Chair for
Reliability Engineering

Mohamad Daher

Efficient Treatment of Boundary Uncertainty in FEM via Mesh Deformation without Remeshing

Master’s Thesis

Nr. 26.02.001

2026

Efficient Treatment of Boundary Uncertainty in FEM via Mesh Deformation without Remeshing

To obtain the academic degree
Master of Science in Mechanical Engineering
from the Faculty of Mechanical Engineering
at the Technische Universität Dortmund
Master's Thesis

submitted by
Mohamad Daher

Dortmund, 2026

Supervisor:	Prof. Dr. Matthias G.R. Faes Chair of Reliability Engineering Technische Universität Dortmund
Co-Supervisor:	Dr. Chao Dang Chair of Reliability Engineering Technische Universität Dortmund
Date of submission	17/03/2026

Abstract

This thesis investigates the efficient treatment of geometric boundary uncertainty in finite element analysis using Laplace-based mesh deformation within an Arbitrary Lagrangian–Eulerian (ALE) framework. The central idea is to preserve a fixed reference mesh topology and update only the nodal coordinates according to prescribed boundary perturbations, thereby avoiding the computational cost of repeated remeshing while maintaining a consistent discretization across uncertainty realizations.

The proposed methodology is validated through benchmark problems of increasing complexity, culminating in a plate-with-hole configuration with uncertain internal boundary shape and position. A refined-reference comparison confirms that the deformation-based solution remains physically accurate within the considered perturbation range, with relative errors in the peak response below 0.1%. A timing decomposition shows that end-to-end speedups of 41–$111\times$ are achieved by eliminating repeated geometry handling and mesh import, with an estimated 9–$13\times$ speedup under in-memory mesh generation.

Two uncertainty propagation settings are demonstrated: a probabilistic model using Monte Carlo simulation, and an interval model using surrogate-based Non-Parallel Bayesian Global Optimization (NPBGO). In both cases, the non-intrusive ALE framework serves as a black-box evaluator, producing well-behaved statistics and tighter response bounds with fewer evaluations than uniform sampling.

Keywords
Laplace-based mesh deformation, geometric boundary uncertainty, finite element analysis, uncertainty quantification, remeshing avoidance.

Abstrakt

Diese Arbeit untersucht die effiziente Behandlung geometrischer Randunsicherheiten in der Finite-Elemente-Analyse unter Verwendung einer Laplace-basierten Netzverformung innerhalb eines Arbitrary-Lagrangian-Eulerian-Rahmens (ALE). Der zentrale Ansatz besteht darin, eine feste Referenznetztopologie beizubehalten und lediglich die Knotenkoordinaten entsprechend vorgegebenen Randstörungen zu aktualisieren. Auf diese Weise werden die Rechenkosten wiederholter Neuvernetzungen vermieden, während gleichzeitig eine konsistente Diskretisierung über alle Unsicherheitsrealisierungen hinweg gewährleistet bleibt.

Die vorgeschlagene Methodik wird anhand von Benchmark-Problemen mit steigender Komplexität validiert, die in einer Platten-mit-Loch-Konfiguration mit unsicherer innerer Randform und -position kulminieren. Ein Vergleich mit einer verfeinerten Referenzlösung bestätigt, dass die verformungsbasierte Lösung innerhalb des betrachteten Störungsbereichs physikalisch genau bleibt, wobei die relativen Fehler des Spitzenwerts der Antwort unter 0,1 % liegen. Eine Zerlegung der Rechenzeiten zeigt, dass durch den Wegfall wiederholter Geometrieerzeugung und Netzimporte End-to-End-Beschleunigungen von 41- bis 111-fach erreicht werden, mit einer geschätzten 9- bis 13-fachen Beschleunigung bei der Netzgenerierung im Arbeitsspeicher.

Es werden zwei Ansätze zur Unsicherheitsfortpflanzung demonstriert: ein probabilistisches Modell auf Basis einer Monte-Carlo-Simulation sowie ein Intervallmodell unter Verwendung einer surrogatbasierten nicht-parallelen Bayes'schen globalen Optimierung (NPBGO). In beiden Fällen fungiert das nicht-intrusive ALE-Framework als Black-Box-Evaluator, der mit weniger Auswertungen als bei gleichmäßiger Stichprobenahme stabile statistische Größen und engere Antwortgrenzen liefert.

Schlüsselwörter
Laplace-basierte Netzverformung, geometrische Randunsicherheit, Finite-Elemente-Analyse, Unsicherheitsquantifizierung, Vermeidung wiederholter Neuvernetzung.

List of Abbreviations

ALE	Arbitrary Lagrangian–Eulerian
DOE	Design of Experiments
EI	Expected Improvement
FEM	Finite Element Method
GP	Gaussian Process
MC	Monte Carlo
NPBGO	Non-Parallel Bayesian Global Optimization
PBGO	Bayesian Global Optimization
PDF	Probability Density Function
QoI	Quantity of Interest
UQ	Uncertainty Quantification

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Problem Description and Research Limitations

Geometric boundary uncertainty remains one of the most computationally expensive parameter spaces to explore in high-fidelity structural simulations [23]. Although the Finite Element Method (FEM) is a well-established numerical technique in mechanical and structural engineering, enabling engineers to simulate and predict the behavior of complex systems under given loads, boundary conditions, and material laws [32, 43], the precision of such simulations strongly depends on an accurate geometric description. As modern engineering systems become increasingly complex and demanding, the importance of reliable and efficient methods for evaluating uncertainties related to loads, geometry, material properties, manufacturing processes, and operational conditions has grown considerably [4]. In practice, uncertainties arise from several sources, including geometry, material properties, loading, and boundary conditions, each of which can significantly influence the predicted response of a structure and may interact in a non-additive manner [15].

A particularly critical and often overlooked source of uncertainty comes from imprecise boundary descriptions. Manufacturing tolerances, measurement errors, or unknown contact interfaces can cause small but important deviations of structural boundaries from their nominal geometry [23]. Standard finite element analyses generally assume a perfectly defined and fixed geometric description, an assumption that is known to influence the accuracy of numerical results [42, 17]. When uncertainty is introduced, traditional approaches rely on remeshing the computational domain for every realization of the random geometry. Although this method is conceptually simple, it is computationally expensive, introduces interpolation errors between different meshes, and complicates the comparison of simulation results [23].

The computational burden of repeated remeshing is further compounded by the uncertainty quantification (UQ) requirements. Stochastic finite element simulations often require hundreds or thousands of deterministic analyses to estimate statistical quantities such as mean, variance, or probability of failure. Monte Carlo sampling is widely used for this purpose due to its conceptual simplicity, robustness, and unbiased nature, making it a common reference method in uncertainty quantification [37]. However, when applied to high-fidelity finite element models, the computational cost of Monte Carlo simulations can become substantial, as each sample requires a full solution of the governing equations. In the presence of geometric uncertainty, this cost is further increased if remeshing is required for each realization, motivating strategies that improve efficiency at the level of geometric modeling rather than at the sampling level.

Alternative strategies have been developed to address different aspects of uncertainty propagation in finite element analysis. For example, Stocki et al. [37] proposed efficient sampling techniques such as Optimal Latin Hypercube Sampling (OLHS) to reduce the number of required model evaluations, while Moens and Vandepitte [26] reviewed non-probabilistic approaches such as interval and fuzzy finite element methods for handling epistemic uncertainty. However, these approaches primarily focus on the representation or sampling of uncertainty and do not directly address the computational overhead associated with repeated geometry regeneration in the presence of geometric uncertainty.

From a geometric modeling standpoint, mesh deformation techniques represent a promising alternative to remeshing-based approaches for the treatment of geometric uncertainty. Instead of regenerating a new mesh for each perturbed geometry, the existing mesh is smoothly deformed according to prescribed boundary changes [24]. The Arbitrary Lagrangian-Eulerian (ALE) formulation provides the theoretical foundation for such approaches by introducing an independent reference configuration that decouples mesh motion from material deformation [28]. Within the ALE framework, boundary motion can be consistently propagated into the interior of the computational domain without altering mesh topology, thereby avoiding repeated remeshing and maintaining mesh connectivity.

A practical realization of mesh motion within ALE-based frameworks is obtained by solving a Laplace-type equation for the mesh displacement field, ensuring a smooth and harmonic propagation of boundary perturbations into the interior of the domain. This Laplace-based deformation strategy has been successfully applied to finite element analysis on random domains, where it enables the mapping of geometrically uncertain configurations to a fixed reference mesh while preserving solution consistency across realizations [41].

In the treatment of geometric boundary uncertainty, two fundamentally different modeling paradigms can be distinguished. A probabilistic description characterizes uncertain geometric parameters as random variables and aims at estimating statistical measures of structural response. In contrast, a non-probabilistic interval description represents uncertain parameters by bounded sets without prescribing probability distributions. These two perspectives lead to different computational objectives. While probabilistic uncertainty quantification focuses on the estimation of statistical moments or response distributions, interval-based analysis seeks to determine the minimum and maximum attainable values of selected quantities of interest within an admissible parameter domain. The latter can be interpreted as a global optimization problem in the space of uncertain geometric parameters and poses additional computational challenges when coupled with high-fidelity finite element models, in which detailed numerical simulations are performed using fine discretizations and the full governing equations [27].

Despite the growing interest in mesh deformation techniques and ALE-based formulations for handling moving and uncertain boundaries, their systematic benchmarking against conventional remeshing within a fully non-intrusive uncertainty quantification workflow remains limited. To the best of the author's knowledge, a systematic evaluation of Laplace-based mesh deformation against a conventional remeshing baseline with respect to accuracy, robustness, and computational efficiency within a non-intrusive workflow has not yet been reported. Moreover, the influence of boundary perturbation magnitude and mesh resolution on the performance of such deformation-based approaches has not been comprehensively investigated.

To address these gaps, this thesis implements and validates a non-intrusive numerical framework for treating geometric boundary uncertainty using Laplace-based mesh deformation. The proposed approach is designed to retain computational efficiency and solution consistency across different realizations while avoiding repeated remeshing. In contrast to intrusive and weakly-intrusive stochastic finite element approaches on random domains (e.g., the weakly-intrusive formulation of [41]), the proposed framework retains a fully deterministic and non-intrusive finite element formulation on the deformed mesh and integrates both probabilistic and interval uncertainty treatments within a unified computational setting.

1.2 Research Objective

The main objective of this thesis is the implementation and validation of a numerical framework for the treatment of geometric boundary uncertainty in finite element analysis using Laplace-based mesh deformation. The framework aims to avoid repeated remeshing by deforming a reference mesh according to prescribed boundary perturbations, while preserving solution consistency across different realizations. Computational efficiency and numerical accuracy are assessed in comparison to conventional remeshing-based approaches.

The investigation comprises both probabilistic and interval descriptions of geometric boundary uncertainty. In the probabilistic setting (UM1), statistical response characteristics under geometric variability are analyzed within a Monte Carlo framework. In the interval setting (UM2), the determination of minimum and maximum values of selected quantities of interest is interpreted as a global optimization problem over the admissible parameter domain, addressed using a surrogate-assisted Bayesian optimization strategy.

The specific research questions addressed in this thesis are:

1. To what extent can Laplace-based mesh deformation accurately reproduce the geometric and mechanical effects of boundary perturbations in finite element simulations?
2. How does the proposed mesh deformation framework affect computational efficiency compared to conventional remeshing-based approaches when applied within a Monte Carlo uncertainty propagation setting?
3. How does the Laplace-based deformation approach perform under deliberately severe boundary perturbations and across different mesh densities?

The scope of this research is deliberately limited to linear static finite element problems with geometric boundary perturbations for which the Laplace-based mesh deformation preserves element validity. Within this setting, the proposed framework enables a systematic comparison between Laplace-based mesh deformation and conventional remeshing strategies. Nonlinear material behavior, large deformations, dynamic effects, and time-dependent uncertainties are outside the scope of this study, but represent relevant directions for future extensions.

The methodology builds upon established concepts in mesh motion and the Arbitrary Lagrangian-Eulerian (ALE) formulation, which provide a kinematic framework for mesh deformation without altering topology. Laplace-based mesh deformation is employed to propagate prescribed boundary perturbations smoothly into the interior of the computational domain.

For the probabilistic setting (UM1), Monte Carlo sampling serves as a reference approach, enabling a systematic evaluation of statistical response characteristics. The interval setting (UM2) complements this analysis by focusing on the efficient identification of response bounds within prescribed geometric parameter limits.

Primary Contributions: The main contributions of this thesis are:

- A modular, numerical implementation of a non-intrusive, Laplace-based ALE framework for geometric uncertainty quantification in finite element analysis, validated for linear elliptic problems with moderate boundary perturbations.
- A systematic quantification of the accuracy and computational efficiency of this approach against a conventional remeshing baseline, demonstrating significant end-to-end workflow speedups while preserving solution fidelity.
- A unified demonstration of the framework's applicability to both probabilistic (Monte Carlo) and interval (Bayesian global optimization) uncertainty descriptions for a representative two-dimensional benchmark problem, highlighting the necessity of optimization-based strategies for reliable interval bound identification.

The remainder of this thesis is structured as follows. Chapter 2 reviews the theoretical background on finite element methods, mesh motion techniques, and uncertainty quantification. Chapter 3 introduces the proposed numerical methodology. Chapter 4 describes the computational implementation. Chapter 5 presents numerical examples and validation studies. Finally, Chapter 6 summarizes the main findings and outlines future research directions.

Chapter 2

Theoretical Background

2.1 Finite Element Method for the Poisson Problem

In order to introduce the finite element formulation employed throughout this thesis, the scalar Poisson problem is adopted as a reference model. The Poisson equation represents a broad class of linear elliptic boundary value problems and is widely used in computational mechanics to illustrate fundamental finite element concepts in a clear and compact manner [32, 43]. Moreover, its mathematical structure closely resembles the Laplace equation employed for mesh deformation, making it particularly suitable for the present study.

Let $\Omega \subset \mathbb{R}^d$ $(d = 1, 2, 3)$ denote the computational domain with boundary $\partial\Omega = \Gamma_D \cup \Gamma_N$, where Γ_D and Γ_N represent the Dirichlet and Neumann parts of the boundary, respectively.

The strong form of the Poisson problem is given by

$$-\nabla \cdot (k\nabla u) = f \quad \text{in } \Omega, \tag{2.1}$$

$$u = \bar{u} \quad \text{on } \Gamma_D, \tag{2.2}$$

$$k\nabla u \cdot \mathbf{n} = \bar{t} \quad \text{on } \Gamma_N, \tag{2.3}$$

where u is the unknown scalar field, k denotes a material or diffusion coefficient, f is a prescribed source term, and $\mathbf{n}$ is the outward unit normal vector on the boundary.

To derive the weak formulation, a test function v is introduced such that $v = 0$ on Γ_D. Multiplying the governing equation by v, integrating over the domain, and applying integration by parts yields the variational problem: find $u \in \mathcal{V}$ such that

$$\int_\Omega k\nabla u \cdot \nabla v \, \mathrm{d}\Omega = \int_\Omega f v \, \mathrm{d}\Omega + \int_{\Gamma_N} \bar{t} v \, \mathrm{d}\Gamma \qquad \forall v \in \mathcal{V}_0, \tag{2.4}$$

where $\mathcal{V} \subset H^1(\Omega)$ denotes the space of admissible trial functions possessing square-integrable first derivatives and satisfying the Dirichlet boundary conditions, and $\mathcal{V}_0 \subset H^1(\Omega)$ is the corresponding test function space strictly enforcing $v = 0$ on Γ_D [43].

The domain Ω is discretized into a finite number of non-overlapping elements forming a mesh $\mathcal{T}_h$. The approximate solution u_h is expressed in terms of nodal shape functions as

$$u_h(\mathbf{x}) = \sum_{i=1}^{N} N_i(\mathbf{x}) U_i, \tag{2.5}$$

where N_i are the shape functions associated with the mesh nodes and U_i are the corresponding nodal degrees of freedom. Substitution of this approximation into the weak form and application of a Galerkin procedure lead to a linear system of equations of the form

$$\mathbf{K}\mathbf{U} = \mathbf{F}, \tag{2.6}$$

where the global stiffness matrix $\mathbf{K}$ and load vector $\mathbf{F}$ are assembled from element-level contributions [32].

In the context of this thesis, the Poisson problem serves two closely related purposes. First, it provides a representative physical model for investigating the influence of geometric boundary perturbations on the numerical solution. Second, its variational structure mirrors that of the Laplace equation employed for mesh deformation, in which a similar elliptic boundary value problem is solved to propagate prescribed boundary motion smoothly into the interior of the domain. This parallel highlights the consistency of the proposed framework, in which both the physical response and the mesh motion are governed by elliptic problems defined on the same geometric domain. Although the scalar Poisson problem is adopted here for clarity, the finite element formulation extends naturally to vector-valued problems such as linear elasticity, in which the scalar unknown u would be replaced by a displacement vector field. Such extensions are not pursued in the present work but represent a natural direction for future research.

2.2 Mesh Motion and the Arbitrary Lagrangian-Eulerian Formulation

In finite element analysis, the numerical solution is defined with respect to a computational mesh that discretizes the physical domain. When the geometry of the domain changes, for example due to prescribed boundary perturbations, the question arises how the mesh should be updated in order to maintain solution accuracy and numerical stability. Traditional approaches rely on remeshing the domain for each new geometry, which can be computationally expensive and may introduce inconsistencies between successive solutions. An alternative is to treat the mesh itself as a deformable entity [12] whose motion is governed by additional kinematic relations [7, 2].

2.2.1 Remeshing Strategies in Evolving Geometries

In finite element analysis of evolving geometries, one of the most widely adopted approaches for maintaining geometric conformity and mesh quality is remeshing, i.e., the regeneration of the finite element discretization whenever the computational domain changes. Remeshing techniques are commonly employed in problems involving large deformations, moving interfaces, crack propagation, fluid-structure interaction, and shape optimization, where the original mesh may become severely distorted or no longer represent the geometry accurately [43, 13].

Modern mesh generation algorithms aim to produce discretizations that satisfy element quality criteria, such as bounds on aspect ratios, minimum angles, and element regularity. Comprehensive treatments of mesh generation and adaptive remeshing strategies are available in the literature, where mesh optimization, refinement, and local re-triangulation procedures are discussed in detail [13]. In practical applications, remeshing may be triggered by geometric updates, error indicators, or mesh quality measures that fall below prescribed thresholds.

While remeshing provides high geometric flexibility and robustness under severe deformations or topology changes, it introduces important computational implications. Each regeneration of the mesh requires reconstruction of element connectivity, redefinition of nodal coordinates, and reassembly of the global system matrices. Even when the underlying partial differential equation remains unchanged, the discrete problem must be rebuilt entirely for each new geometry. In simulations where geometry updates occur repeatedly, this process can contribute substantially to the overall computational cost.

These considerations become particularly relevant in uncertainty quantification contexts. Sampling-based approaches, such as Monte Carlo simulation, require repeated evaluations of the deterministic forward model [29]. When geometric parameters are uncertain and remeshing is performed for each realization, the cost of mesh generation and matrix assembly scales directly with the number of samples. As emphasized in the uncertainty quantification literature, the efficiency of each deterministic solve plays a decisive role in the overall feasibility of large-scale studies [29].

An additional aspect frequently encountered in remeshing-based simulations is the need to transfer solution fields between successive meshes. When the mesh topology changes, solution variables defined on the previous discretization are typically interpolated or projected onto the updated mesh in order to maintain continuity of the numerical procedure. Although such projection operators are well established, they introduce additional computational effort and may influence numerical accuracy, particularly in transient or multi-step simulations [35, 22].

For these reasons, alternative mesh update strategies have been developed, particularly within the Arbitrary Lagrangian-Eulerian (ALE) framework. Instead of regenerating the mesh topology, the computational mesh can be deformed continuously according to prescribed rules, thereby preserving connectivity while accommodating boundary motion [16, 6]. In this setting, mesh motion is treated as an additional kinematic field, whose computation is decoupled from the physical problem.

The choice between remeshing and mesh deformation therefore represents a trade-off between geometric flexibility and computational efficiency. Remeshing remains indispensable in situations involving extreme distortion or topological change. However, for moderate geometric perturbations that preserve domain topology, mesh deformation methods offer an attractive alternative by avoiding repeated topology reconstruction. The implications of this distinction for discretization consistency and solution comparison are discussed in the following subsection.

2.2.2 Lagrangian and Eulerian descriptions

Two classical descriptions of motion are commonly distinguished in continuum mechanics. In a purely Lagrangian description, the computational mesh is attached to the material, and mesh nodes move together with material points. This approach is well suited for solid mechanics problems involving moderate deformations, but it may lead to severe mesh distortion when large or repeated geometric changes occur. In contrast, an Eulerian description employs a mesh that is fixed in space, while the material flows through the computational domain. Although this avoids mesh distortion, it complicates the tracking of material interfaces and boundary conditions. [7]

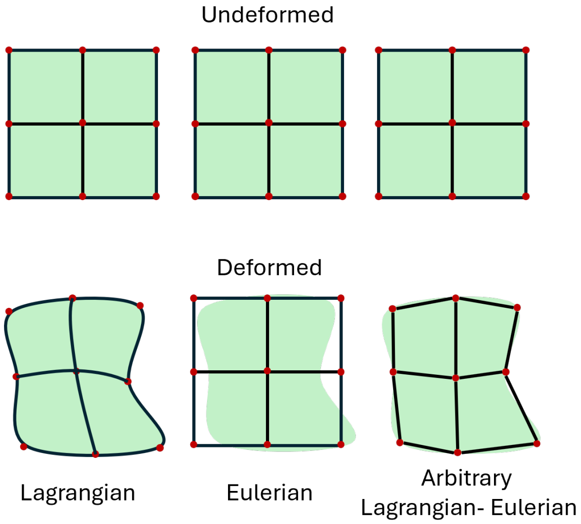


Figure 2.1: Conceptual comparison of Lagrangian, Eulerian, and Arbitrary Lagrangian-Eulerian (ALE) descriptions of motion. The top row shows the undeformed reference configuration with an identical computational mesh. In the Lagrangian description, the mesh follows the material deformation. In the Eulerian description, the mesh remains fixed in space while the material deforms through it. In the ALE formulation, mesh motion is decoupled from material deformation, allowing the mesh to move independently for numerical convenience while preserving mesh connectivity.

