulations without limitations in terms of accuracy loss [3]. The basic idea was already proposed by Toropov et al. in [7], namely to leverage the utilization of Low Fidelity models as a basis for rapidly and effectively constructing approximations of High Fidelity models. Indeed, multi-fidelity models exploit the large number of information provided at low cost by LFMs in order to capture a global trend of the objective function, thus allowing for an initial understanding of the general behaviour of the expensive model output [1]. Subsequently, a few pieces of information derived from a small sized sample from High Fidelity simulations allows a calibration of the cheap objective function, in order to obtain a highly accurate approximation of the expensive model's response [1].

An important concept to emphasize is that, typically, the adoption of multi-fidelity modeling techniques leads to the creation of surrogate models (SMs), often referred to in the literature as metamodels and introduced in the late 1980s by the works of Box and Draper [8] and Kleijnen [9]. A surrogate model can be defined as a mathematical approximation that reproduces the behaviour of a computationally expensive system or model across the entire domain of interest or regions thereof [10]. The significant advantage of these models lies in their being "cheap to evaluate" [11]. In fact, evaluating a surrogate model at a particular point or set of points requires minimal computational resources and can be practically instantaneous. Therefore, once constructed and trained, the surrogate model is used to replace the expensive one, allowing the prediction of the objective function values at points in the domain that are not part of the sampling plans used for the model's own training [11]. By training a surrogate model with data belonging to different levels of fidelity, a multi-fidelity surrogate model (MFSM) is obtained [3].
In recent years, the use of surrogate models in the context of multi-fidelity modeling has been the subject of numerous studies and works, given the benefits achievable from their application. Some remarkable applications in this field are listed below. Methods for approximating High Fidelity models through a local scaling process of Low Fidelity data have been proposed by Alexandrov et al. [12] and Gano et al. ([13],[14]). These methods, known as correction-based methods, employ a certain correction function, or bridge function, of additive, multiplicative or a combination of the former types, to merge HF and LF data [1]. Multi-fidelity surrogate models based on response surface modeling and basis function regression have been investigated by Vitali et al. [15], Choi et al. [16] and Goldsmith et al. [17], while artificial neural networks-based SMs are addressed by Liu and Wang [18] and Minisci and Vasile [19]. As cited by Fernandez-Godino [3], Kriging is one of the primary and most widely used types of surrogate model for the development of MFSMs. Kriging is an interpolation technique originally used in geostatistics, developed by G. Matheron based on the studies of Danie Krige, a South African mining engineer; it is typically used as a single-fidelity technique, however, the extension to the multi-fidelity domain of Kriging is known as Co-Kriging, which involves the use of data from different fidelity levels to construct a more efficient surrogate model [3]. In the literature, the study of surrogate models based on Co-Kriging is manifested in the works of Kennedy and O'Hagan [20], LeGratiet [21], Chung and Alonso [22] and Forrester [23], as well as many more studies, especially in more recent years. The use of surrogate models constructed with Kriging or Co-Kriging techniques is particularly beneficial for both optimization problems and uncertainty quantification applications [3]. In the field of aerospace engineering, surrogate models are widely used in the context of aerodynamic shape optimization for airfoils, turbine blades and turbomachinery, as well as in the design of atmospheric or space vehicles, structural design and the generation of

aerodynamic databases [24]. Specifically, the use of Co-Kriging surrogate models for generating aerodynamic data, predicting aerodynamic coefficients and optimizing aerodynamic surface shapes has been extensively discussed in various works, such as Han et al. [24], Wang et al. [25], and Peng et al. [26]. For applications related to turbomachinery and turbine blade optimization, reference can be made to works like Keane [27].
Further information on surrogate models, such as their implementation in the context of Co-Kriging, is provided in the following sections, particularly in Chapter 2 and Chapter 4. A detailed theoretical and mathematical description of Kriging and Co-Kriging techniques is presented below; the latter will be subsequently employed in a case study to evaluate the performance of a cold gas thruster and optimize its geometry (Chapter 4).

1.2 Kriging

This section delves into the theory underlying Kriging and provides a derivation thereof as reported in Jones et al. [28].

Kriging, often referred to as Gaussian Process regression or interpolation, is a technique rooted in geostatistics, introduced in 1951 by South African geologist Danie Krige as a statistical interpolation model aimed at locating potential promising mining areas in a given spatial domain [29]. A rigorous mathematical formulation was subsequently developed by Georges Matheron [30], again within the framework of spatial statistics. The introduction of the technique into the engineering field of computer analysis is attributed to Sacks et al. [31], aiming to approximate the results of computer experiments.
In general, the Kriging method allows to obtain an estimate of a certain parameter or objective function in unexplored regions of the domain of interest through the interpolation of sampled data at a specific surrounding probe positions, where values are known. Moreover, a spatial correlation exists among the values at the sampling points, delineating what is commonly termed as the dataset's *covariance* (Equation 1.2) [29]. To clarify this concept, one may consider the implementation of the technique proposed by Jones et al. in [28]: the objective function to be approximated is not known a priori throughout the entire domain of interest. Therefore, when one seeks to make a prediction at a certain point $\mathbf{x}$ within the domain, there exists uncertainty regarding the value assumed by the function at that point. In Kriging formulations, a method is proposed to model such uncertainty by considering that the value assumed by the function at the sampling point $\mathbf{x}$ is equivalent to the value assumed by a random variable, namely $Y(\mathbf{x})$, which follows a normal distribution with mean μ and variance σ^2 [28]. In other words, the function value

at point $\mathbf{x}$ may vary randomly around the mean value μ within a range of variations depending on σ. Now, considering two points, $\mathbf{x_i}$ and $\mathbf{x_j}$, there will still be some uncertainty regarding the values assumed by the function at those points. However, if the function can be considered continuous, then it is reasonable to assume that the closer $\mathbf{x_i}$ and $\mathbf{x_j}$ are, the closer the values of $Y(\mathbf{x_j})$ and $Y(\mathbf{x_j})$ will be. Therefore, if the distance $\|\mathbf{x_i} - \mathbf{x_j}\|$ is small, the random variables $Y(\mathbf{x_i})$ and $Y(\mathbf{x_j})$ will be more highly correlated [28]. At the core of Kriging lies the assumption that the correlation between random variables follows the form

$$\mathrm{Corr}[Y(\mathbf{x}_i), Y(\mathbf{x}_j)] = \exp\left(-\sum_{l=1}^{d}(\theta_l|\mathbf{x}_{il} - \mathbf{x}_{jl}|^{p_l}\right) \tag{1.1}$$

It follows from Equation (1.1) that if $\mathbf{x_i} = \mathbf{x_j}$, then the correlation is maximum and equals 1, while as the distance between the two points tends to infinity, the correlation will tend to 0.
The parameters θ_l and p_l are called model *hyperparameters* and influence the spatial correlation between the sampling points. In particular, the parameter p_l determines the smoothness of the correlation function along the direction of the coordinate l [23].

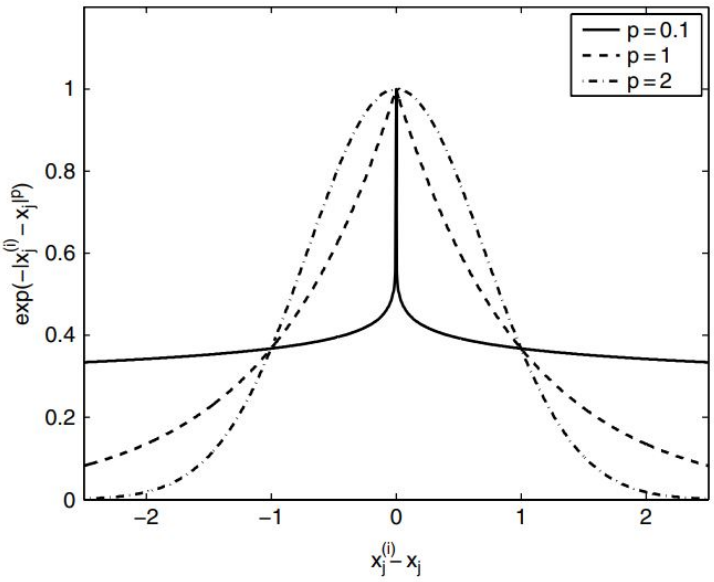

Figure 1.3: Correlations with varying p_l [5]

Figure 1.3 shows the effect of p_l on the points correlation: $p = 2$ implies a smoother and continuous correlation function that can be used to model functions characterized by regular variations in the data. It is important to remember that the regularity or irregularity of the correlation function plays a fundamental role

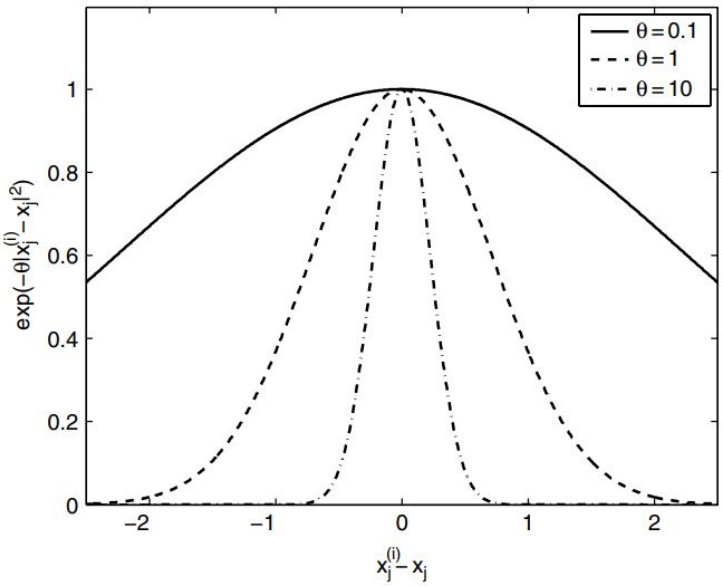

Figure 1.4: Correlations with varying θ_l [5]

in the effectiveness of the Kriging model in capturing the spatial variation of the data [5]. The decrease of p_l, instead, determines a more rapid initial decay of the correlation function as the distance between $\mathbf{x_i}$ and $\mathbf{x_j}$ increases [5]. For small values of p_j, the correlation function presents a sort of discontinuity for $\mathbf{x_i} = \mathbf{x_j}$ and can be used to model functions characterized by sharp variations in the data. As far as θ_l is concerned, this parameter indicates how quickly the spatial correlation between $\mathbf{x_i}$ and $\mathbf{x_j}$ decays as the two points $\mathbf{x_i}$ and $\mathbf{x_j}$ move away from each other [23]. The effect of θ_l is shown in Figure 1.4.
High values of θ_l are used to model functions that are highly active in the variable l, i.e. functions whose values vary greatly over small distances [28]. Low values of θ_l are related to functions exhibiting low variability within the domain of interest [5].

Once a sampling plan is chosen and a set of n points in d dimensions $\mathbf{x} = \{\mathbf{x_1}, \ldots, \mathbf{x_n}\}^T$ is obtained, it is possible to encapsulate the observed values of the function to be approximated at the points in $\mathbf{x}$ in the vector $\mathbf{y} = \{y_1, \ldots, y_n\}^T$. As mentioned earlier, the function values will be characterized by some uncertainty, which is represented by the vector $\mathbf{Y} = \{Y(\mathbf{x_1}), \ldots, Y(\mathbf{x_n})\}^T$ [28]. The latter has a mean value of $\mathbf{1}\mu$ ($\mathbf{1}$ is a $\mathbb{R}^{n\times 1}$ vector of ones) and covariance matrix given by:

$$Cov(\mathbf{Y}, \mathbf{Y}) = \sigma^2 \mathbf{R} \tag{1.2}$$

$\mathbf{R}$ is a $n \times n$ matrix, which entries are described by Equation 1.1. The *covariance* measures the correlation that exists between two or more sets of random variables

[5]. Given generic X and Y the relation between covariance and correlation is shown in Equation 1.3

$$Corr(X,Y) = \frac{Cov(X,Y)}{\sigma_x \sigma_y} \tag{1.3}$$

where σ_x and σ_y are X and Y's respective standard deviations.

Before constructing the Kriging regressor, optimal values for the *hyperparameters* p_l and θ_l, as well as μ and σ must be evaluated: the values of these parameters are chosen in order to maximize the likelihood function of the observed data, i.e.

$$\frac{1}{(2\pi)^{\frac{n}{2}}(\sigma^2)^{\frac{n}{2}}|\mathbf{R}|^{\frac{1}{2}}} \exp\left[-\frac{(\mathbf{y}-\mathbf{1}\boldsymbol{\mu})^T\mathbf{R}^{-1}(\mathbf{y}-\mathbf{1}\boldsymbol{\mu})}{2\sigma^2}\right] \tag{1.4}$$

By maximizing the likelihood function, one would wish the Kriging approximation to be as consistent as possible with the observed values of the function in $\mathbf{x}$ [28]. In general, a more practical way of doing this, is by maximizing the natural logarithm of the likelihood function, disregarding the constant terms that arise; thus, the problem becomes

$$-\frac{n}{2}\log(\sigma^2) - \frac{1}{2}\log(|\mathbf{R}|) - \frac{(\mathbf{y}-\mathbf{1}\boldsymbol{\mu})^T\mathbf{R}^{-1}(\mathbf{y}-\mathbf{1}\boldsymbol{\mu})}{2\sigma^2} \tag{1.5}$$

Now, by differentiating Equation 1.5 with respect to μ and σ^2 and setting these derivatives equal to zero, it is possible to derive optimal values for the mean and variance in terms of the spatial correlation matrix $\mathbf{R}$. Consequently, denoting those values with the symbol $\hat{}$, we obtain

$$\hat{\mu} = \frac{\mathbf{1}^T\mathbf{R}^{-1}\mathbf{y}}{\mathbf{1}^T\mathbf{R}^{-1}\mathbf{1}} \tag{1.6}$$

$$\hat{\sigma}^2 = \frac{(\mathbf{y}-\mathbf{1}\boldsymbol{\mu})^T\mathbf{R}^{-1}(\mathbf{y}-\mathbf{1}\boldsymbol{\mu})}{n} \tag{1.7}$$

Introducing Equations 1.6 and 1.7 in Equation 1.5, a "concentrated log-likelihood" function is obtained:

$$-\frac{n}{2}\log(\sigma^2) - \frac{1}{2}\log(|\mathbf{R}|) \tag{1.8}$$

Maximizing Equation 1.8, which only depends on $\mathbf{R}$, and thus p_l and θ_l, estimates for these parameters are obtained. Then, $\hat{\mu}$ and $\hat{\sigma}^2$ are derived from Equations 1.6 and 1.7 with the estimated *hyperparameters* [28].
At this point, Kriging model parameters are found and the interpolation phase has been completed. A regressor is now needed in order to predict the value of

the Kriging approximation in unprobed locations. Continuing in line with the approach proposed by Jones et al. [28], let's denote by $\hat{y}$ a new prediction at a point $\hat{\mathbf{x}}$, which does not belong to the initial set of points used to train the model. This new prediction must, of course, be consistent with the observed values $\mathbf{y}$ at points $\mathbf{x}$ used in the previous phase and, therefore, must also be consistent with the *hyperparameters* θ_l and p_l obtained. To ensure this, a new maximum likelihood estimate (MLE) must be made, this time of the set of sample data $\mathbf{y}$ and new prediction $\hat{y}$. Therefore, the $\mathbf{y}$ vector is augmented, as well as the correlation matrix $\mathbf{R}$:

$$\tilde{\mathbf{y}} = \begin{bmatrix} \mathbf{y} \\ \hat{y} \end{bmatrix} \tag{1.9}$$

$$\tilde{\mathbf{R}} = \begin{bmatrix} \mathbf{R} & \mathbf{r} \\ \mathbf{r}^T & 1 \end{bmatrix} \tag{1.10}$$

where

$$\mathbf{r} = \begin{bmatrix} \mathrm{Corr}[Y(\hat{\mathbf{x}}), Y(\mathbf{x_1})] \\ \vdots \\ \mathrm{Corr}[Y(\hat{\mathbf{x}}), Y(\mathbf{x_n})] \end{bmatrix} \tag{1.11}$$

Maximizing the new log-likelihood function given in Equation 1.12

$$-\frac{n}{2}\ln(2\pi) - \frac{n}{2}\ln(\sigma^2) - \frac{1}{2}\ln|\tilde{\mathbf{R}}| - \frac{(\tilde{\mathbf{y}} - \mathbf{1}\boldsymbol{\mu})^T \tilde{\mathbf{R}}^{-1} (\tilde{\mathbf{y}} - \mathbf{1}\boldsymbol{\mu})}{2\sigma^2} \tag{1.12}$$

the MLE for $\tilde{y}$, i.e. the Kriging predictor, is found

$$\tilde{y}(\mathbf{x}) = \hat{\mu} + \mathbf{r}^T \mathbf{R}^{-1} (\mathbf{y} - \mathbf{1}\hat{\mu}) \tag{1.13}$$

Equation 1.13 can be rewritten as

$$\tilde{y}(\mathbf{x}) = a + \sum_{i=1}^{n} (b_i \phi(\hat{\mathbf{x}} - \mathbf{x}_i)) \tag{1.14}$$

Equation 1.14 shows that the prediction provided by the Kriging model can be viewed as a combination of a linear regression function (the second term), typically of polynomial type, and a stochastic Gaussian process with zero mean, variance σ^2 and correlation matrix given by Equation 1.11 [32]. The regression term consists of known function, called bases, of the form of Equation 1.1 and unknown coefficients, called weights, that need to be evaluated.
Finally, as stated by Jones et al. [28], a potential estimate of the error committed by the predictor in estimating the objective function is given by the curvature of

the augmented log-likelihood function (Equation 1.12): in particular, an inversely proportional relationship is identified, thus the flatter the log-likelihood function, the greater the potential error in the prediction. The curvature of this function can be computed through the second derivative of the augmented log-likelihood function with respect to $\hat{y}$. The estimate of the potential error is given by

$$s^2(\hat{\mathbf{x}}) = \hat{\sigma}^2 \left[1 - \mathbf{r}^T \mathbf{R}^{-1} \mathbf{r} + \frac{(1 - \mathbf{r}^T \mathbf{R}^{-1} \mathbf{r})^2}{\mathbf{1}^T \mathbf{R}^{-1} \mathbf{1}} \right] \tag{1.15}$$

1.3 Co-Kriging

In Section 1.2, a comprehensive overview of Kriging was provided as a statistical interpolation technique aimed at approximating an unknown objective function, typically available only at a limited number of points, arising from simulations or describing the output of a computationally expensive model. Kriging finds widespread application as a surrogate model in aerodynamic and propulsion fields, particularly in profile shape optimization, approximation of costly CFD simulations, aerodynamic databases creation and data fusion. Several variations of Kriging have been proposed in the literature over the years, including Simple Kriging, Ordinary Kriging, Stochastic Kriging, Blind Kriging and so on, yet these fall beyond the scope of this thesis. Instead, the key focus is on extending Kriging, a single-fidelity model approach, to the realm of multi-fidelity modeling, the benefits of which have been introduced in previous sections. Indeed, incorporating a large amount of rapidly and inexpensively generated data, through the use of a Low Fidelity model, reduces the number of computationally expensive High Fidelity evaluations in the process of constructing the surrogate model [11]. In this context, Co-Kriging leverages prior information about the behaviour of the objective function provided by LF models and merges it with a limited number of HF data, yielding a more accurate surrogate model compared to what can be achieved with standard Kriging, given the same costly observations. One of the main challenges underlying Co-Kriging is how to properly scale data of different fidelity, namely by choosing the appropriate scaling factor ρ, one of the model's *hyperparameters*. Furthermore, constructing the Co-Kriging prediction involves estimating the covariance and cross-variance of the data used for model training [11].

Below is a comprehensive discussion of the Co-Kriging technique, leveraging the derivations proposed by Forrester et al. [23] and Kennedy and O'Hagan [20]; the aim is to approximate a function whose evaluation is costly by utilizing data from rapid and less expensive, albeit less accurate, evaluations of that function. The discussion is confined to the case of a dual fidelity, as reported in [23].
Let us consider two sets of data, one referred to as cheap, represented by the points

$\mathbf{X_c}$ and the low-accuracy values $\mathbf{y_c}$, and one as expensive, consisting of the samples $\mathbf{X_e}$ and the accurate values $\mathbf{y_e}$. For simplicity, a nested configuration is assumed for the points $\mathbf{X_e}$, which will be contained within the $\mathbf{X_c}$ vector; this facilitates the evaluation of the scaling factor ρ between the data of the two fidelities [23]. Concatenating the two vectors yields

$$\mathbf{X} = \begin{bmatrix} \mathbf{X_c} \\ \mathbf{X_e} \end{bmatrix} = \left(\mathbf{x_c}^{(1)}, \ldots, \mathbf{x_c}^{(n_c)}, \mathbf{x_e}^{(1)}, \ldots, \mathbf{x_e}^{(n_e)} \right)^T \tag{1.16}$$

Just as in Kriging, in Co-Kriging, the sampled values at generic points $\mathbf{X}$ are regarded as the outcomes of a random variable following a Gaussian distribution [23]. Specifically, the following random field is obtained

$$\mathbf{Y} = \begin{bmatrix} \mathbf{Y_c}(\mathbf{X_c}) \\ \mathbf{Y_e}(\mathbf{X_e}) \end{bmatrix} = \left(Y_c\left(\mathbf{x_c}^{(1)}\right), \ldots, Y_c\left(\mathbf{x_c}^{(n_c)}\right), Y_e\left(\mathbf{x_e}^{(1)}\right), \ldots, Y_e\left(\mathbf{x_e}^{(n_e)}\right) \right)^T \tag{1.17}$$

As advocated by Kennedy and O'Hagan [20], it is assumed that the values obtained at points $\mathbf{X_e}$ through costly simulations are truthful, hence not subject to error, and that any error is solely associated with values obtained from less expensive simulations. More precisely, if the value of the expensive function is known at a given point $\mathbf{x}^{(i)}$, then at that point, the cheap function cannot provide any relevant information, as the value $Y_e\left(\mathbf{x}^{(i)}\right)$ is known form an exact simulation (this assumption is often referred to as the Markov property) [23]. Mathematically, this translates into

$$Cov\left\{ Y_e\left(\mathbf{x}^{(i)}\right), Y_c(\mathbf{x}) | Y_c\left(\mathbf{x}^{(i)}\right) \right\} = 0, \forall \mathbf{x} \neq \mathbf{x}^{(i)} \tag{1.18}$$

Therefore, the values that the High and Low Fidelity functions assume are associated with Gaussian Processes (GP), which will be referred to as $Z_e(\cdot)$ and $Z_c(\cdot)$ respectively. The Co-Kriging will determine an approximation of the expensive High Fidelity function as the linear combination of the scaled low-cost function by a certain constant factor ρ and a Gaussian Process denoted by $Z_d(\cdot)$, which is nothing but the difference between $Z_e(\cdot)$ and $\rho Z_c(\cdot)$

$$Z_e(\mathbf{x}) = \rho Z_c(\mathbf{x}) + Z_d(\mathbf{x}) \tag{1.19}$$

As in Kriging, also in this case a covariance matrix must be constructed; the structure of this matrix will be more complex compared to Equation 1.2 seen in the case of Kriging. Firstly, below are reported the relations necessary to construct the covariance matrix of Co-Kriging:

$$Cov\left\{\mathbf{Y}_c\left(\mathbf{X}_c\right),\mathbf{Y}_c\left(\mathbf{X}_c\right)\right\}=Cov\left\{Z_c\left(\mathbf{X}_c\right),Z_c\left(\mathbf{X}_c\right)\right\}=\sigma_c^2\mathbf{R}_c\left(\mathbf{X}_c,\mathbf{X}_c\right) \tag{1.20}$$

$$Cov\left\{\mathbf{Y}_e\left(\mathbf{X}_e\right),\mathbf{Y}_c\left(\mathbf{X}_c\right)\right\}=Cov\left\{\rho Z_c\left(\mathbf{X}_c\right)+Z_d\left(\mathbf{X}_c\right),Z_c\left(\mathbf{X}_e\right)\right\}=\rho\sigma_c^2\mathbf{R}_c\left(\mathbf{X}_c,\mathbf{X}_e\right) \tag{1.21}$$

$$\begin{aligned}Cov\left\{\mathbf{Y}_e\left(\mathbf{X}_e\right),\mathbf{Y}_e\left(\mathbf{X}_e\right)\right\}&=Cov\left\{\rho Z_c\left(\mathbf{X}_e\right)Z_d\left(\mathbf{X}_e\right),\rho Z_c\left(\mathbf{X}_e\right)+Z_d\left(\mathbf{X}_e\right)\right\}=\\&=\rho^2Cov\left\{Z_c\left(\mathbf{X}_e\right),Z_c\left(\mathbf{X}_e\right)\right\}+Cov\left\{Z_d\left(\mathbf{X}_e\right),Z_d\left(\mathbf{X}_e\right)\right\}=\\&=\rho^2\sigma_c^2\mathbf{R}_c\left(\mathbf{X}_e,\mathbf{X}_e\right)+\sigma_d^2\mathbf{R}_d\left(\mathbf{X}_e,\mathbf{X}_e\right)\end{aligned} \tag{1.22}$$

In Equations 1.20, 1.21 and 1.22, $\mathbf{R}_c\left(\mathbf{X}_c,\mathbf{X}_e\right)$ denotes the matrix of spatial correlations between $\mathbf{X}_c$ and $\mathbf{X}_e$, as reported in Equation 1.11. These equations give rise to the covariance matrix $\mathbf{C}$

$$\mathbf{C}=\begin{bmatrix}\sigma_c^2\mathbf{R}_c\left(\mathbf{X}_c,\mathbf{X}_c\right) & \rho\sigma_c^2\mathbf{R}_c\left(\mathbf{X}_c,\mathbf{X}_e\right)\\ \rho\sigma_c^2\mathbf{R}_c\left(\mathbf{X}_e,\mathbf{X}_c\right) & \rho^2\sigma_c^2\mathbf{R}_c\left(\mathbf{X}_e,\mathbf{X}_e\right)+\sigma_d^2\mathbf{R}_d\left(\mathbf{X}_e,\mathbf{X}_e\right)\end{bmatrix} \tag{1.23}$$

In this case as well, the correlations follow the form of Equation 1.1, however, in Co-Kriging there will be two correlations, one for the cheap case and one for the expensive case. This results in an increase in the model *hyperparameters*, which are now θ_c, θ_e, p_c, p_e and the scaling parameter rho.
As stated by Forrester et al. [23], one can consider the cheap Low Fidelity data to be independent with respect to expensive HF data, thus the maximization of the log-likelihood for cheap values will lead to an estimate for μ_c, σ_c^2, θ_c and p_c.

$$-\frac{n_c}{2}\log(\sigma_c^2)-\frac{1}{2}\log\left|det\left(\mathbf{R}_c\left(\mathbf{X}_c,\mathbf{X}_c\right)\right)\right|-\frac{\left(\mathbf{y}_c-\mathbf{1}\boldsymbol{\mu}_c\right)^T\mathbf{R}_c\left(\mathbf{X}_c,\mathbf{X}_c\right)^{-1}\left(\mathbf{y}_c-\mathbf{1}\boldsymbol{\mu}_c\right)}{2\sigma_c^2} \tag{1.24}$$

The MLEs $\hat{\mu_c}$ and $\hat{\sigma_c^2}$ are found deriving Equation 1.23 with respect to μ_c and σ_c^2 and setting them equal to zero:

$$\hat{\mu}_c=\frac{\mathbf{1}^T\mathbf{R}_c\left(\mathbf{X}_c,\mathbf{X}_c\right)^{-1}\mathbf{y}_c}{\mathbf{1}^T\mathbf{R}_c\left(\mathbf{X}_c,\mathbf{X}_c\right)^{-1}\mathbf{1}} \tag{1.25}$$

$$\hat{\sigma}_c^2=\frac{\left(\mathbf{y}_c-\mathbf{1}\hat{\mu}_c\right)^T\mathbf{R}_c\left(\mathbf{X}_c,\mathbf{X}_c\right)^{-1}\left(\mathbf{y}_c-\mathbf{1}\hat{\mu}_c\right)}{n_c} \tag{1.26}$$

Similarly to what has been done in Section 1.2, by substituting Equations 1.25 and 1.26 into Equation 1.24, we obtain the concentrated log-likelihood, the maximization of which enables the determination of $\hat{\theta}_c$ and $\hat{p}_c$

$$-\frac{n_c}{2}\log(\hat{\sigma}_c^2) - \frac{1}{2}\log\left|det\left(\mathbf{R}_c\left(\mathbf{X}_c, \mathbf{X}_c\right)\right)\right| \tag{1.27}$$

As far as the difference model is concerned, a distance parameter $\mathbf{d}$ as follows:

$$\mathbf{d} = \mathbf{y}_e - \rho\mathbf{y}_c\left(\mathbf{X}_e\right) \tag{1.28}$$

If the values of $\mathbf{y}_c$ are not known at the points $\mathbf{X}_e$, one could use Kriging employing the *hyperparameters* $\hat{\theta}_c$ and $\hat{p}_c$ derived earlier, in order to obtain an estimate of $\hat{\mathbf{y}}_c\left(\mathbf{X}_e\right)$ [23]. MLEs $\hat{\mu}_d$ and $\hat{\sigma}_d^2$ are derived from the maximization of the log-likelihood function of $\mathbf{d}$:

$$-\frac{n_e}{2}\log(\sigma_d^2) - \frac{1}{2}\log\left|det\left(\mathbf{R}_d\left(\mathbf{X}_e, \mathbf{X}_e\right)\right)\right| - \frac{\left(\mathbf{d} - \mathbf{1}\boldsymbol{\mu}_d\right)^T\mathbf{R}_d\left(\mathbf{X}_e, \mathbf{X}_e\right)^{-1}\left(\mathbf{d} - \mathbf{1}\boldsymbol{\mu}_d\right)}{2\sigma_d^2} \tag{1.29}$$

Thus we obtain

$$\hat{\mu}_d = \frac{\mathbf{1}^T\mathbf{R}_d\left(\mathbf{X}_e, \mathbf{X}_e\right)^{-1}\mathbf{d}}{\mathbf{1}^T\mathbf{R}_d\left(\mathbf{X}_e, \mathbf{X}_e\right)^{-1}\mathbf{1}} \tag{1.30}$$

$$\hat{\sigma}_d^2 = \frac{\left(\mathbf{d} - \mathbf{1}\hat{\mu}_d\right)^T\mathbf{R}_d\left(\mathbf{X}_e, \mathbf{X}_e\right)^{-1}\left(\mathbf{d} - \mathbf{1}\hat{\mu}_d\right)}{n_e} \tag{1.31}$$

Finally, the MLEs of the *hyperparameters* $\hat{\theta}_d$ and $\hat{p}_d$, as well as $\hat{\rho}$, are found maximizing the concentrated log-likelihood

$$-\frac{n_e}{2}\log(\hat{\sigma}_d^2) - \frac{1}{2}\log\left|det\left(\mathbf{R}_d\left(\mathbf{X}_e, \mathbf{X}_e\right)\right)\right| \tag{1.32}$$

As observed, in order to obtain estimates of the aforementioned parameters, it is necessary to maximize Equations 1.27 and 1.32, process that involves the use of numerical algorithms. However, in cases involving a large number of data points and a significant number of parameters to tune, this process may render the construction of the Co-Kriging surrogate model overly resource-intensive [23]. The likelihood maximization, in fact, entails the computation of numerous matrix inversions, a task that becomes impractical for large-sized matrices, while the number of steps required in the Maximum Likelihood Estimation process is a function of the number

of parameters to be evaluated [23]. To mitigate this issue, as stated by Forrester et al. [23], it is often necessary to impose constraints on certain parameters, albeit at the cost of reducing the accuracy of the Co-Kriging model.
As far as the Co-Kriging predictor is concerned, one can follow the same logical steps described in Section 1.2 for Kriging: datasets are augmented with a prediction in a certain point, an augmented covariance matrix is constructed and a MLE is performed keeping the model parameters fixed. The Co-Kriging predictor takes the following form:

$$\hat{y}_e(\mathbf{x}) = \hat{\mu} + \mathbf{c}^T \mathbf{C}^{-1} \left(\mathbf{y} - \mathbf{1}\hat{\mu}\right) \tag{1.33}$$

where $\mathbf{C}^{-1}$ is the inverse of the covariance matrix and

$$\mathbf{c} = \begin{bmatrix} \rho\hat{\sigma}_c^2 \mathbf{r}_c\left(\mathbf{X}_c, \mathbf{x}\right) \\ \left(\rho^2\hat{\sigma}_c^2 + \hat{\sigma}_d^2\right) \mathbf{r}_d\left(\mathbf{X}_e, \mathbf{x}\right) \end{bmatrix} \tag{1.34}$$

$$\hat{\mu} = \frac{\mathbf{1}^T \mathbf{C}^{-1} \mathbf{y}}{\mathbf{1}^T \mathbf{C}^{-1} \mathbf{1}} \tag{1.35}$$

In Equation 1.34, $\mathbf{r}_c\left(\mathbf{X}_c, \mathbf{x}\right)$ a column vector of correlations between expensive data $\mathbf{X}_e$ and the new point in wich we want to make a prediction [23]. As reported by Forrester et al. [23], if the new point belongs to one of the expensive sample points, then Equation 1.33 will interpolate the expensive data, while if the new point is one of the cheap points, then Co-Kriging predictor will regress $\mathbf{y}_c$, if $\mathbf{y}_c \neq \mathbf{y}_e$. Moreover, given that Co-Kriging interpolates the expensive data points $\mathbf{y}_e$, one would expect a zero error at these sample points ($\mathbf{X}_e$). This can be inferred from the formula of the estimated mean squared error

$$s^2(\mathbf{x}) = \hat{\rho}^2\hat{\sigma}_c^2 + \hat{\sigma}_d^2 - \mathbf{c}^T \mathbf{C}^{-1} \mathbf{c} \tag{1.36}$$

1.4 One variable Co-Kriging example

In many publications in the literature, the one-variable toy problem commonly used to demonstrate the application of Co-Kriging for surrogate model creation is the one proposed by Forrester et al. [23]. The implementation involves the following High and Low Fidelity functions, defined over the interval [0,1]:

$$f_e(x) = (6x - 2)^2 sin(12x - 4) \tag{1.37}$$

$$f_c(x) = Af_e + B(x - 0.5) - C \tag{1.38}$$

where f_e and f_c denote respectively the HF and LF functions and A = 0.5, B = 10 and C = 5. As far as sample points are concerned, the Low Fidelity model is evaluated at the cheap points $\mathbf{X}_c = \{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1\}$, while

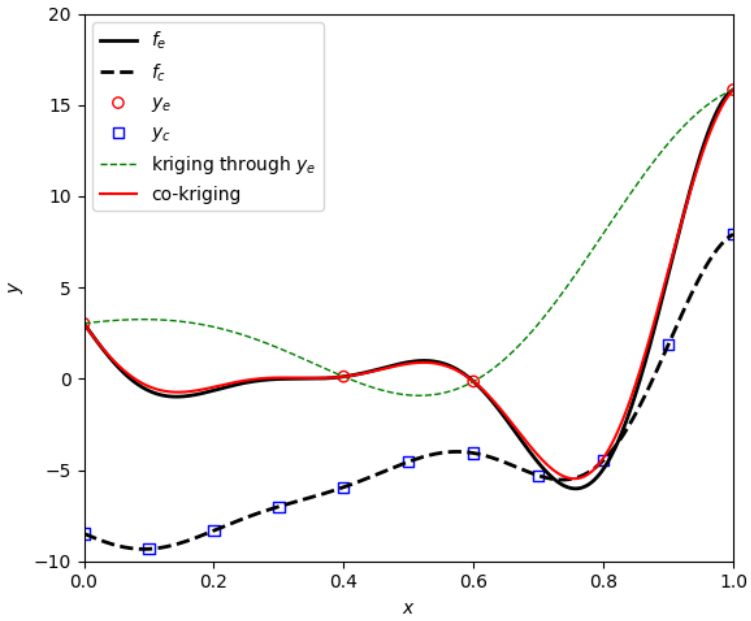

Figure 1.5: One variable Kriging and Co-Kriging example as reported in Forrester et al. [23]

only four evaluations of the expensive model are carried out in $\mathbf{X}_e = \{0, 0.4, 0.6, 1\}$. The Kriging and Co-Kriging estimates can be seen in Figure 1.5
As evident, an approximation of the High Fidelity function has been generated through Kriging using four expensive points, indicated with red dots in Figure 1.5. However, the Kriging approximation proves to be less accurate, except at the sample points, where the error is zero, as the exact values given by $f_e(\mathbf{X}_e)$ are known. To achieve a better prediction through Kriging, a larger number of High Fidelity points would be necessary, which generally entails a very high additional computational cost.
Conversely, the benefit of using Co-Kriging over Kriging is immediately apparent: by incorporating information derived from the Low Fidelity function, whose evaluations are less computationally intensive and thus feasible even in large quantities, it is possible to obtain a highly satisfactory approximation of the objective function, with the same number of expensive points.
As previously mentioned in Section 1.2 and stated by Forrester et al. [23], given the fact that a simple relation exists between HF and LF data, the *hyperparameters* θ_c and p_c of the Low Fidelity model only depend on cheap data, thus they can be derived from Equation 1.27, i.e. maximizing the concentrated log-likelihood function. The same can be done for the *hyperparameters* concerning the difference model given by Equation 1.28, whose MLEs are obtained using Equation 1.32.

In order to construct the Co-Kriging prediction, a suitable value for the scaling factor ρ must be found; for the case presented in Figure 1.5, Forrester et al. [23] demonstrated that the MLE $\hat{\rho}$, which is a scaling parameter between the Gaussian Processes $Z_c(\cdot)$ and $Z_e(\cdot)$, only depends on the value of A that appears in the Equation 1.38. Precisely, it is shown that $\hat{\rho} \approx 1/A$ and a value of $\hat{\rho} = 1.87$ is found based on the available data shown in Figure 1.5.

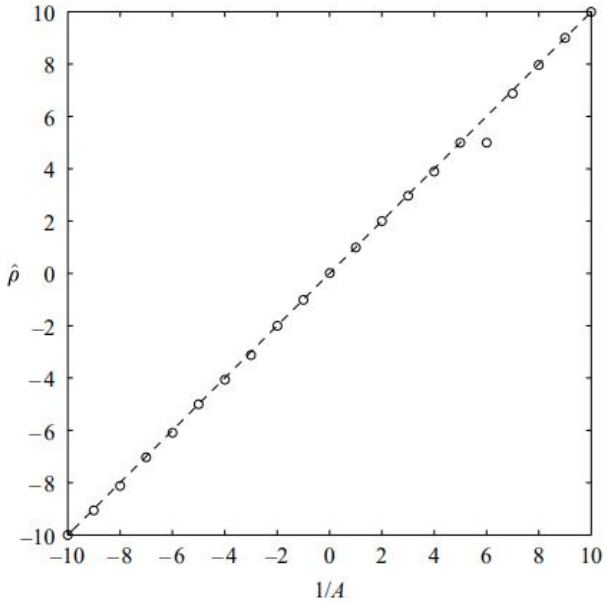

Figure 1.6: Plot of $\hat{\rho}$ versus 1/A [23]

The MLE for the scaling parameter is usually found employing genetic algorithms (GA) for the maximization of the concentrated log-likelihood given in Equation 1.32; however, the derivation of $\hat{\rho}$ is often anything but a trivial problem and can sometimes lead to inaccurate results. Finding the best possible estimate for the scaling factor is therefore one of the key points in implementing the Co-Kriging technique, as the closer $\hat{\rho}$ is to the real value, the better the surrogate model's ability to approximate the High Fidelity function, thereby reducing the error.

1.5 Thesis outline and objective

The purpose of this thesis is to develop a surrogate model using the multi-fidelity Co-Kriging technique for predicting the performance of a cold gas microthruster, a propulsion system typically used for attitude control maneuvers of small satellites in orbit, in terms of the thrust coefficient C_F, varying with the geometry of the nozzle's divergent section. To achieve this, it will be necessary to identify two

models, one High Fidelity and one Low Fidelity, for calculating the thrust coefficient and to populate two datasets to subsequently create the surrogate model. This process will involve a series of CFD simulations, evaluating the flow within the nozzle first by solving the Euler equations, thus obtaining an approximate result that does not account for viscosity effects, and subsequently solving the Navier-Stokes equations to obtain very accurate and realistic C_F values. To minimize the computational costs required for generating the surrogate model, an extensive study on adaptive sampling algorithms for the High Fidelity points is carried out in Chapter 2, comparing the performance of three different criteria, i.e., Maximum Variance, Maximum Expected Improvement and Maximum Information Gain, for various test functions of one and two variables. The most satisfactory algorithm will then be used to generate the Co-Kriging model related to the thrust coefficient for the cold gas thruster. Chapter 3 provides a general discussion on such propulsion systems and introduces an evaluation of the exhaust nozzle performance of a cold gas thruster through CFD simulations. The application of adaptive sampling techniques and the practical implementation of the Co-Kriging surrogate for the case study is carried out in Chapter 4: following an extensive discussion on the exploration of the surrogate model, the phase of exploiting this model is also addressed. The surrogate model will be used in place of the High Fidelity model response, which is unknown except for the few points used for training and validating the Co-Kriging. The ultimate objective is to set up a multi-objective optimization that identifies a set of optimal solutions to maximize the thrust coefficient of the cold gas thruster, thereby enhancing propulsion performance while minimizing the mass of the propulsion system as much as possible. This will allow for the identification of possible nozzle geometries to be implemented in the design phase, based on a trade-off of the obtained optimal values.

Chapter 2

Building a Co-Kriging surrogate

In Chapter 1, a theoretical discussion regarding surrogate models was provided, specifically Kriging and Co-Kriging techniques, primarily employed in the field of optimization. This chapter presents an implementation of the Co-Kriging technique to approximate a High Fidelity objective function from two databases: one Low Fidelity (LF) with numerous points but limited accuracy with respect to the objective function and one High Fidelity (HF), characterized by a few evaluations of high accuracy.
First of all, a series of fundamental concepts for creating surrogate models are introduced, with particular attention given to the selection of sampling points for model training and to potential algorithms for identifying a subset of High Fidelity points for Co-Kriging. One of the main benefits that one can achieve through surrogate models is the possibility to obtain a good approximation of the objective function while keeping computational costs reasonable and contained.

2.1 Building a multi-fidelity surrogate model

As previously mentioned, the use of surrogate models has achieved significant success in the aerospace field, particularly in the generation of aerodynamic databases and related optimization processes [33]. These processes typically require numerous CFD simulations, which can be critical in terms of simulation time and computational cost. The implementation of a surrogate model with an effective sampling strategy thus enables a reduction in the number of required CFD simulations while maintaining a high level of accuracy in approximating the behaviour of the physical system under study [34]. For example, the evaluation of forces and moments on an aircraft, as well as the calculation of the structural loads acting on it, would usually require, in

the absence of a surrogate model, a full-factorial approach for sampling in a design space with multiple parameters. This would necessitate an extremely high number of simulations or wind tunnel measurements, which are typically conducted in a more advanced phase of the design process [34]. The utility of surrogate models thus becomes evident and the sampling strategy of the design space points also becomes vitally important, with the aim of minimizing the computational burdens while maintaining the desired accuracy compared to the full-factorial approach. Figure 2.1 outlines the main steps to follow in the implementation of a surrogate model:

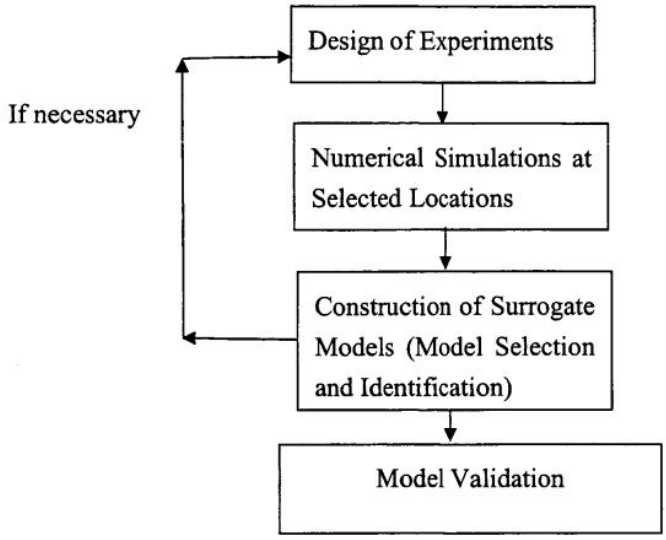

Figure 2.1: Key stages of the surrogate-based modeling approach [33]

The first step involves defining the Design of Experiments (DOE), which is the sampling strategy for selecting points within the variable space at which the objective function will be evaluated [33]. When implementing a surrogate model using Co-Kriging, since two functions are being evaluated (one computationally expensive and the other cheap) two DOEs must be considered. The selection of High Fidelity points is subject to more constraints, as the computational resources required to evaluate the expensive function's observations are significantly high. In contrast, for the cheap function, one generally has access to a larger number of samples, thereby enabling the acquisition of extensive information about the behaviour of the Low Fidelity model across the entire parameter space or at least large portions of it. The sampling strategy, in this case, will vary depending on the Low Fidelity model and the available computational budget. However, it is usually possible to populate a large LF dataset with relative ease. Starting from the knowledge of the Low Fidelity model's response in a substantial number of samples, the ideal

scenario would be to minimize the number of High Fidelity function evaluations or simulations required. This is one of the fundamental challenges in implementing a good Co-Kriging surrogate model and it does not have a general solution in terms of a minimum threshold of points necessary to ensure a certain accuracy. The number and selection of High Fidelity points heavily depend on the specific problem being studied and the choice of one particular sample plan over another impacts the surrogate model's approximation accuracy. While there is no universal criterion that guarantees finding both the minimum threshold of points and their optimal positions in the variable space, there are many criteria and algorithms in the literature that provide guidance on choosing these points. These will be presented later in this chapter.
Once the datasets with the sample points for both the Low Fidelity and High Fidelity models have been constructed, in the case of Co-Kriging, the required simulations can be performed to populate the two datasets, thereby enabling the construction of the surrogate model. To achieve this, one must first derive the parameters that characterize the model, known as *hyperparameters*. In the case of Kriging-based and Co-Kriging-based surrogate models, these values are obtained by constructing a likelihood function and maximizing it, as discussed in Sections 2.2 and 2.3 of Chapter 1. The resulting trained model allows for the prediction of the High Fidelity model's response at positions within the variable space that have not yet been tested, thus enabling exploration of this model at virtually no computational cost.
As previously mentioned, the choice of sampling strategy influences the results obtainable with the surrogate. Therefore, it is necessary to evaluate the accuracy of the constructed model relative to the expensive objective function it aims to approximate. This leads to the validation of the model, with the goal of estimating its ability to predict values at untested locations and assessing what is known as the generalization error [33].

