ABSTRACT

Title of Dissertation: EFFICIENT COMPUTATIONAL ALGORITHMS FOR MAGNETIC EQUILIBRIUM IN A FUSION REACTOR

Jiaxing Liang

Doctor of Philosophy, 2024

Dissertation Directed by: Professor Howard C. Elman

Department of Computer Science,

University of Maryland, College Park.

Professor Tonatiuh Sánchez-Vizuet

Department of Mathematics,

University of Arizona.

In a magnetic confinement fusion reactor, like a Tokamak, hydrogen isotopes are injected into the chamber and heated to form a plasma. The hot plasma tends to expand, so it's crucial to confine it within the center of the reactor to prevent contact with the reactor walls. This confinement is achieved through magnetic fields generated by currents running through external coils surrounding the reactor. However, these currents may suffer from uncertainties, which could arise from factors like temperature fluctuations and material impurities. These variables introduce a level of unpredictability in the plasma's behavior and the overall stability of the fusion process.

This thesis aims to investigate the impact that stochasticity in the current intensities has on the confinement properties of the plasma and to estimate the expected behavior of the magnetic field. While the focus is on the variability in current intensities, the tools developed can be applied to other sources of uncertainty, such as the positioning of coils and the source term parameters. To quantify the variability in model predictions and to evaluate the statistical properties of solutions over a range of parameter values, traditional sampling methods like Monte Carlo, often require intensive and expensive nonlinear computations. To tackle this challenge, we propose three approaches.

Firstly, we focus on the development and application of a surrogate function, constructed via a stochastic collocation approach on a sparse grid in the parameter space. This surrogate function is employed to replace the nonlinear solution in Monte Carlo sampling processes. For our numerical experiments, we evaluate the efficiency and accuracy of the outcomes produced by the surrogate, in comparison with those obtained through direct nonlinear solutions. Our

findings indicate that a carefully selected surrogate function reduces the sampling cost – achieving acceleration factors ranging from 7 to over 30 – while preserving the accuracy of the results.

The second part of the thesis explores the multilevel Monte Carlo approach, investigating its potential for cost savings compared to simple Monte Carlo. This method involves conducting the majority of computations on a sequence of coarser spatial grids compared to what a simple Monte Carlo simulation would typically use. We examine this approach with non-linear computation, using both uniformly refined meshes and adaptively refined grids guided by a discrete error estimator. Numerical experiments reveal substantial cost reductions achieved through multilevel methods, typically ranging from a factor of 60 to exceeding 200. Adaptive gridding results in more accurate computation of relevant geometric parameters.

In the last part of this work, we explore hybrid methods that integrate surrogates with multilevel Monte Carlo to further reduce the sampling cost. We establish the optimal construction and sampling costs for the surrogate-based multilevel Monte Carlo. Numerical results demonstrate that surrogate-based multilevel Monte Carlo remarkably reduces the computational burden, requiring only 0.1 to 14 seconds for a target relative mean square error ranging from 8×10^{-3} to 2×10^{-4}, reducing the cost of direct computation by factors of 50 to 300. In terms of accuracy, the surrogate-based sampling results exhibit close congruence with those obtained via direct computation, both in plasma boundary and geometric descriptors.

The primary contributions of our work entail the application of stochastic collocation techniques and multilevel Monte Carlo methods to analyze plasma behavior under uncertainties in current within fusion reactors. Furthermore, we establish the universal sampling cost for the surrogate-enhanced multilevel Monte Carlo approach. Our methodology presents a paradigm in how we approach and manage computational challenges in this field.

EFFICIENT COMPUTATIONAL ALGORITHMS FOR MAGNETIC EQUILIBRIUM IN A FUSION REACTOR

by

Jiaxing Liang

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2024

Advisory Committee:
Professor Howard C. Elman, Chair/Advisor
Professor Tonatiuh Sánchez-Vizuet, Co-Advisor
Professor James D. Baeder
Professor Matt Landreman
Professor Ricardo H. Nochetto

Acknowledgments

As I stand on the threshold of completing this thesis, I am deeply indebted to individuals who have played pivotal roles in its fruition, enriching my academic journey in ways beyond measure.

First and foremost, I would like to express my deepest gratitude to my advisors, Professors Howard Elman and Tonatiuh Sánchez-Vizuet, whose guidance and support have been the cornerstone of my academic journey. They have played pivotal roles in giving me an invaluable opportunity to work on challenging and interesting projects and shaping the trajectory of my research in my doctoral studies. Without their extraordinary theoretical ideas and computational expertise, this thesis would have been a distant dream. I also appreciate their patient help on improving my academic writing and presentation skills. Beyond the academic realm, I am grateful for their wisdom which has been invaluable in navigating the complexities of both academia and life. Their influence extends far beyond the confines of the classroom. I consider myself exceptionally fortunate to have had them as my advisors, and I sincerely appreciate their dedication, patience, and mentorship.

I extend my appreciation to Professors James Baeder, Matt Landreman, and Ricardo Nochetto for graciously agreeing to serve on my thesis committee. Their insightful feedback and the invaluable time dedicated to reviewing the manuscript have been pivotal to the refinement of this work. Moreover, I want to express my thankfulness to

- Professor James Baeder, I am grateful for your guidance in my study of computational fluid dynamics. Your CFD courses have provided me with a profound understanding of numerical methods for PDEs, shaping the foundation of my research.

- Professor Maria Cameron, whose well-organized lectures on numerical PDE, data science, and stochastic processes have been instrumental in shaping my academic journey. I am thankful for the knowledge and insights gained from the classes I took under your guidance.

- Professor Matt Landreman, your intuitive questions and suggestions have been a pleasure to engage with, making our interactions a source of inspiration for my research endeavors.

- Professor Ricardo Nochetto, I consider myself extremely fortunate to be your student. Your fantastic course on numerical PDE has been transformative, contributing significantly to my academic development.

- Professor Tobias von Petersdorff, your intuitive questions and suggestions have left an indelible mark on my research work, and I am grateful for your support.

A heartfelt thank you goes to my friends – your encouragement, shared challenges, and celebratory moments have made this experience truly memorable. Your friendship has been a constant source of strength, and I am grateful for the camaraderie we share.

To my beloved family – your enduring support has been the cornerstone of my achievements. Your sacrifices, wisdom, and unwavering belief in my abilities have been a guiding force. I am profoundly grateful for the love and encouragement that have sustained me throughout this academic pursuit.

In closing, this thesis is a collective effort, and I am fortunate to have had the support

of exceptional professors, steadfast friends, and a loving family. Each of you has played a unique role in my academic and personal growth. Thank you all for being an integral part of this journey.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

MC	Monte Carlo
MC-FE	Monte Carlo Finite-Element
MLMC	Multilevel Monte Carlo
MLMC-FE	Multilevel Monte Carlo Finite-Element
ML-ML	Multilevel sparse grid stochastic collocation
SL-ML	Single-level sparse grid stochastic collocation on the multilevel spatial grids
SL-SL	Single-level sparse grid stochastic collocation on the single-level spatial grid

Chapter 1: Introduction

This thesis incorporates the findings and methodologies outlined in the following papers, adhering to the permissions provided by Elsevier's policies:

1. Surrogate approximation of the Grad-Shafranov free boundary problem via stochastic collocation on sparse grids. (Journal of Computational Physics, 448 (2022), paper No. 110699. doi: 10.1016/j.jcp.2021.110699)

2. Multilevel Monte Carlo methods for the Grad-Shafranov free boundary problem. (Computer Physics Communications, 298 (2024), paper No. 109099. doi: 10.1016/j.cpc.2024.109099)

The goal is the advancement of computational algorithms aimed at quantifying uncertainty in the magnetic equilibrium of fusion reactors, particularly in the context of the Grad-Shafranov free boundary problem. We will focus on equilibrium computations on Tokamaks, a family of axially symmetric reactors whose design dates back to the work of Soviet scientists in the late 1960's [3, 72]. In the introductory chapter, we provide a background on the model problem and introduce the intricacies of uncertainty quantification. Then we discuss three distinct computational approaches employed to assess the impact of uncertainty on magnetic equilibrium. The chapter concludes with a summary of contributions and an outline for the subsequent chapters.

1.1 Background

A fusion reaction takes place when two light atomic nuclei overcome the Coulomb force barrier, merging into a single nucleus bonded by the strong nuclear force. The mass of the initial two nuclei is heavier than that of the newly formed nucleus, with the mass discrepancy being converted into a substantial amount of energy. In a fusion reactor, hydrogen isotopes

such as deuterium and tritium are utilized as fuel gas, and injected into a vacuum vessel. This gas is then heated to high temperatures, causing the removal of electrons from atoms and turned into a plasma – a state of matter comprising charged particles like ions and electrons. The high temperature empowers ions with sufficient kinetic energy for collisions, enabling them to approach closely enough to overcome the repulsive force barrier, leading to fusion into heavier nuclei and the release of energy.

To maintain the requisite conditions for continuous fusion reactions and prevent direct contact between the high-temperature plasma and the chamber wall, effective plasma confinement is imperative. Plasmas, being conductive and magnetically tractable, can be confined using devices that generate magnetic fields. Devices such as Tokamaks utilize magnetic fields generated by coils surrounding the vessel to achieve plasma confinement. Throughout the confinement, charged particles undergo the interplay of hydrostatic force and magnetic forces generated by external coils and the internal plasma current, ultimately reaching an equilibrium state.

The partial differential equation describing the magnetic equilibrium depends on several physical parameters, including the spatial arrangement of coils, the intensity of electric currents running through them, magnetic permeability, etc. The uncertainty and variability inherent in these parameters prompts an exploration of their impact on plasma confinement attributes. Our project investigates the effects of uncertainty arising from the currents in the coils, which we treat as stochastic parameters, on the properties of confined plasma [18]. This research aims to quantify variability in predictions, especially in determining the expected solution for magnetic equilibrium. Additionally, our objective extends to the computation of derived quantities from this expectation, such as the plasma boundary and other geometric features. However, the dimensionality of the parameter space (twelve in the particular model of our studies) presents significant computational challenges in approximating the stochastic integral for the expectation. To address this, sampling methods, particularly Monte Carlo simulations, are commonly employed despite their inherent computational drawbacks stemming from their square root convergence rate.

1.2 Contributions of the thesis

To mitigate the cost associated with directly computing the parametrized solutions, surrogate approximations to the solution are commonly employed as an alternative. Constructing a surrogate may involve creating an interpolating function based on solving the discrete system with a designated set of parameter values. However, exploring the multidimensional parameter space can quickly become computationally intractable, posing computational challenges in both surrogate construction and sampling. In response to this challenge, the *sparse grid stochastic collocation* (SC) approach [5, 67] is considered for our work. This method establishes a quadrature on a smaller subset of full tensor grid nodes, referred to as *sparse grids*, using Smolyak's scheme [67]. The resulting surrogates generated from this approach demonstrate reliability comparable to that obtained with full tensor product grids [5, 67, 74] and will be utilized for sampling across various methods. Additionally, we explore an alternative sampling strategy – multilevel Monte Carlo [12, 28, 61, 69] – aimed at enhancing the efficiency of Monte Carlo through the adoption of a multilevel strategy in the physical space. This method effectively reduces sampling costs by concentrating the majority of computations on a sequence of coarser spatial grids compared to what a standard Monte Carlo simulation typically employs. In this thesis, we will investigate three sampling approaches to efficiently approximate the expectation of the magnetic equilibrium: implementing a surrogate function, utilizing multilevel sampling methods, and a hybrid approach that integrates both techniques.

The thesis is structured as follows. In Chapter 2, we provide an introduction to the model problem, incorporating uncertainty, and outline the objectives of this thesis. Chapter 3 is dedicated to presenting the sparse grid stochastic collocation approach. The construction of surrogates for the function model based on sparse grid stochastic collocation is detailed in Chapter 4. Moving on to Chapter 5.1, we delve into the discussion of the Monte Carlo Finite-Element approach, considering samplings with both direct non-linear computation and surrogate function. In Chapter 5.2, we explore the multilevel Monte Carlo Finite-Element approach, including samplings with direct computation on both uniform and adaptive meshes. Additionally, we investigate both single-level and multilevel surrogate-based samplings on the uniform mesh set. Chapter 6 is devoted to presenting numerical experiments comparing the efficiency and accuracy of these sampling strategies. Finally, concluding remarks are

provided in Chapter 7. For comprehensive coverage, technical mathematical and algorithmic aspects pertaining to the problem and the methods discussed are included as an Appendix.

Chapter 2: Model problem

In this chapter, we present the deterministic partial differential equation that governs static magnetic equilibrium within an axially symmetric context. We explore fundamental equations and discuss plasma confinement configurations, geometric parameters, and the variational formulation for our model. Extending this deterministic framework, we introduce a version that incorporates uncertainty, acknowledging the stochastic nature of key parameters. The chapter concludes by outlining the thesis goal.

2.1 The deterministic problem

2.1.1 The Grad-Shafranov free boundary problem

In our model problem, the axially symmetric geometry is conveniently described using cylindrical coordinates (r, z, φ), where the "poloidal" component lies in the meridian plane and the "toroidal" component aligns with the toroidal direction φ. By exploiting the axial-symmetry property and assuming independence in the toroidal direction, we simplify the three-dimensional problem to a two-dimensional representation in the poloidal plane. In this work, instead of using (r, z) to denote positions on the plane at any fixed angle φ, we adopt the notation (x, y) to represent the distance from the axis of symmetry and the height, respectively.

We will elucidate the dynamics of a plasma system, considering parameters such as mass density ρ, mean velocity of ion particles $\boldsymbol{v}$, total current density $\boldsymbol{J}$, hydrodynamic pressure p, and total magnetic field $\boldsymbol{B}$. The momentum equation governing the plasma motion is formulated as an equilibrium between the acceleration of fluid elements $(d\boldsymbol{v}/dt)$, arising from the Lorentz force $(\boldsymbol{J} \times \boldsymbol{B})$, and the gradient of the scalar pressure, described by

$$\rho \frac{d\boldsymbol{v}}{dt} = \boldsymbol{J} \times \boldsymbol{B} - \nabla p,$$

If we assume a time-independent static equilibrium, the inertial term $\rho d\boldsymbol{v}/dt$ can be neglected [2, 33]. Consequently, a magnetic equilibrium is established at each temporal instance, where the magnetic forces from external coils and internal plasma current balance the hydrostatic pressure, as expressed by

$$\boldsymbol{J} \times \boldsymbol{B} = \nabla p. \tag{2.1}$$

Let $\boldsymbol{H}$ be the magnetic field and μ be the magnetic permeability, then $\boldsymbol{B} = \mu \boldsymbol{H}$. Additionally, the plasma must satisfy two fundamental equations:

$$\text{curl } \mathbf{H} = \mathbf{J} \quad \text{(Ampère's Law)}, \tag{2.2}$$

$$\text{div } \mathbf{B} = 0 \quad \text{(Gauss's law)}. \tag{2.3}$$

These, along with the magnetic equilibrium condition, give rise to the *Grad-Shafranov equation* [32, 55, 65] – a second-order nonlinear elliptic equation that relates the *poloidal flux function* $\psi(x, y)$ to the toroidal current density J_φ. In an axially symmetric setting, our model is reduced to a two-dimensional space, and the magnetic field $\mathbf{B}$ can be expressed in terms of two scalar functions $\psi(x, y)$ and $g(x, y)$ as

$$\mathbf{B} = \widehat{\boldsymbol{\phi}} \times \frac{\text{grad } \psi}{r} + \widehat{\boldsymbol{\phi}} \frac{g}{r}, \tag{2.4}$$

where $\widehat{\boldsymbol{\phi}}$ is the unit vector in the direction of increasing angle ϕ. The equation can be expressed as

$$-\nabla \cdot \left(\frac{1}{\mu x} \nabla \psi \right) = J_\varphi,$$

where ∇ and $\nabla\cdot$ denote the Cartesian gradient and divergence operators in two dimensions respectively; the magnetic permeability $\mu = \mu_0$ in vacuum, and is a function of the magnetic field $\mu = \mu(|\nabla\psi|^2/x^2)$ within any ferromagnetic structure. The toroidal current density can be represented using source term functions, *hydrostatic pressure* $p(\psi)$ and *toroidal field* $g(\psi)$, as

$$J_\varphi = \frac{d}{d\psi} p(\psi) + \frac{1}{2\,\mu x} \frac{d}{d\psi} g^2(\psi).$$

The function g is associated with the total current flowing in the poloidal direction, I_p, through the relation $I_p = -2\pi g$. Both g and the pressure p are functions of the poloidal flux ψ alone. The formulation can be extended in free space, where the toroidal current density J_φ includes contributions from currents I_k passing through external coils. This results in the

non-linear equation

$$-\nabla \cdot \left(\frac{1}{\mu x}\nabla\psi\right) = \begin{cases} \frac{d}{d\psi}p(\psi) + \frac{1}{2\,\mu x}\frac{d}{d\psi}g^2(\psi) & \text{in } \Omega_p(\psi) \\ I_k/S_k & \text{in } \Omega_{C_k} \\ 0 & \text{elsewhere,} \end{cases} \tag{2.5a}$$

where Ω_p is the region where the plasma is present and Ω_{C_k} represents the regions of external coils with cross-sectional areas S_k. The delineation of Ω_p is dependent on the last closed level set, either passing through a saddle point or running tangentially to the limiter – an installed plate to prevent the plasma from contacting the reactor wall. This level set determines the plasma boundary $\partial\Omega_p$, which is dependent on ψ and must be determined simultaneously along with the magnetic field. It is *not known a priori*, resulting in a *free boundary problem*. In addition, the physical properties of anti-symmetry of the solution and finite energy result in the following conditions on ψ

$$\psi(0, y) = 0; \qquad \lim_{\|(x,y)\|\to\infty} \psi(x, y) = 0, \tag{2.5b}$$

which guarantees the well-posedness of the model problem.

The form of the source term in the plasma that we will use is

$$\frac{d}{d\psi}p(\psi) = \frac{\beta}{x_0}\left(1 - \psi_N^{\alpha}\right)^{\gamma}, \qquad \frac{1}{2}\frac{d}{d\psi}g^2(\psi) = \mu_0 x_0(1-\beta)\left(1 - \psi_N^{\alpha}\right)^{\gamma}. \tag{2.5c}$$

These forms were proposed by Luxon and Brown [56] as the simplest model that was statistically adequate to describe experimental results and has been widely used for theoretical studies ever since. The term

$$\psi_N := \frac{\psi - \psi_{BD}}{\psi_{MA} - \psi_{BD}}$$

is a normalization of the unknown with respect to the value of ψ at the plasma boundary, ψ_{BD}, and the maximum value of ψ within the plasma, ψ_{MA}, which determines the location of the point known as the *magnetic axis*. The value x_0 corresponds to the outer radius of the vacuum chamber, the parameters α and γ control how sharply the current peaks near the magnetic axis; β is known as *poloidal beta*, a parameter measuring the ratio between the hydrostatic pressure in the plasma due to temperature (usually called *plasma pressure*) and the pressure due to the external magnetic field (or *magnetic pressure*). Since the aim of magnetic confinement is to balance the effects of the internal hydrostatic pressure with those of the external magnetic field, the value of β effectively measures the strength of confinement

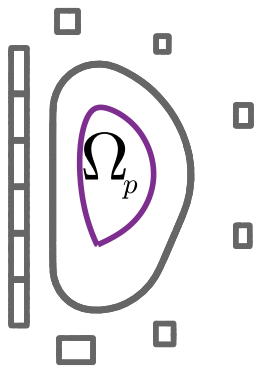

Figure 2.1: The plasma confinement region Ω_p is enclosed by the violet line. The rectangles represent the external coils C_k; the grey curved line represents the exterior wall of the vacuum chamber.

of a reactor; in practice this value is much smaller than one. All these parameters are determined by the user, depending on the design of the reactor or the particularities of an experiment, mainly the desired current and pressure profiles in the plasma.

A schematic of a cross-section, for $x > 0$, of a tokamak is depicted in Figure 2.1. The confinement region Ω_p is characterized as the largest region that contains the magnetic axis (defined as the point where ψ has a global maximum) and that is bounded by a closed level set $\psi = constant$. The solution to this free boundary problem is ubiquitous in nuclear fusion and several computational codes have been developed over the years (see for instance [21, 31, 35, 36, 39, 45] and references therein).

2.1.2 Limited and diverted configurations

Several scenarios can occur for plasma contained in a tokamak. Cross-sections of typical plasma configurations are shown in Figure 2.2. Depending on the configuration, in the interior of the vacuum vessel either divertor or limiter plates (solid black) are installed to protect the walls. In the generic top left image, the plasma is contained in a hollow toroidal vacuum vessel (violet line) and the confinement field is generated by an external array of coils (purple blocks).

Depending on the location of the coils and the intensity of the current passing through them, there are three possible configurations of the magnetic field that are relevant for

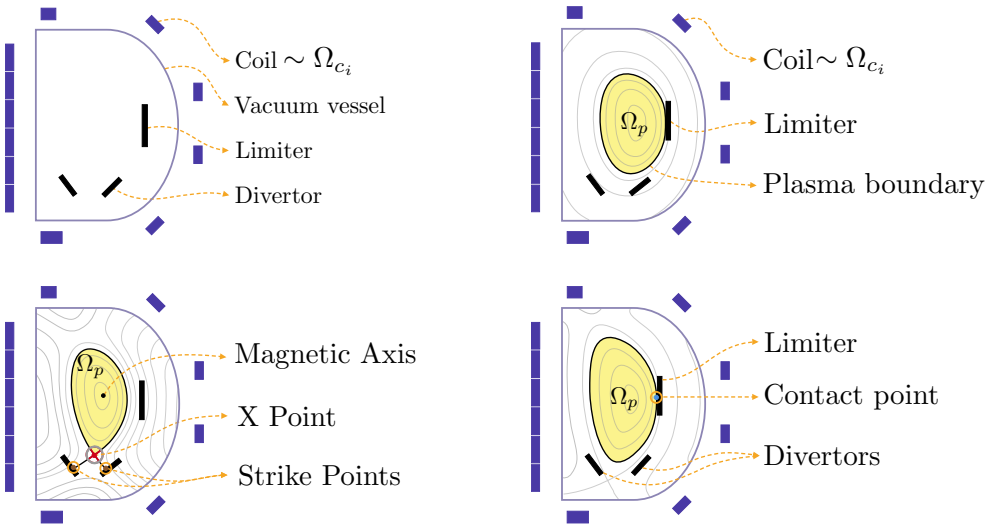

Figure 2.2: Schematics of a cross section of a tokamak and possible plasma configurations. Top left: simplified cross-section of a tokamak depicting some of the essential components for magnetic confinement. Top right: a *limited configuration*. Bottom left: a *diverted configuration*. Bottom right: the separatrix intersects the limiter plate.

confinement. The top right image in Figure 2.2 shows a *limited* configuration, where the contour lines of ψ contained in the vacuum vessel are all closed and nested, interrupted only by the walls of the chamber. In this situation, common in older reactors, the plasma could come in contact with the walls and therefore a physical barrier—typically a tungsten plate depicted as a solid black line in the schematic—has to be positioned to prevent the plasma from reaching the walls [72]. The level set of ψ tangential to the limiter determines the boundary of the confinement region Ω_p. While this solution succeeds at shielding the walls of the reactor, the fact that the plasma is in direct contact with the limiter has some undesirable effects including a reduced range of temperatures attainable and the introduction of pollution due to the ablation of the limiter plate.

Alternatively, as in the bottom left image of Figure 2.2, the magnetic field can give rise to a situation where the interior of the vessel is divided into a sector where the contour lines are closed and one in which they are open. In this *diverted* configuration, the stream function has a separatrix that divides the interior of the vessel into these two regions. As confinement is possible only in regions where the field lines are closed, if this line is not obstructed by any structure, then the plasma remains confined within the separatrix without the need for a

limiter. The point where the separatrix intersects itself is called an *x-point*. Since heavy ions or impurities leaving the plasma do so by moving along magnetic stream lines, the vicinity of the x-point, where the separatrix intersects itself, is an ideal location for placing a filtering device known as *divertor* that removes such ions and absorbs the excess heat, protecting the walls without coming directly in contact with the bulk of the plasma [15, 43]. The area surrounding the points where the separatrix intersects the divertor plates—the *strike points*—is also of interest, since it is the most likely to get in contact with fast particles and impurities leaving the plasma.

Owing to these and other advantages, reactors with magnetic x-points and diverted configurations are favored nowadays; designs with two or more divertors have been proposed and studied [7, 46, 64]. It is clear that accurate determination of the magnetic equilibrium configuration and in particular the location of a magnetic x-point on a given reactor is of great relevance for theoretical, experimental and design purposes.

A third possibility, shown in the bottom right schematic of Figure 2.2, is that the stream function gives rise to a separatrix that intersects the walls of the vacuum chamber or some other structure contained in it, such as the divertor or a limiter. In this case, although confinement is still possible, the separatrix no longer determines the plasma boundary, as the disrupting structure acts effectively as a limiter. Instead, the plasma boundary is determined by the first closed contour inside the separatrix that is tangential to said structure at a contact point. This scenario is undesirable, for the plasma walls or structures inside of the vacuum chamber, other than the limiter and divertor plates, are not designed to be in direct contact with the plasma and can be severely damaged by such interactions. As will be seen later, even if reactors are designed to avoid this scenario, unexpected variations in current intensity can give rise to such disruptions. Being able to quantify the probability of this happening (and the most likely location of contact with any structure) is therefore of paramount importance.

2.1.3 Plasma shaping

Some geometric properties of the plasma boundary have been found to have a significant influence on important physical processes such as stability and transport. Substantial theoretical and experimental efforts have been devoted to the design and realization of particular confinement geometries in what has come to be known as *plasma shaping* [6, 24, 40, 48, 73].

The geometric features that have been found to be most relevant are referred to as *elonga-*

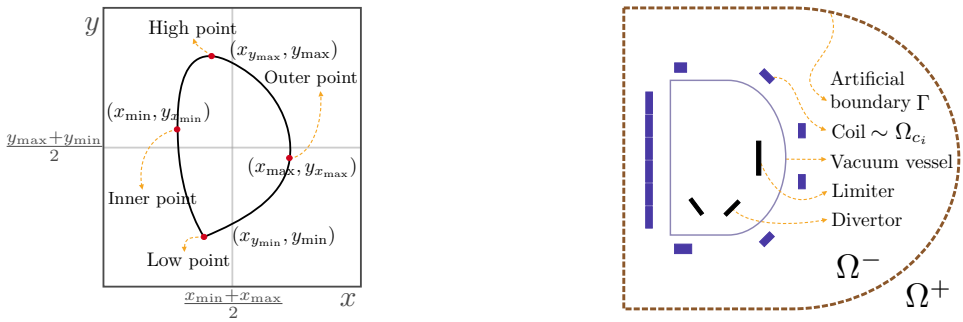

Figure 2.3: Left: Locations of the high, low, inner and outer points in a typical plasma boundary. These points are used to define the shaping parameters in (2.6): major and minor radii, aspect ratio, elongation and triangularity. Right: An artificial boundary Γ enclosing the reactor on all the relevant geometric structures is introduced. The region contained within Γ is meshed with a geometry-fitting triangulation, where the equation (2.5a) is discretized.

tion and *triangularity*. We define these terms by introducing a few auxiliary quantities—the notation and terminology are standard in the fusion literature, see for instance [11, 54]:

$$R_{geo} := (x_{max} + x_{min})/2 \qquad \text{Major radius,} \tag{2.6a}$$

$$a := (x_{max} - x_{min})/2 \qquad \text{Minor radius,} \tag{2.6b}$$

$$\epsilon := a/R_{geo} \qquad \text{Inverse aspect ratio,} \tag{2.6c}$$

$$\kappa := (y_{max} - y_{min})/2a \qquad \text{Elongation,} \tag{2.6d}$$

$$\delta_u := (R_{geo} - x_{y_{max}})/a \qquad \text{Upper triangularity,} \tag{2.6e}$$

$$\delta_l := (R_{geo} - x_{y_{min}})/a \qquad \text{Lower triangularity.} \tag{2.6f}$$

The left panel of Figure 2.3 depicts the points that determine the height and major and minor radii on a typical plasma boundary with an x-point. Note that in general the locations of the Inner/Outer and High/Low points need not be aligned, nor does the shape of the plasma need to be symmetric.

2.1.4 Variational formulation

The mathematical and numerical exploration of the free boundary problem (2.5) holds profound significance in the realm of fusion science. Among the noteworthy recent endeavors

in this domain are the contributions made by Heumann and collaborators [21, 34, 35, 36]; here we rely on their contributions for the numerical computation of the equilibrium configuration, in particular, the `Matlab` implementation of the free boundary solver described in [35].

The main challenges posed by (2.5) are: (1) the unboundedness of the domain where the problem is posed, (2) the nonlinearity of the partial differential equation stemming from the source term and—in the presence of iron components—the permeability coefficient, and (3) the nonlinearity of the problem owing to the unknown nature of the plasma region Ω_p. Points (2) and (3) are addressed through an iterative Newton approach by linearizing around an appropriately chosen initial guess. Perhaps point (1) is the one that requires the most elaborate treatment, and we will describe it in some more detail in what follows.

We will make use of some results and terminology from the theory of boundary integral equations. We refer to [41] for a concise overview of the subject.

An artificial boundary Γ enclosing the entire reactor and remaining structures is introduced, as shown in the right panel of Figure 2.3. Γ must be such that the support of the right hand side of (2.5a) is compactly contained inside. This curve divides the half plane into the region enclosed by it, which we will denote by Ω^-, and its unbounded complement $\Omega^+ := \mathbb{R}^2 \setminus \overline{\Omega^-}$. For simplicity we will leave Γ as general as possible, but its precise form will be made explicit in due time.

The solution to the free boundary problem can then be considered to be of the form $\psi = \psi^- + \psi^+$, where the two parts ψ^- and ψ^+ are supported inside Ω^- and Ω^+ respectively. The continuity of ψ is preserved by requiring both interior and exterior solutions ψ^- and ψ^+ (and their normal derivatives) to match on the interface Γ. More precisely, problem (2.5) can be reformulated as a system consisting of the interior problem

$$-\nabla \cdot \left(\frac{1}{\mu x}\nabla\psi^-\right) = F(\psi^-, x, \Omega_p) := \begin{cases} x\frac{d}{d\psi^-}p(\psi^-) + \frac{1}{2\,\mu x}\frac{d}{d\psi^-}g^2(\psi^-) & \text{in } \Omega_p(\psi^-) \\ I_i/S_i & \text{in } \Omega_{C_i} \\ 0 & \text{Elsewhere.} \end{cases} \quad \text{in } \Omega^- \tag{2.7a}$$

$$\psi^-(0, y) = 0, \tag{2.7b}$$

and the exterior problem

$$-\nabla \cdot \left(\frac{1}{\mu x}\nabla\psi^+\right) = 0 \qquad \text{in } \Omega^+ \tag{2.7c}$$

$$\psi^+(0,y) = 0, \qquad \lim_{\|(x,y)\|\to\infty} \psi^+(x,y) = 0,$$

connected through the continuity conditions

$$\psi^- = \psi^+ \qquad \text{on } \Gamma \tag{2.7d}$$

$$\partial_\nu\psi^- = \partial_\nu\psi^+ \qquad \text{on } \Gamma. \tag{2.7e}$$

Testing with a function φ satisfying the boundary condition (2.7b) but otherwise arbitrary, one arrives at

$$\int_{\Omega^-} \frac{1}{\mu x}\nabla\psi^- \cdot \nabla\varphi \, dA - \int_\Gamma \frac{1}{\mu x}\partial_\nu\psi^- \, \varphi \, d\Gamma = \int_{\Omega^-} F(\psi^-, x, \Omega_p)\, \varphi \, dA. \tag{2.8a}$$

We now turn our attention to the exterior problem. In order to deal with the unbounded domain we will use an integral representation of the exterior solution of the form

$$\psi^+ = \int_\Gamma G\, \partial_\nu\psi^+ \, d\Gamma - \int_\Gamma \partial_\nu G\, \psi^+ \, d\Gamma,$$

where $G = G(\boldsymbol{x}_1, \boldsymbol{x}_2)$ is the Green's function for (2.5) vanishing on Γ. To avoid cumbersome notation, the explicit expression of the Green's function will not be given in this discussion. The terms in the right hand side are known respectively as *single layer potential* and *double layer potential*. Note that the values of the restrictions to the boundary and the normal derivatives appearing in the right hand side are not known a priori and have to be determined as part of the solution process.

Using the expression above to take the restriction of ψ^+ to the boundary Γ enables the use of the *jump properties* of the single and double layer potentials [41], which yield the following integral equation on Γ

$$\psi^- = \frac{1}{2}\psi^- + \int_\Gamma \partial_\nu G\, \psi^- \, d\Gamma + \int_\Gamma G\, \partial_\nu\psi^- \, d\Gamma$$

where (2.7d) and (2.7e) were used to connect the exterior and interior problems by writing the integral equation in terms of the restrictions of the interior solution only. This equation

can be posed weakly by testing with an arbitrary function λ defined on Γ, yielding

$$\begin{aligned}\int_\Gamma \frac{1}{2}\psi^- \,\lambda\, d\Gamma - \int_\Gamma\int_\Gamma \partial_{\nu(\boldsymbol{x}_1)} G(\boldsymbol{x}_1,\boldsymbol{x}_2)\,\psi^-(\boldsymbol{x}_1)\,\lambda(\boldsymbol{x}_2)\,d\Gamma(\boldsymbol{x}_1)\,d\Gamma(\boldsymbol{x}_2)\\ - \int_\Gamma\int_\Gamma G(\boldsymbol{x}_1,\boldsymbol{x}_2)\,\partial_{\nu(\boldsymbol{x}_1)}\psi^-(\boldsymbol{x}_1)\,\lambda(\boldsymbol{x}_2)\,d\Gamma(\boldsymbol{x}_1)\,d\Gamma(\boldsymbol{x}_2) = 0.\end{aligned} \tag{2.8b}$$

Equations (2.8a)-(2.8b) involve only the unknowns ψ^- and $\partial_\nu\psi^-$ so, if they can be solved simultaneously, the solution ψ to the original problem can be recovered.

So far, we have followed the general strategy introduced by Johnson and Nédélec in their celebrated article [44]. However, as shown by Hsiao and collaborators [25, 42], if Γ is taken to be a semi-circle, equation (2.8b) can be solved exactly for $\partial_\nu\psi^+$ in terms of the tangential derivative of ψ^- along Γ. Substitution of this expression for $\partial_\nu\psi^+$ back into (2.8a) gives the following uncoupled equation for ψ^-, which must hold for all sufficiently regular test functions φ vanishing on the y axis:

$$\begin{aligned}&\int_{\Omega^-} \frac{1}{\mu x}\nabla\psi^-\cdot\nabla\varphi\, dA - 2\int_\Gamma\int_\Gamma \frac{1}{\mu x_1} G(\boldsymbol{x}_1,\boldsymbol{x}_2)\,\partial_{\tau(\boldsymbol{x}_1)}\psi^-(\boldsymbol{x}_1)\,\partial_{\tau(\boldsymbol{x}_2)}\varphi(\boldsymbol{x}_2)\,d\Gamma(\boldsymbol{x}_1)\,d\Gamma(\boldsymbol{x}_2)\\ &= \int_{\Omega^-} F(\psi^-,x,\Omega_p)\,\varphi\, dA.\end{aligned}$$

The symbol ∂_τ denotes (weak) tangential differentiation along the curve Γ. In this equation, the unknowns are the stream function inside of the fictitious boundary, ψ^-, and the support of the plasma, Ω_p, which appears through the source term of the equation $F(\psi^-,x,\Omega_p)$—see (2.7). The boundary of the plasma region is the *closed* level set of ψ^- that either: a) passes through a saddle point of ψ^- located inside of the reactor, or b) is tangential to the inner wall of the reactor—if the curve defined by a) intersects any structure inside the reactor or if no curve satisfies the condition a). See Figure 2.2 for graphical depictions of these conditions.

Once ψ^- is found, the continuity conditions (2.7d) and (2.7e) along with the integral representation for ψ^+ can be used to evaluate ψ in the unbounded component Ω^+ if needed. Note that even if it is supported on the entire component Ω^+, the function ψ^+ appears in these equations only through the integrals over Γ, which avoids the discretization of an unbounded domain.

The explicit expression for the equation above, derived tersely in [1], is obtained by substituting the appropriate Green's function and integrating by parts the term involving the tangential derivatives. We can now state the variational problem that must be solved.

Variational Formulation. Consider a semi-circle centered at the origin, boundary Γ and with radius ρ such that it fully contains all the relevant reactor components depicted in Figure 2.1. If Ω^- denotes the region surrounded by Γ then, by construction, for any in $(\Omega^-)^c$ the right-hand side of (2.5a) will vanish identically. We will then, following [33], consider the space of real-valued functions

$$Z := \left\{ \psi : \Omega^- \to \mathbb{R} \,\middle|\, \int_{\Omega^-} \psi^2 x \, dx \, dy < \infty \, ; \, \int_{\Omega^-} \frac{|\nabla \psi|^2}{x} \, dx \, dy < \infty \, ; \, \psi(0, y) = 0 \right\} \cap \mathcal{C}^0 \left(\overline{\Omega^-} \right). \tag{2.9}$$

This space arises naturally when testing equation (2.5a) using the weighted L_2 inner product defined by

$$\langle \psi, \varphi \rangle := \int_{\Omega^-} \psi \varphi x \, dxdy,$$

which leads to the finite energy requirement appearing in the second inequality in the definition (2.9). The third requirement ($\psi(0, y) = 0$) is a result of the anti-symmetry of the problem with respect to reflections across the axis of symmetry of the reactor and has the effect of ensuring that the quantity

$$\|\psi\|_Z := \left(\int_{\Omega^-} \frac{|\nabla \psi|^2}{x} dxdy \right)^{1/2} \tag{2.10}$$

does indeed define a norm in the space Z. We will refer to this norm as the *energy norm*. The space Z defined in (2.9) endowed with the energy norm (2.10) is the natural function space to look for variational solutions to the deterministic linearized Grad-Shafranov equation (2.5). The variational formulation of the problem, as derived in [1, 25, 42], is that of finding $\psi \in Z$ such that for every test $\varphi \in Z$ it holds that:

$$\begin{aligned} &\int_{\Omega^-} \frac{1}{\mu x} \nabla \psi \cdot \nabla \varphi \, dA + \int_{\Gamma} \psi \, N \, \varphi \, d\Gamma \\ &+ \int_{\Gamma} \int_{\Gamma} \left(\psi(\boldsymbol{x}_1) - \psi(\boldsymbol{x}_2) \right) M(\boldsymbol{x}_1, \boldsymbol{x}_2) \left(\varphi(\boldsymbol{x}_1) - \varphi(\boldsymbol{x}_2) \right) \, d\Gamma(\boldsymbol{x}_1) \, d\Gamma(\boldsymbol{x}_2) = \int_{\Omega^-} F(\psi, x, \Omega_p) \, \varphi \, dA, \end{aligned} \tag{2.11}$$

where $\boldsymbol{x}_i = (x_i, y_i)$ and

$$
\begin{aligned}
N &:= \frac{1}{x}\left(\frac{1}{\delta_+} + \frac{1}{\delta_-} - \frac{1}{\rho_\Gamma}\right), \\
\delta_\pm &:= \sqrt{x^2 + (\rho_\Gamma \pm y)^2}, \\
M(\boldsymbol{x}_1, \boldsymbol{x}_2) &:= \frac{\kappa(\boldsymbol{x}_1, \boldsymbol{x}_2)}{2\pi(x_1 x_2)^{3/2}}\left(\frac{2-\kappa^2(\boldsymbol{x}_1, \boldsymbol{x}_2)}{2-2\kappa^2(\boldsymbol{x}_1, \boldsymbol{x}_2)} E(\kappa(\boldsymbol{x}_1, \boldsymbol{x}_2)) - K(\kappa(\boldsymbol{x}_1, \boldsymbol{x}_2))\right), \\
\kappa(\boldsymbol{x}_1, \boldsymbol{x}_2) &:= \sqrt{\frac{4x_1x_2}{(x_1+x_2)^2 + (y_1-y_2)^2}},
\end{aligned}
$$

and $E(\kappa(\boldsymbol{x}_1, \boldsymbol{x}_2))$ and $K(\kappa(\boldsymbol{x}_1, \boldsymbol{x}_2))$ are, respectively, complete elliptic integrals of first and second kind.