2.2.3 Arbitrary Lagrangian-Eulerian formulation

The Arbitrary Lagrangian-Eulerian (ALE) formulation combines the advantages of both descriptions by decoupling mesh motion from material deformation. Within the ALE framework, the computational mesh is allowed to move arbitrarily, independent of the physical motion of the material, provided that this motion is accounted for consistently in the governing equations [28, 7]. This introduces an additional mapping between a reference configuration and the current computational configuration, which enables controlled mesh motion without altering mesh topology.

From a kinematic standpoint, the ALE formulation distinguishes between three configurations: a reference configuration, a current physical configuration, and a computational mesh configuration. The mesh motion is prescribed or computed for numerical convenience, rather than being dictated by the underlying physics [18]. As a result, boundary motion can be propagated into the interior of the domain in a smooth and consistent manner, while preserving element connectivity and nodal numbering.

2.2.4 Relevance of ALE for geometric boundary perturbations

The ALE formulation is particularly well suited for problems involving geometric boundary perturbations [28, 6, 39]. Instead of regenerating a new mesh for each realization of a perturbed geometry, the ALE framework allows the existing mesh to be smoothly deformed according to the imposed boundary displacements, an approach that has been successfully applied in the context of random domains without remeshing [41]. This approach avoids repeated remeshing and ensures that the finite element discretization remains consistent across different realizations, which is essential for meaningful comparison of numerical solutions in uncertainty quantification studies.

In the context of this thesis, ALE provides the theoretical foundation that legitimizes mesh deformation as a numerical tool for handling boundary uncertainty. The specific manner in which mesh motion is computed within this framework is addressed in the following section, where Laplace-based mesh deformation is introduced as a practical and robust realization of ALE mesh motion.

2.3 Laplace-Based Mesh Deformation

Within the Arbitrary Lagrangian-Eulerian (ALE) framework, mesh motion is not dictated by the physical deformation of the material but is instead prescribed to maintain numerical accuracy and mesh quality. A widely used approach for defining such mesh motion is based on solving an elliptic partial differential equation for the mesh displacement field. Among elliptic formulations, Laplace-based mesh deformation represents a simple and robust strategy that has been successfully applied in a variety of moving-boundary and deforming-domain problems [6, 28, 39]. A variety of mesh smoothing and mesh motion techniques have been proposed in the literature to improve mesh quality or to accommodate boundary motion. Classical approaches include Laplacian smoothing, in which each interior node is relocated based on the average position of its neighbors, as well as enhanced variants designed to reduce mesh shrinkage effects [9]. More advanced strategies formulate mesh smoothing as an optimization problem targeting specific element quality measures, or employ centroidal Voronoi tessellations to achieve quasi-uniform meshes [8, 1]. Elasticity-based or spring-analogy mesh motion techniques, in which the mesh is treated as a pseudo-elastic solid or a network of interconnected springs, have also been widely used to control mesh distortion under large deformations [3, 36].

In the present work, Laplace-based mesh deformation is adopted due to its simplicity, robustness, and low computational cost. Unlike optimization-based or elasticity-based approaches, it requires no additional material parameters and leads to a symmetric, positive-definite system that can be assembled once and reused efficiently across multiple realizations [6, 40]. These properties make Laplace-based deformation particularly attractive for uncertainty quantification problems, where a large number of geometry updates must be performed.

In Laplace-based mesh deformation, the displacement of interior mesh nodes is obtained by solving a Laplace equation for the mesh displacement field $\mathbf{u}_m$,

$$\nabla^2 \mathbf{u}_m = \mathbf{0} \quad \text{in } \Omega, \tag{2.7}$$

$$\mathbf{u}_m = \bar{\mathbf{u}}_{pert} \quad \text{on } \Gamma_{pert}, \tag{2.8}$$

$$\mathbf{u}_m = \mathbf{0} \quad \text{on } \Gamma_{fixed}, \tag{2.9}$$

where Ω denotes the computational domain.

In practice, the prescribed boundary displacement field $\bar{\mathbf{u}}_{pert}$ must be explicitly mapped from the abstract geometric uncertainty parameters $\boldsymbol{\Xi}$. If the uncertain boundary is parameterized by a coordinate function $\mathbf{x}(\xi, \boldsymbol{\Xi})$, where ξ is a local surface coordinate, the required Dirichlet boundary condition for the mesh deformation is defined as the spatial difference between the perturbed and nominal configurations:

$$\bar{\mathbf{u}}_{pert}(\xi) = \mathbf{x}(\xi, \boldsymbol{\Xi}_{perturbed}) - \mathbf{x}(\xi, \boldsymbol{\Xi}_{nominal}). \tag{2.10}$$

This explicit mapping bridges the gap between the stochastic parameter space and the spatial boundary value problem, allowing the resulting harmonic displacement field to ensure a smooth propagation of boundary motion into the interior of the domain.

From a numerical perspective, the Laplace equation acts as a smoothing operator: localized boundary perturbations are distributed gradually across the mesh, preventing excessive element distortion and preserving mesh connectivity. In contrast to remeshing-based approaches, the mesh topology remains unchanged, and nodal correspondence between different realizations is maintained. This property is particularly advantageous in uncertainty quantification, where repeated evaluations on geometrically perturbed domains are required.[6, 39]

Laplace-based mesh deformation can be interpreted as a specific realization of ALE mesh motion, in which the mesh displacement field is governed by an elliptic operator rather than by physical constitutive laws. As emphasized by Donea and Huerta [6] and Nackenhorst [28], such elliptic mesh motion techniques provide a computationally inexpensive and stable means of accommodating boundary motion while avoiding mesh degeneration. Similar concepts have also been employed in the context of moving interfaces and time-dependent domains, as discussed by Tezduyar [39].

More recently, Zheng et al. [41] extended Laplace-based mesh deformation techniques to the treatment of geometric uncertainty by mapping random boundary configurations to a fixed reference domain. In this approach, the Laplace equation is used to define a smooth geometric transformation between different realizations, thereby eliminating the need for repeated remeshing and enabling consistent finite element discretizations across samples. This strategy demonstrates the suitability of Laplace-based deformation for uncertainty propagation on random domains and provides a key methodological foundation for the present work.

In this thesis, Laplace-based mesh deformation is employed as a geometric transformation tool to handle boundary uncertainty within finite element simulations. The method enables the systematic comparison of responses obtained on perturbed geometries while retaining a fixed mesh topology, thereby isolating the effects of geometric uncertainty from numerical artifacts associated with remeshing.

2.3.1 Geometric Boundary Uncertainty and Reference Domains

In many practical engineering applications, the exact geometry of a structure is not known with certainty. Manufacturing tolerances, measurement errors, wear, or imperfect boundary descriptions introduce deviations from the nominal geometry that can significantly influence the numerical response predicted by finite element simulations. When such geometric uncertainty is present, the computational domain itself becomes a random quantity, and classical finite element formulations based on a single fixed geometry are no longer sufficient.

A common approach for treating geometric uncertainty is to generate a new mesh for each realization of the perturbed boundary. While conceptually straightforward, this strategy introduces several practical difficulties: it leads to high computational cost due to repeated mesh generation, complicates the comparison of solutions across realizations, and may introduce additional numerical errors associated with mesh-to-mesh interpolation. These drawbacks become particularly pronounced in uncertainty quantification settings, where a large number of realizations must be evaluated.[39, 41]

An alternative and more efficient strategy is to introduce a fixed reference domain and to describe geometrically perturbed configurations through smooth geometric transformations. Within this framework, the physical domain corresponding to each realization is obtained by mapping the reference configuration according to prescribed boundary perturbations.[6, 28] By maintaining a consistent mesh topology on the reference domain, nodal correspondence between different realizations is preserved, and numerical results can be compared directly.

Laplace-based mesh deformation provides a natural mechanism for constructing such geometric transformations [6, 39]. Prescribed boundary perturbations are imposed on the reference domain, and the resulting Laplace equation governs the smooth propagation of these perturbations into the interior. The resulting mapping is continuous, harmonic, and computationally inexpensive to evaluate, making it well suited for repeated simulations on geometrically uncertain domains.

In the context of uncertainty quantification, this reference-domain approach allows geometric variability to be incorporated without altering the underlying finite element discretization. As demonstrated in recent work on stochastic finite element analysis on random domains, such mappings enable consistent and efficient propagation of geometric uncertainty while avoiding the numerical overhead associated with remeshing. [41] This concept forms the basis for the numerical methodology developed in the subsequent chapter, where boundary uncertainty is modeled through prescribed perturbations and propagated using Laplace-based mesh deformation.

2.3.2 Mesh Deformation versus Remeshing: Discretization Consistency

When geometric boundary perturbations are present, the numerical treatment of the computational domain plays a crucial role in the accuracy, consistency, and efficiency of finite element simulations. Figure 2.2 illustrates the fundamental distinction between remeshing-based approaches and mesh deformation techniques based on a fixed reference discretization.

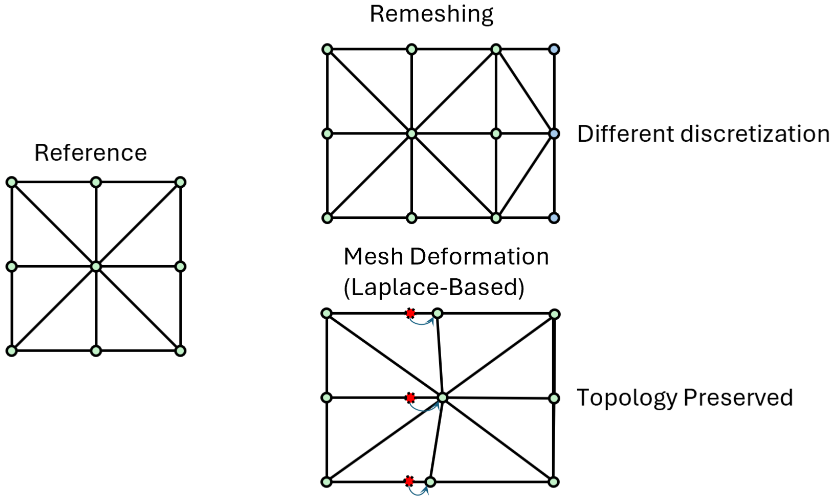


Figure 2.2: Conceptual comparison between remeshing-based and mesh-deformation-based treatments of geometric boundary perturbations. Starting from a reference mesh, remeshing generates a new discretization for the perturbed geometry, resulting in different mesh connectivity and loss of nodal correspondence. In contrast, Laplace-based mesh deformation incorporates prescribed boundary displacements through a smooth geometric mapping while preserving mesh topology and nodal connectivity. This distinction enables consistent comparison of numerical solutions across different geometric realizations.

In remeshing-based approaches, a new finite element mesh is generated for each realization of the perturbed geometry. While this strategy offers great flexibility in handling large or complex geometric changes, it also implies that each realization is associated with a distinct discretization. As a consequence, the finite-dimensional approximation spaces differ between realizations, and nodal correspondence is generally lost. Comparisons of numerical solutions across different meshes therefore involve an inherent discretization-induced variability, which may obscure the effects of geometric perturbations themselves [43, 13].

In contrast, mesh deformation techniques operate on a fixed reference mesh whose topology and connectivity are preserved. Geometric perturbations are incorporated through a continuous mapping of nodal coordinates, while the finite element connectivity remains unchanged. As a result, all realizations share the same discrete approximation space, and nodal correspondence is maintained across samples. This property enables a consistent comparison of numerical solutions obtained for different geometric configurations, which is particularly advantageous in uncertainty quantification contexts [6, 28].

From a numerical standpoint, the two strategies therefore differ not only in computational cost but also in the structure of the resulting discrete problem. Remeshing alters both the geometry and the discretization, whereas mesh deformation modifies only the geometric mapping while keeping the discrete operator structure intact [6]. The latter allows the influence of geometric boundary perturbations to be isolated more clearly from discretization effects, provided that mesh quality remains acceptable throughout the deformation process.

It is important to emphasize that neither strategy can be regarded as a universal reference. Remeshing is often preferred in cases involving severe deformations or topological changes, while mesh deformation methods are well suited for moderate geometric variations that preserve domain topology [40]. In the present work, both approaches are considered as legitimate numerical treatments of geometric boundary perturbations. Their comparison is therefore not intended to establish an exact reference solution, but rather to quantify the trade-off between computational efficiency and discretization consistency when handling geometric uncertainty in finite element simulations.

2.4 Uncertainty Quantification

2.4.1 Role of Uncertainty in Finite Element Analysis

Uncertainty quantification (UQ) aims at assessing the influence of uncertain model parameters on the response of computational models [38]. In finite element analysis, uncertainties may arise from various sources, including material properties, loading conditions, boundary conditions, and geometric descriptions. A large body of research has focused on material parameter uncertainty and stochastic loading, for which probabilistic modeling techniques are well established.

In contrast, geometric boundary uncertainty has received comparatively less systematic attention, despite its significant influence on numerical predictions [23, 15]. When geometric parameters are uncertain, the computational domain itself becomes variable, and classical finite element formulations based on a single fixed geometry are no longer sufficient.

The Laplace-based mesh deformation framework introduced in the previous sections provides a consistent mechanism to incorporate geometric variability while preserving mesh topology and discretization structure. Within this setting, geometric uncertainty can be propagated through the finite element model using different uncertainty descriptions.

Two principal paradigms are considered in the present work. In a probabilistic setting, uncertain geometric parameters are modeled as random variables and statistical characteristics of the structural response are estimated. In a non-probabilistic interval setting, uncertain parameters are described by bounded sets, and the objective becomes the determination of extremal response values within the admissible parameter domain.

2.4.2 Probabilistic Uncertainty Propagation

In the probabilistic framework, uncertain model parameters are represented as random variables or random vectors defined on a probability space $(\Theta, \mathcal{F}, \mathbb{P})$. Let Ξ denote a random input vector with joint probability density function f_Ξ. In the context of geometric uncertainty, this vector governs the amplitude and spatial distribution of the boundary perturbations. For a deterministic computational model $Q = g(\Xi)$, the model response becomes a random variable. It is important to emphasize that evaluating $g(\Xi)$ is not a simple functional evaluation; it encapsulates the entire finite element procedure, including the geometric update of the domain, the assembly of the system matrices, and the solution of the governing partial differential equations.The objective of probabilistic uncertainty quantification is the estimation of statistical quantities, primarily the mathematical expectation

$$\mathbb{E}[Q] = \int_{\mathbb{R}^p} g(\boldsymbol{\xi}) f_\Xi(\boldsymbol{\xi}), d\boldsymbol{\xi}, \tag{2.11}$$

and the variance, which quantifies the dispersion of the response around the mean:

$$\mathrm{Var}[Q] = \mathbb{E}\big[(Q - \mathbb{E}[Q])^2\big]. \tag{2.12}$$

Because analytical integration of these high-dimensional integrals is impossible for complex finite element models, sampling-based methods such as Monte Carlo (MC) simulation are commonly employed. For N independent geometric realizations, the sample mean estimator is defined as

$$\hat{\mu}_Q = \frac{1}{N} \sum_{i=1}^{N} Q(\Xi_i), \tag{2.13}$$

and the corresponding unbiased sample variance estimator is given by

$$\hat{\sigma}_Q^2 = \frac{1}{N-1} \sum_{i=1}^{N} \big(Q(\Xi_i) - \hat{\mu}_Q\big)^2. \tag{2.14}$$

According to the law of large numbers and the central limit theorem, the sample mean $\hat{\mu}_Q$ converges to the true expectation $\mathbb{E}[Q]$ with a rate of $\mathcal{O}(N^{-1/2})$ [38, 34]. While Monte Carlo simulation is highly robust and its convergence rate is independent of the stochastic dimension p, this slow $\mathcal{O}(N^{-1/2})$ convergence imposes a severe computational burden. To reduce the estimation error by half, the number of samples must be quadrupled.

Consequently, a reliable estimation of statistical moments requires thousands of deterministic evaluations.This requirement for massive sampling highlights the critical bottleneck of traditional stochastic finite element approaches. If a standard remeshing strategy were employed, the $\mathcal{O}(N_{dof})$ topological mesh generation algorithm would have to be executed from scratch thousands of times. By explicitly defining the probabilistic sampling loop over a fixed reference topology using Laplace-based mesh deformation, the framework proposed in this thesis significantly reduces the computational overhead associated with repeated mesh generation, rendering robust Monte Carlo evaluation more tractable for repeated finite element evaluations under geometric variability.

2.4.3 Interval Uncertainty Propagation

In situations where probabilistic information is unavailable, insufficient, or when strict worst-case scenarios must be evaluated, uncertain parameters can be described by interval variables. An interval vector is defined as

$$\boldsymbol{\Xi} = (\Xi_1, \dots, \Xi_p)^\top \in \mathbb{R}^p, \tag{2.15}$$

where each component Ξ_i is bounded by prescribed lower and upper limits,

$$\Xi_i \in [\underline{\Xi}_i, \overline{\Xi}_i], \qquad i = 1, \dots, p. \tag{2.16}$$

The admissible parameter domain therefore forms a hyper-rectangular set

$$\mathcal{I} = \left\{ \boldsymbol{\Xi} \in \mathbb{R}^p \;\middle|\; \underline{\Xi}_i \leq \Xi_i \leq \overline{\Xi}_i,\ i = 1, \dots, p \right\}. \tag{2.17}$$

In the context of this thesis, the interval vector $\boldsymbol{\Xi}$ possesses a direct physical interpretation: the bounds $[\underline{\Xi}_i, \overline{\Xi}_i]$ represent strict manufacturing tolerances, maximum admissible wear limits, or known bounds on assembly imperfections. The hyper-rectangle $\mathcal{I}$ therefore defines an envelope of all physically possible geometric configurations of the structural boundary.

For a deterministic computational model response $Q = g(\boldsymbol{\Xi})$, the resulting output is a bounded set. The lower and upper response bounds are defined by the global optimization problems

$$Q_{\text{min}} = \min_{\boldsymbol{\Xi} \in \mathcal{I}} g(\boldsymbol{\Xi}), \qquad Q_{\text{max}} = \max_{\boldsymbol{\Xi} \in \mathcal{I}} g(\boldsymbol{\Xi}). \tag{2.18}$$

This formulation constitutes the standard interval uncertainty propagation problem in non-probabilistic analysis [30, 5].

However, solving this optimization problem in the context of high-fidelity finite element analysis presents severe computational challenges. The objective function $g(\boldsymbol{\Xi})$ is not a simple analytical expression; it is a complex, implicitly defined black-box model [11, 20]. Evaluating $g(\boldsymbol{\Xi})$ for a single candidate geometry requires sequentially solving the Laplace-based mesh deformation problem to compute the updated nodal coordinates, assembling the global stiffness matrix for the perturbed domain, solving the physical boundary value problem, and finally extracting the quantity of interest (e.g., maximum field value or a regional average).

Furthermore, the mapping from the geometric boundary parameters to the structural response is generally highly non-linear. Standard gradient-based optimization algorithms (such as Sequential Quadratic Programming) are completely impractical here. The analytical gradient $\nabla g(\boldsymbol{\Xi})$ is not readily available because of the complex chain rule linking the finite element solution, the Laplace mesh mapping, and the boundary parameters. Approximating these gradients via finite differences would require $p + 1$ expensive finite element evaluations per optimization step [11], effectively defeating the purpose of an efficient framework.

Consequently, identifying the strict interval bounds Q_{min} and Q_{max} necessitates the use of gradient-free, surrogate-assisted global optimization strategies[33, 20, 5]. These methods must be capable of navigating the computationally expensive parameter space $\mathcal{I}$ with a minimal number of exact deterministic finite element evaluations.

2.4.4 Bayesian Global Optimization for Interval Bound Identification

As established in Section 2.4.3, the computation of interval response bounds reduces to the global optimization problems for Q_{min} and Q_{max} over the admissible parameter domain $\mathcal{I}$. Since each evaluation of the model response $g(\boldsymbol{\Xi})$ involves the solution of a mesh deformation problem followed by a finite element boundary value problem, g is computationally expensive to evaluate and gradient information is not readily available.

Classical global optimization strategies based on dense sampling or deterministic search can therefore become prohibitively expensive when applied directly to g. To alleviate this difficulty, surrogate-based optimization techniques have been developed, in which the expensive objective function is approximated by a computationally inexpensive statistical model that is iteratively refined [11].

Bayesian global optimization (BGO), also referred to as Efficient Global Optimization (EGO), constitutes a prominent class of such surrogate-based methods [20]. In this framework, the unknown function $g(\Xi)$ is modeled as a realization of a stochastic process. A probabilistic surrogate model is constructed based on previously evaluated samples, and an acquisition function is used to determine new candidate points that balance exploration of uncertain regions and exploitation of promising regions.

In contrast to purely deterministic surrogate modeling approaches, Bayesian optimization explicitly quantifies the uncertainty of the surrogate prediction. This uncertainty information is subsequently exploited to guide the search for global extrema in a principled manner. The probabilistic surrogate model most commonly employed in BGO is Gaussian process regression, which provides closed-form expressions for predictive means and variances [33].

Recently, BGO methods have been extended to interval uncertainty propagation problems, where the objective is the identification of both lower and upper bounds of a model response over a prescribed parameter domain. In particular, Dang et al. [5] proposed a parallel Bayesian global optimization (PBGO) framework that exploits Gaussian process surrogates and acquisition strategies tailored to bound estimation. While their original framework allows multiple candidate points to be evaluated simultaneously, the present work adapts this methodology into a sequential, Non-Parallel Bayesian Global Optimization (N-PBGO) scheme. This sequential adaptation preserves the efficient bounding properties of the acquisition functions while remaining suitable for the standard computational resources and the non-intrusive finite element solver employed in this study.