As shown in Figure 2.1, if the obtained surrogate does not ensure the required accuracy or is unable to correctly predict the behaviour of the objective function in a certain region of the study domain, it is possible to modify the initial Design of Experiment to introduce new points for training the surrogate or to choose a different strategy. This process, of course, necessitates a careful cost-benefit analysis, aiming to find a balance between exploration and exploitation of the model [5]. Specifically, one must assess the feasibility of introducing new sample points in light of the improvement achievable from such additions and the associated costs, before proceeding to the actual use of the model, for instance, to perform optimization or uncertainty quantification.

2.2 Design of Experiment for multi-fidelity models

When utilizing Co-Kriging, or generally in the implementation of a multi-fidelity model, two distinct Design of Experiments (DOE) need to be identified, one for the Low Fidelity and one for High Fidelity models. This results in obtaining the necessary points for the evaluations of the respective functions and the population of the two datasets. Typically, the points that make up the expensive dataset can either fall in different positions compared to the cheap ones (non-nested design) or constitute a subset of them (nested design) [1]. In the context of multi-fidelity modeling, particularly for Co-Kriging, it has been observed that using High Fidelity points contained within those of the Low Fidelity model facilitates the evaluation of the scaling factor ρ between the two models [23] and helps reduce the approximation error, in terms of Mean Squared Error (MSE) across the variable space, compared to the non-nested configuration [34].

Since the constructed surrogate model will be more accurate near the points where the High Fidelity objective function value is known, it is typically desirable to create a sampling plan that is space-filling. This means that the points should be distributed to avoid leaving unexplored areas in the variable space in terms of the model's accurate response behaviour [5]. However, the number of expensive points will typically be very low due to the associated computational costs, making it challenging to uniformly explore the parameter space in the High Fidelity case. Conversely, because Low Fidelity data are usually much quicker and less costly to obtain, generating a space-filling sampling plan is crucial for obtaining large amounts of information about the response behaviour of the cheap model. This allows the enhancement of the few expensive evaluations and reduces the model's uncertainty as well as the approximation error. Therefore, for generating LF samples, classical space-filling techniques, such as the full-factorial approach, the construction of a Latin Square Sample (LSS) or a Latin Hypercube Sample (LHS), can be applied [5]. In contrast, for generating the expensive dataset, many studies, including those by Mackman and Allen [34], have demonstrated that adaptive sampling strategies (i.e., the progressive addition of points in the HF subset based on "interim models of the data" [34]) outperform the classical space-filling strategies mentioned earlier.

Regarding space-filling approaches, the simplest to implement is the so-called full-factorial method, which involves sampling a rectangular grid of points within the model's variable space. An example of full-factorial sampling in three dimensions is shown in Figure 2.2. This type of sampling allows for uniform coverage of

the variable space. However, as can be seen in Figure 2.2, projecting the points onto the axes does not guarantee a uniform distribution for all variables. To address this issue, one can use the approach known as stratified random sampling or its extension, the Latin Hypercube sampling [5].

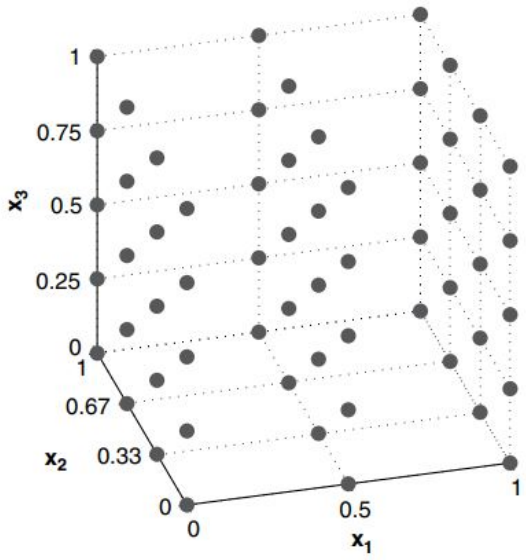

Figure 2.2: Example of a three-dimensional full-factorial sampling plan [5]

For approaches that ensure greater stratification of points in the design space, there are the Latin Square sampling (in two dimensions) and the Latin Hypercube sampling (in three or more dimensions). For the former, given n as the number of points to be sampled, an $n \times n$ square grid is generated by placing a permutation of the vector {1,2,...,n} in each row and column. Figure 2.3 shows an example

2	**1**	3	4
3	2	4	**1**
1	4	2	3
4	3	**1**	2

Figure 2.3: Latin Square sampling example for $n = 4$ [5]

for $n = 4$: the n points to be chosen can be those indicated by 1 or by the other numbers, resulting in a uniformly distributed sampling plan.
The extension to the multidimensional case, namely the use of Latin Hypercube sampling, allows for the division of the domain of interest into hypercubes of equal size, also known as bins; within each bin, a sample point is placed such that the resulting design is uniformly distributed [5]. This ensures that moving from one

occupied bin parallel to the axes of the various variables does not encounter another point. An implementation in three dimensions is shown in Figure 2.4:

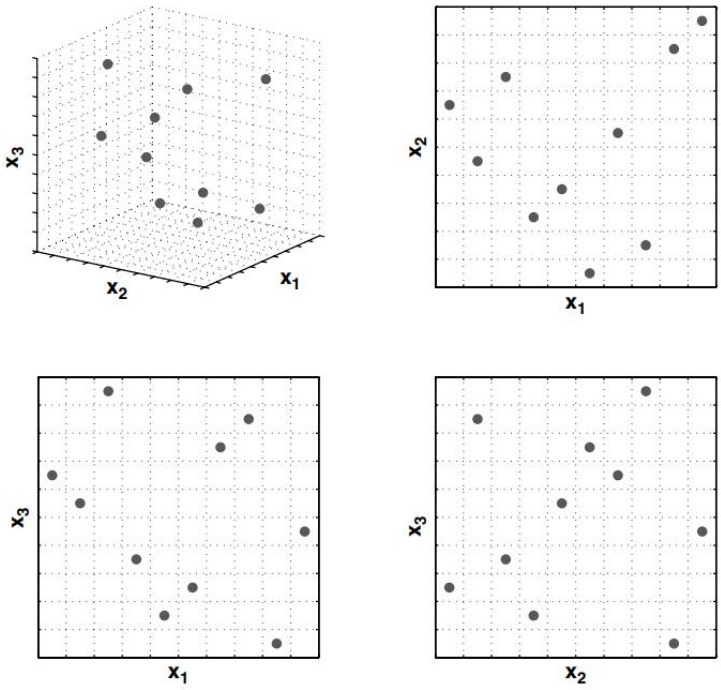

Figure 2.4: Three-variable, ten-point Latin Hypercube sampling plan shown in three dimensions (top left), along with its two-dimensional projections [5]

However, there may be cases where the generated Latin Hypercube does not ensure complete coverage of the design domain, such as when the sampling points align along the diagonal of the cube. To ensure that the sample plan is truly space-filling, one can opt for a DOE that adheres to the *maximin* criterion, introduced by Johnson et al. [35]. Let d_1, d_2, ..., d_m be the sorted unique distances between all pairs of points in a sampling plan $\mathbf{X}$ and let J_1, J_2, ..., J_m be defined such that J_j is the number of pairs of points in $\mathbf{X}$ separated by distance d_j [5]. Following the Morris-Mitchel definition [36], the sample plan, among all the possible ones, that respects the *maximin* criterion is the one that maximizes d_1 and, among those, minimizes J_1 and then, among those, maximizes d_2 and minimize J_2 and so on until the minimization of J_m. With the term distance, the p-norm of the design space is intended:

$$d_p\left(\mathbf{x}^{(i_1)}, \mathbf{x}^{(i_2)}\right) = \left(\sum_{j=1}^{k} \left|x_j^{(i_1)} - x_j^{(i_2)}\right|^p\right)^{1/p} \tag{2.1}$$

In practice, the definition proposed by Morris and Mitchel [36] can also be interpreted as follows: given the quantity $\mathbf{\Phi_q}(\mathbf{X})$, defined as

$$\mathbf{\Phi_q}(\mathbf{X}) = \left(\sum_{j=1}^{m} J_j d_j^{-q} \right)^{1/q} \tag{2.2}$$

the Latin Hypercube sample characterized by the best space-filling properties is the one with the minimum value of $\mathbf{\Phi_q}(\mathbf{X})$. For further details on the proposed criterion and for an implementation in Matlab of the search for a Latin Hypercube sample that ensures a space-filling distribution within the domain of interest, one can refer to [36] and Forrester et al. [5].

In summary, the use of a full-factorial strategy or Latin Hypercube sampling is well suited for generating sampling points for the Low Fidelity model, as it provides a large amount of information across the entire parameter space, which will later assist the few computationally expensive observations. The more information obtained from the cheap model, the better the Co-Kriging surrogate prediction will be. However, in some cases, finding the set of points with the best space-filling properties for the Latin Hypercube can be computationally expensive, so this aspect must be considered.

For selecting points for the High Fidelity model, the main constraint is the computational resources available for performing the necessary CFD simulations or evaluating the expensive objective function. Consequently, if, for example, the HF points are chosen as a subset of the LF ones, only a small number can be selected and a subset with adequate space-filling properties cannot always be guaranteed. Instead of uniformly exploring the entire parameter space, it is often more effective to sample expensive points in regions where the objective function's behaviour is significant or where the surrogate model's uncertainty is high. The statistical aspect thus becomes crucial in predicting which points in the design space are significant for training an effective surrogate model, minimizing the number of required observations and, consequently, the samples to be selected.

2.2.1 Adaptive sampling for High Fidelity points

Regarding the sampling strategy for High Fidelity points, adaptive methods have proven highly successful in the field of Kriging and Co-Kriging surrogate models. These methods replace one-shot approaches, where sampling points are determined before evaluating model responses at those points (e.g., factorial, Latin Hypercubes), with sequential algorithms that identify the most promising point at each iteration to be added to the High Fidelity database, based on various criteria or statistical

measures [37]. In practice, starting with a very small number of expensive points, an iterative algorithm is implemented. At each step, the algorithm leverages the abundant Low Fidelity data, the limited High Fidelity values available and the information provided by the surrogate model generated from these datasets. Thus, this process is used to select the next point to be included in the sampling plan and once it is sampled, the accurate model response is observed thereof and the iterations continue, until either the computational budget is exhausted or the error falls below a certain threshold at validation points used for the surrogate's approximation capability evaluation.

Given that Kriging and Co-Kriging techniques have their roots in statistics, it is evident how statistically-based criteria for implementing an adaptive algorithm, aimed at progressively identifying points to add to the High Fidelity sample plan, have found significant application in numerous studies and publications in the literature [34]. In particular, the entropy of the HF points system can be evaluated, which measures the amount of information provided by the various points in the dataset and represents a measure of the model's level of uncertainty within the variable space [34]. Thus, maximizing the entropy of the system allows for maximizing the information gain by positioning a new sample point where the mean squared error (MSE) s^2 of the Kriging or Co-Kriging model (Equation 1.36) is the highest [34]. At each iteration, the goal is to identify the point in the design space that can provide the most information, specifically where the model's predicted uncertainty is high. This approach mainly results in space-filling sampling plans that progressively reduce the global uncertainty of the surrogate model, thereby enhancing its predictive capabilities.
Another application of statistical adaptive methods for selecting High Fidelity points pertains to the global optimization of the surrogate, specifically the search for global maximum or minimum points in the variable space. The objective is to explore the design space to progressively find a value closer to the global maximum or minimum of the objective function, while ensuring a good approximation across the entire variable space [5]. In this context, an often considered optimization criterion is the so-called Expected Improvement (EI), introduced in the late '70s and described by Jones et al. [38] and thoroughly described by Forrester in [5].
Other criteria used in sequential sampling techniques and in the generation of aerodynamic databases include the leave-one-out cross-validation error (LOOCV) [37] or the study of the gradient or Laplacian of a set of points, inserting a new evaluation where these values are highest [39].

In the following sections, implementations of adaptive methods for constructing a High Fidelity sampling plan based on an existing Low Fidelity one are presented. For simplicity, the Low Fidelity plan can be considered equispaced in

the one-variable case or factorial in the multi-variable case. Firstly, the ooDACE Matlab toolbox [32], used for implementing Co-Kriging and calculating the model's hyperparameters, is introduced.

2.3 Co-Kriging implementation in Matlab: the ooDACE Toolbox

To achieve a simple, quick, and effective implementation of the Co-Kriging method introduced in the previous chapter, one can utilize the ooDACE Toolbox for Matlab [32]. This toolbox provides a series of functions for constructing Gaussian Process based Kriging and Co-Kriging surrogate models. The toolbox implements all the steps described in Sections 1.2 and 1.3 and constructs a surrogate model based on the available data provided by the user in terms of sample points and observations of the objective function. Specifically, let n denote the number of observations made and d the number of input parameters. The user needs to create an n x d matrix named **samples**, which contains the sample points used for evaluating the objective function, and an n x *1* matrix named **values**, which will hold the corresponding values that the objective function assumes at the specified sample points. The toolbox then provides a function *oodacefit.m* that takes the user-generated datasets, i.e., the two matrices **samples** and **values** and optionally an options structure, and generates a Kriging object. This object contains all the necessary information for the surrogate implementation, such as the model's *hyperparameters*.
For the implementation of a Co-Kriging surrogate, it is required to specify both the sample points related to the cheap Low Fidelity function and those related to the expensive High Fidelity function, as well as the values these functions assume at those points. Therefore, the inputs **samples** and **values** must now be cell arrays of length two: specifically, the first element of **samples** will contain an n_c x d matrix of cheap sample points, while the second element will specify the expensive ones (n_e x d); similarly, for the observations, **values**$\{1\}$ will contain n_c x *1* values of f_c and **values**$\{2\}$ will contain n_e x *1* values of f_e. To summarize, using the two simple calls

$$\text{Kriging\_model} = \text{oodacefit}(\mathbf{samples}, \mathbf{values});$$

$$\text{Cokriging\_model} = \text{oodacefit}(\mathbf{samples}, \mathbf{values});$$

with the input data for Kriging and Co-Kriging in the form specified above, it is possible to generate a Kriging and a Co-Kriging model ready for use, with *hyperparameters* directly computed through the call to the *oodacefit.m* function. The computational cost for generating the models primarily depends on data fitting and the calculation of *hyperparameters* through the maximization of the likelihood

function, which must be evaluated multiple times [32]. To enhance this process, the covariance matrix is factorized using Cholesky decomposition, the computation of which constitutes the main part of the computational cost of the code. In terms of time cost, this is $O(n^3)$, where n is the number of rows of the matrix [32].
Regarding the evaluation of the prediction of the Kriging or Co-Kriging model at a certain point within the domain of interest, at various points, or directly across the entire parameter space, this can be easily achieved using the *predictor.m* function provided by ooDACE. Specifically, given $\mathbf{x}$ as the vector containing the points at which one wants to calculate the value assumed by the surrogate model, the following call allows not only to obtain these values but also to compute the estimated mean squared error s^2, which for Co-Kriging refers to Equation 1.36:

$$[y, s^2] = \text{predictor}(\mathbf{x}, \text{model});$$

2.4 One-variable adaptive sampling implementation

2.4.1 Maximum Variance/MSE criterion

Assuming we have a sampling plan for Low Fidelity simulations and have already evaluated the response of this model at the sampled points, we can initially select two points in the design space to construct a High Fidelity dataset. This initial dataset allows us to generate a preliminary surrogate model, which will almost certainly provide a poor approximation of the costly objective function but will offer valuable information in terms of the surrogate model's variance s^2. For a Gaussian Process based prediction, s^2 can be defined as [28]

$$\hat{s}^2(\mathbf{x}) = \hat{\sigma}^2 \left[1 - \boldsymbol{\psi}^T \boldsymbol{\Psi}^{-1} \boldsymbol{\psi} + \frac{1 - \mathbf{1}^T \boldsymbol{\Psi}^{-1} \boldsymbol{\psi}}{\mathbf{1}^T \boldsymbol{\Psi}^{-1} \mathbf{1}} \right]$$

This parameter, expressed in Equation 1.36 for Co-Kriging, estimates the model's prediction uncertainty at various points where the surrogate's response is evaluated. Consequently, it identifies points with the highest uncertainty, suggesting the addition of a High Fidelity sample where the uncertainty is maximum. Specifically, high variance values occur where s^2 is elevated, indicating regions where adding a sampling point would significantly reduce uncertainty due to the maximum potential gain in information. As noted in Chapter 1, the model's variance, sometimes referred to as the estimated Co-Kriging mean squared error (MSE), is zero at the High Fidelity points where the expensive function's value is known; therefore, by iteratively selecting the point with the highest s^2 in the design space, the predictive uncertainty will be progressively reduced. Thus, the surrogate model

will increasingly converge to the objective function, thereby progressively reducing the actual error across the domain at points not used for Co-Kriging training. The following figure presents an initial application of the algorithm for inserting a new High Fidelity point at the position of maximum predictive variance of the available Co-Kriging surrogate model at a given iteration.

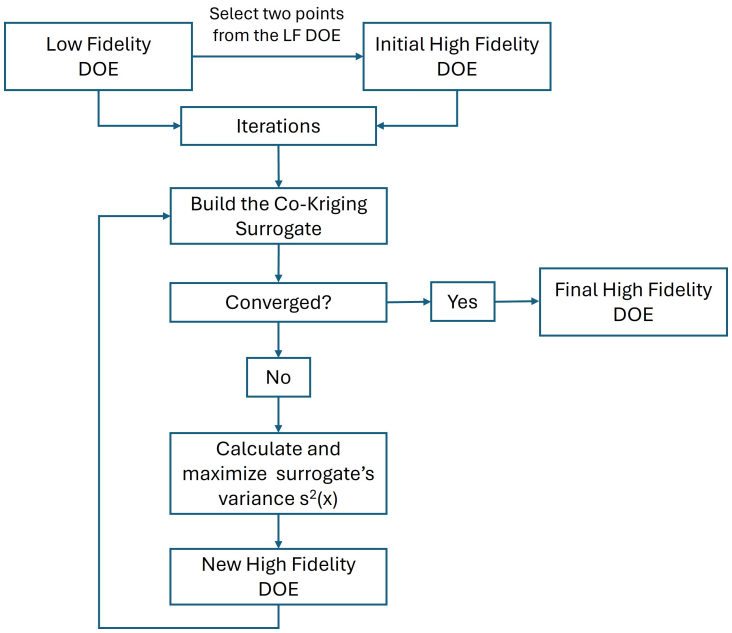

Figure 2.5: Algorithm for choosing new High Fidelity points based on the Maximum Variance/MSE criterion

To achieve this, at least two points for the expensive model must be known. In subsequent implementations, these points will be positioned at the extremes of the design space, coinciding with the first and last points of the Low Fidelity sample vector (following a nested approach for the HF points, the new samples will be chosen from LF locations). In the first iteration, a Co-Kriging surrogate is constructed using the LF points and the two HF observations. The predicted values of this model in the study domain and the estimated prediction error s^2 at the cheap points locations are then evaluated, selecting the point with the highest variance for addition to the costly dataset.
Regarding the stopping criterion for the iterative algorithm, several approaches can be followed. If the High Fidelity function and its behaviour are known, the mean absolute error between this function and the surrogate model can be evaluated, stopping the process when the approximation error falls below a certain threshold. This approach is generally feasible when studying algorithms on test functions with known analytical expressions but becomes inapplicable when the response of the

expensive model is only known at the few available sampling points. For example, if the HF dataset is populated through CFD simulations, the response is known only at a few points and it is not possible to evaluate the distance between the expensive model and the surrogate, as the information is available only at the HF points, where the values coincide. Therefore, the search for new values may be halted if the uncertainty associated with the model falls below a certain threshold, or if the maximum s^2 is less than a specified value.

The described algorithm is applied to three pairs of High and Low Fidelity test functions, considering a Low Fidelity DOE as a vector of $n_c = 30$ evenly spaced points within the interval where the different functions are defined. In terms of the stopping criterion, in this case where known test functions are evaluated, the addition of High Fidelity points is halted when the mean absolute error between the expensive function f_e and the surrogate prediction falls below a certain tolerance, such as 5% (thus, $mean(|f_e(x) - f_{ck}(x)|) < 0.05$ is the stopping criterion adopted). The studied functions are as follows:

- **Test Function n° 1 in [1,4]**:

$$f_e(x) = \sin(x^2) + \cos(2\mathrm{x})$$
$$f_c(x) = 0.5\,\sin(x^2) + \cos(\mathrm{x})$$

- **Forrester function [23] in [0,1]**:

$$f_e(x) = (6x - 2)^2 \cdot \sin(12\mathrm{x} - 4)$$
$$f_c(x) = 0.5 f_e(x) + 10(x - 0.5) - 5$$

- **Test Function n° 3 [3] in [0,1]**:

$$f_e(x) = 2x\sin(20x + 2) + 10e^x + 20(x - 1)^2$$
$$f_c(x) = 0.7 f_e(x) + 10(x - 0.5) + 5$$

Thus, we proceed to leverage the algorithm defined in Figure 2.5 to find the High Fidelity DOE for the first test function presented:

$$f_e(x) = \sin(x^2) + \cos(2x)$$
$$f_c(x) = 0.5\sin(x^2) + \cos(x)$$

The High and Low Fidelity functions with the respective DOEs (the initial one for the expensive model) are reported in Figure 2.6.
In order to build the surrogate model and to evaluate its s^2 distribution in the Low Fidelity locations the ooDACE Toolbox for Matlab is employed, as described

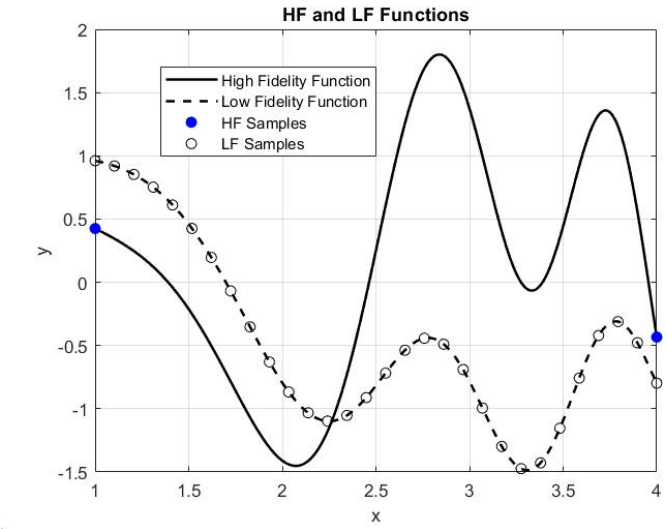

Figure 2.6: High and Low Fidelity functions

in Section 2.3. In Figure 2.8 are reported the results of the progressive iterations until the threshold for the mean absolute error is reached: in the first iteration, the Co-Kriging model is generated leveraging only the two HF data points available and the estimated mean squared error is obtained, then its maximum value is found and a new point is set in its location. The figure shows the evolution of the Co-Kriging predictor and the variance at every step; the last image refers to the absolute error.

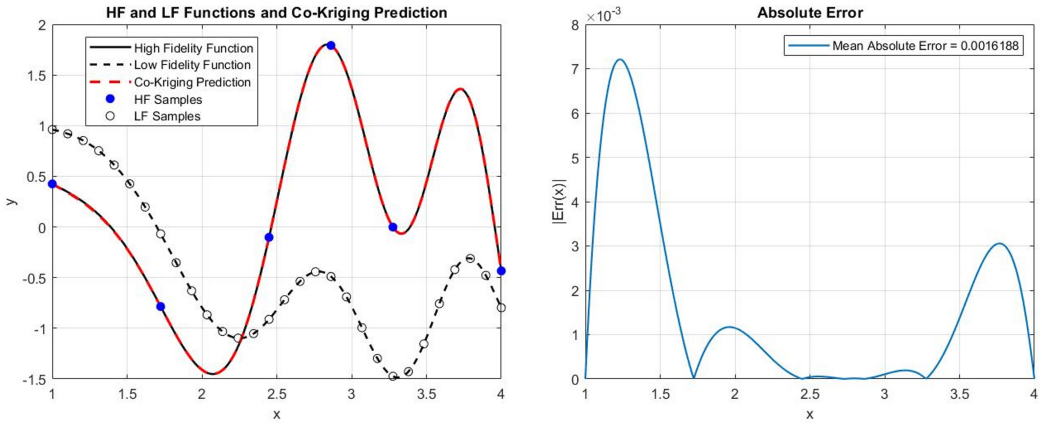

Figure 2.7: Converged Co-Kriging Model after 5 iterations, leading to a 6-points High Fidelity DOE

The High Fidelity sampling plan that leads to the convergence of the Co-Kriging

surrogate to the expensive objective function is shown in Figure 2.7. The resulted DOE is also space-filling, uniformly exploring the design space. The number of required High Fidelity points also depends on the number of samples for the cheap model: increasing $\mathbf{X}_c$ allows to provide more information on the Low Fidelity response, better supporting the few expensive observations, leading to a better surrogate. Thus, if we would have chosen $n_c < 30$, probably a higher number of HF points would have been found, requiring more computational budget.

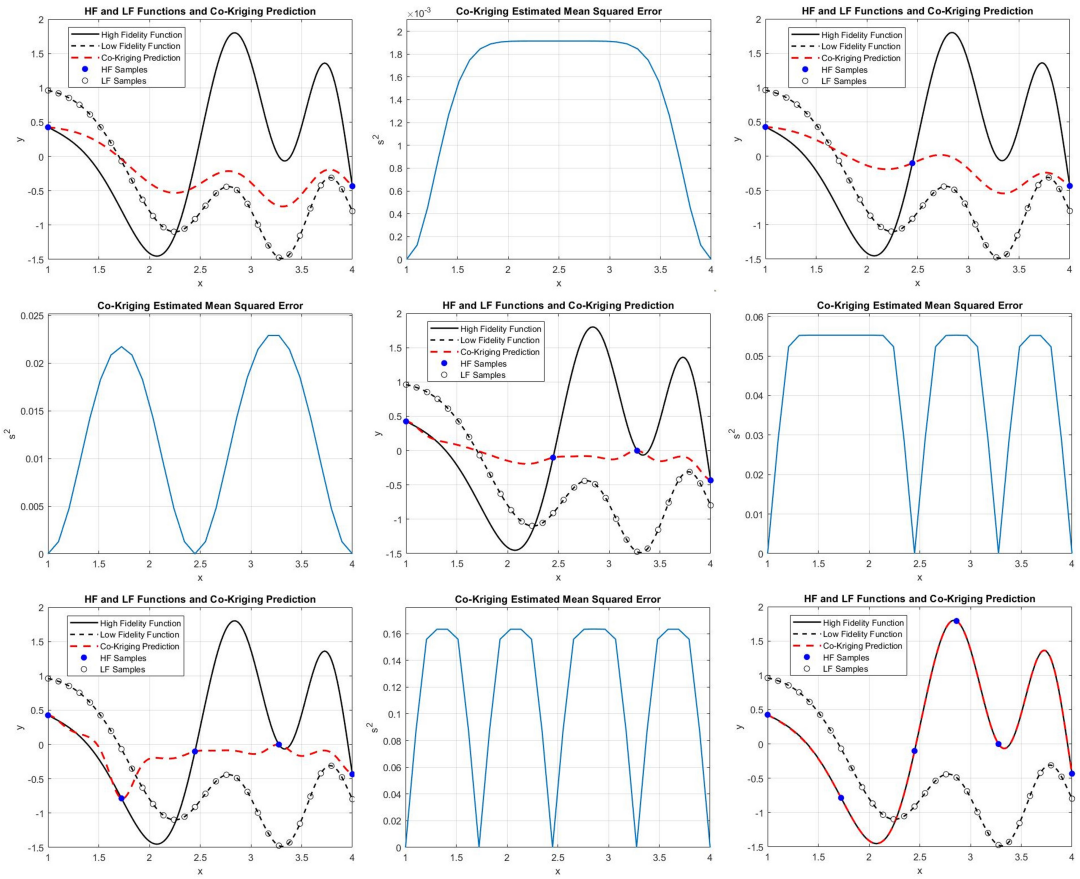

Figure 2.8: Maximum Variance sampling iterations for the first test function

To further testing the Maximum MSE algorithm, the Forrester Function [23] is considered:

$$f_e(x) = (6x - 2)^2 \cdot \sin(12x - 4)$$
$$f_c(x) = 0.5 f_e(x) + 10(x - 0.5) - 5$$

The High and Low Fidelity functions with the respective DOEs (the initial one for the expensive model) are reported in Figure 2.9.

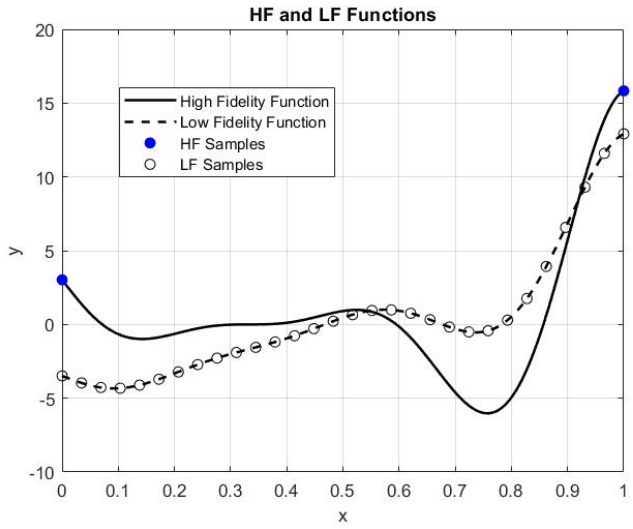

Figure 2.9: High and Low Fidelity Forrester functions

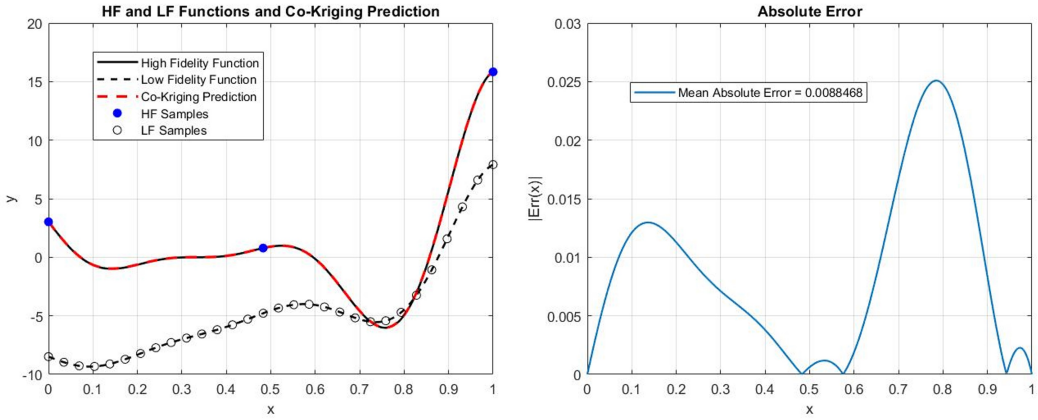

Figure 2.10: Converged Co-Kriging Model after 2 iterations, leading to a 3-points High Fidelity DOE

In this case, the algorithm converges in just two iterations, leading to the results shown in Figure 2.10. The surrogate model's estimated uncertainty when convergence is obtained for both the first test function and the Forrester function, becomes really low (10^{-4} for the first function and 10^{-5} for Forrester), with respect to the previous iterations, thus one might leverage the reduction in estimated uncertainty as a stopping criterion different than the one proposed, that requires the analytic expression of the expensive function $f_e(x)$. However, it is not guaranteed that a small value of s^2 could imply convergence; in certain cases, in the following

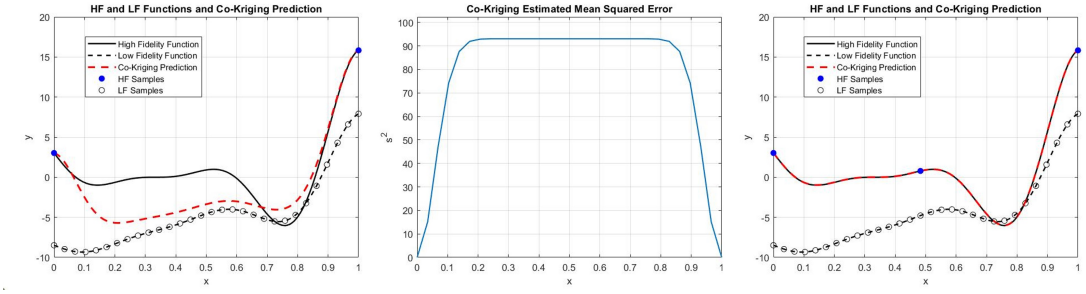

Figure 2.11: Maximum Variance sampling iterations for the Forrester function

iteration s^2 may rise, while the surrogate model gets closer to the objective function, thus a stopping criterion based on the magnitude of s^2 could sometimes lead to a premature interruption of the iterative cycle. For example, this behaviour will be seen when studying the last test function below.

Finally, the Maximum Variance criterion can be applied to the last test function mentioned, i.e.

$$f_e(x) = 2x\sin(20x+2) + 10e^x + 20(x-1)^2$$
$$f_c(x) = 0.7f_e(x) + 10(x-0.5) + 5$$

The High and Low Fidelity functions with the respective DOEs (the initial one for the expensive model) are reported in Figure 2.12.

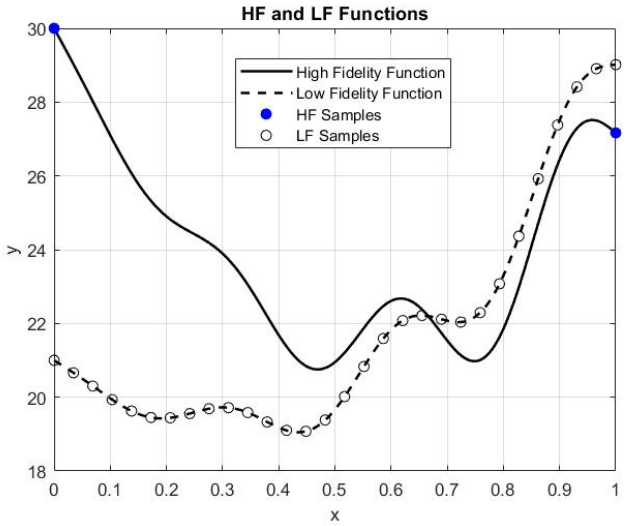

Figure 2.12: High and Low Fidelity functions

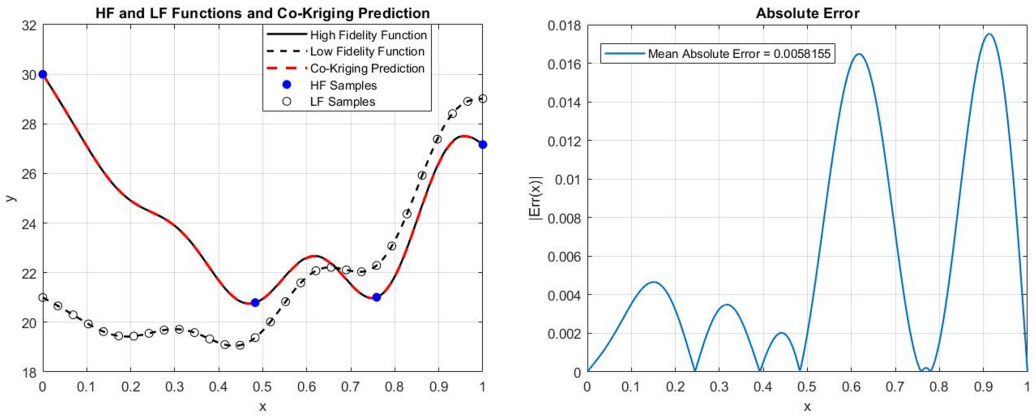

Figure 2.13: Converged Co-Kriging Model after 3 iterations, leading to a 4-points High Fidelity DOE

The iterations required to achieve a mean absolute approximation error of less than 5% are now three, as summarized in Figures 2.13 and 2.14. The estimated mean squared error s^2 reached when convergence is obtained is of the order of 10^{-8}. Although for the three test functions studied the variance when the stopping criterion is satisfied drops to really low values,one still can question the practical use of the maximum variance value obtained in an iteration as a possible stopping criterion for the cycle, for certain functions, it might happen that, at a certain point in the iteration cycle, the variance becomes very small even if the surrogate model is not accurately approximating the objective function. This could lead to deceiving results. A possible criterion that could be implemented would be to compare the surrogate model values between one iteration and the next, stopping the cycle when the difference between them falls below a threshold. However, doing so might result in a higher number of High Fidelity points than found in this section, or the addition of a point even before convergence might change the Co-Kriging approximation very little, thus prematurely and incorrectly stopping the cycle.
Upon closer analysis, this criterion proves to be usable and yields the same results for all three pairs of test functions studied if the iterations are stopped when the difference between the Co-Kriging surrogate model at step k+1 and that at step k is less than 0.1 at every point in the design space. In this way, the number of High Fidelity points will be the same as those calculated in Figures 2.7, 2.10 and 2.13 but the computational cost will be slightly higher, as it will be necessary to know the values assumed by the Co-Kriging surrogate at the next step as well.

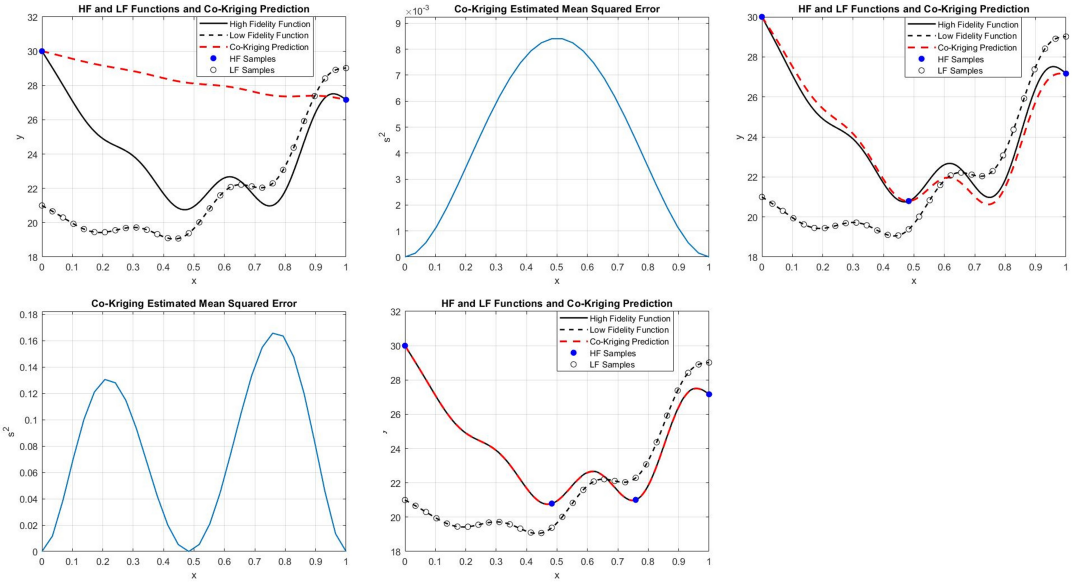

Figure 2.14: Maximum Variance sampling iterations for the third test function

2.4.2 Maximum Expected Improvement criterion

Another sampling strategy that can be utilized for selecting High Fidelity points is based on the parameter known as Expected Improvement (E[I(x)]), which balances the exploration and exploitation of the model, effectively balancing global and local search [5, 26]. In practice, selecting a point corresponding to the maximum value of the estimated mean squared error, as discussed in the previous subsection, improves the surrogate's approximation from a global perspective. However, in the case of objective functions with multiple local maxima or minima or with a global stationary point, this approach may not accurately approximate these values with a limited number of points. Thus, the aim is to balance global search, based on the study of s^2, with a more local search that strives to approximate the minima or maxima as accurately as possible. Following the description by Jones et al. [28], if the goal is to find the absolute minimum of the surrogate, the initial High Fidelity observations are used to determine the minimum value $y_{min} = \min(y^{(1)}, ..., y^{(n)})$ as the best available value. When adding a new point to the expensive database, the High Fidelity function value at that point is initially unknown, leading to uncertainty that can be expressed as the realization of a random variable with mean $\hat{y}(\mathbf{x})$ and variance $\hat{s}^2(\mathbf{x})$. The mean is simply the value obtained from the Co-Kriging predictor, while the variance is the estimated mean squared error. By leveraging these values, it is possible to construct, at each point x, a density function indicating where the addition of a new point could lead to the greatest possibility of improving the minimum value of the objective function y_{min} [28]. Evaluating the

possibilities of improvement and their respective densities yields an index called Expected Improvement (E[I(x)]), which identifies the position to sample a new point expected to find a value lower than the current y_{min}, thus searching for the absolute minimum point. An expression for the Expected Improvement is given below [5]:

$$E[I(\mathbf{x})] = \begin{cases} (y_{\min} - \hat{y}(\mathbf{x}))\Phi\left(\frac{y_{\min}-\hat{y}(\mathbf{x})}{\hat{s}(\mathbf{x})}\right) + \hat{s}\phi\left(\frac{y_{\min}-\hat{y}(\mathbf{x})}{\hat{s}(\mathbf{x})}\right) & \text{if } s > 0 \\ 0 & \text{if } s = 0 \end{cases} \tag{2.3}$$

where $\Phi(.)$ and $\phi(.)$ represent respectively the cumulative distribution function and the probability density function [5]. Clearly, the Expected Improvement at the High Fidelity sample points is zero, as the value of the expensive function is known at these points and they are already part of the dataset. Thus, adding a point at these locations would not provide any additional information or improvement. Starting from an initial set of observations, at each iteration, the E[I(x)] is calculated, the point where the maximum value occurs is found and this point is added to the High Fidelity sample vector. Figure 2.15 illustrates the algorithm based on the maximization of the Expected Improvement:

It is therefore possible to apply the described algorithm to the three test functions

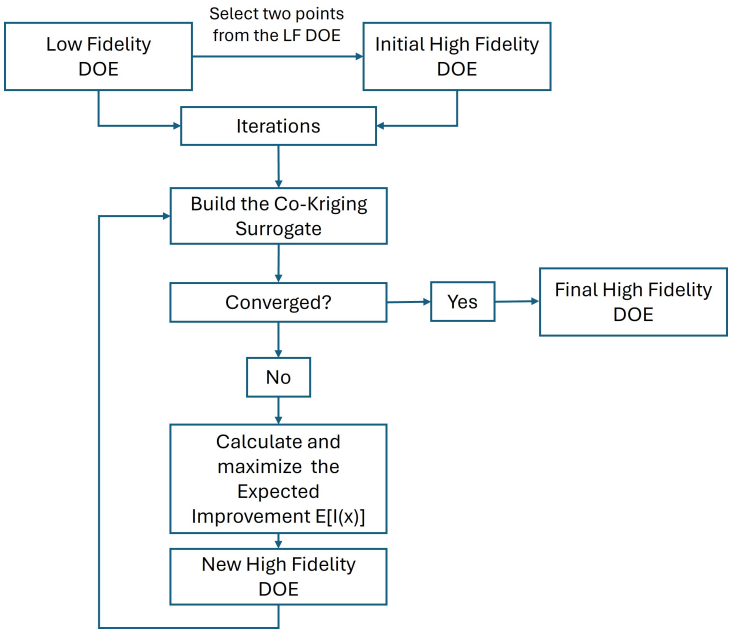

Figure 2.15: Algorithm for choosing new High Fidelity points based on the Maximum Expected Improvement criterion

studied in the context of the algorithm based on the maximum variance of the

surrogate model:

- **Test Function n° 1 in [1,4]**:

$$f_e(x) = \sin(x^2) + \cos(2\mathrm{x})$$
$$f_c(x) = 0.5\,\sin(x^2) + \cos(\mathrm{x})$$

- **Forrester function [23] in [0,1]**:

$$f_e(x) = (6x-2)^2 \cdot \sin(12\mathrm{x} - 4)$$
$$f_c(x) = 0.5 f_e(x) + 10(x - 0.5) - 5$$

- **Test Function n° 3 in [1,4]**:

$$f_e(x) = 2x\sin(20x + 2) + 10e^x + 20(x-1)^2$$
$$f_c(x) = 0.7 f_e(x) + 10(x - 0.5) + 5$$

The results are shown below for the three test cases.