2.2 Incorporating uncertainty

The most obvious sources of uncertainty are the values of the currents $\{I_i\}$ flowing through the coils. These currents are subject to variations due to factors such as small oscillations in the power supply or temperature variations, and material impurities in the conducting wire that affect the conductivity. Even if the reactor has been built to very high engineering standards, the precise location of the coils within the reactor with respect to its projected location in the theoretical design will also be subject to a certain amount of uncertainty. Yet one more source of uncertainty is the value of the magnetic permeability, which can only be determined approximately through measurements, and under experimental conditions its value will most certainly differ from the one considered in theoretical computations.

2.2.1 Parameter space

In this work, we will consider that the uncertainty in the model (2.5a) is concentrated in the values of the currents I_i going through the external coils. As a result, the function ψ (and all quantities derived from it) are random variables. Obtaining a full description of their probability density functions might not be possible, but an approximate picture can be obtained by exploring the parameter space and computing sample approximations of its expectation and variance. We will model the array of currents as a d-dimensional random variable $\boldsymbol{\omega} := (\omega_1, \ldots, \omega_d)$, where d is the number of confinement coils in the reactor, and the k-th component of $\boldsymbol{\omega}$ is the current going through the k-th coil. We will consider that $\boldsymbol{\omega}$ is

uniformly distributed around a baseline vector $\boldsymbol{I} = (I_1, \ldots, I_d)$ corresponding to the desired current values in a deterministic model. We will often refer to $\boldsymbol{I}$ as either the *reference* or *unperturbed* currents. Letting $\tau > 0$ denote the size of the possible perturbation in the current values (relative to the components of $\boldsymbol{I}$), the vector $\boldsymbol{\omega}$ is then uniformly distributed over the d-dimensional parameter space

$$W := \prod_{k=1}^{d} \left[I_k - \tau |I_k|, I_k + \tau |I_k| \right]. \tag{2.12}$$

Since coils are independent of each other, the stochastic random currents $\{\omega_k\}_{k=1}^{d}$ are uncorrelated and the joint density function of $\boldsymbol{\omega}$ is given by $\pi(\boldsymbol{\omega}) = \prod_{k=1}^{d} \pi_k(\omega_k) = \prod_{k=1}^{d} \frac{1}{2\tau|I_k|}$. The equilibrium configuration determined by the solution to (2.5a) is then the random variable $\psi(x, y, \boldsymbol{\omega})$.

2.2.2 Bochner space

We now consider the stochasticity in the currents and allow the vector of currents to be a d-dimensional random variable $\boldsymbol{\omega}$ uniformly distributed over the parameter space W defined in (2.12). It is clear that in this case for any particular realization of the currents $\boldsymbol{\omega}$ we will obtain a *different* equilibrium configuration $\psi(\boldsymbol{\omega})$ that belongs to the Banach space Z defined in (2.9). Moreover, since for every $\boldsymbol{\omega} \in W$ the resulting equilibrium configuration has finite energy, it then holds for the expected value of the equilibrium that

$$\mathbb{E}\left(\|\psi\|_Z^2 \right) < \infty.$$

In mathematical terms, we say that the stochasticity of the currents transforms the solution ψ to (2.5) into a Banach space-valued random variable with finite expected energy. If $\boldsymbol{\omega}$ belongs to a complete and separable probability space $(W, \Sigma, \mathbb{P})$, the class of such random variables forms what is known as a *Bochner space* [13]. In our particular case, solutions to (2.5) are mappings from the parameter space W to the Banach space Z that, as functions of $\boldsymbol{\omega}$, belong to the Bochner space

$$L^2(W, \Sigma, \mathbb{P}; Z) := \{ u : W \to Z \mid u \text{ strongly measurable}, \ \|u\|_{L^2(W,Z)} < \infty \},$$

where the norm $\| \cdot \|_{L^2(W,Z)}$ is precisely defined in terms of the expected energy

$$\|u\|_{L^2(W,Z)} := \left(\int_W \|u(\cdot, \omega)\|_Z^2 \, d\mathbb{P}(\omega) \right)^{1/2} = \left(\mathbb{E}\left(\|u(\cdot, \omega)\|_Z^2 \right) \right)^{1/2}. \tag{2.13}$$

2.3 Objectives

Our objective is to quantify the impact of variations in current intensity by efficiently constructing an approximation to the expected value of the equilibrium configuration

$$\mathbb{E}\left[\psi(r,z,\boldsymbol{\omega})\right] = \int_W \psi(r,z,\boldsymbol{\omega})\pi(\boldsymbol{\omega})d\boldsymbol{\omega}, \tag{2.14}$$

and to compute some derived quantities from (2.14), such as the plasma boundary, the location of the x-points, etc.

Since the location and shape of the plasma boundary depend on the values of the coil currents, variations of these values could lead to contacts between the plasma and the wall or even loss of confinement. This fact translates into a possible non-smoothness of the mapping between coil currents and the solutions of (2.5) which may then cause techniques such as stochastic collocation to underperform. Moreover, the computational effort associated with cubature methods scales exponentially with the dimension of the parameter space, seriously limiting their feasibility for estimating (2.14). This leads to the use of Monte Carlo methods (Chapter 5.1), which are agnostic to both the smoothness of the mapping and the dimensionality of the problem [57], although they have a slow convergence rate (1/2) that tends to make them costly. This will be addressed through the use of a multi-level approach in Chapter 5.2. Both methods require solving the system for a large number of realizations of the parameter set, and sample statistics such as means, variances, and probability estimates are computed using these sample solutions. Given that computing each such solution involves solving a discrete version of the system (2.7), simulations using this approach become computationally expensive. As an alternative, we investigate the use of the *stochastic collocation method* [5, 67] to construct a surrogate approximation $\widehat{\psi}_h$ for the discrete solution ψ_h. This surrogate is constructed based on a collection of *sparse grids* in the parameter space. We then perform sampling using this surrogate, demonstrating its ability to significantly reduce the computational burden without compromising the accuracy of the results.

Chapter 3: Sparse grid stochastic collocation

To reduce the costs associated with direct computation of the parametrized solution, surrogate evaluations are frequently utilized as a more efficient alternative. The construction of a surrogate typically entails developing an interpolating function, which is realized by solving the discrete system at specific parameter values and then interpolating between these solutions. In multidimensional settings, the set of parameter nodes can be chosen as a full tensor grid, constituted by taking the Cartesian product of one-dimensional grids. However, as the dimensionality increases, the consequent exponential growth in the number of grid points leads to the so-called "curse of dimensionality", rendering full tensor grids computationally unwieldy. In response to this challenge, the *sparse grid stochastic collocation* approach [5, 67, 74] offers a practical solution. It introduces sparse grids, a substantially more manageable subset of full tensor grid nodes, constructed via Smolyak's algorithm [67]. Surrogates built upon sparse grids facilitate efficient sampling and mitigate the trade-off between computational efficiency and the curse of dimensionality, making them a robust framework for surrogate model construction. This chapter presents an overview of the sparse grid stochastic collocation approach, following [50].

3.1 Stochastic collocation

We will describe the method in terms of a generic solution, u, to a PDE involving stochastic parameters $\boldsymbol{\omega}$. We need to develop a method that provides an accurate approximation of u that maps from a parameter space $[0,1]$ to a physical space contained in $\mathbb{R}$. Let $X = \{x_j \in [0,1], 1 \leq j \leq p\}$ be a set of p prescribed nodes in the parameter space. The SC method constructs an approximation of u by evaluating deterministic realizations $u(X,\cdot)$ at these fixed grids and employs a collection of interpolation functions $\{\phi_i\}_{i=1}^{p}$ satisfying

$$\phi_i(x_j) = \delta_{ij}, \quad \text{for } 1 \leq i, j \leq p,$$

where

$$\delta_{ij} = \begin{cases} 1, & \text{if } i = j, \\ 0, & \text{otherwise.} \end{cases}$$

By introducing the one-dimensional interpolation operator I_X associated with the node set X of size p, we express the interpolation formula for u as

$$\mathrm{I}_X[u] = \sum_{j=1}^{p} u(x_j, \cdot)\, \phi_j \,.$$

We then generalize this concept to construct a d-dimensional multivariate interpolant defined on the hypercube $[0,1]^d$. Using a tensor product construction, we define the interpolation operator $\mathcal{I}$ for u as

$$\begin{aligned} \mathcal{I}[u] = \left(\mathrm{I}_{X_1} \otimes \cdots \otimes \mathrm{I}_{X_d}\right)[u] &:= \sum_{j_1=1}^{p_1} \cdots \sum_{j_d=1}^{p_d} u\left(x_{j_1}^{(1)}, \ldots, x_{j_d}^{(d)}, \cdot\right) \left(\phi_{j_1}^{(1)} \otimes \cdots \otimes \phi_{j_d}^{(d)}\right), \\ &= \sum_{j=1}^{P} u(\mathbf{x}_j, \cdot)\, \Phi_j \,, \end{aligned}$$

where X_k represents the set of nodes with cardinality p_k along the k-th component, and the subscript k indicates the corresponding component in the d-dimensional space. The interpolatory functions $\{\Phi_j\}_{j=1}^{P}$ exhibit a tensor product structure. The set of support nodes is referred to as the *full grid*, and its size is given by $P = \Pi_{k=1}^{d} p_k$. As the dimension of the parameter space d increases, the size of the full grid grows exponentially, leading to the phenomenon known as the *curse of dimensionality*.

3.2 Smolyak operator and sparse grids

The sparse grid stochastic collocation method offers an antidote to mitigate the challenges of the curse of dimensionality in the full tensor product while maintaining accuracy comparable to the full grid. Consider a d-dimensional multivariate index $\boldsymbol{i} = (i_1, \ldots, i_d) \in \mathbb{N}_+^d$ with $|\boldsymbol{i}| = i_1 + \cdots + i_d$. At the core of this approach is the Smolyak operator $\mathcal{S}_{q,d}$, which is defined for an integer $q \geq 0$ as

$$\mathcal{S}_{q,d} := \sum_{|\boldsymbol{i}| \leq q+d} \left(\Delta^{i_1} \otimes \cdots \otimes \Delta^{i_d}\right), \tag{3.1}$$

where $\Delta^{i_k} = \mathrm{I}_{X_k} - \mathrm{I}_{X_{k-1}}$ and $\mathcal{I}^0 = 0$. An alternative expression for $\mathcal{S}_{q,d}$ is given by

$$\mathcal{S}_{q,d} = \sum_{q+1 \leq |\boldsymbol{i}| \leq q+d} (-1)^{q+d-|\boldsymbol{i}|} \binom{d-1}{q+d-|\boldsymbol{i}|} \cdot (\mathrm{I}_{X_1} \otimes \cdots \otimes \mathrm{I}_{X_d}) \,. \tag{3.2}$$

Equation (3.2) reveals that the sparse grid operator retains the tensor product construction but it utilizes a sparse version of the interpolation formula, where the function of interest is evaluated only at a selected set of nodes instead of the complete tensor grid. Let $\xi(i_k)$ be an increasing function of the index i_k that determines the number of support nodes in the k-th component of $\boldsymbol{i}$. To compute $\mathcal{S}_{q,d}(u)$, the sparse grid interpolant reproduces the function u at all nodes in the sparse grid. The associated collection of points in $[0,1]^d$ is denoted by $\mathbf{X}^{\boldsymbol{i}} := (X^{i_1} \times \cdots \times X^{i_d}) \subset [0,1]^d$, where X^{i_k} represents a collection of nodes in the interval $[0,1]$ with a size $\xi(i_k)$ for $1 \leq k \leq d$. The *sparse grid*, denoted as $H_{q,d}$, is defined as

$$H_{q,d} := \bigcup_{|\boldsymbol{i}| \leq q+d} \mathbf{X}^{\boldsymbol{i}} = \bigcup_{q+1 \leq |\boldsymbol{i}| \leq q+d} \left(X^{i_1} \times \cdots \times X^{i_d}\right), \tag{3.3}$$

Insight into the structure of $H_{q,d}$ can be obtained from Figure 3.1, which shows on the left are examples of the individual contributions $\bigcup_{|\boldsymbol{i}|=q+d} \mathbf{X}^{\boldsymbol{i}}$ to $H_{q,d}$ for $d = 2$, and $q = 0$ through 3. In this image, the individual indices come from the values $i_p = 1$ through 4 and $\xi(1) = 1$, $\xi(i_p) = 2^{i_p - 1} + 1$ for $i_p > 1$. We observe that when the value of q is small or d is large, the sparse grid $H_{q,d}$ exhibits a significant reduction in size compared to the full grid [5, 50, 67]. For instance, in our problem where $d = 12$, representing the number of coils in the reactor model, the sparse grids at level 3 and level 4 have sizes of $2,649$ and $17,265$ respectively. In contrast, the corresponding full grids have much larger sizes of 2.8×10^{11} and 5.8×10^{14}. The grids comprising $H_{q,d}$ can take either a nested or non-nested form. In the nested case, q represents the *level* of the sparse grids, while for non-nested grids, the level is denoted by $q + d$. To minimize the cost of surrogate construction, it is preferable to use nested one-dimensional nodal sets in (3.1) and nested sparse grids in (3.3). For nested sparse grids, we have the inclusion and decomposition relationships as

$$X^i \subset X^{i+1}, \qquad H_{q,d} = \bigcup_{|\boldsymbol{i}|=q+d} \left(X^{i_1} \times \cdots \times X^{i_d}\right) = H_{q-1,d} \cup \Delta H_{q,d}. \tag{3.4}$$

where $\Delta H_{q,d} := H_{q,d} \backslash H_{q-1,d}$ represents the nodes $\mathbf{X}^{\boldsymbol{i}}$ with $|\boldsymbol{i}| = q+d$. In our experiments, we use the nested Clenshaw-Curtis nodes as the interpolation points. These nodes are nested and

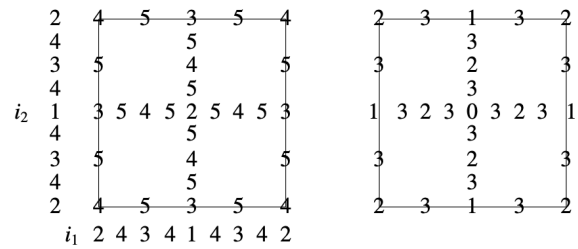

Figure 3.1: Left: multi-index values (i_1, i_2) and contributions to $H_{q,d}$ from $\bigcup_{|\boldsymbol{i}|=q} \mathbf{X}^{\boldsymbol{i}}$, for $d = 2$ and $q = i_1 + i_2 = 2$ through 5. Right: Highest grid level associated with each node.

defined by $x_j^k = -\cos\left(\frac{(j-1)\pi}{p_k - 1}\right)$ for $1 \leq j \leq p_k$ and are chosen as the extrema of Chebyshev polynomials of degree $p_k = \xi(i_k)$ in the $k-$th component. The basis functions are then built using these p_k interpolation points and have a polynomial degree of $p_k - 1$. In addition, the Clenshaw-Curtis abscissa requires $x_1^k = 0$ if $\xi(i_k) = 1$ and $\xi(1) = 1, \xi(i_k) = 2^{i_k - 1} + 1$ for $i_k > 1$.

To construct the interpolant on $H_{q,d}$ using the relation (3.4), we decompose the Smolyak interpolation operator for u as a sum of the interpolant on $H_{q-1,d}$ and a correction term defined on nodes in $\Delta H_{q,d}$. This can be expressed as

$$\mathcal{S}_{q,d}(u) = \sum_{|\boldsymbol{i}| \leq q+d-1} \left(\Delta^{i_1} \otimes \cdots \otimes \Delta^{i_d}\right)(u) + \sum_{|\boldsymbol{i}| = q+d} \left(\Delta^{i_1} \otimes \cdots \otimes \Delta^{i_d}\right)(u) \;=\; \mathcal{S}_{q-1,d}(u) + \Delta\mathcal{S}_{q,d}(u).$$

This Smolyak construction uses the relation (3.4) by building the interpolant on $H_{q,d}$ as the sum of the interpolant on $H_{q-1,d}$ (i.e., $\mathcal{S}_{q-1,d}(u)$) and a correction $\Delta\mathcal{S}_{q,d}(u)$ that interpolates $u(\mathbf{x}) - \mathcal{S}_{q-1,d}(u)(\mathbf{x})$ at the nodes in $\Delta H_{q,d}$ and is 0 on $H_{q-1,d}$. We elaborate on this using the left image of Figure 3.1, where $q = 5$ and $d = 2$. In the figure, the square represents the unit square and the numbers indicate the location of interpolation nodes. The Smolyak interpolant $\mathcal{S}_{4,2}(u)$ consists of polynomials that interpolate u at all the nodes in the image with labels 4 or smaller ($H_{4,2}$), and $\Delta\mathcal{S}_{5,2}(u)$ is a polynomial that has the value 0 at all these nodes and value $u(\mathbf{x}) - \mathcal{S}_{4,2}(u)(\mathbf{x})$ at all nodes with label 5 ($\Delta H_{5,2}$). Evaluation of $\mathcal{S}_{5,2}(u)$ at any other point $\mathbf{x}$ in the domain is done recursively as $\mathcal{S}_{5,2}(u)(\mathbf{x}) = \mathcal{S}_{4,2}(u)(\mathbf{x}) + \Delta\mathcal{S}_{5,2}(u)(\mathbf{x})$. There are 16 terms in the sum contributing to $\Delta\mathcal{S}_{5,2}(u)(\mathbf{x})$, one for each of the nodes labeled 5. Consider for example the bottom left of these nodes situated on the left boundary of the domain, which we refer to here as $\mathbf{x}_* = (x_*, y_*)$. The summand associated with $\mathbf{x}_*$ contributing to $\Delta\mathcal{S}_{5,2}(u)$ has the form of a product $(u(\mathbf{x}_*) - \mathcal{S}_{4,2}(\mathbf{x}_*))\phi_1(x)\phi_2(y)$, where ϕ_1 is a Lagrange polynomial of degree 2 with value 1 at x_* and 0 at the two other nodes for

which $y = y_*$, and similarly ϕ_2 is a degree 4 polynomial with value 1 at y_* and 0 at the other nodes along the vertical line. Use of this hierarchical structure allows the interpolant to be implemented using relatively simple data structures and bookkeeping while also enabling the re-use of all the information present in $H_{q,d}$ if a higher order interpolant $H_{q+i,d+j}$ is needed at a later time, thus speeding up the off-line component of the process; see [50] for details.

3.3 Error analysis

In sparse grid stochastic collocation, when $p_k = \xi(i_k)$ interpolation points are used in a component, the basis functions are built from interpolating polynomials of degree p_k, and interpolation points consist of the extrema of the Chebyshev polynomials of degree p_k, $x_j = \cos\left(\frac{(j-1)\pi}{p_k-1}\right)$, $1 \leq j \leq p_k$. Let $D^\beta = \dfrac{\partial^{|\beta|}}{\partial_{x_1}^{\beta_1} \cdots \partial_{x_d}^{\beta_d}}$ denote the mixed derivatives, where $\beta = (\beta_1, \ldots, \beta_d)$ with $\beta_i \in \mathbb{N}$, and $|\beta| = \sum \beta_i$. With this strategy, the interpolation error is bounded as [50, Theorem 8]

$$\left\| u - \mathcal{S}_{q,d}(u) \right\|_\infty = O\left(P^{-\rho} |\log P|^{(\rho+2)(d-1)+1} \right) \tag{3.5}$$

where P is the number of nodes in the sparse grid and ρ, a measure of smoothness of u, is the largest integer such that $D^\beta u$ is continuous if $\beta_k \leq \rho$ for all $k = 1, \ldots, d$. This bound shows that accuracy is good for large P but there is a logarithmic factor that may limit its utility, as the number of nodes must be at least $P \sim e^{((\rho+2)(d-1)+1)/\rho}$ before the logarithmic growth is controlled. This makes for a long pre-asymptotic regime that becomes more acute as the number of parameter dimensions d grows—an example of this can be seen in the right panel of Figure 6.1. Notably, the error of the piecewise multilinear approximation on a full tensor grid with $\widetilde{P}$ grid nodes follows $\mathcal{O}(\widetilde{P}^{-2/d})$ [49], demonstrating a considerably slower decay compared to the sparse grid method as d increases.

Furthermore, drawing upon insights from [60, 68], consider $u \in C^0(W, Z)$, where W and Z are defined in (2.12) and (2.9) respectively. Let the single-dimensional interval along the k-th dimension be

$$W_k = \left[I_k - \tau \left| I_k \right|, I_k + \tau \left| I_k \right| \right],$$

with its corresponding complementary multi-dimensional parameter space being

$$W_k^c = \prod_{i=1, i \neq k}^{d} W_i.$$

Given an arbitrary element ω_k^c in W_k^c, and for each ω_k in W_k, we assume the function $u(\omega_k, \omega_k^c, \cdot) : W_k \to C^0(W_k^c; Z)$ admits an analytic extension $u(z, \omega_k^c, \cdot)$ in the region

$$W_k^* := \{z \in \mathbb{C} : \mathrm{dist}(z, W_k) \leq \iota_k \ \text{ for some } \iota_k > 0\}$$

of the complex plane, then the interpolation error exhibits an algebraic convergence rate

$$\left\| u - \mathcal{S}_{q,d}(u) \right\|_\infty = CP^{-\mu_1}, \tag{3.6}$$

where P denotes the sparse grid node count, C is a constant dependent on dimension d and analytic extension proximity to the interval W_k, and $\mu_1 = \sigma/(1 + \log(2d))$, with σ relating to the function's analytic extension in the complex plane.

Compared to the regularity assumption in (3.5), the assumption for (3.6) is stronger in the sense that the solution u with respect to the random variable $\boldsymbol{\omega}$ can be analytically extended into the complex plane region by varying with one dimension of the random variable while keeping the other dimensions fixed. This enhancement allows for a tighter interpolation error bound compared to the regularity assumption in (3.5).

Chapter 4: Surrogate construction

4.1 Introduction

Surrogate computations consist of two processes: offline and online. The offline phase focuses on developing an accurate and reliable surrogate model prior to its application in sampling or predictive tasks. Upon completion of this phase, the algorithm transitions to the online phase, where the surrogate is employed for efficient evaluations. The online phase enables rapid computations or simulations by leveraging the surrogate model instead of directly solving the computationally intensive original problem. This separation into offline preparation and online application ensures that surrogate models can serve as efficient tools in various sampling methods and predictive analyses, despite the initial offline costs. Conversely, direct approaches may seem cost-effective at the outset but typically demand greater resources during the online application.

This chapter addresses the construction of surrogate models offline, examining two main strategies: *single-level sparse grid stochastic collocation*, with both single and multi-spatial level options, and *multilevel sparse grid stochastic collocation*, featuring a multilevel approach in both parameter and physical spaces. We will assess the costs involved in building these three surrogate models.

Let u_h represent a discretization of u using a spatial mesh size h and M be the number of spatial grid nodes. Note that $M = M(h)$. Consider a set of sparse grid nodes $\{\boldsymbol{\omega}_{\mathrm{s}}^{(i)}\}_{1\leq i\leq P}$, which give rise to a sample of P realizations $u^{(i)} = u(\boldsymbol{\omega}_{\mathrm{s}}^{(i)})$, and discrete realizations $u_h^{(i)} = u_h(\boldsymbol{\omega}_{\mathrm{s}}^{(i)})$. The functions discussed in this Chapter belong to the space defined in (2.9) as

$$Z := \left\{ \psi : \Omega^- \to \mathbb{R} \;\middle|\; \int_{\Omega^-} \psi^2 x \, dx \, dy < \infty \, ; \int_{\Omega^-} \frac{|\nabla \psi|^2}{x} \, dx \, dy < \infty \, ; \, \psi(0, y) = 0 \right\} \cap \mathcal{C}^0 \left(\overline{\Omega^-} \right).$$

with norm $\|\cdot\|_Z$ defined in (2.10) as

$$\|\psi\|_Z := \left(\int_{\Omega^-} \frac{|\nabla\psi|^2}{x} dxdy\right)^{1/2}$$

We begin by assuming that the sample-wise discretization error satisfies

$$\|u^{(i)} - u_h^{(i)}\|_Z \leq C_1^{(i)} M^{-\alpha_1}, \tag{4.1}$$

where α_1 is the order of the sample-wise discretization error and the constant $C_1^{(i)}$ depends only on the problem geometry.

4.2 Single-level sparse grid stochastic collocation and single-level spatial discretization surrogate

In this section, we explore the surrogate constructed via the single-level sparse grid stochastic collocation and single-level spatial discretization. Let $\widehat{u}_{q,h}$ denote the surrogate obtained from the Smolyak interpolant $\mathcal{S}_{q,d}(u_h)$ as outlined in (3.2). Denote P the number of sparse grid nodes, which is dependent on q. The process of constructing the surrogate entails the evaluation of the function at a pre-defined set of sparse grid nodes $\{\boldsymbol{\omega}_{\mathrm{s}}^{(i)}\}_{1\leq i\leq P}$, with the evaluations represented as $\widehat{u}_{q,h}^{(i)} = \widehat{u}_{q,h}(\boldsymbol{\omega}_{\mathrm{s}}^{(i)})$. We assume that the sample-wise interpolation error satisfies

$$\|u_h^{(i)} - \widehat{u}_{q,h}^{(i)}\|_Z \leq C_2^{(i)} P^{-\mu_1}, \tag{4.2}$$

where μ_1 is the order of the sample-wise interpolation error in (3.6) and the constant $C_2^{(i)}$ relies on the particular values of $\boldsymbol{\omega}_{\mathrm{s}}^{(i)}$.

For this approach, consider the sparse grid level $q = q_s$ in (3.4), where 's' denotes the use of a single-level sparse grid. Let d represent the dimension of the parameter space W in (2.12). Its surrogate function $\widehat{u}_h^s$ is defined by applying the Smolyak interpolation formula (3.2) to u_h as

$$\widehat{u}_h^s := \mathcal{S}_{q_s,d}[u_h].$$

To evaluate the convergence of $\widehat{u}_h^s$ to the exact solution u, we quantify the difference between the two functions using the Bochner norm in (2.13). The total error is decomposed into two components – discretization error and interpolation error – via the triangle inequality, as

$$\|u - \widehat{u}_h^s\|_{L^2(W,Z)} \leq \underbrace{\|u - u_h\|_{L^2(W,Z)}}_{\text{discretization error}} + \underbrace{\|u_h - \widehat{u}_h^s\|_{L^2(W,Z)}}_{\text{interpolation error}}. \tag{4.3}$$

Drawing upon the sample-wise error estimates presented in (4.1) and (4.2), along with the total error in (4.3), we can determine the requisite numbers of spatial and sparse grid nodes in terms of a user-specified tolerance ϵ. Let us introduce $\theta_1, \theta_2 \in (0, 1)$ as parameters to guide the allocation of this tolerance between discretization and interpolation errors respectively. These parameters, known as the *splitting ratios*, dictate the maximum allowable contribution of each type of error to the total error budget, aiming to achieve an overall model estimation accuracy of $(\theta_1 + \theta_2)\epsilon$ for the function u using the surrogate model. For the discussion of this chapter, we assume that $\theta_1 + \theta_2 < 1$. The strategy for selecting suitable values for θ_1 and θ_2 will be discussed in Chapter 5.1.2 and Chapter 5.2.3. The splitting of the total error involves ensuring that

$$\|u - u_h\|_{L^2(W,Z)} \leq C_m M^{-\alpha_1} \leq \theta_1 \epsilon, \quad \text{and} \quad \|u_h - \widehat{u}_h^s\|_{L^2(W,Z)} \leq C_p P^{-\mu_1} \leq \theta_2 \epsilon, \tag{4.4}$$

where C_m is a positive constant dependent only on the geometry of the problem, while C_p, as elaborated in (3.6), depends on the parameter space, with both being independent of random variables. From (4.4), we determine the numbers of spatial and sparse grid node M and P as

$$M \geq \left(\frac{\theta_1 \epsilon}{C_m}\right)^{-\frac{1}{\alpha_1}}, \qquad P \geq \left(\frac{\theta_2 \epsilon}{C_p}\right)^{-\frac{1}{\mu_1}}. \tag{4.5}$$

The estimates in (4.5) suggest that the number of sparse grid nodes P is a function of the number of spatial grid nodes M, specifically through $P = P(M) = \left(\frac{\theta_2 C_m}{\theta_1 C_p}\right)^{-1/\mu_1} M^{\alpha_1/\mu_1}$. With this optimal P, the total error, as described in (4.3), should follow $\left(1 + \frac{\theta_2}{\theta_1}\right) C_m M^{-\alpha_1}$. However, achieving the optimal number of spatial and sparse grid nodes M and P as specified in (4.5) presents challenges in practical implementations, as these optimal numbers are often rational instead of integers. Even when rounding them up using the ceiling function to the smallest integers larger than the optimal values, practical implementations still fail to achieve these precise numbers. This discrepancy stems from several factors: the actual count of spatial grid nodes depends on the available mesh set; while the count of sparse grid nodes, as indicated by (3.4), is influenced by the chosen sparse grid level and the dimension of the parameter space. Consequently, the practical count of sparse grid nodes often deviates, typically exceeding, from the ideal number P specified in (4.5). Such discrepancies between practical and theoretical values of both M and P can lead to a misalignment, potentially affecting the total error in the estimation process.

To estimate the computational effort in building the single-level sparse grid stochastic collocation and single-level spatial discretization surrogate, considering that the computational cost, $C(u_h^{(i)})$, of obtaining a sample may vary with the specific realization $\boldsymbol{\omega}^{(i)}$ for our nonlinear problem, we assume the *average* sample cost is independent of the realizations and follows $C = \mathcal{O}(M^\gamma)$ for some $\gamma > 0$ depending on the solver. Since the sparse grid nodes are special sample points in the parameter space, the overall cost associated with constructing this surrogate, measured in terms of ϵ, is given by

$$\mathscr{W}_{\text{SL-SL}}^{\text{off}} \leq (P+1)C \leq \mathcal{O}\left(\epsilon^{-\frac{1}{\mu_1}}\right) \cdot \mathcal{O}\left(\epsilon^{-\frac{\gamma}{\alpha_1}}\right) = \mathcal{O}\left(\epsilon^{-\frac{1}{\mu_1}-\frac{\gamma}{\alpha_1}}\right), \tag{4.6}$$

where the subscript 'SL-SL' denotes the surrogate built with a single-level sparse grid and a single-level spatial grid, the superscript 'off' represents an offline process, ϵ represents the desired accuracy level, μ_1 is the convergence rate of the interpolation error, γ is the growth rate characterizing the average cost per sample and α_1 is the convergence rate of the discretization error.

4.3 Single-level sparse grid stochastic collocation and multilevel spatial discretization surrogate

The surrogate built with single-level sparse grid stochastic collocation and multilevel spatial discretization comprises a sequence of surrogates built with single-level sparse grid stochastic collocation and single-level spatial discretization on a set of uniformly refined spatial meshes. We describe this set of multilevel spatial meshes by $\{\mathcal{T}_k\}_{0\leq k\leq \ell}$ with *level* k up to level ℓ and increasing grid node counts $\{M_k\}_{0\leq k\leq \ell}$ that satisfies $M_k = sM_{k-1}$ for $s > 1$. Let u_k be the discrete approximation of u on $\mathcal{T}_k$, q_s and P_{q_s} denote the level and the number of sparse grid nodes. The single-level sparse grid stochastic collocation and multilevel spatial discretization surrogate (denoted as $\widehat{u}_\ell^s$, where the subscript s denote the single-level sparse grid) can be represented as a telescoping sum of surrogate corrections between consecutive spatial mesh levels, written as

$$\begin{aligned}\widehat{u}_\ell^s := \mathcal{S}_{q_s,d}[u_\ell] &= \sum_{k=0}^{\ell} \mathcal{S}_{q_s,d}\left[u_k - u_{k-1}\right] = \sum_{k=0}^{\ell} \mathcal{S}_{q_s,d}\left[Y_k\right], \\ &= \sum_{k=0}^{\ell} \left(\widehat{u}_{q_s,k} - \widehat{u}_{q_s,k-1}\right) = \sum_{k=0}^{\ell} \widehat{Y}_{q_s,k},\end{aligned} \tag{4.7}$$

where $Y_k := u_k - u_{k-1}$ for $k \geq 1$ and $Y_0 := u_0$ represent the correction of the discretized approximation u_0 on two consecutive spatial mesh levels; $\widehat{Y}_{q_s,k} := \widehat{u}_{q_s,k} - \widehat{u}_{q_s,k-1}$ for $k \geq 1$ and $\widehat{Y}_{q_s,0} := \widehat{u}_{q_s,0}$ denote the surrogate correction with the same sparse grid nodes $\{\boldsymbol{\omega}_{\mathrm{s}}^{(i)}\}_{1 \leq i \leq P_{q_s}}$ on spatial meshes $\mathcal{T}_{k-1}$ and $\mathcal{T}_k$. As discussed in [19], we can obtain the realization of surrogate $\widehat{u}_{q_s,k-1}$ on the coarser grid from $\widehat{u}_{q_s,k}$ on the finer grid using interpolation, namely $\widehat{u}_{q_s,k-1} = \mathcal{I}(\mathcal{S}_{q_s,d}[u_k])$, where $\mathcal{I}$ denotes a simple Lagrangian interpolation operator. Note that we do not consider $\widehat{u}_{q_s,k-1} = \mathcal{S}_{q_s,d}[\mathcal{I}(u_k)]$, as the commutativity between the Lagrangian interpolation operator (which constructs the approximation function on the fine spatial grid in physical space) and the Smolyak $\mathcal{S}_{q_s,d}$ operator (which constructs a surrogate based on sparse grids in parameter space) cannot be generally guaranteed.

As described in [68], an assumption regarding the sample-wise error of the surrogate correction with a general sparse grid level q and the number of sparse grid nodes P_q is proposed as follows

$$\left\| Y_k^{(i)} - \widehat{Y}_{q,k}^{(i)} \right\|_Z \leq C_3^{(i)} M_k^{-\alpha_2} P_q^{-\mu_1} \quad \text{for} \quad k \geq 0. \tag{4.8}$$

In particular, we will demonstrate that for the single-level sparse grid surrogate, $q = q(\ell)$, and for the multilevel sparse grid surrogate, $q = q(k)$. Based on this assumption, we will establish the convergence of spatial multilevel surrogate functions with both single-level and multilevel in the sparse grids.

The convergence of this single-level sparse grid stochastic collocation and multilevel spatial discretization surrogate follows a similar error splitting as single-level sparse grid stochastic collocation and single-level spatial discretization. Particularly, for the interpolation error of this surrogate, we apply assumption (4.8) to the correction terms resulting from the telescoping sum of the multilevel representation. This yields the following upper bound for the

total error

$$
\begin{aligned}
\|u-\widehat{u}_\ell^s\|_{L^2(W,Z)} &\le \|u-u_\ell\|_{L^2(W,Z)} + \|u_\ell-\widehat{u}_\ell^s\|_{L^2(W,Z)}, \\
&\le \|u-u_\ell\|_{L^2(W,Z)} + \left\|\sum_{k=0}^{\ell}\left(Y_k-\widehat{Y}_{q_s,k}\right)\right\|_{L^2(W,Z)}, \\
&\le \|u-u_\ell\|_{L^2(W,Z)} + \sum_{k=0}^{\ell}\left\|Y_k-\widehat{Y}_{q_s,k}\right\|_{L^2(W,Z)}, \\
&\le C_m M_\ell^{-\alpha_1} + C_p P_{q_s}^{-\mu_1}\sum_{k=0}^{\ell} M_k^{-\alpha_2},
\end{aligned}
\tag{4.9}
$$

where C_m and C_p are positive constants independent of the random variables. To ensure that the discretization error falls below the tolerance $\theta_1\epsilon$, we estimate M_ℓ and the required spatial grid level ℓ as

$$
M_\ell \ge \left(\frac{\theta_1\epsilon}{C_m}\right)^{-\frac{1}{\alpha_1}}, \quad \ell = \left\lceil \frac{1}{\alpha_1}\log_s(C'\epsilon^{-1})\right\rceil, \tag{4.10}
$$

where $C' = C_m/(\theta_1 M_0^{\alpha_1})$ and $\lceil\cdot\rceil$ denotes the ceiling function, which returns the smallest integer greater than or equal to its argument. Notably, the inequality for ℓ due to the ceiling function and the complexity for $\ell+1$ will be extensively used in subsequent analysis and proofs in the Appendix

$$
\ell \le \frac{1}{\alpha_1}\log_s(C'\epsilon^{-1}) + 1, \quad \text{and} \quad \ell+1 = \mathcal{O}\left(|\log\epsilon|\right). \tag{4.11}
$$

To estimate the sparse grid node count P_{q_s}, we use assumption (4.8) to estimate an upper bound for the interpolation error, ensuring it remains below the tolerance $\theta_2\epsilon$. This is expressed as

$$
\|u_\ell-\widehat{u}_\ell^s\|_{L^2(W,Z)} \le C_p P_{q_s}^{-\mu_1}\sum_{k=0}^{\ell} M_k^{-\alpha_2} = C_p P_{q_s}^{-\mu_1} M_0^{-\alpha_2}\sum_{k=0}^{\ell} s^{-\alpha_2 k} \le P_{q_s}^{-\mu_1}\frac{C_p M_0^{-\alpha_2}}{1-s^{-\alpha_2}} \le \theta_2\epsilon.
$$

This inequality leads to the optimal number of sparse grid nodes

$$
P_{q_s} \ge \left(\frac{\left(1-s^{-\alpha_2}\right)\theta_2\epsilon}{C_p M_0^{-\alpha_2}}\right)^{-\frac{1}{\mu_1}}. \tag{4.12}
$$

Estimates in (4.10) and (4.12) indicate that $P_{q_s} = P_{q_s}(M_\ell) \ge \left(\frac{\theta_2 C_m(1-s^{-\alpha_2})}{\theta_1 C_p M_0^{-\alpha_2}}\right)^{-1/\mu_1} M_\ell^{\alpha_1/\mu_1}$ and the total error $\|u-\widehat{u}_\ell^s\|_{L^2(W,Z)}$ behaves as $\left(1+\frac{\theta_2}{\theta_1}\right)C_m M_\ell^{-\alpha_1}$.