Gaussian Process Regression

Bayesian global optimization relies on a probabilistic surrogate model to approximate the expensive objective function $g(\Xi)$. In the present work, Gaussian process (GP) regression is employed for this purpose. Gaussian processes provide a flexible non-parametric framework for modeling unknown functions and are widely used in the context of surrogate modeling and Bayesian optimization [33].

A Gaussian process is defined as a collection of random variables, any finite number of which follow a joint Gaussian distribution. Formally, the unknown function $g(\Xi)$ is modeled as

$$g(\Xi) \sim \mathcal{GP}\big(m(\Xi),\, k(\Xi, \Xi')\big), \tag{2.19}$$

where $m(\Xi)$ denotes the mean function and $k(\Xi, \Xi')$ is the covariance kernel. The mean function represents prior knowledge about the global trend of the function, while the kernel encodes assumptions about smoothness and correlation structure.

In many practical applications, including the present work, a constant mean function is adopted,

$$m(\Xi) = \beta, \tag{2.20}$$

with β estimated from the data. To model smooth response surfaces, the squared exponential kernel (also known as radial basis function kernel) is commonly used [33]. In its automatic relevance determination (ARD) form, it is defined as

$$k(\Xi, \Xi') = \sigma_f^2 \exp\left(-\sum_{i=1}^{p} \frac{(\Xi_i - \Xi_i')^2}{2\ell_i^2} \right), \tag{2.21}$$

where σ_f^2 denotes the signal variance and ℓ_i are characteristic length scales associated with each input dimension. The ARD formulation allows different sensitivity scales for different uncertain parameters.

Assume that n evaluations of the expensive model have been performed, resulting in the dataset

$$\mathcal{D}_n = \{(\Xi^{(i)}, y^{(i)})\}_{i=1}^{n}, \qquad y^{(i)} = g(\Xi^{(i)}). \tag{2.22}$$

Let $\mathbf{X} = [\Xi^{(1)}, \ldots, \Xi^{(n)}]^\top$ denote the design matrix and $\mathbf{y} = [y^{(1)}, \ldots, y^{(n)}]^\top$ the vector of observations. Under the Gaussian process prior, the joint distribution of the observed data and the function value at a new point Ξ remains Gaussian. Conditioning on the observations yields closed-form expressions for the posterior predictive mean and variance [33]:

$$m_n(\Xi) = m(\Xi) + \mathbf{k}(\Xi, \mathbf{X})\mathbf{K}^{-1}\big(\mathbf{y} - m(\mathbf{X})\big), \tag{2.23}$$

$$\sigma_n^2(\Xi) = k(\Xi, \Xi) - \mathbf{k}(\Xi, \mathbf{X})\mathbf{K}^{-1}\mathbf{k}(\mathbf{X}, \Xi), \tag{2.24}$$

where $\mathbf{K}$ is the $n \times n$ covariance matrix with entries $K_{ij} = k(\Xi^{(i)}, \Xi^{(j)})$, $\mathbf{k}(\Xi, \mathbf{X})$ denotes the $1 \times n$ vector of covariances between the new candidate point and the training inputs, and $m(\mathbf{X})$ is the $n \times 1$ vector containing the mean function evaluated at all training points (i.e., $m(\mathbf{X}) = \beta\mathbf{1}$).

In practice, a small observational noise variance σ^2 is often added to the diagonal of $\mathbf{K}$ to ensure numerical stability and positive definiteness, leading to the regularized system $\mathbf{K} + \sigma^2\mathbf{I}$. In the limit of vanishing noise, the Gaussian process interpolates the observed data exactly, and the predictive variance vanishes at sampled points.

The predictive mean $m_n(\Xi)$ serves as a smooth approximation of the expensive function $g(\Xi)$, while the predictive variance $\sigma_n^2(\Xi)$ quantifies the epistemic uncertainty of the surrogate model. This uncertainty measure plays a central role in guiding the sequential sampling strategy of Bayesian global optimization, as described in the following subsection.

Expected Improvement Criterion

Once a Gaussian process surrogate model has been constructed, a strategy is required to determine where the expensive objective function $g(\Xi)$ should be evaluated next. In Bayesian global optimization, this decision is guided by an acquisition function, which quantifies the expected utility of sampling at a candidate point. Among various acquisition functions, the Expected Improvement (EI) criterion introduced by Jones et al. [20] is one of the most widely used.

Consider the minimization problem

$$g_{\text{min}} = \min_{i=1,\dots,n} g(\Xi^{(i)}), \tag{2.25}$$

where g_{min} denotes the best (smallest) observed value after n evaluations. At a new candidate point Ξ, the surrogate model predicts a normally distributed response,

$$g(\Xi) \sim \mathcal{N}\big(m_n(\Xi), \sigma_n^2(\Xi)\big). \tag{2.26}$$

The improvement random variable is defined as

$$I(\Xi) = \max\big(g_{\text{min}} - g(\Xi),\, 0\big), \tag{2.27}$$

which measures the potential reduction in the objective value relative to the current best observation. The Expected Improvement is obtained by taking the expectation of $I(\Xi)$ with respect to the predictive distribution:

$$\text{EI}(\Xi) = \mathbb{E}\big[I(\Xi)\big]. \tag{2.28}$$

Because the predictive distribution is Gaussian, this expectation admits a closed-form expression [20, 11]:

$$\text{EI}(\Xi) = \big(g_{\text{min}} - m_n(\Xi)\big)\Phi(z) + \sigma_n(\Xi)\phi(z), \tag{2.29}$$

where

$$z = \frac{g_{\text{min}} - m_n(\Xi)}{\sigma_n(\Xi)}, \tag{2.30}$$

and $\Phi(\cdot)$ and $\phi(\cdot)$ denote the cumulative distribution function and probability density function of the standard normal distribution, respectively.

The Expected Improvement criterion naturally balances two competing objectives. The first term, $(g_{\text{min}} - m_n)\Phi(z)$, promotes exploitation by favoring regions where the predictive mean is small. The second term, $\sigma_n\phi(z)$, encourages exploration by assigning value to regions with high predictive uncertainty. As a result, EI automatically trades off local refinement near promising candidates against global exploration of poorly sampled regions.

In the absence of observational noise, the predictive variance $\sigma_n^2(\Xi)$ vanishes at previously sampled points, implying that $\text{EI}(\Xi) = 0$ there. Consequently, the method avoids redundant evaluations and sequentially refines the surrogate model in unexplored regions of the admissible parameter domain.

Sequential Identification of Response Bounds

In the context of interval uncertainty propagation, the objective is the determination of the global minimum and maximum of the model response over the admissible parameter domain $\mathcal{I}$. Using the surrogate-based framework described above, this task can be formulated as a sequence of Bayesian optimization problems.

To identify the lower bound, the Expected Improvement criterion for minimization is maximized over the admissible domain:

$$\Xi_{n+1} = \arg\max_{\Xi \in \mathcal{I}} \mathrm{EI}(\Xi). \tag{2.31}$$

The expensive function $g(\Xi)$ is then evaluated at Ξ_{n+1}, the dataset is augmented, and the Gaussian process surrogate is updated accordingly. This sequential procedure is repeated until a suitable convergence criterion is met.

The identification of the upper response bound can be performed analogously. Since

$$\max_{\Xi \in \mathcal{I}} g(\Xi) = -\min_{\Xi \in \mathcal{I}} \big(-g(\Xi) \big), \tag{2.32}$$

the maximization problem can be reduced to a minimization problem by applying the same Bayesian optimization procedure to the negated objective function.

In practice, the lower and upper bounds are determined through two distinct optimization phases. As established earlier, while the framework proposed by Dang et al. [5] was originally designed to select multiple candidate points for simultaneous evaluation (PBGO), the present implementation restricts the acquisition to a single candidate per iteration, resulting in a Non-Parallel Bayesian Global Optimization (N-PBGO) scheme.

The choice of a non-parallel approach is deliberate. The primary objective of incorporating this optimizer is to demonstrate that the proposed Laplace-based mesh deformation framework can be successfully and non-intrusively coupled with advanced surrogate-assisted global optimization strategies. A sequential approach provides the necessary algorithmic simplicity and transparency to validate this coupling without introducing the software and hardware complexities associated with parallel computing management. Furthermore, for the 2D linear boundary value problems considered in this study, the computational cost of a single finite element evaluation is sufficiently low that parallelization is not strictly required to achieve a tractable interval analysis.

Ultimately, this N-PBGO scheme provides a highly efficient strategy for identifying extremal responses. By explicitly accounting for predictive uncertainty, it requires significantly fewer model evaluations than standard Interval Monte Carlo or grid-based search methods, fulfilling the objective of accelerating interval uncertainty propagation while maintaining a clear and robust implementation.

Chapter 3

Numerical Methodology

This chapter establishes the numerical framework developed for the propagation of geometric boundary uncertainty in finite element analysis. The proposed methodology leverages an Arbitrary Lagrangian-Eulerian (ALE) formulation defined on a fixed reference domain. By decoupling the geometric description from the physical solver, the approach allows for the efficient analysis of multiple geometric realizations without the need for repeated remeshing.

Geometric uncertainty is modeled as a stochastic perturbation of the domain boundary, which is propagated into the interior via a Laplace-based mesh deformation strategy. This ensures that mesh topology and nodal connectivity remain invariant across all realizations, preserving the algebraic structure of the finite element system.

The methodology is fully non-intrusive with respect to the deterministic finite element solver. For each uncertainty realization, only the nodal coordinates of the mesh are modified, while the underlying discretization, degrees of freedom, and solution procedure remain unchanged. This property is essential for uncertainty quantification studies, as it enables consistent comparison of numerical results across samples without repeated remeshing.

The present chapter focuses on the abstract formulation of the numerical framework. Problem-specific governing equations, discretization choices, and implementation details are deferred to the subsequent chapter on numerical implementation and test cases.

3.1 Reference Domain and Problem Setting

In the presence of geometric boundary uncertainty, the computational domain of a finite element model is no longer fixed but varies across uncertainty realizations. Let $\Omega(\Xi) \subset \mathbb{R}^d$ denote a family of physical domains parametrized by a set of uncertain variables Ξ, which describe deviations of the boundary from a nominal configuration. A direct finite element analysis on $\Omega(\Xi)$ would require the generation of a new mesh for each realization, leading to high computational cost and loss of mesh consistency.

To avoid repeated remeshing, this work adopts the fixed reference domain strategy utilized by Zheng et al. [41]. A fixed reference domain Ω_0 is introduced, and all geometrically perturbed configurations are obtained via a smooth geometric transformation

$$\mathbf{x} = \mathbf{\Phi}(\mathbf{X}, \Xi), \tag{3.1}$$

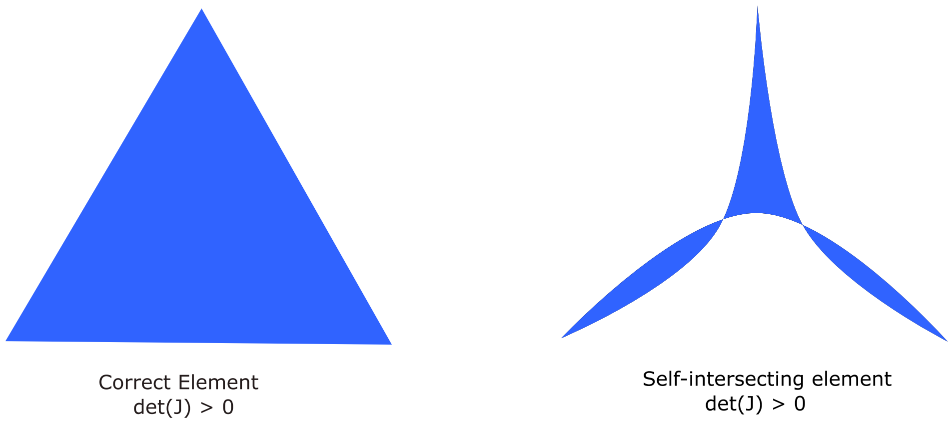
following the standard ALE framework [10], where $\mathbf{X} \in \Omega_0$ denotes a point in the reference domain and $\mathbf{x} \in \Omega(\Xi)$ is the corresponding point in the perturbed physical domain.

For the finite element formulation to remain well-defined on the deformed configuration, the geometric mapping must be admissible. In particular, the element-wise mapping from the reference element to the physical element must be injective, so that no folding or self-intersection of elements occurs. In finite element mesh validity theory, this requirement is commonly expressed by demanding that the Jacobian determinant of the mapping remains strictly positive everywhere in the domain:

$$J_{\mathbf{\Phi}}(\mathbf{X}) = \det\left(\nabla_{\mathbf{X}}\mathbf{\Phi}\right) = \det\left(\mathbf{I} + \nabla_{\mathbf{X}}\mathbf{u}_m\right) > 0, \quad \forall \mathbf{X} \in \Omega_0. \tag{3.2}$$

Violation of this condition ($J_{\mathbf{\Phi}} \leq 0$) corresponds to numerical element inversion, rendering the resulting system of equations singular or physically meaningless [19].

For linear triangular elements, the mapping is affine and the Jacobian determinant is constant within each element. In this case, the determinant is proportional to the signed area of the triangle, so that positivity of the signed area is equivalent to an orientation-preserving and non-degenerate element mapping. Accordingly,



Correct Element
det(J) > 0

Self-intersecting element
det(J) > 0

Figure 3.1: Valid triangular finite element with positive Jacobian determinant (left) and an inverted element with negative Jacobian determinant (right).

mesh validity in the present work is monitored by verifying that the minimum element Jacobian (equivalently, the signed area for T_3 elements) remains strictly positive across the deformed mesh. The use of Jacobian-based criteria for assessing element validity and quality is standard in mesh optimization and untangling literature [21].

Within this framework, the finite element mesh is generated once on the reference domain and reused for all realizations. Geometric uncertainty is incorporated exclusively through the mapping $\mathbf{\Phi}$, which updates nodal coordinates while preserving mesh connectivity and element topology. As a result, all uncertainty realizations share an identical discretization structure, which enables direct comparison of numerical solutions and facilitates non-intrusive uncertainty propagation.

The reference domain approach adopted in this thesis is particularly well suited for boundary uncertainty problems, where geometric variability is localized on selected parts of the boundary. By prescribing boundary displacements on the reference domain and computing a smooth interior deformation, geometrically perturbed domains can be analyzed efficiently without modifying the underlying finite element formulation.

3.2 Modeling of Geometric Boundary Uncertainty

In the proposed framework, geometric uncertainty is introduced exclusively through perturbations of the boundary of the reference domain. This choice reflects the fact that, in many engineering applications, uncertainties arise primarily from imprecise boundary descriptions[25], manufacturing tolerances[31], or unknown interface conditions.

Let the boundary of the reference domain be decomposed as

$$\partial\Omega_0 = \Gamma_{\text{pert}} \cup \Gamma_{\text{fixed}}, \tag{3.3}$$

where Γ_{pert} denotes the portion of the boundary subject to geometric uncertainty, and Γ_{fixed} represents boundary segments that remain unchanged across all realizations. Geometric uncertainty is modeled by prescribing displacement fields on Γ_{pert}, while homogeneous conditions are imposed on Γ_{fixed}.

For a given uncertainty realization characterized by the random vector $\mathbf{\Xi}$, the perturbed boundary configuration is defined through a boundary displacement field

$$\mathbf{u}_m(\mathbf{X}, \mathbf{\Xi}) = \mathbf{g}(\mathbf{X}, \mathbf{\Xi}) \qquad \text{on } \Gamma_{\text{pert}}, \tag{3.4}$$

where $\mathbf{g}$ specifies the magnitude and direction of the boundary perturbation. The function $\mathbf{g}$ may be defined through probabilistic, interval-based, or parametric uncertainty models, depending on the nature of the uncertainty being considered. On the fixed part of the boundary, the mesh displacement is constrained to vanish,

$$\mathbf{u}_m(\mathbf{X}, \mathbf{\Xi}) = \mathbf{0} \qquad \text{on } \Gamma_{\text{fixed}}. \tag{3.5}$$

In the computational implementation, this boundary perturbation is applied directly as discrete nodal displacements on the mesh boundary. Rather than defining an abstract continuous field, the methodology allows for direct coordinate manipulation, such as applying a uniform random translation vector to a subset of nodes (e.g., an outer edge) or utilizing a node-by-node parametric mapping (e.g., deforming a nominal circular boundary into an ellipse).

Because the T_3 finite element shape functions inherently interpolate nodal values, prescribing these discrete, "manual" boundary displacements automatically defines a mathematically admissible, piecewise-continuous Dirichlet boundary condition for the Laplace PDE. This direct nodal manipulation provides maximum flexibility, cleanly separating the stochastic parameterization of the geometry from the internal numerical treatment of the mesh deformation.

The present methodology does not impose a specific form on the boundary perturbation model. Instead, it provides a general mechanism for incorporating geometric uncertainty through prescribed boundary displacements on a reference domain. This abstraction allows different uncertainty descriptions to be treated within a unified computational framework. Simple synthetic perturbations may be employed for verification purposes, while more structured models-such as interval-based bounds or low-dimensional parametric representations-can be introduced to study physically meaningful uncertainty scenarios.

The only restriction is that the magnitude of $\mathbf{g}$ must remain sufficiently small to ensure that the resulting mesh deformation, governed by the Laplace operator described in the next section, preserves the topological validity of the discretization (i.e., Eq. 3.2 holds).

By localizing uncertainty at the boundary and expressing it through prescribed displacements, the methodology cleanly separates the modeling of geometric uncertainty from the numerical treatment of mesh deformation and the solution of the physical boundary value problem. This separation is essential for the non-intrusive nature of the framework and forms the basis for the Laplace-based mesh deformation strategy introduced in the following section.

3.3 Laplace-Based Mesh Deformation

Once geometric boundary perturbations have been prescribed on the reference domain, the corresponding deformation of the interior of the mesh must be determined. Within the proposed framework, this task is accomplished by computing a mesh displacement field that smoothly propagates the imposed boundary motion into the interior of the domain while preserving mesh connectivity and element topology.

The mesh is treated as a kinematic entity whose motion is governed by an auxiliary partial differential equation. Among the various mesh motion strategies available in the literature, a Laplace-based formulation is adopted in this thesis due to its simplicity, robustness, and low computational cost. By solving a Laplace equation for the mesh displacement field, a harmonic extension of the boundary perturbations into the interior of the domain is obtained.

Let $\mathbf{u}_m : \Omega_0 \to \mathbb{R}^d$ denote the mesh displacement field. The strong form of the mesh deformation problem is:

$$\nabla^2 \mathbf{u}_m = \mathbf{0} \quad \text{in } \Omega_0, \tag{3.6}$$

subject to the Dirichlet boundary conditions defined in Section 3.2.

3.3.1 Weak Formulation

To solve (3.6) using the Finite Element Method, we introduce the standard Sobolev space $\mathcal{V} = [H^1(\Omega_0)]^d$. Let $\mathcal{V}_0$ be the subspace of test functions vanishing on the boundary $\partial\Omega_0$. The weak formulation reads: Find $\mathbf{u}_m \in \mathcal{V}$ such that $\mathbf{u}_m|_{\partial\Omega_0}$ satisfies Eq. (3.4) and:

$$\int_{\Omega_0} \nabla \mathbf{u}_m : \nabla \mathbf{v} \, d\Omega_0 = 0, \quad \forall \mathbf{v} \in \mathcal{V}_0. \tag{3.7}$$

In a two-dimensional setting ($d = 2$), this vector-valued problem decouples into two independent scalar Laplace problems for the components $u_{m,x}$ and $u_{m,y}$. Since both components share the same differential operator and mesh, they result in identical stiffness matrices, offering a computational advantage during assembly.

3.3.2 Mesh Update and Quality

The solution of the weak problem provides the nodal displacement field $\mathbf{u}_m(\mathbf{X}, \Xi)$. The spatial coordinates of the mesh nodes are updated via the ALE mapping:

$$\mathbf{x}^* = \mathbf{X} + \mathbf{u}_m(\mathbf{X}, \Xi). \tag{3.8}$$

Since the connectivity of the mesh (the element topology) remains invariant, the updated mesh $\mathcal{T}^*$ preserves the algebraic structure of the reference discretization.

While the Laplace operator does not explicitly maximize element quality metrics (unlike optimization-based smoothing [21]), it effectively distributes localized boundary distortions into the bulk of the domain. For the magnitude of perturbations considered in this thesis, this smoothing is sufficient to maintain a positive Jacobian ($J > 0$) without the need for complex iterative untangling algorithms.

3.4 Non-Intrusive Coupling with the Physical Solver

A central objective of this thesis is to propagate geometric uncertainty without modifying the core formulation or implementation of the physical solver. To achieve this, we adopt a sequential, non-intrusive coupling strategy where the mesh deformation acts as a pre-processing step for the physical analysis. This procedure decouples the geometric update from the physical assembly.

For a specific realization of the uncertain parameters Ξ, the solution procedure involves two sequential steps performed on the reference discretization:

Step 1: Geometry Update The mesh deformation problem defined in Section 3.3 is solved first, yielding the updated nodal coordinates $\mathbf{x}^*$ as defined in Section 3.3. This step is purely geometric; it does not require information from the physical problem (e.g., stress or temperature fields).

Step 2: Physical Solution The physical boundary value problem (e.g., the Poisson equation) is then solved on the deformed configuration $\Omega(\Xi)$. The element connectivity matrix remains identical to that of the reference mesh. The geometric changes are reflected solely in the updated Jacobian matrices of the isoparametric mapping and the numerical integration weights (element areas) used during the assembly of the stiffness matrix $\mathbf{K}(\mathbf{x}^*)$ and load vector $\mathbf{F}(\mathbf{x}^*)$.