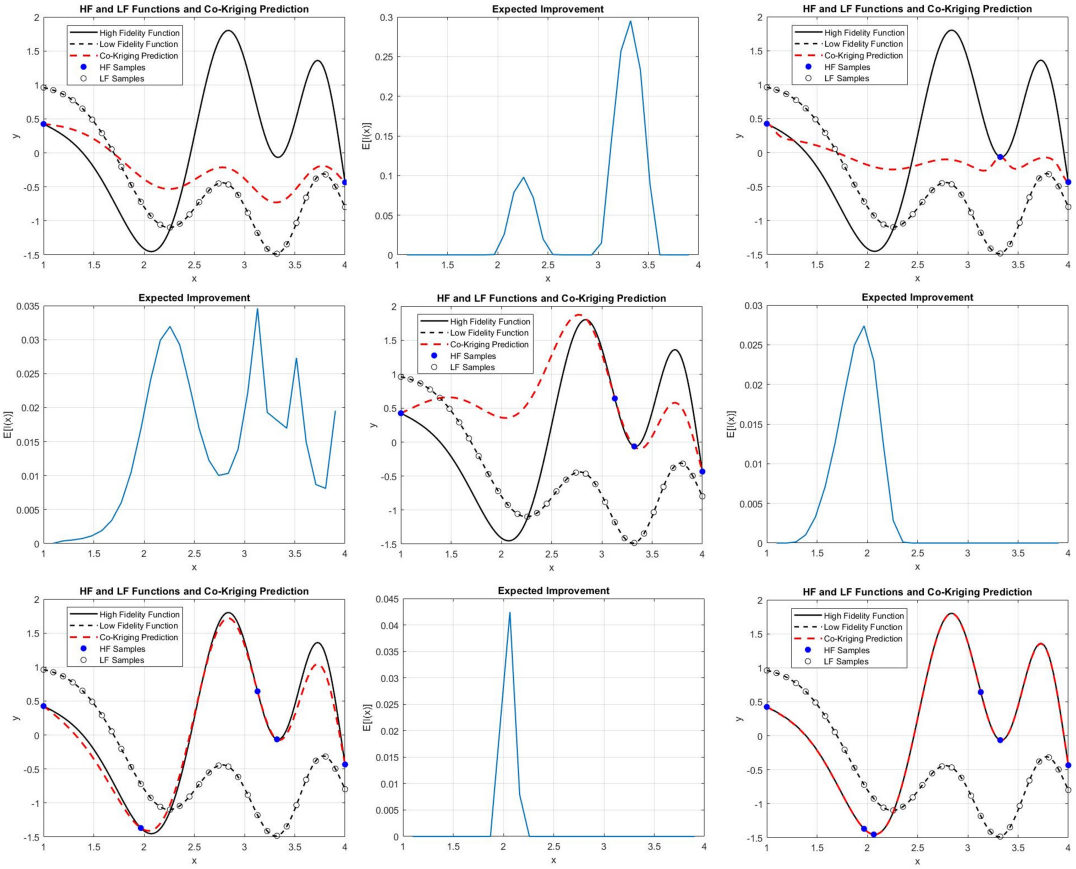

Figure 2.16: Maximum Expected Improvement sampling iterations for the first test function

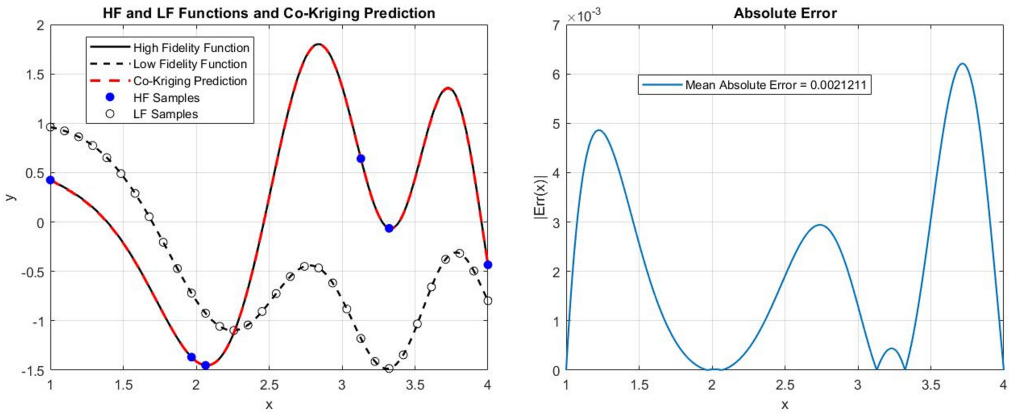

Figure 2.17: Converged Co-Kriging Model after 5 iterations, leading to a 6-points High Fidelity DOE

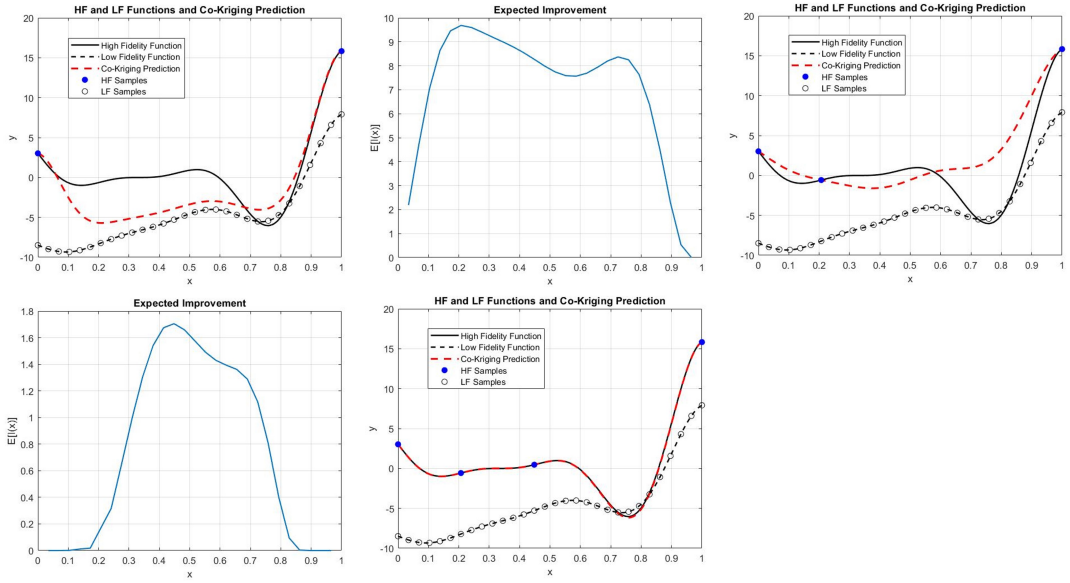

Figure 2.18: Maximum Epected Improvement sampling iterations for the Forrester function

From the results shown in these figures, it is evident that the maximum Expected Improvement criterion also achieves a convergent Co-Kriging surrogate model for the High Fidelity objective function. One can thus compare the results obtained form the Maximum Expected Improvement algorithm with the Maximum Variance criterion previously tested. For the first test function, convergence is obtained with the same number of expensive points, but their locations are different; moreover,

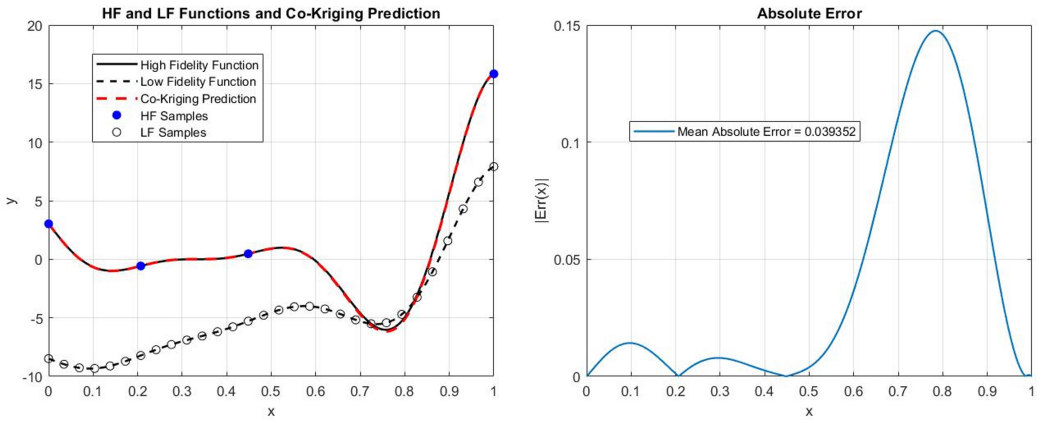

Figure 2.19: Converged Co-Kriging Model after 3 iterations, leading to a 4-points High Fidelity DOE

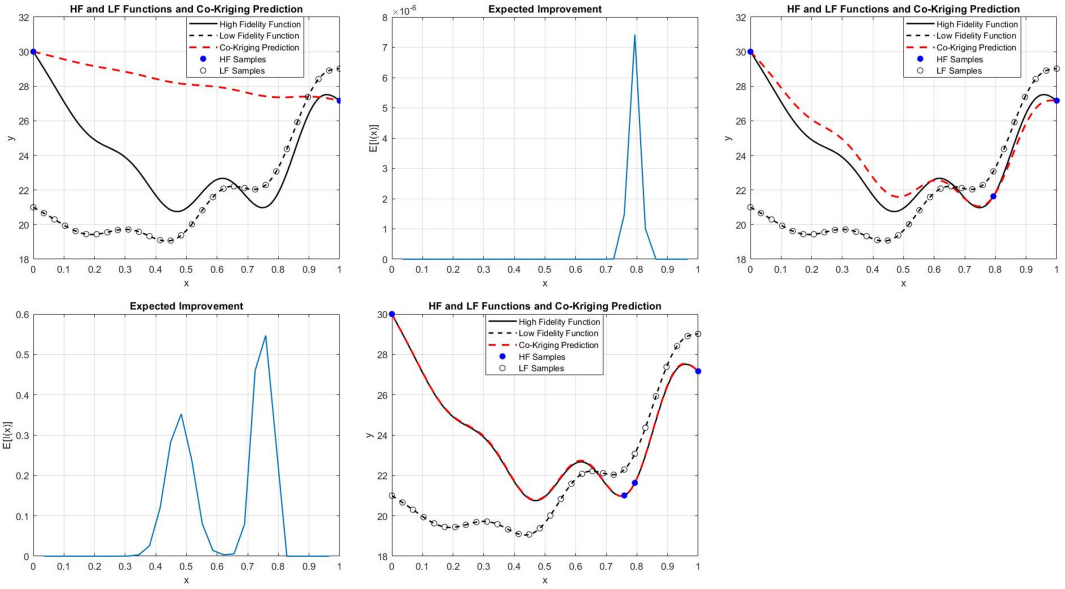

Figure 2.20: Maximum Expected Improvement sampling iterations for the third test function

the mean absolute error between the High Fidelity function and the Co-Kriging surrogate prediction is slightly higher for the Maximum E[I(x)] case, although both are below 0.25% (one may say that for the first test function the two criteria yield practically the same surrogate model). The surrogate model approximating the Forrester High Fidelity function now requires four points for near-total convergence, compared to the three points identified by the previous algorithm. Regarding the

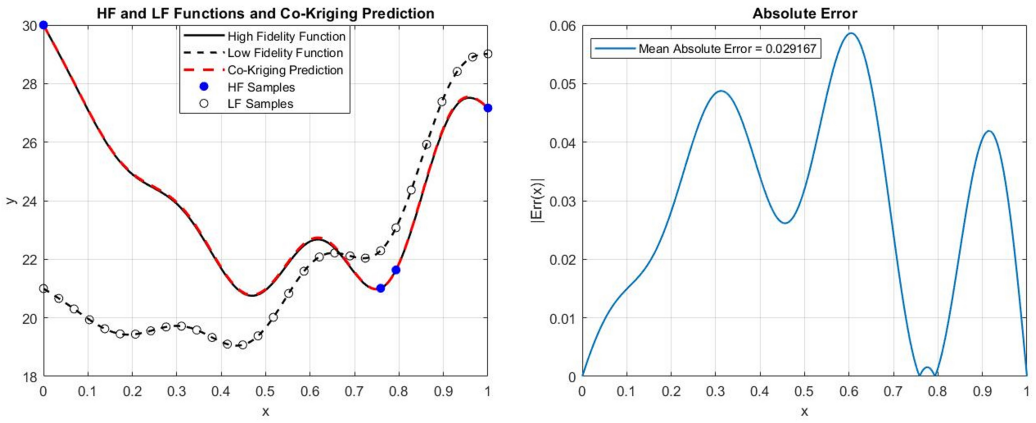

Figure 2.21: Converged Co-Kriging Model after 3 iterations, leading to a 4-points High Fidelity DOE

third test function, the Maximum Expected Improvement performs as equal as the Maximum Variance algorithm, finding a slightly higher error. In terms of the space-filling characteristic of the High Fidelity DOEs, the E[I(x)] algorithm tends to cluster the new expensive points in regions where the global or local minima of the objective functions are located; this observation is supported by Figures 2.17 and 2.21. Since the definition provided by Equation 2.3 is aimed at finding the global minimum of the surrogate, if one wants to apply the algorithm to functions presenting a global maximum, it is still possible to use Equation 2.3 by simply placing a negative sign in front of the Low and High Fidelity function values.
In conclusion, the algorithm still demonstrates good performance. As for the stopping criterion for the iterations, the same considerations reported for the variance maximization algorithm apply.

2.4.3 Maximum Information Gain criterion

Another possible algorithm for the sequential search of new points to add to the High Fidelity dataset is based on the concept of Information Gain. This involves sampling points within the design space where the available information is maximal, aiming to reduce the uncertainty of the surrogate model or obtain a better estimate of the optimal point of the expensive function [40], whose values are unknown outside the few HF points that are sampled. The use of so-called information-based methods is typical in Bayesian Optimization (BO) problems, where the expensive function is treated as a black box due to the computational costs required for various observations; the typical objective is to identify the position and value of the global optimum point of this function, minimizing the costs required to locate it as

much as possible [40]. In the context of single-fidelity Bayesian Optimization, some information-based algorithms that have been proposed include Entropy Search (ES), introduced by Hennig and Schuler [41], and Predictive Entropy Search, proposed by Hernandez-Lobato et al. [42]. These algorithms enable the definition of acquisition functions based on the information gain in terms of the position of the optimal point of the objective function $\mathbf{x^*}$ [40]. In practice, a new sample is placed at the position that maximizes the index indicating the information gain about the global maximum (or minimum) of the surrogate model resulting from the insertion of a new point into the design space. A further evolution of Entropy Search was introduced by Wang and Jegelka [43] with Max-value Entropy Search (MES), which constructs the acquisition function based on the information gain in terms of the value of the global maximum of the objective function $f(\mathbf{x^*})$, thereby reducing computational burdens. The extension of information-based methods to the multi-fidelity domain can be observed in the works of Swerski et al. [44], Zhang et al. [45], and Takeno et al. [40]. In general, despite the fact that information-based algorithms perform very well, even in relation to criteria such as Expected Improvement and are of global utility, the calculation of the Information Gain can be challenging and may require a series of complex approximations or computations that can slow down the algorithm [40].

Below, an information-based adaptive sampling criterion will be implemented, aiming to construct an index for each candidate point in the design space that evaluates the potential reduction in uncertainty resulting from the addition of a new High Fidelity point to the respective dataset. In information theory, entropy measures the uncertainty associated with a random variable [46], thus the greater the uncertainty it embodies, the greater the information that can be obtained from it. In the case of a random variable with a normal distribution and variance σ^2, a definition of entropy, or differential entropy, is provided by [46]:

$$H = \frac{1}{2} log\left(2\pi e\sigma^2\right) \tag{2.4}$$

In Co-Kriging, the values of the expensive objective function are treated as realizations of a Gaussian Process, meaning they are essentially realizations of a random variable characterized by a normal distribution with a certain mean value and variance. Thus, starting from an initial dataset of High Fidelity points, such as those considered in the implementation of the two previously discussed criteria, it is possible to construct an initial Co-Kriging surrogate and evaluate the variance s^2 at all candidate points, which constitute the Low Fidelity dataset (a nested design approach for expensive points is followed). Once this is done, the entropy distribution at various points can be evaluated using Equation 2.4, with σ^2 coinciding with the surrogate's s^2. The greater the uncertainty, the higher the initial entropy values will be.

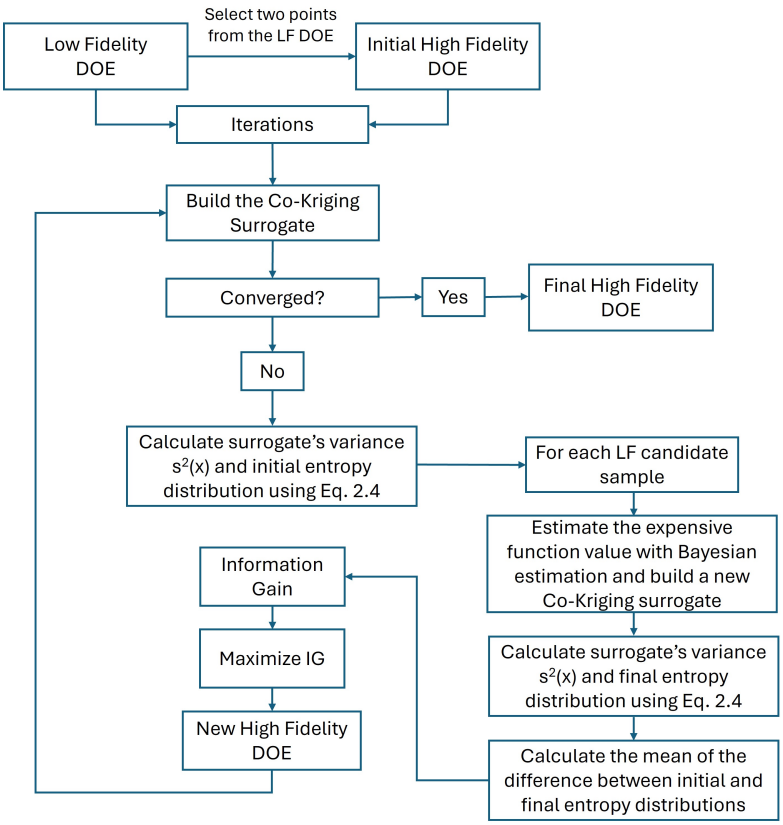

Figure 2.22: Maximum Information Gain algorithm

We now aim to determine the point whose addition would reduce the global uncertainty associated with the surrogate model. For each candidate point, we want to evaluate the entropy distribution that would result from adding that point to the High Fidelity dataset. However, this would require knowing the values of the expensive function at the various candidate points, rendering this algorithm impractical. Nonetheless, an approximation can be introduced to estimate a value for the HF function at the candidate point, using a Bayesian estimation based on the costly observations available at that iteration (the approximated high fidelity function value is estimated considering the midpoint of a 95% confidence interval for its mean value). Once a value is estimated for the candidate point to be added to the expensive dataset, a new Co-Kriging surrogate is constructed and the new variance distribution in the design space is obtained, from which the final entropy distribution can be derived. For each candidate point, the mean of the difference between the initial (H_0) and final (H_1) entropy is evaluated and the Information Gain (IG) is constructed as $mean(H_0 - H_1)$. If the addition of a certain point reduces s^2, that is, the uncertainty, whose values are typically small (thus resulting

in negative initial entropy due to the logarithm), the final entropy values will be more negative, leading to a positive IG index. The maximum IG identifies the point whose addition to the High Fidelity dataset most significantly reduces the global average uncertainty of the surrogate, thereby progressively converging the Co-Kriging model to the objective function. In this sense, the maximum IG represents an information index about the reduction in surrogate's uncertainty.
Note that if the candidate point selected is a High Fidelity sample point, the Information Gain there will be nonexistent, because there the objective function value is already known and no uncertainty is present there. Figure 2.22 shows a general concept for the Maximum IG algorithm we want to implement.
The Maximum Information gain will be applied to the test functions previously described, on which the Maximum Variance and Maximum Expected Improvement criteria have been performed. The results are shown below:

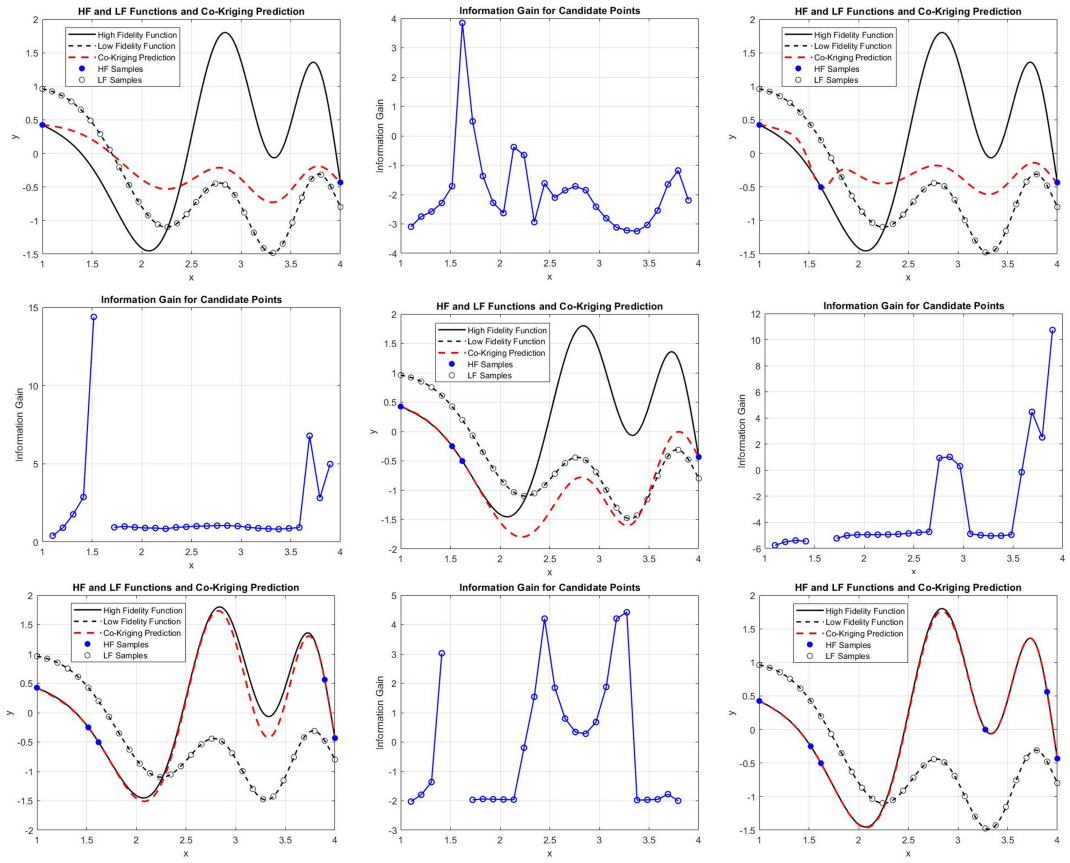

Figure 2.23: Maximum Information Gain sampling iterations for the first test function

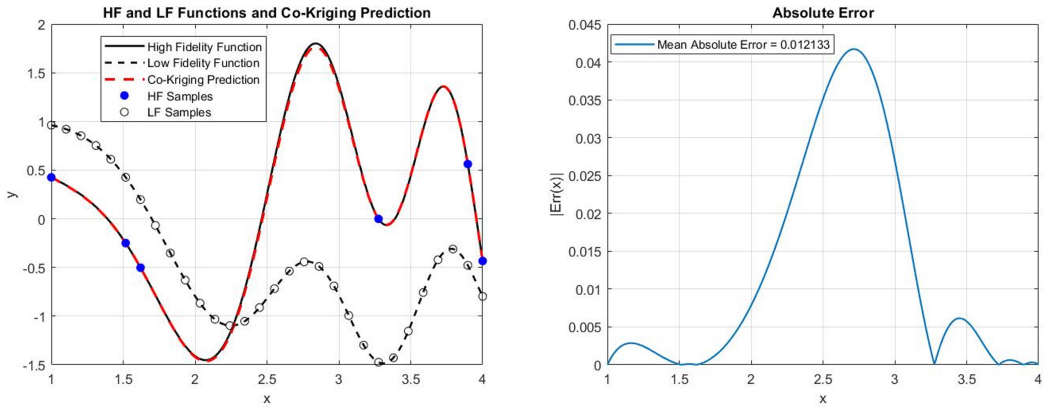

Figure 2.24: Converged Co-Kriging Model after 5 iterations, leading to a 6-points High Fidelity DOE

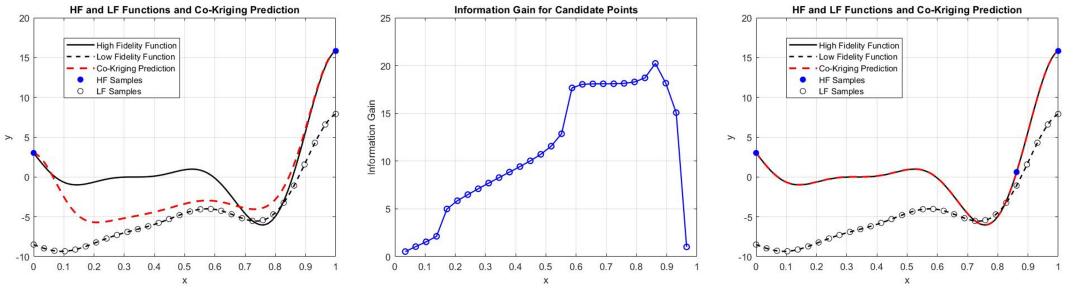

Figure 2.25: Maximum Information Gain sampling iterations for the Forrester function

From the figures presented, it is evident that the maximum Information Gain algorithm results in better performance compared to Expected Improvement, especially for the Forrester function and the third test function, by reducing the number of necessary High Fidelity points while keeping the error extremely low. For the first test function, the final number of points in the expensive dataset is the same, but the error is higher. In any case, it can be seen that the performance of the individual algorithms is still dependent on the High and Low Fidelity functions studied, but all three remain highly effective for implementing the Co-Kriging surrogate model. Below, several comparison tables of the various algorithms proposed for the three test functions are presented.

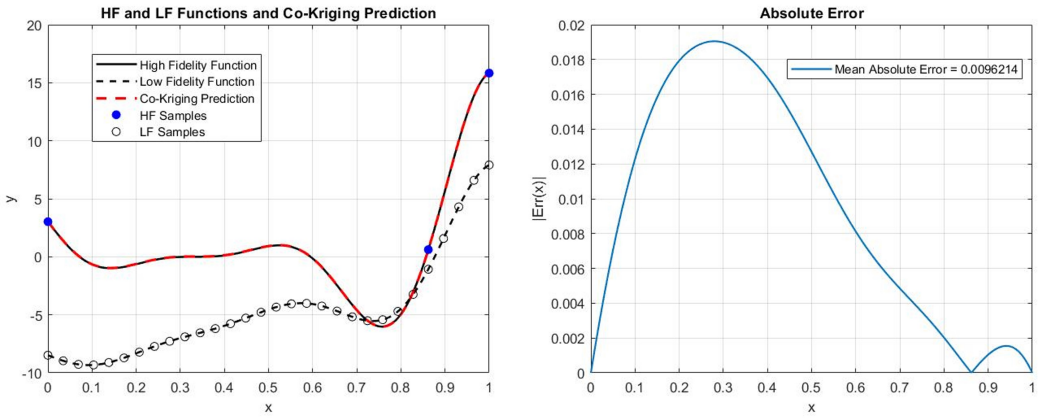

Figure 2.26: Converged Co-Kriging Model after 2 iteration, leading to a 3-points High Fidelity DOE

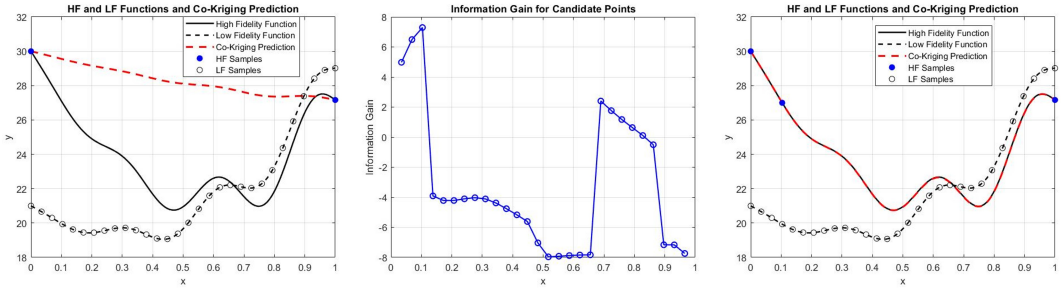

Figure 2.27: Maximum Information Gain sampling iterations for the third test function

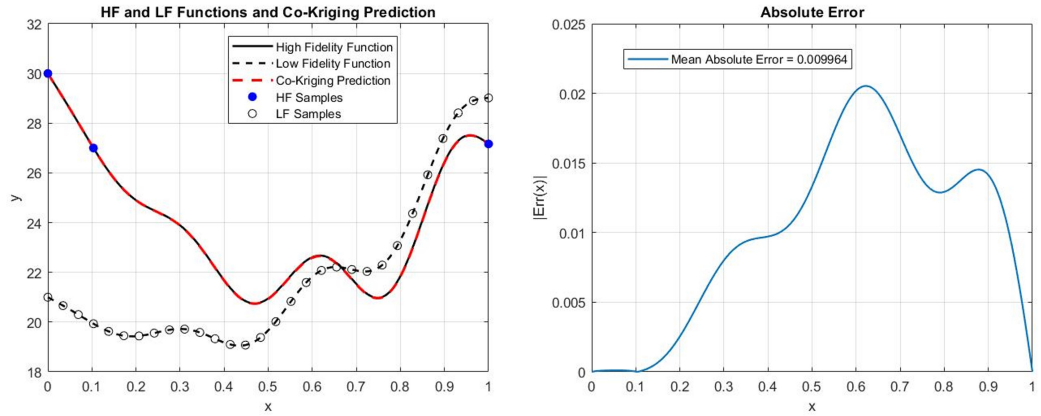

Figure 2.28: Converged Co-Kriging Model after 2 iteration, leading to a 3-points High Fidelity DOE

Sampling Criterion	LF Samples	HF Samples (convergence)	Mean Absolute Error (%)
Max Variance	30	6	0.162
Max E[I(x)]	30	6	0.212
Max IG(x)	30	6	1.213

Table 2.1: Results comparison between the three different algorithms for the first test function

Sampling Criterion	LF Samples	HF Samples (convergence)	Mean Absolute Error (%)
Max Variance	30	3	0.885
Max E[I(x)]	30	4	3.935
Max IG(x)	30	3	0.962

Table 2.2: Results comparison between the three different algorithms for the Forrester function

Sampling Criterion	LF Samples	HF Samples (convergence)	Mean Absolute Error (%)
Max Variance	30	4	0.581
Max E[I(x)]	30	4	2.916
Max IG(x)	30	3	0.996

Table 2.3: Results comparison between the three different algorithms for the third test function

2.5 Two-variable adaptive sampling implementation

The algorithms demonstrated for the single-variable case are now implemented for two-variable test functions. Extending this approach to functions with more than two variables is straightforward, following the same methodology presented below and utilizing the ooDACE toolbox for Matlab.

2.5.1 Maximum Variance/MSE criterion

As in the case of a single variable, the implementation for two-variable functions follows the logical framework presented in Figure 2.5. Starting from the Low Fidelity Design of Experiment, which can be generated using a factorial approach or by constructing a space-filling Latin Hypercube as explained in Section 2.2, four initial points are chosen for the High Fidelity dataset. These points typically represent the extremes of the design space in the x_1-x_2 plane. Subsequently, an adaptive algorithm is employed to calculate the model's variance at the sample points constituting the Low Fidelity DOE (we want the High Fidelity DOE to be nested).
For constructing the Low Fidelity dataset, one can implement a space-filling Latin Hypercube that optimizes the Morris-Mitchell criterion, as described in Section 2.2. To achieve this, the Matlab function *bestlh.m* provided by Forrester in [5] can be utilized. For instance, if one aims to obtain a space-filling DOE with 100 points that also includes the design space extremes, the function *bestlh.m* can be called to generate a Latin Hypercube with 96 points, followed by adding the four corner points of the x_1-x_2 plane. Regarding the High Fidelity DOE, starting with these four corner points, at each step, the position within the LF points where the Co-Kriging surrogate, constructed with the data available at that iteration, exhibits the highest variance s^2 is identified and a new HF sample is placed there. The iterations can be halted by following the guidelines proposed for the single-variable case; in this case the weighted mean relative percentage error between the High Fidelity function and the Co-Kriging surrogate at the 100 Low Fidelity points will define the stopping criterion. Alternatively, the cycle can be interrupted when the difference between the surrogate values at iteration *k+1* and iteration k falls below a certain threshold, a method useful when the analytical expressions of the expensive and cheap functions are not available.

For the two-variable implementation, the test functions that will be leveraged are the following ones:

- **Himmelblau Function [47] -** x_1**,** $x_2 \in$ **[-4,4]**:

$$f_e(x_1, x_2) = (x_1^2 + x_2 - 11)^2 + (x_2^2 + x_1 - 7)^2$$
$$f_c(x_1, x_2) = f_e(0.5x_1, 0.8x_2) + x_2^3 - (x_1 + 1)^2$$

- **Booth Function [47] -** x_1**,** $x_2 \in$ **[-10,10]**:

$$f_e(x) = (x_1 + 2x_2 - 7)^2 + (2x_1 + x_2 - 5)^2$$
$$f_c(x) = f_e(0.4x_1, x_2) + 1.7x_1x_2 - x_1 + 2x_2$$

- **Adjustable Branin Function [48] -** $x_1 \in$ **[-5,10]**, $x_2 \in$ **[0,15]**, **a** $\in$ **[0,1]**:

$$f_e(x) = \left(x_2 - (5.1\frac{x_1^2}{4\pi^2}) + \frac{5x_1}{\pi} - 6\right)^2 + \left(10cos(x_1)(1 - \frac{1}{8\pi})\right) + 10$$
$$f_c(x) = f_e(x_1, x_2) - (a + 0.5)\left(x_2 - (5.1\frac{x_1^2}{4\pi^2}) + \frac{5x_1}{\pi} - 6\right)^2$$

Starting from the Himmelblau Function, a representation of the two fidelities can be seen in Figure 2.29.

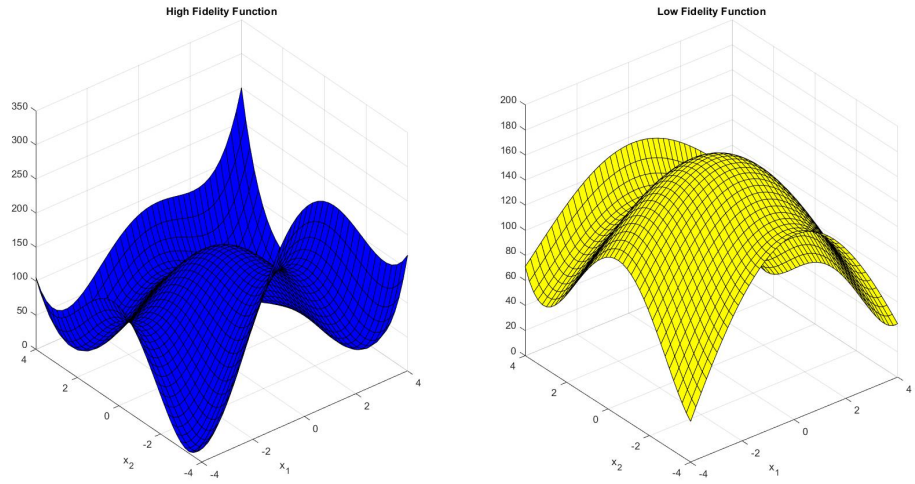

Figure 2.29: High Fidelity (left) and Low Fidelity (right) Himmelblau Function defined in x_1,$x_2 \in$ [-4,4]

As for the Low Fidelity Design of Experiment, both a space-filling Hypercube and a factorial approach will be explored. Starting from the first case, a 100 points DOE is generated using the *bestlh.m* Matlab function [5], including the four corner points of the domain. The resulting sampling plan is shown in Figure 2.30.
The initial High Fidelity dataset is then constructed sampling the four corner points and evaluating the expensive function response thereof, leading to a four-points Design of Experiment. The iterative part of the algorithm can now begin,

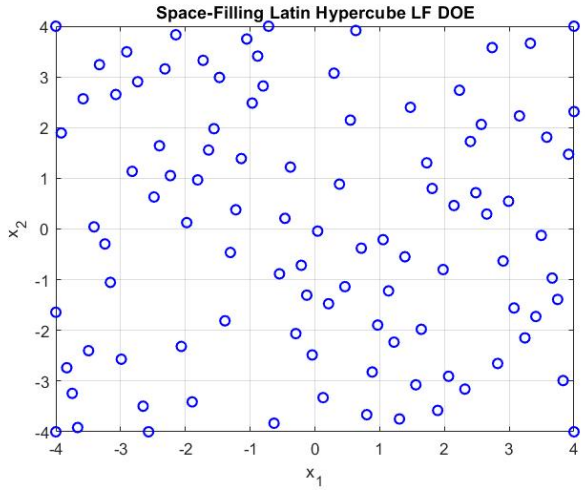

Figure 2.30: Space-Filling Low Fidelity Latin Hypercube DOE

generating the Co-Kriging surrogate model in order to obtain the variance s^2 in the different Low Fidelity locations (nested approach). The point characterized by the maximum variance value is then sampled and a new surrogate model is generated; the iterations can be stopped by analyzing the weighted relative percentage error. Specifically, the relative percentage error between the High Fidelity function and the Co-Kriging surrogate at the Low Fidelity points is evaluated using the following formula:

$$Err\% = 100 \cdot \frac{|f_e(x_1, x_2) - f_{ck}(x_1, x_2)|}{|f_e(x_1, x_2)| + \epsilon} \tag{2.5}$$

where ϵ is a small constant to avoid division by zero when the High Fidelity function value is zero at that point. This relative percentage error is then multiplied by an importance function, which represents the weight associated with the relative percentage error at various points. In regions where the objective function values are small, the weight of the error will be lower, thus preventing extremely large values when both the High Fidelity function and surrogate are near zero. As an importance function, one can consider the absolute value of the expensive function's values. This results in a weighted error and a weighted mean value is calculated using the formula:

$$\text{Weighted Mean Relative Error } (\%) = \frac{\sum_{i=1}^{n} \left(100 \times \frac{|f_e - f_{ck}|_i}{|f_e|_i + \epsilon} \times |f_e|_i\right)}{\sum_{i=1}^{n} |f_e|_i} \tag{2.6}$$

This serves as the stopping criterion for the iterations, allowing for a more robust representation of the error, without excessive overshoots when calculating the error

associated with target function values that are much smaller than the maximum value. The stopping criterion can be considered as verified when the weighted mean relative error is less than 3% and the final High Fidelity dataset is obtained.
For the generation of the Co-Kriging surrogate model using Matlab's ooDACE toolbox, it is advisable to scale the data, both in terms of the input variables (x_1 and x_2) and the function values, to ensure the correct generation of the surrogate by the *oodacefit.m* function. Specifically, for the Himmelblau Function, x_1 and x_2 should be scaled such that x_1, x_2 are in the range [-1, 1]. The High Fidelity and Low Fidelity functions will be scaled by dividing by the maximum value that these functions assume within the defined domain. Consequently, the variance and error values will also be scaled accordingly, but this does not affect the final result.

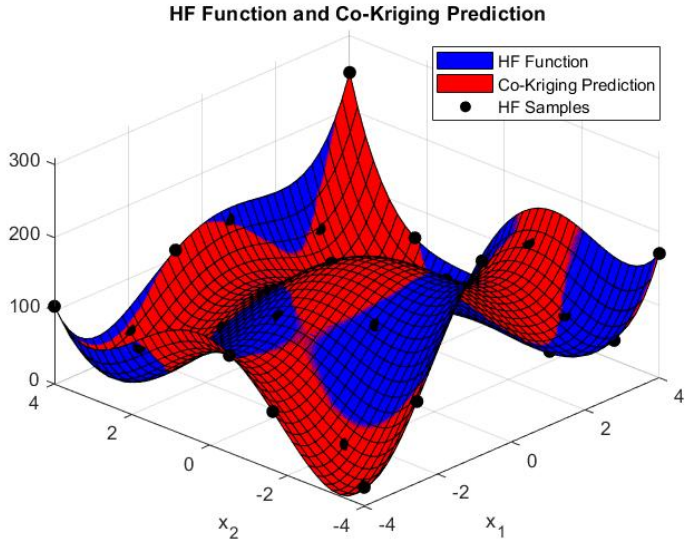

Figure 2.31: High Fidelity Function and Co-Kriging prediction for the Himmelblau Function with 27 HF samples

For the Himmelbalu Function, convergence is obtained after 24 iterations, leading to a 27 High Fidelity points DOE, shown in Figure 2.31. Figure 2.32 shows the surrogate's scaled variance distribution at iteration 12, when the High Fidelity Design of Experiment is characterized by 15 sample points, the relative percentage error obtained when the stopping criterion is satisfied and the final High Fidelity DOE with respect to the Low Fidelity samples. The mean error after 24 iterations reaches 0.117%, which means that the surrogate model approximates almost perfectly the expensive target function. Observing Figure 2.32, one can see that there is a peak in the relative percentage error, characterized by a really high value (it

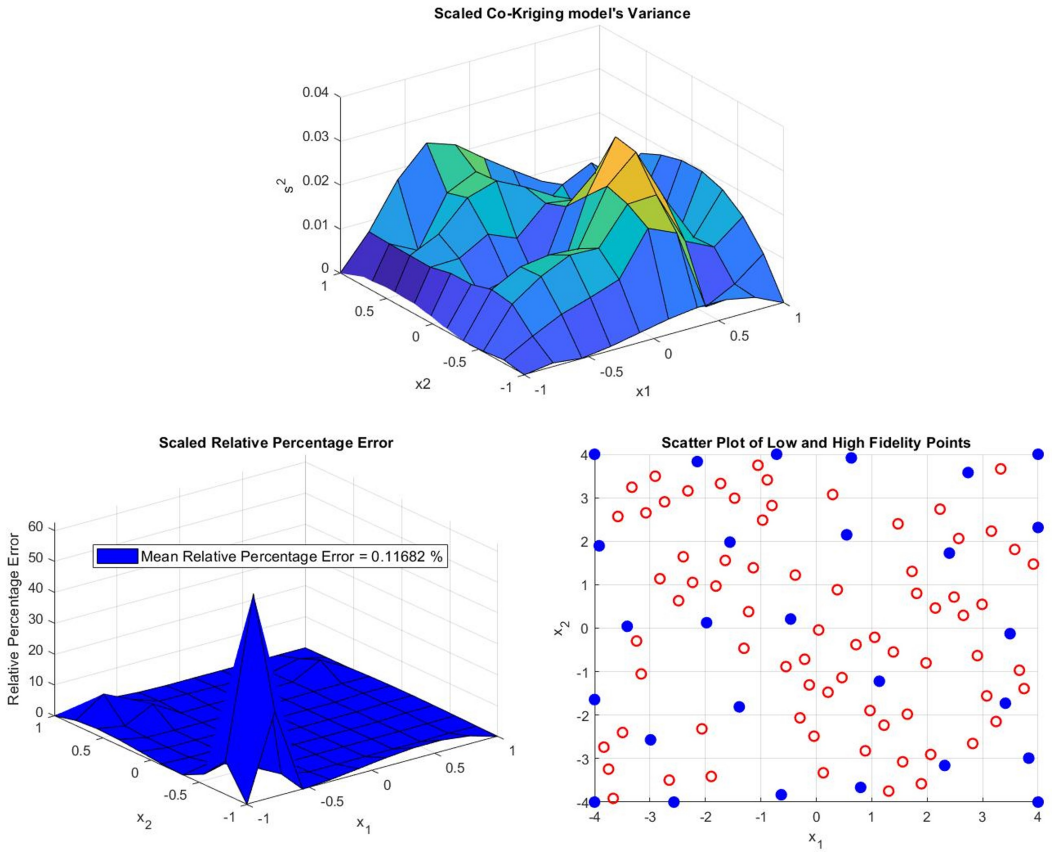

Figure 2.32: Co-Kriging model's scaled s^2 at iteration 12 (Up), final relative percentage error (Left) and final High Fidelity samples distribution in the design space (Right)

)belongs to a point where the High Fidelity function is close to zero). It is precisely for this reason that the weighted mean error was introduced, in order to give less weight to the errors caused by small function values compared to the expensive function's maximum. Moreover, one can see that the Maximum Variance algorithm, for the Himmelblau Function, leads to a space-filling High Fidelity dataset, thus exploring uniformly the design space. As a result, it can be observed that even in the case of functions of two variables, the maximum variance criterion leads to highly satisfactory results. In this instance, the Low Fidelity sampling plan involved the implementation of a DOE based on a space-filling Latin Hypercube. One might then wonder how the results would vary if, instead, a factorial approach were chosen for the sampling of cheap points. Aiming to maintain the number of Low Fidelity points at 100, while also including the points at the extremes of the design space, the factorial DOE represented in Figure 2.33 is obtained:

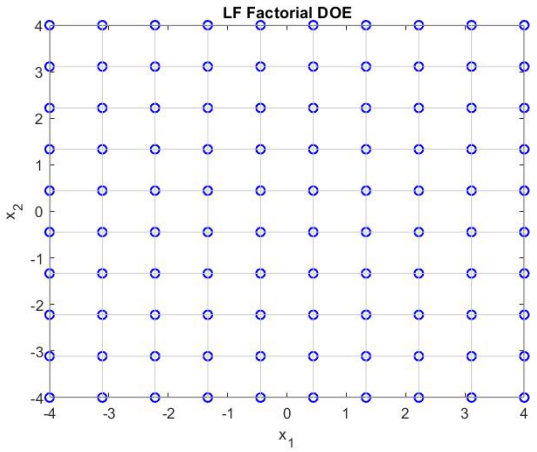

Figure 2.33: Low Fidelity factorial Design of Experiment

Applying the same algorithm with the 100 points Low Fidelity factorial sampling plan for the Himmelblau Function yields a 22 points final High Fidelity Design of Experiment, thus reducing the number of expensive function observations and the required computational cost. The mean relative percentage error obtained in this case is higher, but still under the 3% threshold, implying a highly accurate approximation with less expensive points; the results are shown in Figure 2.34 and 2.35.