The cost associated with constructing the single-level sparse grid stochastic collocation and multilevel spatial discretization surrogate hinges upon several factors: the number of sparse grid nodes on each spatial mesh, the count of spatial grid levels, and the average cost per sample using direct computation. Assuming $C_k = \mathcal{O}(M_k^\gamma)$ is the average cost per sample, using (4.11), the sum of the cost per sample across ℓ spatial levels is

$$\sum_{k=0}^{\ell} C_k = M_0^\gamma \sum_{k=0}^{\ell} s^{\gamma k} < M_0^\gamma \frac{s^{\gamma \ell}}{1-s^{-\gamma}} < M_0^\gamma \frac{s^{\frac{\gamma}{\alpha_1}\log_s\left(C'\epsilon^{-1}\right)+\gamma}}{1-s^{-\gamma}} = \frac{\left(\frac{C_m}{\theta_1}\right)^{\frac{\gamma}{\alpha_1}} s^\gamma}{1-s^{-\gamma}} \epsilon^{-\frac{\gamma}{\alpha_1}} = \mathcal{O}\left(\epsilon^{-\frac{\gamma}{\alpha_1}}\right),$$

together with (4.12), we derive the total cost of building this surrogate as

$$\mathscr{W}_{\text{SL-ML}}^{\text{off}} \leq (P_{q_s}+1)\sum_{k=0}^{\ell} C_k = \mathcal{O}\left(\epsilon^{-\frac{1}{\mu_1}-\frac{\gamma}{\alpha_1}}\right), \tag{4.13}$$

where the subscript 'SL-ML' denotes the surrogate built with a single-level sparse grid and a multilevel spatial discretization.

4.4 Multilevel in both parameter and spatial grids surrogate

Unlike the surrogate built with single-level sparse grids depending on the finest spatial grids $q = q(\ell)$, the multilevel (in both parameter and spatial grids) surrogate [68], comprises a series of single-level stochastic collocation and single-level spatial discretization surrogates built with varying levels of sparse grids q determined by the respective spatial mesh levels k, i.e. $q = q(k)$. Let the surrogate for multilevel in both parameter and spatial grids be denoted as $\widehat{u}_\ell^m$, where the superscript 'm' indicates the multilevel approach in the parameter space and subscript 'ℓ' represents the multilevel strategy in the physical space, this surrogate is defined as

$$\widehat{u}_\ell^m := \sum_{k=0}^{\ell} \mathcal{S}_{q,d}\left[u_k - u_{k-1}\right] = \sum_{k=0}^{\ell}\left(\widehat{u}_{q,k} - \widehat{u}_{q,k-1}\right) = \sum_{k=0}^{\ell} \widehat{Y}_{q,k}, \tag{4.14}$$

where $\widehat{Y}_{q,k} := \widehat{u}_{q,k} - \widehat{u}_{q,k-1}$ for $k \geq 1$ and $\widehat{Y}_{q,0} := \widehat{u}_{q,0}$ is the surrogate correction. Notably, the same sparse grids are used for $\widehat{u}_{q,k}$ and $\widehat{u}_{q,k-1}$ within the correction term. The level q depends on the fine spatial level k, resulting in $\widehat{Y}_{q,k} = \widehat{u}_{q(k),k} - \widehat{u}_{q(k),k-1}$, but not $\widehat{u}_{q(k),k} - \widehat{u}_{q(k-1),k-1}$. This consistency in the sparse grid nodes for two consecutive spatial grids is pivotal in upholding the convergence assurance outlined in assumption (4.8).

To evaluate the accuracy of this surrogate relative to the exact solution, we establish an upper bound for the total error using assumptions (4.1) and (4.8) as

$$\begin{aligned}\|u - \widehat{u}_\ell^m\|_{L^2(W,Z)} &\leq \|u - u_\ell\|_{L^2(W,Z)} + \|u_\ell - \widehat{u}_\ell^m\|_{L^2(W,Z)}, \\ &\leq C_m M_\ell^{-\alpha_1} + C_p \sum_{k=0}^{\ell} M_k^{-\alpha_2} P_q^{-\mu_1}.\end{aligned} \tag{4.15}$$

With the same tolerance and error decomposition ratios, the number of spatial grid node M_ℓ and the level ℓ for this surrogate are identical to those for the single-level stochastic collocation and multilevel spatial discretization surrogate in (4.10). However, since $q = q(k)$ for the multilevel approach in both parameter and spatial grids, the optimal number of sparse grid nodes P_q cannot be explicitly determined as in the single-level sparse grid scenario. To estimate P_q, we formulate an optimization problem that minimizes the surrogate construction cost while ensuring the interpolation error remains within the tolerance $\theta_2\epsilon$, which is written as

$$P_q = \arg\min \sum_{k=0}^{\ell} P_q C_k \quad \text{such that} \quad \|u_\ell - \widehat{u}_\ell^m\|_{L^2(W,Z)} \leq C_p \sum_{k=0}^{\ell} M_k^{-\alpha_2} P_q^{-\mu_1} \leq \theta_2\epsilon.$$

The Lagrangian for this optimization problem is

$$\mathcal{L} = \sum_{k=0}^{\ell} P_q C_k + \lambda \left(C_p \sum_{k=0}^{\ell} M_k^{-\alpha_2} P_q^{-\mu_1} - \theta_2\epsilon + t^2 \right), \tag{4.16}$$

where λ and t^2 are the Lagrange multiplier and slack variable [29, 62], respectively. By treating P_q as a continuous variable and applying the Karush-Kuhn-Tucker (KKT) conditions [8, 62], we obtain the optimal P_q as

$$P_q = \left(\frac{C_k}{\lambda \mu_1 C_p M_k^{-\alpha_2}} \right)^{-\frac{1}{\mu_1+1}} = \left(\frac{C_p}{\theta_2\epsilon} \right)^{\frac{1}{\mu_1}} \frac{M_k^{-\frac{\alpha_2}{\mu_1+1}}}{C_k^{\frac{1}{\mu_1+1}}} \left(\sum_{k=0}^{\ell} M_k^{-\frac{\alpha_2}{\mu_1+1}} C_k^{\frac{\mu_1}{\mu_1+1}} \right)^{\frac{1}{\mu_1}}. \tag{4.17}$$

The estimates in (4.10) and (4.17) imply that

$$P_q = P_q(M_k) \geq \left(\frac{\theta_1 C_p}{\theta_2 C_m M_\ell^{-\alpha_1}} \right)^{\frac{1}{\mu_1}} \frac{M_k^{-\frac{\alpha_2}{\mu_1+1}}}{C_k^{\frac{1}{\mu_1+1}}} \left(\sum_{k=0}^{\ell} M_k^{-\frac{\alpha_2}{\mu_1+1}} C_k^{\frac{\mu_1}{\mu_1+1}} \right)^{\frac{1}{\mu_1}}, \tag{4.18}$$

and the total error $\|u - \widehat{u}_\ell^m\|_{L^2(W,Z)}$ in (4.15) follows $\left(1 + \frac{\theta_2}{\theta_1}\right) C_m M_\ell^{-\alpha_1}$, converging at the same rate as both the discretization error.

Notice that the solutions P_q in (4.17) and (4.18) are not numbers of sparse grid points, but are solutions to the optimization problem (4.16). We use the ceiling function $\lceil P_q \rceil$, giving the smallest integer greater than or equal to P_q, to specify the number of sparse grid points. This procedure facilitates the estimation of the construction cost for the multilevel surrogate across both sparse and spatial grids, resulting in

$$\mathscr{W}^{\text{off}}_{\text{ML-ML}} = \sum_{k=0}^{\ell} P_q C_k \leq \left(\frac{C_p}{\theta_2 \epsilon}\right)^{\frac{1}{\mu_1}} \left(\sum_{k=0}^{\ell} M_k^{-\frac{\alpha_2}{\mu_1+1}} C_k^{\frac{\mu_1}{\mu_1+1}}\right)^{\frac{1}{\mu_1}+1} + \sum_{k=0}^{\ell} C_k, \tag{4.19}$$

where the subscript 'ML-ML' indicates the use of a multilevel surrogate in both parameter and spatial grids. In general, there is no explicit formula directly relating the number of sparse grid nodes P to the corresponding sparse grid level q_s. As indicated in [60, Lemma 4.8], the relationship between the total sparse grid node count P_q and the sparse grid level q of Clenshaw-Curtis abscissas should satisfy the condition

$$q \geq \frac{\log(P_q)}{1 + \log(2d)} - 1.$$

Using this relation, and the fact that $C_\ell = M_\ell^\gamma$ and $M_\ell = M_0 s^\ell$, P_q and q can be rewritten and estimated as

$$\begin{aligned} P_q &\geq \left\lceil \left(\frac{C_p}{\theta_2 \epsilon M_0^{\alpha_2}}\right)^{\frac{1}{\mu_1}} s^{-\frac{\alpha_2+\gamma}{\mu_1+1}k} \left(\sum_{k=0}^{\ell} s_1^k\right)^{\frac{1}{\mu_1}} \right\rceil, \\ q &\geq \left\lceil \frac{\left(-\frac{\alpha_2+\gamma}{\mu_1+1}\log s\right)k + \frac{1}{\mu_1}\log\left(\frac{C_p}{\theta_2 \epsilon M_0^{\alpha_2}} \sum_{k=0}^{\ell} s_1^k\right)}{1+\log(2d)} - 1 \right\rceil, \end{aligned}$$

where $s_1 = s^{-\frac{\alpha_2 - \gamma\mu_1}{\mu_1+1}}$. The observed inverse coupling between the sparse grid level q and spatial grid level k indicates less accurate interpolations on fine spatial approximations and more accurate interpolations on coarse grids.

The following theorem, derived from [68], presents a detailed analysis of the overall construction cost as defined in (4.19) for constructing the multilevel surrogate $\widehat{u}_\ell^m$ using the optimal P_q obtained from (4.17).

Theorem 1. *Suppose there exist positive constants $\alpha_1, \alpha_2, \mu_1, \gamma$ such that $\alpha_1 \geq \min(\alpha_2, \mu_1\gamma)$,*

(i) $\|u - u_k\|_{L^2(W,Z)} = \mathcal{O}(M_k^{-\alpha_1})$,

(ii) $\|Y_k - \widehat{Y}_{q,k}\|_{L^2(W,Z)} = \mathcal{O}\left(M_k^{-\alpha_2} P_q^{-\mu_1}\right)$,

(iii) $C_k = \mathcal{O}\left(M_k^{\gamma}\right)$.

Then for any $\epsilon < e^{-1}$ *small enough, there exists spatial grid level* ℓ *and sparse grid level* $q = q(k)$ *for* $0 \leq k \leq \ell$ *such that*

$$\|u - \widehat{u}_\ell^m\|_{L^2(W,Z)} < \epsilon,$$

and the total surrogate construction cost with bound

$$\mathscr{W}_{\text{ML-ML}}^{\text{off}} = \begin{cases} \mathcal{O}\left(\epsilon^{-\frac{1}{\mu_1}}\right), & \alpha_2 > \mu_1\gamma, \\ \mathcal{O}\left(\epsilon^{-\frac{1}{\mu_1}} \left|\log \epsilon\right|^{1+\frac{1}{\mu_1}}\right), & \alpha_2 = \mu_1\gamma, \\ \mathcal{O}\left(\epsilon^{-\frac{1}{\mu_1}-\frac{\gamma}{\alpha_1}+\frac{\alpha_2}{\alpha_1\mu_1}}\right), & 0 < \alpha_2 < \mu_1\gamma. \end{cases}$$

This theorem sheds light on the distribution of construction effort in the multilevel in both stochastic collocation and spatial discretization surrogate. When the ratio α_2/μ_1, indicating the convergence of surrogate corrections, exceeds the growth rate γ of the cost per sample, the primary construction effort is concentrated on coarse grids. Conversely, most work is directed towards fine grids. When α_2/μ_1 equals γ, the workload is evenly distributed across all levels.

Notably, when utilizing single-level sparse grid nodes, both single-level and multilevel spatial discretization surrogates exhibit the same construction cost complexity, differing only by a constant factor. However, the complexity of these two approaches is larger than that of the multilevel surrogate built across both sparse and spatial grids.

Chapter 5: Sampling methods

In this chapter, our primary focus will be on two online sampling methods aimed at approximating the expectation in (2.14) of interest: Monte Carlo Finite-Element and multilevel Monte Carlo Finite-Element sampling methods.

5.1 Monte Carlo finite element method

In this section, we embark on an exploration of the Monte Carlo Finite-Element sampling method, examining its application with both the direct solver and the utilization of surrogate approximations.

5.1.1 Sampling with the direct solver

To begin, we examine the Monte Carlo Finite-Element sampling method using the direct solver. We will describe the method in terms of a generic solution, u, to a PDE involving stochastic parameters and its finite element (FEM) approximation u_h, where h is the mesh parameter of the discretization. Let $\{\boldsymbol{\omega}^{(i)}\}_{1\leq i\leq N}$ be a set of N realizations of the random variable $\boldsymbol{\omega}$ giving rise to a sample of N realizations $u^{(i)} = u(\boldsymbol{\omega}^{(i)})$ and $u_h^{(i)} = u_h(\boldsymbol{\omega}^{(i)})$ of the exact solution and its finite element discretization. We will assume that all these functions belong to a functional space Z endowed with a norm $\|\cdot\|_Z$ (see Chapter 2.1.4 for details), and we will consider the standard FEM error estimate $\|u^{(i)} - u_h^{(i)}\|_Z \leq C^{(i)} h^p$, where p is the order of the FEM discretization and the constant $C^{(i)}$ depends only on the problem geometry and the particular values of $\boldsymbol{\omega}^{(i)}$. The Monte Carlo Finite-Element (MC-FE) estimator $A_{\mathrm{MC}}(u_h)$ for $\mathbb{E}(u)$ is defined as the sample mean

$$A_{\mathrm{MC}}(u_h) := \frac{1}{N}\sum_{i=1}^{N} u_h^{(i)}. \tag{5.1}$$

This estimator is easily shown to be unbiased and to satisfy $\mathbb{E}(A_{\mathrm{MC}}) = \mathbb{E}(u_h)$.

A quantity that serves as a foundation for examining the spatial and statistical accuracy of the MC-FE estimator is the *mean squared error* (MSE) defined as

$$\mathcal{E}^2_{A_{\mathrm{MC}}} := \mathbb{E}\left[\|\mathbb{E}(u) - A_{\mathrm{MC}}(u_h)\|^2_Z\right] = \|\mathbb{E}(u) - A_{\mathrm{MC}}(u_h)\|^2_{L^2(W,Z)}$$

It can be shown (see, for instance [4, Theorem 4.3]) that, for linear problems, the Monte Carlo estimator accurately approximates the expected value in the sense that

$$\mathcal{E}^2_{A_{\mathrm{MC}}} \leq K(N^{-1/2} + M^{-\alpha_1})^2,$$

where the constant $K > 0$ depends on the problem geometry and the expected values of the problem data, and integers N and M represent the sample size and number of spatial grid nodes respectively. The MSE can be decomposed into terms related to the bias and variance [4, Lemma 4.1] as

$$\begin{aligned}\mathcal{E}^2_{A_{\mathrm{MC}}} &= \|\mathbb{E}(u) - \mathbb{E}(u_h)\|^2_Z + \mathbb{E}\left[\|\mathbb{E}(u_h) - A_{\mathrm{MC}}(u_h)\|^2_Z\right] \\ &= \|\mathbb{E}(u) - \mathbb{E}(u_h)\|^2_Z + \frac{\mathbb{V}(u_h)}{N} = \mathcal{E}^2_{\mathrm{Bias}} + \mathcal{E}^2_{\mathrm{Stat}},\end{aligned}$$

where $\mathbb{V}(u) := \mathbb{E}[\|u - \mathbb{E}(u)\|^2_Z]$ and $\mathbb{V}(A_{\mathrm{MC}}(u)) = \mathbb{V}(u)/N$. The last two terms at the end of the expression above implicitly define the discretization error $\mathcal{E}_{\mathrm{Bias}}$ and the sampling (or statistical) error $\mathcal{E}_{\mathrm{Stat}}$ respectively.

To assess the disparity between datasets of models at various scales, we will consider a non-dimensional statistic – the *normalized mean square error* $\overline{\mathcal{E}}^2_{A_{\mathrm{MC}}}$. This metric normalizes the MSE by a factor of $\|\mathbb{E}(u)\|^2_{L^2(W,Z)}$ when the expectation of u is non-zero. The scaling factor becomes $\|\mathbb{E}(u)\|^2_Z$ since $\mathbb{E}(u)$ is independent of the random variables in the parameter space, rendering $\|\mathbb{E}(u)\|_Z$ a scalar, i.e.

$$\begin{aligned}\|\mathbb{E}(u)\|^2_{L^2(W,Z)} = \mathbb{E}\left[\|\mathbb{E}(u)\|^2_Z\right] &= \int_W \left\| \int_W u(x,y,\boldsymbol{\omega}) d\boldsymbol{\omega} \right\|^2_Z d\boldsymbol{\omega}, \\ &= \left\| \int_W u(x,y,\boldsymbol{\omega}) d\boldsymbol{\omega} \right\|^2_Z = \|\mathbb{E}(u)\|^2_Z.\end{aligned}$$

Since the exact random variable u is not available, we will approximate the relative mean squared error (nMSE) by

$$\overline{\mathcal{E}}^2_{A_{\mathrm{MC}}} \approx \frac{\|\mathbb{E}(u) - \mathbb{E}(u_h)\|_Z^2}{\|\mathbb{E}(u_h)\|_Z^2} + \frac{\mathbb{V}(u_h)}{N\,\|\mathbb{E}(u_h)\|_Z^2} = \overline{\mathcal{E}}^2_{\mathrm{Bias}} + \overline{\mathcal{E}}^2_{\mathrm{Stat}}, \tag{5.2}$$

where $\overline{\mathcal{E}}_{\mathrm{Bias}}$ and $\overline{\mathcal{E}}_{\mathrm{Stat}}$ are relative analogues to the discretization and statistical errors defined above. If the number of grid points for the FEM discretization is M then, in two dimensions, it is standard to assume that $\overline{\mathcal{E}}_{\mathrm{Bias}} = \mathcal{O}(h^{2\alpha_1})$.

Given a target tolerance ϵ, the contribution of the statistical and discretization errors towards the total nMSE can be controlled by requiring that

$$\overline{\mathcal{E}}^2_{\mathrm{Bias}} = C_m^2 M^{-2\alpha_1} \leq (1-\theta)\epsilon^2, \qquad \overline{\mathcal{E}}^2_{\mathrm{Stat}} = \frac{V_h}{N} \leq \theta\epsilon^2, \tag{5.3}$$

where C_m is a positive constant related only to the problem geometry and remains independent of random variables, $\theta \in (0,1)$ is known as the *splitting parameter*, and $V_h := \mathbb{V}(u_h) / \|\mathbb{E}(u_h)\|_Z^2$. This in turn allows us to estimate the sample size N and the number of grid points M required to achieve the desired tolerance as

$$M \geq \left(\frac{\sqrt{1-\theta}\epsilon}{C_m}\right)^{-\frac{1}{\alpha_1}}, \qquad N \geq \left\lceil \frac{V_h}{\theta\epsilon^2} \right\rceil. \tag{5.4}$$

If we assume that the average cost to obtain one sample (i.e. to solve (2.5) for one particular value of the coil-currents) is $C = \mathcal{O}(M^\gamma)$ for some $\gamma > 0$, then the total computational cost of the MC-FE estimator can be estimated as

$$\mathscr{W}^{\mathrm{on}}_{\mathrm{Direct}} = \mathcal{O}(NM^\gamma) = \mathcal{O}\left(\epsilon^{-2-\frac{\gamma}{\alpha_1}}\right), \tag{5.5}$$

where the superscript 'on' denotes the online sampling and the subscript 'Direct' represents the sampling conducted with the direct solver.

5.1.2 Sampling with the surrogate

To mitigate the sampling cost associated with direct computation in Monte Carlo sampling, we propose integrating the surrogate function constructed using the sparse grid stochastic collocation method introduced in Chapter 4.2 with the MC-FE sampling. As Monte Carlo simulations require a single level of the spatial grid, we employ a surrogate of the single-level stochastic collocation and single-level spatial discretization type ($\widehat{u}_h^s$), resulting in the surrogate-enhanced Monte Carlo Finite-Element approach.

Let $\widehat{u}_h^{(i)} := \widehat{u}_h^s(\boldsymbol{\omega}^{(i)})$ denote the realization of the i-th sample $\boldsymbol{\omega}^{(i)}$ for the single-level stochastic collocation and single-level spatial discretization surrogate $\widehat{u}_h^s$. The surrogate-based Monte Carlo Finite-Element estimator $A_{\mathrm{MC}}(\widehat{u}_h)$ for $\mathbb{E}(u)$ is defined as the sample mean

$$A_{\mathrm{MC}}\left(\widehat{u}_h\right) := \frac{1}{N}\sum_{i=1}^{N}\widehat{u}_h^{(i)}. \tag{5.6}$$

The mean square error for the surrogate-enhanced Monte Carlo is defined as

$$\mathcal{E}_{A_{\mathrm{MC}}}^2 := \mathbb{E}\left[\left\|\mathbb{E}(u) - A_{\mathrm{MC}}\left(\widehat{u}_h\right)\right\|_Z^2\right].$$

It can be decomposed into two terms: bias $\mathcal{E}_{\mathrm{Bias}}$ and variance $\mathcal{E}_{\mathrm{Stat}}$. The bias term $\mathcal{E}_{\mathrm{Bias}}$ can be further split into a sum of discretization error $\mathcal{E}_{\mathrm{DisErr}}$ and interpolation error $\mathcal{E}_{\mathrm{InterpErr}}$, which is given by

$$\begin{aligned}
\mathcal{E}_{A_{\mathrm{MC}}}^2 &= \mathbb{E}\left[\left\|\mathbb{E}(u) - \mathbb{E}\left(\widehat{u}_h\right)\right\|_Z^2\right] + \mathbb{E}\left[\left\|\mathbb{E}\left(\widehat{u}_h\right) - A_{\mathrm{MC}}\left(\widehat{u}_h\right)\right\|_Z^2\right], \\
&= \left\|\mathbb{E}(u) - \mathbb{E}\left(\widehat{u}_h\right)\right\|_Z^2 + \frac{\mathbb{V}\left(\widehat{u}_h\right)}{N} = \mathcal{E}_{\mathrm{Bias}}^2 + \mathcal{E}_{\mathrm{Stat}}^2, \\
&\leq \left(\left\|\mathbb{E}(u) - \mathbb{E}(u_h)\right\|_Z + \left\|\mathbb{E}(u_h) - \mathbb{E}\left(\widehat{u}_h\right)\right\|_Z\right)^2 + \frac{\mathbb{V}\left(\widehat{u}_h\right)}{N} = \left(\mathcal{E}_{\mathrm{DisErr}} + \mathcal{E}_{\mathrm{InterpErr}}\right)^2 + \mathcal{E}_{\mathrm{Stat}}^2.
\end{aligned}$$

If the expectation of u is nonzero, the nMSE can be approximated as

$$\overline{\mathcal{E}}_{A_{\mathrm{MC}}}^2 \approx \frac{\left\|\mathbb{E}(u) - \mathbb{E}\left(\widehat{u}_h\right)\right\|_Z^2}{\left\|\mathbb{E}(u_h)\right\|_Z^2} + \frac{\mathbb{V}\left(\widehat{u}_h\right)}{N\left\|\mathbb{E}(u_h)\right\|_Z^2} \leq \left(\overline{\mathcal{E}}_{\mathrm{DisErr}} + \overline{\mathcal{E}}_{\mathrm{InterpErr}}\right)^2 + \overline{\mathcal{E}}_{\mathrm{Stat}}^2,$$

where $\overline{\mathcal{E}}_{\mathrm{DisErr}}$, $\overline{\mathcal{E}}_{\mathrm{InterpErr}}$ and $\overline{\mathcal{E}}_{\mathrm{Stat}}$ are the relative errors for discretization, interpolation and statistics respectively. Given a target tolerance ϵ for the nMSE, the contribution of these three errors can be controlled by requiring that

$$\begin{aligned}
\overline{\mathcal{E}}_{\mathrm{DisErr}} &\leq \frac{\left\|u - u_h\right\|_{L^2(W,Z)}}{\left\|\mathbb{E}(u_h)\right\|_Z} \leq \frac{C_m M^{-\alpha_1}}{\left\|\mathbb{E}(u_h)\right\|_Z} \leq \theta_1\epsilon, \\
\overline{\mathcal{E}}_{\mathrm{InterpErr}} &\leq \frac{\left\|u_h - \widehat{u}_h^s\right\|_{L^2(W,Z)}}{\left\|\mathbb{E}(u_h)\right\|_Z} \leq \frac{C_p P^{-\mu_1}}{\left\|\mathbb{E}(u_h)\right\|_Z} \leq \theta_2\epsilon, \\
\overline{\mathcal{E}}_{\mathrm{Stat}}^2 &= \frac{V_h}{N} \leq \theta\epsilon^2,
\end{aligned} \tag{5.7}$$

where $\theta \in (0,1)$ is the splitting parameter between the relative bias and statistical errors, and θ_1 and θ_2 are the splitting parameters between $\overline{\mathcal{E}}_{\mathrm{DisErr}}$ and $\overline{\mathcal{E}}_{\mathrm{InterpErr}}$, such that $\theta_2 =$

$\sqrt{1-\theta} - \theta_1$, ensuring that the total nMSE remains below the desired tolerance ϵ and the discretization error does not exceed $\theta_1\epsilon$. Here, $V_h := \mathbb{V}(\widehat{u}_h)/\left\|\mathbb{E}(u_h)\right\|_Z^2$. The number of grid points M and sparse grids P are determined by (4.5). The required sample size N to achieve the desired tolerance follows the same form as that of the direct solver in (5.4), with the only difference being the value of V_h.

To estimate the computational cost associated with using the surrogate function in Monte Carlo sampling, we introduce C^e as the average cost to compute a single realization of the surrogate function. We assume this cost is independent of individual samples but depends on the number of spatial grid points M and sparse nodes P. Its computational complexity is characterized by $\mathcal{O}(M^{\gamma_1}P^{\gamma_2})$, where $\gamma_1 = 1$ since each sample realization involves a weighted sum of function evaluations at collocation points, with each evaluation being linear in M; the positive value of γ_2 depends on the specific interpolation method and collocation points used in constructing the surrogate. With this average realization cost C^e, the total expense of conducting surrogate-based Monte Carlo samplings is captured by

$$\mathscr{W}_{\text{SL-SL}}^{\text{on}} \leq (N+1)C^e = \mathcal{O}\left(\epsilon^{-2-\frac{\gamma_1}{\alpha_1}-\frac{\gamma_2}{\mu_1}}\right), \tag{5.8}$$

where the subscript 'SL-SL' represents the use of single-level stochastic collocation and single-level spatial discretization surrogate for the online Monte Carlo sampling. It is observed that under the condition $\frac{\gamma}{\alpha_1} \geq \frac{\gamma_1}{\alpha_1} + \frac{\gamma_2}{\mu_1}$, the computational complexity of the surrogate-based Monte Carlo sampling cost is always smaller than that in (5.5) of the direct solver-based counterpart. However, if this condition does not hold, the sampling cost of the surrogate model will surpass that of the direct solver, particularly as the tolerance ϵ becomes sufficiently small. However, even though sampling with the surrogate incurs a slightly larger complexity (differing by $\mathcal{O}(\epsilon^{-\frac{\gamma_2}{\mu_1}})$), the relatively small constant (as opposed to complexity) in the cost per sample C^e for surrogate evaluations ensures the viability of surrogate-based sampling as a cost-effective strategy. We will show in Chapter 6.3 that for our problem scenario, $\gamma \approx \gamma_1 = \gamma_2 = 1$. This aspect, alongside a detailed discussion on the comparative analysis of computational costs and the implications for modeling efficiency, is further elaborated in Chapter 6.3.

5.2 Multilevel Monte Carlo finite element

To enhance the computational efficiency of the Monte Carlo Finite-Element approach, we introduce an alternative sampling technique known as the multilevel Monte Carlo Finite-Element (MLMC-FE) method [12, 28, 61, 69]. Both MC-FE and MLMC-FE approaches exhibit convergence rates that remain independent of dimensionality due to the central limit theorem. Unlike Monte Carlo, which employs a single spatial grid level for sampling, the multilevel Monte Carlo reduces the computational cost associated with sampling—which in our case involves the numerical solution of a non-linear PDE in a target computational mesh—by approximating the expectation of the quantity of interest on the finest mesh using a sequence of approximations on a set of coarse physical grids that are cheaper to compute [20]. Using the linearity of expectation, the MLMC estimator expresses the quantity of interest on the finest spatial grid, $\mathbb{E}(u_h)$, by a telescoping sum involving the numerical approximations of u on coarser grids. Consequently, the workload of MLMC-FE is shifted from the fine grid to coarser grids, making it more efficient than MC-FE [12]. In this chapter, we will explore MLMC-FE samplings with both direct computation and the surrogate. Particularly, for direct computation, we will consider sampling on both uniform and adaptive grids. For surrogate-based samplings, we will consider three combinations: MLMC-FE with a direct solver on a sequence of uniform spatial grids, MLMC-FE with a direct solver on an adaptively refined spatial grid set, and the surrogate-based MLMC-FE.

To construct meshes that are easy to describe for both uniform and adaptive mesh refinement, we will characterize them using the number of grid points rather than the mesh size. We will refer to $\ell = 0, \ldots, L$ as the *level* of a mesh $\mathcal{T}_\ell$ containing $\{M_\ell\}$ grid points. We will then consider a sequence of meshes $\mathcal{T}_0, \ldots, \mathcal{T}_L$ with increasing resolution so that $\{M_\ell\}_{0 \leq \ell \leq L}$ defines an increasing sequence and $\mathcal{T}_L$ is the finest mesh, and we will denote by u_ℓ the approximation of u on the mesh $\mathcal{T}_\ell$.

5.2.1 MLMC sampling with direct solver on the uniformly refined mesh

The expectation of the function u can be approximated by the expectation of the finest approximation u_L. Since $u_L = u_0 + (u_1 - u_0) + (u_2 - u_1) + \cdots + (u_L - u_{L-1})$, $\mathbb{E}(u)$ can be

approximated as the telescoping sum

$$\mathbb{E}(u) \approx \mathbb{E}(u_L) = \mathbb{E}(u_0) + \sum_{\ell=1}^{L} \mathbb{E}(u_\ell - u_{\ell-1}) = \sum_{\ell=0}^{L} \mathbb{E}(Y_\ell), \tag{5.9}$$

where each of the terms

$$Y_0 := u_0 \quad \text{and} \quad Y_\ell := u_\ell - u_{\ell-1} \quad (\text{for } \ell \geq 1) \tag{5.10}$$

can be regarded as a correction of the coarsest approximation u_0. If each of the terms $\mathbb{E}(Y_\ell)$ is estimated by gathering N_ℓ samples at level ℓ and computing the sample expectations

$$\mathbb{E}(Y_0) \approx \widehat{Y}_0 := \frac{1}{N_0} \sum_{i=1}^{N_0} u_0^{(i)}, \qquad \mathbb{E}(Y_\ell) \approx \widehat{Y}_\ell := \frac{1}{N_\ell} \sum_{i=1}^{N_\ell} \left(u_\ell^{(i)} - u_{\ell-1}^{(i)} \right) \quad (\text{ for } \ell \geq 1),$$

then the MLMC-FE estimator at level L will be unbiased and can be written as

$$A_{\text{MLMC}}(u_L) := \sum_{\ell=0}^{L} \widehat{Y}_\ell = \frac{1}{N_0} \sum_{i=1}^{N_0} u_0^{(i)} + \sum_{\ell=1}^{L} \frac{1}{N_\ell} \sum_{i=1}^{N_\ell} \left(u_\ell^{(i)} - u_{\ell-1}^{(i)} \right). \tag{5.11}$$

Recalling that $\mathbb{E}(\widehat{Y}_\ell) = \mathbb{E}(Y_\ell)$ and $\mathbb{V}(\widehat{Y}_\ell) = \mathbb{V}(Y_\ell)/N_\ell$ we conclude that, for the MLMC-FE estimator, it follows that $\mathbb{E}\left(A_{\text{MLMC}}\right) = \sum_{\ell=0}^{L} \mathbb{E}(\widehat{Y}_\ell) = \mathbb{E}(u_L)$ and $\mathbb{V}\left(A_{\text{MLMC}}\right) = \sum_{\ell=0}^{L} \mathbb{V}(\widehat{Y}_\ell) = \sum_{\ell=0}^{L} \mathbb{V}(Y_\ell)/N_\ell$.

Note that each of the terms $Y_\ell^{(i)} := u_\ell^{(i)} - u_{\ell-1}^{(i)}$ appearing in (5.11) requires the approximation of $u^{(i)}$ on adjacent refinement levels *using the same value of the parameter* $\boldsymbol{\omega}^{(i)}$. However, when using a FEM discretization, the numerical implementation of this term does not require the solution of the PDE on the two grids $\mathcal{T}_\ell$ and $\mathcal{T}_{\ell-1}$. The realization on the coarse grid can be obtained from the one on the fine grid by either Galerkin projection or interpolation. Projection is the more accurate of the two, but requires the solution of a system involving the mass matrix for each realization. Thus we prefer interpolation, for which costs are minimal and which is of second-order accuracy in our application. To avoid introducing correlation across discretization levels, none of the samples involved in the computation of Y_ℓ is reused for the finer level $\ell + 1$. In other words, sampling is done such that, for $n \neq m$, the estimates Y_n and Y_m are uncorrelated. However, for *any particular* Y_ℓ the strong correlation between $u_\ell^{(i)}$ and $u_{\ell-1}^{(i)}$ makes the variance of the correction terms much smaller than the variance of the approximation u_L in the finest mesh, further improving the statistical approximation.

Just like for MC-FE, the mean squared error $\mathcal{E}^2_{A_{\text{MLMC}}}$ associated with MLMC-FE can be split into contributions due to bias and variance as

$$\mathcal{E}^2_{A_{\text{MLMC}}} = \mathbb{E}\left[\|\mathbb{E}(u) - A_{\text{MLMC}}(u_L)\|_Z^2\right] = \|\mathbb{E}(u) - \mathbb{E}(u_L)\|_Z^2 + \sum_{\ell=0}^{L} \frac{\mathbb{V}\left(Y_\ell\right)}{N_\ell} = \mathcal{E}^2_{\text{Bias}} + \mathcal{E}^2_{\text{Stat}}.$$

Similarly, approximating $\mathbb{E}(u)$ by $\mathbb{E}(u_L)$, the normalized mean squared error $\overline{\mathcal{E}}^2_{A_{\text{MLMC}}}$ can be approximated by

$$\overline{\mathcal{E}}^2_{A_{\text{MLMC}}} \approx \frac{\|\mathbb{E}(u) - \mathbb{E}(u_L)\|_Z^2}{\|\mathbb{E}(u_L)\|_Z^2} + \sum_{\ell=0}^{L} \frac{V_\ell}{N_\ell} = \overline{\mathcal{E}}^2_{\text{Bias}} + \overline{\mathcal{E}}^2_{\text{Stat}},$$

where

$$V_\ell := \mathbb{V}\left(Y_\ell\right) / \|\mathbb{E}(u_L)\|_Z^2 \,. \tag{5.12}$$

Just as before, the parameter $\theta \in (0,1)$ can be used to split the contributions of the relative discretization error $\overline{\mathcal{E}}_{\text{Bias}}$ by requiring that $\overline{\mathcal{E}}^2_{\text{Bias}} \leq (1-\theta)\epsilon^2$ and $\overline{\mathcal{E}}^2_{\text{Stat}} \leq \theta\epsilon^2$ where ϵ is a predetermined tolerance such that $\overline{\mathcal{E}}^2_{A_{\text{MLMC}}} \leq \epsilon^2$.

To quantify the computational effort of the MLMC-FE estimator, let $M_{\ell,i}$ be the number of grid points for the i-th sample on mesh level ℓ. We will assume that the computational cost to obtain one sample of $u_\ell^{(i)}$ is $C_{\ell,i} := C(u_\ell^{(i)}) = \mathcal{O}(M_{\ell,i}^\gamma)$, where the exponent $\gamma > 0$ depends on the solver, and will denote the cost of computing the correction term $Y_\ell^{(i)}$ by $C_{\ell,i} := C(Y_\ell^{(i)}) = \mathcal{O}(M_{\ell,i}^\gamma)$ for $\ell \geq 0$ and $M_{-1,i} = 0$. For a nonlinear problem like the one at hand, the particular realization $\boldsymbol{\omega}^{(i)}$ will influence the cost, hence for the forthcoming analysis, we will consider the *average* cost to be of the form

$$C_\ell = \mathcal{O}(M_\ell^\gamma), \tag{5.13}$$

and will use this to estimate the total cost as

$$C(A_{\text{MLMC}}) = \sum_{\ell=0}^{L}\sum_{i=1}^{N_\ell} C_{\ell,i} = \sum_{\ell=0}^{L} N_\ell C_\ell = \sum_{\ell=0}^{L} N_\ell \cdot \mathcal{O}\left(M_\ell^\gamma\right).$$

We will use the expression above to estimate the sample size N_ℓ through an optimization procedure that minimizes the work of the MLMC-FE estimator utilizing the method of Lagrange multipliers subject to the inequality constraint $\overline{\mathcal{E}}^2_{\text{Stat}} \leq \theta\epsilon^2$ [28]. The Lagrangian for this problem is written as

$$\mathcal{L} = \sum_{\ell=0}^{L}\sum_{i=1}^{N_\ell} C_{\ell,i} + \frac{1}{\lambda^2}\left(\sum_{\ell=0}^{L} \frac{V_\ell}{N_\ell} - \theta\epsilon^2 + t^2\right) = \sum_{\ell=0}^{L} N_\ell C_\ell + \frac{1}{\lambda^2}\left(\sum_{\ell=0}^{L} \frac{V_\ell}{N_\ell} - \theta\epsilon^2 + t^2\right),$$

where λ is the Lagrange multiplier and the auxiliary variable t^2, known as a *slack variable* in optimization [29, 62], was introduced to transform the inequality constraint into an equality constraint. Treating N_ℓ as a continuous variable and under a set of suitable assumptions, known as the Karush-Kuhn-Tucker optimality conditions (or simply KKT conditions) [8, 62], the optimal sample size can be shown [28] to be

$$N_\ell = \left\lceil \frac{1}{\theta\epsilon^2}\sqrt{\frac{V_\ell}{C_\ell}}\sum_{k=0}^{L}\sqrt{V_k C_k}\right\rceil. \tag{5.14}$$

With this expression for N_ℓ, the optimal total cost for the MLMC-FE estimator is

$$\mathscr{W}_{\text{Direct}}^{\text{on}} \leq \frac{1}{\theta\epsilon^2}\left(\sum_{\ell=0}^{L}\sqrt{V_\ell C_\ell}\right)^2 + \sum_{\ell=0}^{L} C_\ell, \tag{5.15}$$

where the subscript 'Direct' indicates the usage of a direct solver for conducting the online multilevel Monte Carlo sampling.