The linear system for the physical variables $\mathbf{U}$ is given by:

$$\mathbf{K}(\mathbf{x}^*)\mathbf{U} = \mathbf{F}(\mathbf{x}^*). \tag{3.9}$$

By strictly separating Step 1 from Step 2, the deterministic solver functions as a "black box," necessitating only access to the updated nodal coordinate array. This one-way coupling assumes that the physical response does not influence the mesh deformation, which is consistent with the standard ALE treatment for fixed-topology problems [6, 10].

3.4.1 Complexity and Sparsity Preservation

The primary theoretical advantage of the sequential coupling described above lies in the preservation of the algebraic graph of the finite element system. In standard remeshing algorithms, the topological connectivity of the mesh changes for every realization Ξ. Consequently, the non-zero sparsity pattern of the global stiffness matrix $\mathbf{K}$ must be recomputed, and memory reallocation must be performed continuously. This topological regeneration introduces additional overhead inside a stochastic sampling loop, which is further compounded when the mesh generator is invoked as an external process.

By utilizing the ALE reference domain mapping, the global node numbering and element connectivity remain strictly invariant. The sparsity pattern of $\mathbf{K}(\mathbf{x}^*)$ is established exactly once during the pre-processing stage. During the uncertainty propagation loop, only the numerical values of the non-zero matrix entries are updated via the deformed element Jacobians. This transforms the geometric update from repeated topological mesh generation into an algebraic spatial update on a fixed connectivity, substantially reducing the per-sample overhead in Monte Carlo and global optimization workflows.

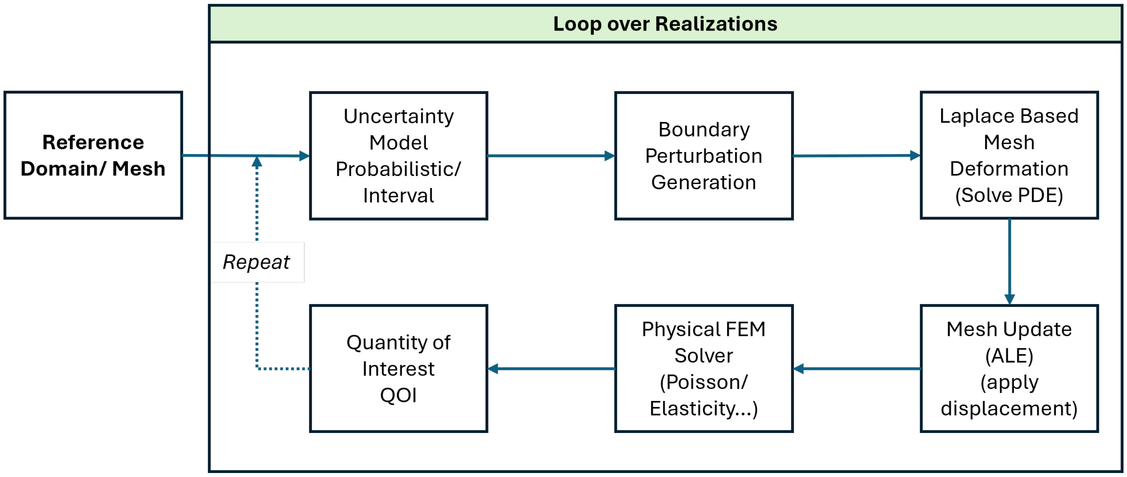


Figure 3.2: Algorithmic workflow of the proposed non-intrusive framework. Starting from a fixed reference domain, boundary perturbations are generated according to the chosen uncertainty model, propagated into the interior via Laplace-based mesh deformation, and followed by the solution of the physical finite element problem.

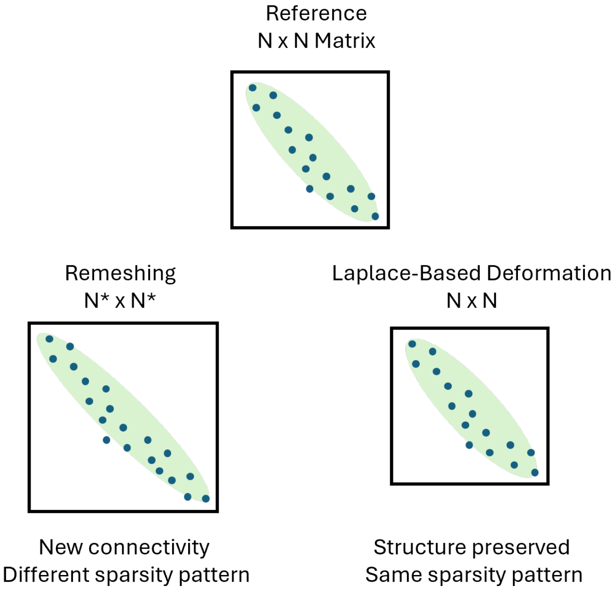


Figure 3.3: Conceptual comparison of the global stiffness matrix sparsity patterns. Standard remeshing (left) alters the topological graph, resulting in an $N^* \times N^*$ matrix with a completely different sparsity pattern for every geometric realization. In contrast, the proposed Laplace-based ALE mesh deformation (right) preserves the exact finite element connectivity, maintaining a constant $N \times N$ matrix structure throughout the uncertainty quantification loop.

3.5 Definition of the Computational Model

To facilitate the uncertainty propagation analysis, it is necessary to abstract the coupled mesh-physics procedure described in the previous sections into a deterministic input-output mapping. Let $\boldsymbol{\Xi} \in \mathcal{D}_{\boldsymbol{\Xi}} \subset \mathbb{R}^p$ denote the vector of uncertain geometric parameters. We define the computational model $\mathcal{M}$ as a mapping that transforms a realization $\boldsymbol{\xi}$ into a scalar Quantity of Interest (QoI), denoted by Q:

$$\begin{aligned} \mathcal{M} : \mathcal{D}_{\boldsymbol{\Xi}} &\to \mathbb{R} \\ \boldsymbol{\xi} &\mapsto Q(\boldsymbol{\xi}). \end{aligned} \tag{3.10}$$

For a fixed realization $\boldsymbol{\xi}$, the evaluation of $Q(\boldsymbol{\xi}) = \mathcal{M}(\boldsymbol{\xi})$ involves the execution of the full non-intrusive workflow defined in Sections 3.3 and 3.4. This process is summarized as follows:

1. **Input:** Receive parameter vector $\boldsymbol{\xi}$.
2. **Boundary Perturbation:** Compute boundary displacement $\mathbf{g}(\mathbf{X}, \boldsymbol{\xi})$ (Eqs. 3.4 and 3.5).
3. **Mesh Deformation:** Solve the Laplace system (Eq. 3.7) to obtain $\mathbf{u}_m$.
4. **Validity Check:** Verify $\min(2A_e) > 0$. If invalid, the realization is rejected.
5. **Physical Solve:** Assemble and solve the physical system on the deformed mesh to obtain the field $u_h(\boldsymbol{\xi})$.
6. **Output:** Compute the QoI via a functional $\mathcal{Q}(u_h)$, such as the maximum field value or a regional integral.

By defining this abstract mapping $\mathcal{M}$, the uncertainty quantification problem is reduced to analyzing the propagation of the input uncertainty in $\boldsymbol{\Xi}$ through the black-box function $Q(\boldsymbol{\Xi})$. This abstraction allows for the independent selection of sampling or optimization strategies, as detailed in the next section.

3.6 Uncertainty Propagation Strategies

Based on the abstract model defined in Eq. (3.10), this thesis investigates two distinct uncertainty quantification problems, depending on the characterization of the input parameters $\boldsymbol{\Xi}$.

3.6.1 Probabilistic Quantification (Monte Carlo)

When $\boldsymbol{\Xi}$ is modeled as a random vector with a known probability density function, the objective is to estimate the statistical moments of the response. We employ the standard Monte Carlo (MC) method [34] due to its robustness and non-intrusive nature. The expectation of the QoI is estimated by the sample mean:

$$\mathbb{E}[Q] \approx \hat{\mu}_Q = \frac{1}{N} \sum_{i=1}^{N} Q(\boldsymbol{\Xi}_i), \tag{3.11}$$

where $\{\boldsymbol{\Xi}_i\}_{i=1}^N$ are independent samples drawn from the input distribution. This method serves as the baseline for assessing the physical trends in the results.

3.6.2 Interval Quantification (Bayesian Optimization)

When $\boldsymbol{\Xi}$ is defined only by interval bounds $\boldsymbol{\Xi} \in [\mathbf{L}, \mathbf{U}]$, the objective is to determine the range of the model response:

$$Q_{\min} = \min_{\boldsymbol{\Xi}} Q(\boldsymbol{\Xi}), \quad Q_{\max} = \max_{\boldsymbol{\Xi}} Q(\boldsymbol{\Xi}). \tag{3.12}$$

This constitutes a global optimization problem. Since the evaluation of $Q(\boldsymbol{\Xi})$ involves a computationally expensive Finite Element analysis, standard sampling methods are inefficient.

To address this, we adopt the **Non-Parallel Bayesian Global Optimization (NPBGO)** strategy developed by Dang et al. [5]. As detailed in the theoretical background (see Section 2.4.4), this method constructs a Gaussian Process (GP) surrogate model of the response surface and utilizes an expected improvement acquisition function to intelligently select samples that refine the estimates of the extrema. By evaluating candidates sequentially, this approach significantly reduces the computational cost compared to random sampling.

Chapter Summary and Outlook

The methodology presented in this chapter provides a modular, dimension-independent framework for handling geometric boundary uncertainty. By strictly separating the mesh deformation from the physical solver, the approach ensures non-intrusiveness and reproducibility.

In the following chapter (Chapter 4), this framework is instantiated for a sequence of benchmark problems. The governing equations for the Poisson problem, the specific finite element discretization (T3 elements), and the detailed implementation of the random sampling and optimization algorithms are presented to validate the proposed approach.

Chapter 4

Numerical Implementation

This chapter details the discrete finite element formulation and algorithmic implementation of the non-intrusive uncertainty quantification framework. The methodology is implemented in MATLAB, utilizing a modular architecture that separates the reference domain setup, the ALE mesh deformation, and the physical solver.

To bridge the gap between the continuous formulations presented in Chapter 3 and the computational code, all governing equations are explicitly derived at the discrete level. The primary numerical implementation is carried out in two spatial dimensions using linear triangular finite elements (T_3).

4.1 Software Architecture and Modularity

To ensure reproducibility and extensibility, the numerical framework is implemented in MATLAB using a modular, function-based folder architecture. The codebase is separated into distinct operational environments to decouple the deterministic finite element operations from the stochastic sampling algorithms.

The architecture consists of the following core modules:

- **`config/`**: Contains initialization scripts (`cfg_2D_probabilistic.m`, `cfg_2D_interval.m`) that define the geometry, boundary perturbation bounds, mesh selection, and algorithmic hyperparameters.
- **`setup/`**: Handles the pre-processing phase, including parsing the Gmsh [14] mesh files, identifying physical boundaries, and generating the nominal reference domain.
- **`FEM/`**: The deterministic core engine. This module contains highly vectorized routines for assembling the Laplace operator (`assembleLaplaceKm_T3.m`), computing the physical Poisson response, and enforcing boundary conditions via the lifting method.
- **`pbgo/` & `General/`**: Houses the high-level uncertainty quantification wrappers. These scripts execute the Monte Carlo loops and manage the surrogate training and acquisition phases of the N-PBGO optimizer.
- **`post/`**: Contains dedicated functions for post-processing the QoIs, verifying mesh validity via the element Jacobians, and generating visualizations.

By isolating the `FEM` module from the `General` evaluation loops, the deterministic solver remains entirely agnostic to whether it is being called by a probabilistic Monte Carlo simulation or an interval-based optimization routine.

4.2 Discrete Mesh Deformation

Let the reference domain $\Omega_0 \subset \mathbb{R}^2$ be discretized by a conforming finite element mesh $\mathcal{T}_h$ consisting of N_e elements and N_n nodes. The reference nodal coordinates are collected in the vector $\mathbf{X} \in \mathbb{R}^{2N_n}$.

The boundary of the reference mesh is partitioned into sets of nodes: Γ_{pert} (subject to geometric uncertainty) and Γ_{fixed} (spatially fixed). The objective of the discrete mesh deformation is to compute the nodal displacement vector $\mathbf{U}_m \in \mathbb{R}^{2N_n}$ that smoothly propagates the boundary perturbations into the interior.

Because the Laplace operator is linear and the vector components $(u_{m,x}, u_{m,y})$ are decoupled in the absence of cross-directional body forces, the mesh deformation problem reduces to two identical scalar systems:

$$\mathbf{K}_m \mathbf{U}_{m,x} = \mathbf{0}, \quad \mathbf{K}_m \mathbf{U}_{m,y} = \mathbf{0}. \tag{4.1}$$

In the implementation, these two systems are assembled into a single block-diagonal system $\text{blkdiag}(\mathbf{K}_m, \mathbf{K}_m)\, \mathbf{U}_m = \mathbf{0}$, where $\mathbf{U}_m = [\mathbf{U}_{m,x};\, \mathbf{U}_{m,y}] \in \mathbb{R}^{2N_n}$, and solved in a single linear solve, subject to the nodal Dirichlet conditions:

$$\mathbf{U}_m(\mathbf{X}_i) = \begin{cases} \mathbf{g}(\mathbf{X}_i, \Xi) & \forall i \in \Gamma_{\text{pert}}, \\ \mathbf{0} & \forall i \in \Gamma_{\text{fixed}}. \end{cases} \tag{4.2}$$

The global mesh stiffness matrix $\mathbf{K}_m \in \mathbb{R}^{N_n \times N_n}$ depends exclusively on the reference geometry Ω_0. It is assembled only once during the initialization phase, significantly reducing the computational overhead of the uncertainty propagation loop.

4.2.1 Linear Triangular Element (T_3) Formulation

The global matrix $\mathbf{K}_m$ is assembled from the element-level stiffness matrices $\mathbf{k}_e^m$. For a linear triangular element defined by three nodes $(X_1, Y_1), (X_2, Y_2), (X_3, Y_3)$, the coordinate mapping is affine. The area of the reference element is computed via the determinant:

$$2A_0 = \det \begin{bmatrix} 1 & 1 & 1 \\ X_1 & X_2 & X_3 \\ Y_1 & Y_2 & Y_3 \end{bmatrix}. \tag{4.3}$$

The linear shape functions are $N_i(\mathbf{X}) = \frac{1}{2A_0}(a_i + b_i X + c_i Y)$, where the geometric constants are defined cyclically (e.g., $b_1 = Y_2 - Y_3$, $c_1 = X_3 - X_2$).

Because the gradients of the shape functions are constant within the T_3 element ($\nabla N_i = \frac{1}{2A_0}[b_i, c_i]^T$), the integration of the weak form over the element area is exact. The element stiffness matrix for the Laplace smoothing operator simplifies to the algebraic form [43] utilized directly in the computational implementation:

$$\mathbf{k}_e^m = \frac{1}{4A_0}\left(\mathbf{b}\mathbf{b}^T + \mathbf{c}\mathbf{c}^T\right), \tag{4.4}$$

where $\mathbf{b} = [b_1, b_2, b_3]^T$ and $\mathbf{c} = [c_1, c_2, c_3]^T$.

4.2.2 Strong Enforcement of Boundary Conditions (Lifting Method)

To evaluate the mesh displacement for a specific uncertainty realization Ξ, the prescribed boundary perturbations must be strongly enforced upon the block-diagonal system. In the MATLAB implementation, this is achieved via the algebraic lifting method (matrix partitioning) [43] to preserve matrix symmetry, which is critical for the efficiency of sparse linear solvers.

Let the global degrees of freedom be partitioned into free displacements (subscript f) and prescribed Dirichlet displacements (subscript c). The global Laplace system is reordered and partitioned as:

$$\begin{bmatrix} \mathbf{K}_{ff} & \mathbf{K}_{fc} \\ \mathbf{K}_{cf} & \mathbf{K}_{cc} \end{bmatrix} \begin{bmatrix} \mathbf{U}_f \\ \mathbf{U}_c(\Xi) \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{F}_c \end{bmatrix}. \tag{4.5}$$

The prescribed nodal displacements $\mathbf{U}_c(\Xi)$ are populated by evaluating the boundary perturbation models on Γ_{pert}. The unknown interior mesh displacements $\mathbf{U}_f$ are then computed by shifting the known quantities to the right-hand side, yielding the reduced symmetric system:

$$\mathbf{K}_{ff}\mathbf{U}_f = -\mathbf{K}_{fc}\mathbf{U}_c(\Xi). \tag{4.6}$$

Solving this reduced system efficiently yields the interior node updates without altering the topology of the reference mesh. In the MATLAB implementation, this partitioning is realized through an equivalent row/column elimination procedure that modifies the full-size system in place, zeroing out constrained rows and columns and placing unit entries on the diagonal, which preserves the system dimensions and avoids explicit index reordering.

4.3 The Discrete Physical Problem

For a given realization Ξ, the mesh deformation system (Eq. 4.1) is solved, and the nodal coordinates are updated to the deformed configuration: $\mathbf{x}^* = \mathbf{X} + \mathbf{U}_m$.

Element validity is immediately verified by computing the deformed area $A^*(\boldsymbol{\Xi})$ for all elements. If $A^* \leq 0$ for any element, the realization is rejected to prevent non-physical singularities. If valid, the physical boundary value problem (the Poisson problem) is assembled on the deformed mesh $\mathcal{T}^*$.

The discrete physical system reads:

$$\mathbf{K}_{phys}(\mathbf{x}^*)\mathbf{U}_{phys} = \mathbf{F}_{phys}(\mathbf{x}^*). \tag{4.7}$$

While the finite element topology and basis functions remain identical to the reference configuration, the numerical values adapt to the geometric uncertainty. Specifically, the element stiffness matrices and load vectors are evaluated dynamically using the updated coordinates $\mathbf{x}^*$.

For the Poisson equation $-\Delta u = f$, the element stiffness matrix $\mathbf{k}_e^{phys}$ shares the identical algebraic structure as the Laplace operator (Eq. 4.4), but evaluated using the deformed area A^* and deformed geometric constants $\mathbf{b}^*$ and $\mathbf{c}^*$:

$$\mathbf{k}_e^{phys} = \frac{1}{4A^*}\left(\mathbf{b}^*(\mathbf{b}^*)^T + \mathbf{c}^*(\mathbf{c}^*)^T\right). \tag{4.8}$$

Assuming a constant source term f, the exact integration of the element load vector over the T_3 element yields equal distribution to the three local nodes:

$$[\mathbf{f}_e^{phys}]_i = \int_{\Omega_e^*} f N_i \,\mathrm{d}\Omega = \frac{fA^*}{3}, \quad i = 1, 2, 3. \tag{4.9}$$

These local contributions are assembled into the global system, boundary conditions are applied via the lifting method described in Section 4.2.2, and the physical field $\mathbf{U}_{phys}$ is computed.

4.4 Algorithmic Workflow

The sequential, non-intrusive operations required to propagate the geometric uncertainty are summarized in Algorithm 1. By moving the assembly of $\mathbf{K}_m$ outside the main evaluation loop, the per-realization cost is reduced to one linear solve for the geometry update, followed by the reassembly and solution of the physical system on the deformed coordinates. While the physical stiffness matrix and load vector must be recomputed for each realization (since the element areas and shape function gradients depend on the deformed geometry), the assembly cost scales linearly with the number of elements and remains modest relative to the savings achieved by avoiding repeated remeshing.

Algorithm 1 Non-Intrusive Uncertainty Propagation via Laplace ALE

```
Require: Reference mesh $(\mathbf{X}, \mathcal{T}_h)$, Boundary sets $(\Gamma_{\text{pert}}, \Gamma_{\text{fixed}})$, Samples $N_s$
Ensure: Array of QoI evaluations $\mathbf{Q} = \{Q^{(1)}, \dots, Q^{(N_s)}\}$
Initialization: Assemble static reference mesh stiffness $\mathbf{K}_m$ (Eq. 4.4)
for $s = 1 \dots N_s$ do
    Geometry Update Phase:
    Sample geometric boundary parameters $\boldsymbol{\Xi}^{(s)}$
    Compute boundary displacements $\mathbf{g}(\mathbf{X}, \boldsymbol{\Xi}^{(s)})$ on $\Gamma_{\text{pert}}$
    Apply Dirichlet BCs via lifting: generate reduced system $[\mathbf{K}_{m,mod}, \mathbf{F}_{m,mod}]$
    Solve for mesh displacement: $\mathbf{K}_{m,mod}\mathbf{U}_m^{(s)} = \mathbf{F}_{m,mod}$
    Update nodal coordinates: $\mathbf{x}^* = \mathbf{X} + \mathbf{U}_m^{(s)}$
    Validity Check:
    if min$(2A_e^*) \leq 0$ then
        Mark realization as invalid; continue
    end if
    Physical Solve Phase:
    Assemble physical matrix $\mathbf{K}_{phys}(\mathbf{x}^*)$ and load $\mathbf{F}_{phys}(\mathbf{x}^*)$
    Apply physical Dirichlet BCs: generate reduced system $[\mathbf{K}_{p,mod}, \mathbf{F}_{p,mod}]$
    Solve physical boundary value problem: $\mathbf{K}_{p,mod}\mathbf{U}_{phys} = \mathbf{F}_{p,mod}$
    Extract Quantity of Interest: $Q^{(s)} = \mathcal{Q}(\mathbf{U}_{phys})$
end for
```

This modular loop serves as the deterministic evaluation function for both the probabilistic Monte Carlo framework and the interval-based Bayesian optimization strategy detailed in the following sections.

Remark on invalid mesh handling. The treatment of invalid realizations ($\min(2A_e^*) \leq 0$) differs between the two uncertainty propagation modes. In the probabilistic Monte Carlo setting, the invalid realization is skipped entirely: the corresponding QoI entry is excluded from the statistical evaluation. In the interval optimization setting, a large penalty value ($Q^{(s)} = 10^{30}$) is returned to the surrogate-based optimizer in order to discourage further exploration of parameter regions that lead to mesh inversion.