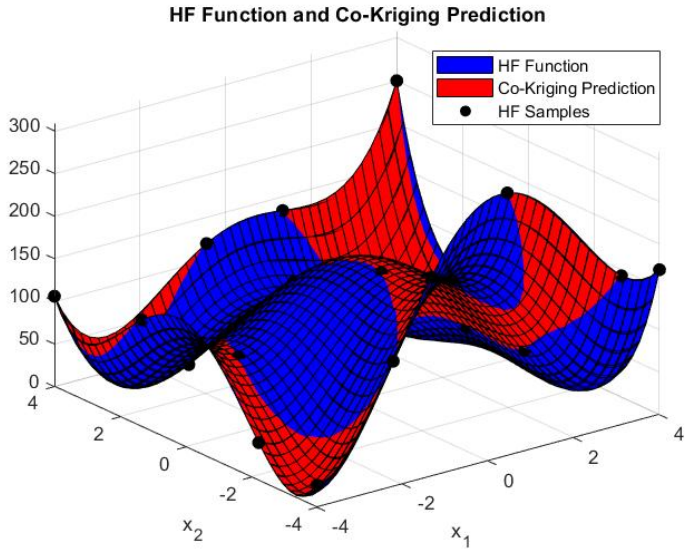

Figure 2.34: 22 points Co-Kriging surrogate model when a 100 points factorial LF DOE is employed

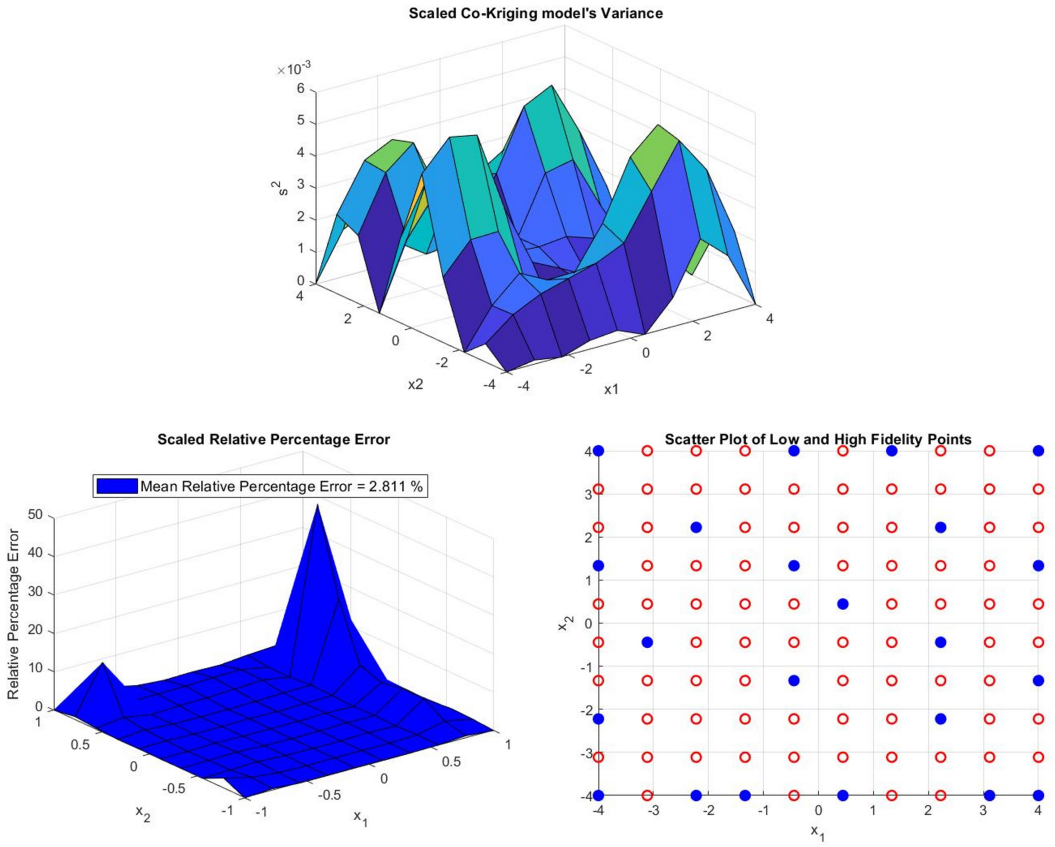

Figure 2.35: Co-Kriging model's scaled s2 at iteration 12 (Up), final relative percentage error (Left) and final High Fidelity samples distribution in the design space (Right) when a factorial LF DOE is employed

To avoid overloading the discussion, for the next two test functions, the implementation of the algorithm will only be examined considering the Low Fidelity Design of Experiment obtained by constructing a space-filling 100 points Latin Hypercube using the *bestlh.m* function [5], as depicted in Figure 2.30.

We now move on to the second test function listed above, the Booth Function [47], for which the behaviours of the High and Low fidelity functions are represented in Figure 2.36.
Once again, a 100 points space-filling Latin Hypercube is generated and the algorithm is applied until the weighted mean relative error falls below the 3% threshold. The Low Fidelity Design of Experiment will be shown in Figure 2.38 along the High Fidelity selected points when convergence is reached. The Maximum Variance criterion for the Booth Function gives the results shown in Figure 2.37 and, of course, Figure 2.38.

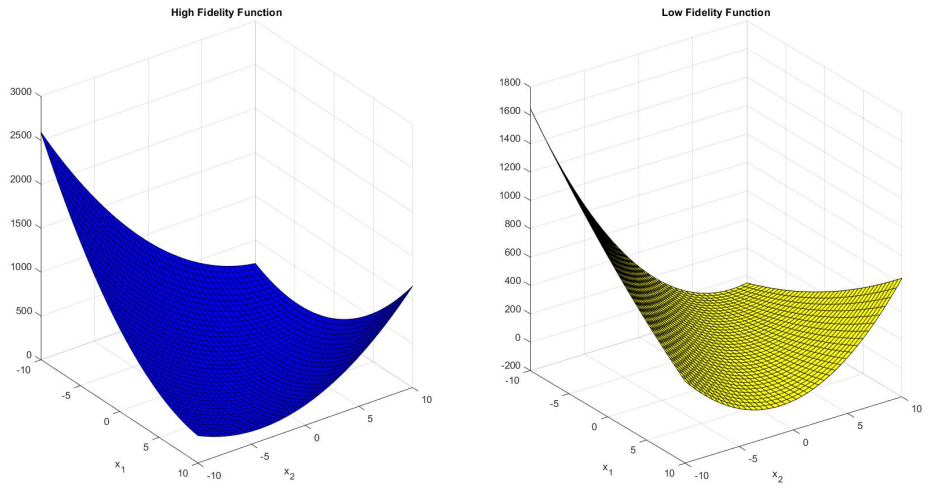

Figure 2.36: High Fidelity (left) and Low Fidelity (right) Booth Function defined in x_1,$x_2 \in$ [-10,10]

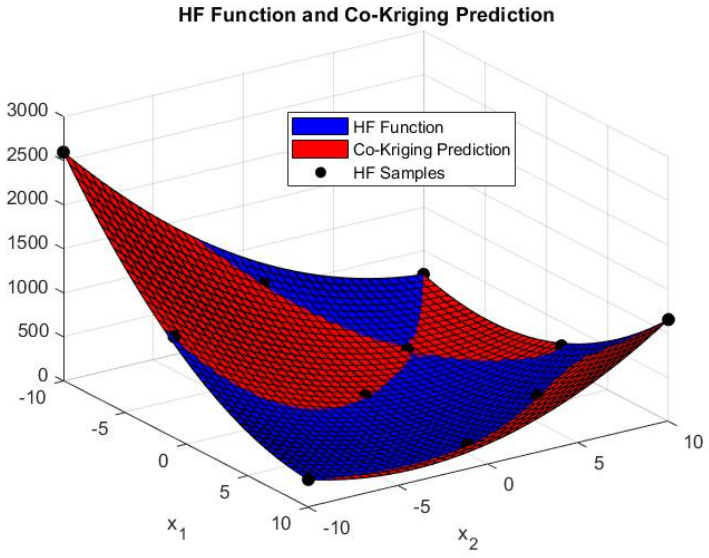

Figure 2.37: High Fidelity Function and Co-Kriging prediction for the Booth Function with 11 HF samples

In this case, the stopping criterion is satisfied with just 11 High Fidelity points, leading to a weighted mean percentage error of the order of 0.555%, giving a practically perfect approximation of the target expensive function. Once again, the

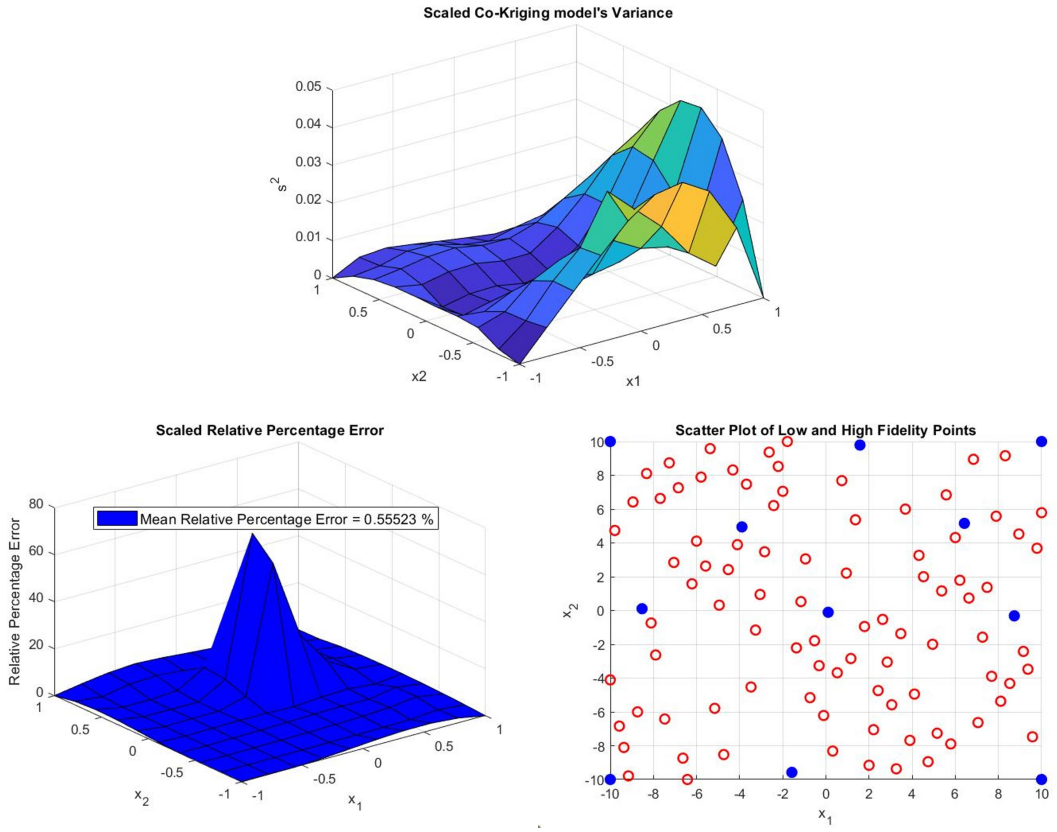

Figure 2.38: Co-Kriging model's scaled s^2 at iteration 4 (Up), final relative percentage error (Left) and final High Fidelity samples distribution in the design space (Right)

resulting High Fidelity DOE is space-filling. As previously mentioned, the case which provides for the implementation of a factorial Low Fidelity DOE is omitted to streamline the discussion; the steps to follow if one wishes to study the effect of a factorial sampling plan are the same as those presented for the Himmelblau Function.
The final function to be analyzed is the Adjustable Branin Function [42], whose two fidelity levels are depicted in Figure 2.39. For the Low Fidelity function the parameter a is set to 0.2.
Leveraging the 100 points space-filling Latin Hypercube as the Low Fidelity Design of Experiment, which will be reported in Figure 2.40, the algorithm is applied and the obtained results are shown in Figure 2.40 and 2.41.
Particularly, the stopping criterion is satisfied after 20 iterations, obtaining a final 23 points High Fidelity dataset. In this case, the weighted mean relative error is of the order of 2.553%, higher than the other analyzed functions, but still low, giving

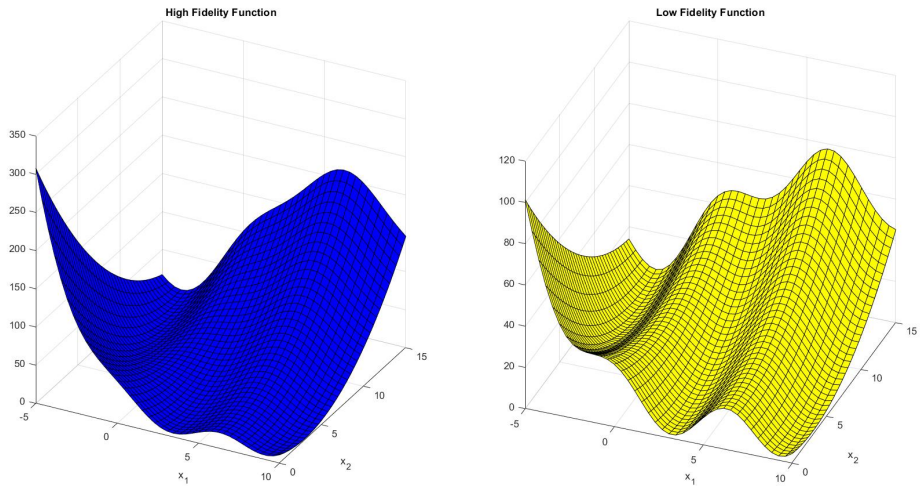

Figure 2.39: High Fidelity (left) and Low Fidelity (right) Branin Function defined in $x_1 \in$ [-5,10],$x_2 \in$ [0,15]

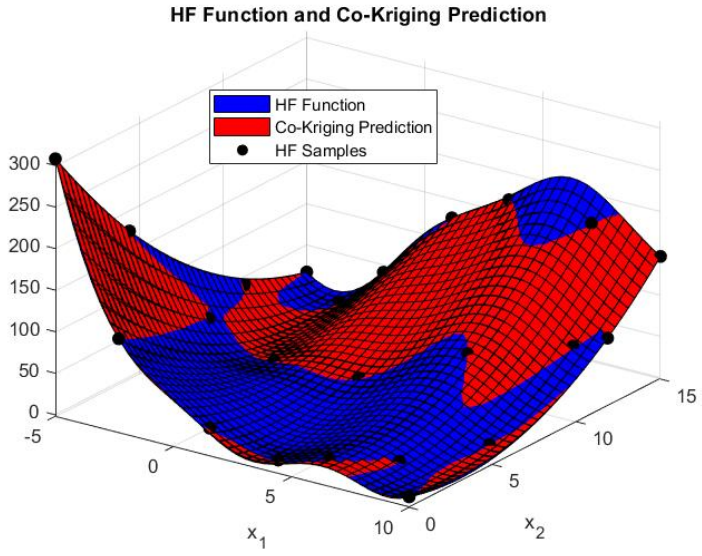

Figure 2.40: High Fidelity Function and Co-Kriging prediction for the Branin Function with 23 HF samples

an accurate objective function approximation in the design space. Finally, as well as in the other previously studied cases, the Maximum Variance algorithm gives a High Fidelity space-filling DOE.

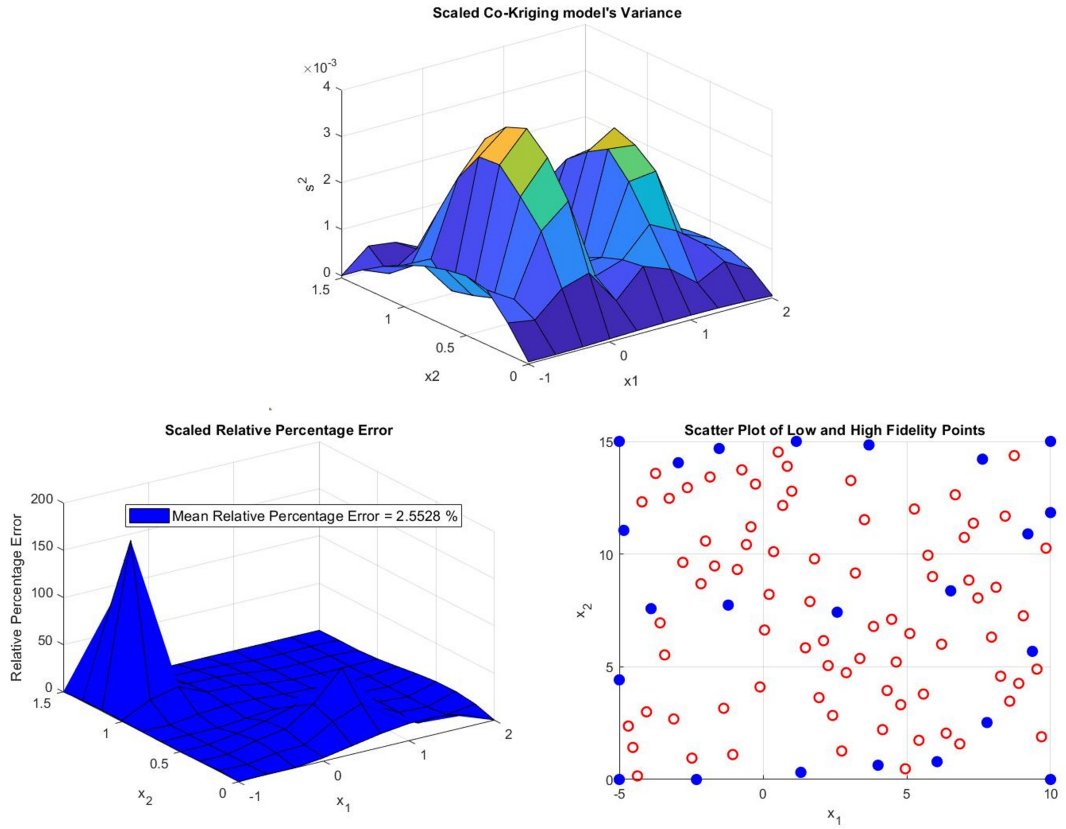

Figure 2.41: Co-Kriging model's scaled s^2 at iteration 12 (Up), final relative percentage error (Left) and final High Fidelity samples distribution in the design space (Right)

In the next section, the Maximum Expected Improvement criterion is extended to the two-variable case, validating it by considering the three test functions analyzed.

2.5.2 Maximum Expected Improvement criterion

The analysis of Expected Improvement, introduced in the context of single-variable functions, can be extended to multi-variable functions while maintaining the same logical framework presented in Figure 2.15. Specifically, the initial High Fidelity sampling plan will consist of the four points defining the extremes of the x_1-x_2 plane. At each iteration, the variance s^2 of the Co-Kriging surrogate model is utilized to calculate the Expected Improvement for all candidate points (it is worth noting that a nested approach is adopted for expensive points, hence the candidate points are those that constitute the Low Fidelity DOE). The point exhibiting the maximum value of E[I(x_1,x_2)] will be selected and added to the High Fidelity

dataset, thereby allowing the surrogate model to be updated and the iterations to proceed. Regarding the stopping criterion, which provides a threshold between the exploration of the design space and the effective exploitation of the surrogate, all considerations discussed in previous sections apply; in this case as well, the algorithm is terminated when the weighted mean relative error (Equation 2.5) is less than 3%. The functions to which the algorithm will be applied are the same as those reported in the previous section, namely

- **Himmelblau Function [47] -** x_1**,** $x_2 \in$ **[-4,4]**:

$$f_e(x_1, x_2) = (x_1^2 + x_2 - 11)^2 + (x_2^2 + x_1 - 7)^2$$
$$f_c(x_1, x_2) = f_e(0.5x_1, 0.8x_2) + x_2^3 - (x_1 + 1)^2$$

- **Booth Function [47] -** x_1**,** $x_2 \in$ **[-10,10]**:

$$f_e(x) = (x_1 + 2x_2 - 7)^2 + (2x_1 + x_2 - 5)^2$$
$$f_c(x) = f_e(0.4x_1, x_2) + 1.7x_1x_2 - x_1 + 2x_2$$

- **Adjustable Branin Function [48] -** $x_1 \in$ **[-5,10],** $x_2 \in$ **[0,15], a** $\in$ **[0,1]**:

$$f_e(x) = \left(x_2 - (5.1\frac{x_1^2}{4\pi^2}) + \frac{5x_1}{\pi} - 6\right)^2 + \left(10cos(x_1)(1 - \frac{1}{8\pi})\right) + 10$$
$$f_c(x) = f_e(x_1, x_2) - (a + 0.5)\left(x_2 - (5.1\frac{x_1^2}{4\pi^2}) + \frac{5x_1}{\pi} - 6\right)^2$$

For the Low Fidelity dataset, both a space-filling Latin Hypercube and a factorial sampling approach will be taken into consideration for the Himmelblau Function, allowing for a results comparison for the effect of the two different cheap Design of Experiments.

Starting from the Himmelblau Function, the maximum Expected Improvement algorithm is employed to find the new High Fidelity data points; considering the 100 points Latin Hypercube sampling plan shown in Figure 2.30, the stopping criterion is satisfied after 23 iterations, leading to a 26 points expensive dataset. The resulting Co-Kriging surrogate model is depicted in Figure 2.42, while Figure 2.43 shows the Expected Improvement distribution in the design space for the 12° iteration, the relative percentage error when convergence is obtained and the position of the sampled High Fidelity points. It is worth noting that the weighted mean relative error achieved is about 0.433%. By comparing the results obtained for the Himmelblau Function using the Maximum Variance and Maximum Expected Improvement, it is immediately evident that the second algorithm leads to a reduction of the number of High Fidelity samples from 27 to 26, while keeping the relative error below 0.5%, thus obtaining a near perfect High Fidelity function

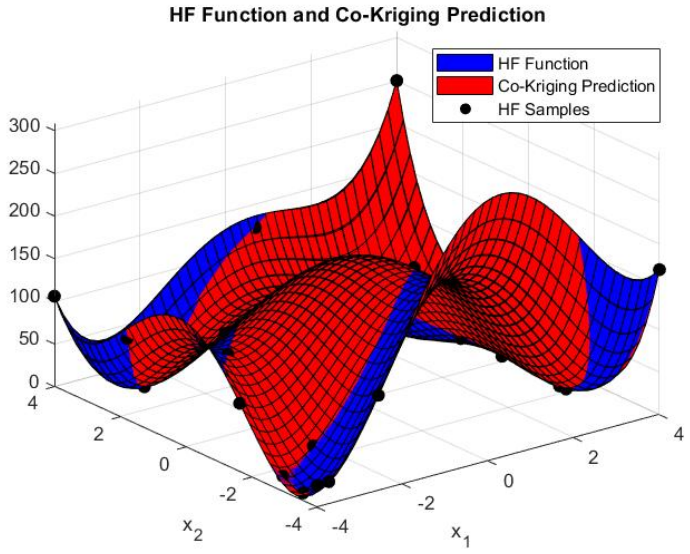

Figure 2.42: High Fidelity Function and Co-Kriging prediction for the Himmelblau Function with 26 HF samples

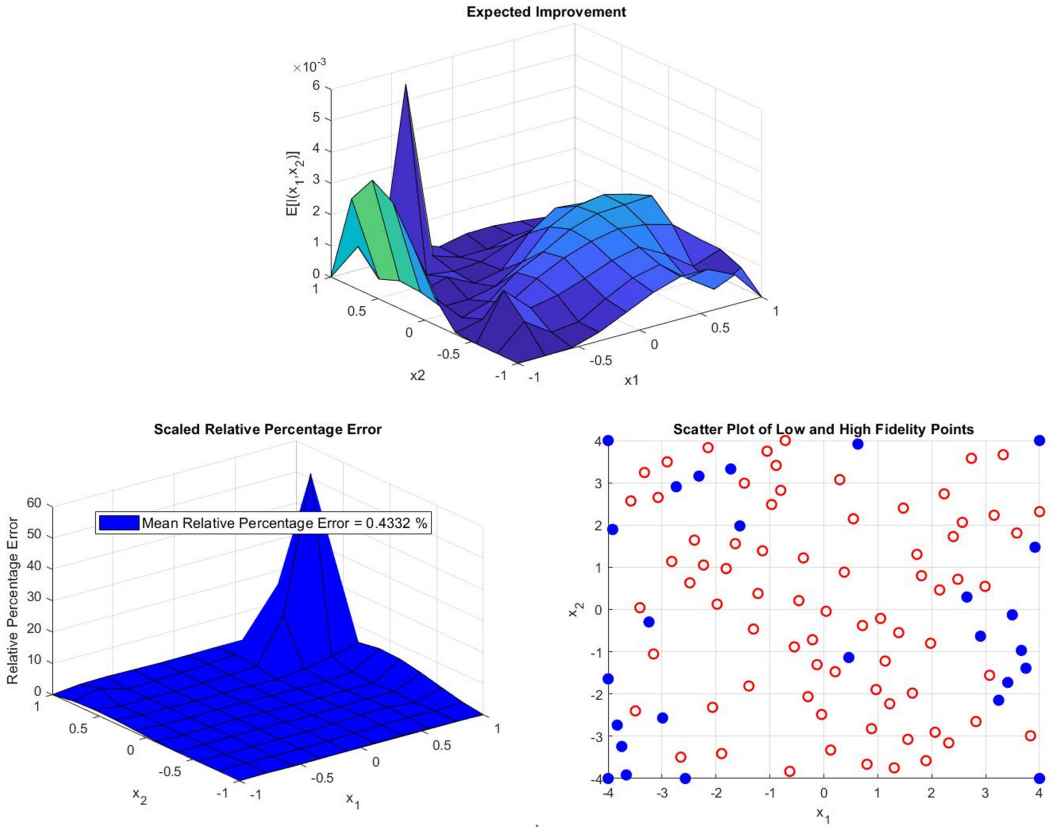

Figure 2.43: Co-Kriging model's Expected Improvement at iteration 12 (Up), final relative percentage error (Left) and final High Fidelity samples distribution in the design space (Right)

approximation with one point less (the reduction in computational cost is little, but still an improvement is obtained). Another thing that can be pointed out is the fact that the Expected Improvement algorithm, in this case, provides a High Fidelity Design of Experiment that no longer exhibits the space-filling characteristic of the dataset resulting from the variance maximization algorithm. Instead, it tends to concentrate the points near the local minima of the Himmelblau Function, as shown in Figure 2.43. This may be because the Expected Improvement metric measures the potential for finding a point to add to the High Fidelity dataset that would enhance the surrogate model's estimate of the global minimum of the objective function, striving at each step to estimate a value increasingly closer to it.
Considering instead the Low Fidelity Design of Experiment obtained by following a factorial approach for sampling the cheap points, as shown in Figure 2.33, the results obtained from the application of the Maximum Expected Improvement algorithm are depicted in Figure 2.44 and 2.45.

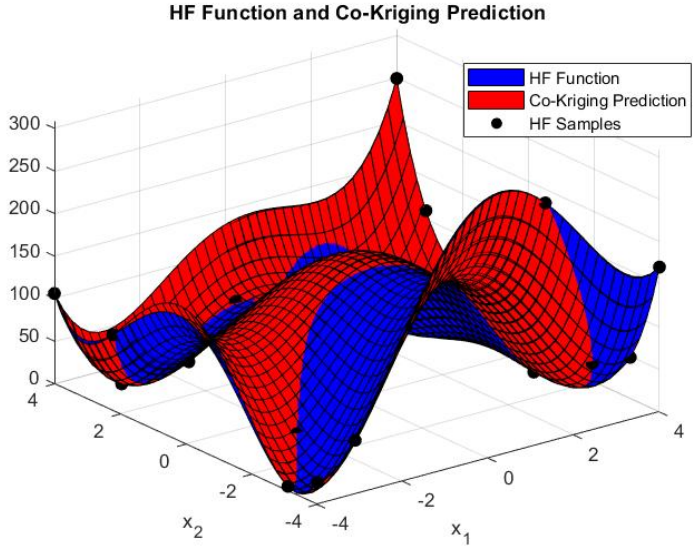

Figure 2.44: 23 points Co-Kriging surrogate model when a 100 points factorial LF DOE is employed

In this case, 23 High Fidelity points are required to satisfy the stopping criterion, however, the weighted mean relative error is higher than the Latin Hypercube case, although still of the order of 2%. As for the Maximum Variance case, one case see that the factorial approach leads to a reduction of the High Fidelity samples, which entails a lower computational cost required to obtain a good surrogate model. Figure 2.45 reaffirms that the Expected Improvement algorithm tends to sample new points in the regions where the expensive function's minima are located.

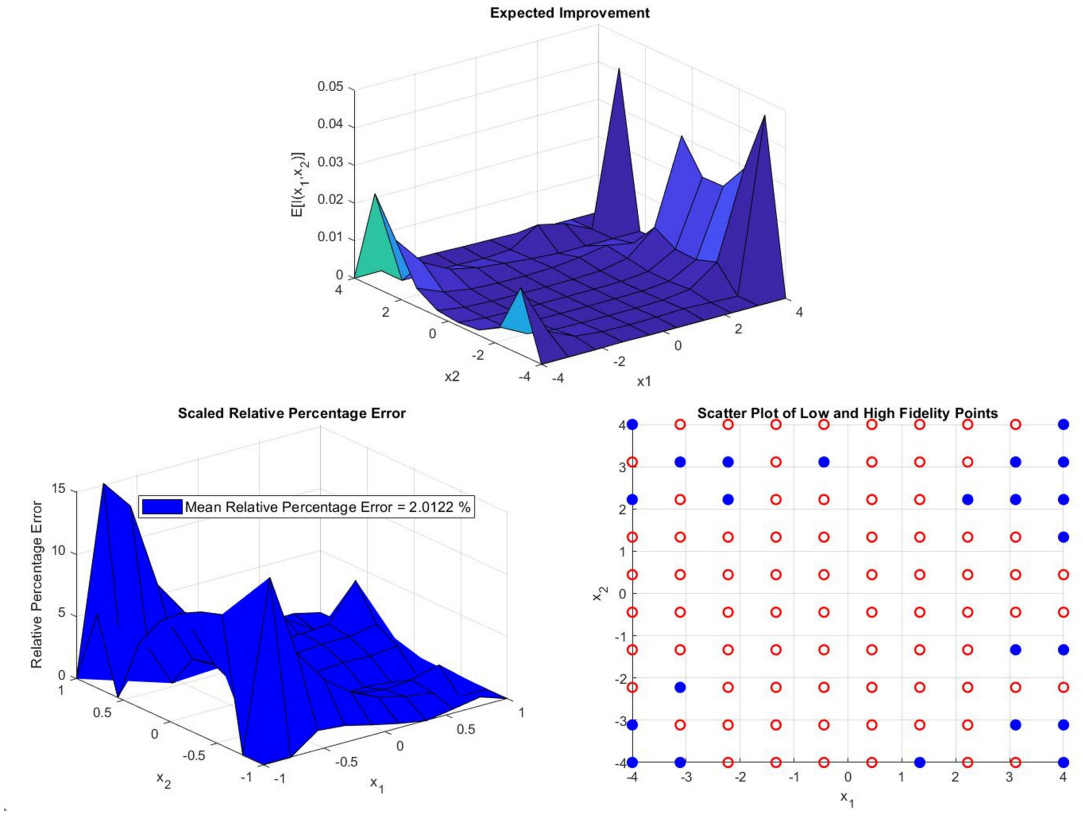

Figure 2.45: Co-Kriging model's Expected Improvement at iteration 12 (Up), final relative percentage error (Left) and final High Fidelity samples distribution in the design space (Right) when a factorial LF DOE is employed

We now take into consideration the second test function presented, i.e. the Booth Function: from now on, only the Latin Hypercube approach will be leveraged for the Low Fidelity dataset creation, setting aside the comparison with the factorial sampling plan. Thus, the Maximum Expected Improvement algorithm is performed on the Booth Function, obtaining the results shown in Figure 2.46 and 2.47. For the Booth Function, the Expected Improvement algorithm performs exceptionally well, achieving a High Fidelity Design of Experiment consisting of 10 points, one fewer than what was obtained using the Maximum Variance criterion. The weighted mean relative error is around 1%, making the result highly satisfactory, with the Co-Kriging surrogate model converging almost perfectly to the expensive function. In this case, it is also noticeable that the points added to the High Fidelity dataset tend to concentrate near the single minimum point of the function, leaving a large portion of the space unexplored. However, this does not prevent the surrogate model from accurately approximating the objective function.

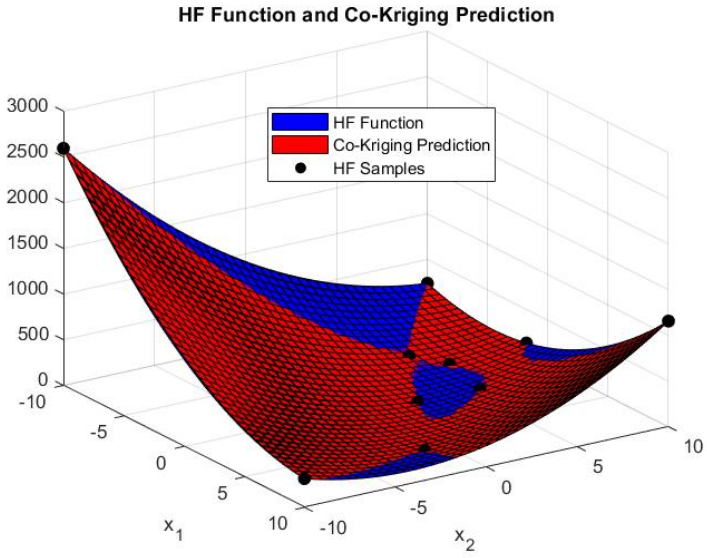

Figure 2.46: High Fidelity Function and Co-Kriging prediction for the Booth Function with 10 HF samples

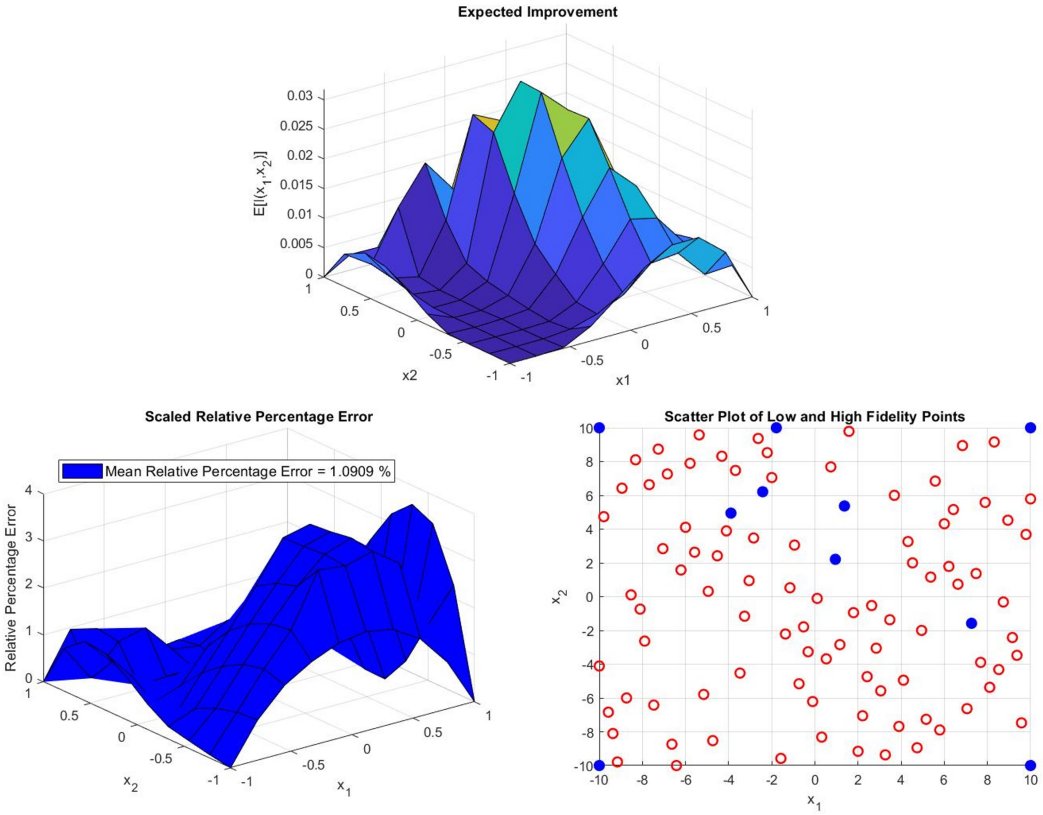

Figure 2.47: Co-Kriging model's Expected Improvement at iteration 4 (Up), final relative percentage error (Left) and final High Fidelity samples distribution in the design space (Right)

Finally, the algorithm can be applied to the last test function, the Branin Function, with the results presented in Figures 2.48 and 2.49.

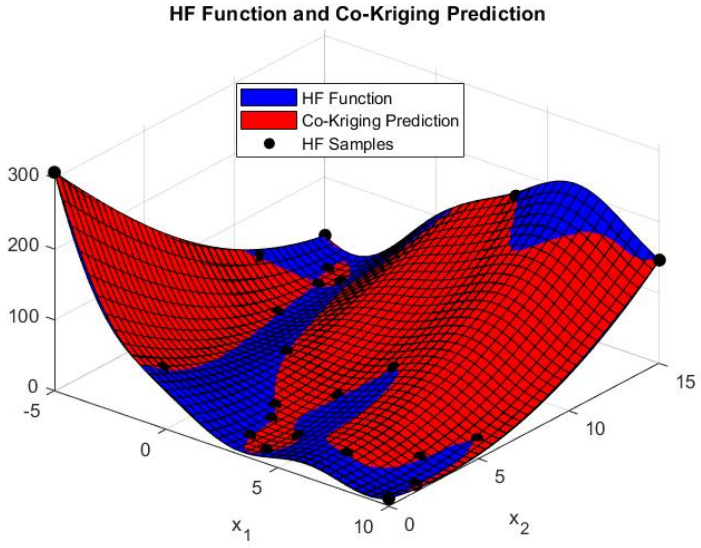

Figure 2.48: High Fidelity Function and Co-Kriging prediction for the Branin Function with 24 HF samples

For this final case, the High Fidelity points constituting the final dataset are 24, one more than what was found using the maximum s^2 algorithm of the Co-Kriging surrogate model. Additionally, the dataset obtained with the Expected Improvement criterion results in a weighted mean relative error of approximately 2.497%, slightly lower than that achieved with the previous criterion. The performances of the two algorithms applied to the Branin Function are practically comparable, as is also the case for the other two functions, i.e. Himmelblau and Booth. However, for these latter two functions, the Expected Improvement algorithm achieves a slight improvement over the Maximum MSE criterion, identifying a smaller number of expensive samples in both cases, albeit by just one point. It can therefore be concluded that the criterion based on E[I(x_1,x_2)] for two-variable functions is extremely useful for the progressive addition of points to the initial High Fidelity dataset, with the aim of constructing a surrogate model that best approximates the objective function while limiting the computational cost required to achieve this goal.

In the next section, the performance of the third and final proposed algorithm, based on the Information Gain, as described in the case of single-variable functions,

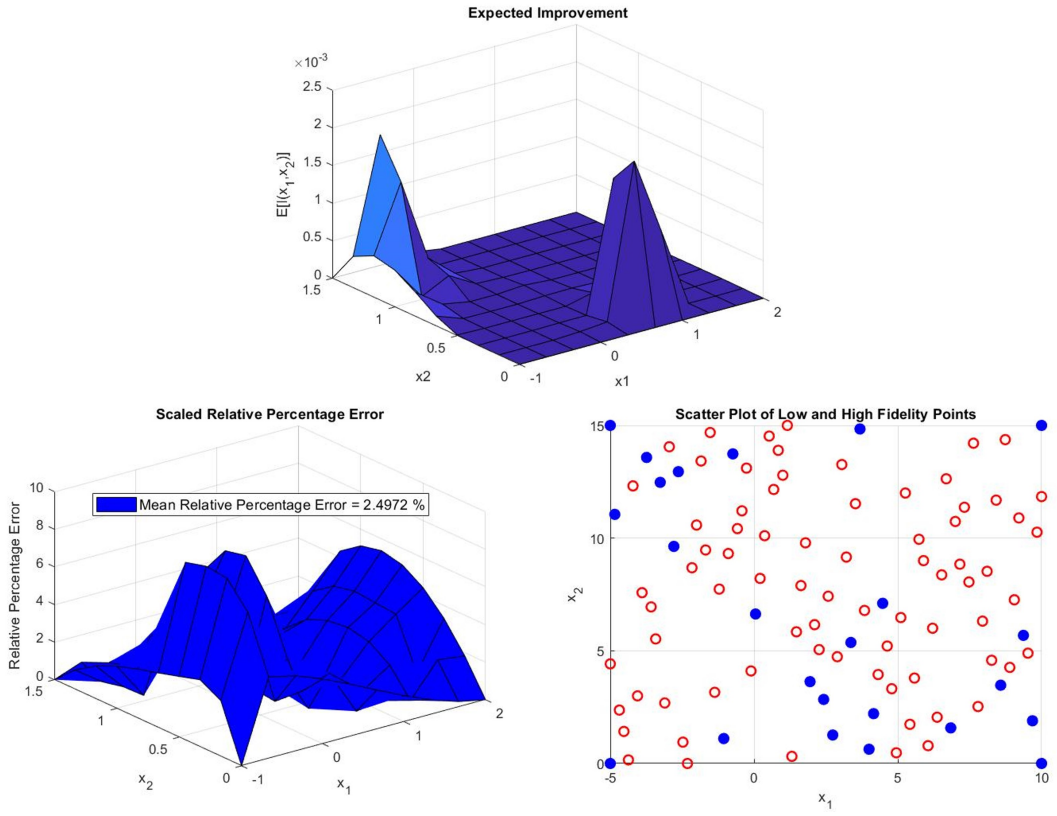

Figure 2.49: Co-Kriging model's Expected Improvement at iteration 12 (Up), final relative percentage error (Left) and final High Fidelity samples distribution in the design space (Right)

will be assessed and implemented for the three two-variable test functions.

2.5.3 Maximum Information Gain criterion

The adaptive sampling criterion based on the Information Gain parameter, as previously defined for single-variable functions, can now be extended to the case of two-variable functions. The underlying concept remains the same: starting with an initial dataset of Low Fidelity points, constructed, for example, by generating a Latin Hypercube or following a factorial approach, and an initial High Fidelity dataset with four points located at the extremes of the design space, a preliminary Co-Kriging surrogate model is built. The variance distribution across the candidate points, which will obviously belong to the cheap dataset (nested approach), is then evaluated. By knowing the s^2, the initial entropy distribution is calculated as expressed by Equation 2.4. Subsequently, for each candidate point, the entropy

resulting from adding that point to the High Fidelity dataset is evaluated. This involves constructing a new Co-Kriging surrogate model that includes the new sample in the existing dataset at that iteration. To achieve this, a Bayesian estimation of the expensive function value at the candidate point is performed, also calculating a 95% confidence interval for the mean value estimate of the prediction, allowing the surrogate model to be built without knowing the actual High Fidelity objective function value. The Information Gain is then obtained by averaging the difference between the initial and final entropy, bearing in mind that for points already sampled in the High Fidelity dataset, this parameter loses significance and can thus be assigned an arbitrary negative value. The point that maximizes this information index in terms of reducing the model's uncertainty will be the new point to add to the expensive data and the algorithm will continue the search until the stopping criterion is met. This stopping criterion, as previously mentioned for two-variable functions, requires the weighted mean relative error to be less than 3% to halt the cycle. The test functions studied are the Himmelblau Function, the Booth Function, and the Branin Function, namely:

- **Himmelblau Function [47] -** x_1**,** $x_2 \in$ **[-4,4]**:

$$f_e(x_1, x_2) = (x_1^2 + x_2 - 11)^2 + (x_2^2 + x_1 - 7)^2$$
$$f_c(x_1, x_2) = f_e(0.5x_1, 0.8x_2) + x_2^3 - (x_1 + 1)^2$$

- **Booth Function [47] -** x_1**,** $x_2 \in$ **[-10,10]**:

$$f_e(x) = (x_1 + 2x_2 - 7)^2 + (2x_1 + x_2 - 5)^2$$
$$f_c(x) = f_e(0.4x_1, x_2) + 1.7x_1x_2 - x_1 + 2x_2$$

- **Adjustable Branin Function [48] -** $x_1 \in$ **[-5,10],** $x_2 \in$ **[0,15],** $a \in$ **[0,1]**:

$$f_e(x) = \left(x_2 - (5.1\tfrac{x_1^2}{4\pi^2}) + \tfrac{5x_1}{\pi} - 6\right)^2 + \left(10cos(x_1)(1 - \tfrac{1}{8\pi})\right) + 10$$
$$f_c(x) = f_e(x_1, x_2) - (a + 0.5)\left(x_2 - (5.1\tfrac{x_1^2}{4\pi^2}) + \tfrac{5x_1}{\pi} - 6\right)^2$$

As previously done, for the Himmelblau Function, both a space-filling Latin Hypercube and a factorial sampling plan will be considered. Starting from the Himmelblau Function with the Low Fidelity Design of Experiment shown in Figure 2.30, the Maximum Information Gain criterion is applied and the following results are obtained. Figure 2.51 shows the scaled Information Gain at iteration 12 as a heat map on which the candidate points are represented; the black dot is the location of the maximum IG(x_1,x_2) value at that iteration and that point will therefore be added to the High Fidelity dataset.