The formula (5.14) suggests an iterative procedure for the approximation of $\mathbb{E}(u)$. Starting from a computational mesh $\mathcal{T}_0$, gather an initial number $\widetilde{M}_0$ of samples $u_0^{(i)}$ and estimate $\overline{\mathcal{E}}_{\text{Bias}}$, $\overline{\mathcal{E}}_{\text{Stat}}$, and V_0. If $\overline{\mathcal{E}}_{\text{Stat}}$ is larger than the prescribed tolerance, use (5.14) to update $\widetilde{M}_0$ and gather additional samples; if $\overline{\mathcal{E}}_{\text{Bias}}$ is above the prescribed tolerance, then add an additional level of spatial refinement. The process continues adding discretization levels and collecting additional samples until both $\overline{\mathcal{E}}_{\text{Bias}}$ and $\overline{\mathcal{E}}_{\text{Stat}}$ fall below the required tolerance, at which point $\mathbb{E}(u)$ is approximated using equation (5.9). This simple algorithm, however, presents one challenge: the term V_ℓ in equation (5.14) requires the computation of the term

$$\mathbb{V}(Y_\ell) = \frac{1}{N_\ell - 1}\left(\sum_{i=1}^{N_\ell}\left\|Y_\ell^{(i)}\right\|_Z^2 - \frac{1}{N_\ell}\left\|\sum_{i=1}^{N_\ell} Y_\ell^{(i)}\right\|_Z^2\right). \tag{5.16}$$

However, the estimated sample sizes $\{N_\ell\}$ are available only for pre-existing discretization levels, hence whenever an additional mesh refinement is needed, N_{L+1} cannot be approximated by (5.14) as this formula uses V_{L+1} to compute N_{L+1}. This inconvenience can be overcome by noting that

$$\mathbb{V}(u - u_\ell) = \mathbb{E}\left[\|u - u_\ell\|_Z^2\right] - \|\mathbb{E}\left(u - u_\ell\right)\|_Z^2 \leq \mathbb{E}\left[\|u - u_\ell\|_Z^2\right]. \tag{5.17}$$

Hence, the variance $\mathbb{V}(u - u_\ell)$ can be estimated by the expectation of the squared discretization error $\|u - u_\ell\|_Z^2$. For a uniformly refined grid, we can resort to a standard *a priori*

error estimate and assume that $\mathbb{E}[\|u - u_\ell\|_Z^2] = \mathcal{O}(M_\ell^{-\beta_1})$ for some $\beta_1 > 0$. It then follows that $V_\ell = \mathcal{O}(M_\ell^{-\beta_1})$ and following [59] we can then approximate V_{L+1} in terms of the known variance V_L by

$$V_{L+1} = (M_{L+1}/M_L)^{-\beta_1} V_L. \tag{5.18}$$

A summary of the preceding analysis on the computational cost to compute the MLMC-FE estimator in terms of the desired relative accuracy ϵ was established in full generality in [28, Theorem 1].

Theorem 2. *Suppose there exist positive constants $\alpha_1, \beta_1, \gamma$ such that $\alpha_1 \geq \frac{1}{2}\min(\beta_1, \gamma)$,*

(i) $\|\mathbb{E}(u - u_\ell)\|_Z = \mathcal{O}\left(M_\ell^{-\alpha_1}\right)$,

(ii) $V_\ell = \mathcal{O}\left(M_\ell^{-\beta_1}\right)$,

(iii) $C_\ell = \mathcal{O}\left(M_\ell^{\gamma}\right)$.

Then for any positive $\epsilon < e^{-1}$ small enough, there exists level L and sample size N_ℓ for which the multilevel estimator A_{MLMC} has an nMSE with

$$\frac{\|\mathbb{E}(u) - A_{MLMC}(\widehat{u}_L)\|_{L^2(W,Z)}^2}{\|\mathbb{E}(u)\|_{L^2(W,Z)}^2} < \epsilon^2,$$

and the total sampling cost with bound

$$\mathscr{W}_{Direct}^{on} = \begin{cases} \mathcal{O}\left(\epsilon^{-2}\right), & \beta_1 > \gamma, \\ \mathcal{O}\left(\epsilon^{-2}\left|\log \epsilon\right|^2\right), & \beta_1 = \gamma, \\ \mathcal{O}\left(\epsilon^{-2-\frac{\gamma-\beta_1}{\alpha_1}}\right), & 0 < \beta_1 < \gamma. \end{cases}$$

See [12] for a detailed proof of this theorem. The theorem states that if the decay rate of the variance β_1 surpasses the growth rate of cost per sample γ, then $\mathbb{V}(Y_\ell)$ diminishes more rapidly than the increase of C_ℓ as ℓ grows, leading to the primary computational effort being concentrated on the coarsest grid level. Conversely, if β is lower than γ, the predominant computational load shifts to the finest grid. When β equals γ, the computational workload is evenly distributed across all grid levels [28].

5.2.2 MLMC sampling with direct solver on the adaptively refined mesh

In view of the benefits of approximating the quantity of interest across a sequence of increasingly finer meshes, and with the goal of further reducing the computational cost associated with reducing the bias associated with the numerical discretization, it is natural to focus the refinement only on those parts of the mesh where the error is concentrated. Our goal is then to, starting from a computational mesh $\mathcal{T}_0$, generate a family of adaptively refined meshes $\{\mathcal{T}_\ell\}_{0\leq\ell\leq L}$ that will produce better approximations of $\mathbb{E}(u)$ than the ones resulting from consecutive uniform refinements of the initial grid. With this goal in mind, the use of an *a posteriori* error estimator to guide the construction of the family of meshes has been proposed in the context of multilevel Monte Carlo methods [17, 37, 38, 47].

A key ingredient in an adaptive solver is a local error estimator. In our case, for each element K on the mesh $\mathcal{T}_\ell$, we will use the simple residual-based a posteriori error indicator

$$\eta_{K,\ell}(\boldsymbol{\omega}) := h_K^2 \left\| \nabla \cdot \left(\tfrac{1}{\mu r} \nabla u_\ell(\boldsymbol{\omega}) \right) - f(u_\ell(\boldsymbol{\omega})) \right\|_K + h_K^{3/2} \left\| \left[\!\left[\tfrac{1}{\mu r} \nabla u_\ell(\boldsymbol{\omega}) \cdot \boldsymbol{n} \right]\!\right] \right\|_{\partial K \setminus \partial\Omega}, \quad (5.19)$$

where ∂K is the boundary of the element K, h_K is the diameter of K, $\boldsymbol{n}$ is the outward unit normal to the element K, $[\![\cdot]\!]$ denotes the jump across the edge of an interior element, and f is the source term defined piecewise on the right-hand side of (2.5a). Following [17, 47], we will further define the mean local and mean global error estimator respectively as

$$\eta_{K,\ell} := \mathbb{E}\left(\eta_{K,\ell}(\boldsymbol{\omega})\right) \quad \text{and} \quad \eta_\ell^2 := \sum_{K\in\mathcal{T}_\ell} \eta_{K,\ell}^2. \quad (5.20)$$

For linear deterministic problems, estimators of this form can be shown to be such that there are constants $C_1, C_2 > 0$ such that $C_1\eta_\ell \leq \|u - u_\ell\|_Z \leq C_2\eta_\ell$ [70]. Therefore, the error estimator will accurately locate the regions of high error density and will decay at the same rate as the true error [52]. The global error can then be approximated by adding the local estimators over the entire triangulation. Using these error estimators, the adaptive analogue of (5.17) can be written as

$$\mathbb{V}(u - u_\ell) \leq \mathbb{E}\left[\|u - u_\ell\|_Z^2\right] \approx \eta_\ell^2, \quad (5.21)$$

which then leads to the following adaptive analogue of the extrapolation formula (5.18)

$$V_{L+1} = \left(\eta_{L+1}/\eta_L\right)^2 V_L. \quad (5.22)$$

This estimate can then be used in combination with (5.14) to obtain an update for the sample size required at each adaptive level.

With these definitions in place, we can then describe our strategy, which follows the "SOLVE → ESTIMATE → MARK → REFINE" paradigm familiar from deterministic adaptive solvers [10], as:

1. **Solve:** Starting from a fixed number of samples, the problem (2.5a) is solved on the initial mesh $\mathcal{T}_0$.

2. **Estimate:** The local mean error estimator is approximated from the sample gathered.

3. **Mark:** The set $\mathcal{M}_\ell$ containing the smallest possible number of elements in $\mathcal{T}_0$ satisfying

$$\sum_{K \in \mathcal{M}_\ell} \eta_{K,\ell}^2 \geq \zeta \eta_\ell^2, \tag{5.23}$$

   for some predetermined value $\zeta \in [0, 1]$ is marked for refinement—this marking strategy is known as *Dörfler marking* in the adaptive finite element community [16].

4. **Refine:** The elements marked are refined in such a way that the resulting triangulation $\mathcal{T}_{\ell+1}$ is shape-regular and conforming. Efforts should be made to make sure that the growth of the number of elements is kept at bay. In our case, we used the implementation given in [23] of the algorithms described in [9, 22, 58].

The steps above are repeated until the error estimator η falls below a certain predetermined value.

For uniformly refined grids, the notion of mesh level is natural: starting from a mesh $\mathcal{T}_\ell$, one step of uniform refinement decreases the mesh parameter h across the grid by a factor of $1/2$; the resulting mesh is said to have level $\ell + 1$ and is denoted by $\mathcal{T}_{\ell+1}$. For adaptively refined grids, where the mesh parameter is not constant through the grid, the notion of the level does not come so naturally. We will use the fact that, for a uniform refinement, the numerical error decays by a factor of $(1/2)^p$, (where p is the order of the FEM solver) with each successive level to extend the notion of mesh level to adaptively refined grids.

Consider a numerical approximation u_ℓ obtained on a mesh $\mathcal{T}_\ell$ with an associated error estimation given by η_ℓ. We will say that a mesh has level $\ell + 1$ and will denote it by $\mathcal{T}_{\ell+1}$

if it was obtained from $\mathcal{T}_\ell$ by cycling over the adaptive loop using the value $(1/2)^p\eta_\ell$ as the stopping tolerance. In other words, we will say that an adaptively refined mesh has level $\ell+1$ if it produces a numerical solution with an error $(1/2)^p$ times smaller than one with level ℓ, just like in the uniform case. We will refer to $q := (1/2)^p$ as the decay factor. In terms of discretization accuracy, after ℓ steps of adaptive refinement, an adaptively refined mesh with level ℓ will have an associated error estimation $\eta_\ell = q^\ell\eta_0$, where η_0 corresponds to the error estimation at the initial mesh. In our numerical experiments, since the convergence rate of the piecewise linear solver is 2 (when measured in the L^2 norm), we shall use a decay factor $q = 1/4$ to define our adaptively refined meshes.

Ideally, in the stochastic setting, all the error estimations collected from the totality of samples would be used to drive the adaptive refinement forward and build an optimal set of meshes at every level. However, due to the iterative nature of the algorithm arising from (5.14), the optimal mesh at every level would have to be corrected with every new batch of samples and the solutions corresponding to all previous realizations $\boldsymbol{\omega}^{(i)}$ would have to be recomputed. The computational cost of re-sampling in this manner quickly becomes impractical.

Instead, following [37, 38, 51], we will construct a sequence of deterministic adaptive grids with partial knowledge about $\mathbb{E}(u)$ as follows. Starting from a sample $\{\boldsymbol{\omega}^{(i)}\}_{1\le i\le N}$ (where N is small and arbitrarily chosen) and a mesh $\mathcal{T}_0$, the PDE is solved and the local error is estimated for every solution $u_0^{(i)}$, resulting in N local error estimators $\{\eta_{K,0}(\boldsymbol{\omega}^{(i)})\}_{1\le i\le N}$. The mean local and global error estimators $\eta_{K,0}$ and η_0 are then approximated by the sample means of the individual estimators. Using this approximation of $\eta_{K,0}$, the mesh $\mathcal{T}_0$ is refined. This process is continued until the approximated mean error estimator satisfies $\eta \le q\eta_0$; the resulting mesh is stored and labeled as $\mathcal{T}_1$ (mesh level 1). The previous steps are repeated until a target number of meshes $\{\mathcal{T}_\ell\}_{0\le\ell\le L}$ have been generated. The process is described in Algorithm 1. Since the family of meshes produced is constructed using random samples of $\boldsymbol{\omega}$, they approximately reduce the error for the approximate expectation $\mathbb{E}(u_h)$ by a factor of q with every increasing level. This family of meshes is then kept fixed during the MLMC run.

Algorithm 1: Generate adaptive deterministic mesh set

Input: Initial mesh $\mathcal{T}_0$, maximum mesh level L, element marking percentage $\zeta \in (0,1)$, sample size N, error decay factor $q \in (0,1)$.

Output: Adaptive mesh set $\{\mathcal{T}_\ell\}_{\ell=0}^{L}$.

```
for ℓ = 1, ··· , L do
    Refine = TRUE.
    Set j = 1 and T^(j) = T_{ℓ-1}.
    while Refine do
        for i = 1, ··· , N do
            (i) Draw a random sample ω^(i) from W.
            (ii) Obtain u^(i) by solving the free boundary problem on on T^(j) using
                 ω^(i).
            (iii) Calculate and accumulate the local and global error estimators
                 η_{K,j}(ω^(i)) and η_j(ω^(i)).
        Compute the mean estimators η_{K,j} and η_j from the accumulated samples.
        Mark & refine the mesh T^(j) according to η_{K,j} to obtain T^(j+1).
        if η_j ≤ q η_1 then
            Store T^(j) as T_ℓ.
            Refine = FALSE.
        j = j + 1
```

5.2.3 MLMC sampling with surrogates on the uniformly refined mesh

Despite the improvement of MLMC over the MC methods, both approaches conventionally rely on "direct solve" strategies, which entail the computationally intensive task of numerically solving discretized partial differential equations for each sample realization. This section proposes the utilization of the surrogate introduced in Chapter 4 as a strategic substitute for the direct solve process during MLMC sampling aimed at significantly boosting its efficiency. We will consider the two types of multilevel surrogates (with either single-level sparse grid (4.7) or multilevel sparse grid(4.14)) for MLMC samplings. Note that both surrogates can be characterized similarly, with the key distinction lying in whether the sparse level q varies with the spatial mesh level k. We denote the generic representation as "$\check{u}_\ell$", with a variable level "q" representing either the single-level stochastic collocation and multilevel

Algorithm 2: Multilevel Monte Carlo Finite Element method

Input: Initial mesh level $L = 0$, sequence of mesh available $\{\mathcal{T}_\ell\}_{\ell\geq 0}$, root nMSE ϵ, $\zeta \in (0,1)$, initial sample size $N_{\text{old}} = \{N_\ell\}_{\ell=0}^L$, counter $j = 1$, sample size corrections $dN_\ell^j = \{N_\ell\}_{\ell=0}^L$.

Output: $\{N_\ell\}_{\ell=0}^L$, $\{V_\ell\}_{\ell=0}^L$, A_{MLMC}.

```
while ∑_ℓ(dN_ℓ^j) > 0 do
    for 0 ≤ ℓ ≤ L do
        for i = 1, ··· , dN_ℓ^j do
            Solve the free boundary problem on T_ℓ to get u_ℓ^(i) for the i-th sample.
    Approximate {V_ℓ}_{ℓ=0}^L by (5.16).
    Update the sample size estimation {N_ℓ}_{ℓ=0}^L by (5.14).
    j = j + 1.
    dN_ℓ^j = {N_ℓ}_{ℓ=0}^L − N_old.
    N_old = {N_ℓ}_{ℓ=0}^L.
    if ∑_ℓ(dN_ℓ^j) = 0 then
        if The a posteriori error estimator falls below √(1 − θ)ε ‖E(u)‖_Z, then
            Compute A_MLMC by (5.11) and terminate the loop.
        else
            L = L + 1.
            Approximate V_L by (5.18) and compute {N_ℓ}_{ℓ=0}^L and go to Step 1.
```

spatial discretization surrogate $\widehat{u}_\ell^s$ or the multilevel in both stochastic collocation and spatial discretization surrogate $\widehat{u}_\ell^m$, specifying the relevant one as needed.

The expectation of u can be approximated by the expectation of the surrogate function on the finest spatial level L, denote this surrogate as $\widehat{u}_L$ (either the single-level stochastic collocation and multilevel spatial discretization surrogate $\widehat{u}_L^s$ or the multilevel in both parameter and spatial grids surrogate $\widehat{u}_L^m$). By exploiting the linearity of the expectation operator, we express the surrogate on the fine grid $\mathbb{E}\left(\widehat{u}_L\right)$ as a telescoping sum of surrogate corrections on coarser grids as

$$\mathbb{E}\left(\widehat{u}_L\right) = \sum_{\ell=0}^{L} \mathbb{E}(\check{u}_\ell - \check{u}_{\ell-1}) = \sum_{\ell=0}^{L} \mathbb{E}(\check{Y}_\ell),$$

where the surrogate corrections $\check{Y}_\ell$ are given by

$$\check{Y}_\ell := \check{u}_\ell - \check{u}_{\ell-1} \quad \text{for} \quad \ell \geq 1, \qquad \text{and} \qquad \check{Y}_0 := \check{u}_0. \tag{5.24}$$

Let N_ℓ represent the number of samples required on spatial mesh level ℓ. The unbiased MLMC-FE multilevel surrogate estimator $A_{\text{MLMC}}(\widehat{u}_L)$ for $\mathbb{E}(u)$ is defined as

$$A_{\text{MLMC}}\left(\widehat{u}_L\right) := \frac{1}{N_\ell} \sum_{i=1}^{N_\ell} \sum_{\ell=0}^{L} \left(\check{u}_\ell^{(i)} - \check{u}_{\ell-1}^{(i)}\right) = \frac{1}{N_\ell} \sum_{i=1}^{N_\ell} \sum_{\ell=0}^{L} \check{Y}_\ell^{(i)}. \tag{5.25}$$

For this surrogate estimator, it holds that

$$\mathbb{E}\left(A_{\text{MLMC}}\right) = \mathbb{E}(\widehat{u}_L), \qquad \mathbb{V}\left(A_{\text{MLMC}}\right) = \sum_{\ell=0}^{L} \frac{\mathbb{V}(\check{u}_\ell^{(i)} - \check{u}_{\ell-1}^{(i)})}{N_\ell}.$$

Notably, within the correction, $\check{u}_\ell^{(i)}$ and $\check{u}_{\ell-1}^{(i)}$ use the same sample on two consecutive spatial meshes, considering the telescoping sum in (4.7) and (4.14), the evaluation of the surrogate correction in (5.24) can be simplified as

$$\check{Y}_\ell^{(i)} = \sum_{k=0}^{\ell} \widehat{Y}_{q,k}^{(i)} - \sum_{k=0}^{\ell-1} \widehat{Y}_{q,k}^{(i)} = \widehat{Y}_{q,\ell}^{(i)} \quad \text{for } \ell \geq 0, \tag{5.26}$$

which implies that MLMC-FE sampling with the multilevel surrogates for the correction term is equivalent to multilevel sampling with a single-level stochastic collocation and single-level spatial discretization surrogate. Thus, the surrogate estimator in (5.25) is reduced to

$$A_{\text{MLMC}}\left(\widehat{u}_L\right) = \frac{1}{N_\ell} \sum_{i=1}^{N_\ell} \sum_{\ell=0}^{L} \widehat{Y}_{q,\ell}^{(i)} = \frac{1}{N_0} \sum_{i=1}^{N_0} \widehat{u}_{q,0}^{(i)} + \sum_{\ell=1}^{L} \frac{1}{N_\ell} \sum_{i=1}^{N_\ell} \left(\widehat{u}_{q,\ell}^{(i)} - \widehat{u}_{q,\ell-1}^{(i)}\right), \tag{5.27}$$

where $q = q_s = q(L)$ for the single-level stochastic collocation and multilevel spatial discretization surrogate and $q = q(\ell)$ for the multilevel in both stochastic collocation and spatial discretization surrogate, rather than maintaining the multilevel structure in the surrogate such that $q = q(\ell)$ and $q = q(k)$ respectively. Moreover, samples for surrogate corrections at different levels are independently drawn. As ℓ grows, the strong correlation between $\widehat{u}_{q,\ell}^{(i)}$ and $\widehat{u}_{q,\ell-1}^{(i)}$ leads to a reduction in variance $\mathbb{V}(\check{Y}_\ell^{(i)})$, contributing to a decreased sample size as the mesh gets finer and thereby improving the efficiency of the MC approach.

Similar to the accuracy analysis of the MC-FE surrogate estimator, let $\mathcal{E}^2_{A_{\text{MLMC}}}$ denote the mean square error for the MLMC-FE multilevel surrogate estimator. The decomposition

of the MSE error is expressed as

$$
\begin{aligned}
\mathcal{E}^2_{A_{\text{MLMC}}} &= \mathbb{E}\left[\|\mathbb{E}(u) - \mathbb{E}\left(\widehat{u}_L\right)\|_Z^2\right] + \mathbb{E}\left[\|\mathbb{E}\left(\widehat{u}_L\right) - A_{\text{MLMC}}\left(\widehat{u}_L\right)\|_Z^2\right], \\
&= \|\mathbb{E}(u) - \mathbb{E}\left(\widehat{u}_L\right)\|_Z^2 + \sum_{\ell=0}^{L} \frac{\mathbb{V}\left(\widehat{u}_{q,\ell} - \widehat{u}_{q,\ell-1}\right)}{N_\ell} = \mathcal{E}^2_{\text{Bias}} + \mathcal{E}^2_{\text{Stat}}, \\
&\leq \left(\|\mathbb{E}(u) - \mathbb{E}(u_L)\|_Z + \|\mathbb{E}(u_L) - \mathbb{E}\left(\widehat{u}_L\right)\|_Z \right)^2 + \sum_{\ell=0}^{L} \frac{\mathbb{V}\left(\widehat{u}_{q,\ell} - \widehat{u}_{q,\ell-1}\right)}{N_\ell} \\
&= \left(\mathcal{E}_{\text{DisErr}} + \mathcal{E}_{\text{InterpErr}}\right)^2 + \mathcal{E}^2_{\text{Stat}}.
\end{aligned}
$$

Using $\|\mathbb{E}(u_L)\|_Z^2$ as the normalization factor, the nMSE for the MLMC-FE surrogate estimator can be approximated as

$$
\overline{\mathcal{E}}^2_{A_{\text{MLMC}}} \approx \frac{\|\mathbb{E}(u) - \mathbb{E}\left(\widehat{u}_L\right)\|_Z^2}{\|\mathbb{E}(u_L)\|_Z^2} + \sum_{\ell=0}^{L} \frac{V_\ell}{N_\ell},
$$

where $V_\ell = \mathbb{V}\left(\widehat{u}_{q,\ell} - \widehat{u}_{q,\ell-1}\right) / \|\mathbb{E}(u_L)\|_Z^2$. To achieve an nMSE tolerance of ϵ for the MLMC-FE multilevel surrogate estimator ($\overline{\mathcal{E}}^2_{A_{\text{MLMC}}} \leq \epsilon^2$), the relative discretization, interpolation, and statistical errors should satisfy conditions with splitting parameters $\theta, \theta_1, \theta_2$ such that

$$
\begin{aligned}
\overline{\mathcal{E}}_{\text{DisErr}} &\leq \frac{\|u - u_L\|_{L^2(W,Z)}}{\|\mathbb{E}(u_L)\|_Z} \leq \frac{C_m M_L^{-\alpha_1}}{\|\mathbb{E}(u_L)\|_Z} \leq \theta_1 \epsilon, \\
\overline{\mathcal{E}}_{\text{InterpErr}} &\leq \frac{\|u_L - \widehat{u}_L\|_{L^2(W,Z)}}{\|\mathbb{E}(u_L)\|_Z} \leq \frac{C_p P^{-\mu_1}}{\|\mathbb{E}(u_L)\|_Z} \leq \theta_2 \epsilon, \\
\overline{\mathcal{E}}^2_{\text{Stat}} &= \sum_{\ell=0}^{L} \frac{V_\ell}{N_\ell} \leq \theta \epsilon^2,
\end{aligned}
\tag{5.28}
$$

where $\theta \in (0,1)$ is the splitting parameter between the relative bias and statistical errors, θ_1 and θ_2 (equals $\sqrt{1-\theta} - \theta_1$) are the splitting parameters between $\overline{\mathcal{E}}_{\text{DisErr}}$ and $\overline{\mathcal{E}}_{\text{InterpErr}}$, and $V_\ell := \mathbb{V}(\widehat{Y}_{q,\ell}) / \|\mathbb{E}(u_h)\|_Z^2$.

To quantify the sampling cost associated with the multilevel surrogate estimator, we assume the average cost of evaluating a sample for the multilevel surrogate is $C^e_\ell = \mathcal{O}\left(M_\ell^{\gamma_1} P_q^{\gamma_2}\right)$, with $\gamma_1 = 1$ and $\gamma_2 > 0$ relies on the method and type of sparse grids for the multilevel surrogate. The total cost is estimated as

$$
\mathscr{W}^{\text{on}}_{\text{Surrog}} = \sum_{\ell=0}^{L} N_\ell C^e_\ell = \sum_{\ell=0}^{L} N_\ell \cdot \mathcal{O}\left(M_\ell^{\gamma_1} P_q^{\gamma_2}\right), \tag{5.29}
$$

where the subscript 'Surrog' denotes the usage of either the single-level sparse grid and multilevel spatial grid surrogate ('SL-ML') or the multilevel in both parameter and spatial grids surrogate ('ML-ML') during online sampling for multilevel Monte Carlo.

An optimization problem is formulated by minimizing this cost subject to a bounded statistical error ($\overline{\mathcal{E}}^2_{\text{Stat}} \leq \theta\epsilon^2$), which yields the sample size estimation N_ℓ [19, 27] given by

$$N_\ell = \left\lceil \frac{1}{\theta\epsilon^2}\sqrt{\frac{V_\ell}{C^e_\ell}}\sum_{k=0}^{L}\sqrt{V_k C^e_k}\right\rceil. \tag{5.30}$$

With the optimal sample size N_ℓ, the sampling cost for the MLMC-FE surrogate estimator is expressed as

$$\mathscr{W}^{\text{on}}_{\text{Surrog}} \leq \frac{1}{\theta\epsilon^2}\left(\sum_{\ell=0}^{L}\sqrt{V_\ell C^e_\ell}\right)^2 + \sum_{\ell=0}^{L} C^e_\ell. \tag{5.31}$$

The term V_ℓ involves the computation of $\mathbb{V}(\widehat{Y}_{q,\ell})$ [19] defined as

$$\mathbb{V}\left(\widehat{Y}_{q,\ell}\right) = \frac{1}{N_\ell - 1}\left(\sum_{i=1}^{N_\ell}\left\|\widehat{Y}^{(i)}_{q,\ell}\right\|^2_Z - \frac{1}{N_\ell}\left\|\sum_{i=1}^{N_\ell}\widehat{Y}^{(i)}_{q,\ell}\right\|^2_Z\right). \tag{5.32}$$

Algorithm 2 outlines the surrogate-based MLMC-FE method. In the theorem, estimating the required sample size for a new spatial mesh level involves approximating V_{L+1} based on the known V_L. For this variance extrapolation, we consider seeking an upper bound for the quantity $\mathbb{V}(u - \check{u}_\ell)$ as

$$\mathbb{V}(u - \check{u}_\ell) = \mathbb{E}\left[\|u - \check{u}_\ell\|^2_Z\right] - \|\mathbb{E}\left(u - \check{u}_\ell\right)\|^2_Z \leq \mathbb{E}\left[\|u - \check{u}_\ell\|^2_Z\right],$$

where the right-hand side is the expectation of the squared total error $\|u - \check{u}_\ell\|^2_Z$, or the so-called squared strong error. By selecting the number of sparse grid nodes according to (4.12) and (4.17) for the single-level stochastic collocation and multilevel spatial discretization and multilevel in both stochastic collocation and spatial discretization surrogates and using the fact that P_q is a function of M_ℓ, we have shown that the total errors in (4.9) and (4.15) decay at the same rate as the discretization error. For a uniformly refined grid, we can resort to a standard *a priori* error estimate for the discretization error and assume that $\mathbb{E}[\|u - \check{u}_\ell\|^2_Z] = \mathcal{O}(M_\ell^{-\beta_1})$ for some $\beta_1 > 0$. It then follows that $V_\ell = \mathcal{O}(M_\ell^{-\beta_1})$ for both single-level stochastic collocation and multilevel spatial discretization and multilevel in both

stochastic collocation and spatial discretization surrogates, allowing us to approximate V_{L+1} in terms of the known variance V_L as

$$V_{L+1} = (M_{L+1}/M_L)^{-\beta_1} V_L, \tag{5.33}$$

Given the assumption on V_ℓ, the sampling cost in (5.31) for MLMC-FE with both single-level stochastic collocation and multilevel spatial discretization and multilevel in both stochastic collocation and spatial discretization surrogates can be further expressed in terms of the desired relative accuracy ϵ as in Theorem 3 and Theorem 4 respectively.

Theorem 3. *Suppose there exist positive constants $\alpha_1, \beta_1, \gamma_1$ such that $2\alpha_1 \geq \min(\beta_1, \gamma_1)$,*

(i) $\|\mathbb{E}\left(u - u_\ell\right)\|_Z = \mathcal{O}\left(M_\ell^{-\alpha_1}\right)$,

(ii) $\|Y_\ell - \widehat{Y}_{q_s,\ell}\|_{L^2(W,Z)} = \mathcal{O}\left(M_\ell^{-\alpha_2} P_{q_s}^{-\mu_1}\right)$,

(iii) $V_\ell = \mathcal{O}\left(M_\ell^{-\beta_1}\right)$,

(iv) $C_\ell^e = \mathcal{O}\left(M_\ell^{\gamma_1} P_{q_s}^{\gamma_2}\right)$.

Then for any positive $\epsilon < e^{-1}$ small enough, there exists spatial grid level L, sparse grid level $q(L)$ and sample size N_ℓ for which the multilevel estimator A_{MLMC} has an nMSE with

$$\frac{\|\mathbb{E}(u) - A_{MLMC}(\widehat{u}_L)\|^2_{L^2(W,Z)}}{\|\mathbb{E}(u)\|^2_{L^2(W,Z)}} < \epsilon^2,$$

and the total sampling cost with bound

$$\mathscr{W}^{on}_{SL\text{-}ML} = \begin{cases} \mathcal{O}\left(\epsilon^{-2-\frac{\gamma_2}{\mu_1}}\right), & \beta_1 > \gamma_1, \\ \mathcal{O}\left(\epsilon^{-2-\frac{\gamma_2}{\mu_1}} |\log \epsilon|^2\right), & \beta_1 = \gamma_1, \\ \mathcal{O}\left(\epsilon^{-2-\frac{\gamma_2}{\mu_1}-\frac{\gamma_1-\beta_1}{\alpha_1}}\right), & \beta_1 < \gamma_1. \end{cases}$$

Proof. To estimate the sampling cost of MLMC-FE with the Single-level sparse grid stochastic collocation and multilevel spatial discretization surrogate, we examine the two terms in the upper bound of the sampling cost in (5.31) as

$$H_1 := \frac{1}{\theta\epsilon^2}\left(\sum_{\ell=0}^{L} \sqrt{V_\ell C_\ell^e}\right)^2 \sim \frac{1}{\theta\epsilon^2} P_{q_s}^{\gamma_2}\left(\sum_{\ell=0}^{L} s^{\frac{\ell(\gamma_1-\beta_1)}{2}}\right)^2,$$

$$H_2 := \sum_{\ell=0}^{L} C_\ell^e \sim P_{q_s}^{\gamma_2} \sum_{\ell=0}^{L} M_\ell^{\gamma_1} = P_{q_s}^{\gamma_2} \sum_{\ell=0}^{L} s^{\ell\gamma_1},$$

derived based on the assumptions on the cost per sample for surrogate evaluation $C_\ell^e \sim M_\ell^{\gamma_1} P_{q_s}^{\gamma_2}$ and the variance $V_\ell \sim M_\ell^{-\beta_1}$, where "$\sim$" indicates the omission of constants. The term $\sum_{\ell=0}^{L} s^{\frac{\ell(\gamma_1-\beta_1)}{2}}$ can be estimated using (A.1) with $\eta = \frac{\gamma_1-\beta_1}{2}$, and $\sum_{\ell=0}^{L} s^{\ell\gamma_1} = \mathcal{O}\left(\epsilon^{-\frac{\gamma_1}{\alpha_1}}\right)$ with $\eta = \gamma_1$. With P_{q_s} estimated in (4.12), the two components in the cost can be found as

$$H_1 = \begin{cases} \mathcal{O}\left(\epsilon^{-2-\frac{\gamma_2}{\mu_1}}\right), & \beta_1 > \gamma_1, \\ \mathcal{O}\left(\epsilon^{-2-\frac{\gamma_2}{\mu_1}} (\log\epsilon)^2\right), & \beta_1 = \gamma_1, \\ \mathcal{O}\left(\epsilon^{-2-\frac{\gamma_2}{\mu_1}-\frac{\gamma_1-\beta_1}{\alpha_1}}\right), & \beta_1 < \gamma_1, \end{cases} \quad \text{and} \quad H_2 = \mathcal{O}\left(\epsilon^{-\frac{\gamma_2}{\mu_1}-\frac{\gamma_1}{\alpha_1}}\right).$$

In the following, we assert that under the assumption of the parameter condition $2\alpha_1 \geq \min(\beta_1, \gamma_1)$ in Theorem 3, H_1 dominates H_2, resulting in the combined cost $H_1 + H_2$ behaving similarly to H_1. Consequently, the cost of $\mathscr{W}_{\text{SL-ML}}^{\text{off}}$ is governed by the behavior of H_1.

When $\beta_1 > \gamma_1$, the assumption implies $2 \geq \frac{\gamma_1}{\alpha_1}$ and $2 + \frac{\gamma_2}{\mu_1} \geq \frac{\gamma_2}{\mu_1} + \frac{\gamma_1}{\alpha_1}$, leading to $\mathcal{O}\left(\epsilon^{-2-\frac{\gamma_2}{\mu_1}}\right) \geq \mathcal{O}\left(\epsilon^{-\frac{\gamma_2}{\mu_1}-\frac{\gamma_1}{\alpha_1}}\right)$. When $\beta_1 < \gamma_1$, the condition ensures $2 \geq \frac{\beta_1}{\alpha_1}$ and $2 + \frac{\gamma_2}{\mu_1} + \frac{\gamma_1-\beta_1}{\alpha_1} \geq \frac{\gamma_2}{\mu_1} + \frac{\gamma_1}{\alpha_1}$, leading to $\mathcal{O}\left(\epsilon^{-2-\frac{\gamma_2}{\mu_1}-\frac{\gamma_1-\beta_1}{\alpha_1}}\right) \geq \mathcal{O}\left(\epsilon^{-\frac{\gamma_2}{\mu_1}-\frac{\gamma_1}{\alpha_1}}\right)$. When $\beta_1 = \gamma_1$, it follows since $\mathcal{O}\left(\epsilon^{-2-\frac{\gamma_2}{\mu_1}} (\log\epsilon)^2\right) \geq \mathcal{O}\left(\epsilon^{-2-\frac{\gamma_2}{\mu_1}}\right)$. □

The theorem states that if the variance decay rate β_1 exceeds γ_1, which reflects the cost per sample of surrogate evaluation, then the primary sampling occurs on the coarsest grid. If the decay rate is less than γ_1, the majority of the computational effort will be concentrated on the finest grid. Finally, when these two rates equal, the computational workload is evenly spread among all grid levels.

Theorem 4. *Suppose there exist positive constants* $\alpha_1, \alpha_2, \beta_1, \gamma_1, \gamma_2, \gamma, \mu_1$ *with* $\xi_1 = \gamma_1 - \frac{(\alpha_2+\gamma)\gamma_2}{\mu_1+1}$, $\xi_2 = \frac{\mu_1\gamma-\alpha_2}{\mu_1+1}$, *such that* $2\alpha_1 \geq \min(\beta_1, \xi_1)$,

(i) $\|u - u_\ell\|_{L^2(W,Z)} = \mathcal{O}(M_\ell^{-\alpha_1})$,

(ii) $\|Y_\ell - \widehat{Y}_{q,\ell}\|_{L^2(W,Z)} = \mathcal{O}\left(M_\ell^{-\alpha_2} P_q^{-\mu_1}\right)$,

(iii) $V_\ell = \mathcal{O}\left(M_\ell^{-\beta_1}\right)$,

(iv) $C_\ell = \mathcal{O}\left(M_\ell^{\gamma}\right)$, $C_\ell^e = \mathcal{O}\left(M_\ell^{\gamma_1} P_q^{\gamma_2}\right)$.