4.5 Implementation of Uncertainty Propagation Algorithms

As established in Section 4.4, the coupled mesh deformation and physical finite element solver acts as a deterministic, black-box mapping $\Xi \mapsto Q(\Xi)$. To execute the uncertainty quantification, this core evaluation function is wrapped within outer algorithmic loops depending on the nature of the uncertainty model.

4.5.1 Probabilistic Uncertainty: Monte Carlo Simulation

For probabilistic uncertainty propagation, a standard Monte Carlo (MC) loop is implemented. The continuous probability density functions of the input parameters Ξ are sampled using MATLAB's pseudo-random number generators.

For a specified number of samples N_s, the deterministic solver is called independently. The statistical moments (mean and variance) and the empirical Probability Density Function (PDF) of the quantity of interest are computed as a post-processing step on the resulting array of outputs $\mathbf{Q}$. Because each realization is strictly independent, this loop is trivially parallelizable, although sequential execution is utilized in this work to accurately measure base computational costs.

4.5.2 Interval Uncertainty: Optimizer Integration

When geometric uncertainty is characterized by bounded intervals, the determination of the physical response range becomes a gradient-free global optimization problem.

The core mathematical scripts for the Non-Parallel Bayesian Global Optimization (N-PBGO) algorithm-including the Latin Hypercube Sampling, Gaussian Process regression, and the Teaching-Learning-Based Optimization (TLBO) of the acquisition function-were developed and provided by Dr. Chao Dang [5]. Consequently, the implementation effort of this thesis regarding the interval uncertainty propagation focuses entirely on the programmatic integration: formulating the complex finite element pipeline as an admissible, deterministic objective function for this provided optimizer.

To decouple the optimization logic from the physical simulation, the entire sequence described in Algorithm 1 (mesh perturbation, Laplace smoothing, validity checking, physical assembly, and QoI extraction) is encapsulated within a single deterministic MATLAB function handle. From the perspective of the N-PBGO optimizer, this function acts as an expensive, black-box evaluator:

```
obj_fun = @(x) evaluateQoI_intervalHole(x, RefMesh, K_m, …)
```
(4.10)

where `x` represents the candidate vector of geometric parameters $\Xi \in [\mathbf{L}, \mathbf{U}]$.

The algorithmic coupling proceeds seamlessly: the provided N-PBGO framework generates a candidate vector Ξ_{next} and passes it to `obj_fun`. The finite element module executes the geometry update and physical solve, returning the exact QoI. The optimizer then updates its surrogate model and iterates. By abstracting the finite element framework into this parameterized functional handle, the methodology cleanly isolates the structural mechanics codebase from the external optimization algorithms.

Chapter Summary

This chapter detailed the transition from the continuous theoretical framework to a discrete, executable numerical implementation. By isolating the computation of the fixed Laplace mesh stiffness matrix $\mathbf{K}_m$, deriving the exact T_3 element formulation, and cleanly decoupling the geometry update from the physical FEM assembly, a highly efficient and non-intrusive evaluation engine was constructed.

With the numerical architecture established, Chapter 5 will proceed to validate this framework through a series of increasingly complex numerical experiments, ranging from a one-dimensional analytical benchmark to a two-dimensional plate subjected to internal boundary uncertainty.

Chapter 5

Numerical Examples and Results

The numerical methodology and implementation framework developed in the previous chapters are designed to be general, non-intrusive, and independent of the specific physical equations being solved. To validate this framework and demonstrate its capabilities, this chapter presents three numerical examples of increasing complexity.

The selected examples serve distinct purposes in evaluating the proposed approach:

1. **One-Dimensional Cantilever Beam (Section 5.1):** This problem serves as the fundamental validation benchmark. Because a closed-form analytical solution exists, it allows for a rigorous, mathematically exact verification of the discrete mesh deformation, the uncoupled finite element assembly, and the statistical consistency of the Monte Carlo estimators.
2. **Two-Dimensional Rectangle (Section 5.2):** This intermediate benchmark acts as a computational stress test for the Laplace-based Arbitrary Lagrangian-Eulerian (ALE) mapping. By applying synthetic, highly irregular boundary perturbations, the capability of the Laplace operator to maintain positive element Jacobians ($J > 0$) without remeshing is evaluated.
3. **Plate with an Uncertain Elliptical Hole (Section 5.3):** This final example represents a more demanding two-dimensional benchmark, adapting the configuration proposed by Zheng et al. [41]. It investigates geometric uncertainty located exclusively on an internal boundary. This problem unifies the framework by comparing probabilistic Uncertainty Quantification (Monte Carlo) with interval bound identification (Non-Parallel Bayesian Global Optimization).

5.1 One-Dimensional Validation Example: Cantilever Beam

The one-dimensional cantilever beam is selected as the first numerical example. Despite its simplicity, this benchmark provides a fully transparent setting in which all algorithmic components, mesh generation, ALE mapping, physical solving, and statistical post-processing, can be verified independently and in combination.

5.1.1 Problem Definition and Analytical Reference

The reference computational domain is defined as the interval $\Omega_0 = (0, L_{\text{ref}})$, where L_{ref} denotes the nominal beam length. The beam is modeled according to the Euler-Bernoulli beam theory with constant flexural rigidity EI.

As illustrated in Figure 5.1, the left end of the beam is clamped, while a concentrated transverse point load P is applied at the free right end. The governing equation for the transverse displacement $w(x)$ reads:

$$EI\frac{\mathrm{d}^4 w}{\mathrm{d}x^4} = 0 \quad \text{in } \Omega(\Xi), \tag{5.1}$$

subject to the clamped boundary conditions at the left node ($x = 0$), which enforce strictly zero displacement and rotation:

$$w_1 = 0, \qquad \theta_1 = 0. \tag{5.2}$$

The concentrated force P is applied as a Neumann condition at $x = L(\Xi)$.

For a deterministic beam of length L, the analytical solution for the tip displacement δ is given by:

$$w(L) = \frac{PL^3}{3EI}. \tag{5.3}$$

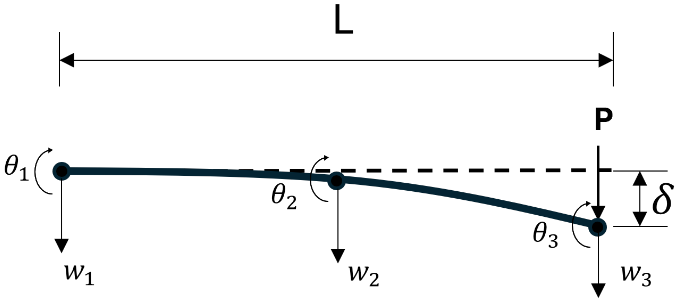


Figure 5.1: Schematic illustration of the one-dimensional cantilever beam. The beam of length L is clamped at the left end ($w_1 = \theta_1 = 0$) and subjected to a concentrated vertical load P at the free end. The tip deflection $\delta = w(L)$ is used as the quantity of interest.

This closed-form expression is used as the exact reference to validate the finite element solution for every individual realization of the random geometry.

5.1.2 Geometric Uncertainty Model

Geometric uncertainty is introduced exclusively through variations of the beam length. The uncertain length is modeled parametrically as:

$$L(\Xi) = L_{\text{ref}}(1 + \Xi), \qquad \Xi \sim \mathcal{U}[-\alpha, \alpha], \tag{5.4}$$

where α controls the amplitude of the geometric perturbation. For each realization, the free boundary at $x = L_{\text{ref}}$ is displaced according to the sampled length, while the clamped boundary remains strictly fixed.

5.1.3 Discretization: Decoupling ALE and Physical Spaces

A core challenge and distinct feature of the proposed framework is the strict mathematical decoupling between the numerical treatment of the ALE mesh motion and the physical boundary value problem.

In this example, the reference mesh is generated using Gmsh [14] (Figure 5.2a) and remains topologically fixed. The formulation relies on two completely different finite element spaces:

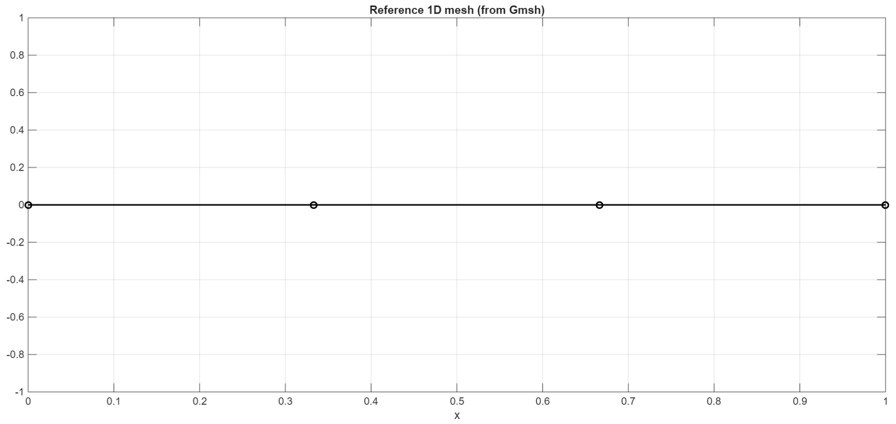
- **ALE Mesh Deformation (C^0 continuity):** The geometric mapping is governed by the Laplace equation. In one dimension, the standard finite element discretization of the Laplace operator utilizes standard linear shape functions. Consequently, the mesh deformation reduces to a simple *linear interpolation* of the prescribed boundary displacement across the interior nodes.
- **Physical Problem (C^1 continuity):** The Euler-Bernoulli beam theory requires the solution of a fourth-order differential equation. Therefore, *Cubic Hermite beam elements* are employed for the physical solver, possessing two degrees of freedom per node (transverse displacement w_i and rotation θ_i).

This distinction emphasizes that the geometric update is purely a coordinate transformation mapping, entirely independent of the higher-order basis functions required to accurately capture the physical bending mechanics.

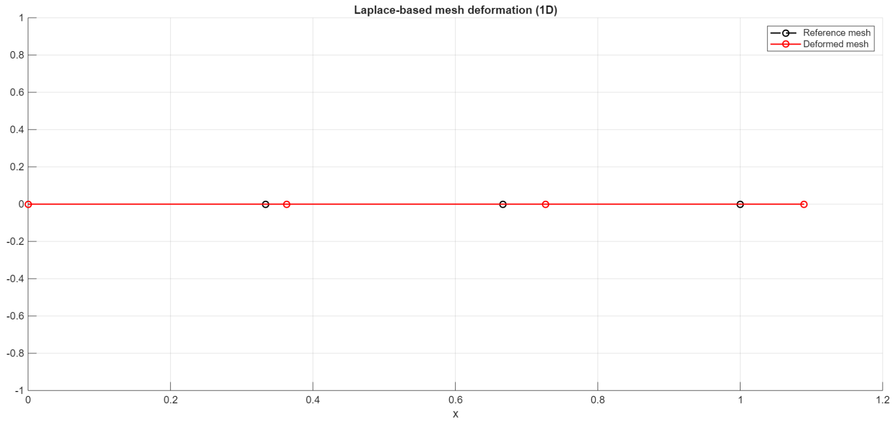
5.1.4 Results: Mesh Deformation and Response Validation

For each Monte Carlo realization, the boundary displacement is propagated to all interior nodes. Figure 5.2b illustrates a representative realization of this deformation, which visually manifests as a smooth stretching or compression of the element lengths. No loss of mesh validity (element inversion) can occur in this 1D formulation, isolating the validation of the displacement accuracy.

The primary quantity of interest is the tip deflection $w(L)$. Because the analytical solution (Eq. 5.3) scales with the cube of the length (L^3), the mapping from the uniformly distributed geometric input to the physical response is highly nonlinear. This behavior is successfully captured by the finite element model, as reflected in the skewed, non-uniform Monte Carlo distribution of the tip deflection shown in Figure 5.3.



(a) Reference one-dimensional mesh imported from `Gmsh`.



(b) Reference and Laplace-deformed 1D mesh for a representative uncertainty realization.

Figure 5.2: One-dimensional reference mesh and its Laplace-deformed counterpart. The same topology and node ordering are reused for all uncertainty realizations.

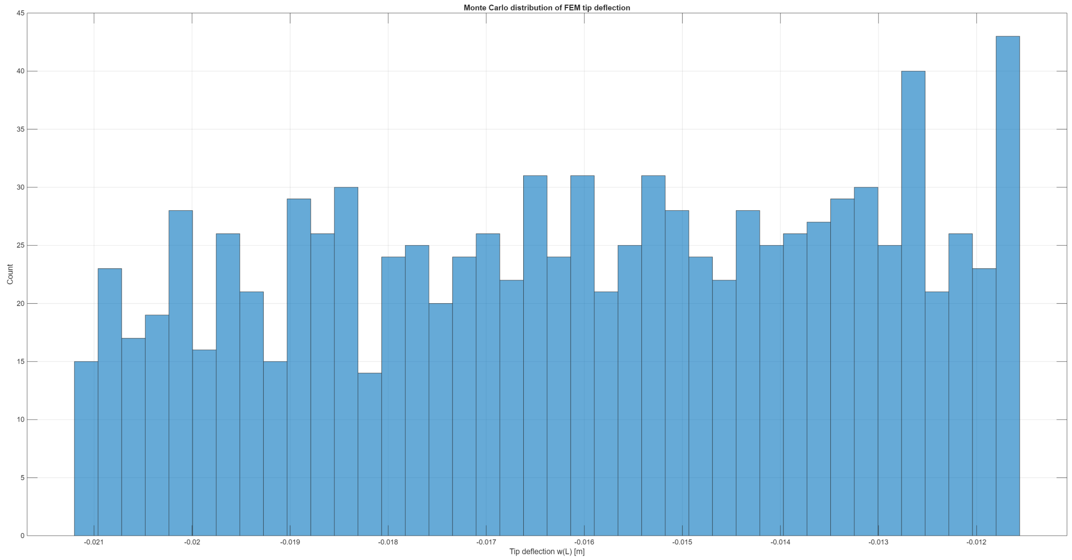


Figure 5.3: Monte Carlo distribution of the finite element tip deflection under uniformly distributed boundary uncertainty. The skewness reflects the nonlinear L^3 dependence.

A direct validation of the computational pipeline is achieved by plotting the finite element tip deflection against the analytical reference value for all N_s Monte Carlo realizations (Figure 5.4). The data points align perfectly on the identity relation.

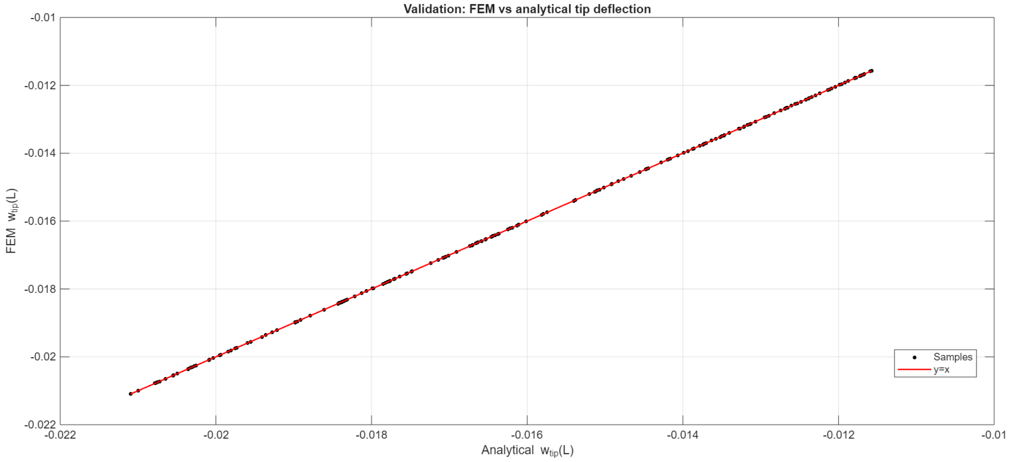


Figure 5.4: Validation plot: finite element tip deflection versus analytical reference for all Monte Carlo realizations. The solid line indicates the exact identity relation.

To quantify the numerical accuracy, the relative error between the FEM and analytical solutions is evaluated for every sample. As shown in Figure 5.5, the relative error distribution remains strictly at the level of machine precision ($\sim 10^{-15}$).

This level of accuracy is not coincidental; it is a direct mathematical consequence of the chosen discretization. The exact analytical displacement of an Euler-Bernoulli cantilever subjected to a concentrated end load is a cubic polynomial in x. Because the physical finite element space is spanned by cubic Hermite shape functions, the discrete solution perfectly captures the exact continuum solution without any spatial discretization error (ignoring floating-point round-off). Consequently, this proves that the non-intrusive coordinate updating scheme introduces zero numerical bias. Any variance observed in the output is driven entirely by the parameterized geometric boundary uncertainty, isolating the UQ pipeline from classical FEM convergence issues.

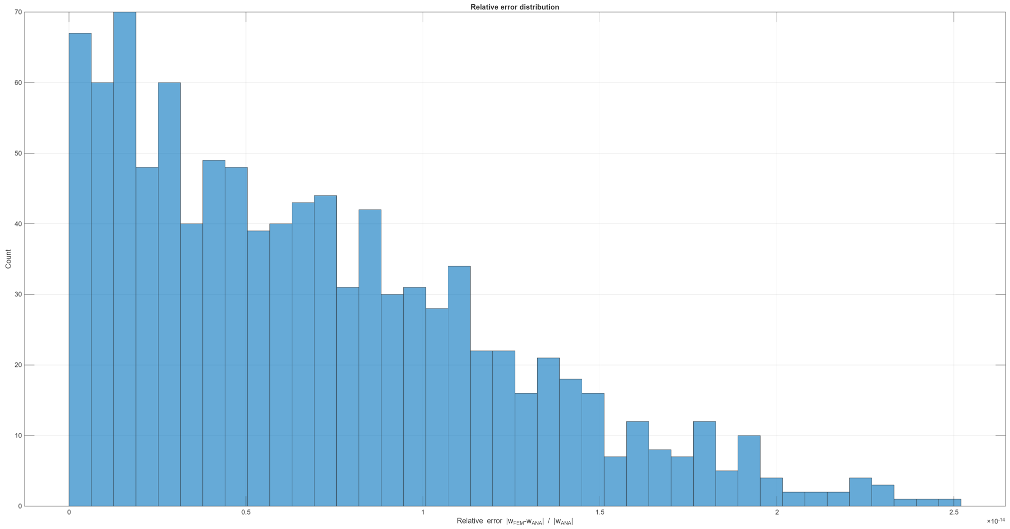


Figure 5.5: Relative error distribution between the finite element and analytical tip deflection, demonstrating accuracy down to machine precision limits.

5.1.5 Monte Carlo Convergence

The statistical consistency of the sampling-based approach is verified by monitoring the convergence of the Monte Carlo estimators. Figure 5.6 displays the running sample mean of the tip deflection as a function of the number of evaluations. As dictated by the Law of Large Numbers, the running mean rapidly stabilizes and converges to the true analytical mean value.

Similarly, Figure 5.7 shows the convergence of the running variance. While higher-order moments naturally exhibit larger fluctuations for small sample sizes, the variance estimator successfully stabilizes, confirming the robustness of the probabilistic outer loop.

Figure 5.6: Monte Carlo convergence of the running mean of the tip deflection. The horizontal line indicates the exact analytical mean.

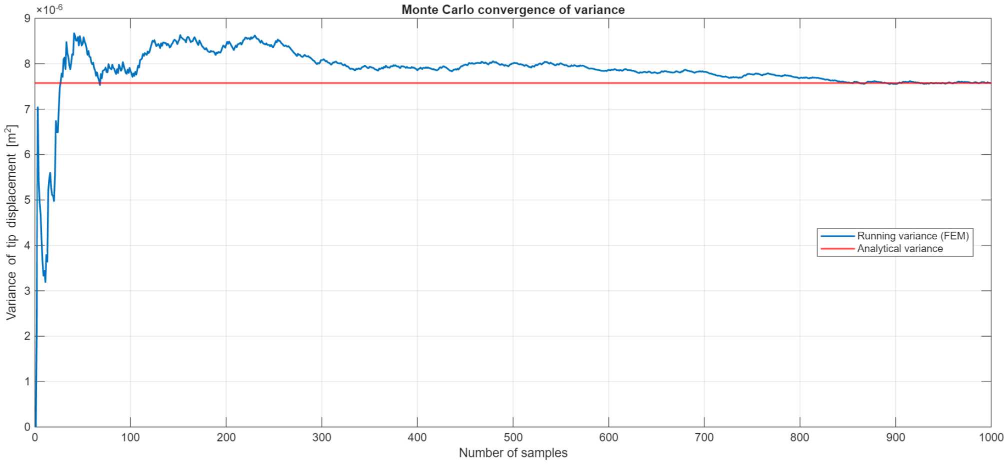


Figure 5.7: Monte Carlo convergence of the running variance of the tip deflection as the sample size increases.

Having established the absolute correctness of the non-intrusive methodology in this analytically tractable setting, the framework is subsequently applied to two-dimensional geometries, where mesh validity and interior smoothing become non-trivial constraints.

5.2 Two-Dimensional Verification Example: Rectangular Domain

This section presents the numerical results for the two-dimensional rectangular benchmark. The objective of this intermediate example is to verify the numerical behavior of the proposed non-intrusive framework in two spatial dimensions, where mesh deformation and geometric variability lead to non-trivial changes in element Jacobians and numerical integration.

Unlike the one-dimensional cantilever beam, no exact analytical reference solution is available for this configuration. Consequently, the focus of the analysis shifts to the assessment of spatial mesh deformation behavior, element validity preservation, and the statistical stability of the finite element solution under boundary-induced geometric perturbations.