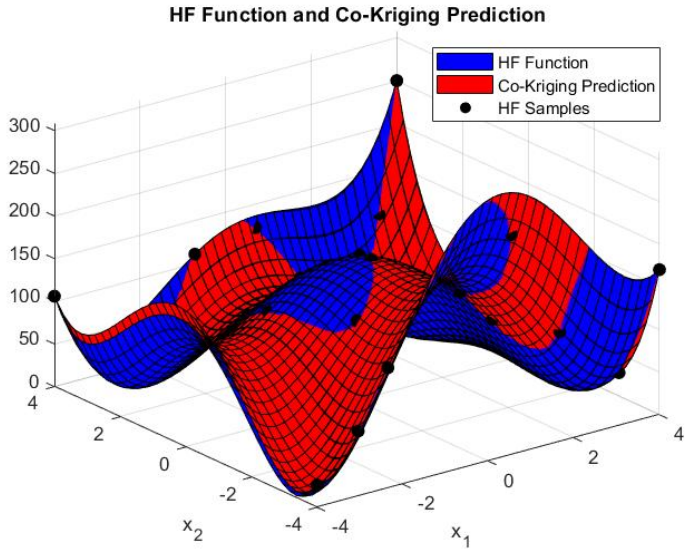

Figure 2.50: High Fidelity Function and Co-Kriging prediction for the Himmelblau Function with 23 HF samples

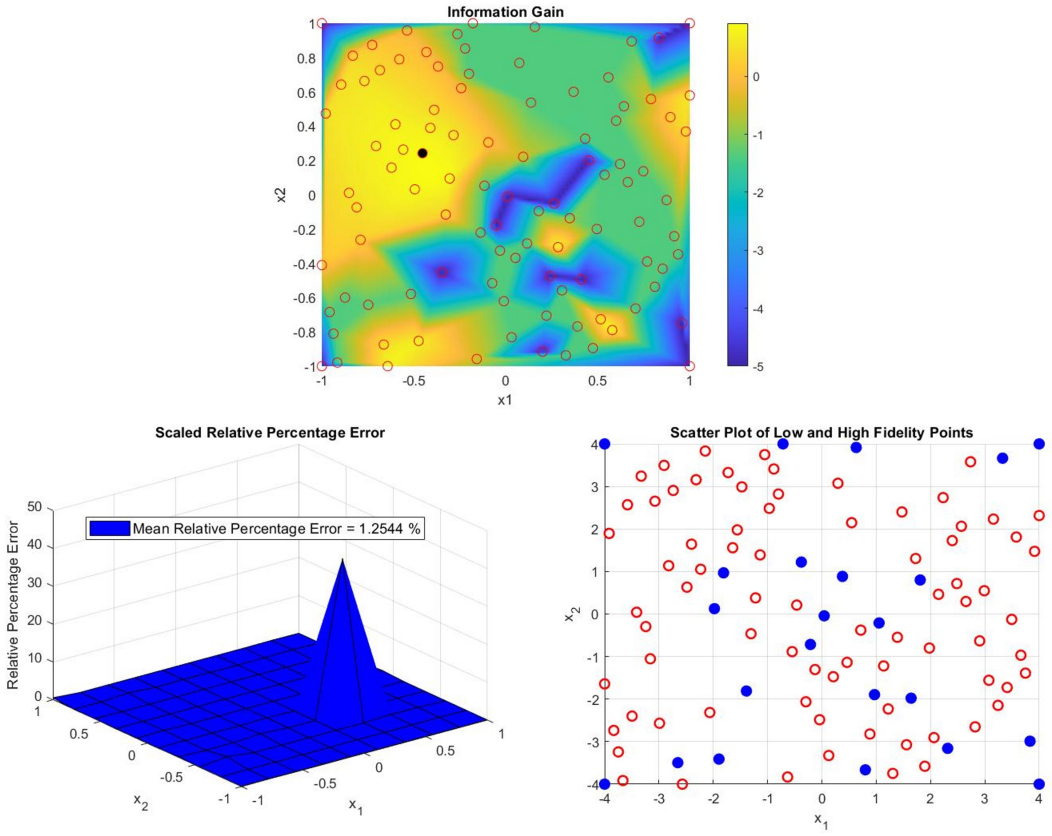

Figure 2.51: Co-Kriging model's scaled Information Gain at iteration 12 (Up), final relative percentage error (Left) and final High Fidelity samples distribution in the design space (Right)

In this case, by leveraging the information-based criterion, it is possible to further reduce the number of High Fidelity points required compared to both the Maximum Variance criterion (27 expensive points) and the Expected Improvement criterion (26 HF points), now achieving a Design of Experiment consisting of 23 costly points. The reduction in computational burden in terms of necessary evaluations is evident in the case of the Himmelblau Function and the final weighted mean error obtained is 1.254%, which is extremely low. Since each iteration requires the construction of multiple surrogate models, each characterized by adding one of the candidate points to the set available in the given iteration, the time required by the algorithm to identify the point to be sampled based on the Information Gain will certainly be greater than that required by the previous two algorithms. However, it remains extremely manageable and certainly incomparable to the time required, for example, by a High Fidelity numerical simulation. The potential of this algorithm is thus highlighted.

Considering instead the Low Fidelity factorial Design of Experiment shown in Figure 2.33, the results obtained from applying the Information Gain maximization algorithm are presented below.

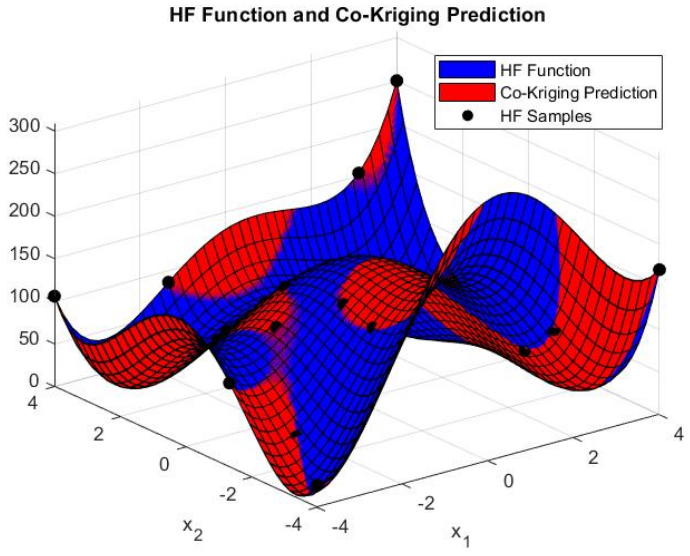

Figure 2.52: High Fidelity Function and Co-Kriging prediction for the Himmelblau Function with 24 HF samples when a 100 points LF factorial sampling plan is employed

Even when employing a Low Fidelity factorial sampling plan, the Information Gain algorithm results in a reduction in the number of High Fidelity points required,

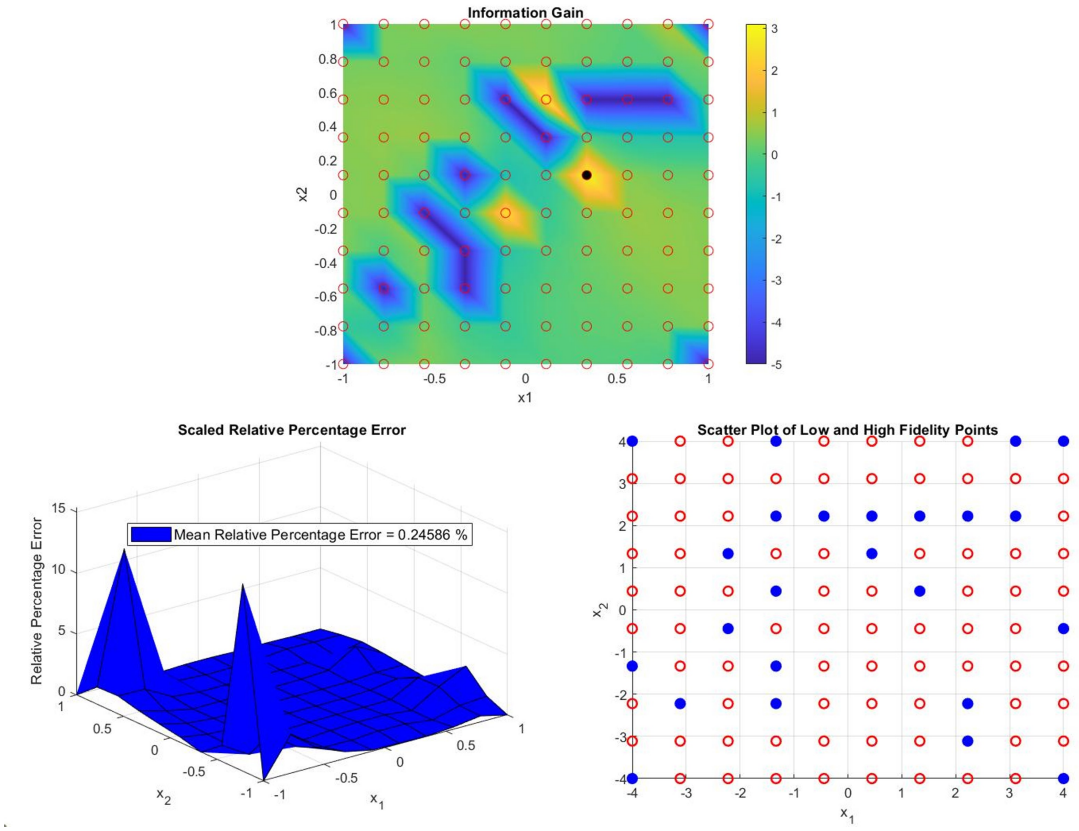

Figure 2.53: Co-Kriging model's scaled Information Gain at iteration 12 (Up), final relative percentage error (Left) and final High Fidelity samples distribution in the design space (Right)

in this case, 24. The final approximation error obtained is extremely low, around 0.246%. Therefore, it is possible to proceed with studying the remaining two test functions. Regarding the Booth Function, the application of the algorithm yields the results shown in Figure 2.54 and 2.55: the Information Gain algorithm leads to a reduction of the High Fidelity points with respect to the Maximum Variance criterion, from 11 to 10 points and the final weighted mean error reaches 2.227%. Compared to the Maximum Expected Improvement algorithm, the High Fidelity samples number is the same, i.e. 10 points, but the error is slightly higher for the case presented in this section.

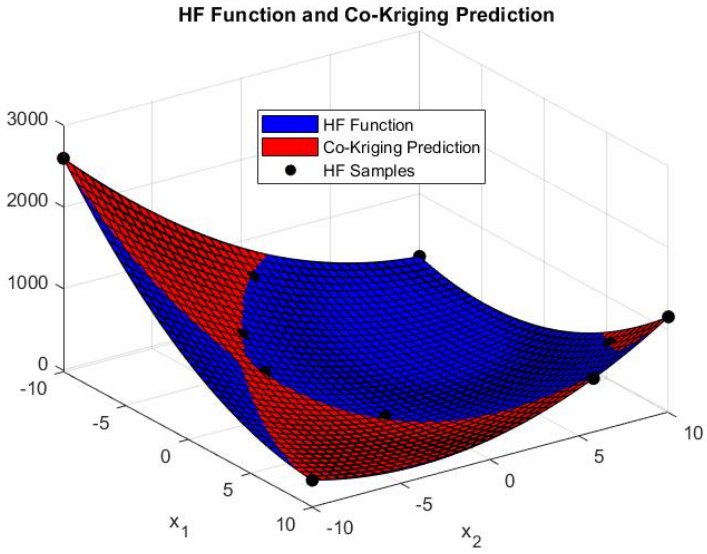

Figure 2.54: High Fidelity Function and Co-Kriging prediction for the Booth Function with 10 HF samples when a 100 points LF factorial sampling plan is employed

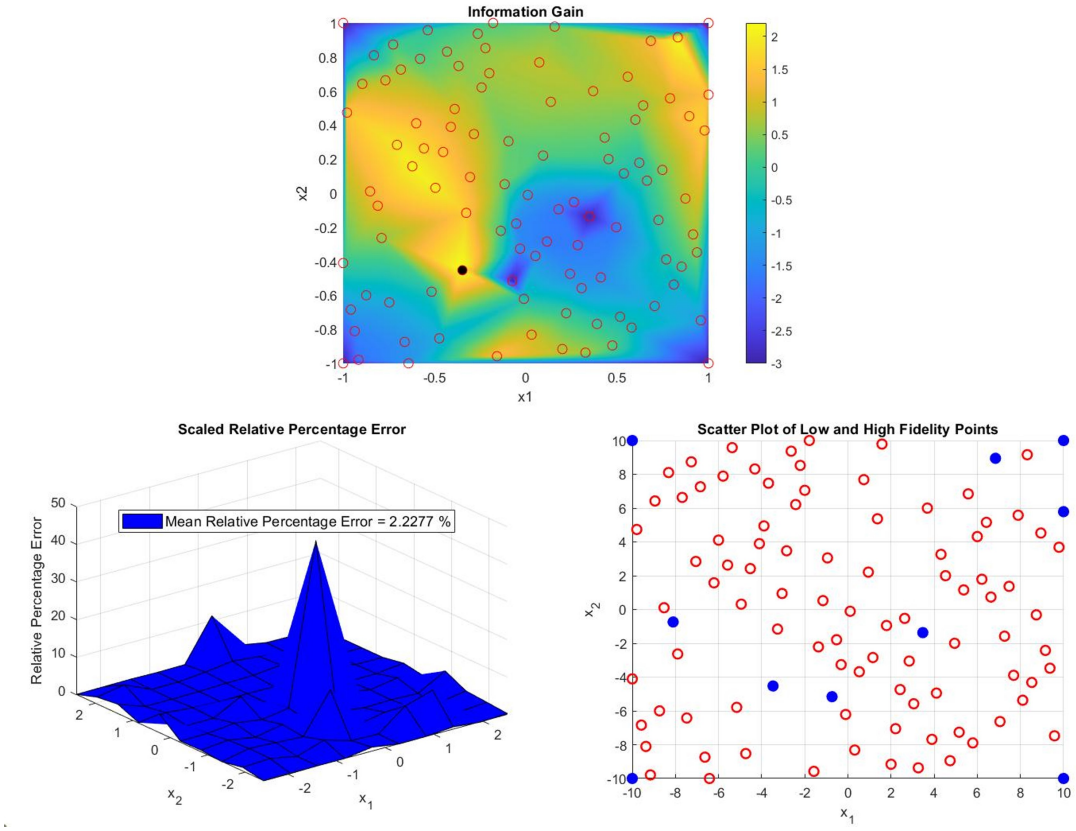

Figure 2.55: Co-Kriging model's scaled Information Gain at iteration 4 (Up), final relative percentage error (Left) and final High Fidelity samples distribution in the design space (Right)

Finally, the application of the Information Gain algorithm to the Branin Function leads to the results shown in Figures 2.56 and 2.57.

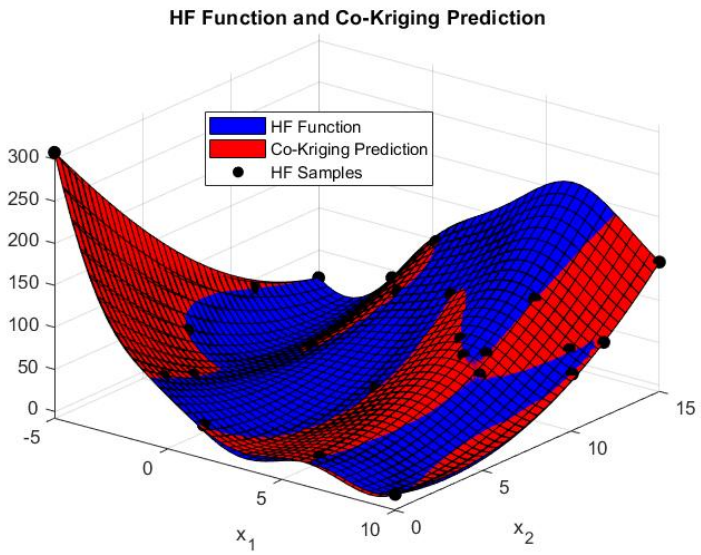

Figure 2.56: High Fidelity Function and Co-Kriging prediction for the Branin Function with 24 HF samples when a 100 points LF factorial sampling plan is employed

In this case, the High Fidelity dataset will consist of 24 points, as determined by the application of the Maximum Expected Improvement algorithm, now yielding a slightly higher error of approximately 2.974%. For the Branin Function, both the Maximum Information Gain and Maximum E[I(x1, x2)] algorithms perform almost equally, each identifying exactly one more point compared to the Maximum Variance criterion. The latter found instead the maximum number of points for both the Himmelblau and Booth Functions.
After conducting various tests on the three presented functions and validating the adaptive sampling algorithms, it is possible to summarize the main results obtained by reporting in tables the number of High Fidelity sampling points and the weighted mean relative error achieved when the stopping criterion is satisfied. It is important to note that the Maximum Variance and Expected Improvement algorithms provide the final result in less time compared to the Information Gain-based algorithm, which, as previously mentioned, requires the construction of multiple surrogate models at each iteration. Although this generation is practically instantaneous thanks to the use of the ooDACE toolbox, it is evident that the time required to identify the point to be added to the High Fidelity dataset will indeed be greater. For example, considering the Himmelblau Function, the time required to reach

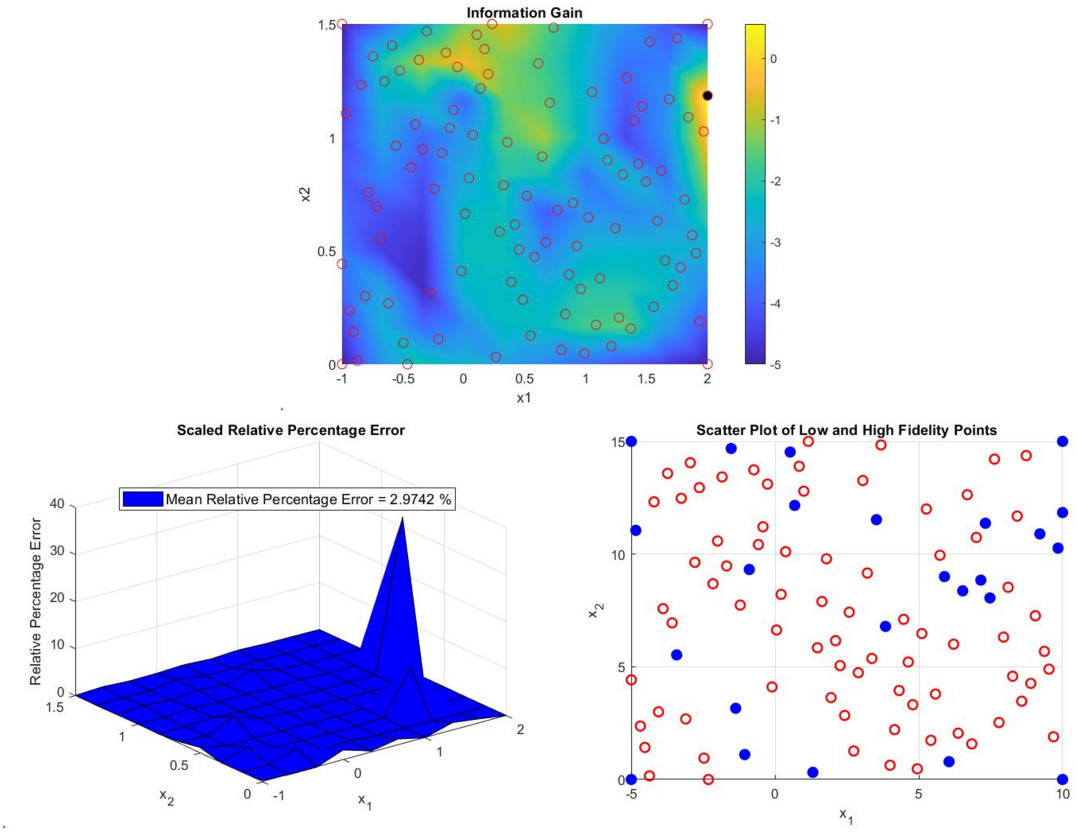

Figure 2.57: Co-Kriging model's scaled Information Gain at iteration 12 (Up), final relative percentage error (Left) and final High Fidelity samples distribution in the design space (Right)

convergence with the Maximum Variance algorithm is approximately 30.8 seconds, 23 seconds for the Expected Improvement and 426.3 seconds for the Information Gain (times evaluated on a simple desktop architecture). The computational cost in terms of time is significantly higher in the latter case, but for the Himmelblau Function, this algorithm allows reducing the High Fidelity objective function evaluations from 27 (Variance) and 26 (EI) to 23, a considerable computational saving. Regarding the other functions, namely Booth and Branin, it has been observed that the advantage of the Information Gain-based algorithm in terms of reducing the number of High Fidelity points is similar to that obtainable from the Expected Improvement algorithm, which is also much faster. In these cases, the final choice of the algorithm may favor the one that achieves the result with fewer operations, given the same number of entries in the expensive dataset. However, another parameter to consider is the final approximation error, which can tip the balance in favor of one algorithm over another.

By examining Tables 2.4, 2.6 and 2.7, it can be concluded that it is not possible to precisely predict which algorithm will require the fewest High Fidelity points; in other words, the results depend on the specific problem being analyzed and on the Low and High Fidelity functions. For the Himmelblau Function, the Information Gain criterion outperforms the other two algorithms, but this is not the case for the Branin Function, where the Maximum Variance algorithm actually requires fewer evaluations, despite being the least performant for the other two test functions.

Himmelblau Function - Latin Hypercube			
Sampling Criterion	LF Samples	HF Samples	Mean Weighted Relative Error (%)
Max Variance	100	27	0.11682
Max E[I(x_1,x_2)]	100	26	0.4332
Max IG(x_1,x_2)	100	23	1.2544

Table 2.4: Results comparison between the three different algorithms for the Himmelblau Function (Low Fidelity Latin Hypercube sampling plan)

Himmelblau Function - Factorial			
Sampling Criterion	LF Samples	HF Samples	Mean Weighted Relative Error (%)
Max Variance	100	22	2.811
Max E[I(x_1,x_2)]	100	23	2.0122
Max IG(x_1,x_2)	100	24	0.24586

Table 2.5: Results comparison between the three different algorithms for the Himmelblau Function (Low Fidelity factorial sampling plan)

Booth Function - Latin Hypercube			
Sampling Criterion	LF Samples	HF Samples	Mean Weighted Relative Error (%)
Max Variance	100	11	0.5552
Max E[I(x_1,x_2)]	100	10	1.0909
Max IG(x_1,x_2)	100	10	2.2277

Table 2.6: Results comparison between the three different algorithms for the Booth Function

Branin Function - Latin Hypercube			
Sampling Criterion	LF Samples	HF Samples	Mean Weighted Relative Error (%)
Max Variance	100	23	2.5528
Max E[I(x_1,x_2)]	100	24	2.4972
Max IG(x_1,x_2)	100	24	2.9742

Table 2.7: Results comparison between the three different algorithms for the Branin Function

Chapter 3

Cold Gas Micropropulsion

3.1 Introduction

Cold gas systems represent one of the simplest architectures for space propulsion and have been employed since the early spacecrafts of the 1960s as attitude control systems [49]. In modern times, such propulsion systems are still widely utilized, particularly in the field of *nanosatellites*, commonly referred to as "CubeSats". Indeed, the use of such satellites, whose dimensions are typically on the order of cm^3 and mass usually in the range of a few tens of kilograms, necessitates, first and foremost, the employment of a lightweight propulsion system characterized by low power requirements, and must be simple, reliable and cost-effective. In these regards, cold gas systems have always been preferred over other types of propulsion systems for missions requiring limited total impulses, for which they represent the lightest and most straightforward implementation solution [50]. Therefore, in addition to their small dry masses and the few Watts of electrical power required to ensure operability, a further advantage of such systems lies in the use of non-toxic inert substances and the fact that the propellant exiting from the thruster nozzle does not produce residues that could interfere with mechanical actuators or sensors inherent to the spacecraft [50].
As previously mentioned, the primary application of these cold gas systems is in the attitude and trajectory control of satellites, being used for orbit maintenance and to perform small maneuvers, which typically require low thrust values and limited total impulses ($\Delta V < 50 m/s$) [51]. Generally, cold gas thrusters are designed to provide thrust ranging from tens of milliNewtons (mN) to a few Newtons (N), depending on the total temperature and pressure conditions at the exhaust nozzle inlet [51]. Despite these systems being characterized by small impulse bits (I-bit), hence great readiness and response time, their specific impulses (I_{sp}) are typically low compared to other types of propulsion systems, such as chemical or electric,

reaching values between 10 and 100 s depending on the propellant used (generally gaseous nitrogen, xenon or krypton) [52].

In cold gas systems, thrust is generated by accelerating the propellant (pressurized gas or compressed liquid) through a nozzle, thereby converting the energy contained within the propellant into kinetic energy, without undergoing a combustion process or heating the gas. The higher the velocity of the exhaust at the exit section of the nozzle, the greater the thrust produced. The absence of propellant combustion makes cold gas systems extremely simple and safe, but it also limits the achievable thrust levels, as only a small portion of the energy inherent in the chemical bonds of the propellant is utilized.
From an architectural standpoint, the simplicity of these propulsion systems is illustrated in Figure 3.1.

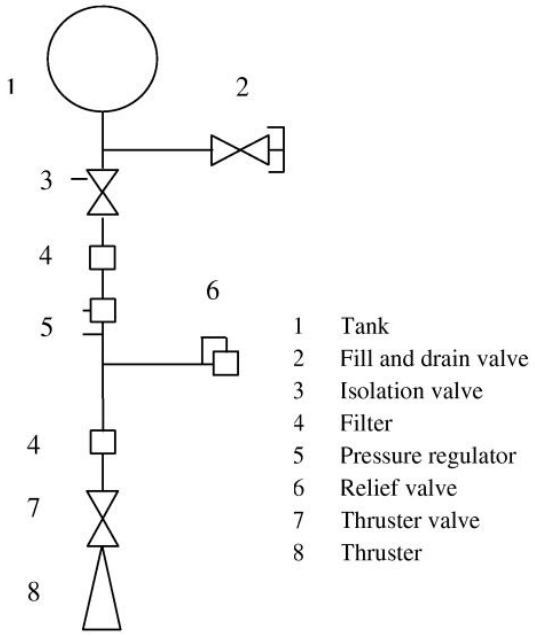

Figure 3.1: Layout of cold gas system [51]

In general, the propellant, typically in the gaseous state, is contained within a pressurized tank and flows towards the exhaust nozzle through a pipeline, propelled by the pressure differential between the two aforementioned components. The gas then passes through a series of valves along the supply line, such as a pressure regulator valve, tasked with reducing the pressure from the high tank value to the desired value at the inlet section of the nozzle, which determines the achievable thrust. Subsequently, there is a thruster valve, also referred to as solenoid valve, which allows for the regulation of propellant flow towards the nozzle through

opening and closing actions. The component responsible for thrust generation, defined as the cold gas thruster, consists of the combination of the solenoid valve and the nozzle, through which the gas expands into the outer space. Usually, a cold gas system includes several thrusters in order to achieve the desired thrust to perform the required maneuvers, granting stability and attitude control.
As far as propellants for cold gas systems are concerned, both gaseous and liquid propellants can be stored inside the tank, the only constraints are that the propellant must be in gas phase when it expands within the nozzle and no combustion must take place [53]. Focusing on gas propellants, Table 3.1 displays some of the possible candidates and their performances in terms of molecular mass, density, specific heat ratio and theoretically achievable specific impulse in vacuum.

Propellant	Molecular Mass (g/mol)	Density (lb/ft^3)	γ	I_{sp} (s)
Hydrogen	2.0	1.77	1.40	284
Helium	4.0	3.54	1.67	179
Methane	16.0	14.1	1.30	114
Nitrogen	28.0	24.7	1.40	76
Air	28.9	25.5	1.40	74
Argon	39.9	35.3	1.67	57
Krypton	83.8	74.1	1.63	50

Table 3.1: Properties of cold gas propellants. Density is measured at 5000 psia and 20°C, I_{sp} measured in vacuum with nozzle area ratio 50:1 and initial temperature 20°C [49].

The choice of propellant is highly significant as it impacts the sizing of the tank in terms of mass and volume. Generally, one would want a gas with low boiling and low melting temperature, with a view to achieve an efficient storage of the propellant [53]. In the case of cold gas micropropulsion, mission constraints may render the use of gasses such as high-pressure hydrogen and helium unfeasible: as evidenced in Table 3.1, these gasses exhibit much higher specific impulses compared to the rest of proposed alternatives, however, their density is extremely low, resulting in large gas storage volumes and thus in heavier tanks. Most significant leakage issues may affect cold gas systems employing hydrogen and helium [49]. Among all of the proposed gasses, nitrogen is certainly the most commonly used in various missions. It is an inert and non-toxic gas, characterized by a favourable density compared to helium and hydrogen, allowing for a significant reduction in volumes and overall system weight. Furthermore, its relatively low molecular mass enables specific impulses on the order of 70 seconds, still low but better than other propellants, such

as argon and krypton. As mentioned earlier, another strength of gaseous nitrogen as a propellant lies in its non-contaminating properties for spacecraft sensors and mechanical surfaces. Other propellants used for cold gas propulsion include freon and xenon or liquid-state propellants in the tank, such as carbon dioxide, propane, butane and ammonia [54]. However, resorting to liquid propellants introduces various complexities in the system that may not lead to a better solution than gaseous nitrogen [53]. To address the issue of the low density of the gasses listed in Table 3.1, one may consider storing the propellant in solid-form within the tank, subsequently igniting and vaporizing the solid, in order to provide the nozzle with gaseous propellant [55].

3.2 Nozzle design

The design of the thruster, comprising the solenoid valve and the nozzle assembly, stands as one of the most pivotal aspects, as the performance of the cold gas propulsion system hinges upon these components. Particularly, the required thrust level for executing maneuvers and attitude control necessitates a thorough examination of the type and geometry of the exhaust nozzle: in the case of cold gas microthrusters, this typically entails a convergent-divergent (De Laval) nozzle, which accelerates the gas from a subsonic velocity at the converging section to a supersonic velocity in the divergent portion. At the throat section, the velocity reaches sonic conditions, thus maximizing the mass flow rate through the nozzle and consequently the thrust performance. Regarding the divergent portion of the nozzle, it typically comes in a conical or bell-shaped form; however, given the highly compact dimensions characteristics of these types of thrusters and manufacturing constraints, there is a tendency to favor the use of conical nozzles [56]. Nevertheless, bell-shaped nozzles are still employed in various microthrusters.

As previously mentioned, the performance of cold gas propulsion systems is heavily dependent on the nozzle. Therefore, aiming to maximize the achievable performance entails selecting the optimal geometry of the nozzle, i.e. in terms of the ratio between the exit area and the throat area, as well as the angles that characterize the divergent section.
In Chapter 4, a Co-Kriging based optimization will be conducted to determine the optimal values for the angles θ_1 and θ_2 that define the bell-shaped geometry of the divergent part of the nozzle, taking into account the thrust coefficient C_F and the thruster weight, which we aim respectively to maximize and minimize.

3.2.1 Ideal nozzle performance

In order to obtain a preliminary indication of the performance provided by a cold gas thruster, it is possible to calculate the main parameters of interest such as thrust, mass flow rate, Mach number and outlet pressure using a set of fundamental thermodynamic equations, derived under the assumption of ideal flow. These equations, which describe the flow behaviour inside the nozzle as quasi-one-dimensional, neglect the sources of irreversibility and losses, as well as real phenomena and initially provide reasonably accurate results, which can be further refined by setting aside the hypothesis of ideal flow [49]. The main assumptions underlying the ideal thruster model are as follows [32,35]:

1. Throughout the nozzle, propellant gasses are homogeneous and their composition is invariant (frozen flow)
2. Only gasses flow through the nozzle, any liquid or solid condensed phases are neglected
3. The gasses follow the perfect gas law
4. The flow inside the nozzle is adiabatic and heat transfer across the walls is neglected
5. No friction at the walls is present, thus boundary layer effects are neglected
6. Throughout the nozzle, the flow is steady and constant, gas expansion is uniform and no transient effects, shock waves or discontinuity are present
7. The gas velocity at the outlet section of the nozzle is axially directed
8. Gas velocity, pressure, temperature and density are uniform across any section normal to the nozzle axis

As a result, the expansion in the nozzle can be studied relying on isentropic relations, since the flow can be considered both adiabatic and reversible (this implies that total temperature and total pressure are conserved in the nozzle).
The geometry of a generic bell-shaped De Laval convergent-divergent nozzle is shown in Figure 3.2: the nozzle consists of a convergent section where a subsonic flow is accelerated until it reaches sonic velocity at the throat section, denoted by A_t in the figure. The inlet conditions of the gas in terms of temperature and pressure are denoted by the subscript 'c'; specifically, the pressure value p_c depends on the regulator valve placed along the supply line. The latter reduces the pressure from the tank value to the required value to generate a certain thrust. The higher the nozzle inlet pressure, the greater the achievable thrust. It must be remembered that in cold gas thrusters, combustion does not occur. At the throat section, the

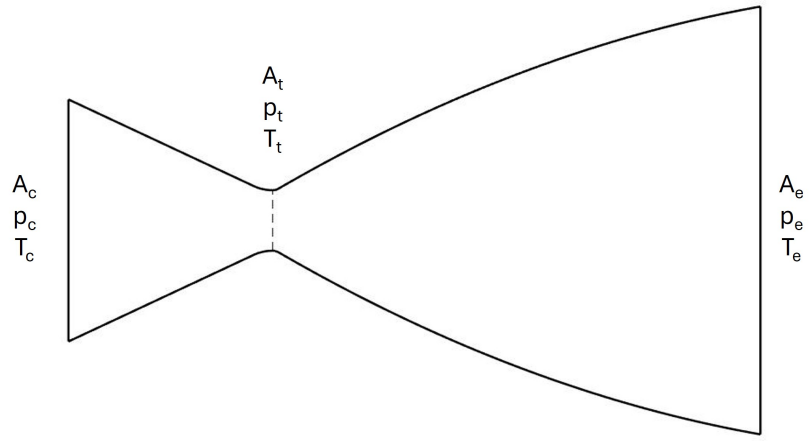

Figure 3.2: Convergent-Divergent nozzle geometry

flow reaches critical conditions, where the mass flow rate reaches its maximum value and the Mach number is $M_t = 1$. Therefore, in the convergent section, the gas accelerates and its pressure decreases (subsonic flow in a convergent section), while downstream of the throat section, the flow continues to accelerate (supersonic flow in divergent section) and the gas expands until it exits into the vacuum. The exit section of the nozzle and the corresponding pressure and temperature conditions are denoted by the subscript 'e'. The exit flow velocity actively influences the achievable thrust value, so higher velocities are desired to optimize the propulsion system performance. Finally, thrust is also influenced by the exit pressure p_e and area A_e. In summary, the nozzle generates thrust by momentum exchange between the exhaust gasses and the spacecraft and by the pressure imbalance at the exit section of the nozzle (in vacuum $p_0 = 0$) [52]:

$$F = \dot{m}_e V_e + A_e p_e \tag{3.1}$$

Below are the fundamental relations to calculate the performance of a cold gas thruster, in the case of ideal nozzle and flow. Since the flow is adiabatic, the total pressure at the nozzle throat will be equal to both the total pressure at the inlet and exit section, thus

$$T_t^0 = T_c^0 = T_e^0 \tag{3.2}$$

Given the Poisson relations (Equation 3.3), critical values for temperature, pressure, density and sound speed can be obtained at the nozzle throat (Equations 3.4, 3.5, 3.6 and 3.7):

$$\frac{p_2}{p_1} = \left(\frac{T_2}{T_1}\right)^{\frac{\gamma}{\gamma-1}}, \frac{\rho_2}{\rho_1} = \left(\frac{T_2}{T_1}\right)^{\frac{1}{\gamma-1}}, \frac{a_2}{a_1} = \left(\frac{T_2}{T_1}\right)^{\frac{1}{2}} \tag{3.3}$$

$$\frac{T_t}{T_c} = \frac{2}{\gamma+1} \tag{3.4}$$

$$\frac{p_t}{p_c} = \frac{2}{\gamma+1}^{\frac{\gamma}{\gamma-1}} \tag{3.5}$$

$$\frac{\rho_t}{\rho_c} = \frac{2}{\gamma+1}^{\frac{1}{\gamma-1}} \tag{3.6}$$

$$\frac{a_t}{a_c} = \frac{2}{\gamma+1}^{\frac{1}{2}} \tag{3.7}$$

where a is the speed of sound and γ is the ratio of specific heats of the gas. Combining Equations 3.2 and 3.4-3.7 and remembering that the Mach numbers $M_t = 1$, $M_c = 0$ and $M_e = V_e/\sqrt{\gamma \cdot R \cdot T_e}$, a relation between the exit velocity V_e and the pressure ratio p_e/p_c is obtained:

$$V_e = \sqrt{\frac{2\gamma R}{\gamma - 1} \cdot T_c \left[1 - \left(\frac{p_e}{p_c}\right)^{\frac{\gamma.1}{\gamma}}\right]} \tag{3.8}$$

Equation 3.8 shows that the higher the temperature at the inlet section and the higher the gas constant R, the greater the exhaust velocity, thus the thrust.

As a rule, the inlet conditions are given as a problem data, hence p_c and T_c are known; furthermore, given the gas flowing throughout the nozzle, the values of γ and R are also known. Therefore, to obtain the value of V_e it is necessary to know the pressure at the outlet section of the nozzle. Typically, another parameter assumed as a problem data is the expansion ratio ε, defined as the ratio of the exit area of the nozzle to the throat area:

$$\varepsilon = \frac{A_e}{A_t} \tag{3.9}$$

As implied by its name, the area ratio ε serves as an indicator of the extent of expansion that the gas undergoes along the nozzle; consequently, the higher the expansion ratio, the lower the gas pressure at the exit section (the gas has undergone a significant expansion). The optimal value of ε is achieved when the exit pressure p_e reaches the value of the environmental pressure outside the nozzle,

which is zero in the case of a spacecraft operating in vacuum. However, to have $p_e = 0$, an infinite Ae/A_t would be required, so ε is chosen to be as large as possible, but compliant with the size and weight constraints of the propulsion system [52].

A relation linking the expansion ratio and the pressure ratio p_e/p_c can be found starting from the mass flow rate evaluation. This parameter remains constant throughout the nozzle and can be generally defined as follows:

$$\dot{m} = \rho \cdot A \cdot V \tag{3.10}$$

Using equations 3.3 and 3.8, the following expression is obtained

$$\dot{m} = \frac{p_c A_e}{\sqrt{RT_c}} \cdot \sqrt{\frac{2\gamma}{\gamma - 1}\left[\left(\frac{p_e}{p_c}\right)^{\left(\frac{2}{\gamma}\right)} - \left(\frac{p_e}{p_c}\right)^{\frac{\gamma+1}{\gamma}}\right]} \tag{3.11}$$

Now, specifying mass flow rate in the throat section, where $V_t = a_t$ and substituting in Equation 3.11, the expansion ratio is found as a function of p_e/p_c:

$$\frac{A_e}{A_t} = \frac{\sqrt{\gamma}\left(\frac{2}{\gamma+1}\right)^{\frac{\gamma+1}{2(\gamma-1)}}}{\left(\frac{p_e}{p_c}\right)^{\frac{1}{\gamma}}\sqrt{\frac{2\gamma}{\gamma-1}\left[1 - \left(\frac{p_e}{p_c}\right)^{\frac{\gamma-1}{\gamma}}\right]}} \tag{3.12}$$

Thus, for example, for a given ε and p_c, the value of p_e can be derived; subsequently, the mass flow rate can be calculated from Equation 3.11, obtaining all the parameters required in the thrust evaluation. In fact, one can substitute Equations 3.8, 3.11 and 3.12 in 3.1, obtaining the thrust F as a function of p_e/p_c:

$$F = p_c A_t \sqrt{\frac{2\gamma^2}{\gamma - 1}\left(\frac{2}{\gamma + 1}\right)^{\frac{\gamma+1}{\gamma-1}}\left[1 - \left(\frac{p_e}{p_c}\right)^{\frac{\gamma-1}{\gamma}}\right]} + p_e A_e \tag{3.13}$$

Equation 3.13 gives the ideal thrust produced by the cold gas propulsion system in vacuum. Furthermore, the thrust coefficient C_F and the specific impulse I_{sp} can be estimated. As far as C_F is concerned, this parameter is defined as the ratio of the thrust and the product of throat area and inlet pressure

$$C_F = \frac{F}{p_c \cdot A_t} \tag{3.14}$$

Using Equation 3.13, we have that

$$C_F = \sqrt{\frac{2\gamma^2}{\gamma-1}\left(\frac{2}{\gamma+1}\right)^{\frac{\gamma+1}{\gamma-1}}\left[1-\left(\frac{p_e}{p_c}\right)^{\frac{\gamma-1}{\gamma}}\right]} + \frac{p_e A_e}{p_c A_t} \tag{3.15}$$

C_F is heavily dependant on the expansion ratio ε and better performance is obtained the greater this parameter. Typical values for C_F in vacuum can reach up to 1.7-1.8 depending on ε.

Regarding specific impulse I_{sp}, it can be defined as the ratio of the generated thrust F to the product of the mass flow rate passing through the nozzle and Earth's gravitational acceleration ($g_0 = 9.81 m/s$):

$$I_{sp} = \frac{F}{\dot{m} \cdot g_0} \tag{3.16}$$

Specific impulse highly depends on the propellant used and also provides an estimate of the propulsion system's fuel consumption: high values of I_{sp} require small propellant flow rates to generate thrust F, while lower Isp results in higher propellant consumption to achieve the same thrust. As mentioned earlier, in the case of cold gas microthrusters, such as those based on gaseous nitrogen GN_2, specific impulse values are around 70 s, much lower than those typical of electric or chemical propulsion systems. This is one of the reasons why cold gas technology is not used for primary propulsion, as missions requiring significant ΔV would demand enormous propellant flow rates.

Finally, another useful parameter to evaluate is the Mach number at the exit section of the nozzle. This parameter can be assessed in various ways, but one of the simplest methods involves exploiting the isentropic nature of the flow ($p_c^0 = p_e^0$), allowing to write:

$$p_c = p_e\left(1+\frac{\gamma-1}{2}M_e^2\right)^{\frac{\gamma}{\gamma-1}} \longrightarrow M_e = \sqrt{\frac{2}{\gamma-1}\left[\left(\frac{p_c}{p_e}\right)^{\frac{\gamma-1}{\gamma}}-1\right]} \tag{3.17}$$

The value of p_e/p_c is known from Equation 3.12 for a certain ε, thus M_e is rapidly obtained. Being the expansion ratio usually very high for cold gas microthrusters and given the fact that the temperature of the exhaust gasses will be highly low, we expect great values of the Mach number M_e.
The equations presented in this subsection relate to the ideal case, where losses and the real aerothermodynamic behaviour of the flow are neglected. Below, the analysis of real nozzle performance is introduced.

3.2.2 Real nozzle performance

The assumptions made in the case of ideal nozzle and flow, while providing an excellent starting point for studying the performance of a rocket in general, and a cold gas thruster in particular, still represent an idealization that does not account for some important physical and thermodynamic phenomena that impact the propulsion system's performance. Hence, the real value of the thrust will obviously be lower that that calculated with the equations presented in the previous subsection. In particular, the flow inside the nozzle is two-dimensional and axisymmetric, thus more complex than previously described; furthermore, the assumption of constant velocity, pressure, temperature and density on each section perpendicular tho the axis of the nozzle falls away, resulting in a non-uniform distribution with higher values in the central region and progressively decreasing values away from the thruster's symmetry plane. Regarding the throat Mach number, for instance, in the ideal case it is considered to be one at the section located exactly at the throat radius; however, in the real case, the unity value is reached on a more or less curved surface, slightly downstream of A_t [49].
Moreover, in the real case, there are several sources of energy loss inside the nozzle and not all the enthalpy stored inside the gas is effectively converted into kinetic energy. Typically, losses involve real gas effects, losses due to flow divergence and losses related to viscous effects. Below are various sources of losses [49]:

1. The flow in the nozzle exit section is not axially oriented, thus the velocity V_e calculated in Equation 3.8 will be lower due to the presence of a radial velocity component. The shape of the nozzle influences divergence losses: for conical nozzles, higher values of the divergence angle determines lower values of V_e, thus less thrust, because the radial component will be important. Bell-shaped nozzles help reducing flow divergence losses

2. A slight loss in exhaust velocity and thrust can be caused by small nozzle contraction ratios, i.e. the ratio between the inlet area of the nozzle and the throat area A_c/A_t. Despite these losses being minimal, usually a contraction ratio of the order of 4 is chosen, to reduce pressure losses in the inlet

3. Viscous effects, above all the formation of a boundary layer on nozzle walls, where the velocity evolves from the wall value of zero to blend with the velocity outside the boundary layer itself, determine a loss in terms of V_e

4. Performance can be affected by non-uniform gas composition, for example due to turbulence

5. Real gas effects and rarefaction may affect nozzle performance, although the latter effects can be neglected if the thruster produces more than 1mN [57]

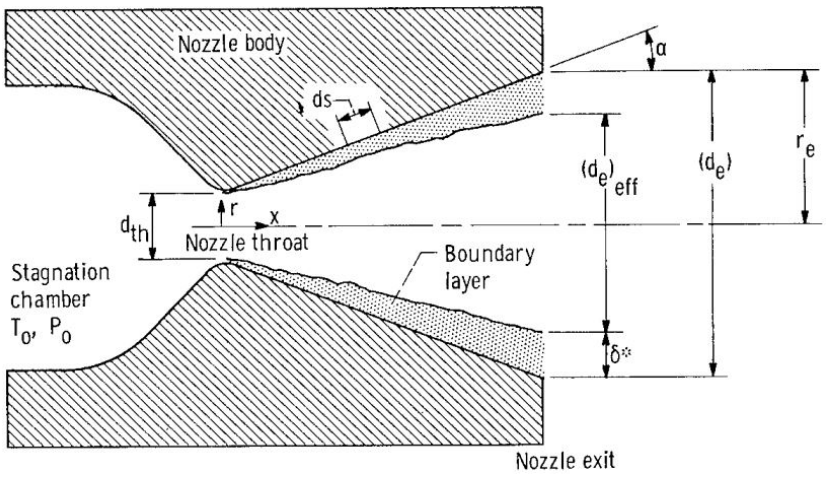

Figure 3.3: Schematic representation of a conical convergent-divergent nozzle, including boundary layer [58]

The phenomena listed above, therefore, contribute to reducing the achievable performance in terms of gas exit velocity and generated thrust. Evaluating the losses due to individual causes is not always straightforward; however, the major sources of performance reduction are related to viscosity and flow divergence. Thus, these will be studied more in detail, temporarily setting aside the other mentioned causes.
Firstly, concerning divergence, the nozzle geometry plays a fundamental role: considering, for example, a conical geometry, to achieve the most axial oriented exhaust velocity possible, small divergent angles α would be necessary (see Figure 3.3). However, this implies a high length of the nozzle that can lead, on one hand, to an excessive weight of the thruster, which must adhere to specific mission constraints, and on the other hand, it allows the viscous boundary layer to develop over a greater wall length, further reducing the velocity V_e (the boundary layer will occupy a larger portion of the nozzle exit section). Higher α angles, instead, allow for shorter nozzles, reducing losses due to boundary layer but increasing those due to velocity misalignment. Typically, divergence angles in the range on 12°-18° are chosen for conical nozzles, taking into account that divergence losses scale with the cosine of α [57]. This type of losses can be reduced by opting for a bell-shaped (parabolic) geometry of the divergent section of the nozzle; all considerations made in the previous case still hold true.