Then for any positive $\epsilon < e^{-1}$ small enough, there exists spatial grid level L, sparse grid level $q(\ell)$ and sample size N_ℓ for which the multilevel estimator A_{MLMC} has an nMSE with

$$\frac{\|\mathbb{E}(u) - A_{MLMC}(\widehat{u}_L)\|^2_{L^2(W,Z)}}{\|\mathbb{E}(u)\|^2_{L^2(W,Z)}} < \epsilon^2.$$

Let $\xi_3 = \max(0, \xi_2)$, $\xi_4 = \frac{\gamma_2}{\mu_1}\chi_{(\xi_2=0)}$, and the total sampling cost with bound

$$\mathscr{W}^{on}_{\text{ML-ML}} = \begin{cases} \mathcal{O}\left(\epsilon^{-2-\frac{\gamma_2}{\mu_1}\left(1+\frac{\xi_3}{\alpha_1}\right)} |\log\epsilon|^{\xi_4}\right), & \beta_1 > \xi_1, \\ \mathcal{O}\left(\epsilon^{-2-\frac{\gamma_2}{\mu_1}\left(1+\frac{\xi_3}{\alpha_1}\right)} |\log\epsilon|^{\xi_4+2}\right), & \beta_1 = \xi_1, \\ \mathcal{O}\left(\epsilon^{-2-\frac{\gamma_2}{\mu_1}\left(1+\frac{\xi_3}{\alpha_1}\right)-\frac{\xi_1-\beta_1}{\alpha_1}} |\log\epsilon|^{\xi_4}\right), & \beta_1 < \xi_1. \end{cases}$$

Proof. To estimate the sampling cost for MLMC-FE with the multilevel in both parameter and spatial grids surrogate, we substitute the corresponding number of sparse grid nodes (4.17) into C^e_ℓ and V_ℓ within the general cost formula (5.31), resulting in the expressions

$$\begin{aligned} C^e_\ell &\sim M_\ell^{\gamma_1} P_q^{\gamma_2} = \left(\frac{C_p}{\theta_2\epsilon}\right)^{\frac{\gamma_2}{\mu_1}} M_\ell^{\gamma_1 - \frac{(\alpha_2+\gamma)\gamma_2}{\mu_1+1}} \left(\sum_{\ell=0}^{L} M_\ell^{\frac{\mu_1\gamma-\alpha_2}{\mu_1+1}}\right)^{\frac{\gamma_2}{\mu_1}}, \\ &= \left(\frac{C_p}{\theta_2\epsilon}\right)^{\frac{\gamma_2}{\mu_1}} M_0^{\gamma_1 - \frac{\alpha_2\gamma_2}{\mu_1}} s^{\ell\xi_1} \left(\sum_{\ell=0}^{L} s^{\ell\xi_2}\right)^{\frac{\gamma_2}{\mu_1}}, \\ V_\ell &\sim M_\ell^{-\beta_1} = M_0^{-\beta_1} s^{-\ell\beta_1}, \end{aligned}$$

where $\xi_1 = \gamma_1 - \frac{(\alpha_2+\gamma)\gamma_2}{\mu_1+1}$, $\xi_2 = \frac{\gamma\mu_1-\alpha_2}{\mu_1+1}$. Following the procedure to analyze cost in Theorem 1 and Theorem 3, the sampling cost of MLMC-FE with the multilevel in both parameter and spatial grids surrogate can also be decomposed into two components reflecting the main (H_1) and minor (H_2) costs such that $H_1 \leq \mathscr{W}^{\text{off}}_{\text{ML-ML}} \leq H_1 + H_2$, with

$$\begin{aligned} H_1 &= \frac{1}{\theta\epsilon^2}\left(\sum_{\ell=0}^{L}\sqrt{V_\ell C^e_\ell}\right)^2 \sim \frac{1}{\theta\epsilon^2}\left(\frac{C_p}{\theta_2\epsilon}\right)^{\frac{\gamma_2}{\mu_1}} M_0^{\gamma_1-\beta_1-\frac{\alpha_2\gamma_2}{\mu_1}} \left(\sum_{\ell=0}^{L} s^{\ell\xi_2}\right)^{\frac{\gamma_2}{\mu_1}} \left(\sum_{\ell=0}^{L} s^{\frac{\ell(\xi_1-\beta_1)}{2}}\right)^2, \\ &\sim \epsilon^{-\frac{\gamma_2}{\mu_1}-2}\,[B_2]^{\frac{\gamma_2}{\mu_1}}\,[B_3]^2, \\ H_2 &= \sum_{\ell=0}^{L} C^e_\ell \sim \left(\frac{C_p}{\theta_2\epsilon}\right)^{\frac{\gamma_2}{\mu_1}} M_0^{\gamma_1-\frac{\alpha_2\gamma_2}{\mu_1}} \left(\sum_{\ell=0}^{L} s^{\ell\xi_1}\right)\left(\sum_{\ell=0}^{L} s^{\ell\xi_2}\right)^{\frac{\gamma_2}{\mu_1}} \sim \epsilon^{-\frac{\gamma_2}{\mu_1}}\,[B_1]\,[B_2]^{\frac{\gamma_2}{\mu_1}}, \end{aligned}$$

where $B_1 = \sum_{\ell=0}^{L} s^{\ell\xi_1}$, $B_2 = \sum_{\ell=0}^{L} s^{\ell\xi_2}$, $B_3 = \sum_{\ell=0}^{L} s^{\ell(\xi_1-\beta_1)/2}$. Using (A.1), these terms can be further represented as

$$B_1 = \begin{cases} \mathcal{O}(1), & \xi_1 < 0 \\ \mathcal{O}\left(|\log \epsilon|\right), & \xi_1 = 0 \\ \mathcal{O}\left(\epsilon^{-\frac{\xi_1}{\alpha_1}}\right), & \xi_1 > 0 \end{cases}, \quad B_2 = \begin{cases} \mathcal{O}(1), & \xi_2 < 0 \\ \mathcal{O}\left(|\log \epsilon|\right), & \xi_2 = 0 \\ \mathcal{O}\left(\epsilon^{-\frac{\xi_2}{\alpha_1}}\right), & \xi_2 > 0 \end{cases}, \quad B_3 = \begin{cases} \mathcal{O}(1), & \xi_1 < \beta_1 \\ \mathcal{O}\left(|\log \epsilon|\right), & \xi_1 = \beta_1 \\ \mathcal{O}\left(\epsilon^{-\frac{\xi_1-\beta_1}{2\alpha_1}}\right), & \xi_1 > \beta_1 \end{cases}. \tag{5.34}$$

The upper bound of the total cost is then expressed as

$$H_1 + H_2 \sim \epsilon^{-\frac{\gamma_2}{\mu_1}} \left[B_2\right]^{\frac{\gamma_2}{\mu_1}} \left(\epsilon^{-2}\left[B_3\right]^2 + \left[B_1\right]\right). \tag{5.35}$$

To derive the assumption stated in Theorem 4, we rely on the observation that the complexity of the minor cost H_2 is smaller than the primary cost H_1, which implies that the term $\sum_{\ell=0}^{L} C_\ell^e$ is negligible when compared to $\frac{1}{\theta\epsilon^2}\left(\sum_{\ell=0}^{L}\sqrt{V_\ell C_\ell^e}\right)^2$ attributed to the ceiling function. Consequently, both lower and upper bounds of H_1 reflect the behavior of the total sampling cost $\mathscr{W}_{\text{ML-ML}}^{\text{off}}$. Define $B_4 := \epsilon^{-2}\left[B_3\right]^2$, which is written explicitly as

$$B_4 = \begin{cases} \mathcal{O}(\epsilon^{-2}), & \xi_1 < \beta_1 \\ \mathcal{O}\left(\epsilon^{-2}\left|\log \epsilon\right|^2\right), & \xi_1 = \beta_1 \\ \mathcal{O}\left(\epsilon^{-2-\frac{\xi_1-\beta_1}{\alpha_1}}\right), & \xi_1 > \beta_1 \end{cases}.$$

In (5.35), the complexities of H_1 and H_2 are reflected through B_4 and B_1. As per (5.34), B_1 attains a maximum complexity of $\mathcal{O}\left(\epsilon^{-\frac{\xi_1}{\alpha_1}}\right)$ when $\xi_1 > 0$. In this context, we need all three scenarios of B_4 to dominate $\mathcal{O}\left(\epsilon^{-\frac{\xi_1}{\alpha_1}}\right)$. If $\xi_1 < \beta_1$, we require $B_4 = \mathcal{O}(\epsilon^{-2})$ to dominate the complexity of $\mathcal{O}\left(\epsilon^{-\frac{\xi_1}{\alpha_1}}\right)$, which indicates that $2 > \frac{\xi_1}{\alpha_1}$ and $2\alpha_1 \geq \min\left(\xi_1, \beta_1\right)$. If $\xi_1 > \beta_1$, dominance of $B_4 = \mathcal{O}\left(\epsilon^{-2-\frac{\xi_1-\beta_1}{\alpha_1}}\right)$ over $\mathcal{O}\left(\epsilon^{-\frac{\xi_1}{\alpha_1}}\right)$ requires $2 + \frac{\xi_1-\beta_1}{\alpha_1} > \frac{\xi_1}{\alpha_1}$ and $2\alpha_1 \geq \min\left(\xi_1, \beta_1\right)$. If $\xi_1 = \beta_1$, considering that $\mathcal{O}\left(\epsilon^{-2}\right) < \mathcal{O}\left(\epsilon^{-2}\left|\log \epsilon\right|^2\right)$, the assumption is satisfied since the complexity $\xi_1 < \beta_1$ of B_4 hold. Thus, we yield the assumption for Theorem 4 as

$$2\alpha_1 \geq \min\left(\xi_1, \beta_1\right).$$

Under this condition, the dominant cost of $H_1 + H_2$ is H_1. Ignoring the contribution of H_2, the total sampling cost using H_1 is derived based on various scenarios of B_2 and B_3 as summarized in Table 5.1.

	$\xi_2 < 0$	$\xi_2 = 0$	$\xi_2 > 0$
$\xi_1 < \beta_1$	$\mathcal{O}\left(\epsilon^{-2-\frac{\gamma_2}{\mu_1}}\right)$	$\mathcal{O}\left(\epsilon^{-2-\frac{\gamma_2}{\mu_1}} \lvert\log\epsilon\rvert^{\frac{\gamma_2}{\mu_1}}\right)$	$\mathcal{O}\left(\epsilon^{-2-\frac{\gamma_2}{\mu_1}\left(1+\frac{\xi_2}{\alpha_1}\right)}\right)$
$\xi_1 = \beta_1$	$\mathcal{O}\left(\epsilon^{-2-\frac{\gamma_2}{\mu_1}} \lvert\log\epsilon\rvert^2\right)$	$\mathcal{O}\left(\epsilon^{-2-\frac{\gamma_2}{\mu_1}} \lvert\log\epsilon\rvert^{\frac{\gamma_2}{\mu_1}+2}\right)$	$\mathcal{O}\left(\epsilon^{-2-\frac{\gamma_2}{\mu_1}\left(1+\frac{\xi_2}{\alpha_1}\right)} \lvert\log\epsilon\rvert^2\right)$
$\xi_1 > \beta_1$	$\mathcal{O}\left(\epsilon^{-2-\frac{\gamma_2}{\mu_1}-\frac{\xi_1-\beta_1}{\alpha_1}}\right)$	$\mathcal{O}\left(\epsilon^{-2-\frac{\gamma_2}{\mu_1}-\frac{\xi_1-\beta_1}{\alpha_1}} \lvert\log\epsilon\rvert^{\frac{\gamma_2}{\mu_1}}\right)$	$\mathcal{O}\left(\epsilon^{-2-\frac{\gamma_2}{\mu_1}\left(1+\frac{\xi_2}{\alpha_1}\right)-\frac{\xi_1-\beta_1}{\alpha_1}}\right)$

Table 5.1: The value of H_1 generated from different scenarios of B_2 and B_3.

In summary, let $\xi_3 = \max(0, \xi_2)$, $\xi_4 = \frac{\gamma_2}{\mu_1}\chi_{(\xi_2=0)}$. The total cost is then

$$H_1 = \begin{cases} \mathcal{O}\left(\epsilon^{-2-\frac{\gamma_2}{\mu_1}\left(1+\frac{\xi_3}{\alpha_1}\right)} \lvert\log\epsilon\rvert^{\xi_4}\right), & \xi_1 < \beta_1, \\ \mathcal{O}\left(\epsilon^{-2-\frac{\gamma_2}{\mu_1}\left(1+\frac{\xi_3}{\alpha_1}\right)} \lvert\log\epsilon\rvert^{\xi_4+2}\right), & \xi_1 = \beta_1, \\ \mathcal{O}\left(\epsilon^{-2-\frac{\gamma_2}{\mu_1}\left(1+\frac{\xi_3}{\alpha_1}\right)-\frac{\xi_1-\beta_1}{\alpha_1}} \lvert\log\epsilon\rvert^{\xi_4}\right), & \xi_1 > \beta_1. \end{cases}$$

□

The theorem states that when the decay rate of the variance β_1 exceeds ξ_1, which serves as a proxy for the cost per sample in multilevel surrogate evaluation, the primary sampling cost is incurred on the coarsest grid. Conversely, the majority of the computational effort is concentrated on the finest grid. When β_1 equals ξ_1, the computational workload is evenly distributed across all grid levels.

When comparing the two sampling costs across three scenarios, it's evident that the proxy ξ_1 consistently remains less than γ_1. Specifically, when the decay rate of variance β_1 surpasses γ_1, the complexity of the multilevel strategy deviates from that of the single-level strategy by a factor of $O(\epsilon^{-\frac{\gamma_2\xi_3}{\mu_1\alpha_1}} |\log\epsilon|^{\xi_4})$. In cases where the product $\mu_1\gamma$ equals the rate α_2, reflecting the convergence of the correction term $\widehat{Y}_{q,l}$ with respect to M_ℓ, resulting in $\xi_2 = 0$, the positive part of ξ_2 is zero ($\xi_3 = 0$), and $\xi_4 = \gamma_2/\mu_1$ – which shows the ratio between the convergence of correction $\widehat{Y}_{q,l}$ and the cost per sample with respect to P_q. A ratio of ξ_3/α_1 below $1/e$ signifies an efficiency improvement with the multilevel strategy due to reduced complexity. Both strategies exhibit comparable complexities when the growth rate of cost per sample for direct solver γ is less than the ratio α_2/μ_1 that reflects the convergence rate of surrogate correction. However, for $\mu_1\gamma$ greater than α_2 ($\xi_2 > 0$), the complexity of the multilevel strategy escalates by a factor of $O(\epsilon^{-\frac{\gamma_2\xi_2}{\mu_1\alpha_1}})$. Notably, for ϵ values

not extremely small, the multilevel surrogate outperforms the single-level surrogate owing to a small constant $(1-s^{-\alpha_2})^{-\gamma_2/\mu_1}$ in the sampling cost. However, as tolerance ϵ diminishes, the complexity of the multilevel strategy eventually surpasses that of the single-level strategy.

The total sampling cost is contingent on both the sample size estimation and the cost per sample for surrogate evaluation, with the latter reliant on the number of sparse grid nodes used to construct the surrogate. Consequently, the sampling cost is intricately linked to the surrogate employed. The optimization of the sampling cost is manifested through the sample size estimation in equation (5.14), whereby the same sample size formula (5.30) implies that the sampling cost is optimized for both single-level and multilevel surrogates. Although the multilevel strategy is tailored to minimize both sampling expenses and construction costs, it's important to note that the surrogate with the optimized construction cost may not necessarily yield the minimum sampling costs across all potential surrogate choices. For instance, as discussed previously, in scenarios where the variance decay rate β_1 surpasses γ_1 and $\gamma \leq \alpha_2/\mu_1$, the cost of the single-level strategy outperforms that of the multilevel approach. This indicates that the cost of the multilevel strategy may not always be the minimum among the sampling costs of different surrogates, such as the single-level strategy.

Theorem 5. *Suppose there exist positive constants $\alpha_1, \beta_1, \gamma_1$ such that $2\alpha_1 \geq \min(\beta_1, \gamma_1)$,*

(i) $\|\mathbb{E}\left(u - u_\ell\right)\|_Z = \mathcal{O}\left(M_\ell^{-\alpha_1}\right)$,

(ii) $V_\ell = \mathcal{O}\left(M_\ell^{-\beta_1}\right)$,

(iii) $C_\ell^e = \mathcal{O}\left(M_\ell^{\gamma_1}\right)$.

Then for any positive $\epsilon < e^{-1}$ small enough, there exists level L and sample size N_ℓ for which the multilevel estimator A_{MLMC} has an nMSE with

$$\frac{\|\mathbb{E}(u) - A_{MLMC}(\widehat{u}_L)\|^2_{L^2(W,Z)}}{\|\mathbb{E}(u)\|^2_{L^2(W,Z)}} < \epsilon^2,$$

and the total sampling cost with bound

$$\mathscr{W}^{on}_{SL\text{-}ML} = \begin{cases} \mathcal{O}\left(\epsilon^{-2}\right), & \beta_1 > \gamma_1, \\ \mathcal{O}\left(\epsilon^{-2}\left|\log \epsilon\right|^2\right), & \beta_1 = \gamma_1, \\ \mathcal{O}\left(\epsilon^{-2-\frac{\gamma_1-\beta_1}{\alpha_1}}\right), & 0 < \beta_1 < \gamma_1. \end{cases}$$

Algorithm 3: MLMC-FE

Input: Initial mesh level $L = 0$, sequence of mesh available $\{\mathcal{T}_\ell\}_{\ell\geq 0}$, root n-MSE ϵ, $\zeta \in (0,1)$, initial sample size $N_{\text{old}} = \{N_\ell\}_{\ell=0}^L$, counter $j = 1$, sample size corrections $dN_\ell^j = \{N_\ell\}_{\ell=0}^L$, surrogates.

Output: $\{N_\ell\}_{\ell=0}^L$, $\{V_\ell\}_{\ell=0}^L$, A_{MLMC}.

while $\sum_\ell (dN_\ell^j) > 0$ **do**

  for $0 \leq \ell \leq L$ **do**

    for $i = 1, \cdots, dN_\ell^j$ **do**

      EVALUATE the surrogate on $\mathcal{T}_\ell$ using the corresponding sparse grid level q to obtain $\widehat{u}_{q,\ell}^{(i)}$ for the i-th sample.

  Approximate $\{V_\ell\}_{\ell=0}^L$ by (5.32).

  Update the sample size estimation $\{N_\ell\}_{\ell=0}^L$ by (5.30).

  $j = j + 1$.

  $dN_\ell^j = \{N_\ell\}_{\ell=0}^L - N_{\text{old}}$.

  $N_{\text{old}} = \{N_\ell\}_{\ell=0}^L$.

  if $\sum_\ell (dN_\ell^j) = 0$ **then**

    if The discretizaton error $\|u - u_L\|_{L^2(W,Z)}$ falls below $\theta_1 \epsilon \|\mathbb{E}(u)\|_Z$, **then**

      Compute A_{MLMC} by (5.27) and terminate the loop.

    else

      $L = L + 1$.

      Approximate V_L by (5.33) and compute $\{N_\ell\}_{\ell=0}^L$ and go to Step 1.

Chapter 6: Numerical experiments

In this chapter, we will numerically test three sampling methodologies introduced in Chapter 1 to efficiently approximate the expectation in (2.14) as

$$\mathbb{E}\left[\psi(r,z,\boldsymbol{\omega})\right] = \int_W \psi(r,z,\boldsymbol{\omega})\pi(\boldsymbol{\omega})d\boldsymbol{\omega},$$

Following [21], we consider an ITER geometry with 12 coils and a "baseline" vector of target current intensities $\boldsymbol{I}$ given by

$$\begin{aligned}
&I_1 = -1.4\times 10^6 A, && I_2 = -9.5\times 10^6 A, && I_3 = -2.0388\times 10^7 A,\\
&I_4 = -2.0388\times 10^7 A, && I_5 = -9\times 10^6 A, && I_6 = 3.564\times 10^6 A,\\
&I_7 = 5.469\times 10^6 A, && I_8 = -2.266\times 10^6 A, && I_9 = -6.426\times 10^6 A,\\
&I_{10} = -4.82\times 10^6 A, && I_{11} = -7.504\times 10^6 A, && I_{12} = 1.724\times 10^7 A.
\end{aligned} \tag{6.1}$$

Figure 6.1 shows the numbering of the coils; each component of the vector $\mathbf{I} = [I_1, \ldots, I_{12}]$ corresponds to the current value going through the coil with the same number. We will refer to these values as the *reference currents*. The profiles for p and g on the right hand side of (2.5a) follow the form given in (2.5c) with the specific values $r_0 = 6.2m$, $\beta = 0.5978$, $\alpha_1 = 2$, and $\alpha_2 = 1.395$. In our experiments, we will take the vector of current intensities to be subject to uncertainty modeled as a uniformly distributed perturbation of relative magnitude τ centered around the reference values above.

For the solution of (2.5), we used the finite element-based solver `FEEQS.m` [34] developed by Holger Heumann and collaborators as a lightweight Matlab implementation of the code `CEDRES++` [21, 35]. The code implements a piecewise linear finite element discretization of a weak formulation of (2.5) and employs a globalized variation of Newton's method to resolve the nonlinearity. Surrogate functions for the parameter-dependent discrete stream function ψ_h were built using increasing levels of sparse grids as described in Chapter 3 and Chapter 4.

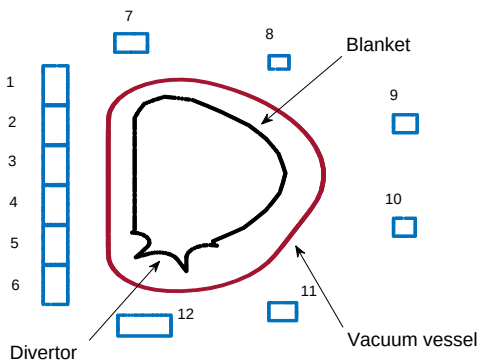

Figure 6.1: Reactor geometry used for the numerical simulations. The divertor, inner wall (black line), and outer wall of the vacuum vessel (red line) are shown along with the numbered coils.

Before describing the experiments, we comment on one aspect of our particular application that is affected by the use of the surrogate-based collocation. A key step in the computations is the determination of the location of the x-point. This is the point where the streamline of the magnetic field crosses itself, and therefore determines the separatrix between the region of the plane where the lines are closed (allowing for confinement) and the region where they are open. As such, in the absence of disrupting structures, the segment of the separatrix describing a closed curve determines the plasma boundary Γ. Geometrically, the x-point corresponds to a saddle point on the graph of the scalar field ψ; this characteristic is used to determine its location on the discrete approximations ψ_h coming from the direct solver `FEEQS.m` and $\widehat{\psi}_h$ coming from the evaluation of the surrogate.

By definition, a saddle point must be a local maximum in one direction, and a local minimum in another one. A discrete version of this fact is implemented in `FEEQS.m` and used as the criterion to identify candidates for the location of the x-point. An approximation to the directional derivative of the piecewise linear approximation ψ_h in the direction of each of the element edges originating at an element vertex $\mathbf{v}$ is given by the difference $\psi_h(\mathbf{v}) - \psi_h(\mathbf{v}_i)$, where the subindex i on $\mathbf{v}_i$ runs over all the mesh nodes connected to $\mathbf{v}$ by an edge. At a saddle point of a piecewise linear approximation, the directional derivative must change sign at least four times—this situation is depicted in Figure 6.2. The code computes the number of sign changes of the directional derivative for every node within the reactor and identifies those vertices where the sign changes at least four times as candidates for the saddle

Figure 6.2: In a piecewise linear approximation of ψ, the saddle points are located at those element vertices $\mathbf{v}$ for which the difference $\psi_h(\mathbf{v}) - \psi_h(\mathbf{v}_i)$ for all its neighboring vertices $\mathbf{v}_i$ changes sign at least four times. The sign of this difference indicates whether the discrete function ψ_h along the element vertices is increasing or decreasing. The point surrounded by the arrows in the figure corresponds to a saddle, as it is a local maximum in one direction and a local minimum in another one. This is reflected by the number of times the arrows change direction.

point. For numerical solutions coming out of the direct solver this criterion is typically enough to identify the discrete x-point correctly. However, due to the oscillatory nature of polynomial interpolation, evaluations of the surrogate require some additional work. In our computations the following heuristic was implemented to choose between several possible candidates. Using a classical gradient recovery technique introduced by Zienkiewicz and Zhu [75, 76], it is possible to build a second-order approximation to $\nabla\psi$ from the nodal values of $\widehat{\psi}_h$ at each vertex and its neighbors. If more than one node satisfies the "alternating sign" criterion, then the magnitude of the recovered gradient is evaluated at each candidate node and all of its neighbors. The candidate node with the smallest local average magnitude is then identified as the saddle point. The logic behind this strengthened criterion is that the spurious oscillations originating from the surrogate evaluations are highly localized in nature, and therefore must present larger gradients than the ones associated with the underlying function ψ.

6.1 Monte Carlo sampling with both direct computation and surrogate

For the initial experiment, we will examine a single Monte Carlo sampling method using both the direct solver and the surrogate function.

6.1.1 Experiment description

Our investigation is structured around three main aspects. First, we aim to determine the probability that the magnetic field will be conducent to confinement, i.e., whether the level sets of ψ produce closed lines completely contained in the vacuum chamber. If confinement is achieved, we explore the nature of the plasma boundary, exploring whether it's defined by an uninterrupted separatrix (in which case the locations of the x-point and strike points are of relevance) or by contact with any structure within the reactor (in which case the location of the contact point is also of interest). Finally, for all cases where confinement is achievable (regardless of the presence of an x-point or a contact point), we examine the geometric properties of the plasma boundary such as aspect ratio, elongation, and triangularity, as defined in Chapter 2.1.3.

6.1.2 Adaptive solver

In this experiment, to enhance the resolution and accuracy of the solution provided by `FEEQS.M`, we focus on an adaptive solver that employs a local refinement strategy. Considering that each realization of the solution requires several steps of a Newton iteration and that a large number of realizations is necessary for parameter sampling, the additional cost associated with uniform refinement can quickly become unmanageable.

Given that in our application the most relevant quantities are related to the location of the plasma boundary, we resort instead to the following local refinement strategy. The free boundary problem is solved on a coarse grid and the approximate location of the plasma boundary is determined by first finding the location of either the x-point, in the case where the plasma is not in contact with the reactor, or of the point where the streamlines are tangential to a disrupting structure. Let us denote the coordinates of either of these points by (x^*, y^*). Since the plasma boundary corresponds to the level set of $\psi_h(x^*, y^*)$, the elements in the neighborhood of the separatrix are estimated by fixing $\alpha^* \in (0, 1)$ and identifying all the nodes (x, y) where

$$|\,\psi_h(x, y) - \psi_h(x^*, y^*)\,| \leq \alpha^* |\,\psi_h(x^*, y^*)\,|.$$

All the elements having a vertex that satisfies the above condition are then marked and refined—in our experiments we set $\alpha^* = 0.05$. The new mesh has a band of finer elements

in the vicinity of the separatrix. The problem is then solved again on the finer mesh and the process can be repeated. In our experiments, the adaptive cycle is stopped after two iterations of local refinement. Given that, whenever they are present, the x-point, the strike points or the contact points are located in the vicinity of the separatrix, the local refinement leads to a better estimation of their positions. This is especially true for the location of the x-points, which are bound to element vertices: by enriching the mesh with additional vertices more possible locations become accessible to the x-points. Figure 6.3 shows a refinement cycle consisting of an initial coarse grid and two levels of adaptive refinement. With refinement limited to a neighborhood of the separatrix, the growth in system unknowns is curtailed while resolution is improved. If needed, the growth of the number of degrees of freedom can be limited even more by decreasing the value of the marking parameter α^* as the mesh gets finer—in our experiments, the value of $\alpha^* = 0.05$ was kept fixed across refinement levels.

Since the free boundary problem must be solved at every level of the adaptive algorithm, the solution obtained on a coarser level is interpolated to the new grid and used as an initial guess for the Newton iteration. Moreover, the stopping tolerance for the Newton iterations at coarse refinement levels can be set to be relatively large on the initial coarse grid and reduced dynamically as the mesh is refined. As stopping criterion for the Newton process, the code `FEEQS.M` measures the magnitude of the nonlinear updates to the solution operator relative to the norm of the matrix arising from the discretization of (2.11). In our experiments, the tolerance for the adaptive solver with R refinements was determined by the formula $\mathrm{TOL} = 10^{-11((i+1)/(R+1))}$. In the sequel, we will refer to `FEEQS.M` as the direct solver and to the numerical solutions provided by this code as direct solutions.

Our ultimate goal is to build a surrogate model based on interpolation between numerical solutions associated with a small number of carefully selected current values (specific details are given in Chapter 3). However, since the values of the currents passing through the coils determine the location of the plasma boundary, different realizations will result in different adaptively refined meshes. Therefore, the corresponding numerical approximations will belong to different discrete spaces and the different solutions must be projected into a common space (a common mesh) prior to the construction of the surrogate. In our experiments, all the numerical solutions start from a common coarse grid that is adaptively refined around the boundary of the plasma in each realization. The numerical solutions are then L^2-projected into a finer common mesh built by uniformly refining the interior of the

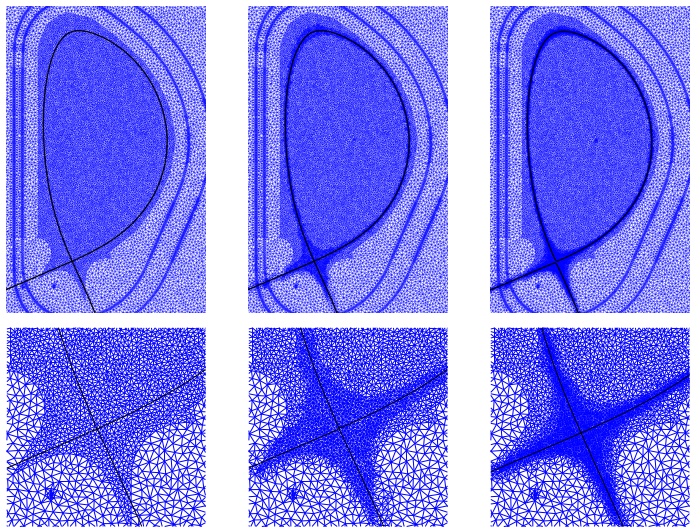

Figure 6.3: Top row: The free boundary problem is solved in a sequence of refined meshes, starting from an initial coarse mesh (left). The location of the separatrix drives the refinement strategy and at every level a banded region around it is marked for refinement. Bottom row: Close-up of the neighborhood of the magnetic x-point for the three refinement levels shown at the top.

reactor from the initial, common, coarse mesh. The surrogate obtained in this way is then, by construction, defined on the finer common mesh.

This step introduces an additional projection error that is initially negligible compared to the surrogate error but can eventually become dominant as the surrogate gets refined. To curtail the projection error, in our experiments the common mesh is refined as many times as the prescribed depth of the adaptive algorithm. It is important to note that no numerical solutions are ever computed on the finer common grid; instead only surrogate evaluations are needed. As we shall show later, the time required for such evaluations is often considerably shorter than the time required for the solution of the free boundary problem, even on the coarsest grid.

6.1.3 Accuracy study

We now present the results of numerical experiments that illustrate the accuracy and cost savings obtained using the surrogate in place of the direct solution of the discrete version of (2.5). We will first assess the approximation properties of the surrogate function by

comparing the point-wise difference between direct solutions and evaluations of surrogates built with increasing levels of refinement in the sparse grid. We will then test the surrogate's accuracy of reproducing physically relevant quantities such as locations of x-points or plasma shaping parameters, as well as its ability to predict unwanted interactions between the plasma and components of the reactor. These comparisons are done for identical sets of randomly chosen current values. After establishing the approximation capabilities of the surrogate function, we will explore its efficiency by comparing the timings associated with surrogate evaluations and direct solutions. We will introduce a dynamic sampling strategy designed to avoid unnecessarily large sample sizes, thus testing each of the two methods under ideal circumstances. Finally, we conclude with experiments that showcase the information that can be extracted from fast simulations made using the surrogate function.

We start with several tests to assess the accuracy of the surrogate approximation to ψ_h. To begin, we evaluated the surrogates at $n_s = 100$ random sample values of the currents $\{\mathbf{I}_i\}_{i=1}^{n_s}$, solved the free boundary problem at the same sample currents, and then we assess the error using the average of the relative error in the L_∞ norm

$$E_{n_s} := \frac{1}{n_s} \sum_{i=1}^{n_s} \frac{\|\widehat{\psi}_h(\mathbf{I}_i) - \psi_h(\mathbf{I}_i)\|_\infty}{\|\psi_h(\mathbf{I}_i)\|_\infty} \tag{6.2}$$

For this test, we considered perturbed versions of all twelve currents, as well as perturbations of subsets of them of sizes 2 through 6. The smaller subsets comprise parameter sets of smaller dimensions and enable the use of higher sparse-grid levels. The results of these tests for 1% variability are shown in Figure 6.4 and Table 6.1. They indicate some benefit of increasing the sparse grid level although these improvements are not monotone and they are not dramatic for levels greater than four. They also show that accuracy tends to be higher for smaller numbers of parameters; we attribute this to the logarithmic factor in the error bound (3.5). To single out the approximation properties of the surrogate and separate them from the projection error mentioned in Chapter 6.1.2, these convergence studies were performed using only a common grid without any adaptive refinement around the plasma boundary.

Next, we will explore how effective the surrogate model is for extracting information about the confinement properties of the magnetic field. To assess the effect that the stochasticity may have on physically relevant quantities, such as the location of x-points and the incidence of contacts between the plasma and the walls of the reactor, we will make use of a "reference

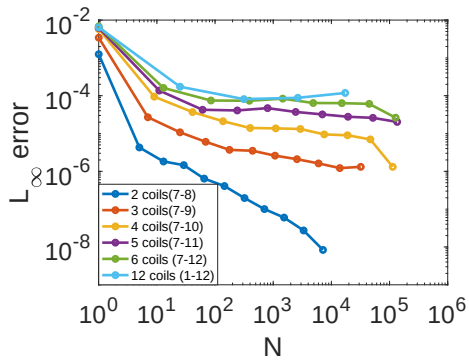

Figure 6.4: Mean relative difference between surrogate evaluations and numerical solutions to the free boundary problem of 100 randomly chosen current values, with 1% variability in currents. Results of the experiment for different numbers of currents perturbed are displayed in different colors.

	Coils 7-8		Coils 7-9		Coils 7-10		Coils 7-11		Coils 7-12		Coils 1-12	
Level	# Nodes	L_∞ error	# Nodes	L_∞ error	# Nodes	L_∞ error	# Nodes	L_∞ error	# Nodes	L_∞ error	# Nodes	L_∞ error
0	1	1.2f8e-03	1	3.42e-03	1	5.76e-03	1	6.22e-03	1	6.82e-03	1	6.44e-03
1	5	4.28e-06	7	2.69e-05	9	9.43e-05	11	1.37e-04	13	1.60e-04	25	1.73e-04
2	13	1.82e-06	25	1.08e-05	41	3.69e-05	61	4.25e-05	85	7.46e-05	313	8.13e-05
3	29	1.46e-06	69	6.04e-06	137	2.11e-05	241	4.07e-05	389	7.36e-05	2649	8.79e-05
4	65	6.42e-07	177	3.70e-06	401	1.40e-05	801	4.69e-05	1457	8.36e-05	17265	1.18e-04
5	145	4.08e-07	441	3.50e-06	1105	1.37e-05	2433	3.72e-05	4865	6.41e-05	93489	-
6	321	1.97e-07	1073	2.60e-06	2929	1.32e-05	6993	3.20e-05	15121	6.35e-05	442001	-
7	705	1.01e-07	2561	2.10e-06	7537	9.48e-06	19313	2.77e-05	44689	6.09e-05	1887377	-
8	1537	5.99e-08	6017	1.62e-06	18945	8.99e-06	51713	2.60e-05	127105	2.62e-05	7451393	-
9	3329	2.73e-08	13953	1.22e-06	46721	6.99e-06	135073	2.03e-05	350657	-	27649409	-
10	7169	8.35e-09	32001	1.31e-06	113409	1.31e-06	345665	-	943553	-	97566977	-

Table 6.1: Mean relative L_∞ errors (6.2) of surrogate solutions for 100 random sample currents, various sparse-grid levels and numbers of perturbed currents, with 1% noise level; entries in the table marked with a hyphen indicate no estimate due to computational limitations.

equilibrium configuration" corresponding to the ideal case where no variability is present in the coil currents. In the experiments used for this discussion, the surrogate was built using two levels of local spatial adaptivity in the free boundary solver, and surrogate solutions were obtained on a common grid constructed from two levels of uniform refinement. This ensures that the size of the smallest elements in the mesh where the surrogate is built is comparable to that of the adaptive grids where the samples at collocation nodes are taken.

Figure 6.5 depicts the distributions of x-points and strike points found by surrogate approximations determined from 2000 perturbed current values, with 1%, for three levels of sparse grids, together with the analogous results obtained from the direct solution. Visually, the results obtained from all instances of the surrogate computations are indistinguishable from those obtained from direct simulation. These results are quantified in Table 6.2, which counts the numbers of x-points that are found at various distances from a reference x-point, defined to be the point obtained from unperturbed input. The data, displayed in the bottom

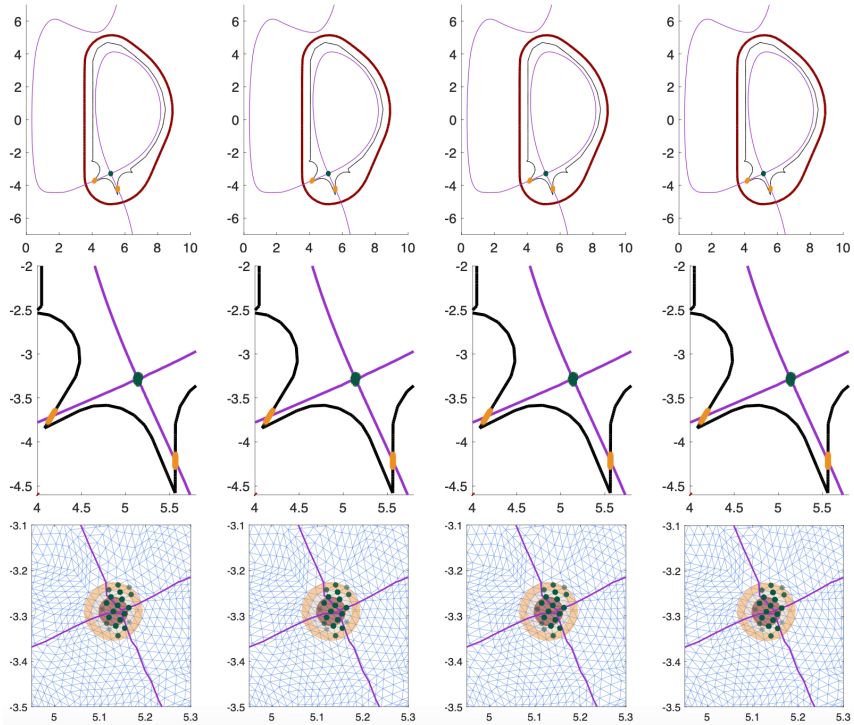

Figure 6.5: Distributions of x-points (green) and strike points (orange), for 2000 samples of perturbed current values with 1% noise. In all figures, the reference plasma boundary is displayed in purple, the inner wall of the reactor is displayed in black, and the outer walls of the reactor are displayed in dark red. From left to right, images correspond to surrogates obtained from sparse grid levels 2, 3 and 4 and direct solves. The top row shows (in purple) the separatrix obtained from the unperturbed problem. The second row shows the x-points and strike points in more detail. The third row shows further magnified neighborhoods of the x-points including the regions and perturbed x-points referenced in Table 6.2, and frequency of x-points corresponds to intensity of their displays.