5.2.1 Problem Definition and Finite Element Discretization

The reference geometric domain is defined as a rectangle of dimensions 20×10. To evaluate the physical response under geometric uncertainty, the stationary Poisson equation is chosen as the governing model. The physical boundary value problem is defined as:

$$\begin{aligned} -\nabla^2 u &= 1 \qquad \text{in } \Omega(\boldsymbol{\Xi}), \\ u &= 0 \qquad \text{on } \partial\Omega(\boldsymbol{\Xi}), \end{aligned} \tag{5.5}$$

where a constant source term $f = 1$ is applied over the domain, and homogeneous Dirichlet boundary conditions are imposed on all four outer boundaries.

The domain is discretized using an unstructured mesh of linear triangular finite elements (T_3). This reference mesh is deliberately coarse. This coarseness is a strategic choice: it serves as a strict stress test for the algorithmic pipeline, allowing elemental distortions to be tracked and visualized easily, rather than aiming for high-fidelity physical accuracy.

Unlike the one-dimensional case, which utilized different element formulations for the geometry and the physics, this two-dimensional benchmark employs the *exact same* finite element space (T_3) for both the ALE mesh deformation mapping and the physical Poisson solver. Throughout the uncertainty propagation process, the mesh topology and element connectivity remain strictly unchanged; only the nodal coordinates are updated.

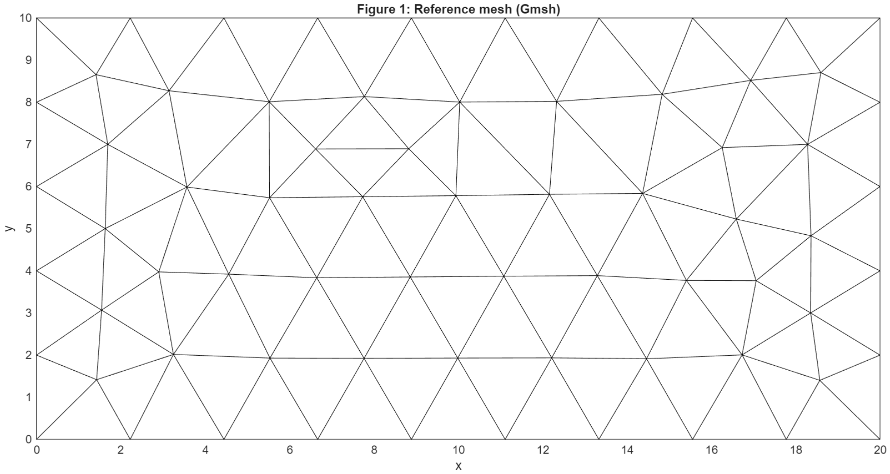


Figure 5.8: Reference finite element mesh of the 20×10 rectangular domain. Linear triangular (T_3) elements are used for both the ALE mapping and the physical solver. The right boundary is subject to geometric perturbations, while the remaining boundaries are held fixed.

5.2.2 Synthetic Boundary Uncertainty Model

Geometric uncertainty is introduced synthetically by prescribing random displacement perturbations to the nodes located exclusively on the right boundary of the domain (Γ_{pert}). For each Monte Carlo realization s, a displacement vector is assigned independently to each boundary node $i \in \Gamma_{\text{pert}}$:

$$\mathbf{g}_i^{(s)} = \begin{bmatrix} g_{x,i}^{(s)} \\ g_{y,i}^{(s)} \end{bmatrix}, \tag{5.6}$$

where the displacement components are sampled from uniform distributions:

$$g_{x,i}^{(s)} \sim \mathcal{U}(-\alpha_x L_x, \alpha_x L_x), \qquad g_{y,i}^{(s)} \sim \mathcal{U}(-\alpha_y L_y, \alpha_y L_y). \tag{5.7}$$

Here, L_x and L_y denote the domain dimensions, and α_x, α_y control the perturbation amplitude. In this benchmark, the perturbation amplitudes are set to $\alpha_x = \alpha_y = 0.02$, and $N_s = 1000$ Monte Carlo realizations are evaluated.

This uncertainty model does not represent a specific manufacturing process. Instead, it is chosen as a computational stress test. The highly irregular and spatially uncorrelated nature of these perturbations (visualized in Figure 5.9) allows the smoothing properties and robustness of the mesh deformation procedure to be rigorously examined.

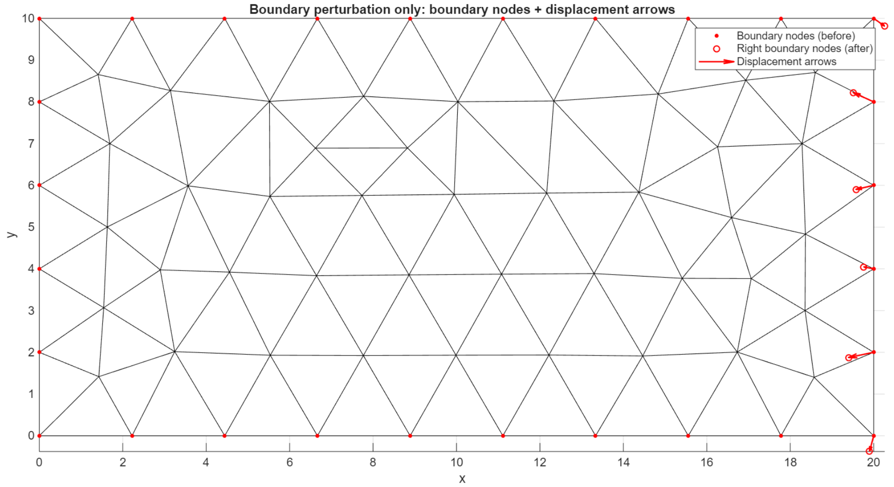


Figure 5.9: Synthetic geometric boundary perturbations applied to the right boundary nodes. The arrows indicate the highly irregular, uncorrelated direction and magnitude of the prescribed displacements prior to Laplace-based smoothing.

5.2.3 Laplace-Based Deformation and Harmonic Smoothing

In contrast to the one-dimensional case, where the Laplace-based mesh deformation reduces to a simple linear interpolation, the two-dimensional deformation field represents a genuine harmonic extension.

Mathematically, solutions to the Laplace equation ($\nabla^2 \mathbf{u}_m = \mathbf{0}$) are *harmonic functions*. A fundamental characteristic of harmonic functions is the *mean-value property*, which implies that the displacement of any interior node is essentially the average of its surrounding domain. Consequently, harmonic functions cannot exhibit local extrema in the interior. This theoretical property explains why the highly irregular, high-frequency boundary perturbations (Fig. 5.9) are naturally filtered and smoothed as they propagate into the interior, resulting in the globally coherent mesh shown in Figure 5.10.

Mesh Validity and Failure Mechanism

To ensure the mathematical validity of the finite element formulation, the oriented area of each triangular element is monitored. For a valid mesh, the oriented element area (and thus the Jacobian determinant J) must remain strictly positive. Across all standard Monte Carlo realizations considered in this benchmark, no element inversion was observed, confirming the robustness of the harmonic smoothing.

However, to characterize the limitations of the method, the boundary perturbation amplitude was deliberately exaggerated until algorithmic failure occurred. As illustrated in Figure 5.11, failure (element inversion, $J \leq 0$) is highly localized in the immediate vicinity of the perturbed boundary.

Before strict mathematical inversion occurs, the mesh undergoes severe geometric degradation. The aspect ratios of the boundary-adjacent triangles become highly skewed as nodes are pushed laterally past their neighbors. This degradation occurs when the induced displacement gradients exceed the capacity of the local

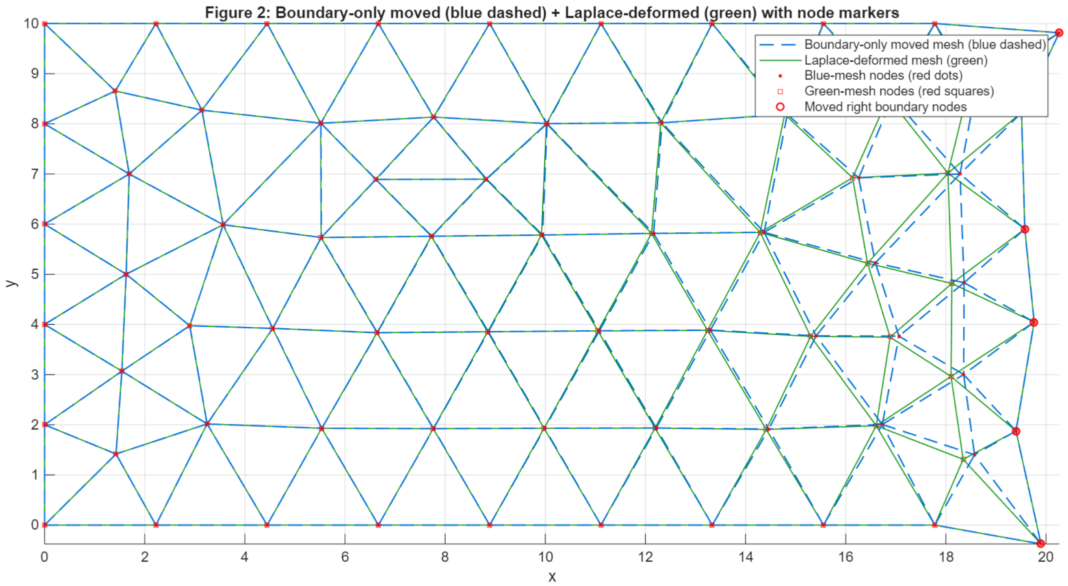


Figure 5.10: Overlay of the reference mesh and a representative Laplace-deformed mesh. The harmonic nature of the Laplace operator filters the irregular boundary noise, propagating the deformation smoothly into the interior.

mesh resolution. If higher physical accuracy were the primary objective, such highly skewed elements would introduce significant interpolation errors into the finite element basis well before actual inversion causes the matrix assembly to fail. Nevertheless, even under these extreme, artificially enlarged perturbations, the interior of the domain remains completely well-behaved, which confirms the regularizing effect of the Laplace operator.

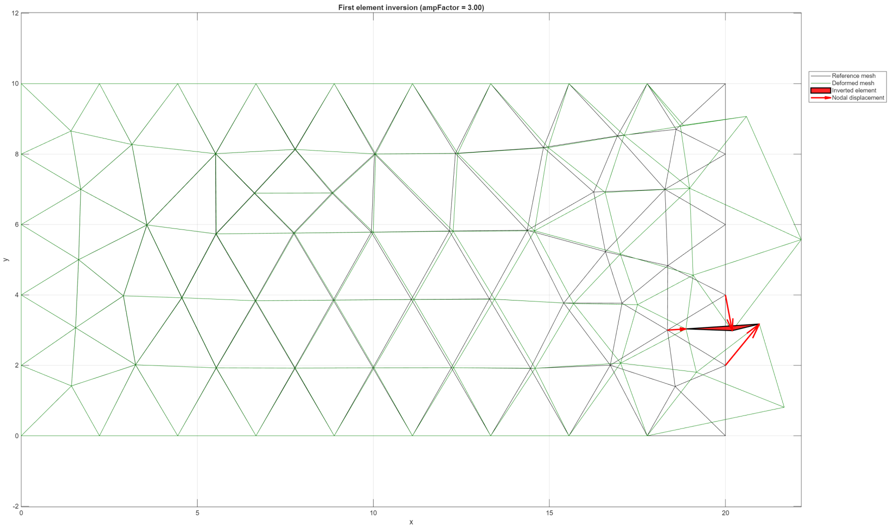


Figure 5.11: First detected element inversion under extreme, artificially enlarged boundary perturbations. The inverted element is highlighted in red (deformed) and green (reference). Failure is strictly localized at the boundary.

5.2.4 Poisson Solution on Deformed Meshes

A deterministic reference solution on the nominal (undeformed) domain is established first (Figure 5.12). To explicitly visualize the effect of the spatial discretization, the solution is presented in two formats. As shown in Figure 5.12(a), the raw solution plotted directly on the elements highlights the deliberately coarse T_3 mesh resolution. In contrast, Figure 5.12(b) displays the spatially interpolated continuous field. The physical solution exhibits a symmetric, parabolic distribution with a single interior maximum, consistent with the constant source term and homogeneous Dirichlet boundaries. Despite the coarse discretization, no numerical artifacts are present.

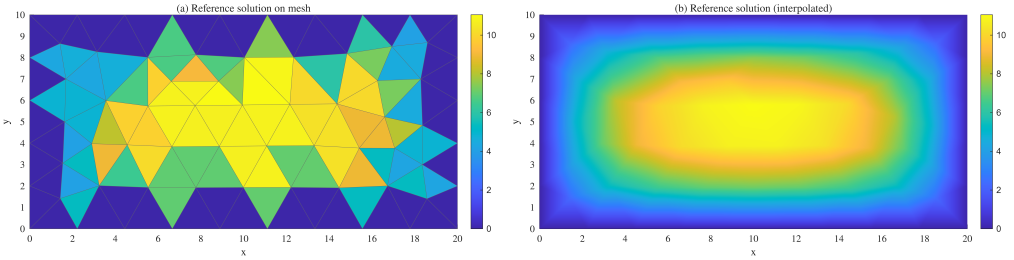


Figure 5.12: Deterministic finite element solution of the Poisson problem ($f = 1$) on the nominal, undeformed reference domain. (a) Raw discrete solution plotted directly on the T_3 mesh topology. (b) Spatially interpolated continuous field demonstrating the symmetric physical response.

The physical problem is subsequently solved on each valid deformed mesh realization. Figure 5.13 illustrates the Poisson solution on four representative deformed meshes. While the qualitative macroscopic structure of the field remains intact, local variations are clearly induced by the updated nodal coordinates entering the element Jacobians.

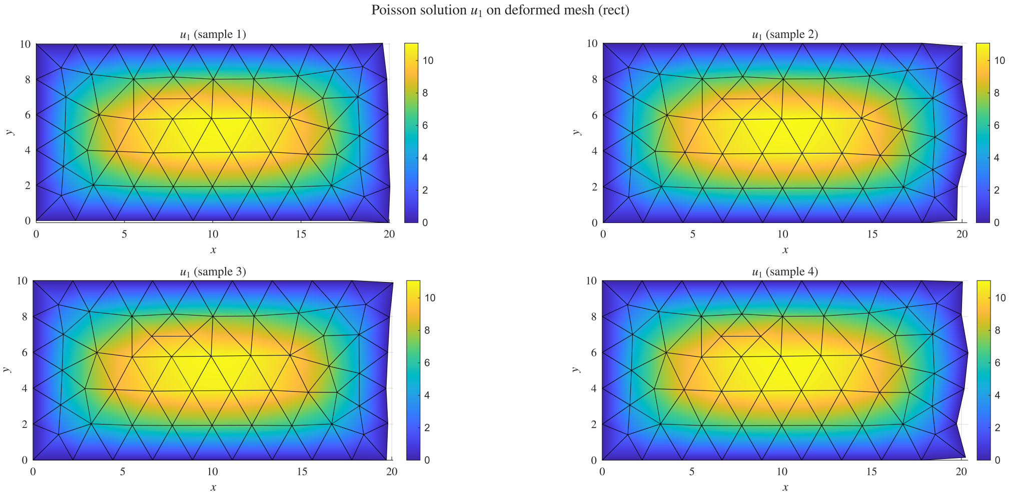


Figure 5.13: Poisson solution on four distinct deformed meshes. Geometric uncertainty enters the solution purely through the modified element Jacobians during the assembly process.

To isolate the effect of geometric uncertainty on the physical response, the difference field $\Delta U^{(s)} = U^{(s)} - U^{\text{ref}}$ is evaluated and plotted on the static reference configuration (Figure 5.14). The difference fields remain smooth and localized, confirming that the response variability is physically driven by the boundary movement rather than numerical instability.

A deeper physical interpretation of these difference fields reveals a direct correlation between the volumetric deformation of the domain and the amplitude of the elliptic PDE response. The stationary Poisson equation with a uniform positive source ($f = 1$) and homogeneous Dirichlet boundaries acts as a spatial diffusion model. The maximum amplitude relies strictly on the distance from the interior to the zero-constrained boundaries.

This behavior is explicitly visible when comparing extreme realizations. In Sample 3 (Figure 5.13), the right boundary is severely compressed inward. By moving the zero-Dirichlet boundary closer to the interior, the effective diffusion distance is reduced, suppressing the solution amplitude. This manifests as a strongly negative difference field (dark blue) in Figure 5.14. Conversely, in Sample 2, the boundary is pulled outward, expanding the local domain volume. This isolates the interior further from the boundary constraints, allowing the solution to accumulate to higher values, which yields a strongly positive difference field (yellow/orange). This predictable, physics-driven response confirms the correctness of the Laplace ALE coordinate-updating scheme.

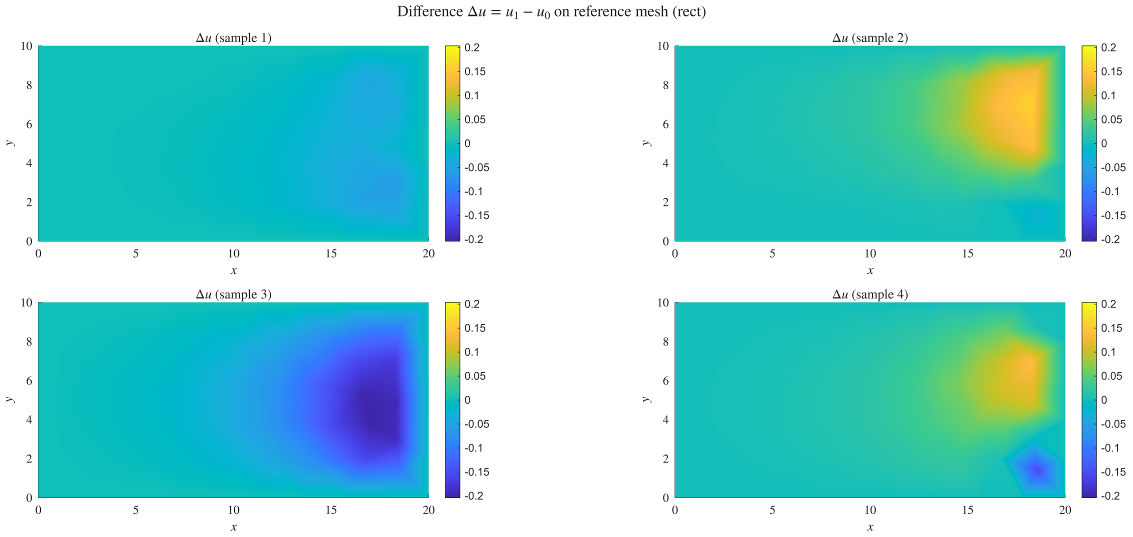


Figure 5.14: Difference field $\Delta U = U^{(s)} - U^{\mathrm{ref}}$ for four realizations, plotted on the static reference mesh to isolate the variance.

5.2.5 Monte Carlo Convergence and Output Statistics

To quantify the stochastic response of the rectangular benchmark, two global quantities of interest (QoIs) are evaluated for each realization:

$$Q_{\text{max}}^{(s)} = \max_{i=1,\ldots,N_n} U_i^{(s)}, \tag{5.8}$$

and

$$Q_{\text{mean}}^{(s)} = \frac{1}{A_{\text{tot}}^{(s)}} \sum_{e=1}^{N_e} \bar{U}_e^{(s)} A_e^{(s)}, \tag{5.9}$$

where $U_i^{(s)}$ denotes the nodal solution value at node i for realization s, $\bar{U}_e^{(s)}$ is the arithmetic mean of the three nodal values of element e, $A_e^{(s)}$ is the area of the deformed element, and

$$A_{\text{tot}}^{(s)} = \sum_{e=1}^{N_e} A_e^{(s)}$$

is the total area of the perturbed domain. In this way, Q_{mean} is computed as an area-weighted discrete domain average and remains consistent with the deformed physical configuration.

Figures 5.15 and 5.16 show the running sample mean and running sample standard deviation for Q_{max} and Q_{mean}, respectively. In both cases, the estimators become progressively more stable as the number of Monte Carlo samples increases, indicating satisfactory convergence of the sampling procedure for the present benchmark.

The empirical distributions of Q_{max} and Q_{mean} are shown in Figure 5.17. Both distributions appear approximately symmetric and bell-shaped, despite the uniformly distributed geometric perturbations at the boundary. This behavior is consistent with the smoothing and aggregating character of the elliptic Poisson operator, which transforms the boundary variability into comparatively regular global response measures.

5.2.6 Summary

The two-dimensional rectangle benchmark demonstrates that the proposed non-intrusive methodology scales effectively to multi-dimensional problems. The harmonic properties of the Laplace-based ALE mapping successfully preserve mesh validity ($J > 0$) without remeshing, even under highly irregular boundary noise. The statistical estimators also converge robustly, demonstrating that the fully decoupled T_3 evaluation pipeline is both stable and computationally reliable. This establishes the necessary foundation for tackling complex internal boundary uncertainties in the final numerical example.

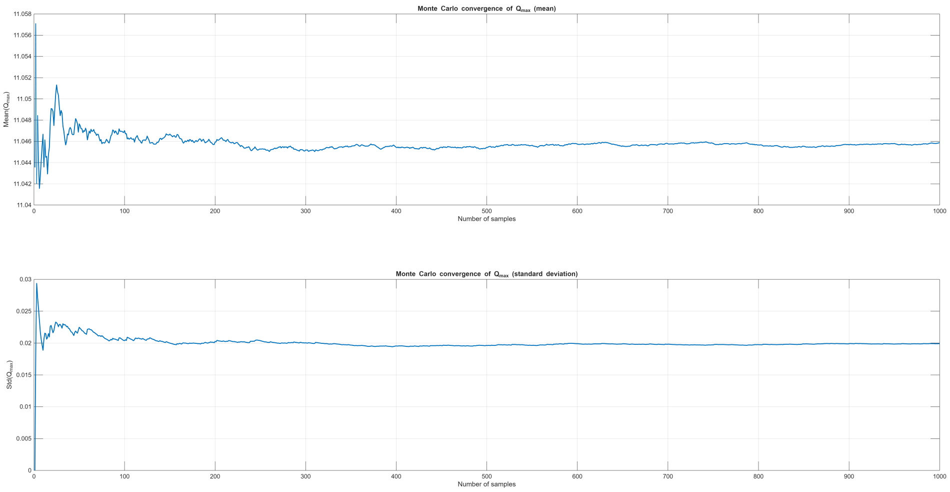


Figure 5.15: Monte Carlo convergence of the running mean and running standard deviation of the maximum solution value $Q_{\max}$.