Regarding viscous effects, in cold gas microthrusters, characterized by extremely small dimensions, the thickness of the boundary layer becomes a crucial aspect to consider, especially for thrusters producing a nominal thrust on the order of a few

millinewtons, the throat radius takes values of a few hundredths of a millimeter [57]. Consequently, one can expect the boundary layer to occupy a significant portion of the nozzle exit section and to also affect the throat region. Viscous effects thus strongly impact the performance of the propulsion system. The fundamental parameter governing viscous effects is the Reynolds number Re, usually evaluated at the throat section:

$$Re_t = \frac{\rho_t \cdot V_t \cdot D_t}{\mu_t} \tag{3.18}$$

where D_t is the throat diameter and μ_t is the dynamic viscosity.
La Torre [57] provides useful values to give an idea of the Reynolds numbers reached in a GN_2 cold gas microthruster for various thrust values: considering an inlet pressure of 10 bar, a temperature of 300 K and an expansion ratio of 16, a nozzle generating 1 N of thrust will have a throat radius of approximately 0.43 mm, resulting in a Reynolds number of $Re = 1.3 \cdot 10^5$. A nozzle generating 1 mN of thrust will be characterized by $R_t = 0.014$ mm and $Re_t = 4.1 \cdot 10^3$.
One might then question whether the flow inside the nozzle is turbulent in light of the reported Reynolds numbers; however, La Torre [57] concludes that even in the case of throat Reynolds on the order of 10^5, the results obtained studying the flow with RANS equations using different turbulence models hardly differ from those obtained by neglecting such models and considering laminar flow. Further evidence will be provided in Section 3.3.2. Therefore, within a cold gas microthruster producing thrust in a range between 1 mN and 1 N, turbulence seems not to have the space or time necessary to develop [57]; hence, the boundary layer can be considered laminar. Moreover, for Reynolds numbers around 10^3 and even 10^4, one can expect that the boundary layer will not be limited to the divergent portion of the nozzle, but will also affect the throat region, reducing the mass flow across the thruster.
Observing Figure 3.3, it's noticeable that the boundary layer thickness at the exit section of the nozzle, which may affect large portions of A_e (up to 20% or more), tends to reduce the effective expansion ratio ε, i.e. the ratio A_e/A_t. The flow indeed perceives an effective exit radius that is smaller than the actual one, precisely due to the boundary layer displacement thickness $\delta*$. The reduction of the effective area ratio will imply less gas expansion in the divergent section, hence a higher p_e/p_c ratio and a loss in terms of exit velocity V_e and achievable thrust. Consequently, the thrust coefficient will be penalized by viscosity and the value calculated using Equation 3.15 will need to be properly corrected to account for the presence of the boundary layer. Some useful empirical formulations to predict the viscosity effect on the performance of cold gas microthrusters are provided by Spisz et al. [58] and by Massier et al. [59]; in particular, the following corrective factors are introduced for the thrust coefficient and flow rate:

$$C_{Fv} = \frac{17.6 \cdot e^{0.0032\frac{A_e}{A_t}}}{\sqrt{0.773 \cdot Re_t}} \tag{3.19}$$

$$\dot{m} = C_d \frac{p_c A_t}{\sqrt{RT_c}} \sqrt{\gamma} \left(\frac{2}{\gamma + 1} \right)^{\frac{\gamma+1}{2(\gamma-1)}} \tag{3.20}$$

C_d in Equation 3.20 is defined as the discharge coefficient and depends on the throat Reynolds number as follows

$$C_d = 0.8825 + 0.0079 \cdot ln(Re_t) \tag{3.21}$$

The thrust coefficient in the real case will be

$$C_F = C_{Fi} - C_{Fv} \tag{3.22}$$

where C_{Fi} is the one specified in Equation 3.15. From Equation 3.19 can be seen that the lower the Reynolds number, the higher the correction factor will be, because boundary layer effects will have a greater impact on the performance, as it is expected.

3.3 Design and performance evaluation of a cold gas thruster

In this section, we aim to compute the performance of a cold gas microthruster by comparing the values obtained using the ideal model equations with the actual flow behaviour inside the nozzle. Specifically, we seek to determine the value of the thrust coefficient C_F and assess the effect of viscosity. As mentioned earlier, after computing the Reynolds number at the throat, the objective is to demonstrate that the flow inside the nozzle is predominantly laminar and that the use of turbulence models for flow analysis does not result in significant variations in results and performance. The content presented in this section serves as a foundation for the case study to be addressed in Chapter 4, where we intend to leverage the creation of a surrogate model using the multi-fidelity Co-Kriging method to optimize the thrust coefficient as a function of the geometry of the divergent section of the nozzle, specifically bell-shaped. In particular, two databases will be created: one Low Fidelity, allowing for the determination of C_F values through flow analysis using Euler's equations, and one High Fidelity, involving the resolution of Navier-Stokes equations. The application of Co-Kriging will enable the development of a surrogate model for the thrust coefficient as a function of the angles θ_1 and θ_2 defining the nozzle bell shape (Figure 3.5), leveraging the extensive information provided by

the Low Fidelity model and a few expensive simulations, thus reducing the number of computationally intensive simulations required while maintaining high accuracy in results.

3.3.1 Nozzle geometry

Having introduced the main features of cold gas microthrusters in the preceding sections, we can now delve into a possible implementation, focusing particularly on the exhaust nozzle, which will be of the convergent-divergent type. Among the two possible shapes for the divergent section, the bell-shaped configuration will be chosen, despite the conical shape being more common in micropropulsion applications, given manufacturing requirements. Below, therefore, a design for the nozzle geometry is proposed; this geometry will be subsequently utilized for the performance evaluation of the thruster through CFD analysis.

For the construction of the geometry, reference is made to the results obtained by Rao [60] for the design of an optimum thrust nozzle, which are summarized in Figure 3.4

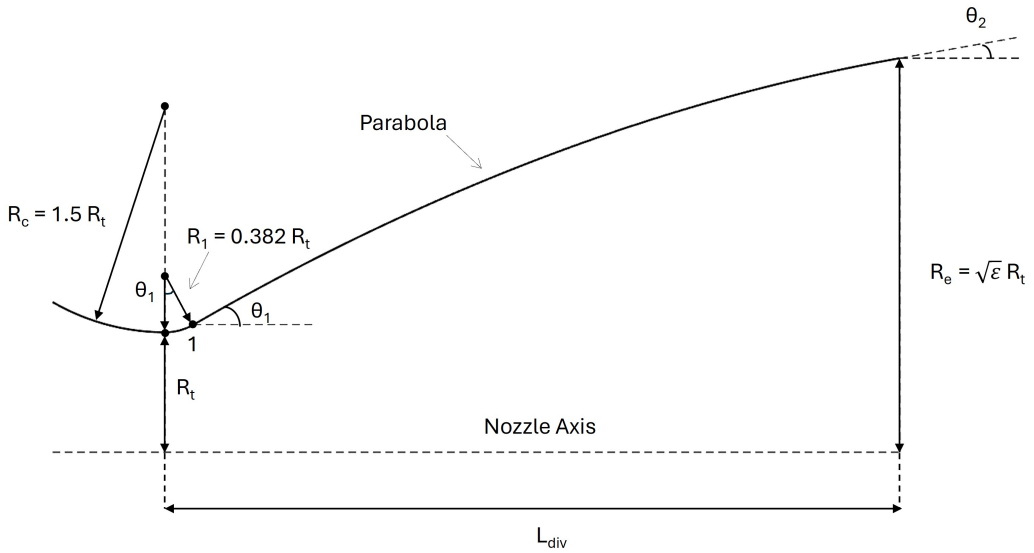

Figure 3.4: Design of a parabolic bell-shaped nozzle based on Rao's research

R_t and R_e denote the throat and exit radii of the nozzle, respectively, while ε represents the area ratio A_e/A_t introduced in the previous sections. The convergent section of the nozzle, whose treatment is deferred to a later stage, connects at the throat radius through a circular arc of radius R_c, identified by Rao as approximately 1.5 times R_t. Subsequently, a further circular fillet connects the nozzle throat to

the point indicated as '1' in Figure 3.4, whose coordinates are obtained through simple trigonometric steps:

$$\begin{aligned} x_1 &= x_t + R_1 \cdot \sin\theta_1 \\ y_1 &= R_t + R_1(1 - \cos\theta_1) \end{aligned} \tag{3.23}$$

The throat dimensions are indicated with the subscript 't' and the radius R_1 is considered as 0.382 times R_t. θ_1 thus defines the position of the starting point of the parabolic segment of the divergent and sets the slope of the parabola at point '1'. This parabolic segment extends to point '2', identified by the radius R_e and the slope of the curvature at this point is given by the value of the tangent of angle θ_2, which also characterizes the velocity vector at the exit section of the nozzle. As the angles θ_1 and θ_2 vary, the shape of the nozzle bell will consequently change, resulting in configurations of varying lengths (greater θ_1 at constant θ_2 will lead to a shorter divergent, while smaller exit angles at constant θ_1 will determine a longer nozzle).
The equation of the parabolic segment can be written as:

$$y = ax^2 + bx + c \tag{3.24}$$

where a, b and c are the unknown coefficient that must be obtained. The following assumptions can be made: the point on the nozzle axis at the throat section is considered as the origin of the reference frame, the coordinates of point '1', x_1 and y_1 are know, as well as the y coordinate of the point '2', which is R_e. Assuming the values of the angles θ_1 and θ_2, the first derivatives of the parabolic function at points '1' and '2' are known. The position x_2 of the point on the nozzle outlet section remains unknown and free, thus varying depending on the chosen combinations of the angles. Therefore, four equations in four unknowns can be written, but the system is not directly solvable because of the unknown x_2 makes it nonlinear. These equations are as follows:

$$\begin{aligned} y_1 &= ax_1^2 + bx_1 + c \\ y_2 &= ax_2^2 + bx_2 + c \\ tan(\theta_1) &= 2ax_1 + b \\ tan(\theta_2) &= 2ax_2 + b \end{aligned} \tag{3.25}$$

The non-linearity can be seen in the second listed equation. This problem can be easily solved by considering a local coordinate system for the derivation of the parabolic equation with its origin in the point '1'. So now the parabola will start in point (0,0) and end in point (x_2^*,$R_e - y_1$); by doing so, one can resolve the system presented in Equation 3.25 following the steps below:

$$
\begin{aligned}
&0 = c \\
&tan(\theta_1) = b \\
&tan(\theta_2) = 2ax_2^* + b \longrightarrow a = \frac{tan(\theta_2) - tan(\theta_1)}{2x_2^*} \\
&R_e - y_1 = \frac{tan(\theta_2) - tan(\theta_1)}{2x_2^*} x_2^{*2} + tan(\theta_1)x_2^* \longrightarrow x_2^* = \\
&= 2 \cdot \frac{R_e - y_1}{tan(\theta_2) + tan(\theta_1)}
\end{aligned}
\tag{3.26}
$$

Thus, the coordinate x_2^* is found and the coefficients a, b and c are also calculated, obtaining the equation for the parabola in the new coordinate system centered in '1'. The equation written in the original reference frame located in the throat can be easily obtained by translating the curve both in the x and y directions:

$$y = y_1 + a(x - x_1)^2 + b(x - x_1) + c \tag{3.27}$$

where a, b and c are the values calculated in Equation 3.26. The point '2' will be defined by the coordinates (x_2,R_e), where $x_2 = x_1 + x_2^*$.

As far as the convergent section of the nozzle is concerned, it has minimal impact on the performance and does not present particular constraints. Therefore, a conical geometry with a 30° angle will be chosen, ensuring a convergent length sufficient in order to limit pressure losses, as mentioned in Section 3.2.2. Additionally, a contraction ratio A_c/A_t will be selected for the same reasons. The conical part is connected to the throat by a circular segment of radius $R_c = 1.5R_t$, as shown in Figure 3.4.

3.3.2 Performance analysis of the nozzle

After detailing the steps required for designing the nozzle geometry, a case study can be set up by assigning values to the geometric parameters shown in Figure 3.4, in terms of throat and exit radii, expansion ratio ε and angles θ_1 and θ_2. Subsequently, a CFD simulation can be configured to study the flow inside the nozzle and to compare the performance obtained by simulating the real case with indicative values provided by the equations of the ideal model. In particular, as mentioned in Section 3.2.2, the effect of viscosity and Reynolds number will be assessed, demonstrating that the flow inside the nozzle is predominantly laminar.

Now, let's consider the implementation of a cold gas microthruster generating a thrust of about 15-17 mN by accelerating gaseous nitrogen GN_2 stored within a

pressurized tank. Based on typical values found in literature, plausible values for the throat diameter D_t may be around 0.25 mm, thus this value will be chosen. As for the expansion ratio, a value of 50 could be suitable for this case (even higher values are found in the literature, however, $\varepsilon = 50$ represents a good compromise). To generate a thrust on the order of approximately 15-17 mN, a stagnation pressure of 2 bar and a temperature of 300 K are considered as inlet conditions for the nozzle.
Choosing two values for the angles θ_1 and θ_2, specifically 30° for the former and 10° for the latter, yields the geometry depicted in Figure 3.5.

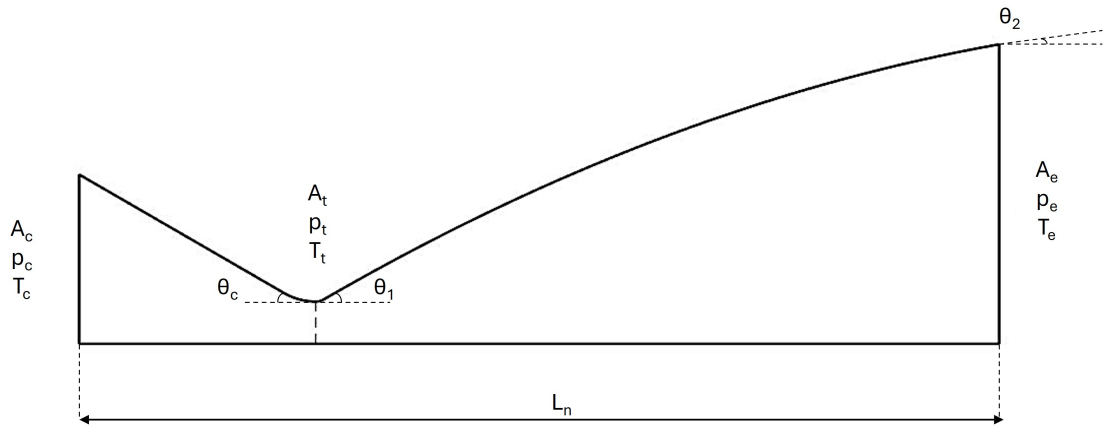

Figure 3.5: Half nozzle geometry with $R_t = 0.125$ mm, $\varepsilon = 50$, $\theta_1 = 30°$, $\theta_2 = 10°$, $\theta_c = 30°$

Given the throat radius of the nozzle, the expansion ratio, as well as the total pressure and temperature at the inlet section, it is possible to utilize the equations presented in Section 3.2.1 to obtain an initial estimate of the quantities characterizing the flow at the exit section and the thrust generated by the nozzle. It is worth noting that in the ideal case, the angles θ_1 and θ_2 have no influence on the performance, as it depends solely on γ, ε, T_c and p_e/p_c. For further clarity, the problem data are provided below:

- Throat diameter: $D_t = 0.25$ mm
- Exit diameter: $D_e = 1.767$ mm
- Expansion ratio: $\varepsilon = 50$
- Inlet total pressure: $p_c^0 = p_c = 2$ bar
- Inlet total temperature: $T_c^0 = T_c = 300$ K

- Specific heats ratio: $\gamma = 1.4$ (GN_2)

The results obtained from the ideal model are as follows:

- Nozzle pressure ratio: $p_e/pc = 6.9219 \cdot 10^{-4}$
- Exit pressure: $p_e = 138.4382$ Pa
- Exit temperature: $T_e = 37.5257$ K
- Exhaust velocity: $V_e = 738.4556$ m/s
- Exhaust Mach number: $M_e = 5.9138$
- Mass flow rate: $\dot{m} = 2.2528 \cdot 10^{-5}$ kg/s
- Thrust: $F = 0.0170$ N
- Thrust coefficient: $C_F = 1.7292$

As evidenced by the obtained results, the temperature reached by the gaseous nitrogen at the end of the expansion assumes very low values, in this case around 37 K, which is lower than the liquefaction temperature of GN_2. Now, the ideal model neglects the possibility of a phase change occurring inside the nozzle, however, in the real case, one may wonder if propellant freezing could indeed occur, leading to a substantial reduction in nozzle performance. Studies conducted in the literature, such as Ranjan et al. [61], have observed that, in the case of air as a propellant, the expected thrust reduction in the event of gas phase change did not occur. Therefore, it can be assumed that the attained temperatures do not lead to such phenomena. However, it is important to pay attention to the specific propellant used and the nozzle exit temperature may represent a critical condition to be evaluated.

We now turn to the study of the real case, introduced in Section 3.2.2. The geometry of the nozzle will certainly have a certain effect on the performance since both losses due to divergence and those related to boundary layer will depend on the angles θ_1 and θ_2. This is because the longer the thruster, the greater the length over which the boundary layer will develop, reducing the velocity at the nozzle outlet and the effective expansion ratio A_e/A_t. As previously stated, it is also expected that turbulence will neither have sufficient time nor space to develop [57], so the flow inside the nozzle will be essentially laminar, which we aim to demonstrate.
In order to study the real flow behaviour through the thruster, the calculation of the quantities of interest will be implemented in ANSYS Fluent by solving Navier-Stokes equations for the 2D axisymmetric nozzle case depicted in Figure 3.5.

Subsequently, the results obtained for the laminar viscous flow will be compared to RANS equations with different turbulence models for closure, such as Spalart-Allmaras, $k-\omega$, $k-\varepsilon$ and Reynolds Stresses.

For mesh generation, Fluent Meshing by ANSYS was employed, resulting in a mesh composed of 72947 elements, of which 72858 are quadrilateral (Quad) and 89 are triangular (Tri), with an element size of 5 μm. Additionally, a smooth transition inflation was introduced to capture the boundary layer along the nozzle walls, characterized by a transition ratio of 0.8, growth rate of 1.2 and 30 maximum layers. The resulting mesh is depicted in Figure 3.6.

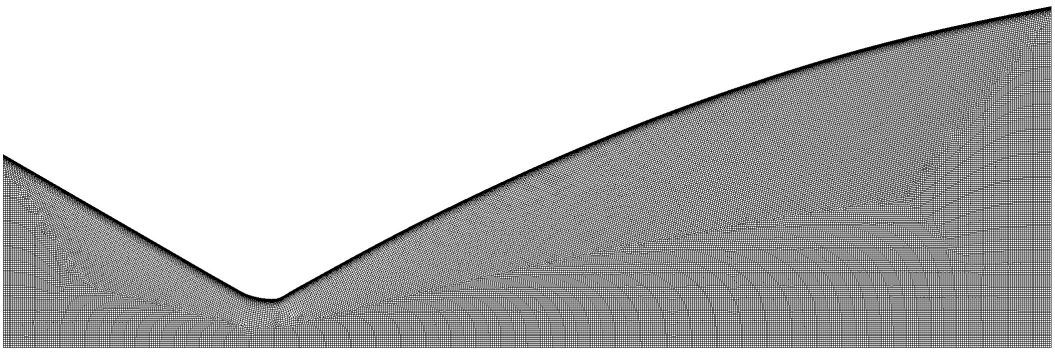

Figure 3.6: 2D nozzle computational mesh consisting of 72947 elements of size 5 μm

To solve the Navier-Stokes equations and obtain the flow field inside the nozzle, the density-based solver Fluent by ANSYS is used, considering the case study as 2D axisymmetric. The pressure value in the operating conditions is set to zero and the energy equation is included. The working fluid is gaseous nitrogen N_2, whose density is evaluated using the ideal gas equation. Nitrogen's molecular weight and specific heat C_p are set to constant values, while the viscosity is evaluated based on Sutherland's law, namely

$$\frac{\mu}{\mu_0} = \left(\frac{T}{T_0}\right)^{\frac{3}{2}} \frac{T_0 + S}{T + S} \tag{3.28}$$

where $\mu_0 = 1.663 \cdot 10^{-5}$ kg/(m s) is the reference viscosity, $T_0 = 273.11$ K is the reference temperature and S is a constant, which, for nitrogen, is 106.67 K.
The viscous model is initially set to laminar, as we expect the flow to be non-turbulent. The control parameter that characterizes the viscous effects is the throat Reynolds number, which can be found using Equations 3.18 and 3.4-3.7 for the evaluation of the density and velocity in the throat section. The Reynolds number

R_t is found to be

$$Re_t = 7396.5 \tag{3.29}$$

As far as boundary conditions are concerned, a pressure-inlet condition is set for the A_c section of the nozzle. Specifically, a stagnation pressure of 2 bar and a total temperature of 300 K are imposed. At the exit section, a pressure-outlet boundary condition is instead applied, specifying the gauge pressure as 120 Pa, slightly smaller than the pressure value p_e calculated for the ideal case- However, since the flow is supersonic in the divergent section, the condition imposed on the outlet is not utilized, as the pressure value there is computed based on the upstream flow.
Launching the simulation and activating the solution steering option (supersonic flow) starting from a Courant number of 5 up to a maximum of 200, convergence is reached after approximately 1500 iterations. The contour plot of the Mach number inside the nozzle is depicted in Figure 3.7.

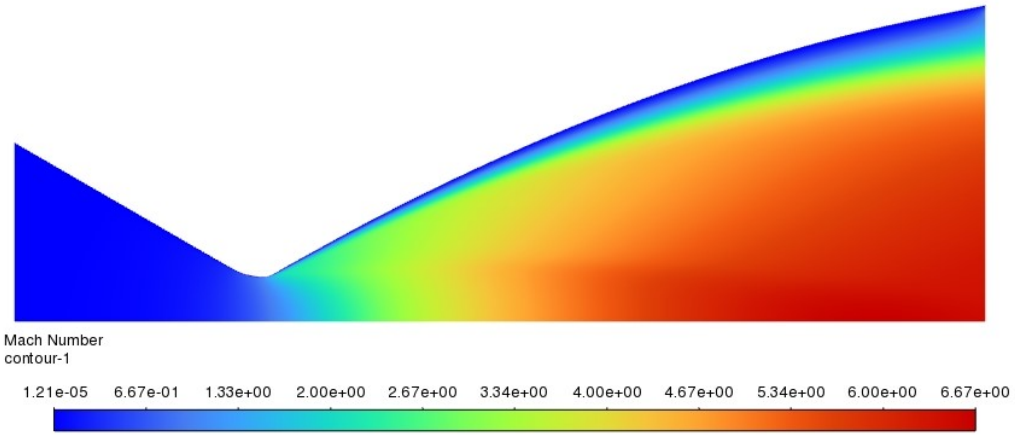

Figure 3.7: Mach contour plot for the laminar case

Observing Figure 3.7, one can draw several observations: at first glance, it becomes apparent that the boundary layer, which develops along the divergent wall but also present in the throat, occupies a significant portion of the nozzle exit section (approximately 20-25%). Moreover, this result was anticipated, given the low Reynolds number, hence a considerable reduction in thrust compared to the calculated ideal value is expected. Secondly, it can be observed that the unity value on the Mach number is reached slightly downstream of the throat section, as described in Section 3.2.2. The mass flow rate through the nozzle is determined by the throat itself and in this case, the boundary layer present there, albeit small, will reduce the $\dot{m}$ value. Furthermore, it is possible to present the trends of temperature, static pressure and velocity magnitude inside the nozzle.

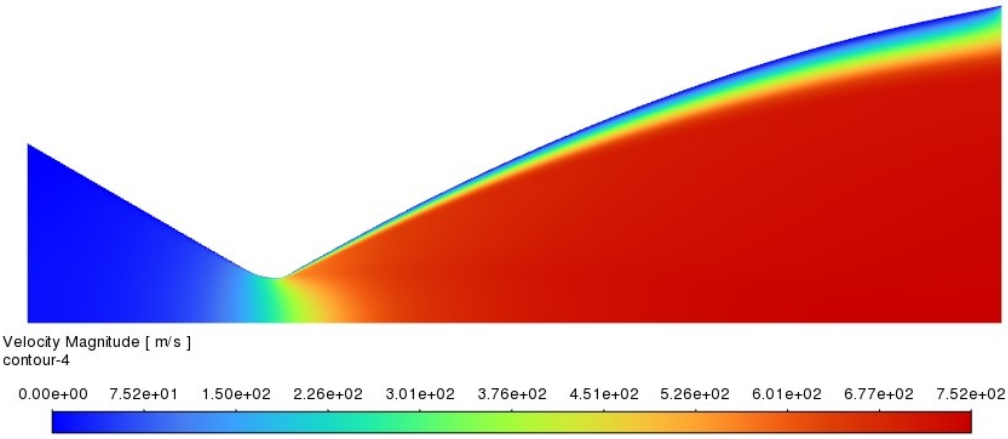

Figure 3.8: Velocity magnitude contour plot for the laminar case

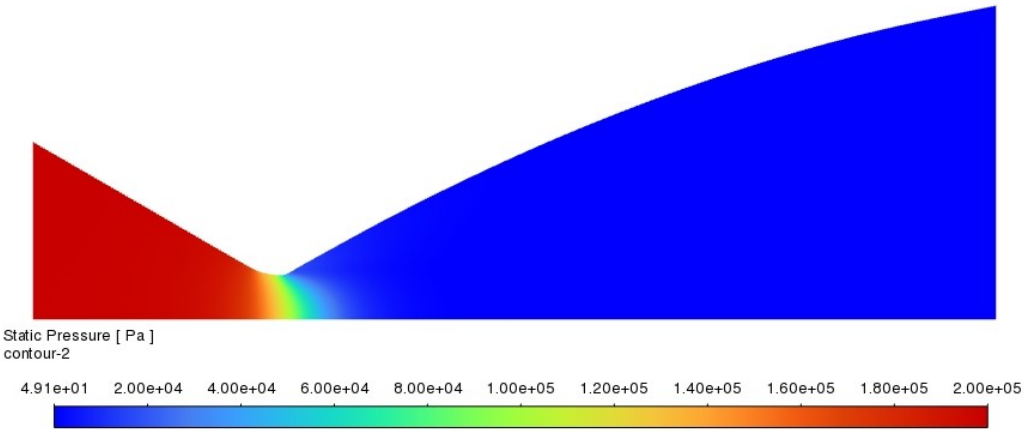

Figure 3.9: Static pressure contour plot for the laminar case

As previously mentioned, in the real case, the velocity, Mach number and pressure values are not constant across the generic section orthogonal to the nozzle axis; instead, those quantities vary along it, assuming higher values near the central region of the thruster and lower values as one moves towards the wall. Regarding static temperature, it is also non-uniform across the generic section. This is quite evident when looking at the contour plot shown in Figure 3.10: in this case, the closer one gets to the nozzle axis, the lower the temperature will be (during expansion, the gas cools), whereas towards the wall region, the temperature is higher due to the presence of the boundary layer, which insulates the all from the cooler underlying flow. Referring again to Figure 3.10, it can be noted that the minimum temperature value reached is around 30.5 K; however, the averaged value over the exit section will be higher due to the different distribution present and the higher

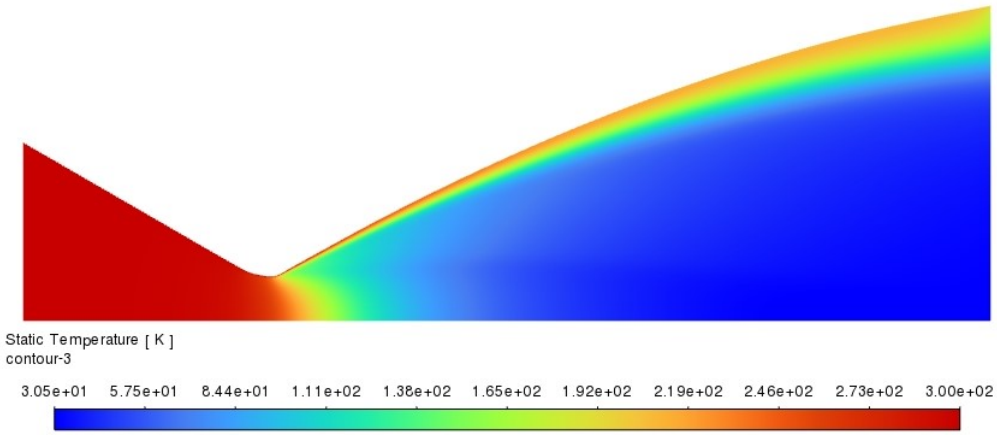

Figure 3.10: Static temperature contour plot for the laminar case

values near the wall. The same applies to velocity, Mach number and pressure, but in reverse: that is, higher values of these quantities are found near the axis, while lower values are observed near the wall.
The main results obtained are summarized below:

	$\dot{m}_e$ (kg/s)	V_e (Axial) (m/s)	p_e (Pa)	T_e (K)	Thrust (mN)	C_F	M_e
Ideal case	2.253×10^{-5}	738.456	138.438	37.526	17	1.729	5.914
Real case (Laminar)	2.167×10^{-5}	690.224	269.423	63.135	15.619	1.591	4.739

Table 3.2: Comparison of results between the ideal and real (laminar flow) models. The pressure and temperature values are averaged over the exit area, while the Mach number and axial velocity values are mass-averaged

The obtained results reflect what has been discussed so far. Particularly noticeable is a significant reduction in axial velocity, as the gas velocity at the outlet section of the nozzle will be directed according to the angle θ_2, resulting in the presence of a radial component and a loss in terms of axial velocity. Moreover, the presence of the boundary layer impacts both on the value of V_e and the exit pressure p_e, which is higher than that calculated according to the ideal model: the reduction in effective area ratio ε causes the gas to expand less than theoretically calculated, also resulting in a higher average exit temperature. The introduction of these various losses thus leads to a reduction in achievable thrust and thrust coefficient C_F by approximately 8%.

Further simulations were conducted to observe the effect of different turbulence model on the results compared to the laminar case previously evaluated. Specifically, RANS equations were solved using the following turbulence models: Spalart-Allmaras (1 equation), k-ω (2 equations), k-ϵ (2 equations) and Reynolds Stresses (5 equations), implemented in ANSYS Fluent. As anticipated by La Torre [57], it is expected that the reference quantities values reported in Table 3.2 will vary very little compared to the laminar case for the calculated Reynolds number; this is confirmed by the results obtained (Table 3.3), where the highest relative percentage error was obtained in the case of the k-ϵ model, of the order of 1.22%. It can therefore be concluded that the flow in the nozzle is laminar and the results reported in Table 3.2 are valid.

Viscosity Model	$\dot{m}_e$ (kg/s)	V_e (m/s)	$p_e A_e$ (Pa m^2)	Thrust (mN)	C_F	Percentage Error (%)
Laminar	2.167 10^{-5}	690.224	6.612 10^{-4}	15.619	1.591	
Spalart-Allmaras	2.166 10^{-5}	690.221	6.755 10^{-4}	15.624	1.591	-2.90 10^{-2}
k-ω	2.165 10^{-5}	690.296	6.708 10^{-4}	15.617	1.5908	1.17 10^{-2}
k-ϵ	2.156 10^{-5}	682.932	7.067 10^{-4}	15.428	1.572	1.22
Reynolds Stresses	2.168 10^{-5}	689.931	6.759 10^{-4}	15.637	1.593	-1.14 10^{-1}

Table 3.3: Results comparison between different viscosity models for RANS

Chapter 4

Co-Kriging based cold gas thruster geometry optimization

In the previous chapter, a detailed description of the main features of cold gas thrusters used for small satellite attitude control was provided, along with a series of equations useful for predicting their performance. Additionally, a specific case study was introduced, considering a thruster consisting of a converging-diverging nozzle whose geometry is depicted in Figure 3.5. The performance in terms of thrust, mass flow rate and C_F were evaluated through a CFD simulation using ANSYS Fluent and the main results are reported in Table 3.2. The obtained results are obviously dependent on the geometry of the considered nozzle, particularly on the area ratio A_e/A_t, which determines how much the gas can expand as it passes through the nozzle, increasing its velocity until exhausting into the vacuum at A_e. The throat diameter D_t also has a strong impact on performance and regulates the effects of viscosity inside the nozzle, contributing to the definition of the throat Reynolds number Re_t. However, fixing the geometric dimensions that define the throat area and the exit area of the thruster, the flow energy losses that occur will vary with the angles θ_1 and θ_2, which define the geometry of the parabolic divergent and the overall length of the nozzle. As discussed previously, indeed, high values of θ_1, for the same θ_2, will result in a shorter nozzle, thus lower losses due to boundary layer, but higher losses due to flow divergence. Fixing θ_1 instead, lower values of angle θ_2 will result in the outlet flow being slightly inclined relative to the axial direction, but greater lengths will be required and therefore the effect of the boundary layer will be higher. As a consequence of all this, it is easy to notice how the thrust coefficient C_F can vary depending on the angles θ_1 and θ_2, so it would be ideal to identify the combination of these angles that maximizes

this parameter. Another fundamental aspect concerns the mass of the thrusters present on the satellite: to meet the requirements in terms of thrust needed to perform the required maneuvers, a satellite mounts a certain number of thrusters, which can even be eight or ten. Now, a nozzle like the one depicted in Figure 3.5, given its extremely small dimensions, will have a substantially negligible weight, however, it must be considered that a cold gas thruster is not just the nozzle itself, but also includes a solenoid valve, a supply line with respective valves and the pressurized tank for gas storage. It follows that the mass of the nozzle can represent an important parameter to be taken into consideration in order not to exceed the constraints imposed by the mission.
Therefore, aiming to select the geometry that allows maximizing the thrust coefficient and thus the performance, it will still be necessary to ensure an ideally minimum value of the nozzle's mass. Hence arises a multi-objective optimization problem that involves a trade-off between performance and the weight of the propulsion system. Regarding the mass of the thruster, this can be easily derived by calculating the volume of the nozzle envelope and multiplying it by the density of the material composing it, for each value of θ_1 and θ_2 (Figure 3.5). Conversely, knowing the value of the C_F for different combinations of these angles requires numerous CFD simulations, which can take a lot of time and computational resources, often not available for the evaluations necessary to create a sufficient database. In this context, the approach to generating a surrogate model using multi-fidelity techniques proves to be of great help: following the construction of a Low Fidelity database that is computationally inexpensive, the number of high-fidelity evaluations of the nozzle performance is greatly reduced, without sacrificing a high level of accuracy. Furthermore, the surrogate model allows predicting the performance even for untested combinations of θ_1 and θ_2 angles, providing the necessary data to carry out the aforementioned multi-objective optimization.

In this chapter, the Co-Kriging technique introduced in Chapters 1 and 2 is employed in order to obtain the nozzle's thrust and thrust coefficient values as a function of its geometry, thus enabling the estimation of these values for a large number of combinations of the angles θ_1 and θ_2 that characterize the divergent section. The resulting surrogate model will subsequently be used to optimize the C_F in relation to the mass of the nozzle in the various estimated configurations.

4.1 Selecting the Low Fidelity and High Fidelity models

First and foremost, the multi-fidelity approach requires the definition of a Low Fidelity (LF) model and a High Fidelity (HF) one. As discussed in Chapter 1,

the types of fidelity used to represent a given physical system can be manifold. However, in the literature, four main categories of fidelity are primarily encountered, which are described by Fernandez-Godino et al. [3] and outlined below:

1. Firstly, there is the 'physics' category, where Low and High Fidelity models differ in terms of the physical model chosen to represent the problem, specifically, the fundamental assumptions that lay at the base of both models. The LF model, for example, will often be a simplified version of the HF one, obtained by neglecting some phenomena and behaviours with the aim reducing the model complexity and speeding up the simulations

2. The second category concerns the accuracy of the numerical solution considered. Specifically, what distinguishes the two Low and High Fidelity models may be a different level of discretization of the equations underlying the physical model or partially converged solutions are considered in the LF case. In some instances, the same equations are solved, but the computational mesh varies, being finer in the HF case

3. Another category involves considering the same physical model, without introducing simplifications, but using different equations or different solution calculation methods for the Low and High Fidelity cases

4. Finally, when experimental measurements are performed, these always represent the highest fidelity level, while the Low Fidelity model may derive from a computational simulation or other fields

Regarding the application case of calculating the thrust coefficient of the cold gas microthruster, the choice of Low and High Fidelity models will involve considering the same physical case, as well as the same computational mesh, but solving different equations for the two cases. In particular, the Low Fidelity model will compute the nozzle performance by solving the Euler equations, which are derived from the Navier-Stokes equations neglecting viscosity and thermal conductivity, while in the High Fidelity case, Navier-Stokes equations will be solved, considering laminar flow as the viscous model (as demonstrated in Section 3.3.2, turbulence can be neglected).
Another possibility would be to leverage Equation 3.15 for computing the thrust coefficient of the ideal nozzle model, thus obtaining a Low Fidelity database to combine, for example, with values obtained from solving the Navier-Stokes equations. The strength of this approach lies in the ability to generate a Low Fidelity database of any size instantaneously, as Equation 3.15 will always yield the same C_F value regardless of the nozzle geometry, i.e., regardless of the angles θ_1 and θ_2, which are the variables of the HF and LF objective functions. Using the ideal model for Low Fidelity, a higher error can be expected for the same selected

High Fidelity points; however, the cost savings in computational expense (no longer requiring CFD simulations of the Low Fidelity model) will provide a greater budget to include more expensive points in the Co-Kriging model generation.

4.1.1 Low Fidelity Model

As far as the implementation of the Low Fidelity model is concerned, there are therefore two possibilities: firstly, one may consider the real physical model for the flow inside the nozzle and evaluate its performance by solving the Euler equations. Since these equations neglect the effect of viscosity, it is expected that losses related to the development of the boundary layer on the walls of the nozzle will no longer be considered, hence the thrust coefficient will be higher compared to that evaluated in the presence of viscous effects. However, the C_F will obviously be lower compared to that obtainable from Equation 3.15 because the effect of the geometry of the divergent section of the nozzle persists, i.e. the losses due to flow divergence, depending on the combination of θ_1 and θ_2. Secondly, the Low Fidelity implementation can be realized following the ideal nozzle model presented in Chapter 3, where neither viscosity nor flow divergence are considered.

Before analyzing the two proposed cases, it is necessary to specify what is usually called the design of experiment (DOE), i.e., the sampling strategy for selecting inexpensive points in the space of the parameters θ_1 and θ_2. As anticipated in Chapter 2, there can be various types of DOEs, including full-factorial sampling, Latin-Hypercube sampling, etc. For the case study, given the fact that the thrust coefficient is a function of just two variables, a full-factorial sampling approach may be a good strategy in order to generate a space-filling Low Fidelity sample, acquiring information uniformly distributed in the design space. The latter represents the space in which the two parameters θ_1 and θ_2 vary, so it is of primary importance to first define the range of interest for the two variables by setting lower and upper limits on the values they assume. In particular, concerning θ_1, angles smaller than (20-25°) result in a particularly reduced inclination of the initial portion of the divergent parabola, leading to a higher nozzle length and thus greater losses due to boundary layer effects. Conversely, excessively high values of θ_1 (above 40-45°) entail a significant initial curvature, resulting in shorter nozzles but with a risk of flow separation and a significant deterioration in performance.
Regarding θ_2, which significantly contributes to determining losses due to velocity divergence, values too small, around 0°, although ideal for maximizing the axial component V_e, lead to very long nozzles, hence heavy, with significant boundary layer development. Values greater than 15° are instead less than ideal as they result in reduced performance.

For the first case mentioned, in order to thoroughly explore the effect of geometry on the thrust coefficient, a 5x5 grid of angle values in the ranges [25°, 45°] for θ_1 and [0°, 15°] for θ_2 is chosen. Particularly, for θ_1 we choose 5 equally spaced points in [25°-45°], while for θ_2, selecting equally spaced samples in [0°-15°] would lead to decimal values for the angle (3.75°, 7.5°, 11.25°), which may not be the best. To overcome this issue and obtain integer values for the angles θ_2, the values are rounded up, resulting in θ_2 = [0° 4° 8° 12° 15°]. As for θ_1, we have θ_1 = [25° 30° 35° 40° 45°]. The Low Fidelity database will therefore consist of 25 points uniformly distributed in the parameters space, as depicted in Figure 4.1.

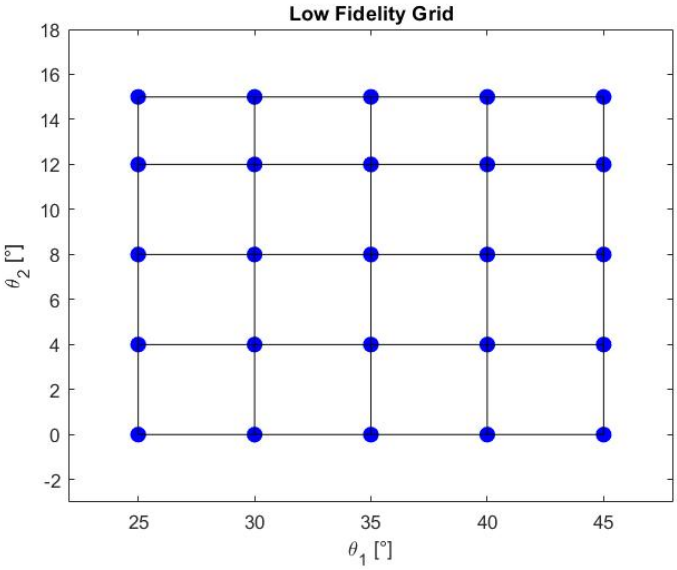

Figure 4.1: Low Fidelity model sample grid (case 1, real nozzle)

Consequently, to construct the Low Fidelity database, it will be necessary to perform 25 CFD simulations involving the resolution of the Euler equations to calculate the quantities of interest for evaluating the thrust coefficient. These simulations will require a certain computational budget, which, however, will be lower compared to that required by High Fidelity simulations. Therefore, it is reasonable to be able to carry out the predetermined number of CFD analysis for the low fidelity case.

Studying the real nozzle model through the resolution of the Euler equations introduces a certain level of simplification, yielding results in terms of mass flow rate, p_e, T_e, V_e and thrust that overestimate the values obtainable, for instance, through the resolution of the Navier-Stokes equations. The Euler equations, derived from the Navier-Stokes equations by neglecting the effects of viscosity and thermal conductivity, in the absence of heat sources, can be written as follows [62]:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{V}) = 0 \tag{4.1}$$

$$\rho \left(\frac{\partial \mathbf{V}}{\partial t} + \mathbf{V} \cdot \nabla \mathbf{V} \right) = -\nabla p + \rho \mathbf{f} \tag{4.2}$$

$$\rho \left(\frac{\partial E}{\partial t} + \mathbf{V} \cdot \nabla E \right) = -\nabla \cdot (p\mathbf{V}) + \rho \mathbf{V} \cdot \mathbf{f} \tag{4.3}$$

The use of such equations, however, is physically acceptable when the flow is characterized by high Reynolds numbers, where the effects of the boundary layer and thermal conductivity are confined to regions of negligible size compared to the study domain [62]. In the case of a cold gas microthruster, as analyzed in Section 3.3.2, the assumption of inviscid flow is quite strong, given the Reynolds number on the order of 10^3 and the results obtained in Table 3.2. Nevertheless, it allows for a rapid and computationally inexpensive estimation of the thrust coefficient's behaviour as a function of nozzle geometry. In practice, the study of the real nozzle model presented in Section 3.2.2 through Euler's equations eliminates the boundary layer effects, thus constituting a Low Fidelity model compared to one where such effects are not neglected. The difference between the High Fidelity and Low Fidelity models, therefore, lies in considering or disregarding viscous phenomena, which will be extremely important and significantly impact performance in the introduced case study.
Equations 4.1, 4.2 and 4.3 can also be written in the conservative integral form as follows [63]:

$$\frac{\partial}{\partial t} \int_{\Omega} \mathbf{W} \, d\Omega + \oint_{\partial\Omega} \mathbf{F}_c \cdot \mathbf{n} \, dS = \int_{\Omega} \mathbf{Q} \, d\Omega \tag{4.4}$$

where $\mathbf{W}$ represent the vector of the conservative variables, i.e. $\mathbf{W} = [\rho, \, \rho\mathbf{V}, \, \rho E]^T$, $\mathbf{F}_c$ is the convective fluxes vector, i.e. $\mathbf{F}_c = [\rho\mathbf{V}, \, \rho\mathbf{V}\mathbf{V} + p\mathbf{I}, \, \rho E\mathbf{V} + p\mathbf{V}]^T$ and, finally, $\mathbf{Q}$ is the source terms vector, $\mathbf{Q} = [0, \, \rho\mathbf{f}, \, \rho\mathbf{V} \cdot \mathbf{f}]^T$ [63]. Written in the form of Equation 4.4, Euler's equations, that represent the convection of mass, momentum and energy in an inviscid fluid, allow for the study and representation of discontinuity phenomena within the flow, such as shocks or expansion fans, which can occur within the nozzle [63].