	Number of x-points between r_{i-1} and r_i $(r_0 = 0)$			
	Surrogate level 2	Surrogate level 3	Surrogate level 4	Direct solver
$r_1 = 0.032 = .5r_3$	1455 (72.75%)	1457 (72.85%)	1459(72.95%)	1457 (72.85%)
$r_2 = 0.048 = .75r_3$	522 (26.10%)	521 (26.05%)	518 (25.90%)	520 (26.00%)
$r_3 = 0.064$	23 (1.15%)	22 (1.10%)	23 (1.15%)	23 (1.15%)

Table 6.2: Number (frequencies) of x-points found within various distances from the reference x-point, for 2000 perturbations with 1% noise in 12 coils. Counts correspond to number of points of distance at most r_1 or within annuli of width $r_2 - r_1$ or $r_3 - r_2$.

Level	# Nodes	E_{2000}	Inverse Aspect Ratio	Elongation	Upper Triangularity	Lower Triangularity	Strike Point 1	Strike Point 2	x-point
0	1	6.39e-03	2.76e-03	2.61e-03	1.65e-02	1.19e-02	3.96e-03	3.36e-03	3.34e-03
1	25	1.58e-04	2.81e-05	9.70e-05	8.92e-04	4.24e-04	4.77e-05	8.94e-06	6.32e-05
2	313	1.94e-04	2.59e-05	4.86e-05	4.52e-04	3.37e-04	3.19e-05	4.48e-06	5.01e-05
3	2649	2.54e-04	2.77e-05	5.11e-05	6.75e-04	2.97e-04	3.72e-05	8.67e-06	4.22e-05
4	17265	3.96e-04	3.49e-05	6.15e-05	1.31e-03	3.76e-04	4.69e-05	9.97e-06	5.47e-05

Table 6.3: Mean relative errors of various geometric features. The relative approximation error E_{n_s} is defined in (6.2). The plasma shaping parameters are defined in (2.6). The experiment used 2000 random samples with 1% perturbation of all 12 currents, for increasing collocation levels.

row of Figure 6.5, indicates that the trends for the surrogates and direct solutions are in fact quantitatively similar and give additional credence to the accuracy of the surrogates. Table 6.3 continues this exercise by showing the errors for a collection of geometric features of the plasma boundary defined in (2.6). The results indicate that reasonable accuracy is obtained from surrogates constructed using sparse grid levels 2 or 3.

Figure 6.6, and Table 6.4 and Table 6.5 show the results of similar experiments in which the size of the perturbations is increased to 2%. We highlight several trends that are somewhat different than those observed for smaller perturbations. First, the images of the resulting x-points found by the surrogate, while still very close to those obtained by direct solution, are no longer identical. This is confirmed by Table 6.4, which shows qualitative agreement in the locations of the perturbed x-points produced by surrogates and direct solutions but also reveals more x-points computed by surrogates that are somewhat further from the reference solution than those found by direct solution. Similar trends are evident for both strike points and contact points. In particular, Figure 6.6 shows that the surrogate

computations yield a small number (9 for sparse grid level 3, 12 for level 4) of false positives for strike points. The third row of images in Figure 6.6 shows examples of contact points found by each of the simulations. The direct solver and surrogates for sparse grid levels 3 and 4 all find the majority of contact points near $y = 1$; the surrogates for these levels also find a small number (on the order of 10 out of 2000) in other places in the domain.

	Number of x-points between r_{i-1} and r_i $(r_0 = 0)$			
	Surrogate level 2	Surrogate level 3	Surrogate level 4	Direct solver
$r_1 = 0.064 = .5r_3$	1819 (90.95%)	1773 (88.92%)	1723 (87.24%)	1814 (91.39%)
$r_2 = 0.096 = .75r_3$	168 (8.40%)	179 (8.98%)	209 (10.58%)	157 (7.91%)
$r_3 = 0.128$	13 (0.65%)	42 (2.11%)	43 (2.18%)	14 (0.71%)
	Number of plasma / reactor contacts			
	0	6	25	15

Table 6.4: Incidence of contacts between the plasma and the reactor, along with number (frequencies) of x-points found within various distances from the reference x-point, for 2000 perturbations with 2% noise in 12 coils. Counts of x-points correspond to number of points of distance at most r_1 or within annuli of width $r_2 - r_1$ or $r_3 - r_2$.

Level	# Nodes	E_{2000}	Inverse Aspect Ratio	Elongation	Upper Triangularity	Lower Triangularity
0	1	1.26e-02	5.46e-03	4.95e-03	2.93e-02	1.65e-02
1	25	2.69e-04	8.49e-05	3.02e-04	3.05e-03	1.11e-03
2	313	2.37e-04	4.19e-05	6.91e-05	3.56e-03	4.49e-04
3	2649	1.69e-03	7.98e-04	9.30e-04	6.44e-03	3.21e-03
4	17265	4.80e-03	2.41e-03	2.66e-03	9.10e-03	6.10e-03

Table 6.5: Mean relative errors of various geometric features. The relative approximation error E_{n_s} is defined in (6.2). The plasma shaping parameters are defined in (2.6). The experiment used 2000 random samples with 2% perturbation of all 12 currents, for increasing collocation levels.

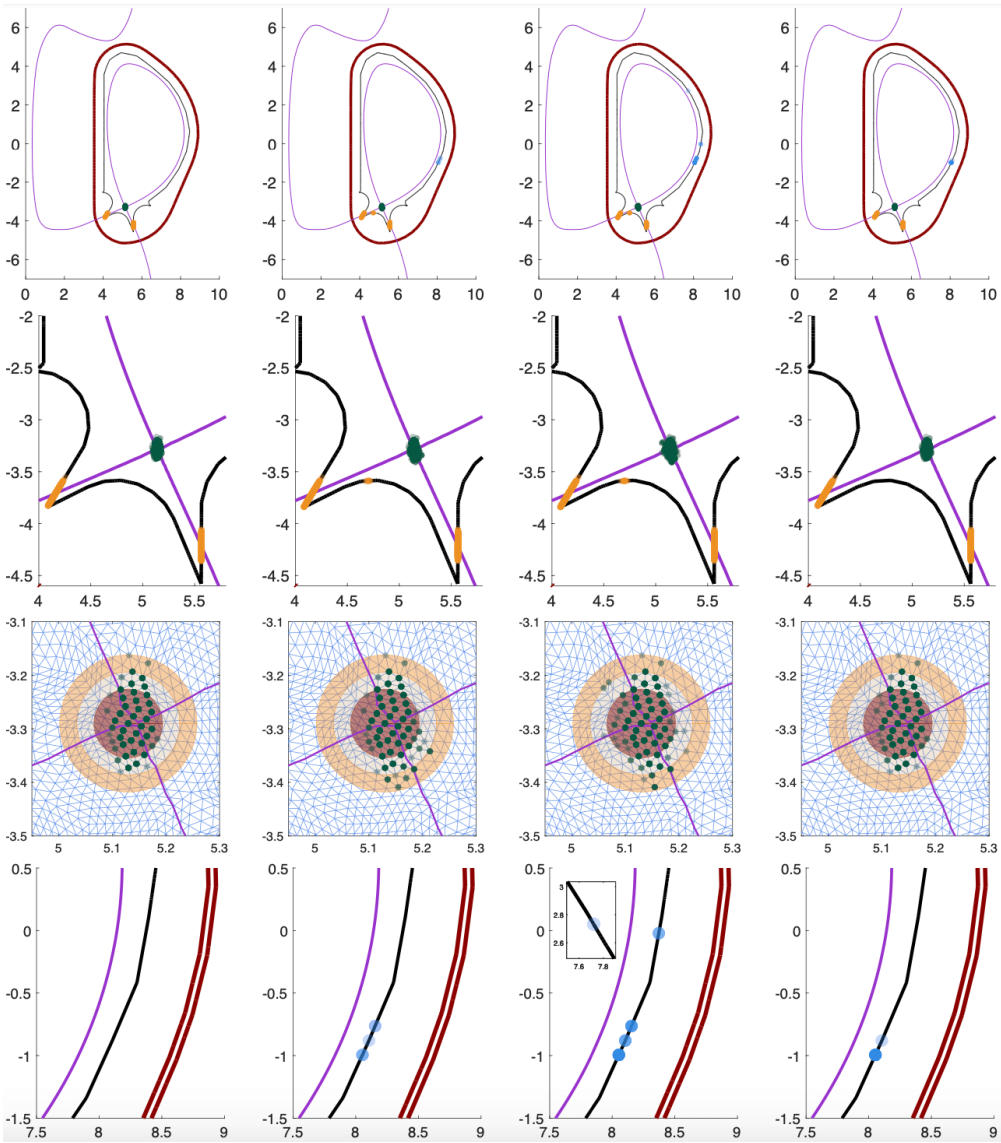

Figure 6.6: Distributions of x-points (green), strike points (orange) and contact points (blue), for 2000 samples of perturbed current values with 2% noise. In all figures, the reference plasma boundary is displayed in purple, the inner wall of the reactor is displayed in black, and the outer walls of the reactor are displayed in dark red. From left to right, images correspond to surrogates obtained from sparse grid levels 2, 3 and 4 and direct solves. The top row shows (in purple) the separatrix obtained from the unperturbed problem. The second row shows the x-points and strike points in more detail. The third row shows further magnified neighborhoods of the x-points including the regions and perturbed x-points referenced in Table 6.4, and the fourth row shows contact points. In the two bottom rows, frequency of x-points and contact points corresponds to intensity of their displays.

Note that all these results identify some effects that variability has on the properties of the solution such as the location of the x-point and strike points. Not surprisingly, increased noise in the currents leads to wider distributions of such quantities, and, for 2% noise, increased likelihood of the occurrence of contact points. The surrogate computations largely reproduce the results of the direct solution albeit with some loss of accuracy occurring with increased perturbation size. None of the examples lead to conclusions that would be dramatically misleading; for example, a false-positive strike point would lead to a slightly over-cautious approach when shielding the reactor.

The variability of the currents also has an effect on the spread of locations for the plasma boundary. To show how the surrogate function captures this behavior accurately, 2000 equilibria were calculated using both a direct solver and a surrogate built with 3 levels of the sparse grid. These computations were done using random perturbations of sizes 1% and

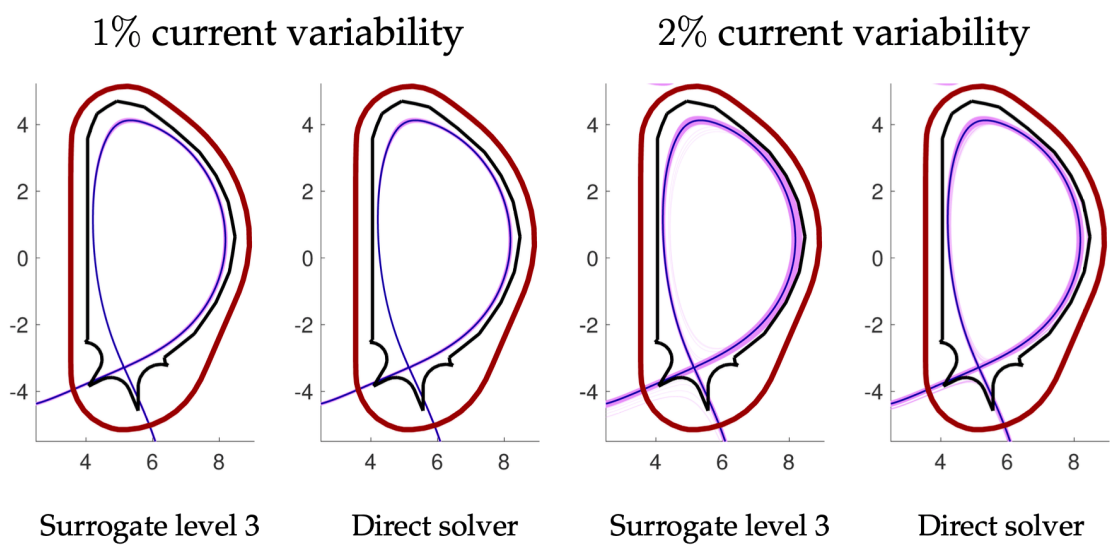

Figure 6.7: Each panel shows plasma boundaries from 2000 random coil current realizations. The solid violet line represents the boundary for reference current values, while light pink lines depict boundaries from random realizations. The inner reactor wall is in solid black, and the outer wall in dark red. Left: Level 3 surrogate function evaluations with 1% current variability. Center left: Direct solver solutions with 1% current variability. Center right: Level 3 surrogate function evaluations with 2% current variability. Right: Direct solver with 2% current variability.

2% about the reference currents. The results of the experiment are shown in Figure 6.7, where the plasma boundary from the reference current is plotted in dark violet, and the ones corresponding to the random perturbations are plotted in light pink and overlayed. In the figure it is evident that the qualitative behavior predicted by the surrogate function coincides with the one from the direct solution. The spread of the boundary focuses on the outer radius of the plasma (on the right side of the plot), increasing the likelihood for contacts on the outer side of the inner wall. This spread is more accentuated for surrogate evaluations, but located in the same regions as the direct solution, resulting in accurate—if slightly too cautious—predictions. This becomes even more evident for 2% noise, where it is also possible to observe isolated boundary curves resulting from plasma/wall contacts for both computational methods.

As the previous experiments have shown, the qualitative and quantitative agreement between direct computations and surrogate evaluations is significant. The key advantage of the use of these surrogate computations lies in their significantly smaller computational cost. These time savings come mostly from the fact that the surrogate bypasses the solution of the nonlinear system, but an additional source of savings comes from the fact that the surrogate can be designed to focus on smaller regions of interest. For the problem at hand, the direct solver is bound to provide an approximate solution over the entire computational domain, whereas for most applications, only the interior of the reactor is of interest. We will take advantage of this fact by constructing a surrogate that is defined only inside the reactor. This will give rise to smaller vectors and shorter computation times.

6.1.4 Efficiency study

We will demonstrate the time efficiency of the surrogate by comparing the performance of simulations done using the surrogate computations, for sparse grid levels 2 through 4, and the direct solver. In these tests, we used a dynamic approach for sampling based on the following ideas. The margin of error for the estimation of the mean value of a distribution based on a sample of size n is given by $\epsilon = z^* \frac{\sigma}{\sqrt{n}}$, where σ is the standard deviation of the distribution and z^*, known as the *confidence coefficient*, is a constant depending on the desired confidence level for the estimate [63]. For a confidence level of 95%, a commonly accepted standard, the value of the confidence coefficient is $z^* = 1.96$. It follows that, with

95% confidence, the margin of error of a sample of size n can be estimated by

$$\epsilon_n = 1.96 \frac{S_n}{\sqrt{n}}, \tag{6.3}$$

where the sample standard deviation $S_n = (n-1)^{-1} \sum_{i=1}^{n} (u_i - \overline{u})^2$ was used as an unbiased estimator for the unknown population standard deviation σ. The sample standard deviation can be computed dynamically using Welford's algorithm [53, 71] as new samples are added to the data set. The process is as follows: the sample mean m and the auxiliary variable s are initialized as $m_w^{(0)} = 0, s_w^{(0)} = 0$. Then, as the i-th sample is drawn, these values are updated according to the formulas

$$m_w^{(i)} = m_w^{(i-1)} + \frac{u^{(i)} - m_w^{(i-1)}}{i}, \quad s_w^{(i)} = s_w^{(i-1)} + \left\langle u^{(i)} - m_w^{(i-1)}, u^{(i)} - m_w^{(i)} \right\rangle. \tag{6.4}$$

Finally, for a sample of size i, the sample standard deviation is given by $S_i = \sqrt{s_w^i/(i-1)}$. Combining this strategy with the approximation (6.3) a stopping criterion for sampling is obtained. As new samples are drawn, the standard deviation is updated, ϵ_k is estimated, and the process is halted when ϵ_k falls below a prescribed threshold. In all our experiments, we will set $\epsilon = 0.01$, thus requiring an estimate with a margin of error that, 95% of the time, will be below 1%.

		Surrogate Level 2	Surrogate Level 3	Surrogate Level 4	Direct Solver
1% Noise	# Samples	5100	5000	5000	5200
	Time per Sample (s.)	0.16	0.59	3.03	5.22
	Total Time (s.)	831	2944	15141	27166
2% Noise	# Samples	20000	25300	31100	19100
	Time per Sample (s.)	0.16	0.59	3.04	5.69
	Total Time (s.)	3264	14911	94602	108620

Table 6.6: Average CPU times, in seconds, for evaluation of the surrogate function and direct solution of the free boundary problem.

We applied this strategy using the maximum entry-wise variance updated with (6.4) of both the solution ψ_h obtained from the direct solver and the surrogate approximation $\widehat{\psi}_h$. The results are shown in Table 6.6, which gives CPU times for current perturbations with 1% and 2% noise. We give several variants of these results. In particular, the number of adaptively chosen samples varied somewhat with the choice of function used, ψ_h or $\widehat{\psi}_h$.

For 1% noise, the direct solver required more samples than the level 2 and 3 surrogates (see Table 6.8); for 2% noise, it required fewer (see Table 6.7). It can be seen from these results that in all cases for level 2 and level 3 sparse grids, the cost of using the surrogate is significantly smaller than that of using the direct solver. In simulations with 1% noise, the direct solver was approximately 33 times more expensive than the level 2 surrogate, and approximately 9 times more than the level 3 surrogate. With 2% noise, the analogous factors were approximately 33 and 7. Here, we are considering the total time of the simulation with dynamic sampling which tends to benefit the direct solver—as it usually requires less samples to achieve a target RMS. As noted above, from the point of view of accuracy, there is essentially no difference between the surrogate and the direct solutions for 1% noise; for 2% noise, the significant cost reduction comes with some compromise in accuracy but little difference in conclusions concerning features of the plasma.

As mentioned above, the number of samples required to fall within a predetermined Monte Carlo tolerance varies depending on whether the direct solver or the surrogate is used to obtain an approximation of the stream function ψ. This points to a difference in the spread of the distributions generated by each of the methods. The means of the distributions, however, coincide with the prescribed tolerance by construction, regardless of whether the samples used are identical or not. For the experiment showcased in Table 6.7, two different sets of coil currents were generated considering a variability of 2%. A surrogate of level

	Evaluation of surrogate Sample A (25300 realizations)		Direct solver Sample B (19100 realizations)	
Quantity of interest	Mean	Variance	Mean	Variance
x point	(5.14,-3.29)	(2.89e-04,1.48e-03)	(5.14,-3.29)	(2.56e-04,1.31e-03)
magnetic axis	(6.41,0.61)	(1.53e-03,1.14e-03)	(6.41,0.61)	(1.19e-03,6.42e-04)
strike points	(4.16,-3.71) (5.56,-4.22)	(2.42e-03,2.38e-03) (5.18e-08,3.20e-03)	(4.16,-3.71) (5.56,-4.21)	(8.59e-04,2.09e-03) (5.06e-08,3.13e-03)
inverse aspect ratio	0.32	5.53e-06	0.32	4.85e-06
elongation	1.86	3.00e-04	1.86	3.12e-04
upper triangularity	0.43	2.87e-04	0.43	2.65e-04
lower triangularity	0.53	2.37e-04	0.53	2.15e-04

Table 6.7: Comparison of sample means and variances for relevant physical quantities extracted from two samples by the surrogate function and the direct solver. The number of realizations for each of the samples was determined dynamically to achieve a Monte Carlo estimation accurate to at least 1% with 95% confidence. The noise level in the current values was of 2% and the surrogate was built with a sparse grid of level 3.

3 required 25300 randomly chosen current values for its Monte Carlo error to fall below 1%, as given by (6.3). These current values constitute sample A. When a direct solver was used instead, the dynamically estimated error needed only 19100 realizations to fall below the 1% threshold. These currents constitute Sample B. As can be seen in Table 6.7, the variances associated with the quantities extracted from the surrogate are uniformly larger than those stemming from the direct solver. The mean values are nonetheless equal to within the prescribed tolerance. Despite the difference in required sample size, the level 3 surrogate was still faster than the direct solver by a factor of more than 7, as reported in Table 6.6.

In a similar vein, Table 6.8 displays analogous results for an experiment where the coil currents were perturbed by 1%. Samples for $\widehat{\psi}$ and ψ_h were taken using either surrogates of levels 2 and 3 or the direct solver. The number of realizations was determined dynamically by setting the Monte Carlo tolerance to 1% and using the estimator (6.3). The mean values

	Evaluation of surrogate (Level 2) 5100 realizations		Evaluation of surrogate (Level 3) 5000 realizations		Direct solver 5200 realizations	
Quantity of interest	Mean	Variance	Mean	Variance	Mean	Variance
x point	(5.14,-3.29)	(1.07e-04,3.58e-04)	(5.14,-3.29)	(1.06e-04,3.55e-04)	(5.14,-3.29)	(1.09e-04,3.57e-04)
magnetic axis	(6.41,0.61)	(2.92e-04,2.84e-04)	(6.41,0.61)	(2.84e-04,4.85e-04)	(6.41,0.61)	(2.98e-04,2.79e-04)
strike points	(4.16,-3.71) (5.56,-4.22)	(1.98e-04,5.43e-04) (1.28e-08,7.93e-04)	(4.16,-3.71) (5.56,-4.22)	(1.97e-04,5.40e-04) (1.32e-08,8.14e-04)	(4.16,-3.71) (5.56,-4.22)	(2.02e-04,5.54e-04) (1.27e-08,7.89e-04)
inverse aspect ratio	0.32	1.19e-06	0.32	1.17e-06	0.32	1.19e-06
elongation	1.86	3.39e-05	1.86	3.44e-05	1.86	3.47e-05
upper triangularity	0.43	6.41e-05	0.43	6.60e-05	0.43	6.58e-05
lower triangularity	0.53	3.71e-05	0.53	3.73e-05	0.53	3.73e-05

Table 6.8: Sample means and variances for relevant physical parameters extracted using levels 2 and 3 of the surrogate and the direct solver. The quantities of interest are the coordinates of the x-point, magnetic axis and strike points, as well as the values for plasma shaping parameters. The experiment was carried over with 1% noise in the coil currents. The sample size was increased dynamically using batches of 100 samples until the estimated Monte Carlo estimation error fell below 0.01.

	Level 2	Level 3	Level 4
1% Total Time (s.)	2289	19375	126276
2% Total Time (s.)	2378	20129	131194

Table 6.9: Processing time, in seconds, required for the construction of surrogate functions of different levels.

of all quantities coincide to within the prescribed tolerance, but the time required to generate the data using the surrogates is substantially shorter (see Table 6.6).

Note that a pre-processing step is required to build the surrogates; the required times are shown in Table 6.9. This "offline" step, while not negligible, is done only once and need not be repeated if multiple simulations are done or if the number of samples needs to be increased. Even if these one-time costs are factored in, the speedup for the level 3 surrogate is still a factor of 3 for 2% perturbations.

6.2 Multilevel Monte Carlo sampling with direct computation

In this experiment, we explore the savings that can be obtained using multilevel Monte Carlo methods to estimate the stochastic integral (2.14), which reduces costs by performing the bulk of the computations on a sequence of spatial grids that are coarser than the one that would typically be used for a simple Monte Carlo simulation. We examine this approach using both a set of uniformly refined grids and a set of adaptively refined grids guided by a discrete error estimator.

6.2.1 Experiment description

For the solution of the perturbed problems, the initial iterate of Newton's method was taken to be the solution corresponding to the reference currents $\boldsymbol{I}$ in (6.1). All tests used the splitting parameter $\theta = 0.5$ in (5.3). The stopping threshold for the relative residual was set to 5×10^{-11}. The user-specified tolerances for the normalized mean squared error range from $\epsilon = 2 \times 10^{-4}$ to 8×10^{-3}. To produce an estimate of the number of samples required on each discretization level, equation (5.14) requires the knowledge of two problem-dependent parameters: the power γ appearing in the estimate for the computational cost (5.13), and the normalized variance of the correction terms V_ℓ as defined in (5.12). The normalization factor $\|\mathbb{E}(u_L)\|_Z$ in (5.12) was estimated by averaging 100 sample realizations on the finest uniform mesh level ($\ell = 5$) to be approximately 8.57×10^{-1}. To estimate the value of γ, 100 random currents are sampled for different mesh sizes M_ℓ, the processing times required to obtain the solutions are averaged for each mesh size, and γ is estimated through a regression. Figure 6.8 shows the behavior of the average cost as a function of the mesh size M_ℓ; from the data displayed, the power law is estimated to be $\gamma \approx 1.09$. Note that this cost estimate is based on Matlab timings and not on the complexity analysis of standard linear solution algorithms. The same samples are also used to estimate the sample means, $\mathbb{E}(Y_\ell)$ or $\mathbb{E}(u_h)$, and variances, $\mathbb{V}(Y_\ell)$ or $\mathbb{V}(u_h)$ dynamically using Welford's algorithm in (6.4).

To perform MLMC-FE simulations, the user typically defines and generates a sequence of spatial grids, where, given a tolerance ϵ, the fineness of the grid is determined by the requirement that the discretization error ($\overline{\mathcal{E}}^2_{\text{Bias}}$ in (5.3)) or an estimate of it be small enough. In this study, we generated two types of grids, a set of *geometry-conforming* uniformly refined grids, and a set of adaptively refined grids constructed using the strategy presented

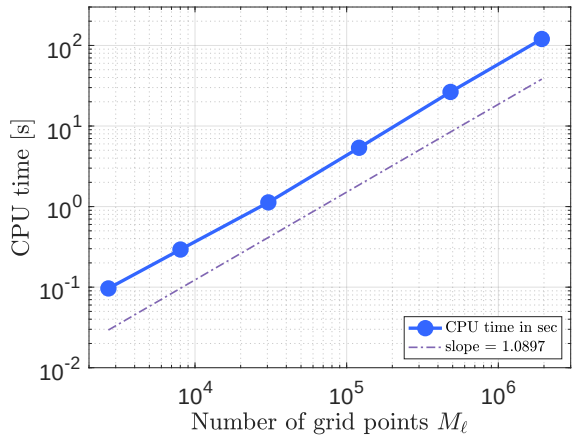

Figure 6.8: Mean CPU time to compute 100 realizations of u_ℓ, as a function of the number of grid points M_ℓ, plotted on a logarithmic scale. The fitted curve indicates that the computational cost C_ℓ behaves approximately like $M_\ell^{1.09}$.

Level ℓ	0	1	2	3	4	5
Uniform	2685	8019	30449	120697	484080	1934365
Adaptive	2685	6090	25099	103968	411913	1552282

Table 6.10: The number of grid points M_ℓ for both geometry-conforming uniform and adaptive ($q = 1/4$) meshes as the mesh levels increase from 0 to 5.

in Chapter 5.2.2. For the uniformly refined grid, we generated a total of six levels of grids. The domain components contain curved boundaries, which we handled by treating them as polygonal structures. The mesh generation entails identifying the curved boundaries using piecewise splines and interpolating along these splines with grids of varying fineness. We note that, due to the increasing accuracy of the spline approximation of the curved boundaries, these meshes are not a direct refinement of each other. Instead, each level is characterized by a mesh parameter h_ℓ being roughly half of the preceding mesh as determined by the `Triangle` mesh generator [66]. The adaptive refinement strategy began with the coarsest mesh from the uniform family and applied the weighted L_2 a posteriori error estimator specified in (5.19) and $q = 1/4$ to reflect an error decay similar to that of uniform refinement. `Triangle` was used to generate the desired adaptively refined meshes. The number of grid points for each of these methods on different grid levels is shown in Table 6.10. Note that the smallest element sizes for the adaptive meshes are not dramatically smaller than those of uniform

meshes, which suggests that the solution, as a function of the spatial variables, is fairly smooth.

6.2.2 Efficiency study

Figure 6.9 shows a variety of computational results, including the error estimator, V_ℓ, and CPU time for the two versions of MLMC (and times for full MC). To investigate the convergence behavior of the discretization error, we calculate the a posteriori error estimator for both uniform and adaptive meshes in the same experiment to obtain an estimate of γ for C_ℓ before conducting the simulations. The results are displayed in the top left plot in Figure 6.9, with a dashed line showing a least square fit, indicating that the discretization error of both methods exhibits an asymptotic rate of $\mathcal{O}(M^{-1})$ (or $\alpha_1 \approx 1$). The similar convergence rate further indicates that the solution to the problem is smooth and the error is equidistributed, rendering the adaptive strategy comparable to uniform mesh refinement. Note that the estimated error is used for variance extrapolation in (5.22) during the MLMC simulations.

The top right plot of Figure 6.9 shows the behavior of V_ℓ of (5.12) for both uniform and adaptive MLMC-FE methods with tolerance $\epsilon = 2\times10^{-4}$, using six levels of mesh refinement. It can be seen that both methods demonstrate a decreasing trend in the values of V_ℓ as the mesh resolution increases, with a power law decay characterized by $\beta_1 \approx 2$ in the least square fit. But there is a regime for a small number of grid points where the asymptotic behavior of the adaptive method is not evident, in contrast to the behavior of the uniform method. As the meshes get finer, the plots of V_ℓ for the two methods are close to being parallel.

The computational effort for uniform MLMC-FE and MC-FE scales as $\mathcal{O}(\epsilon^{-2})$ and $\mathcal{O}(\epsilon^{-3})$ respectively, as indicated by the slopes of the least square fitting lines for the red and yellow curves. This observation is consistent with the theoretical cost predictions in Theorem 2 (with $\beta_1 > \gamma$) and (5.5). Theorem 2 also indicates that the majority of computational work is performed on coarse grids. Table 6.11 shows the sample sizes obtained from (5.4) for MC-FE and (5.14) for MLMC-FE, further demonstrating a decrease in $N_\ell C_\ell$ as ℓ increases for the multilevel methods. We also found that the computational cost associated with the smallest tolerance $\epsilon = 2 \times 10^{-4}$ is so large that we were unable to generate MC-FE results on a fine mesh ($\ell = 5$) with a large sample size. However, we could estimate the costs. In particular, we found that the variance V_h is close to constant across mesh levels. Consequently, we used

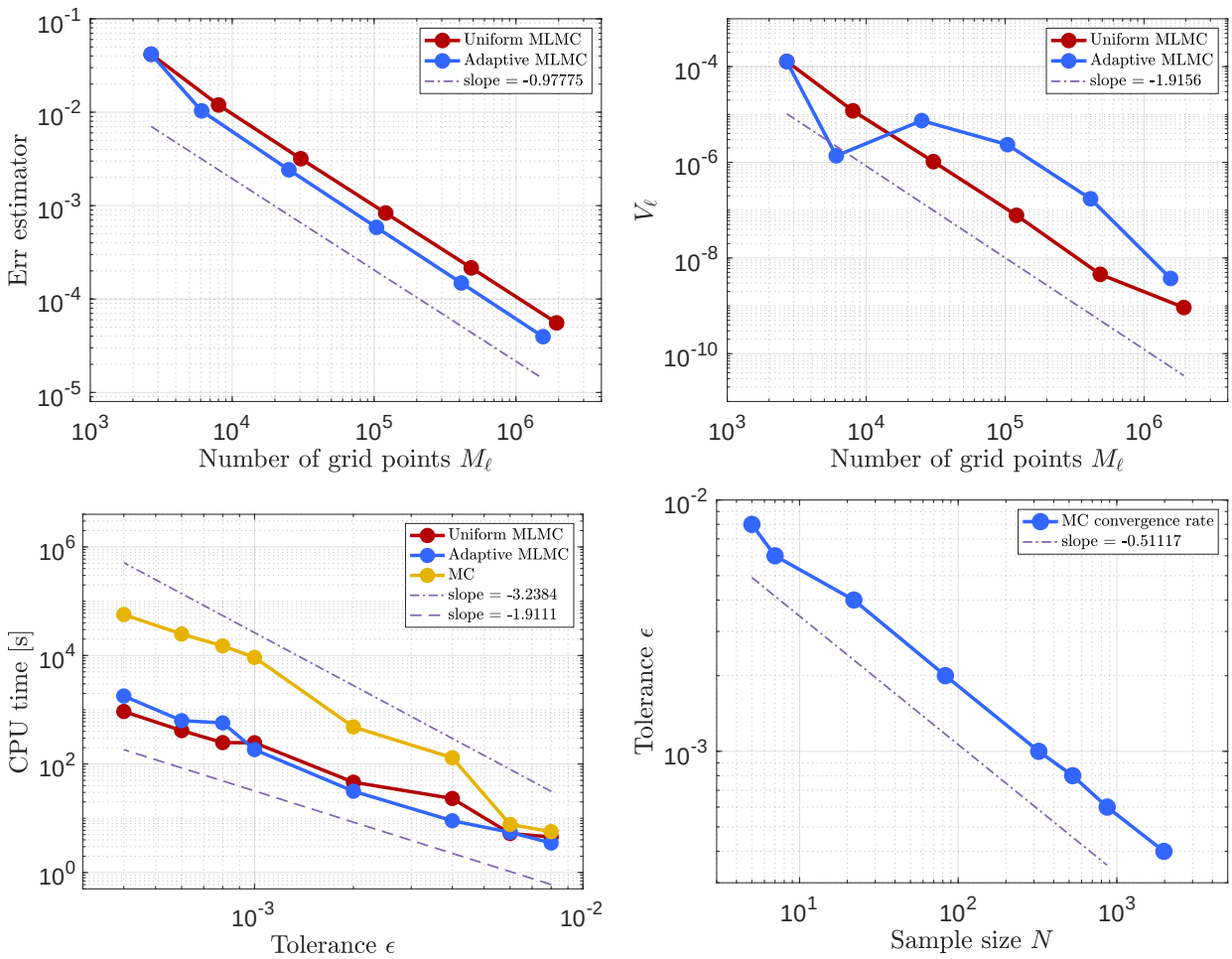

Figure 6.9: Top left: weighted L_2 error (with weight μx) of estimator $C\eta_\ell$ vs. number of grid points M_ℓ plot. Top right: normalized variance V_ℓ vs. number of grid points M_ℓ plot. Bottom left: CPU time in seconds vs. tolerance ϵ. Bottom right: Monte Carlo convergence rate estimate with tolerance ϵ vs. sample size N. This plot is generated from Table 6.11.

(5.4) to estimate the number of required samples as 8000 in Table 6.11, approximately four times the number required for $\epsilon = 4 \times 10^{-4}$. This number was multiplied by the mean CPU time observed for the computations for Figure 6.8 (120.3 seconds, the largest entry appearing in the figure) to give the estimated total CPU time in Table 6.12, which gives quantitative data on the costs in CPU time for the three methods, as well as the speedups achieved by the two multilevel methods. It can be seen that for small tolerance ϵ, both these methods reduce the CPU times dramatically, with many examples of speedups greater than a factor

of 60 and a best-case speedup approximately 200. In contrast to MC-FE, both versions of MLMC-FE successfully generated results with this tolerance.

As seen in the bottom left plot of Figure 6.9, the uniform and adaptive versions of MLMC-FE have similar computational costs of $\mathcal{O}(\epsilon^{-2})$, as evidenced by the similar decay rate of the error estimator and (5.21). According to (5.15), the slightly smaller magnitude of the error estimator for the adaptive MLMC-FE suggests a smaller (or comparable) computational cost in the asymptotic regime. However, when $\epsilon = 4 \times 10^{-4}$, the adaptive MLMC-FE method requires approximately twice as much CPU time (1.79×10^3 seconds) compared to the uniform MLMC-FE approach (9.29×10^2 seconds) due to a notable increase in V_ℓ around a grid of size $M_\ell = 10^4$. This also causes the speedups achieved using adaptive refinement to be somewhat smaller than for uniform refinement. Thus, the traditional advantage of adaptive mesh refinement is not clearly present. We also attribute this to the apparent smoothness of the solution. We will demonstrate some advantages of the adaptive strategy in Chapter 6.2.3.

The bottom right plot of Figure 6.9 shows that the nMSE tolerance ϵ of MC-FE approach declines as $\mathcal{O}(N^{-0.51})$, which is consistent with the well-known square root convergence rate. This rate holds since $\mathbb{V}(u_h)$ remains nearly constant among all levels.

	Level ℓ					
ϵ	0	1	2	3	4	5
8×10^{-3}			5			
6×10^{-3}			7			
4×10^{-3}				22		
2×10^{-3}				83		
10^{-3}					322	
8×10^{-4}					527	
6×10^{-4}					869	
4×10^{-4}					1980	
2×10^{-4}						8000*

	Level ℓ					
ϵ	0	1	2	3	4	5
8×10^{-3}	10	2	2			
6×10^{-3}	12	3	2			
4×10^{-3}	32	5	2	2		
2×10^{-3}	152	26	4	2		
10^{-3}	691	109	18	4	2	
8×10^{-4}	841	129	23	3	2	
6×10^{-4}	1610	231	40	8	2	
4×10^{-4}	3791	589	104	15	3	
2×10^{-4}	15859	2344	375	62	13	2

	Level ℓ					
ϵ	0	1	2	3	4	5
8×10^{-3}	10	2	2			
6×10^{-3}	19	3	3			
4×10^{-3}	38	5	4			
2×10^{-3}	121	18	6	2		
10^{-3}	697	82	49	8		
8×10^{-4}	1446	118	91	27	6	
6×10^{-4}	2070	218	133	21	3	
4×10^{-4}	5075	484	315	61	14	
2×10^{-4}	25871	1961	1668	430	85	14

Table 6.11: The optimal sample size estimation for MC-FE (left), uniform MLMC-FE (middle), and adaptive MLMC-FE (right). The simulations were conducted for a variety of choices of ϵ. The computational cost associated with a tolerance of $\epsilon = 2 \times 10^{-4}$ for Monte Carlo was prohibitive; the entry in the table for this tolerance (with an asterisk) is an estimate.

ϵ	MC-FE Time	Uniform MLMC-FE Time	Uniform MLMC-FE Speedup	Adaptive MLMC-FE Time	Adaptive MLMC-FE Speedup
8×10^{-3}	5.67e+00	4.52e+00	1.3	3.50e+00	1.6
6×10^{-3}	7.69e+00	5.25e+00	1.5	5.51e+00	1.4
4×10^{-3}	1.30e+02	2.32e+01	5.6	9.01e+00	14.4
2×10^{-3}	4.83e+02	4.62e+01	10.5	3.16e+01	15.3
10^{-3}	9.22e+03	2.47e+02	37.3	1.84e+02	50.1
8×10^{-4}	1.50e+04	2.48e+02	60.5	5.73e+02	26.2
6×10^{-4}	2.48e+04	4.13e+02	60.0	6.30e+02	39.4
4×10^{-4}	5.68e+04	9.29e+02	61.1	1.79e+03	31.7
2×10^{-4}	9.62e+05*	4.21e+03	228.5	1.21e+04	79.6

Table 6.12: The CPU time in seconds for MC-FE (left), uniform MLMC-FE (middle), and adaptive MLMC-FE (right), together with speedups for the multilevel methods, for a variety of choices of ϵ. The computational cost associated with a tolerance of $\epsilon = 2 \times 10^{-4}$ for Monte Carlo was prohibitive; the entry in the table for this tolerance (with an asterisk) is an estimate.