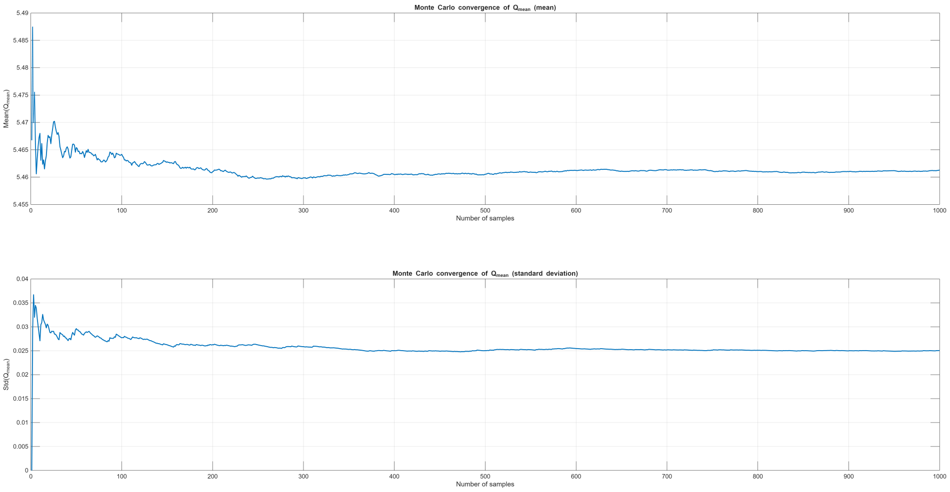


Figure 5.16: Monte Carlo convergence of the running mean and running standard deviation of the area-weighted domain mean Q_{mean}.

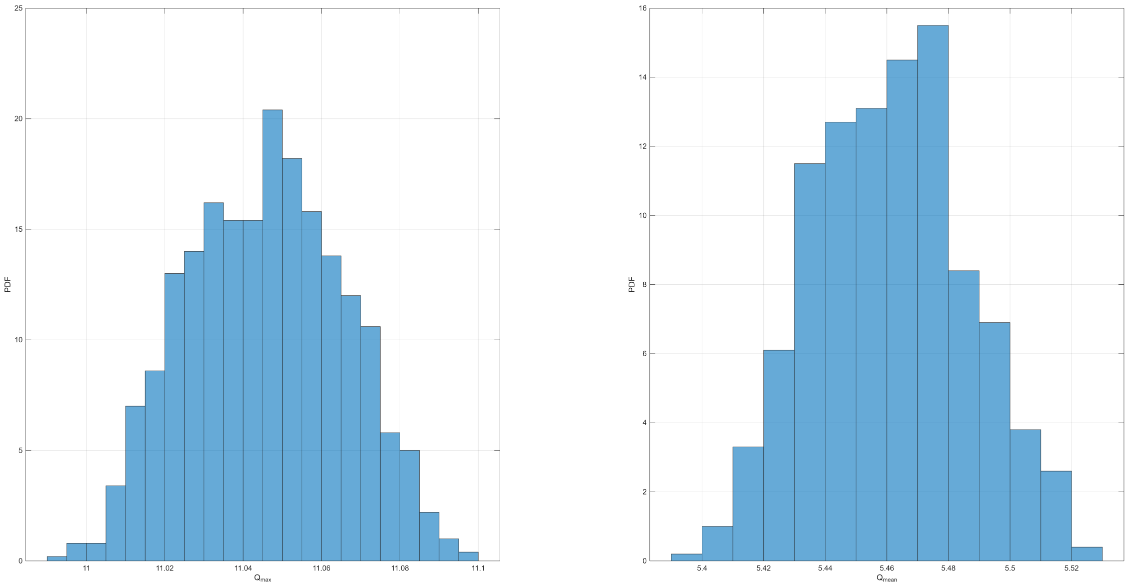


Figure 5.17: Empirical distributions of Q_{max} (left) and Q_{mean} (right) for the rectangular benchmark. Both quantities exhibit approximately symmetric, bell-shaped distributions.

5.3 Final Example: Plate with Uncertain Elliptical Hole

This final numerical example is designed to demonstrate the full scope of the proposed non-intrusive framework for the propagation of geometric boundary uncertainty. The configuration consists of a rectangular plate containing an internal elliptical hole, inspired by the benchmark problem introduced by Zheng et al. [41].

In contrast to the previous verification cases, geometric uncertainty acts exclusively on an internal boundary. This configuration is motivated by typical manufacturing tolerances and shape variability that affect internal features such as holes or cutouts in engineering components. The primary objective is to demonstrate that the Laplace-based ALE strategy can handle internal perturbations without costly remeshing. Furthermore, this example serves as a unifying test case: it emphasizes the theoretical and computational distinctions between modeling the uncertain geometric inputs as probabilistic **random variables** (Monte Carlo) versus bounded **interval variables** (Bayesian Global Optimization).

5.3.1 Problem Definition and Reference Configuration

The reference domain $\Omega^0 \subset \mathbb{R}^2$ represents the nominal (mean) configuration of the geometry. The boundary is decomposed as $\partial\Omega^0 = \Gamma_{\text{out}} \cup \Gamma_{\text{hole}}$, where Γ_{out} denotes the fixed outer boundary and Γ_{hole} represents the internal elliptical hole. The reference hole is centered at (x_c^0, y_c^0) with nominal semi-axes a_0 and b_0.

To evaluate the physical response, a deterministic Poisson problem is solved on the domain. The governing boundary value problem is defined as:

$$\begin{aligned} -\nabla^2 u &= 1 \qquad \text{in } \Omega(\Xi), \\ u &= 0 \qquad \text{on } \partial\Omega(\Xi), \end{aligned} \tag{5.10}$$

subject to a constant source term $f = 1$ and homogeneous Dirichlet boundary conditions on all boundaries (Γ_{out} and Γ_{hole}).

Finite Element Discretization: As established in the two-dimensional rectangular benchmark, both the auxiliary Laplace-based mesh deformation mapping and the physical Poisson solver are discretized using the exact same linear triangular (T_3) finite elements. The reference mesh (Figure 5.18) is generated once. For all subsequent uncertainty realizations, the mesh topology remains strictly invariant; the geometric variations are incorporated exclusively via nodal coordinate updates in the evaluation of the T_3 element Jacobians. Figure 5.19 displays the finite element solution on the undeformed reference geometry. The scalar field vanishes on the boundaries and attains a maximum in the interior ligaments, exhibiting noticeable sensitivity near the hole where

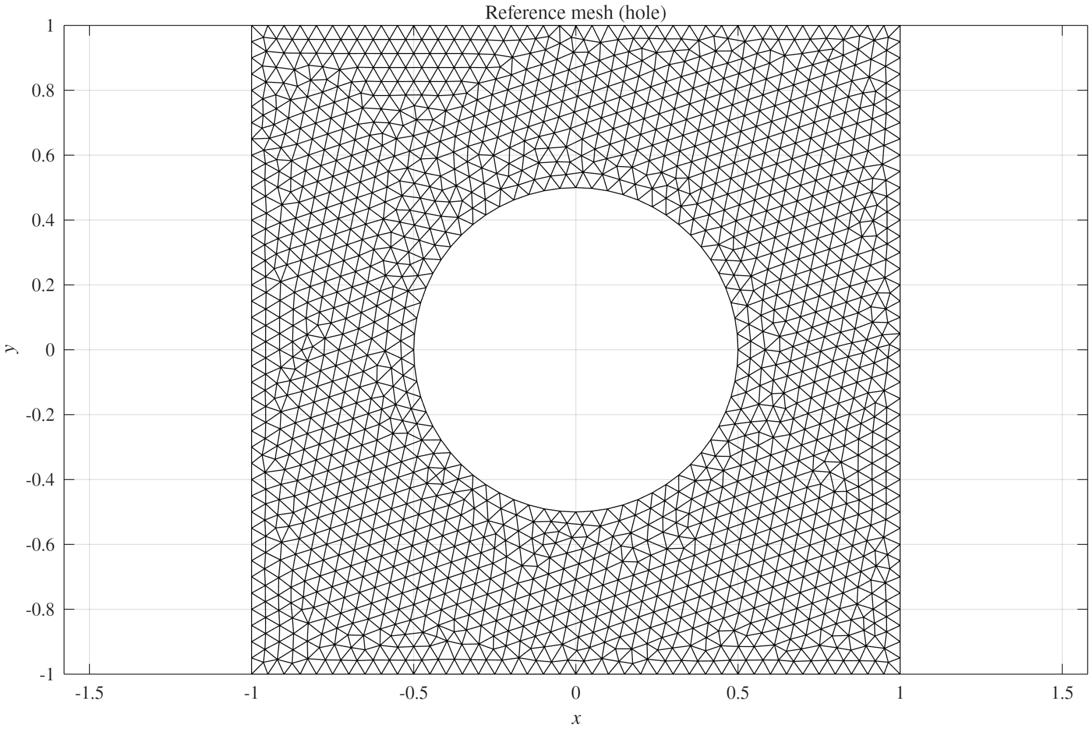


Figure 5.18: Finite element solution of the Poisson problem on the reference plate-with-hole geometry ($f = 1$, homogeneous Dirichlet BCs).

the internal boundary locally constrains the field.

5.3.2 Laplace-Based Deformation and Mesh Validity

Geometric uncertainty is introduced by prescribing boundary displacements $\mathbf{g}(\mathbf{x}, \boldsymbol{\Xi})$ on Γ_{hole}, while imposing zero displacement on Γ_{out}. The boundary motion is propagated to all interior mesh nodes by solving the vector-valued Laplace equation ($\nabla^2 \mathbf{u}_m = \mathbf{0}$).

Figure 5.20 illustrates an overlay of the reference mesh and a representative Laplace-deformed mesh. The harmonic nature of the Laplace operator filters high-frequency perturbations at the boundary, distributing the deformation smoothly throughout the domain without inducing sharp internal gradients.

To guarantee mathematical validity, explicit element inversion monitoring is performed by evaluating the signed oriented area ($2A_e$) of each T_3 element. As shown in Figure 5.21, the minimum oriented area indicator remains strictly positive across all sampled parameter ranges in this study, confirming that mesh integrity is preserved.

Physical Validation Against a Refined Remeshed Reference

Before using the Laplace-based mesh deformation framework for large ensembles of uncertainty-quantification samples, it is necessary to verify that the deformed-mesh solution remains physically consistent with a reference solution obtained from direct remeshing. For this purpose, the plate-with-hole benchmark was examined under three deliberately severe perturbation cases designed to stress the geometric mapping. The nominal geometry for this validation study uses semi-axes $a_0 = 2.0$, $b_0 = 1.5$ with mesh characteristic length $l_c = 0.2$ and a refined reference at $l_c^{\text{ref}} = 0.06$. The three cases are defined numerically in Table 5.1.

A direct nodal comparison between the Laplace ALE solution and a remeshed solution is not meaningful, since both discretizations are defined on different meshes with different node locations and connectivity. A straightforward transfer of one solution onto the nodes of the other mesh would already introduce an additional comparison error associated with interpolation between nonmatching discretizations. To obtain a more robust numerical validation, both the Laplace ALE solution and the standard remeshed solution were therefore

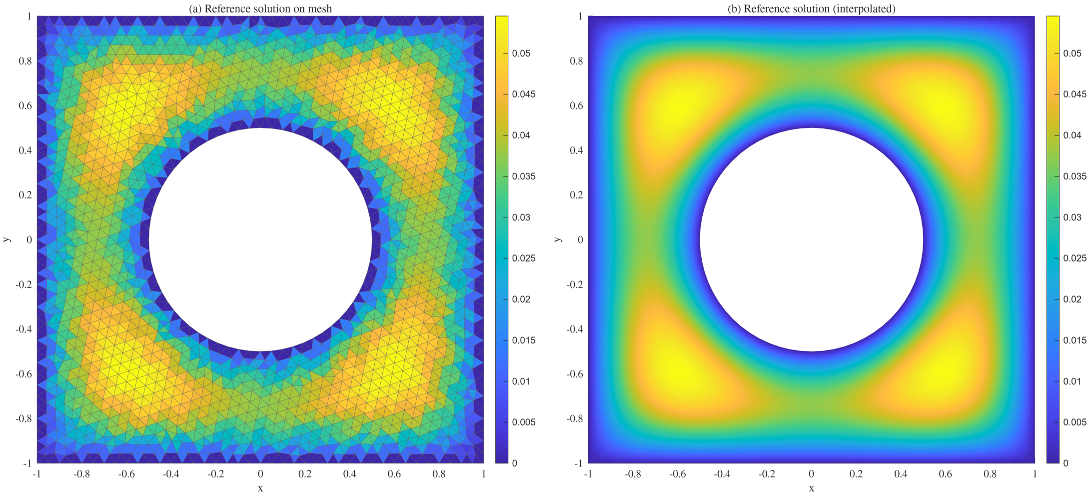


Figure 5.19: Finite element solution of the Poisson problem on the reference plate-with-hole geometry ($f = 1$, homogeneous Dirichlet BCs). (a) Raw discrete solution plotted directly on the T_3 mesh topology. (b) Spatially interpolated continuous field demonstrating the symmetric physical response.

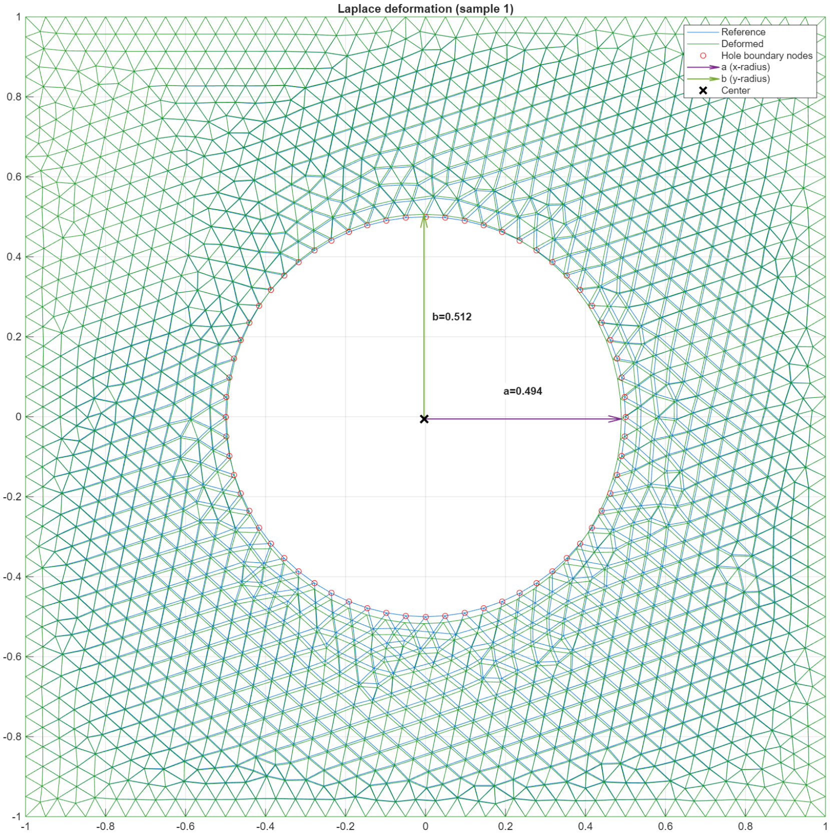


Figure 5.20: Overlay of the reference mesh and a representative Laplace-deformed mesh. The deformation smoothly adapts to the perturbed internal boundary while the outer boundary remains fixed.

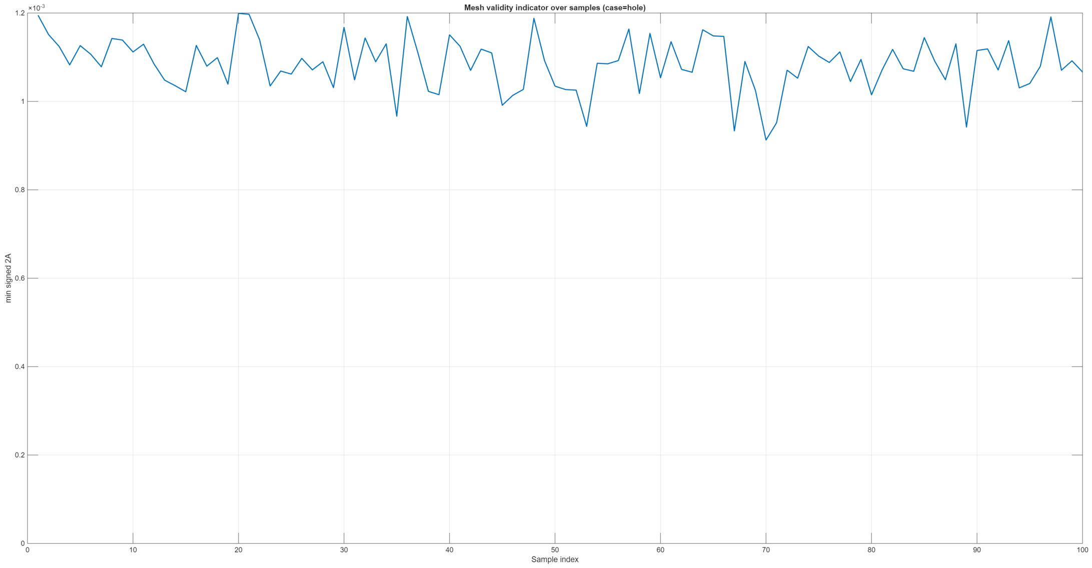


Figure 5.21: Minimum oriented area indicator over all evaluations. Positive values confirm the preservation of a valid, non-inverted finite element mesh.

Table 5.1: Parameter values for the three extreme perturbation cases used in the physical validation. The nominal configuration is $a_0 = 2.0$, $b_0 = 1.5$, centered at the origin.

Case	a	b	Δx	Δy
1: Extreme Translation	2.0	1.5	$+0.6$	-0.6
2: Extreme Eccentricity	2.8	0.9	0.0	0.0
3: Extreme Expansion	2.5	2.0	0.0	0.0

compared against a refined remeshed reference solution constructed on the same perturbed geometry but with a substantially finer mesh.

For each perturbation case, three solutions were computed:

1. the Laplace ALE solution on the deformed reference-topology mesh,
2. the standard remeshed solution on a newly generated mesh of comparable nominal density, and
3. a refined remeshed reference solution on a substantially finer mesh.

Since these three solutions are defined on different discretizations, they were evaluated on a common Cartesian sampling grid over the physical domain. Only grid points for which all three interpolants were defined were retained. It should be noted that the scattered linear interpolation onto this common grid introduces its own approximation error, which may differ between discretizations due to their different node distributions. The reported error metrics therefore reflect a combination of discretization error and interpolation error. Nevertheless, this procedure makes it possible to compare both the Laplace ALE and the standard remeshed solution against the same refined reference field in a symmetric manner.

Figures 5.22–5.24 show the resulting solution comparison for the three extreme perturbation cases. In all cases, the Laplace ALE solution, the standard remeshed solution, and the refined remeshed reference are visually very similar at the contour scale shown. The lower-row difference plots indicate that the remaining discrepancies are localized primarily near the perturbed hole boundary and in regions where the local mesh topology and element distortion differ most strongly. These patterns should therefore be interpreted as a combination of discretization, geometry-representation, and interpolation effects, rather than as evidence of a fundamental breakdown of the Laplace-based mapping.

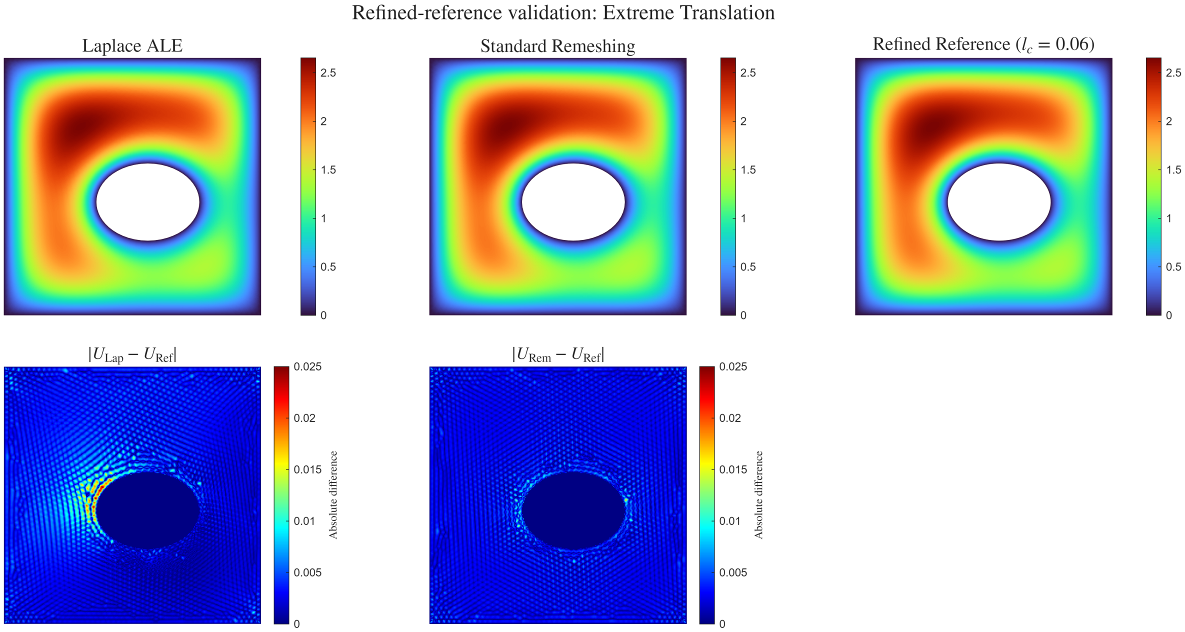


Figure 5.22: Physical validation against a refined remeshed reference for Case 1 (Extreme Translation). The top row compares the Laplace ALE solution, the standard remeshed solution, and the refined remeshed reference. The bottom row shows the absolute difference fields $|U_{\text{Lap}} - U_{\text{Ref}}|$ and $|U_{\text{Rem}} - U_{\text{Ref}}|$ on a common evaluation grid. Both difference plots use the same color scale.