Should one wish to utilize the ideal nozzle model for generating a Low Fidelity database, i.e., calculating the thrust coefficient using Equation 3.15, which is independent of the divergent geometry, it would be potentially feasible to choose a significantly high number of points, given the fact that solving a simple equation entails virtually zero computational cost. The downside, however, is that it involves

considering a model whose performance remains unaffected by the geometric parameters one aims to optimize for maximizing the thrust coefficient, since Equation 3.15 always yields the same C_F value for every combination of θ_1 and θ_2. Opting consistently for the full-factorial sampling technique for $\theta_1 \in [25° \; 45°]$ and $\theta_2 \in [0° \; 15°]$, considering 100 points yields the following Low Fidelity sampling grid:

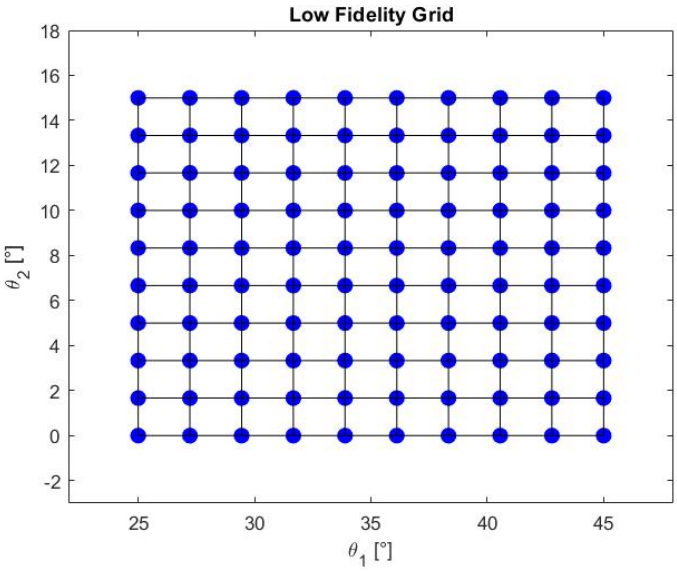

Figure 4.2: Low Fidelity model sample grid (case 2, ideal nozzle)

For the sake of completeness, Equation 3.15 is given below:

$$C_F = \sqrt{\frac{2\gamma^2}{\gamma - 1}\left(\frac{2}{\gamma + 1}\right)^{\frac{\gamma+1}{\gamma-1}}\left[1 - \left(\frac{p_e}{p_c}\right)^{\frac{\gamma-1}{\gamma}}\right]} + \frac{p_e A_e}{p_c A_t} \tag{3.15}$$

4.1.2 High Fidelity model

In the previous subsection, two possible implementations of a Low Fidelity model were introduced, only the first of which requires performing CFD simulations. Specifically, for the first case, since solving Euler's equations is not particularly computationally demanding, 25 sample points were chosen in the parameter space to evaluate the nozzle's thrust coefficient. It was also highlighted that the main difference between the value of C_F calculated with the Low Fidelity model and that evaluated with the High Fidelity one lies in neglecting viscous effects. Now, to obtain a value as consistent as possible with the real value expected from the

physics of the problem, it will be crucial to quantify the losses in the flow due to the boundary layer, that develops on the nozzle walls. The Reynolds number at the nozzle throat section, Re_t, calculated in Chapter 3, is approximately $Re_t = 7396.5$. It has been demonstrated in Section 3.3.2 that the flow within the nozzle is predominantly laminar. This implies that turbulent fluctuations of the primary quantities are negligible, therefore, the equations used to study the gas behaviour inside the nozzle will not be the RANS (Reynolds-Averaged Navier-Stokes equations, which require the introduction of turbulence models to address the closure problem). Instead, it will be necessary to solve the Navier-Stokes equations, which describe the conservation of mass, momentum and energy and can be written as follows for a compressible viscous flow, in the absence of volumetric heat sources Q_v [62]:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{V}) = 0 \tag{4.5}$$

$$\rho \left(\frac{\partial \mathbf{V}}{\partial t} + \mathbf{V} \cdot \nabla \mathbf{V} \right) = -\nabla p + \nabla \cdot \left[\mu (\nabla \mathbf{V} + \nabla \mathbf{V}^T) - \frac{2}{3} \mu (\nabla \cdot \mathbf{V}) \mathbf{I} \right] + \rho \mathbf{f} \tag{4.6}$$

$$\begin{aligned} \rho \left(\frac{\partial E}{\partial t} + \mathbf{V} \cdot \nabla E \right) = &-\nabla \cdot (p \mathbf{V}) + \nabla \cdot \left\{ \left[\mu (\nabla \mathbf{V} + \nabla \mathbf{V}^T) - \frac{2}{3} \mu (\nabla \cdot \mathbf{V}) \mathbf{I} \right] \cdot \mathbf{V} \right\} \\ &+ \rho \mathbf{V} \cdot \mathbf{f} + \nabla \cdot (k \nabla T) \end{aligned} \tag{4.7}$$

Equations 4.5, 4.6 and 4.7 can also be written in the conservative integral form as follows [63]:

$$\frac{\partial}{\partial t} \int_{\Omega} \mathbf{W} \, d\Omega + \oint_{\partial \Omega} (\mathbf{F}_c - \mathbf{F}_v) \cdot \mathbf{n} \, dS = \int_{\Omega} \mathbf{Q} \, d\Omega \tag{4.8}$$

where $\mathbf{W}$ represent the vector of the conservative variables, i.e. $\mathbf{W} = [\rho, \, \rho \mathbf{V}, \, \rho E]^T$, $\mathbf{F}_c$ and $\mathbf{F}_v$ are, respectively, the convective fluxes vector and the viscous fluxes vector, i.e. $(\mathbf{F}_c - \mathbf{F}_c) = [\rho \mathbf{V}, \, \rho \mathbf{V} \mathbf{V} + p \mathbf{I} - \boldsymbol{\tau}, \, \rho E \mathbf{V} + p \mathbf{V} - \boldsymbol{\tau} \cdot \mathbf{V} - k \nabla T]^T$ and, finally, $\mathbf{Q}$ is the source terms vector, $\mathbf{Q} = [0, \, \rho \mathbf{f}, \, \rho \mathbf{V} \cdot \mathbf{f}]^T$ [63].
Solving these equations for the laminar case does not present substantial difficulties; however, the computational cost is higher compared to solving the Euler equations, used in the Low Fidelity model. An example of a High Fidelity simulation for studying the flow in a cold gas microthruster nozzle was already performed in Section 3.3.2. In that case, for a combination of angles $\theta_1 = 30°$ and $\theta_2 = 10°$, the fields of Mach number, velocity, pressure, and temperature were obtained and are shown in Figures 3.7-3.10, where the effect of viscosity is clearly visible. Since the boundary layer will occupy a significant portion of the exit section and will typically also be present at the throat, thrust coefficients are expected to be lower than those

calculated with the Low Fidelity model. Additionally, one can observe the combined effect of divergence losses and boundary layer losses as the nozzle length varies.

As far as the selection of sample points for the High Fidelity model is concerned, their number is primarily constrained by the available computational resources. Ideally, one would aim to find a minimum threshold of points such that the surrogate model created with Co-Kriging accurately approximates the expensive objective function, ensuring a small error. However, finding a general criterion to define this threshold is quite challenging to identify and implement, as the choice of sample points for both High and Low Fidelity models heavily depends on the specific problem at hand. Commonly, for constructing the expensive database, a subset of points from the Low Fidelity grid is considered to facilitate the calculation of the scaling factor ρ for the Co-Kriging. Thus, it is often assumed that the High Fidelity points are nested within the Low Fidelity sample for simplicity. To achieve a good approximation, the expensive grid is expected to be somewhat space-filling, meaning that the accurate objective function values should be known at points distributed in a way that adequately explores the parameter space. In Chapter 2, three different algorithms for adaptive sampling were introduced and validated using various test functions, both single-variable and two-variable. Since the thrust coefficient of the cold gas thruster is a function of two variables, namely $theta_1$ and $theta_2$, the results obtained from applying the Maximum Variance, Maximum Expected Improvement and Maximum Information Gain algorithms to the Himmelblau, Booth and Branin functions can be observed to select one for application to the current case study. Given that CFD simulations are now required to populate the High Fidelity database, it will be essential to minimize the number of points to be evaluated and, importantly, to choose an algorithm with a low computational burden, in addition to the already high cost of the simulations. Consequently, among the Maximum Expected Improvement and Information Gain algorithms, which generally tend to identify the minimum number of points to add, the former might potentially be the most optimal candidate for application to the construction of the surrogate model for the thruster's thrust coefficient. Therefore, a Co-Kriging model will be implemented following the Maximum E[I(x_1,x_2)] criterion for the High Fidelity data points. However, considering that from the tests conducted, E[I(x_1,x_2)] and IG(x_1,x_2) generally exhibit comparable performances, but for certain functions, such as the Himmelblau function, the latter criterion outperforms the former, it is still worth to investigate the behaviour of the algorithm based on information gain for this case study. Therefore, the generation of the Co-Kriging surrogate using this criterion is also included.

4.1.3 Geometry and computational mesh

As for the nozzle geometry, all the values reported in Section 3.3.2 apply; specifically, we have

- Throat Diameter: $D_t = 0.25$ mm
- Exit Diameter: $D_e \approx 1.7677$ mm
- Expansion Ratio: $\varepsilon = 50$
- Convergent Angle: $\theta_c = 30°$
- Inlet Diameter: $D_c = 1$ mm
- Convergent Length: $L_{conv} \approx 0.7$ mm

For the simulations necessary to construct the Low Fidelity database, the angles θ_1 and θ_2 are varied as previously described: $\theta_1 = [25° \ 30° \ 35° \ 40° \ 45°]$ and $\theta_2 = [0° \ 4° \ 8° \ 12° \ 15°]$, resulting in 25 combinations of these parameters, leading to the LF grid depicted in Figure 4.1. All considerations regarding the effect of these angles on the nozzle length and the geometry of the parabola still apply. In particular, it is noted that the thruster will be longer (with greater mass, increased boundary layer losses, and reduced divergence losses) for low values of θ_1 at a constant θ_2 and for low values of θ_2 at a constant θ_1. This assertion can be verified by examining Figures 4.3 and 4.4, which show the impact of the angles on the shape and length of the divergent section.

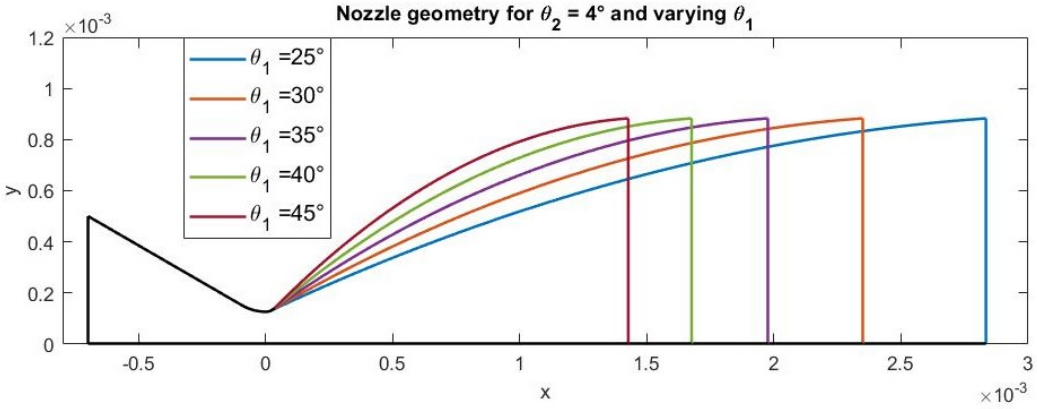

Figure 4.3: Nozzle geometry for a fixed value of $\theta_2 = 4°$ and varying θ_1 in [25°, 45°]

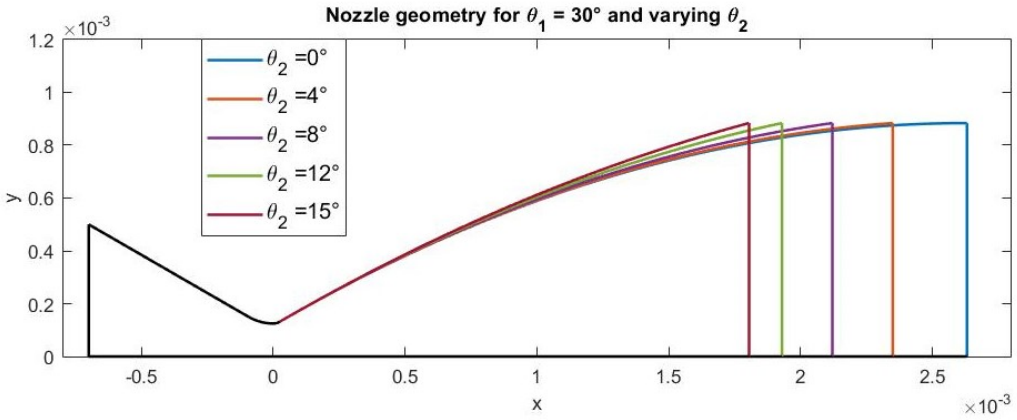

Figure 4.4: Nozzle geometry for a fixed value of $\theta_1 = 30°$ and varying θ_1 in [0°, 15°]

Once the combination of θ_1 and θ_2 is chosen, the geometry for the specific case is fixed and one can proceed to generate the computational mesh for the simulations using ANSYS Meshing. A 2D axisymmetric simulation is then set up. First and foremost, it is necessary to specify the parts of the geometry on which the boundary conditions will be applied. Specifically, the following will be defined:

- Inlet: inlet section of the nozzle of area A_c where a pressure-inlet boundary condition will be imposed
- Outlet: outlet section of the nozzle of area A_e where a pressure-outlet boundary condition will be imposed
- Axis: symmetry axis for the axisymmetric nozzle geometry
- Walls: nozzle walls, where wall-type boundary condition will be imposed

Regarding the mesh generated by ANSYS, it is unstructured and primarily composed of quadrilateral elements, although a few triangular elements are present in areas where curvature of the nozzle walls necessitates them. To accurately capture the boundary layer that forms near the wall, an inflation mesh is applied. This consists of a series of thin layers that follow the wall's shape, starting from an extremely thin initial layer and gradually becoming thicker. These layers allow for the capture of velocity, temperature, and pressure gradients, which are pronounced near the nozzle surface, thereby improving the accuracy in evaluating the flow behaviour in the wall region. Specifically, the inflation layer generation criteria are set to Smooth Transition, with a Transition Ratio of 0.8, a Growth Rate of 1.2, and a maximum number of 30 layers. As for the elements' size, a value of 5 μm is

set in order to obtain a sufficiently fine grid. The number of elements and nodes will obviously depend on the nozzle's geometry and on the θ_1 and θ_2 combination: the smaller the thruster, the fewer the mesh elements will be, maintaining fixed their size. The number of elements and nodes for the different angles combination is reported in Table 4.1:

θ_1 = **25°**			
θ_2	**Size** [m]	**Nodes**	**Elements**
0°	0.000005	116015	115066
4°	0.000005	99373	98524
8°	0.000005	87202	86391
12°	0.000005	82203	81423
15°	0.000005	72167	71437

θ_1 = **30°**				θ_1 = **35°**			
θ_2	**Size** [m]	**Nodes**	**Elements**	θ_2	**Size** [m]	**Nodes**	**Elements**
0°	0.000005	97147	96318	0°	0.000005	83300	82552
4°	0.000005	85931	85155	4°	0.000005	75459	74747
8°	0.000005	77212	76479	8°	0.000005	69027	68343
12°	0.000005	70324	69617	12°	0.000005	63911	63258
15°	0.000005	65983	65305	15°	0.000005	60515	59878
θ_1 = **40°**				θ_1 = **45°**			
θ_2	**Size** [m]	**Nodes**	**Elements**	θ_2	**Size** [m]	**Nodes**	**Elements**
0°	0.000005	72555	71870	0°	0.000005	63901	63271
4°	0.000005	66872	66221	4°	0.000005	59824	59210
8°	0.000005	62259	61628	8°	0.000005	56468	55878
12°	0.000005	58396	57790	12°	0.000005	53459	52875
15°	0.000005	55737	55133	15°	0.000005	51378	50803

Table 4.1: Mesh details for different combinations of θ_1 and θ_2

A representation of the computational mesh for $\theta_1 = 30°$ and $\theta_2 = 12°$ is shown in the figure below:

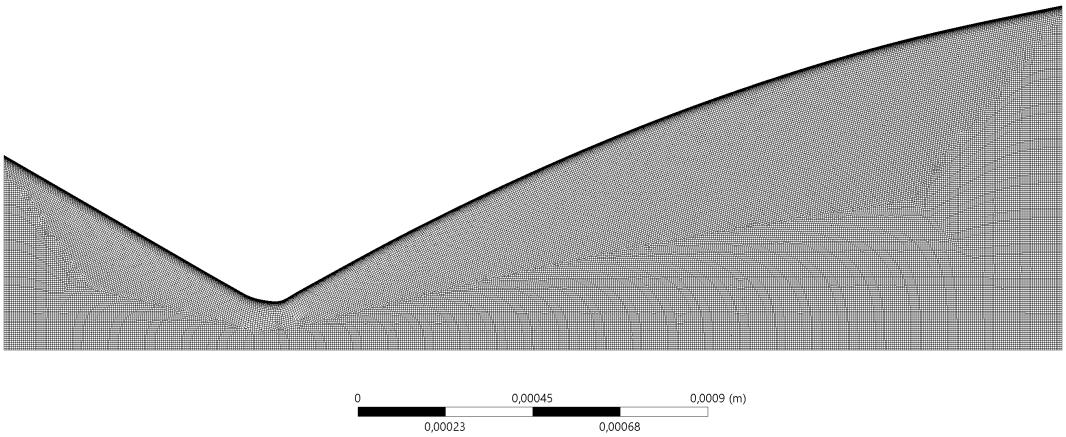

Figure 4.5: Computational mesh for $\theta_1 = 30°$ and $\theta_2 = 12°$

4.1.4 Boundary conditions and working fluid

In terms of boundary conditions, the same ones will be imposed for both the Low and High Fidelity models; specifically we have:

- **Pressure-inlet**:

  Gauge Total Pressure = 2 10^5 Pa
  Total Temperature = 300 K

- **Pressure-outlet**:

  Gauge Pressure = 120 Pa
  Total Temperature = 300 K

Supersonic/Initial Gauge Pressure will be unused if hybrid initialization is employed. As far as the working fluid expanding through the nozzle, gaseous nitrogen (GN_2) is selected; for the Low Fidelity model, viscosity is neglected, so it is only necessary to specify density, specific heat and molecular weight, while for the High Fidelity model one has to also specify the viscosity. Thus we have:

- **Low Fidelity model** GN_2:

  Density $\longrightarrow$ ideal-gas kg/m^3
  Specific Heat $C_p \longrightarrow$ constant = 1040.67 J/(kg K)
  Molecular Weight $\longrightarrow$ constant = 28.0134 kg/kmol

- **High Fidelity model** GN_2:

Density $\longrightarrow$ ideal-gas kg/m^3
Specific Heat C_p $\longrightarrow$ constant = 1040.67 J/(kg K)
Molecular Weight $\longrightarrow$ constant = 28.0134 kg/kmol
Thermal Conductivity $\longrightarrow$ constant = 0.0242 W/(m K)
Viscosity $\longrightarrow$ Sutherland's law kg/(m s)

Sutherland's law is shown in Equation 3.28.

4.1.5 Solver and results

The reported equations for studying the flow in the nozzle for both the Low Fidelity and High Fidelity models are essentially partial differential equations. To be solved numerically, these equations require a discretization method, which approximates the differential or integral terms present in the equations, resulting in an algebraic system that is easier to solve. The ANSYS Fluent solver implements the finite volume method to approximate the volume and surface integrals that appear in Equations 4.4 and 4.8. The computational domain is discretized into finite volumes, or cells, each with a computational node at its center, where the integral mean value of the quantity of interest over the cell volume is concentrated. In addition to the central node, each cell is characterized by boundary surfaces, or faces, on which the boundary conditions are imposed. The values at the cell centers allow for the calculation of fluxes through the cell surfaces and the values of the quantity of interest on the faces, utilizing methods such as interpolation (upwind methods) or approximate solutions to the Riemann problem. For solving the equations of the Low and High Fidelity models, a density-based solver employing finite volumes is utilized, incorporating an implicit solution formulation. To compute the fluxes between the cells, the Roe-FDS (Flux Difference Splitting) scheme is employed, enabling the solution of an approximated Riemann problem. A second-order upwind scheme is utilized for approximating the convective terms in the momentum and energy equations. Regarding the stability of the numerical method, since the adopted scheme is implicit, it is characterized by higher stability than an explicit one, where the time step Δt is limited by the Courant-Friedrichs-Lewy (CFL) condition:

$$CFL = \frac{u\Delta t}{\Delta x} \tag{4.9}$$

where u is the flow velocity and Δx is the cell size. Generally, for explicit schemes, the Courant number must be less than 1, but can assume slightly higher values depending on the problem, while for implicit ones can even be higher, due to the greater numerical stability characterizing the latter schemes. For the nozzle case study a Full Multi-Grid (FMG) approach is selected, aiming at enhancing convergence by solving equations on four grid levels, from coarser to finer. Moreover,

solution steering is applied, modifying the solver's settings in terms of Courant number to accelerate convergence and enhance numerical stability: particularly, the CFL number increases with the number of iterations starting from a value of 5 up to a maximum of 200.

$\theta_1 = 25°$						
θ_2	$\dot{m}$	Ve [m/s]	Pe [Pa]	F [mN]	C_F	Me (mass)
0°	2.227E-05	733.983	148.442	16.713	1.7024	5.771
4°	2.228E-05	733.188	145.161	16.688	1.6999	5.817
8°	2.218E-05	730.576	142.175	16.550	1.6857	5.857
12°	2.230E-05	726.105	142.593	16.521	1.6828	5.863
15°	2.227E-05	722.130	142.600	16.428	1.6734	5.870
$\theta_1 = 30°$						
θ_2	$\dot{m}$	Ve [m/s]	Pe [Pa]	F [mN]	C_F	Me (mass)
0°	2.229E-05	728.151	163.105	16.628	1.6937	5.592
4°	2.230E-05	727.527	157.074	16.606	1.6915	5.672
8°	2.230E-05	724.497	152.953	16.530	1.6837	5.732
12°	2.230E-05	719.688	150.541	16.414	1.6720	5.774
15°	2.230E-05	715.254	149.415	16.317	1.6621	5.798
$\theta_1 = 35°$						
θ_2	$\dot{m}$	Ve [m/s]	Pe [Pa]	F [mN]	C_F	Me (mass)
0°	2.231E-05	718.172	184.214	16.476	1.6782	5.359
4°	2.231E-05	717.698	174.399	16.442	1.6748	5.482
8°	2.231E-05	714.575	166.864	16.348	1.6652	5.581
12°	2.231E-05	709.476	161.842	16.229	1.6531	5.656
15°	2.231E-05	704.698	159.121	16.115	1.6414	5.700
$\theta_1 = 40°$						
θ_2	$\dot{m}$	Ve [m/s]	Pe [Pa]	F [mN]	C_F	Me (mass)
0°	2.233E-05	705.283	208.397	16.257	1.6559	5.146
4°	2.232E-05	705.129	193.987	16.218	1.6519	5.306
8°	2.233E-05	702.051	182.808	16.124	1.6423	5.438
12°	2.234E-05	696.847	174.444	15.995	1.6292	5.542
15°	2.234E-05	691.848	169.928	15.870	1.6165	5.604
$\theta_1 = 45°$						
θ_2	$\dot{m}$	Ve [m/s]	Pe [Pa]	F [mN]	C_F	Me (mass)
0°	2.234E-05	690.695	233.137	16.006	1.6303	4.984
4°	2.235E-05	690.750	215.191	15.965	1.6262	5.148
8°	2.235E-05	687.850	199.933	15.865	1.6160	5.302
12°	2.235E-05	682.059	188.607	15.709	1.6001	5.426
15°	2.235E-05	677.517	182.320	15.592	1.5882	5.501

Table 4.2: Low Fidelity simulations results for different combinations of θ_1 and θ_2.

By performing simulations for the Low Fidelity case, using the grid depicted in Figure 4.1, the results presented in Table 4.2 are obtained. This table shows the values of mass flow rate, exit velocity, pressure at the nozzle's outlet section, generated thrust, thrust coefficient and the mass-averaged exit Mach number, which are obtained by solving the Euler equations, thereby omitting viscosity from the model. Consequently, only the effects of velocity divergence losses can be observed: as θ_2 increases, the angle of inclination of the velocity vector at the exit section becomes larger, leading to an increase in the radial component and a decrease in the axial one V_e, on which thrust depends. Consequently, a reduction in the thrust coefficient is observed for constant θ_1 as θ_2 increases. Regarding the effect of the angle θ_1, the smaller its value, the smaller the expansion angle, resulting in a more gradual expansion. This allows for lower exit pressure p_e values since the nozzle is longer and the flow has ample space to expand. Conversely, the larger θ_1 is, the shorter the nozzle will be, giving the flow less space to expand before reaching the exit section, thus resulting in higher p_e values. Furthermore, as θ_1 increases, the expansion angle of the flow also increases, leading to a greater inclination of the velocity vector and a loss in terms of axial acceleration, leading to a lower value of thrust coefficient C_F. For a fixed θ_1, it is noted that the exit pressure decreases with increasing θ_2, indicating that the flow expands more due to the greater pressure gradient between the nozzle's inlet and outlet sections.
As for the mass flow rate through the nozzle, it increases with higher θ_1 values. A shorter nozzle results in a higher pressure gradient, drawing more mass flow and the greater expansion angle tends to accelerate the gas towards the walls, reducing the axial component but increasing the radial component, thus contributing to the increased mass flow. For a fixed θ_1, as θ_2 increases, the mass flow rate also increases due to the aforementioned pressure gradient.
The thrust coefficient values reported in Table 4.2 are shown in Figure 4.6 for the different θ_1 and θ_2 combinations.

A representation of the flow field obtained by solving the Euler equations can be observed, for example, through the contour plots of Mach number and static pressure within the nozzle, depicted in Figures 4.7 and 4.8. Notably, the absence of the boundary layer and any viscous phenomena is evident. Moreover, the use of a second-order scheme for flow computation reveals an internal shock originating from the throat region, curving towards the nozzle's symmetry axis.

In Chapter 3, the flow fields in terms of Mach number, velocity magnitude, temperature and static pressure within the nozzle were provided, obtained by solving the Navier-Stokes equations under the assumption of laminar flow, constituting the High Fidelity model introduced in this chapter. Thus, considering Figures 3.7 and 3.9, a comparison can be made with the flow field obtained through Euler's equations: even though the contour plot shown in Figure 3.7 corresponds to the

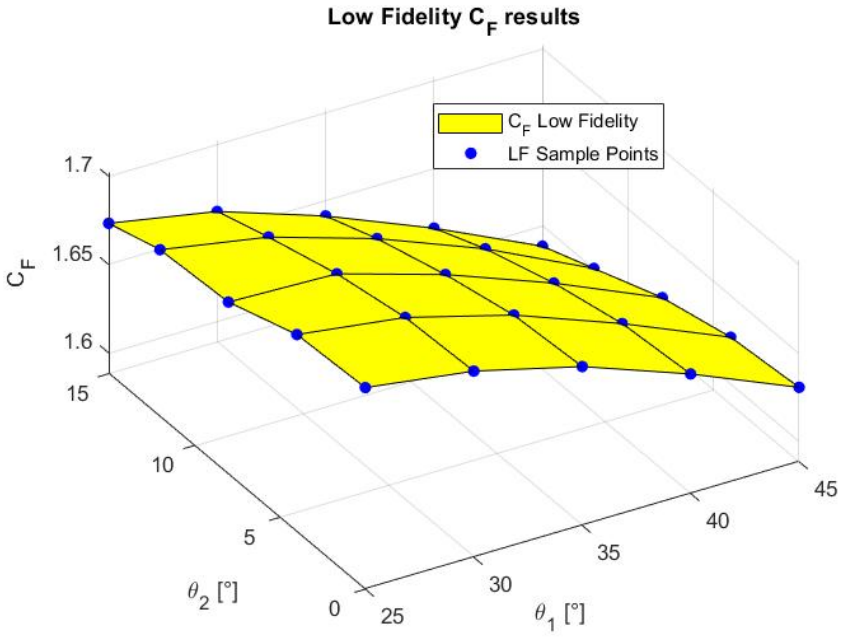

Figure 4.6: Low Fidelity C_F values calculated by solving Euler's equations

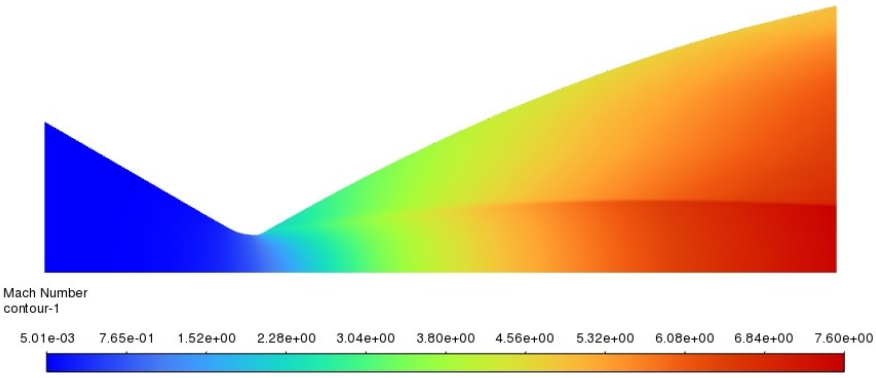

Figure 4.7: Mach number contour plot for $\theta_1 = 30°$ and $\theta_2 = 12°$ obtained by solving Euler's equations (Low Fidelity model)

combination $\theta_1 = 30°$ and $\theta_2 = 10°$, it is possible to observe the large portion occupied by the boundary layer, while the internal shock is always present but significantly less pronounced due to viscosity. Moreover, since the point $\theta_1 = 30°$, $\theta_2 = 10°$ is not included in the grid of candidate points for addition to the High Fidelity database, the results obtained in Chapter 3 can serve to validate the surrogate model, thereby assessing the error between the Co-Kriging prediction

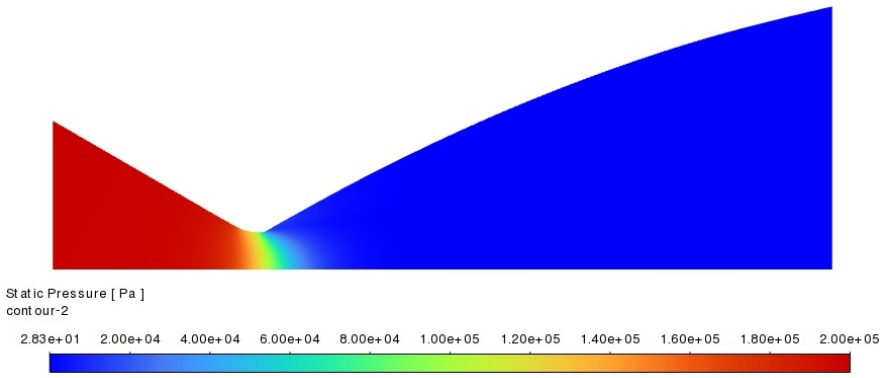

Figure 4.8: Static pressure contour plot for $\theta_1 = 30°$ and $\theta_2 = 12°$ obtained by solving Euler's equations (Low Fidelity model)

and the performed simulation.

4.2 Co-Kriging surrogate model

Thus, the implementation of the Co-Kriging surrogate model can be performed: by choosing firstly the Maximum Expected Improvement algorithm and, subsequently, the Information Gain algorithm as the criterion for finding new High Fidelity points to sample, the most promising point can be identified at each iteration and an expensive simulation can then be launched to evaluate the thrust coefficient given the θ_1 and θ_2 values found. For the initial High Fidelity dataset, the four points at the extremes of the design space are selected and the Navier-Stokes equations are solved for the flow within the nozzle, with the geometry determined by the considered angle combinations. The results of the first four expensive simulations are provided below: Starting from the data presented in Table 4.3, it is possible to

$\theta_1 = 25°$						
θ_2	$\dot{m}$	Ve [m/s]	Pe [Pa]	F [mN]	C_F	Me (mass)
0°	2.180E-05	683.571	311.507	15.665	1.5957	4.419
15°	2.165E-05	693.313	259.781	15.651	1.5942	4.753
$\theta_1 = 45°$						
θ_2	$\dot{m}$	Ve [m/s]	Pe [Pa]	F [mN]	C_F	Me (mass)
0°	2.168-05	655.992	354.438	15.090	1.5371	4.427
15°	2.166-05	654.659	280.503	14.873	1.5149	4.846

Table 4.3: Initial High Fidelity simulations results

construct an initial Co-Kriging surrogate model on which the Maximum Expected

Improvement algorithm will be applied, as described in Chapter 3. Since the High Fidelity model's response is unknown at the various points in the design space, except for the few that constitute the expensive dataset, it is necessary to acquire some thrust coefficient values using the High Fidelity model for validating the surrogate, i.e., for calculating the relative error between the High Fidelity objective function and the Co-Kriging prediction. A crucial aspect to consider is the total computational cost, which includes populating the Low Fidelity dataset, generating the expensive dataset and performing validation. Both stages require CFD simulations. Specifically, 25 evaluations of the thrust coefficient must be performed by solving Euler's equations for the cheap case, which will incur a non-negligible computational burden. Consequently, a significant portion of the initial budget will be consumed by the Low Fidelity simulations. For the cold gas thruster case, solving Euler's equations in a single simulation requires on average between 550 and 750 steps (depending on the geometry analyzed) and performing 25 such simulations entails a substantial cost. In contrast, solving the Navier-Stokes equations requires many more steps, averaging over 2500, so with just 5-6 expensive simulations, the cost would exceed that needed to fully populate the Low Fidelity dataset. For the implementation of the model, a computational budget of 9 points will be considered in total for the High Fidelity CFD simulations, allocated as 6 points for generating the Co-Kriging surrogate and 3 points for its validation.
For the validation of the model, one point is already known, specifically at $\theta_1 = 30°$ and $\theta_2 = 10°$. The other two points can be selected to maximize coverage of the design space and study the error more comprehensively. For this purpose, the points $\theta_1 = 35°$, $\theta_2 = 8°$ and $\theta_1 = 40°$, $\theta_2 = 4°$ can be considered. The results of the High Fidelity simulations for the specified validation points are presented in Table 4.4. Two iterations of the maximum Expected Improvement algorithm

$\theta_1 = 30°$						
θ_2	$\dot{m}$	Ve [m/s]	Pe [Pa]	F [mN]	C_F	Me (mass)
10°	2.167E-05	690.224	269.423	15.619	1.5909	4.739
$\theta_1 = 35°$						
θ_2	$\dot{m}$	Ve [m/s]	Pe [Pa]	F [mN]	C_F	Me (mass)
8°	2.162-05	683.167	291.605	15.487	1.5775	4.690
$\theta_1 = 40°$						
θ_2	$\dot{m}$	Ve [m/s]	Pe [Pa]	F [mN]	C_F	Me (mass)
4°	2.164-05	671.187	319.427	15.311	1.5596	4.574

Table 4.4: High Fidelity simulations results for validation points

are thus performed, evaluating the relative percentage error at each step in the validation points. Since the introduction of viscous phenomena is not expected to significantly alter the overall trend of the thrust coefficient within the design

space, i.e., it should not cause abrupt changes from one point to another, we can be confident that an accurate surrogate model can be obtained with only a few points. The Co-Kriging prediction for the various iterations is shown in Figure 4.9, along with the distribution of the Expected Improvement parameter for the different candidate points, belonging to the Low Fidelity dataset. The resulting surrogate model when the computational budget is exhausted is presented in Figure 4.10, and the error at the validation points is shown in Table 4.5.

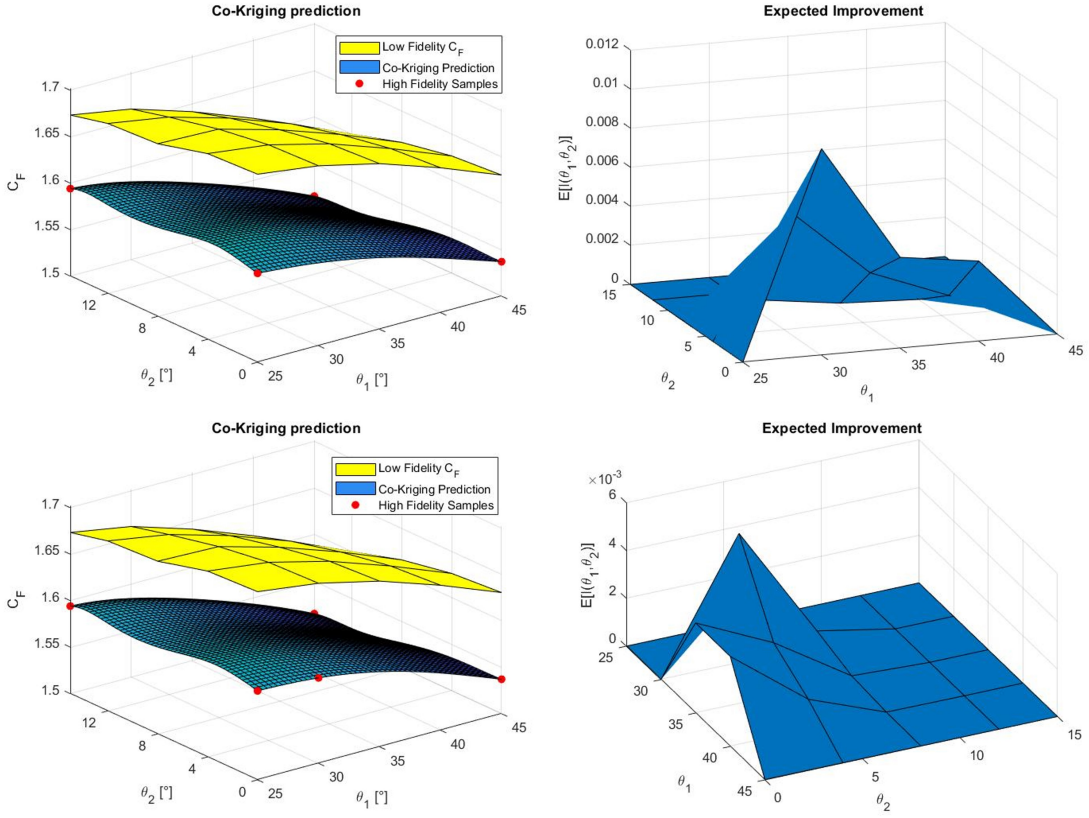

Figure 4.9: Co-Kriging prediction at each step of the Maximum Expected Improvement algorithm

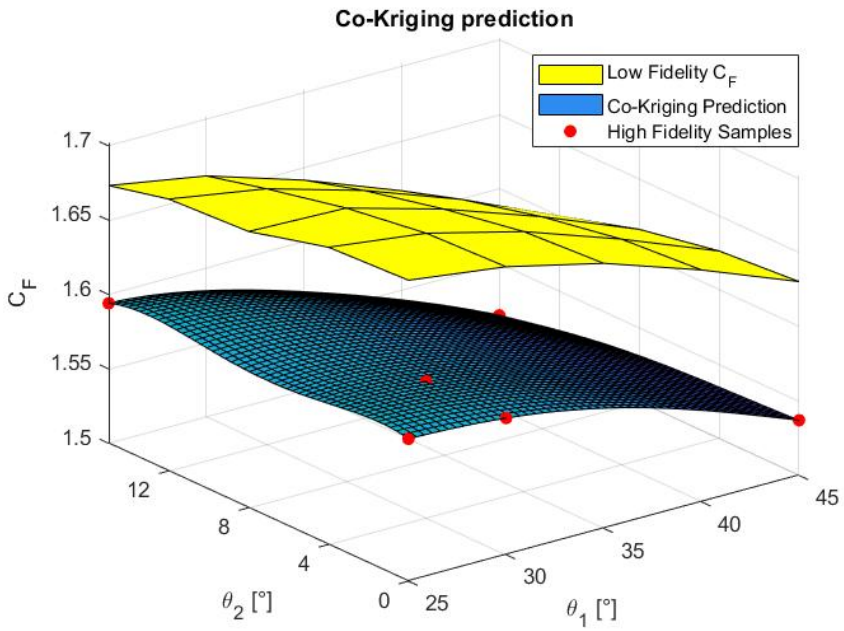

Figure 4.10: Final Co-Kriging surrogate model - Expected Improvement

$\theta_1 = 30°$			
θ_2	**Real** C_F	**Surrogate's** C_F	**Relative Error (%)**
10°	1.5909	1.5861	0.3017
$\theta_1 = 35°$			
θ_2	**Real** C_F	**Surrogate's** C_F	**Relative Error (%)**
8°	1.5775	1.5732	0.2726
$\theta_1 = 40°$			
θ_2	**Real** C_F	**Surrogate's** C_F	**Relative Error (%)**
4°	1.5596	1.5598	0.0128

Table 4.5: Error evaluation at the validation points - First Iteration

$\theta_1 = 30°$			
θ_2	**Real** C_F	**Surrogate's** C_F	**Relative Error (%)**
10°	1.5909	1.5909	0.0029
$\theta_1 = 35°$			
θ_2	**Real** C_F	**Surrogate's** C_F	**Relative Error (%)**
8°	1.5775	1.5763	0.0761
$\theta_1 = 40°$			
θ_2	**Real** C_F	**Surrogate's** C_F	**Relative Error (%)**
4°	1.5596	1.5586	0.0641

Table 4.6: Error evaluation at the validation points - Second Iteration

As evident from the Tables 4.5 and 4.6, the approximation error of the surrogate model relative to the objective function, which is unknown except at the nine points constituting the final High Fidelity Design of Experiment and the validation set, is extremely small. Thus, a highly satisfactory Co-Kriging model has been obtained with only a few points, requiring a very limited number of expensive simulations. It is also noteworthy that the resulting High Fidelity DOE does not exhibit space-filling properties, as it was observed in Chapter 2 when applying the maximum Expected Improvement algorithm to two-variable test functions. Nevertheless, this does not preclude an almost perfect approximation of the High Fidelity model's response. With the final surrogate model now available, we can transition from exploring the design space to exploiting the model itself. In particular, we can observe the effect of viscosity on the thrust coefficient values: in the Low Fidelity case, with a fixed angle θ_1, the C_F exhibits a strictly increasing trend as θ_2 decreases. However, in the High Fidelity model, the thrust coefficient becomes almost flattened, varying much less with a decrease in θ_2. This is because the effect of viscosity in the High Fidelity model counteracts the beneficial effect of small exit angles on velocity divergence losses. The greater length characterizing the nozzle results in more boundary layer development, increasing the losses. The compensation between the effect of velocity divergence and the effect of viscosity thus results in a lower rate of C_F variation for a fixed θ_1.

Now, the second Low Fidelity model is explored. As previously mentioned, the objective is to investigate the impact of using the ideal 1D nozzle model introduced in Chapter 3 as the Low Fidelity model. In this case, the thrust coefficient values are derived from Equation 3.15, eliminating the need for CFD simulations to populate the cheap dataset. Consequently, the entire computational budget can be allocated to High Fidelity simulations, allowing for a greater number of points to be sampled in the design space. Furthermore, the C_F predicted by the Low Fidelity model will be constant, as it is independent of the values of θ_1 and θ_2. Therefore, a higher error is expected with the same number of expensive points selected, given that no sources of losses are considered, significantly deviating from the real case. The cheap Design of Experiment now refers to Figure 4.2. The goal is to evaluate the approximation error relative to the previously generated surrogate model, thus, the two results will be compared.
The Expected Improvement algorithm will thus be applied to this case, using the Low Fidelity 100 points DOE and the ideal model. Iterations will be halted when the Co-Kriging surrogate closely approximates the one obtained using the non-ideal model. To avoid conducting new High Fidelity CFD simulations, since the candidate points are now much more numerous and different from those previously selected, we will consider the predictions of the surrogate in Figure 4.10 as the exact values of the expensive model's response, given that we have demonstrated

the approximation error is practically negligible. The new High Fidelity points will be obtained when the average relative error between the newly constructed surrogate and the one shown in Figure 4.10 falls below 0.2%.

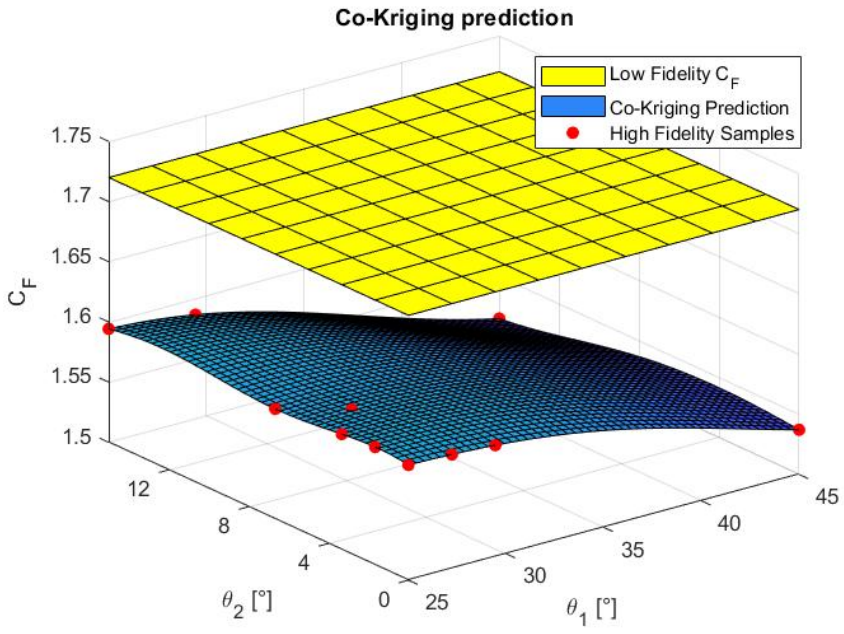

Figure 4.11: Final Co-Kriging surrogate model leveraging the 1D nozzle model as Low Fidelity model - Expected Improvement

As one can observe in Figure 4.11, achieving a Co-Kriging surrogate similar to that obtained from the application of the maximum Expected Improvement algorithm shown in Figure 4.10 now requires 11 High Fidelity points, meaning 11 costly CFD simulations must be performed. The potential of the ideal 1D model is thus evident, as it eliminates the need for the 25 Low Fidelity simulations, which still require a certain computational cost, by leveraging results derived from solving a simple equation, which incurs no computational burden. In the first case, 25 cheap CFD simulations and 6 expensive ones were necessary (not counting the 3 required for model validation). Now, with a Low Fidelity dataset potentially as large as desired, since the cost to generate it is zero, only 11 High Fidelity CFD simulations are needed. As mentioned earlier, if the steps required to solve Euler's equations are approximately 550 and those for the Navier-Stokes equations are about 2500, it becomes clear that using the ideal 1D model for Low Fidelity data significantly reduces the computational burden compared to the first case. Given the same total iterations needed to populate the two databases, this approach also allows for

approximately 3 additional expensive simulations to be used for surrogate validation.