6.2.3 Accuracy study

Next, we will explore the plasma boundaries and geometric descriptors of the expected poloidal flux ψ resulting from the three methods. To ensure a fair comparison, we will use the results obtained from the MC-FE on the finest uniform mesh as a reference benchmark.

Plasma boundary. To ascertain the expected location of the plasma boundary, we first determine the expected solution to the free boundary problem, $\mathbb{E}[\psi]$, and determine the boundary of this deterministic function. This boundary is depicted, in dark violet, in Figure 6.10 along with the plasma boundaries obtained from 50 random currents, which are shown in light violet curves. In Figure 6.11 we present plots depicting the x-points and plasma boundaries of the expected solution ψ computed using only samples and corrections from increasingly finer grids for both uniform and adaptive MLMC-FE approaches. The data was obtained with tolerance $\epsilon = 4 \times 10^{-4}$. As can be seen when moving from left to right in Figure 6.11, the result obtained using the information from the coarsest level (leftmost column) is progressively corrected with information from increasingly finer grids, leading to the desired result depicted in the rightmost column.

Among the three methods, MC-FE yields the smoothest plasma boundaries in the vicinity of the x-point, followed by adaptive MLMC-FE, while the MLMC-FE approach on geometry-

conforming uniform meshes manifests the most pronounced irregularity in the boundary of the expectation of ψ. The boundary of the expected solution generated with the uniform grid MLMC-FE method exhibits irregularities as can be seen in Figure 6.10. These large deformations can be primarily attributed to the challenges arising from the curved boundaries. We will address this point in more detail at the end of this experiment. The top row of Figure 6.11 demonstrates that using a geometry-conforming mesh provides a more accurate approximation of the curved structure (in black) of the configuration than that in

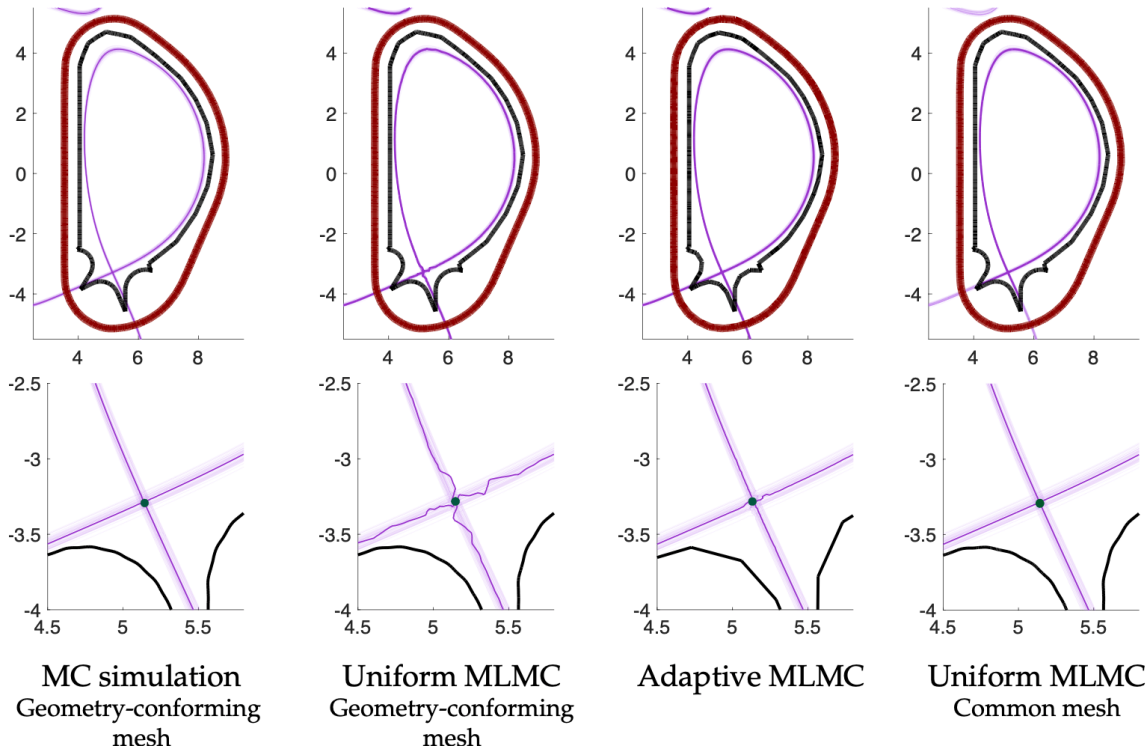

Figure 6.10: The overlayed plasma boundaries of 50 random realizations are displayed in the top row as violet curves. The solid violet line is the plasma boundary of the expected poloidal flux generated with tolerance $\epsilon = 4 \times 10^{-4}$. The inner and outer walls of the reactor are displayed in solid black and dark red respectively. The bottom row shows the regions close to the x-points in more detail. The dark green dots are the x-points of the expected solution. Each column from left to right corresponds to: simulation with the Monte Carlo approach, MLMC simulation on geometry-conforming uniform meshes, and adaptive MLMC simulation. All simulations were performed using the discretization level $\ell = 5$.

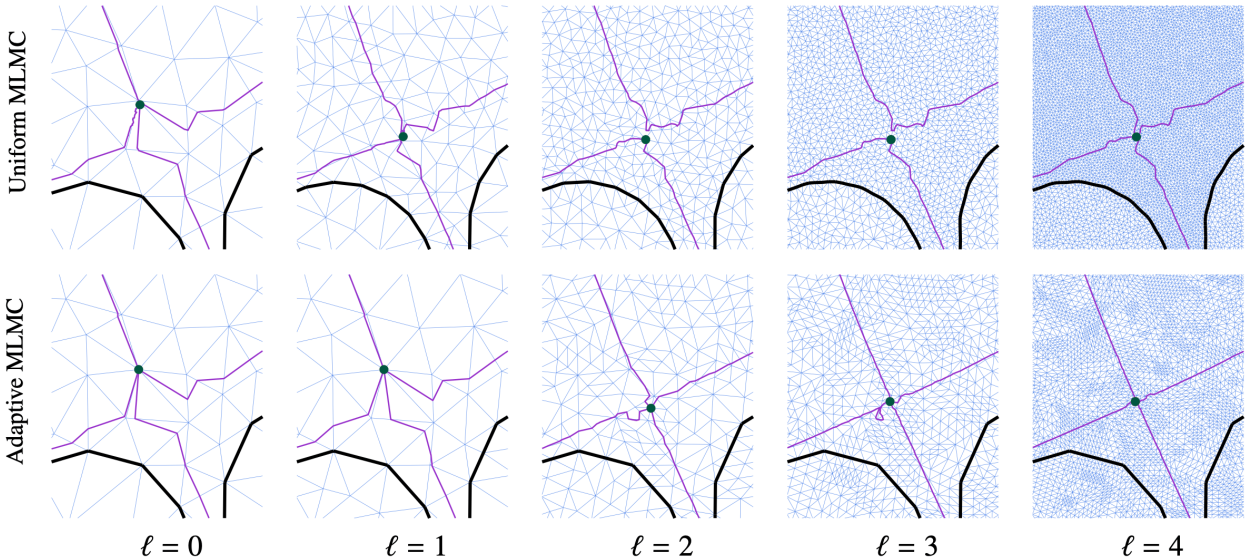

Figure 6.11: The violet curves represent the expected plasma boundaries for simulations using increasingly finer grids when $\epsilon = 4 \times 10^{-4}$ using the sample sizes specified in Table 6.11. Each sub-plot focuses on a region near the x-point, maintaining the same zoom-in ratio as the second row of Figure 6.10. The dark green dots denote the locations of the x-point. The top row shows the results of MLMC-FE on a set of geometry-conforming uniform meshes, while the bottom row displays the results for adaptive MLMC.

the bottom row. These observations underscore the challenge of striking a balance between preserving geometric fidelity when dealing with curved boundaries and the desired statistical accuracy of the solution.

Geometric descriptors. Table 6.13 reports some geometric parameters of the expected poloidal flux $\mathbb{E}[\psi]$ in (2.14). It is observed that these parameters are consistent across different simulation techniques, with agreement typically up to two or, in some cases, one significant digit. Having been derived from $\mathbb{E}[\psi]$ all these values are deterministic. There is, however, uncertainty associated with the corresponding quantities derived from ψ, as they are themselves random variables. Uncertainty on these quantities can be assessed by computing Monte Carlo estimates of their expectations and variances. The multi-level methodology can be used for this purpose without any modification by simply regarding them as quantities of interest in their own right. All the relevant descriptors are computed for each of the samples gathered during the computational loop at the various discretization levels

and their expectations and variances are estimated following the same process described on Algorithm 2. The expectations and variances obtained in this fashion are shown in Table 6.14.

	MC-FE	Uniform MLMC-FE	Adaptive MLMC-FE
x point	(5.14,-3.29)	(5.14,-3.29)	(5.14,-3.28)
magnetic axis	(6.41,0.61)	(6.44,0.56)	(6.46,0.54)
strike points	(4.16,-3.71) (5.56,-4.22)	(4.16,-3.71) (5.56,-4.22)	(4.16,-3.71) (5.56,-4.21)
inverse aspect ratio	0.32	0.32	0.32
elongation	1.86	1.87	1.86
upper triangularity	0.43	0.43	0.43
lower triangularity	0.53	0.53	0.53

Table 6.13: Geometric parameters of the expected poloidal flux ψ from MC-FE, MLMC-FE with geometry-conforming uniform mesh set, and adaptive MLMC-FE. The results are generated with an nMSE 4×10^{-4}.

	MC		MLMC			
	-		Uniform		Adaptive	
Geometric parameters	Mean	Variance	Mean	Variance	Mean	Variance
x point	(5.14,-3.29)	(2.37e-04,1.44e-03)	(5.14,-3.28)	(2.44e-04,6.33e-04)	(5.14,-3.29)	(4.03e-04,2.91e-03)
magnetic axis	(6.41,0.61)	(1.01e-03,6.11e-04)	(6.41,0.60)	(2.65e-02,6.28e-03)	(6.42,0.60)	(3.15e-02,5.81e-03)
strike points	(4.17,-3.71) (5.56,-4.22)	(4.46e-03,2.23e-03) (2.88e-08,3.28e-03)	(4.16,-3.71) (5.56,-4.22)	(1.23e-04,4.30e-05) (4.05e-09,5.54e-05)	(4.17,-3.71) (5.56,-4.21)	(4.18e-04,2.00e-04) (2.02e-08,2.98e-04)
inverse aspect ratio	0.32	4.76e-06	0.32	4.23e-06	0.32	4.31e-06
elongation	1.86	1.50e-04	1.86	1.83e-03	1.86	2.29e-03
upper triangularity	0.43	2.49e-04	0.43	1.45e-03	0.43	2.08e-03
lower triangularity	0.53	1.08e-04	0.53	1.52e-03	0.53	1.31e-03

Table 6.14: Sample mean of the geometric parameters extracted from each realization during three simulations MC-FE, uniform MLMC-FE, and adaptive MLMC-FE. The results are generated with an nMSE 4×10^{-4}.

Despite the advantages of low computational cost, the MLMC-FE-based methods may encounter difficulties in accurately determining the locations of x-points and magnetic axis. Note that the discrete x-points, correspond to saddle points of the piecewise linear approximation of ψ, and thus can only be located at the nodes of the mesh. The numerical identification of their exact locations, which often relies on changes in the sign of the discrete gradient, can be challenging; see [18, 39] for discussions of the computational difficulties.

In summary, simulations using the uniform MLMC-FE on non-nested geometry-conforming uniform meshes may encounter a substantial challenge in accurately identifying the x-point

and achieving less accurate quantities, especially for the plasma boundary, when compared to the results obtained from MC-FE. On the other hand, the adaptive MLMC-FE approach on a nested adaptively refined mesh set produces results that closely align with the MC-FE at a much lower computational cost.

Meshing curved domains and their effect on MLMC estimations. In the deterministic setting the *geometric error* stemming from inaccurately representing the curved boundaries of the spatial domain introduces the undesired consequence of hindering the decay of the discretization error since, as the mesh is refined, the discretized computational domain does not converge to the semicircle bounded by Γ. Moreover, this error adversely affects the accuracy of estimating the variance of correction terms V_ℓ, resulting in inaccurate sample size estimation and impacting the convergence of statistical error. In the stochastic setting the geometric error manifests itself in rendering the Monte Carlo estimator biased and inconsistent. The inconsistency stems from the fact that, as both the sample size and the mesh level increase, the Monte Carlo estimator does not converge to the expectation of the random variable u satisfying the free boundary problem. Instead, the estimator converges to the expectation of the random variable that satisfies a perturbation of (2.11) where the curve Γ is not a semicircle, but the initial polygonal approximation. If the initial mesh is fine enough, this geometric bias will likely be too small to affect the estimation.

On the other hand, if an exact descriptor of the curved boundaries is available, the aforementioned difficulty can be overcome by re-sampling the curved boundaries when building the sequence of finer grids, thus allowing for a resolution of the curved structures consistent with the respective mesh parameter. If an exact descriptor is not available it is possible to approximate it with, for instance, a cubic spline representation that interpolates the original polygonal representation. This spline surrogate is then used to re-sample the boundary as the mesh is refined. This strategy, which gives rise to what we refer to as *geometry conforming meshes*, was implemented for the numerical experiments with *uniformly refined meshes* and can be seen in use in the top row of Figure 6.11, where the curved boundary is represented more accurately as the mesh is refined.

Nevertheless, even if the approximation to the problem geometry is now consistent with the discretization error, this approach creates additional challenges. Since the approximations to the curved boundaries are not fixed across mesh levels, the sequence of meshes is no longer nested—not even in the case of uniform refinements. Moreover, due to the fact that

the sequence of discrete domains no longer coincides across levels, the domains of definition of the respective discrete solutions will not overlap, and an extrapolation step may be needed to compute the multilevel Monte Carlo estimator on a common computational domain. This strategy, used in our numerical experiments, introduces an additional extrapolation error. In our case this is evident, for instance, in the fact that the plasma boundary of the expected solution $\mathbb{E}[\psi_h]$ is considerably less regular in the geometry-conforming case than it is in the non-geometry-conforming one. This can be seen in Figure 6.11. The extrapolation error can be taken care of through careful post-processing. One option is to project or interpolate the numerical solutions into a subdomain common to all grids so that no extrapolation is needed for evaluation. This strategy was employed to produce Figure 6.12 successfully eliminating the spurious oscillations in the plasma boundary. However, doing this requires considerable computational work and reduces the time savings obtained from MLMC.

One further difficulty is that the re-sampling of the boundaries is impossible to perform in a straightforward fashion in the case of adaptively refined meshes. Thus, the geometric approximation remains fixed at the initial level of refinement. This can be seen in the bottom row of Figure 6.11, where the solid black line represents the polygonal approximation to the curved boundary of the divertor. The approximation improves as the mesh is refined for the uniformly refined mesh, but stays fixed for the adaptive strategy.

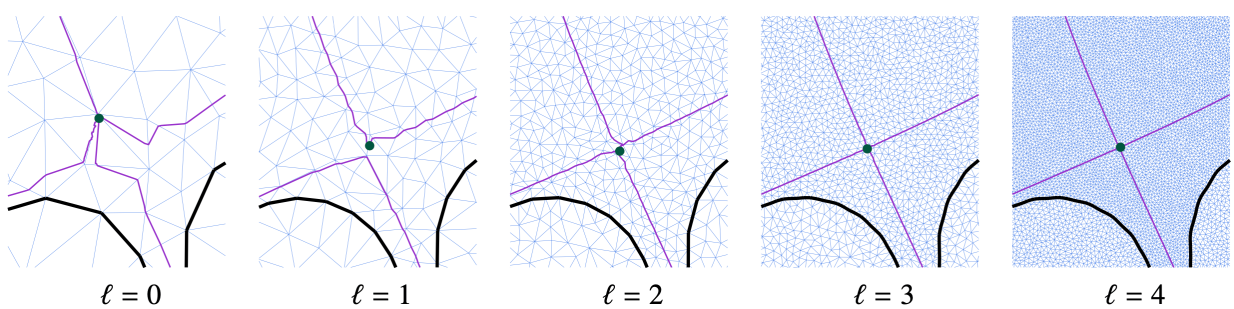

Figure 6.12: The violet curves represent the expected plasma boundaries for post-processed simulations using increasingly finer grids when $\epsilon = 4 \times 10^{-4}$ and sample size specified in Table 6.11. Each sub-plot focuses on a region near the x-point, maintaining the same zoom-in ratio as the second row of Figure 6.10. The dark green dots denote the locations of the x-point.

6.3 Multilevel Monte Carlo sampling with surrogate function

In our concluding experiment, we embark on an exploration of a surrogate-enhanced multilevel Monte Carlo sampling technique, within the framework of uniformly refined grids. Our investigation entails a comprehensive analysis of the efficiency, focusing on both offline and online costs, alongside an assessment of accuracy concerning plasma boundary and geometric descriptors within the context of online costs. Through this study, we aim to provide insights into the performance and efficacy of the surrogate-enhanced multilevel Monte Carlo approach, shedding light on its potential applications and benefits in practical scenarios.

6.3.1 Experiment description

The implementation of MLMC-FE sampling requires a hierarchical spatial mesh set. The number of grid points M_ℓ of the mesh set is shown in Table 6.15, with the reference mesh comprising 30,449 grid points. These meshes are subsequently used in MC-FE and MLMC-FE samplings with both direct solvers and surrogates.

Level	0	1	2	3	4	5
M_ℓ	2685	8019	30449	120697	484080	1934365
P_q	1	25	313	2649	17265	–

Table 6.15: The number of spatial grid points M_ℓ at increasing level $\ell = 0$ to 5, and the number of parameter grid points P_q from level $q = 0$ to 4.

In constructing surrogates, we employ stochastic collocation on Clenshaw-Curtis sparse grids within a 12-dimensional parameter space. The corresponding counts of sparse grid nodes P_q at increasing levels are provided in Table 6.15. For all samplings conducted in this study, the stopping threshold for the relative residual was set to 5×10^{-11}, and the user-specified tolerances for the normalized mean squared error fall within the range of $\epsilon = 2\times10^{-4}$ to 8×10^{-3}. The sample mean and V_ℓ are dynamically updated using Welford's algorithm in (6.4). The resulting normalization factor $\|\mathbb{E}(u_L)\|_Z$ on the finest spatial mesh level ($\ell = 5$) is approximately 0.86. Particularly for surrogate-based samplings, the splitting parameters in (5.7) and (5.28) are selected as $\theta = 0.5, \theta_1 = \theta_2 = \sqrt{0.5}/2$. In the case of direct solve, the stopping threshold for the relative residual of Newton's method is set to 5×10^{-11}. A half-and-half splitting is applied between discretization and sampling errors. All experiments

use MATLAB R2023a on a System 76 Thelio Major with 256GB RAM and a 64-Core @4.6 GHz AMD Threadripper 3 Processor.

6.3.2 Surrogate construction

The surrogate construction requires an understanding of discretization and interpolation errors. The plots of Figure 6.13 demonstrate the discretization error, and interpolation error resulting from perturbing currents in varying numbers of coils within our model problem. The decay rate α_1 of the discretization error is around 1, as shown in [19]. Notably, perturbing currents in three coils results in a discernible reduction in the interpolation error, manifesting at a rate of approximately $\mu_1 \approx 0.47$. In contrast, when all twelve coils undergo perturbations, the interpolation error decays at a slower rate of $\mu_1 \approx 0.04$. This near-zero decay rate implies a rapid escalation in the required sparse grid nodes in (4.5), (4.12), and (4.17) for the single-level sparse grid stochastic collocation and single-level spatial discretization, single-level sparse grid stochastic collocation and multilevel spatial discretization, and multilevel in both parameter and spatial grids surrogates as tolerance ϵ decreases. The rapid growth of sparse nodes and limited computational capabilities for finer spatial meshes introduce computational complexities in both surrogate construction and evaluation. Nevertheless, to enable sampling for our problem, we acknowledge and accommodate a nearly constant interpolation error. To this end, we use level $q = 1$ sparse grid points in construct-

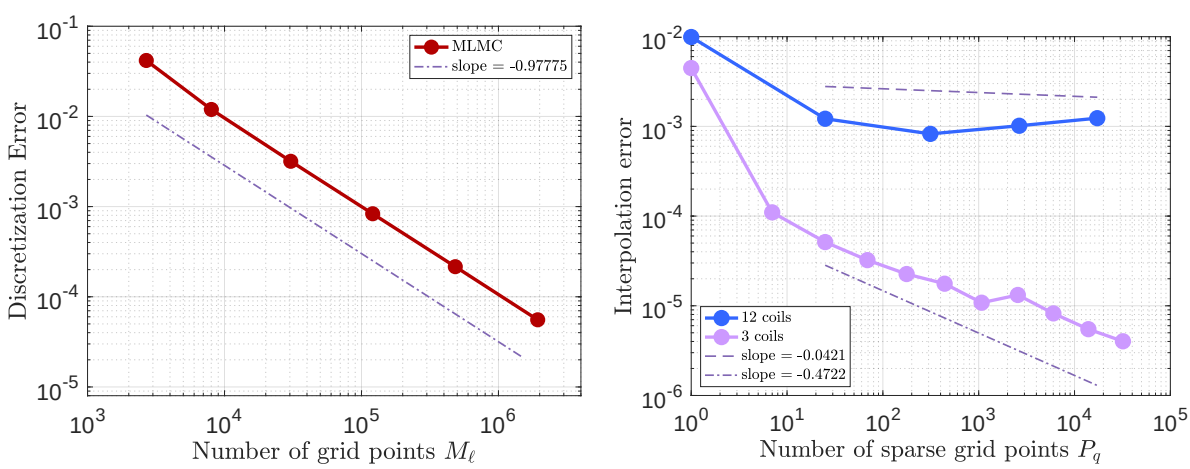

Figure 6.13: Top left: Discretization error vs. number of spatial grid nodes M_ℓ plot. Top right: Interpolation error vs. number of sparse grid points P_q plot.

ing the single-level sparse grid stochastic collocation and single-level spatial discretization surrogate for MC-FE samplings, and single-level sparse grid stochastic collocation and multilevel spatial discretization surrogate for MLMC-FE samplings. In such scenarios where the same sparse grid nodes (same P) are used, the number of sparse grid points P_{q_s} is independent of ϵ, the parameters related to P_{q_s}, including μ_1 and γ_2 in the conditions (ii) and (iv) of the MLMC-FE with the single-level sparse grid stochastic collocation and multilevel spatial discretization surrogate in Theorem 3 become zero, rendering the degeneration of the sampling cost into the basic MLMC cost, as established by Giles [12] in Theorem 2.

To evaluate the cost of constructing surrogates, we first estimate the average sample cost $C_\ell = \mathcal{O}(M_\ell^\gamma)$ through direct computation. This is done by conducting computations with 100 random currents and calculating the mean CPU time, rather than relying on the complexity analysis of standard solution algorithms. The left plot of Figure 6.14 illustrates the results, with regression analysis providing $\gamma = 1.09$, as reported in [19]. Using this estimated cost C_ℓ, we then assess the costs associated with building a series of single-level sparse grid stochastic collocation and single-level spatial discretization surrogates, where the spatial grid level ℓ increases from 0 to 5 and the sparse grid level q remains fixed at 1. The detailed results are presented in Table 6.17. To determine the required spatial grid level for estimating

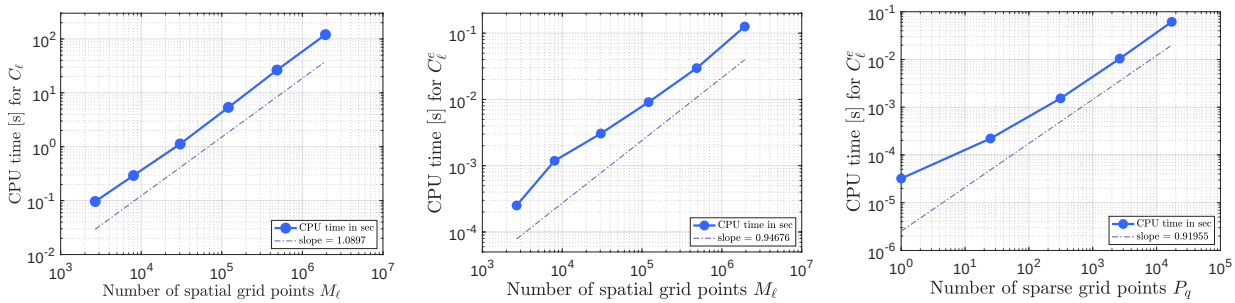

Figure 6.14: Left: The mean CPU time of 100 realizations for direct computations vs. an increasing number of spatial grid points M_ℓ. Middle: The mean CPU time of 100 realizations for surrogate evaluations with a fixed sparse grid level $q = 1$ vs. an increasing number of spatial grid points M_ℓ. Right: The mean CPU time of 100 realizations for surrogate evaluations with a fixed spatial grid level $\ell = 0$ vs. an increasing number of sparse grid points P_q. The fitted curves indicate that C_ℓ and C_ℓ^e behave approximately like $1.63 \times 10^{-5} \times M_\ell^{1.09}$ and $3.08 \times 10^{-8} \times M_\ell^{0.95} P_q^{0.92}$.

the cost of constructing a multi-spatial level surrogate (single-level sparse grid stochastic collocation and multilevel spatial discretization surrogate), we observe that for tolerances ϵ ranging from 2×10^{-4} to 8×10^{-3}, the corresponding required finest spatial levels satisfying the discretization error condition in (4.11) are in Table 6.16. We then use these required finest spatial grid levels along with the single spatial level surrogate cost from Table 6.17 to estimate the offline cost. The top right plot of Figure 6.15 shows the construction costs for both single-level sparse grid stochastic collocation and single-level spatial discretization, and single-level sparse grid stochastic collocation and multilevel spatial discretization surrogates as a function of ϵ. This plot suggests that both surrogates exhibit a similar complexity of approximately $\mathcal{O}(\epsilon^{-1})$, aligning closely with the theoretical cost $\mathcal{O}(\epsilon^{-\gamma/\alpha_1}) \approx \mathcal{O}(\epsilon^{-1.09})$.

Tolerance	2×10^{-4}	4×10^{-4}	6×10^{-4}	8×10^{-4}	10^{-3}	2×10^{-3}	4×10^{-3}	6×10^{-4}	8×10^{-3}
Finest spatial grid level L	6	6	6	5	5	5	4	4	4

Table 6.16: Finest spatial grid level required to perform surrogate-based samplings for various tolerances.

Spatial grid level ℓ	0	1	2	3	4	5
CPU time	2.02e+00	5.95e+00	2.30e+01	1.21e+02	6.60e+02	2.75e+03

Table 6.17: CPU time to construct the single-level sparse grid stochastic collocation and single-level spatial discretization surrogate for sparse grid level $q = 1$ against increasing spatial grid levels.

6.3.3 Efficiency study

We now use the constructed single-level sparse grid stochastic collocation and single-level spatial discretization surrogate for MC-FE samplings, and single-level sparse grid stochastic collocation and multilevel spatial discretization surrogate for MLMC-FE samplings.

To estimate the online sampling cost, we approximate the average cost per sample $C_\ell^e = \mathcal{O}(M_\ell^{\gamma_1} P_q^{\gamma_2})$ for surrogate evaluation by measuring the CPU time of 100 random samples. The results are shown in the middle and right plots of Figure 6.14. Regression analysis reveals rates of $\gamma_1 = 0.95$ and $\gamma_2 = 0.92$, consistent with the anticipated linear behavior in both M_ℓ and P_q [49, Section 3.3.5.4] for surrogate evaluation. The sampling costs associated with various sampling techniques are illustrated in the bottom left plot of Figure 6.15. To

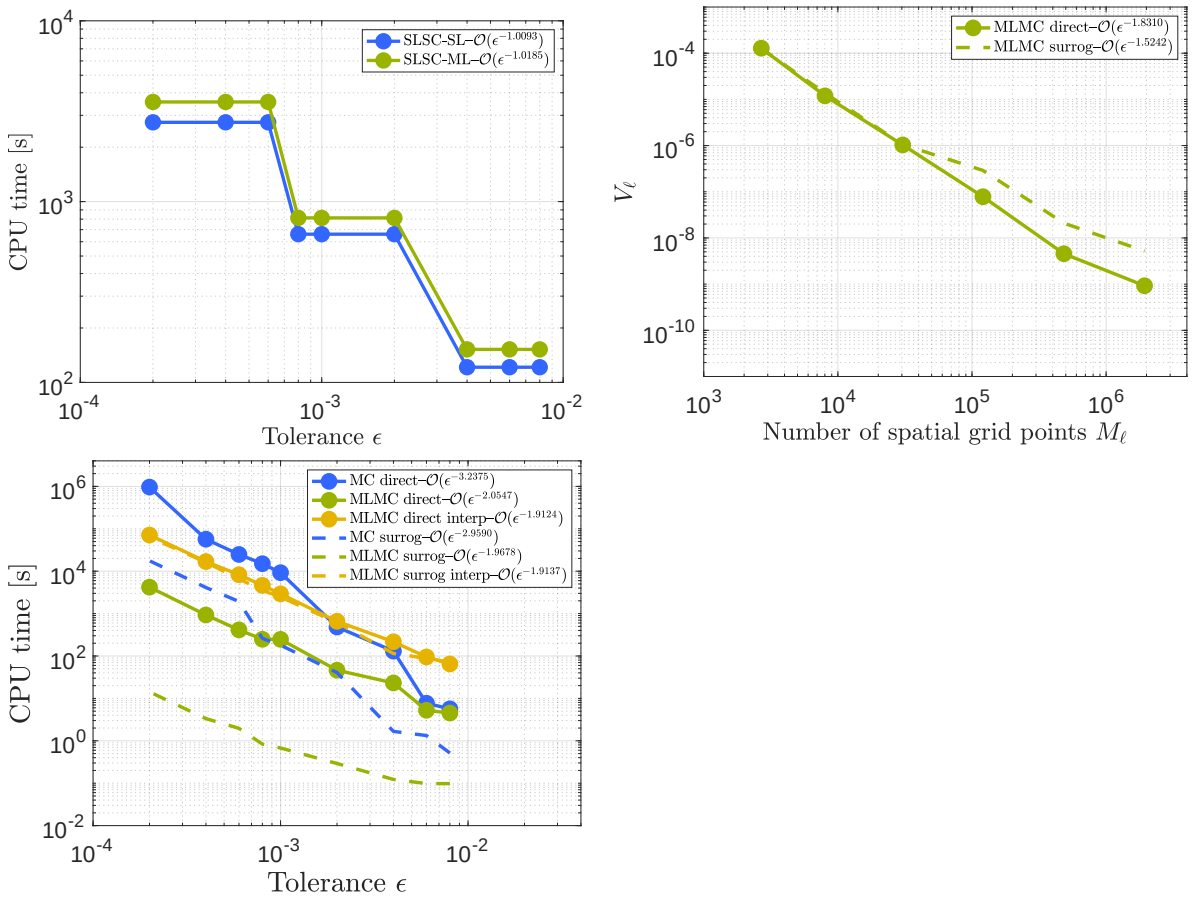

Figure 6.15: Top left: Offline cost for both single-level sparse grid stochastic collocation and single-level spatial discretization surrogate and single-level sparse grid stochastic collocation and multilevel spatial discretization surrogate. Top right: value of V_ℓ vs. number of spatial grid points M_ℓ plot. Bottom left: Estimated sampling CPU time v.s. tolerance ϵ plot, where the yellow curves represent the result of interpolating the MLMC-FE estimator to a common fine mesh of spatial level $\ell = 5$.

compare the results of both MLMC-FE and MC-FE samplings, the same spatial mesh with level defined in (4.11) is used to ensure consistency in discretization errors meeting the nMSE error splitting threshold.

For MC-FE samplings, least-square fittings of the online costs from the bottom left plot of Figure 6.15 indicate that the direct solver and surrogate exhibit computational costs of $\mathcal{O}(\epsilon^{-3.24})$ and $\mathcal{O}(\epsilon^{-2.96})$ respectively. These observed costs closely align with the theoretical values of $\mathcal{O}(\epsilon^{-2-\gamma/\alpha_1}) = \mathcal{O}(\epsilon^{-3.09})$ for the direct solver and $\mathcal{O}(\epsilon^{-2-\gamma_1/\alpha_1}) = \mathcal{O}(\epsilon^{-3})$ for

the surrogate, where μ_1=0 in (5.8) as P_q is independent of ϵ. Given the proximity of the magnitudes of $\gamma; (\approx 1.09)$ in the direct solver and $\gamma_1; (= 1)$ in the surrogate evaluation, the slopes of both solid and dotted blue lines in the cost plot (bottom left plot of Figure 6.15) run almost parallel to each other. Notably, in our model, the direct solver relies on Newton's method to handle nonlinear algebraic systems. This choice of methodology for solving these systems affects the parameter γ, influenced by factors such as the mean cost associated with solving a linearized system due to the nonlinearity of magnetic permeability and the source term, the number of iterations required by Newton's method, complexities arising from the nonlinearity of the free plasma boundary, and the need for reassembling the nonlinear system. These factors, together with the Matlab implementation of sparse matrices using sparse direct methods [14, 30], collectively contribute to shaping the value of γ. We anticipate γ to take values such as 1.5 when considering the nested dissection as ordering for sparse matrices [26] and between 1 to 2 when using a band solver [30, Section 4.3]. Our estimation of γ is based on Matlab timings and not on the complexity analysis of standard linear solution algorithms; hence it may be overly optimistic. In general, when the sample-wise cost of the direct solver has an order γ much larger than 1, the advantages of using the surrogate become more pronounced as tolerance ϵ decreases.

In the same sampling cost plot for MLMC-FE related approaches, the solid and dashed green curves indicate that both the direct solver and surrogate exhibit complexities of approximately $\mathcal{O}(\epsilon^{-2})$, consistent with the theoretical predictions in Theorem 2. Notably, the surrogate approach, represented by the dotted green line, incurs a significantly lower sampling cost in CPU time compared to the nearly parallel solid green line associated with the direct solver. This disparity can be attributed to a smaller constant factor in the asymptotic estimate for the sample-wise cost for surrogate evaluation, as estimated in the caption of Figure 6.14. The decay rates of V_ℓ are approximately 1.83 for direct computation and 1.47 for the surrogate, exceeding the increase in cost, which are $\gamma \approx 1.09$ and $\gamma_1 \approx 1$ respectively, as shown in the top right plot of Figure 6.15 and left plot of Figure 6.14. As per Theorem 2, the majority of the work is concentrated on the coarse spatial grids for both the surrogate-enhanced MLMC-FE and the direct solver counterpart, effectively shifting the workload from the fine spatial mesh to the coarse grids.

Notably, in the plot for V_ℓ, the surrogate exhibits a slightly larger V_ℓ than the direct solver when M_ℓ exceeds 10^5. This discrepancy arises due to the insufficient number of sparse grid

nodes used for constructing the surrogate as the tolerance ϵ decreases. To ensure accuracy, the number of sparse grid nodes P needs to satisfy (5.28). However, in our case, to address a nearly constant interpolation error, we only use a fixed sparse grid level $q = 1$, which violates the accuracy requirement. Consequently, the large interpolation error (around 10^{-3} in the top right plot of Figure 6.15) becomes more pronounced over the discretization error as the tolerance falls below 10^{-3} (when M_ℓ exceeds 10^5). This leads to an increase in the total error $\|u - \check{u}_\ell\|$ of the surrogate approach, resulting in V_ℓ being slightly larger than that of the direct solver.

Similar to the discussion in [19], the estimation of the MLMC-FE estimator for (2.14) involves accumulating sample corrections $\widehat{Y}_{q,\ell}^{(i)}$ across spatial levels. These corrections incorporate $\widehat{u}_{q,\ell-1}^{(i)}$, computed on the coarse spatial grid using $\widehat{u}_{q,\ell}^{(i)}$ on the fine grid, through interpolation as a more efficient yet less accurate alternative to Galerkin projection. However, the use of non-nested geometry-conforming uniform spatial meshes introduces extrapolation errors during interpolation. To address this challenge, we propose a remedy involving the adoption of a sufficiently fine common grid (e.g., $\ell = 5$) that encompasses all coarser meshes, intending to minimize or eliminate extrapolation errors during the interpolation of surrogate corrections from all multi-level spatial coarser grids. In the cost plot displayed at the bottom left of Figure 6.15, considering a common fine grid to mitigate extrapolation error, the MLMC-FE sampling costs with both direct solve (solid) and surrogate (dotted) are highlighted in yellow, with cost estimates for both approaches approximating $\mathcal{O}(\epsilon^{-2})$. We observe a significant increase in cost attributed to the interpolation on a common grid for these two methods, albeit at the expense of heightened accuracy. Particularly noteworthy is that when ϵ is relatively large (above 10^{-3}), the MLMC-FE sampling cost of the two methods with this remedy exceeds the MC-FE cost. However, as ϵ decreases, the cost of MC-FE samplings escalates significantly, eventually surpassing the cost of MLMC-FE when considering the adoption of a common grid for interpolation.

Table 6.18 presents quantitative CPU time-based sampling costs and speedups when employing different strategies against the standard benchmark of MC-FE with direct computation. The last two columns detail the outcomes of mitigating extrapolation error through common fine grid usage for both direct solve and surrogate MLMC-FE. This table reveals that, for tolerances ranging from 2×10^{-4} to 8×10^{-3}, the surrogate-based MLMC method drastically reduces sampling times to between 0.1 and 14 seconds, in contrast to the 0.5 to

1.8×10^4 seconds required for direct computations. This reduction translates to cost savings by factors of 50 to 300. When these findings are juxtaposed with those from MC using direct computation—requiring 5.7 to 10^6 seconds—the enhanced MLMC strategy reveals a more dramatic speedup, at least 60 and often exceeding 10^4 as the tolerance decreases. It is noteworthy, however, that the interpolation process, while reducing extrapolation errors for MLMC-FE with both direct computation and surrogate methods, does not yield the expected efficiency gains. The process of interpolating each sample onto a very fine common grid incurs costs higher than those of obtaining a single realization using either method. Consequently, the total computational effort for both strategies aligns closely, with nearly identical speedups. This observation underscores that the efficiency benefit of interpolating to a common grid is less pronounced than anticipated, particularly in comparison to scenarios without such interpolation.