As previously mentioned, it may be useful to evaluate the performance of the Maximum Information Gain algorithm when applied to the case of the cold gas thruster. As discussed in Chapter 2, this algorithm tends to deliver the best performance for single-variable functions, while for two-variable functions it performs on par with the Maximum Expected Improvement (EI) algorithm. However, in certain cases, it has shown exceptional performance, as demonstrated with the first test function analyzed. Although it is difficult to predict in advance whether IG will outperform or match EI, it is still prudent to explore its application. One of the major drawbacks of the IG algorithm is that it constructs as many surrogate models as there are candidate points for addition to the High Fidelity database in each iteration, resulting in a significant slowdown in operations and a higher computational cost. In scenarios where there are few Low Fidelity points, such as the 25 for the cold gas thruster, the computational burden of using the information-based criterion is not prohibitive. Thus, we proceed with generating the Co-Kriging surrogate model for the thrust coefficient: starting with the 4 High Fidelity points presented in Table 4.3, in each iteration the point that maximizes the IG(x_1, x_2) index will be identified. Regarding the stopping criterion for the cycle, if a computational budget of 6 High Fidelity points in total for training the model was established for Expected Improvement, this maximum threshold can also be applied here to study the error at validation points. If at the first iteration the average error is lower than that found at the last step of Expected Improvement, the search can be halted and the Co-Kriging surrogate can be constructed with only 5 points. In this case, the IG algorithm would perform better, reducing the required points and the approximation error.

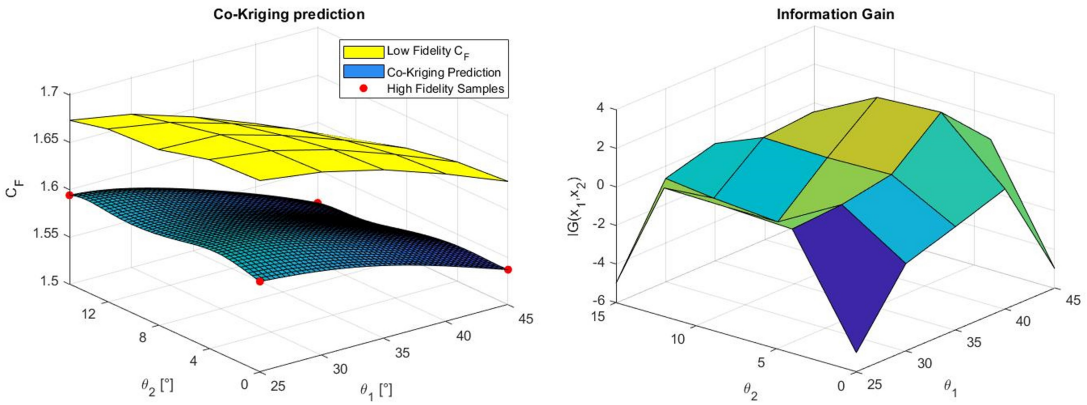

Figure 4.12: Co-Kriging prediction at each step of the Maximum Information Gain algorithm

The first step identifies the new point to be added as the one characterized by θ_1 = 40° and θ_2 = 4°, which belongs to the set of points previously chosen for the validation of the surrogate model. To evaluate the approximation error in this case, one can utilize one of the High Fidelity points identified by the Expected Improvement for which a High Fidelity CFD simulation was conducted to calculate C_F. For instance, the validation set will consist of the points (θ_1,θ_2) = [(30°,10°), (35°,8°), (30°,0°)]. Alternatively, an additional CFD simulation could be performed to make the validation set more uniform in space, but to limit costs, we consider a previously evaluated point. The relative percentage error at the three chosen positions is reported in the following table:

$\theta_1 = 30°$			
θ_2	**Real** C_F	**Surrogate's** C_F	**Relative Error (%)**
10°	1.5909	1.5909	-6.7e-4
$\theta_1 = 35°$			
θ_2	**Real** C_F	**Surrogate's** C_F	**Relative Error (%)**
8°	1.5775	1.5770	0.0368
$\theta_1 = 30°$			
θ_2	**Real** C_F	**Surrogate's** C_F	**Relative Error (%)**
0°	1.5918	1.5902	0.100

Table 4.7: Error evaluation at the validation points - First Iteration - Information Gain

After the first iteration, an average relative percentage error of 0.0458% is achieved, compared to the 0.0477% error found in the last iteration of the Maximum Expected Improvement algorithm. Thus, the cycle can be halted and the surrogate model, in this case, will be constructed with only 5 High Fidelity points, as shown in Figure 4.13.
It can be concluded that the Maximum Information Gain criterion, in the case of the thrust coefficient of the cold gas thruster, performs better than the Expected Improvement criterion. This is evident both in terms of the number of High Fidelity points required for the construction of the surrogate model and in terms of the final average approximation error obtained.
If, as previously done, one wishes to evaluate the effect of using the ideal 1D model as the Low Fidelity dataset, this time applying the information-based algorithm, it is important to note that the number of candidate points becomes quite high. Therefore, a potential gain in reducing the number of points needed to populate the expensive dataset could still entail a significantly higher computational burden for identifying the point to be sampled. Considering that this cost is much lower in terms of time compared to solving the Navier-Stokes equations, the process can proceed without significant limitations. The addition of points is halted when the

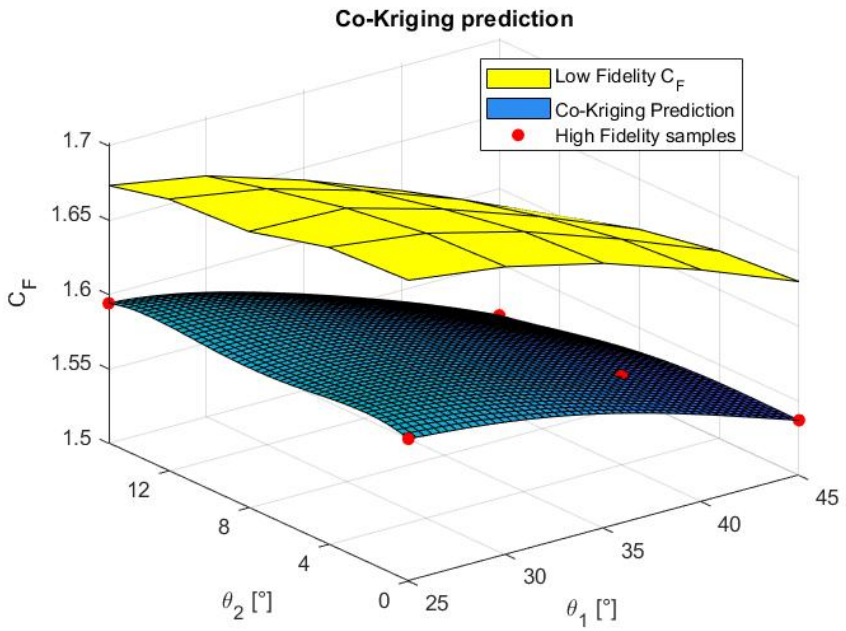

Figure 4.13: Final Co-Kriging surrogate model - Information Gain

approximation error relative to the surrogate model shown in Figure 4.13 falls below 0.2%. The result is shown in Figure 4.14.

By leveraging the ideal 1D nozzle model as the basis for evaluating the thrust coefficient for the High Fidelity model, only 7 points are now required to achieve an approximation similar to that provided by the 25 Low Fidelity CFD simulations. This results in a significant reduction in computational costs, as only 7 expensive simulations and the solution of a simple linear equation are now required, making the generation of the cheap dataset instantaneous. Compared to the Expected Improvement criterion, far fewer points are now needed, confirming the excellent performance of the Information Gain criterion for this case. The only drawback is that the surrogate constructed in this manner, while approximating the objective function very well across almost the entire design space, exhibits less accuracy for θ_1 values around 25°. The surrogate shown in Figure 4.14 appears practically flat and fails to capture the valley that characterized the previous models around $\theta_1 = 25°$ and $\theta_2 = 8°$. This occurs because the sampled points are distant from this region and the Low Fidelity model, being constant everywhere, does not provide significant information for that zone, unlike the Low Fidelity model derived from solving the Euler equations. This issue can lead to less accurate results for angle combinations near the mentioned values; otherwise, the approximation is excellent.

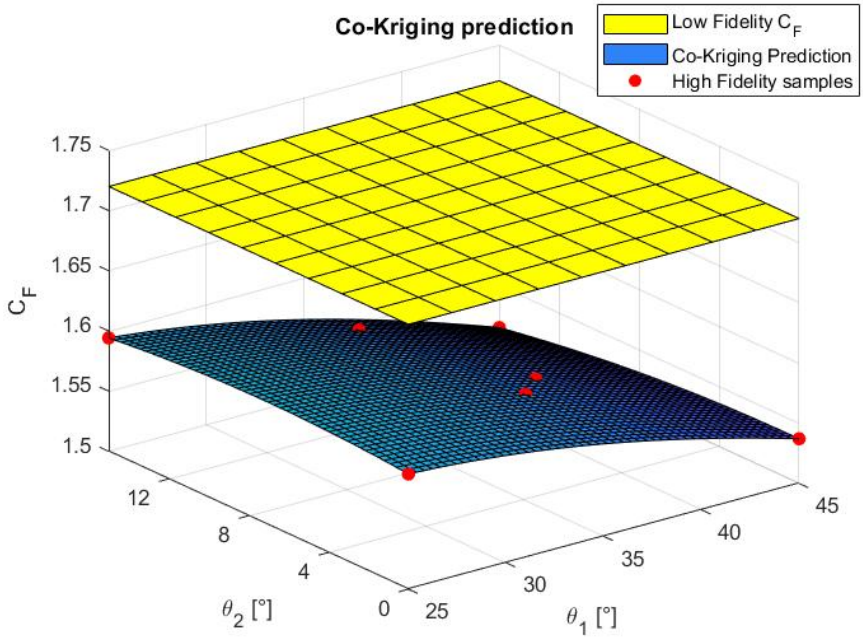

Figure 4.14: Final Co-Kriging surrogate model leveraging the 1D nozzle model as Low Fidelity model - Information Gain

4.3 Co-Kriging based multi-objective optimization

Once a surrogate model that accurately approximates the real behaviour of the thrust coefficient in the design space has been obtained and its accuracy evaluated for various combinations of θ_1 and θ_2, the information provided by this model can be exploited to predict C_F values for untested angle combinations. In this case study, the objective is not to identify the maximum thrust coefficient, but rather to solve a multi-objective optimization problem. As extensively discussed, a fundamental aspect concerns not only thrust generation but also the constraints related to the weight of the propulsion system comprising the cold gas thruster. If the weights of the propellant tank, feed lines, corresponding valves and the solenoid valve preceding the nozzle are considered fixed, then the weight of the nozzle becomes the only free parameter that needs to be minimized. Therefore, the goal is twofold: to achieve the maximum possible thrust coefficient and to minimize the nozzle's mass.
Regarding the nozzle mass, given a fixed material, it will depend exclusively on the geometry, specifically on θ_1 and θ_2, which are the sole variables in the geometric configuration. Given a combination of angles and assuming a constant thickness

for the nozzle's shell, the mass calculation is practically instantaneous: essentially, it involves evaluating the difference between the volume calculated at radius R(x) plus the constant thickness and the internal volume, evaluated at radius R(x). Knowing the curve that describes the nozzle's contour as shown in Figure 3.5 (in 2D), the volume of the solid of rotation around the x-axis can be derived using the following formula:

$$V = \pi\rho \int_a^b \left[(R(x) + t)^2 - R(x)^2\right] dx \tag{4.10}$$

where ρ is the material density [kg/m^3], t is the thickness and R(x) is the radii distribution along the nozzle's length.
Typically, stainless steel is used as the primary material for constructing cold gas microthrusters, with a density of approximately 7500 kg/m^3. For the thickness t, a constant value of $3 \cdot R_t$ is chosen, where R_t is the throat radius.

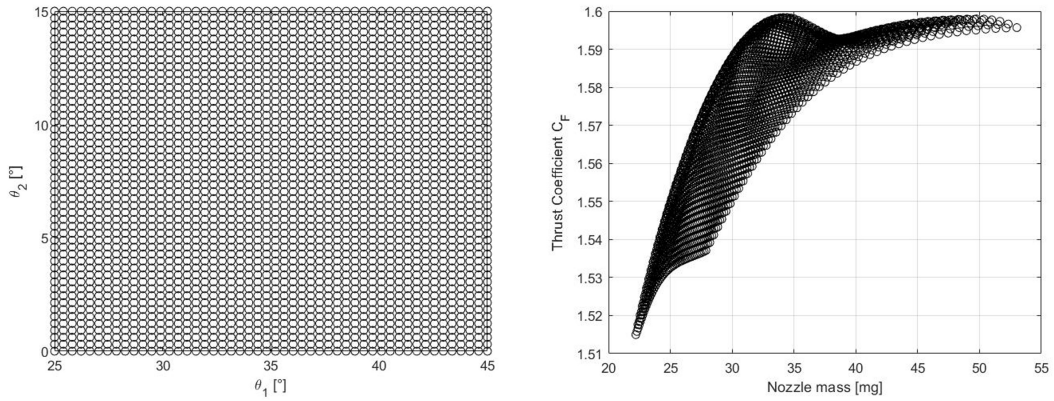

Figure 4.15: Points in the design space (left) used to evaluate C_F and nozzle's mass (right)

Thus, with the Co-Kriging surrogate model now available (for the optimization, the one obtained with the Expected Improvement algorithm is leveraged, as represented in Figure 4.10), it is possible to predict the value of C_F at a large number of points in the design space instantaneously. For instance, the surrogate model can be evaluated on a 50x50 grid in the θ_1-θ_2 plane (Figure 4.15), consisting of 2500 points, thereby obtaining the thrust coefficient for numerous geometric configurations of the nozzle. Simultaneously, the nozzle's mass can be evaluated for the same combinations of θ_1 and θ_2, a process that is also instantaneous using the numerical trapezoidal method for the integral calculation in Equation 4.10, given the known coordinates of the nozzle contour as a function of the angles. This results in two databases, one for the thrust coefficient and one for the mass, necessary to initiate the multi-objective optimization. Clearly, higher mass values will characterize

longer nozzles, i.e., those with smaller values of θ_1 and θ_2, while higher thrust coefficients will be observed for $\theta_1 = 25°$. Maximizing C_F would therefore lead to selecting very long and heavy nozzles, which is obviously impractical and undesirable, as the aim is to minimize the mass. Therefore, we have two conflicting objectives: a gain in thrust coefficient corresponds to a drawback in terms of mass, hence, rather than identifying a single optimal solution, it is preferable to determine a set of solutions that can subsequently facilitate a trade-off between the two objectives. Through multi-objective optimization, the aim is to identify the Pareto optimal set, which comprises solutions that are not dominated by any other solutions in the design space: these solutions cannot be improved in terms of one objective without worsening at least one of the others [64]. The goal of multi-objective optimization is therefore to identify this set of solutions, which translates into the so-called Pareto front in the objective function space. This allows for the evaluation of the advantages and disadvantages of each solution on the front and the execution of a trade-off, as moving from one optimal value to another always involves a gain in one objective at the expense of one or more others [64].

Therefore, for the analyzed cold gas thruster, the objective is to derive the set of optimal solutions and the Pareto front in the mass-thrust coefficient plane. To achieve this, a genetic algorithm (GA) for multi-objective optimization can be employed, which in this case considers only two objectives. Genetic algorithms are well-suited for multi-objective optimization problems due to their population-based nature [64]. Essentially, the algorithm utilizes a set of initial solutions, which constitute the initial population, and with each iteration, it explores different regions of the design space, generating new solutions from the existing ones until the algorithm converges, finding the set of non-dominated solutions that make up the Pareto optimal set. Various types of evolutionary algorithms can be used for optimization with GA, such as MOGA, WBGA, NSGA, SPEA, etc [64]. For the case of the cold gas thruster, the genetic algorithm implemented in Matlab's *gamultiobj.m* function is employed: particularly, it requires specifying the objective functions and their number, in this case, $-C_F$ (since the algorithm minimizes the objectives while we aim to maximize the thrust coefficient) and the nozzle mass, as well as the lower and upper bounds for the variables θ_1 and θ_2 (lb = [25; 0] and ub = [45; 15]) and the size of the initial population, set to 200.
The algorithms reaches convergence leading to a 77-points optima set and the resulting Pareto fornt is represented in Figure 4.16.

As can be observed from Figure 4.13, the optimal solutions generate a continuous and well-distributed Pareto front, providing a broad range of values for trade-offs. Specifically, the front clusters along the outer contour of the curve representing the thrust coefficient as a function of mass, spanning uniformly from very low to high C_F values while narrowing the range of solutions in terms of mass. A visualization

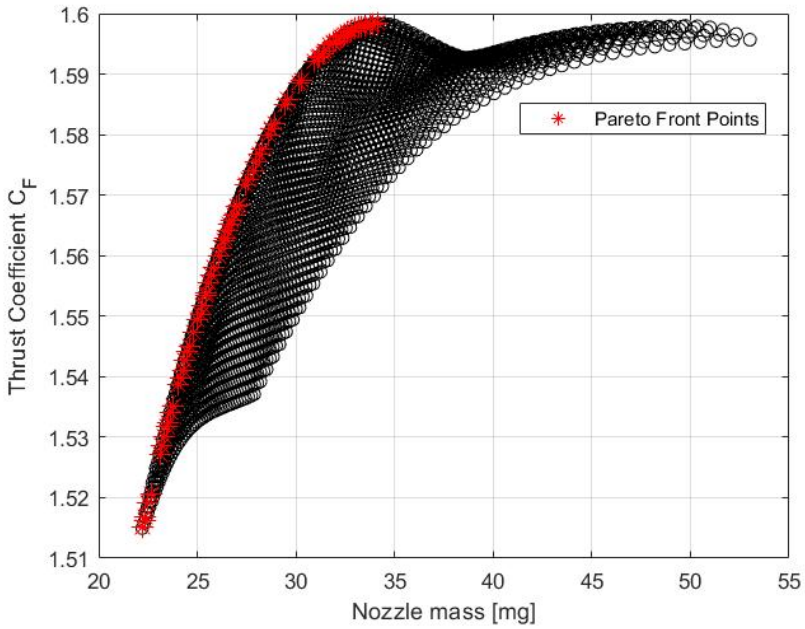

Figure 4.16: Pareto Front representing the optima solutions of the multi-objective optimization

of the positions of the optimal points in the design space of the geometric variables θ_1 and θ_2 is provided in Figure 4.17. This representation is highly beneficial in understanding which combinations of the two angles yield optimal values. As seen from the Pareto front representation, the optimal points span the entire, or at least the majority, of the thrust coefficient range, indicating the possibility of having optimal configurations with either very short and light nozzles (i.e., θ_1 close to 45°) or extremely long and heavy nozzles (i.e., θ_1 close to 25°). Indeed, observing Figure 4.17 reveals a uniform and nearly continuous distribution in the θ_1 direction, while θ_2 values are confined to a small range above approximately 12°. For a given θ_1 angle, having a higher second angle allows for a reduction in the nozzle length, and thus its weight, without significantly reducing C_F, since boundary layer losses are minimized. Furthermore, it has been observed that divergence and boundary layer losses exhibit contrasting behaviours with varying nozzle length, therefore, this trade-off between losses is now compounded by the trade-off related to the thruster's mass. The Pareto front and the respective points in the design space will thus enable the selection of the configuration that best meets the mission requirements and constraints during the design phase. For instance, if priority is given to reducing the weight of the propulsion system, one will tend to select optimal points on the descending segment of the front in Figure 4.16, closer to the

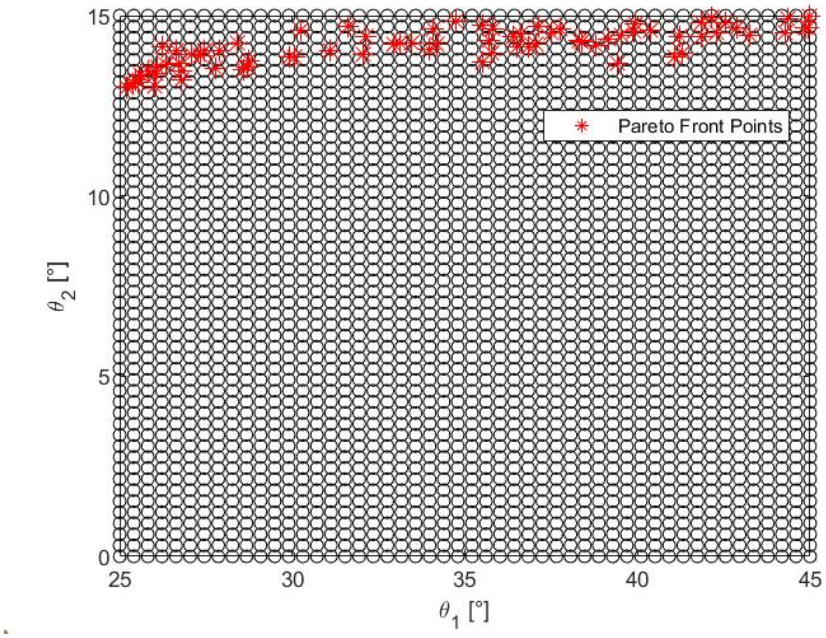

Figure 4.17: Pareto optima points in the design space

lower extreme, accepting lower thrust performance. Conversely, if the priority is on maximizing C_F, points on the right segment of the front will be chosen, accepting larger masses. Such an analysis is crucial, as the nozzle geometry introduced by Rao was studied under conditions of high Reynolds numbers, whereas, in the case of cold gas microthrusters, one works with much smaller values of this control parameter. Consequently, the boundary layer thickness within the nozzle becomes a fundamental and, above all, dimensioning aspect to consider. It will then be up to the designer to choose the most suitable values for the angles θ_1 and θ_2 in light of the obtained results.

Chapter 5

Conclusions and future developments

In conclusion, this study investigated the performance and optimization of the exhaust nozzle of a cold gas thruster using the multi-fidelity Co-Kriging technique, to develop a surrogate model that predicts the thrust coefficient values for various combinations of the angles θ_1 and θ_2, which describe the nozzle's parabolic section. First, the study focused on the theory behind surrogate models, particularly on methodologies to generate an accurate model that minimizes the number of simulations or observations of the High Fidelity model response, identifying adaptive sampling techniques as an excellent solution to achieve this goal. Specifically, the effectiveness of these methods was observed by implementing three different algorithms: two widely used in literature, Maximum Variance and Maximum Expected Improvement, and an information-based criterion based on the variation of information entropy, referred to as the Maximum Information Gain algorithm. From applications on single-variable functions, the Information Gain proved to be highly promising, delivering excellent performance at the cost of higher computational expense in terms of the algorithm, whereas Maximum Variance and Expected Improvement consistently provided good performance but might require a larger number of High Fidelity points. Overall, it is challenging to assert with certainty which of the three algorithms would yield the best result a priori, as the outcome depends on the characteristics of the High and Low Fidelity functions. The same applies to the two-variable functions studied, particularly noting that for the Himmelblau function, the Maximum Information Gain algorithm performed significantly better than the other two, while for the subsequent test functions, the results were practically equivalent to those obtained from the Expected Improvement. Many other algorithms have been implemented in the literature, but an ongoing challenge is finding a criterion that, based on the Low Fidelity model

response, which can be evaluated over much of the design space at low cost, and basic knowledge of the physics underlying the High Fidelity model, can provide an a priori idea of the minimum threshold of expensive points needed for a good approximation and in which areas of the domain these points should be located. To date, adaptive sampling strategies have proven to be superior to one-shot strategies. However, the practicality of the latter, namely knowing in advance the number of simulations to be performed and for which points, remains a desirable goal.

Regarding cold gas thrusters, a performance study was conducted based on a fixed nozzle geometry, evaluating the expansion of the cold gas through a CFD simulation to assess the velocity, Mach number, pressure and temperature fields within the nozzle. It was demonstrated that viscosity plays a crucial role, given that the Reynolds number is on the order of 10^3, implying the development of the boundary layer not only along the divergent section, but also near the throat section of the nozzle. Due to the very small size of these propulsion systems, it was observed that the boundary layer occupies approximately 20% of the thruster's exit section, thereby reducing the effective expansion ratio and consequently the thrust. By comparing the resolution of the Navier-Stokes equations considering laminar flow with RANS equations using various turbulence models, it was shown that the flow inside the nozzle is laminar since the results accounting for turbulence differ by less than 1% from those without it. Furthermore, by varying the nozzle geometry, it was possible to compare the effects of divergence losses and viscous losses as a function of the length of the divergent section. Specifically, longer nozzles, characterized by low θ_1 values, reduce misalignment losses but lead to greater boundary layer development, thus increasing viscous effects. These two contrasting effects limit the attainable C_F values. Another important aspect is the nozzle's mass, a crucial parameter in the final multi-objective optimization.

For the generation of the Co-Kriging surrogate model for the cold gas thruster, High Fidelity and Low Fidelity datasets were created based on the real nozzle model studied by solving the Navier-Stokes equations and the real but non-viscous model studied with Euler's equations, respectively. Subsequently, the ideal 1D model was used to generate a Low Fidelity database to explore the potential of using a highly simplified but instantaneously evaluable model. For the thrust coefficient, the Information Gain-based algorithm yields the best results, requiring only 5 High Fidelity points compared to a Low Fidelity DOE of 25 points, whereas the Expected Improvement method requires 6 points for a satisfactory Co-Kriging, still an excellent outcome. The use of the ideal 1D model increases the number of expensive CFD simulations needed but eliminates the need for cheap simulations to populate the Low Fidelity dataset, thus reducing the total computational burden for both adaptive algorithms. Even with the use of the 1D model, Information

Gain outperforms Expected Improvement.

Finally, the optimization of the nozzle geometry was performed using a genetic algorithm to identify the optimal non-dominated points, forming the Pareto front in the mass-thrust coefficient plane. The result is a set of optimal solutions ranging from the minimum to the maximum value for the θ_1 angle while significantly narrowing the optimal range of θ_2 to angles between 12° and 15°.

Bibliography

[1] Jun Zheng, Xinyu Shao, Liang Gao, Ping Jiang, and Haobo Qiu. «Difference mapping method using least square support vector regression for variable-fidelity metamodelling». In: *Engineering Optimization* 47.6 (2015), pp. 719–736. DOI: 10.1080/0305215X.2014.918114 (cit. on pp. 1, 3, 4, 21).

[2] Gary Wang and S. Shan. «Review of Metamodeling Techniques in Support of Engineering Design Optimization». In: *Journal of Mechanical Design - J MECH DESIGN* 129 (Apr. 2007). DOI: 10.1115/1.2429697 (cit. on p. 1).

[3] M. Giselle Fernández-Godino. «Review of multi-fidelity models». In: *Advances in Computational Science and Engineering* 1.4 (2023), pp. 351–400. ISSN: 2837-1739. DOI: 10.3934/acse.2023015. URL: http://dx.doi.org/10.3934/acse.2023015 (cit. on pp. 1–4, 29, 99).

[4] Daniele Peri and Emilio F. Campana. «High-Fidelity Models in Global Optimization». In: *Global Optimization and Constraint Satisfaction*. Ed. by Christophe Jermann, Arnold Neumaier, and Djamila Sam. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 112–126. ISBN: 978-3-540-32041-8 (cit. on p. 2).

[5] Alexander Forrester, Andras Sobester, and Andy Keane. *Engineering Design Via Surrogate Modelling: A Practical Guide*. July 2008. ISBN: 978-0-470-06068-1. DOI: 10.1002/9780470770801 (cit. on pp. 2, 6–8, 20–25, 36, 47, 48, 53).

[6] Seon Han Choi, Sun Lee, and Tag Gon Kim. «Multi-fidelity modeling simulation methodology for simulation speed up». In: May 2014. ISBN: 978-1-4503-2794-7. DOI: 10.1145/2601381.2601385 (cit. on p. 3).

[7] Vassili Toropov and Valery Markine. «The use of simplified numerical models as mid-range approximations». In: *6th Symposium on Multidisciplinary Analysis and Optimization*. DOI: 10.2514/6.1996-4088 (cit. on p. 3).

[8] G.E.P. Box and N.R. Draper. *Empirical Model-Building and Response Surfaces*. Wiley Series in Probability and Statistics - Applied Probability and Statistics Section Series. Wiley, 1987. ISBN: 9780471810339. URL: https://books.google.it/books?id=QO2dDRufJEAC (cit. on p. 4).

[9] J.P.C. Kleijnen. *Statistical tools for simulation practitioners*. English. Statistics 76. Pagination: XII, 429. Marcel Dekker, 1987. ISBN: 0824773330 (cit. on p. 4).

[10] Kai Tai Fang, Runze Li, and Agus Sudjianto. *Design and modeling for computer experiments*. English (US). Publisher Copyright: © 200. CRC Press, Oct. 2005. ISBN: 9780367578008 (cit. on p. 4).

[11] Guy Raoul Yondo Mine. «Efficient Design and Analysis of Multi-Fidelity Experiments for High Dimensional Aerodynamic Data Space». In: *UPM Thesis* (Nov. 2018). DOI: 10.20868/upm.thesis.52989 (cit. on pp. 4, 10).

[12] Naks Alex, P. Newman, Natalia Alexandrov, R. Lewis, C. Gumbert, and Lawrence Green. «Optimization with Variable-Fidelity Models Applied to Wing Design». In: 2000 (Mar. 2000). DOI: 10.2514/6.2000-841 (cit. on p. 4).

[13] Shawn E. Gano, John E. Renaud, and Brian Sanders. «Hybrid Variable Fidelity Optimization by Using a Kriging-Based Scaling Function». In: *AIAA Journal* 43.11 (2005), pp. 2422–2433. DOI: 10.2514/1.12466 (cit. on p. 4).

[14] Shawn E. Gano. «Simulation-Based Design Using Variable Fidelity Optimization». PhD diss. Notre Dame, Indiana: University of Notre Dame, 2006 (cit. on p. 4).

[15] Roberto Vitali, Raphael Haftka, and Bhavani Sankar. «Multi-fidelity design of stiffened composite panel with a crack». In: *Structural and Multidisciplinary Optimization* 23 (June 2002), pp. 347–356. DOI: 10.1007/s00158-002-0195-1 (cit. on p. 4).

[16] Seongim Choi, Juan Alonso, Ilan Kroo, and Mathias Wintzer. «Multifidelity Design Optimization of Low-Boom Supersonic Jets». In: *Journal of Aircraft - J AIRCRAFT* 45 (Jan. 2008), pp. 106–118. DOI: 10.2514/1.28948 (cit. on p. 4).

[17] Marlana Goldsmith, Bhavani Sankar, Raphael Haftka, and Robert Goldberg. «Effects of Microstructural Variability on the Mechanical Properties of Ceramic Matrix Composites». In: *Journal of Composite Materials* 49 (Jan. 2014). DOI: 10.1177/0021998313519151 (cit. on p. 4).

[18] Dehao Liu and Yan Wang. «Multi-Fidelity Physics-Constrained Neural Network and its Application in Materials Modeling». In: Aug. 2019. DOI: 10.1115/DETC2019-98115 (cit. on p. 4).

[19] Edmondo Minisci and Massimiliano Vasile. «Robust Design of a Reentry Unmanned Space Vehicle by Multifidelity Evolution Control». In: *AIAA Journal* 51 (June 2013), pp. 1284–1295. DOI: 10.2514/1.J051573 (cit. on p. 4).

[20] Marc Kennedy and A O'Hagan. «Predicting the Output from a Complex Computer Code When Fast Approximations Are Available». In: *Biometrika* 87 (Oct. 1998). DOI: 10.1093/biomet/87.1.1 (cit. on pp. 4, 10, 11).

[21] Loic Le Gratiet and Claire Cannamela. «Cokriging-Based Sequential Design Strategies Using Fast Cross-Validation Techniques for Multi-Fidelity Computer Codes». In: *Technometrics* 57 (Oct. 2012). DOI: 10.1080/00401706.2014.928233 (cit. on p. 4).

[22] Hyoung Seog Chung and Juan Alonso. «Design of a Low-Boom Supersonic Business Jet Using Cokriging Approximation Models». In: *9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*. DOI: 10.2514/6.2002-5598 (cit. on p. 4).

[23] Alexander I.J Forrester, András Sóbester, and Andy J Keane. «Multi-fidelity optimization via surrogate modelling». In: *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 463.2088 (2007), pp. 3251–3269. DOI: 10.1098/rspa.2007.1900 (cit. on pp. 4, 6, 7, 10–16, 21, 29, 31, 37).

[24] Zhong-Hua Han, Ralf Zimmermann, and Stefan Görtz. «A New Cokriging Method for Variable-Fidelity Surrogate Modeling of Aerodynamic Data». In: Jan. 2010. DOI: 10.2514/6.2010-1225 (cit. on p. 5).

[25] Ruochen Wang, Yanping Yang, Xiangyu Wang, Bo Wang, and Guoxin Zhang. «Co-Kriging Based Multi-Fidelity Aerodynamic Optimization for Flying Wing UAV with Multi-Shape Wingtip Design». In: *2021 IEEE International Conference on Unmanned Systems (ICUS)*. 2021, pp. 93–98. DOI: 10.1109/ICUS52573.2021.9641491 (cit. on p. 5).

[26] Xuhao Peng, Jiaqing Kou, and Weiwei Zhang. «Multi-fidelity nonlinear unsteady aerodynamic modeling and uncertainty estimation based on Hierarchical Kriging». In: *Applied Mathematical Modelling* 122 (2023), pp. 1–21. ISSN: 0307-904X. DOI: https://doi.org/10.1016/j.apm.2023.05.031 (cit. on p. 5).

[27] Andy J. Keane. «Cokriging for Robust Design Optimization». In: *AIAA Journal* 50.11 (2012), pp. 2351–2364. DOI: 10.2514/1.J051391 (cit. on p. 5).

[28] Donald Jones. «A Taxonomy of Global Optimization Methods Based on Response Surfaces». In: *J. of Global Optimization* 21 (Dec. 2001), pp. 345–383. DOI: 10.1023/A:1012771025575 (cit. on pp. 5–9, 27, 35).

[29] Lennart Verheij. «Multi-fidelity Method for Aerodynamic Optimisation of Axial Compressor Blades». Contributor: M. Pini (mentor), ORCID: 0000-0003-4538-3183. Master's thesis. Delft, Netherlands: Delft University of Technology, Nov. 2017 (cit. on p. 5).

[30] Georges Matheron. *Le Krigeage Universel*. Vol. Fascicule 1. Les Cahiers du Centre de Morphologie Mathematique de Fontainebleau. Fontainebleau: Ecole Nationale Superieure des Mines de Paris, 1969 (cit. on p. 5).

[31] Jerome Sacks, William J. Welch, Toby J. Mitchell, and Henry P. Wynn. «Design and Analysis of Computer Experiments». In: *Statistical Science* 4.4 (1989), pp. 409–423. DOI: 10.1214/ss/1177012413 (cit. on p. 5).

[32] Ivo Couckuyt, Tom Dhaene, and Piet Demeester. «ooDACE toolbox A Matlab Kriging toolbox: Getting started». In: 2013. URL: https://sumo.ilabt.imec.be/sites/default/files/flmngr/gettingstarted.pdf (cit. on pp. 9, 26, 27).

[33] Nestor V. Queipo, Raphael T. Haftka, Wei Shyy, Tushar Goel, Rajkumar Vaidyanathan, and P. Kevin Tucker. «Surrogate-based analysis and optimization». In: *Progress in Aerospace Sciences* 41.1 (2005), pp. 1–28. ISSN: 0376-0421. DOI: https://doi.org/10.1016/j.paerosci.2005.02.001 (cit. on pp. 18–20).

[34] Thomas Mackman, C. Allen, Mehdi Ghoreshi, and Ken Badcock. «Comparison of Adaptive Sampling Methods for Generation of Surrogate Aerodynamic Models». In: *AIAA Journal* 51 (Apr. 2013), pp. 797–808. DOI: 10.2514/1.J051607 (cit. on pp. 18, 19, 21, 25).

[35] M.E. Johnson, L.M. Moore, and D. Ylvisaker. «Minimax and maximin distance designs». In: *Journal of Statistical Planning and Inference* 26.2 (1990), pp. 131–148. ISSN: 0378-3758. DOI: https://doi.org/10.1016/0378-3758(90)90122-B (cit. on p. 23).

[36] Max D. Morris and Toby J. Mitchell. «Exploratory designs for computational experiments». In: *Journal of Statistical Planning and Inference* 43.3 (1995), pp. 381–402. ISSN: 0378-3758. DOI: https://doi.org/10.1016/0378-3758(94)00035-T (cit. on pp. 23, 24).

[37] Yangkun Jin, Jinlong Li, Wenli Du, and Feng Qian. «Adaptive Sampling for Surrogate Modelling with Artificial Neural Network and its Application in an Industrial Cracking Furnace». In: *The Canadian Journal of Chemical Engineering* 94 (Nov. 2015). DOI: 10.1002/cjce.22384 (cit. on p. 25).

[38] Donald Jones, Matthias Schonlau, and William Welch. «Efficient Global Optimization of Expensive Black-Box Functions». In: *Journal of Global Optimization* 13 (Dec. 1998), pp. 455–492. DOI: 10.1023/A:1008306431147 (cit. on p. 25).

[39] Chun Tang, Ken Gee, and Scott Lawrence. «Generation of Aerodynamic Data using a Design Of Experiment and Data Fusion Approach». In: Jan. 2005. ISBN: 978-1-62410-064-2. DOI: 10.2514/6.2005-1137 (cit. on p. 25).

[40] Shion Takeno, Hitoshi Fukuoka, Yuhki Tsukada, Toshiyuki Koyama, Motoki Shiga, Ichiro Takeuchi, and Masayuki Karasuyama. «Multi-fidelity Bayesian Optimization with Max-value Entropy Search and its Parallelization». In: *Proceedings of the 37th International Conference on Machine Learning*. Ed. by Hal Daumé III and Aarti Singh. Vol. 119. Proceedings of Machine Learning Research. PMLR, 13–18 Jul 2020, pp. 9334–9345. URL: https://proceedings.mlr.press/v119/takeno20a.html (cit. on pp. 40, 41).

[41] Philipp Hennig and Christian Schuler. «Entropy Search for Information-Efficient Global Optimization». In: *Journal of Machine Learning Research* 13 (Dec. 2011) (cit. on p. 41).

[42] José Miguel Hernández-Lobato, Matthew W Hoffman, and Zoubin Ghahramani. «Predictive Entropy Search for Efficient Global Optimization of Black-box Functions». In: *Advances in Neural Information Processing Systems*. Ed. by Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger. Vol. 27. Curran Associates, Inc., 2014. URL: https://proceedings.neurips.cc/paper_files/paper/2014/file/069d3bb002acd8d7dd095917f9efe4cb-Paper.pdf (cit. on pp. 41, 55).

[43] Zi Wang and Stefanie Jegelka. «Max-value Entropy Search for Efficient Bayesian Optimization». In: *Proceedings of the 34th International Conference on Machine Learning*. Ed. by Doina Precup and Yee Whye Teh. Vol. 70. Proceedings of Machine Learning Research. PMLR, June 2017, pp. 3627–3635. URL: https://proceedings.mlr.press/v70/wang17e.html (cit. on p. 41).

[44] Kevin Swersky, Jasper Snoek, and Ryan P Adams. «Multi-Task Bayesian Optimization». In: *Advances in Neural Information Processing Systems*. Ed. by C.J. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger. Vol. 26. Curran Associates, Inc., 2013. URL: https://proceedings.neurips.cc/paper_files/paper/2013/file/f33ba15effa5c10e873bf3842afb46a6-Paper.pdf (cit. on p. 41).

[45] Yehong Zhang, Trong Nghia Hoang, Bryan Kian Hsiang Low, and Mohan S. Kankanhalli. «Information-Based Multi-Fidelity Bayesian Optimization». In: 2017. URL: https://api.semanticscholar.org/CorpusID:199386827 (cit. on p. 41).

[46] T.M. Cover and J.A. Thomas. *Elements of Information Theory*. Wiley, 2012. ISBN: 9781118585771 (cit. on p. 41).

[47] Huachao Dong, Baowei Song, Peng Wang, and Shuai Huang. «Multi-fidelity information fusion based on prediction of kriging». In: *Structural and Multidisciplinary Optimization* 51 (Dec. 2014). DOI: 10.1007/s00158-014-1213-9 (cit. on pp. 47, 48, 53, 58, 65).

[48] David J. Toal. «Some considerations regarding the use of multi-fidelity Kriging in the construction of surrogate models». In: *Struct. Multidiscip. Optim.* 51.6 (June 2015), pp. 1223–1245. ISSN: 1615-147X. DOI: 10.1007/s00158-014-1209-5 (cit. on pp. 48, 58, 65).

[49] G.P. Sutton, O. Biblarz, and Knovel (Firm). *Rocket Propulsion Elements*. John Wiley & Sons, 2001. ISBN: 9781601190604 (cit. on pp. 74, 76, 78, 83).

[50] Raymond Bzibziak. «Update of Cold Gas Propulsion at MOOG». In: 465 (Nov. 2000), p. 553. DOI: 10.2514/6.2000-3718 (cit. on p. 74).

[51] A. A. Hashem. «Design and Testing of a Cold Gas System». In: *4th International Spacecraft Propulsion Conference*. Ed. by A. Wilson. Vol. 555. ESA Special Publication. Oct. 2004, p. 86.1 (cit. on pp. 74, 75).

[52] C.D. Brown. *Spacecraft Propulsion*. AIAA Education Series. American Institute of Aeronautics & Astronautics, 1996. ISBN: 9781600861161. URL: https://books.google.it/books?id=P5dBC_JgSLEC (cit. on pp. 75, 79, 81).

[53] H. Nguyen, J. Köhler, and L. Stenmark. «The merits of cold gas micropropulsion in state-of-the-art space missions». In: *The 53rd International Astronautical Congress, IAC 02-S.2.07*. Houston, Texas, 2002 (cit. on pp. 76, 77).

[54] Juergen Mueller and Juergen Mueller. «Thruster options for microspacecraft - A review and evaluation of existing hardware and emerging technologies». In: *33rd Joint Propulsion Conference and Exhibit*. DOI: 10.2514/6.1997-3058. URL: https://arc.aiaa.org/doi/abs/10.2514/6.1997-3058 (cit. on p. 77).

[55] Rhee, Charles M. Zakrzwski, and Thomas. «Highlights of Nanosatellite Propulsion Development Program at NASA-Goddard Space Flight Center». In: 2000. URL: https://api.semanticscholar.org/CorpusID:55292009 (cit. on p. 77).

[56] Aysu Özden, Özgür Baran, Mehmet Aksel, and Mehmet Ak. «SU2 REAL GAS MODELS' PERFORMANCE PREDICTIONS ON A COLD GAS THRUSTER». In: *Isı Bilimi ve Tekniği Dergisi* 41.1 (2021), pp. 77–88. DOI: 10.47480/isibted.979351 (cit. on p. 77).

[57] F. La Torre. «Gas Flow in Miniaturized Nozzles for Micro-Thrusters». Promotor: C. Kleijn, Faculty of Applied Sciences, Department of Multi-Scale Physics. PhD thesis. Delft, Netherlands: Delft University of Technology, June 2011 (cit. on pp. 83–85, 91, 96).

[58] E.W. Spisz, P.F. Brinich, J.R. Jack, United States. National Aeronautics, Space Administration, and Lewis Research Center. *Thrust Coefficients of Low-thrust Nozzles*. NASA technical note. National Aeronautics and Space Administration, 1965 (cit. on pp. 84, 85).

[59] P. F. Massier, L. H. Back, M. B. Noel, and F. Saheli. «Viscous effects on the flow coefficient for a supersonic nozzle». In: *AIAA Journal* 8.3 (1970), pp. 605–607. DOI: 10.2514/3.5728 (cit. on p. 85).

[60] G. V. R. RAO. «Exhaust Nozzle Contour for Optimum Thrust». In: *Journal of Jet Propulsion* 28.6 (1958), pp. 377–382. DOI: 10.2514/8.7324 (cit. on p. 87).

[61] Ravi Ranjan, S K Chou, Fahid Riaz, and Kumarasamy Karthikeyan. «Cold gas micro propulsion development for satellite application». In: *Energy Procedia* 143 (2017). Leveraging Energy Technologies and Policy Options for Low Carbon Cities, pp. 754–761. ISSN: 1876-6102. DOI: https://doi.org/10.1016/j.egypro.2017.12.758 (cit. on p. 91).

[62] Renzo Arina. *Fondamenti di Aerodinamica*. Levrotto & Bella, 2015. ISBN: 9788882181871 (cit. on pp. 101, 102, 104).

[63] Jiri Blazek. *Computational Fluid Dynamics: Principles and Applications*. English. 2nd. Elsevier, 2005. ISBN: 9780080445069 (cit. on pp. 102, 104).

[64] Abdullah Konak, David W. Coit, and Alice E. Smith. «Multi-objective optimization using genetic algorithms: A tutorial». English (US). In: *Reliability Engineering and System Safety* 91.9 (Sept. 2006), pp. 992–1007. ISSN: 0951-8320. DOI: 10.1016/j.ress.2005.11.018 (cit. on p. 125).