Table 6.19 shows the sample size estimations for both direct solver-based and surrogate-based MC-FE and MLMC-FE samplings, as quantified by (5.4), (5.14) and (5.30). The surrogate-based approaches (both MC-FE and MLMC-FE) demand a more stringent discretization error, indicated by a smaller splitting ratio ($\sqrt{0.5}/2$) within the nMSE error splitting, compared to the larger ratio $\sqrt{0.5}$ required by direct solver-based samplings. This implies that surrogate-based samplings might demand higher levels of the finest spatial grid. For

	MC-FE Direct solver	MC-FE Surrogate		MLMC-FE Direct solver		MLMC-FE Surrogate		MLMC-FE DS Interp		MLMC-FE Surrogate Interp	
ϵ	Time	Time	Speedup	Time	Speedup	Time	Speedup	Time	Speedup	Time	Speedup
8×10^{-3}	5.67e+00	5.16e-01	11.0	4.52e+00	1.3	9.78e-02	5.8e+01	6.49e+01	0.08	7.17e+01	0.08
6×10^{-3}	7.69e+00	1.33e+00	5.8	5.25e+00	1.5	9.81e-02	7.8e+01	9.49e+01	0.08	8.84e+01	0.09
4×10^{-3}	1.30e+02	1.67e+00	78.1	2.32e+01	5.6	1.23e-01	1.1e+03	2.16e+02	0.6	1.14e+02	1.1
2×10^{-3}	4.83e+02	4.07e+01	11.9	4.62e+01	10.5	2.91e-01	1.7e+03	6.56e+02	0.7	6.16e+02	0.8
10^{-3}	9.22e+03	1.80e+02	51.3	2.47e+02	37.3	6.77e-01	1.4e+04	2.92e+03	3.2	2.46e+03	3.8
8×10^{-4}	1.50e+04	2.60e+02	57.8	2.48e+02	60.5	8.36e-01	1.8e+04	4.63e+03	3.3	3.53e+03	4.3
6×10^{-4}	2.48e+04	1.93e+03	12.9	4.13e+02	60.0	1.97e+00	1.3e+04	8.26e+03	3.0	6.63e+03	3.8
4×10^{-4}	5.68e+04	4.12e+03	13.8	9.29e+02	61.1	3.33e+00	1.7e+04	1.69e+04	3.4	1.62e+04	3.5
2×10^{-4}	9.62e+05*	1.76e+04	54.6	4.21e+03	228.5	1.43e+01	6.7e+04	7.13e+04	13.5	6.38e+04	15.1

Table 6.18: The CPU time in seconds together with speedups for the multilevel methods, for a variety of choices of ϵ. For MC-FE with direct solver and surrogate, MLMC-FE with direct solver and surrogate, and MLMC-FE with direct solver and surrogate with solution interpolating to a common fine grid of level $\ell = 5$. The computational cost associated with a tolerance of $\epsilon = 2 \times 10^{-4}$ for Monte Carlo was prohibitive; the entry in the table for this tolerance (with an asterisk) is an estimate.

instance, in MC-FE sampling with tolerances $\epsilon = 8\times10^{-3}, 6\times10^{-3}, 2\times10^{-3}, 6\times10^{-4}, 4\times10^{-4}$, the direct solver approach operates with one less spatial refinement level than its surrogate counterpart. This requirement holds constant for MLMC-FE samplings, requiring identical finest spatial levels to those in MC-FE to maintain consistent discretization error across methods. Furthermore, both direct solver-based and surrogate-based samplings exhibit a similar trend in sample size estimation for both MC-FE and MLMC-FE. As the tolerance ϵ diminishes, the need for more samples escalates irrespective of the computational strategy. In MLMC-FE samplings, sample sizes demonstrate a decrease moving from coarser to finer grids. It is observed that surrogate-based MLMC-FE sampling incurs a marginally higher sample size requirement, this is due to the fact the V_ℓ in the bottom right plot of Figure 6.15 indicates a slower reduction for the surrogate compared to the direct solver as well as given both C_ℓ and C_ℓ^e have roughly similar rates at around 1.

6.3.4 Accuracy study

Plasma boundary. The plasma boundary of the expected poloidal flux in (2.14) is displayed in Figure 6.16. As discussed in [19], direct solver-based MC-FE sampling on a single spatial mesh yields a smooth plasma boundary, while direct solver-based MLMC-FE sampling on non-nested geometry-conforming meshes results in significant deformation near the x-point region due to extrapolation errors introduced during the accumulation of sample corrections (first and third columns in Figure 6.16). Surrogate-based sampling methods, both for MC-FE and MLMC-FE, exhibit similar plasma boundary behavior around the x-point region as their direct solver-based counterparts (second and fourth columns in the picture). To

	Level ℓ					
ϵ	0	1	2	3	4	5
8×10^{-3}			5			
6×10^{-3}			7			
4×10^{-3}				22		
2×10^{-3}				83		
10^{-3}					322	
8×10^{-4}					527	
6×10^{-4}					869	
4×10^{-4}					1980	
2×10^{-4}						8000*

	Level ℓ					
ϵ	0	1	2	3	4	5
8×10^{-3}				4		
6×10^{-3}				11		
4×10^{-3}				14		
2×10^{-3}					83	
10^{-3}					347	
8×10^{-4}					501	
6×10^{-4}						901
4×10^{-4}						1995
2×10^{-4}						8326

	Level ℓ					
ϵ	0	1	2	3	4	5
8×10^{-3}	10	2	2			
6×10^{-3}	12	3	2			
4×10^{-3}	32	5	2	2		
2×10^{-3}	152	26	4	2		
10^{-3}	691	109	18	4	2	
8×10^{-4}	841	129	23	3	2	
6×10^{-4}	1610	231	40	8	2	
4×10^{-4}	3791	589	104	15	3	
2×10^{-4}	15859	2344	375	62	13	2

	Level ℓ					
ϵ	0	1	2	3	4	5
8×10^{-3}	10	2	2	2		
6×10^{-3}	10	3	2	2		
4×10^{-3}	37	8	2	2		
2×10^{-3}	151	22	4	2	2	
10^{-3}	673	134	22	5	2	
8×10^{-4}	1017	173	29	8	2	
6×10^{-4}	1867	315	51	15	4	2
4×10^{-4}	4160	686	108	27	6	2
2×10^{-4}	17646	2919	435	102	20	7

Table 6.19: The optimal sample size estimation. Left: MC-FE sampling with the direct solver. Middle left: MC-FE with the surrogate. Middle right: MLMC-FE with the direct solver. Right: MLMC-FE with the surrogate.

mitigate the impact of extrapolation errors in surrogate-based MLMC-FE, we interpolate the surrogate evaluation corrections onto a very fine spatial common grid with a boundary containing all the coarse meshes. The resulting plasma boundary of the expected poloidal flux is presented in Figure 6.17, where the large deformations disappear.

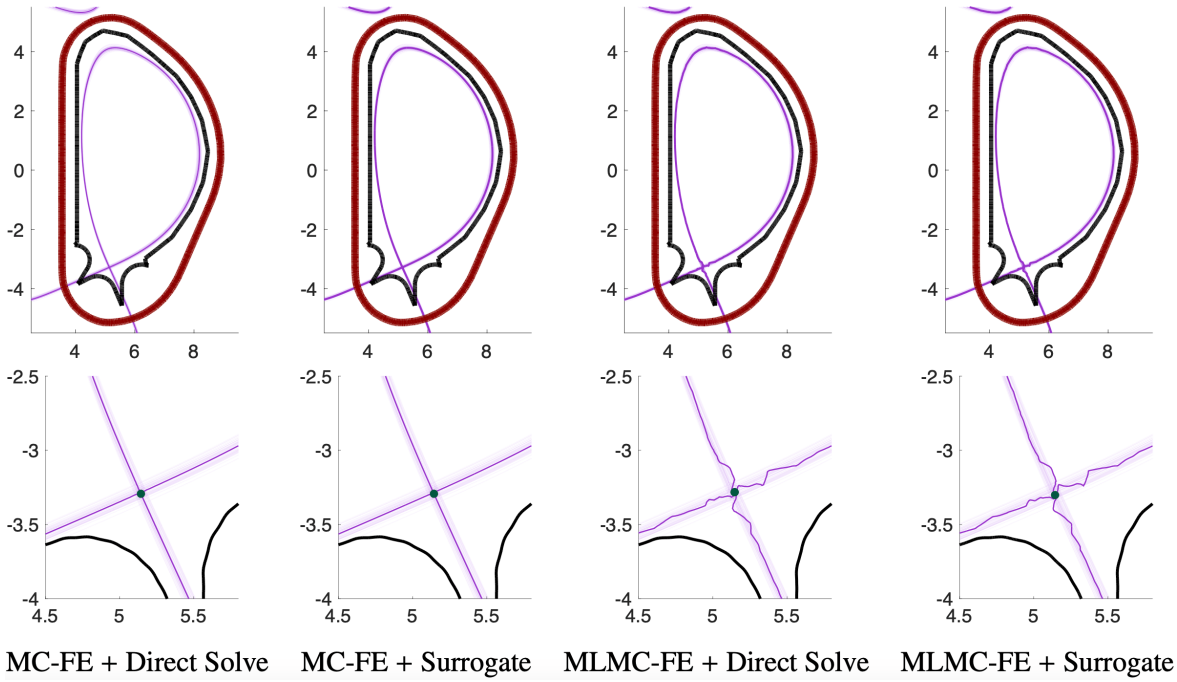

Figure 6.16: The overlayed plasma boundaries of 50 random realizations are displayed in the top row as violet curves (interpolate to the neighbor finer mesh). The solid violet line is the plasma boundary of the expected poloidal flux generated with tolerance $\epsilon = 4 \times 10^{-4}$. The inner and outer walls of the reactor are displayed in solid black and dark red respectively. The bottom row shows the regions close to the x-points in more detail. The dark green dots are the x-points of the expected solution. Each column from left to right corresponds to simulations of the MC-FE approach with the direct solver and surrogate, MLMC-FE with direct solver and surrogate. All simulations were performed using the discretization level $\ell = 5$ on geometry-conforming uniform meshes.

Geometric descriptors. Table 6.20 provides the geometric parameters that are derived from the plasma boundaries of the expected plasma field associated with various hybrid methods. In the table, we set the MC-FE sampling with a direct solver as the benchmark for comparison, we observe that the geometric descriptors match up to two decimal places for the

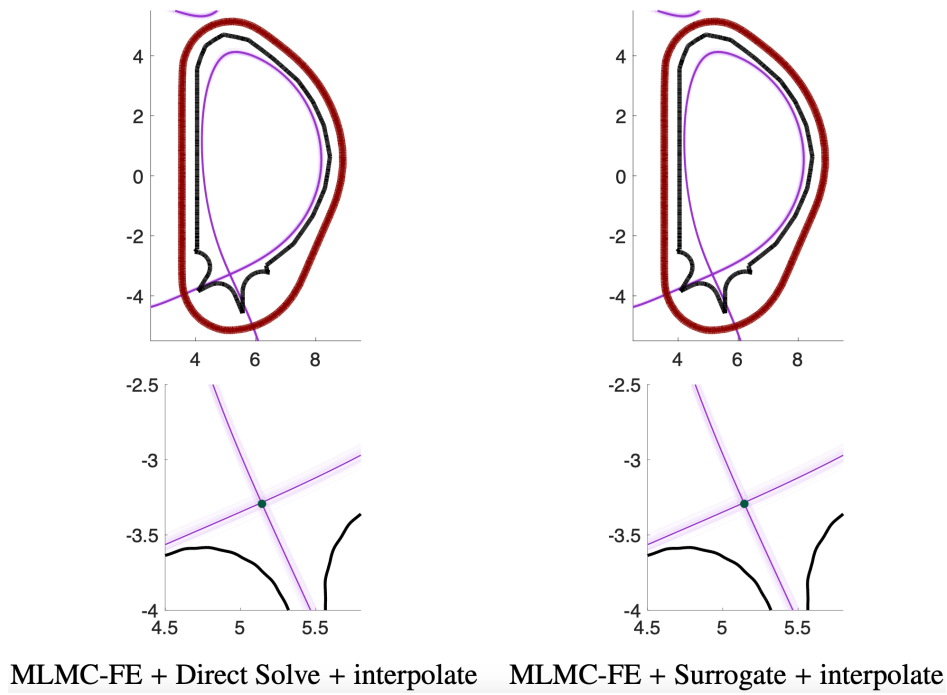

Figure 6.17: The overlayed plasma boundaries of 50 random realizations are displayed in the top row as violet curves. The solid violet line is the plasma boundary of the expected poloidal flux generated with tolerance $\epsilon = 4 \times 10^{-4}$. The inner and outer walls of the reactor are displayed in solid black and dark red respectively. The bottom row shows the regions close to the x-points in more detail. The dark green dots are the x-points of the expected solution. Each column from left to right corresponds to: the simulation with the MLMC-FE approach with a direct solver and surrogate, with solutions interpolated to a common fine mesh of level $\ell = 5$. All simulations were performed using the discretization level $\ell = 5$ on geometry-conforming uniform meshes.

surrogate-based MC-FE and MLMC-FE samplings with interpolation on the common grid. However, for MLMC-FE samplings without interpolation on a common grid, the geometric descriptors align only up to one decimal digit. This discrepancy is attributed to plasma boundary distortion arising from the extrapolation error on a geometry-conforming uniform mesh set.

	MC-FE DS	MC-FE Surrogate	MLMC-FE DS	MLMC-FE Surrogate	MLMC-FE DS (Interp)	MLMC-FE Surrogate (Interp)
x point	(5.14,-3.29)	(5.14,-3.29)	(5.14,-3.29)	(5.14, -3.30)	(5.14,-3.29)	(5.14,-3.29)
magnetic axis	(6.41,0.61)	(6.41,0.61)	(6.44,0.56)	(6.44, 0.56)	(6.41,0.61)	(6.41,0.61)
strike points	(4.16,-3.71) (5.56,-4.22)	(4.16,-3.71) (5.56,-4.22)	(4.16,-3.71) (5.56,-4.22)	(4.16, -3.71) (5.56, -4.22)	(4.16,-3.71) (5.56,-4.22)	(4.16,-3.71) (5.56,-4.22)
inverse aspect ratio	0.32	0.32	0.32	0.32	0.32	0.32
elongation	1.86	1.86	1.87	1.87	1.86	1.86
upper triangularity	0.43	0.43	0.43	0.46	0.43	0.43
lower triangularity	0.53	0.53	0.53	0.53	0.53	0.53

Table 6.20: Geometric parameters of the expected poloidal flux ψ from MC-FE with direct solver, MC-FE with the surrogate, MLMC-FE with direct solver, MLMC-FE with the surrogate, MLMC-FE with direct solver with interpolating solution to a common fine grid of level $\ell = 5$, MLMC-FE with surrogate with interpolating solution to a common fine grid of level $\ell = 5$. The results are generated with an nMSE 4×10^{-4} on the geometry-conforming uniform mesh set.

Chapter 7: Concluding remarks

The work comprised in this thesis focuses on advancing computational algorithms for quantifying uncertainty in the Grad-Shafranov free boundary problem. Employing methodologies such as sparse grid stochastic collocation, Monte Carlo finite element methods, multilevel Monte Carlo finite element methods, and hybrid techniques, the research aims to develop efficient approaches for assessing uncertainty impact on the model problem.

The initial approach focuses on assessing the accuracy and efficiency of the surrogate-based Monte Carlo sampling strategy using sparse grid stochastic collocation to define the surrogate. Regarding accuracy, we investigate the influence of stochasticity in the parameters on key features of the plasma boundary, including the location of the x-point, strike points, and shaping attributes like triangularity and elongation. Numerical findings indicate that surrogate functions closely emulate the direct solver concerning the plasma boundary. While the surrogate computations generally align with the results of the direct solution, a slight loss of accuracy is observed with increasing perturbation size. Our efficiency study primarily concentrates on sampling costs. Surrogate-based approaches incur non-negligible costs associated with surrogate construction. However, surrogates can be stored and reused once constructed, rendering the cost to be a one-time investment amortized over subsequent sampling sessions. Numerical results demonstrate that the use of surrogates reduces the CPU time required for direct Monte Carlo sampling by factors ranging from 7 to over 30. These outcomes underscore the efficiency and accuracy advantages of surrogate-based sampling over direct computation sampling.

The second part of the thesis explores potential cost savings in sampling through the multilevel Monte Carlo method, leveraging a hierarchical spatial mesh set for cost redistribution. In the absence of a predefined hierarchical mesh set for our model problem, we construct a sequence of uniformly refined grids tailored to the intricate mesh geometry, featuring curved boundaries that require careful handling. Our exploration of multilevel

approaches involves sampling on both geometry-conforming uniform grids and adaptively refined grids guided by a discrete a posteriori error estimator [17, 37, 38, 47, 52, 66]. Numerical results reveal substantial efficiency improvement for the multilevel method when sampled on geometry-conforming meshes, ranging from 60 to over 200. However, in terms of accuracy, this method introduces notable distortions in the plasma boundary due to the extrapolation error arising during the accumulation of interpolated sample corrections between non-nested physical meshes. This difficulty is resolved by adaptive gridding, which yields comparable sampling costs but provides more accurate computations of geometric quantities such as x-points associated with the model. Moreover, as a remedy for the extrapolation error observed in uniform MLMC sampling, we mitigate or eliminate it by implementing a post-processing procedure involving interpolation on a very fine common mesh encompassing all coarse meshes. Numerical results demonstrate that this strategy effectively eliminates inaccurate representations of the plasma boundary but it incurs computational costs associated with interpolation, thus reducing time savings for MLMC. Therefore further explorations on the handling geometric with curved boundaries should be considered.

In the last part of the thesis, we dive into hybrid methods that integrate surrogates with multilevel Monte Carlo, aiming to achieve further reductions in sampling costs compared to direct computation within the multilevel Monte Carlo framework. We propose distinct surrogate types designed to replace direct computations in online sampling processes for MC and MLMC to estimate the expectation. For each strategy, we establish the optimal construction and sampling costs through a comprehensive analysis of normalized mean square error. Given the slow decay of large interpolation errors inherent in our model problem, arising from the high dimensionality of the parameter space, our focus is exclusively on exploring MLMC with single-level sparse grid stochastic collocation and multilevel spatial discretization. Our numerical experiments reveal that, for a tolerance range of 2×10^{-4} to 8×10^{-3}, the surrogate-based MLMC method can perform samplings within a remarkably short span of 0.1 to 14 seconds, thus slashing direct computation costs by factors of 50 to 300. When contrasted with MC using direct computation, the enhanced MLMC strategy showcases a substantial speedup, with a minimum of 60 and potentially reaching up to 6×10^4 as the tolerance level is lowered. Accuracy assessments indicate that the plasma boundary and geometric descriptors resulting from surrogate-based sampling closely emulate those obtained through direct computations. Moreover, adopting a common fine grid for interpola-

tion yields smooth plasma boundaries in both surrogate-based and direct computation-based approaches, though this comes at the cost of diminished sampling efficiency.

Appendix A: Appendix

For completeness, we present a detailed proof for the theorems outlined in Chapter 5.2.3.

A.0.1 Proof of Theorem 1

The proof of this theorem follows the methodology outlined in [68]. Recall that the sample-wise cost for the direct solver to compute one realization is given by $C_k = M_k^{\gamma}$, with $M_k = M_0 s^k$ and $s > 1$. To estimate the cost for building the multilevel in both parameter and spatial grids surrogate in (4.19) in terms of tolerance ϵ, let H_1 and H_2 represent the two terms within the upper bound of the cost $\mathscr{W}_{\text{ML-ML}}^{\text{on}}$, given by

$$H_1 := \left(\frac{C_p}{\theta_2 \epsilon}\right)^{\frac{1}{\mu_1}} \left(\sum_{k=0}^{\ell} M_k^{\frac{\gamma\mu_1-\alpha_2}{\mu_1+1}}\right)^{\frac{1}{\mu_1}+1} = \left(\frac{C_p}{\theta_2 \epsilon}\right)^{\frac{1}{\mu_1}} \left(M_0^{\frac{\gamma\mu_1-\alpha_2}{\mu_1+1}} \sum_{k=0}^{\ell} s_1^k\right)^{\frac{1}{\mu_1}+1},$$

$$H_2 := \sum_{k=0}^{\ell} M_k^{\gamma} = M_0^{\gamma} \sum_{k=0}^{\ell} s^{\gamma k},$$

where $s_1 = s^{\frac{\gamma\mu_1-\alpha_2}{\mu_1+1}}$ and $H_1 \leq \mathscr{W}_{\text{ML-ML}}^{\text{on}} \leq H_1 + H_2$. Let $\eta \in \mathbb{R}$. The behavior of $\sum_{k=0}^{\ell} s^{\eta k}$ is described by

$$\sum_{k=0}^{\ell} s^{\eta k} = \begin{cases} \frac{1}{1-s^{\eta}} = \mathcal{O}(1), & \eta < 0, \\ \ell + 1 = \mathcal{O}(|\log \epsilon|), & \eta = 0, \\ \mathcal{O}\left(\epsilon^{-\frac{\eta}{\alpha_1}}\right), & \eta > 0, \end{cases} \tag{A.1}$$

which can be proved using the geometric summation and the optimal spatial mesh level in (4.11). The scenario when $\eta > 0$ can be shown as

$$\sum_{k=0}^{\ell} s^{\eta k} = \frac{s^{\eta(\ell+1)} - 1}{s^{\eta} - 1} < \frac{s^{\eta \ell}}{1 - s^{-\eta}} < \frac{s^{\frac{\eta}{\alpha_1} \log_s\left(C' \epsilon^{-1}\right) + \eta}}{1 - s^{-\eta}} = \frac{\left(C' s^{\alpha_1}\right)^{\frac{\eta}{\alpha_1}}}{1 - s^{-\eta}} \epsilon^{-\frac{\eta}{\alpha_1}} = \mathcal{O}\left(\epsilon^{-\frac{\eta}{\alpha_1}}\right).$$

Using (A.1) and the assumption on parameters in Theorem 1, an upper bound for the geometric sum in H_2 is determined as $\sum_{k=0}^{\ell} s^{\gamma k} \leq \mathcal{O}\left(\epsilon^{-\frac{\gamma}{\alpha_1}}\right) < \mathcal{O}\left(\epsilon^{-\frac{1}{\mu_1}}\right)$. Consequently,

$H_2 = \mathcal{O}\left(\epsilon^{-\frac{1}{\mu_1}}\right)$. To estimate H_1, let $\eta = \frac{\gamma\mu_1 - \alpha_2}{\mu_1+1}$ in (A.1). The behavior of $\sum_{k=0}^{\ell} s_1^k$ and H_1 are characterized by

$$\sum_{k=0}^{\ell} s_1^k = \begin{cases} \frac{1}{1-s_1} = \mathcal{O}(1), & \gamma\mu_1 < \alpha_2, \\ \ell + 1 = \mathcal{O}(|\log\epsilon|), & \gamma\mu_1 = \alpha_2, \\ \mathcal{O}\left(\epsilon^{-\frac{\gamma\mu_1-\alpha_2}{\alpha_1(\mu_1+1)}}\right), & \gamma\mu_1 > \alpha_2, \end{cases} \qquad H_1 = \begin{cases} \mathcal{O}\left(\epsilon^{-\frac{1}{\mu_1}}\right), & \gamma\mu_1 < \alpha_2, \\ \mathcal{O}\left(\epsilon^{-\frac{1}{\mu_1}} |\log\epsilon|^{\frac{1}{\mu_1}+1}\right), & \gamma\mu_1 = \alpha_2, \\ \mathcal{O}\left(\epsilon^{-\frac{1}{\mu_1}-\frac{\gamma\mu_1-\alpha_2}{\alpha_1\mu_1}}\right), & \gamma\mu_1 > \alpha_2. \end{cases}$$

Notice that the second term H_2 is dominated by the first term H_1 in all three scenarios, under the assumption in Theorem 1. This indicates that both the lower bound H_1 and the upper bound $H_1 + H_2$ exhibit the same behavior as H_1 and so does $\mathscr{W}^{\text{on}}_{\text{ML-ML}}$. $\square$

Bibliography

[1] R. Albanese, J. Blum, and O. D. Barbieri. On the solution of the magnetic flux equation in an infinite domain. In S. Methfessel and G. Thomas, editors, 8th. European conference on Computational Physics: Computing in plasma physics, volume 10D of Europhysics Conference Abstracts, pages 41–44. European Physical Society, 1986.

[2] M. Ariola and A. Pironti. Magnetic control of tokamak plasmas. Advances in Industrial Control. Springer-Verlag London, Ltd., London, 2008.

[3] L. Artsimovich, G. Bobrovskii, and E. Gorbunov. Experiments in tokamak devices. pp 157-73 of Plasma Physics and Controlled Nuclear Fusion Research. Vol. I. Vienna, International Atomic Energy Agency, 1969., 10 1969.

[4] A. Barth, C. Schwab, and N. Zollinger. Multi-level Monte Carlo finite element method for elliptic PDEs with stochastic coefficients. Numerische Mathematik, 119(1):123–161, 2011.

[5] V. Barthelmann, E. Novak, and K. Ritter. High dimensional polynomial interpolation on sparse grids. Advances in Computational Mathematics, 12:273–288, 2000.

[6] E. A. Belli, G. W. Hammett, and W. Dorland. Effects of plasma shaping on nonlinear gyrokinetic turbulence. Physics of Plasmas, 15(9):092303, 2008.

[7] K. Bol, M. Okabayashi, and R. Fonck. The poloidal divertor experiment (PDX) and the Princeton beta experiment (PBX). Nuclear Fusion, 25(9):1149–1153, sep 1985.

[8] S. Boyd and L. Vandenberghe. Convex optimization. Cambridge University Press, Cambridge, 2004.

[9] C. Carstensen. An adaptive mesh-refining algorithm allowing for an H_1 stable L_2 projection onto courant finite element spaces. Constructive Approximation, 20(4):549–564, Oct. 2003.

[10] C. Carstensen, M. Feischl, M. Page, and D. Praetorius. Axioms of adaptivity. Computers & Mathematics with Applications, 67(6):1195–1253, Apr. 2014.

[11] A. J. Cerfon and J. P. Freidberg. "One size fits all" analytic solutions to the Grad-Shafranov equation. Physics of Plasmas, 17(3):032502, 2010.

[12] K. A. Cliffe, M. B. Giles, R. Scheichl, and A. L. Teckentrup. Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients. Computing and Visualization in Science, 14(1):3–15, 2011.

[13] G. Da Prato and J. Zabczyk. Encyclopedia of Mathematics and its Applications: Stochastic Equations in Infinite Dimensions, Series Number 152. Cambridge University Press, Cambridge, England, 2 edition, Apr. 2014.

[14] T. A. Davis. Direct methods for sparse linear systems, volume 2 of Fundamentals of Algorithms. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2006.

[15] K. Dietz, S. Chiocchio, A. Antipenkov, G. Federici, G. Janeschitz, E. Martin, R. Parker, and R. Tivey. Engineering and design aspects related to the development of the ITER divertor. Fusion Engineering and Design, 27(Supplement C):96 – 108, 1995. Proceedings of the Third International Symposium on Fusion Nuclear Technology.

[16] W. Dörfler. A convergent adaptive algorithm for Poisson's equation. SIAM Journal on Numerical Analysis, 33(3):1106–1124, 1996.

[17] M. Eigel, C. Merdon, and J. Neumann. An adaptive multilevel Monte Carlo method with stochastic bounds for quantities of interest with uncertain data. SIAM/ASA Journal on Uncertainty Quantification, 4(1):1219–1245, 2016.

[18] H. C. Elman, J. Liang, and T. Sánchez-Vizuet. Surrogate approximation of the Grad-Shafranov free boundary problem via stochastic collocation on sparse grids. Journal of Computational Physics, 448:110699, 20, 2022.

[19] H. C. Elman, J. Liang, and T. Sánchez-Vizuet. Multilevel Monte Carlo methods for the Grad-Shafranov free boundary problem. Computer Physics Communications, 298:109099, 2024.

[20] H. R. Fairbanks, A. Doostan, C. Ketelsen, and G. Iaccarino. A low-rank control variate for multilevel Monte Carlo simulation of high-dimensional uncertain systems. Journal of Computational Physics, 341:121–139, 2017.

[21] B. Faugeras and H. Heumann. FEM-BEM coupling methods for Tokamak plasma axisymmetric free-boundary equilibrium computations in unbounded domains. Journal of Computational Physics, 343:201 – 216, 2017.

[22] S. Funken, D. Praetorius, and P. Wissgott. Efficient implementation of adaptive P1-FEM in Matlab (extended preprint). Technical report, Institute for Analysis and Scientific Computing, Vienna University of Technology, 2008.

[23] S. Funken, D. Praetorius, and P. Wissgott. Efficient implementation of adaptive P1-FEM in Matlab. Computational Methods in Applied Mathematics, 11(4):460–490, 2011.

[24] D. A. Gates, R. Maingi, J. S. Menard, and et al. Effect of plasma shaping on performance in the National Spherical Torus Experiment. Physics of Plasmas, 13(5):056122, 2006.

[25] G. Gatica and G. Hsiao. The uncoupling of boundary integral and finite element methods for nonlinear boundary value problems. Journal of Mathematical Analysis and Applications, 189(2):442 – 461, 1995.

[26] A. George. Nested dissection of a regular finite element mesh. SIAM J. Numer. Anal., 10:345–363, 1973.

[27] M. B. Giles. Multilevel Monte Carlo path simulation. Operations Research, 56(3):607–617, 2008.

[28] M. B. Giles. Multilevel Monte Carlo methods. Acta Numerica, 24:259–328, 2015.

[29] P. E. Gill, W. Murray, and M. H. Wright. Practical Optimization. Academic Press, Inc. [Harcourt Brace Jovanovich, Publishers], London-New York, 1981.

[30] G. H. Golub and C. F. Van Loan. Matrix computations. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, Baltimore, MD, fourth edition, 2013.

[31] P.-A. Gourdain, J.-N. Leboeuf, and R. Neches. High-resolution magnetohydrodynamic equilibrium code for unity beta plasmas. Journal of Computational Physics, 216(1):275–299, July 2006.

[32] H. Grad and H. Rubin. Hydromagnetic equilibria and force-free fields. In Proc. Second international conference on the peaceful uses of atomic energy, Geneva, volume 31,190, New York, Oct 1958. United Nations.

[33] V. Grandgirard. Modelisation de l'equilibre d'un plasma de tokamak. Technical report, Université de Franche-Comté, 1999.

[34] H. Heumann. FEEQS.M. http://www-sop.inria.fr/members/Holger.Heumann/Software.html.

[35] H. Heumann, J. Blum, C. Boulbe, B. Faugeras, G. Selig, J.-M. Ané, S. Brémond, V. Grandgirard, P. Hertout, E. Nardon, and et al. Quasi-static free-boundary equilibrium of toroidal plasma with CEDRES: Computational methods and applications. Journal of Plasma Physics, 81(3):905810301, 2015.

[36] H. Heumann and F. Rapetti. A finite element method with overlapping meshes for free-boundary axisymmetric plasma equilibria in realistic geometries. Journal of Computational Physics, 334:522 – 540, 2017.

[37] H. k. Hoel, E. von Schwerin, A. Szepessy, and R. Tempone. Adaptive multilevel Monte Carlo simulation. In Numerical Analysis of Multiscale Computations, volume 82 of Lect. Notes Comput. Sci. Eng., pages 217–234. Springer, Heidelberg, 2012.

[38] H. k. Hoel, E. von Schwerin, A. Szepessy, and R. Tempone. Implementation and analysis of an adaptive multilevel Monte Carlo algorithm. Monte Carlo Methods and Applications, 20(1):1–41, 2014.

[39] F. Hofmann. FBT - a free-boundary tokamak equilibrium code for highly elongated and shaped plasmas. Computer Physics Communications, 48(2):207–221, Feb. 1988.

[40] C. T. Holcomb and J. R. e. a. Ferron. Optimizing stability, transport, and divertor operation through plasma shaping for steady-state scenario development in DIII-D. Physics of Plasmas, 16(5):056116, 2009.

[41] G. C. Hsiao and W. L. Wendland. Boundary Element Methods: Foundation and Error Analysis, chapter 12. John Wiley & Sons, 2004.

[42] G. C. Hsiao and S. Zhang. Optimal order multigrid methods for solving exterior boundary value problems. SIAM Journal on Numerical Analysis, 31(3):680–694, 1994.

[43] G. Janeschitz, K. Borrass, G. Federici, Y. Igitkhanov, A. Kukushkin, H. D. Pacher, G. W. Pacher, and M. Sugihara. The ITER divertor concept. Journal of Nuclear Materials, 220:73–88, Apr 1995.

[44] C. Johnson and J. C. Nédélec. On the coupling of boundary integral and finite element methods. Mathematics of Computation, 35(152):1063–1079, 1980.

[45] J. Johnson, H. Dalhed, J. Greene, R. Grimm, Y. Hsieh, S. Jardin, J. Manickam, M. Okabayashi, R. Storer, A. Todd, D. Voss, and K. Weimer. Numerical determination of axisymmetric toroidal magnetohydrodynamic equilibria. Journal of Computational Physics, 32(2):212–234, Aug. 1979.

[46] J. Kesner. Tripe x-point tokamak divertor. Nuclear Fusion, 30(3):548–551, mar 1990.

[47] A. Khodadadian, M. Parvizi, and C. Heitzinger. An adaptive multilevel Monte Carlo algorithm for the stochastic drift-diffusion-Poisson system. Computer Methods in Applied Mechanics and Engineering, 368:113163, 23, 2020.

[48] J. E. Kinsey, R. E. Waltz, and J. Candy. The effect of plasma shaping on turbulent transport and E×B shear quenching in nonlinear gyrokinetic simulations. Physics of Plasmas, 14(10):102306, 2007.

[49] A. Klimke, K. Willner, and B. Wohlmuth. Uncertainty modeling using fuzzy arithmetic based on sparse grids: applications to dynamic systems. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 12(6):745–759, 2004.

[50] A. Klimke and B. Wohlmuth. Algorithm 847: spinterp: piecewise multilinear hierarchical sparse grid interpolation in MATLAB. Association for Computing Machinery. Transactions on Mathematical Software, 31(4):561–579, 2005.

[51] R. Kornhuber and E. Youett. Adaptive multilevel Monte Carlo methods for stochastic variational inequalities. SIAM Journal on Numerical Analysis, 56(4):1987–2007, 2018.

[52] S. Larsson and V. Thomée. Partial Differential Equations with Numerical Methods, volume 45 of Texts in Applied Mathematics. Springer-Verlag, Berlin, 2003.

[53] R. F. Ling. Comparison of several algorithms for computing sample means and variances. Journal of the American Statistical Association, 69:859–866, 1974.

[54] T. C. Luce. An analytic functional form for characterization and generation of axisymmetric plasma boundaries. Plasma Physics and Controlled Fusion, 55(9):095009, Jul 2013.

[55] R. Lüst and A. Schlüter. Axialsymmetrische magnetohydrodynamische Gleichgewichtskonfigurationen. Z. Naturf, 12a:850–854, 1957.

[56] J. Luxon and B. Brown. Magnetic analysis of non-circular cross-section tokamaks. Nuclear Fusion, 22(6):813–821, jun 1982.

[57] N. Metropolis and S. Ulam. The Monte Carlo method. Journal of the American Statistical Association, 44:335–341, 1949.

[58] P. Morin, R. H. Nochetto, and K. G. Siebert. Data oscillation and convergence of adaptive FEM. SIAM Journal on Numerical Analysis, 38(2):466–488, Jan. 2000.

[59] M. Motamed and D. Appelö. A multiorder discontinuous Galerkin Monte Carlo method for hyperbolic problems with stochastic parameters. SIAM Journal on Numerical Analysis, 56(1):448–468, 2018.

[60] F. Nobile, R. Tempone, and C. G. Webster. A sparse grid stochastic collocation method for partial differential equations with random input data. SIAM Journal on Numerical Analysis, 46(5):2309–2345, 2008.

[61] F. Nobile and F. Tesei. A multi level Monte Carlo method with control variate for elliptic PDEs with log-normal coefficients. Stochastic Partial Differential Equations. Analysis and Computations, 3(3):398–444, 2015.

[62] J. Nocedal and S. J. Wright. Numerical optimization. Springer Series in Operations Research and Financial Engineering. Springer, New York, second edition, 2006.

[63] H. Robbins and J. Van Ryzin. Introduction to Statistics. Science Research Associates, Chicago, 1975.

[64] D. D. Ryutov and V. A. Soukhanovskii. The snowflake divertor. Physics of Plasmas, 22(11):110901, 2015.

[65] V. D. Shafranov. On magnetohydrodynamical equilibrium configurations. Soviet Physics JETP, 6:545–554, 1958.

[66] J. R. Shewchuk. Delaunay refinement algorithms for triangular mesh generation. Computational Geometry, 22(1-3):21–74, May 2002.

[67] S. A. Smolyak. Quadrature and interpolation formulae on tensor products of certain function classes. Doklady Akademii Nauk SSSR, 148:1042–1045, 1963.

[68] A. L. Teckentrup, P. Jantsch, C. G. Webster, and M. Gunzburger. A multilevel stochastic collocation method for partial differential equations with random input data. SIAM/ASA Journal on Uncertainty Quantification, 3(1):1046–1074, 2015.

[69] A. L. Teckentrup, R. Scheichl, M. B. Giles, and E. Ullmann. Further analysis of multilevel monte carlo methods for elliptic PDEs with random coefficients. Numerische Mathematik, 125(3):569–600, Mar. 2013.

[70] R. Verfürth. A posteriori error estimation techniques for finite element methods. Numerical Mathematics and Scientific Computation. Oxford University Press, Oxford, 2013.

[71] B. P. Welford. Note on a method for calculating corrected sums of squares and products. Technometrics, 4(3):419–420, 1962.

[72] J. Wesson. Tokamaks. Oxford University Press, 4th. edition, 2011.

[73] Y. Xiao and P. J. Catto. Plasma shaping effects on the collisionless residual zonal flow level. Physics of Plasmas, 13(8):082307, 2006.

[74] D. Xiu and J. Hesthaven. High-order collocation methods for differential equations with random inputs. SIAM Journal on Scientific Computing, 27(3):1118–1139, 2005.

[75] O. C. Zienkiewicz and J. Z. Zhu. The superconvergent patch recovery and a posteriori error estimates. part 1: The recovery technique. International Journal for Numerical Methods in Engineering, 33(7):1331–1364, 1992.

[76] O. C. Zienkiewicz and J. Z. Zhu. The superconvergent patch recovery and a posteriori error estimates. part 2: Error estimates and adaptivity. International Journal for Numerical Methods in Engineering, 33(7):1365–1382, 1992.