The extreme translation case in Figure 5.22 is particularly demanding because the shifted hole compresses the local ligament on one side while expanding it on the other. Even under this strongly asymmetric perturbation, both practical discretizations remain close to the refined reference field, with the largest discrepancies concentrated near the hole boundary where the geometry changes most abruptly.

Figure 5.23 shows the eccentricity case, in which the hole is strongly elongated. Here the local mesh geometry differs more strongly near the major axis of the ellipse, and the error patterns become more localized

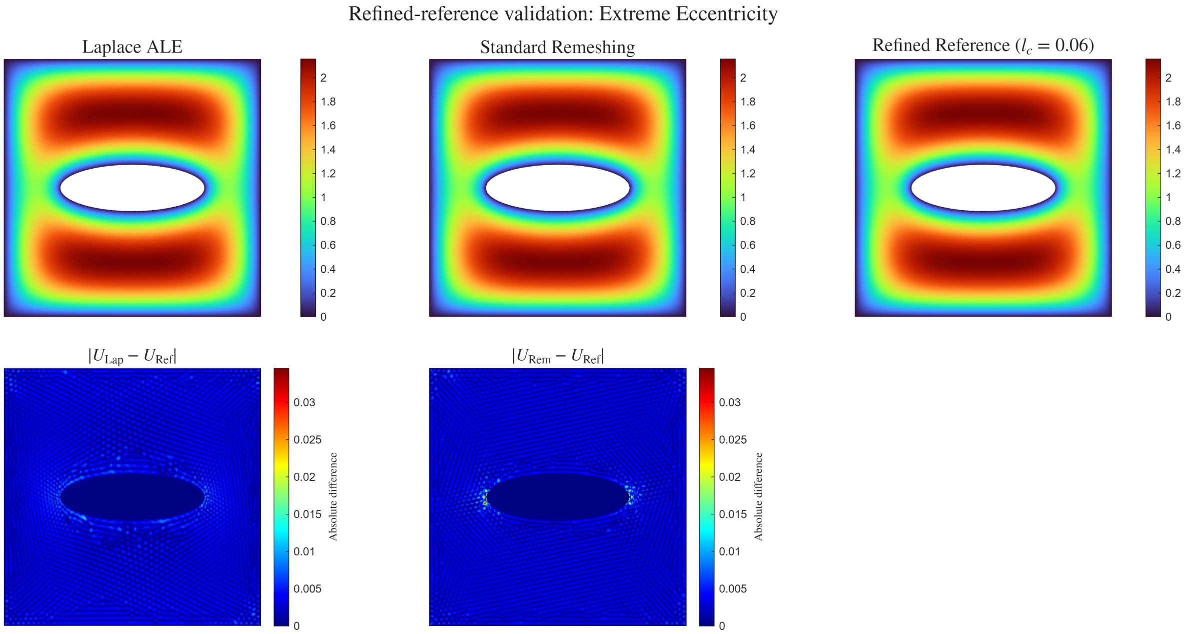


Figure 5.23: Physical validation against a refined remeshed reference for Case 2 (Extreme Eccentricity). The top row compares the Laplace ALE solution, the standard remeshed solution, and the refined remeshed reference. The bottom row shows the absolute difference fields $|U_{\text{Lap}} - U_{\text{Ref}}|$ and $|U_{\text{Rem}} - U_{\text{Ref}}|$ on a common evaluation grid. Both difference plots use the same color scale.

along the stretched boundary region. Nevertheless, the overall solution fields remain visually close to the refined reference.

In the expansion case of Figure 5.24, the hole occupies a substantially larger fraction of the domain. This case is particularly relevant for assessing whether the fixed-topology deformation introduces noticeable inaccuracies under strong volumetric change. The results again show that both the Laplace ALE solution and the standard remeshed solution remain close to the refined reference.

To quantify the agreement, the following discrete relative error measures were computed on the common evaluation grid:

$$e_{L_2} = \frac{\|U - U_{\text{Ref}}\|_2}{\|U_{\text{Ref}}\|_2}, \qquad e_{L_\infty} = \frac{\|U - U_{\text{Ref}}\|_\infty}{\|U_{\text{Ref}}\|_\infty}, \tag{5.11}$$

where U denotes either the Laplace ALE solution or the standard remeshed solution, and U_{Ref} denotes the refined remeshed reference field evaluated on the same valid grid points.

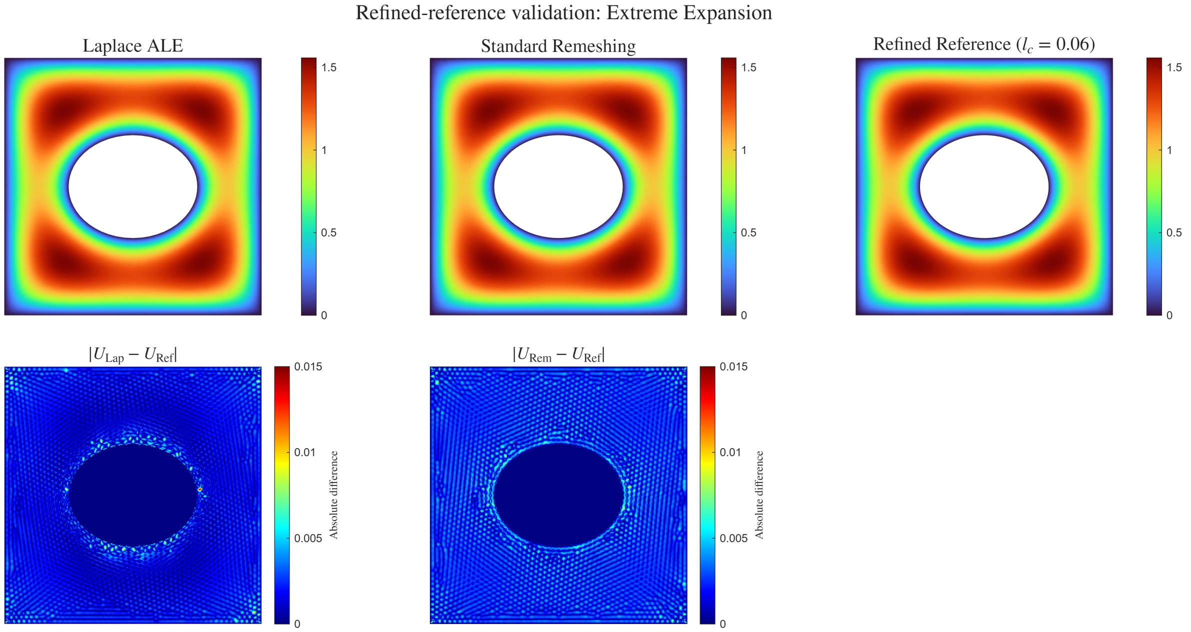
In addition to the field-based error norms, the maximum quantity of interest,

$$Q_{\max} = \max_{\mathbf{x} \in \Omega} u(\mathbf{x}),$$

was compared against the refined reference in order to assess whether the physically relevant peak response is preserved.

Table 5.2 summarizes the resulting error metrics. Several observations are important. First, both the Laplace ALE solution and the standard remeshed solution remain very close to the refined remeshed reference in all three severe perturbation cases. Second, the field-based error norms (e_{L_2}, e_{L_∞}) of the Laplace ALE method are of the same order of magnitude as those of the standard remeshed solution. Third, the relative error in $Q_{\max}$ remains very small in all cases (below 0.1%), although the individual values vary non-uniformly across perturbation types and methods, with neither approach being consistently superior.

The results do not suggest that one of the two practical discretizations is uniformly superior in every norm and every perturbation case. For the field-based norms, both methods show similar accuracy against the refined reference. For $Q_{\max}$, the individual relative errors vary more strongly between cases, but remain below 0.1% in all instances. This is the key validation result: the topology-preserving Laplace deformation does not introduce a



Refined-reference validation: Extreme Expansion

Laplace ALE

Standard Remeshing

Refined Reference ($l_c = 0.06$)

$|U_{\mathrm{Lap}} - U_{\mathrm{Ref}}|$

$|U_{\mathrm{Rem}} - U_{\mathrm{Ref}}|$

Figure 5.24: Physical validation against a refined remeshed reference for Case 3 (Extreme Expansion). The top row compares the Laplace ALE solution, the standard remeshed solution, and the refined remeshed reference. The bottom row shows the absolute difference fields $|U_{\text{Lap}} - U_{\text{Ref}}|$ and $|U_{\text{Rem}} - U_{\text{Ref}}|$ on a common evaluation grid. Both difference plots use the same color scale.

Table 5.2: Comparison of the Laplace ALE and standard remeshed solutions against a refined remeshed reference for the three extreme perturbation cases.

Perturbation Type	$e_{L_2}^{\text{Lap,Ref}}$	$e_{L_\infty}^{\text{Lap,Ref}}$	$e_{L_2}^{\text{Rem,Ref}}$	$e_{L_\infty}^{\text{Rem,Ref}}$	$\Delta Q_{\text{max}}^{\text{Lap,Ref}}$	$\Delta Q_{\text{max}}^{\text{Rem,Ref}}$
Extreme Translation	2.47×10^{-3}	9.42×10^{-3}	1.87×10^{-3}	6.48×10^{-3}	0.0801%	0.0037%
Extreme Eccentricity	2.03×10^{-3}	6.33×10^{-3}	1.91×10^{-3}	1.60×10^{-2}	0.0375%	0.0013%
Extreme Expansion	1.87×10^{-3}	9.63×10^{-3}	2.60×10^{-3}	6.66×10^{-3}	0.0024%	0.0448%

large additional physical error for the benchmark considered, even in deliberately severe geometric perturbation scenarios.

Interpretation of Local Error and Mesh Topology Although the global field discrepancies are small, the local error patterns are not spatially uniform. The standard remeshing pipeline generates a new triangulation for each perturbed geometry, while the Laplace ALE approach preserves the reference topology and updates only the nodal coordinates. As a consequence, the deformed Laplace mesh may contain locally skewed or compressed elements in regions of strong boundary motion, whereas the remeshed solution distributes the same geometric change through a newly adapted triangulation.

This topological distinction is illustrated in Figures 5.25–5.27. These mesh comparisons should not be interpreted as an independent validation measure by themselves, but rather as a geometric explanation for the localized error structures already observed in the refined-reference difference fields.

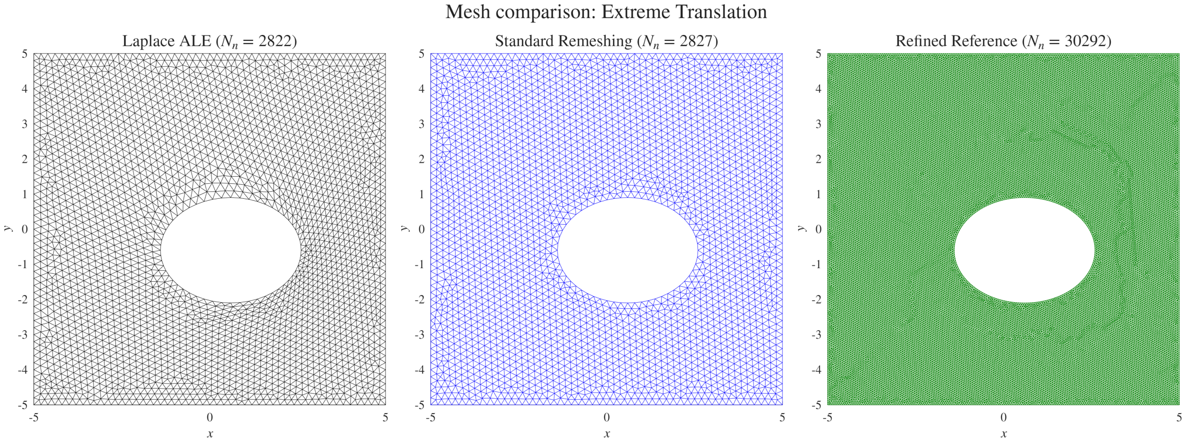


Figure 5.25: Mesh comparison for Case 1 (Extreme Translation). The Laplace ALE method preserves the reference topology and deforms the existing elements, while the remeshing-based workflows generate new triangulations adapted to the perturbed geometry.

In the translation case, the strong asymmetric displacement of the hole leads to localized compression of the surrounding mesh on one side and expansion on the other. The Laplace ALE mesh accommodates this change through continuous deformation of the original connectivity, whereas the remeshed discretizations redistribute element density by constructing a new triangulation.

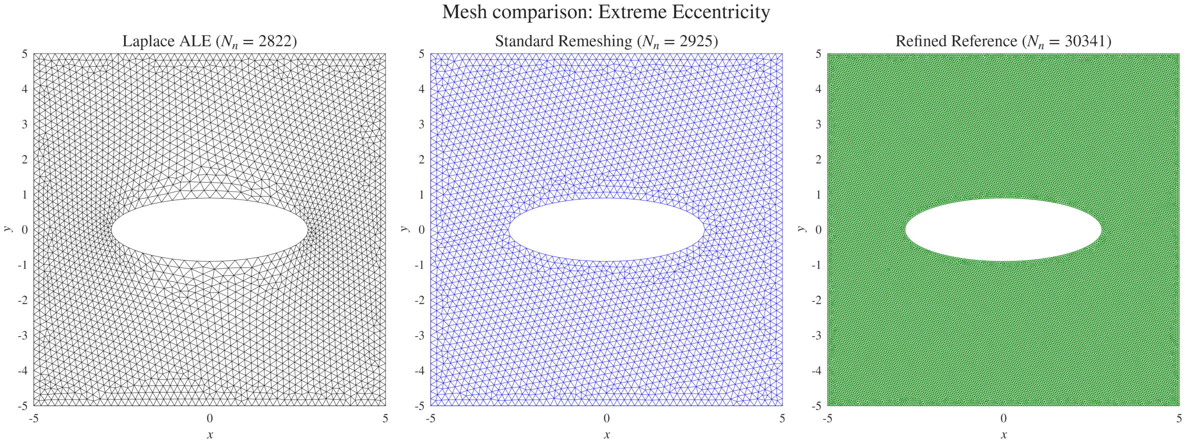


Figure 5.26: Mesh comparison for Case 2 (Extreme Eccentricity). The Laplace ALE method preserves the reference topology and deforms the existing elements, while the remeshing-based workflows generate new triangulations adapted to the perturbed geometry.

In the eccentricity case, the elongated hole induces more pronounced element skewness near the major axis in the Laplace mesh. This provides a plausible explanation for the more localized differences seen in the corresponding solution comparison, without implying a substantial loss of overall physical fidelity.

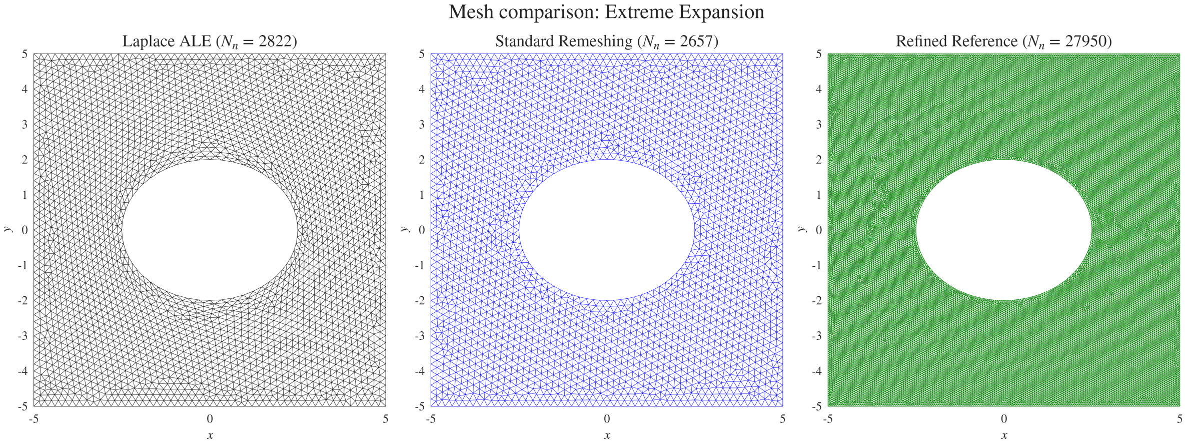


Figure 5.27: Mesh comparison for Case 3 (Extreme Expansion). The Laplace ALE method preserves the reference topology and deforms the existing elements, while the remeshing-based workflows generate new triangulations adapted to the perturbed geometry.

In the expansion case, the reduced physical domain is represented by a denser packing of deformed elements in the Laplace ALE mesh, while the remeshed discretizations reduce or redistribute the node density in a different way. Even in this more demanding geometric configuration, the small global error norms and the very small deviations in $Q_{\max}$ indicate that the induced mesh distortion does not lead to a substantial loss of physical accuracy for the benchmark considered.

Overall, the refined-reference comparison provides strong evidence that the Laplace ALE framework preserves the physical response of the plate-with-hole problem with an accuracy comparable to that of a standard remeshing workflow, while retaining the computational advantages discussed in the subsequent benchmark section.

Computational Efficiency Benchmark

Having established physical consistency between the remeshing-based reference workflow and the proposed Laplace ALE approach, the computational efficiency of both evaluation pipelines was examined. The benchmark was designed to measure the mean CPU time required for repeated evaluations of geometrically perturbed plate-with-hole realizations over $N_s = 100$ samples per mesh level.

Important note on benchmark geometry: The computational efficiency benchmark and the preceding physical validation (Section 5.3.2) use a different nominal geometry than the subsequent uncertainty propagation studies (Sections 5.3.3 and 5.3.4). Specifically, the benchmark employs a larger elliptical hole with nominal semi-axes $a_0 = 2.0$, $b_0 = 1.5$ and perturbation ranges of $\pm 10\%$ on semi-axes and ± 0.2 on center position. In contrast, the UM-1 and UM-2 studies use a nominal circular hole with $a_0 = b_0 = 0.5$ and center shifts of ± 0.05 (see Table 5.5). The larger geometry was chosen for the benchmark and validation studies to produce more visually distinct mesh deformations and to stress-test the framework under larger absolute perturbation magnitudes.

To study the influence of discretization size, the comparison was performed on three mesh refinement levels controlled by the characteristic length l_c, shown in Figure 5.28.

The benchmark compares the complete implemented workflow required to evaluate one sample. For the remeshing-based pipeline, this includes external mesh generation with Gmsh, mesh import into MATLAB, boundary-condition setup on the newly generated mesh, assembly of the physical Poisson system, enforcement of Dirichlet boundary conditions, linear system solution, and quantity-of-interest evaluation. For the Laplace ALE pipeline, the corresponding workflow consists of constructing the boundary perturbation on the reference mesh, solving the auxiliary Laplace mesh-deformation problem in memory, updating the nodal coordinates, assembling and solving the physical Poisson problem on the deformed mesh, and evaluating the same quantity of interest.

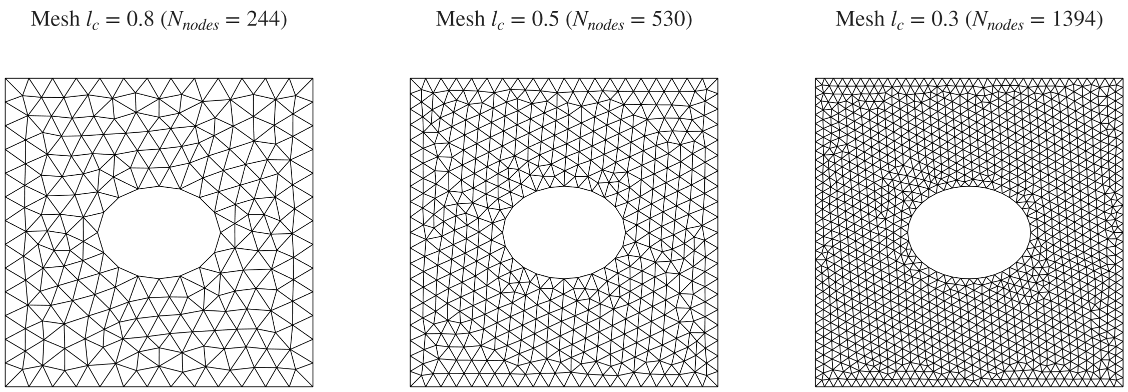


Figure 5.28: Reference mesh refinement levels used in the computational benchmark. From left to right: coarse mesh ($l_c = 0.8$, 244 nodes), medium mesh ($l_c = 0.5$, 530 nodes), and fine mesh ($l_c = 0.3$, 1394 nodes).

A timing decomposition of both pipelines is shown in Figure 5.29, while the corresponding quantitative summary is reported in Table 5.3. The results show that the dominant cost of the remeshing-based workflow is not the physical Poisson solve itself, but the repeated geometry and mesh handling required for every realization. In contrast, the Laplace ALE method performs all geometric updates in memory on a fixed reference topology, which leads to a substantially lower end-to-end cost per sample.

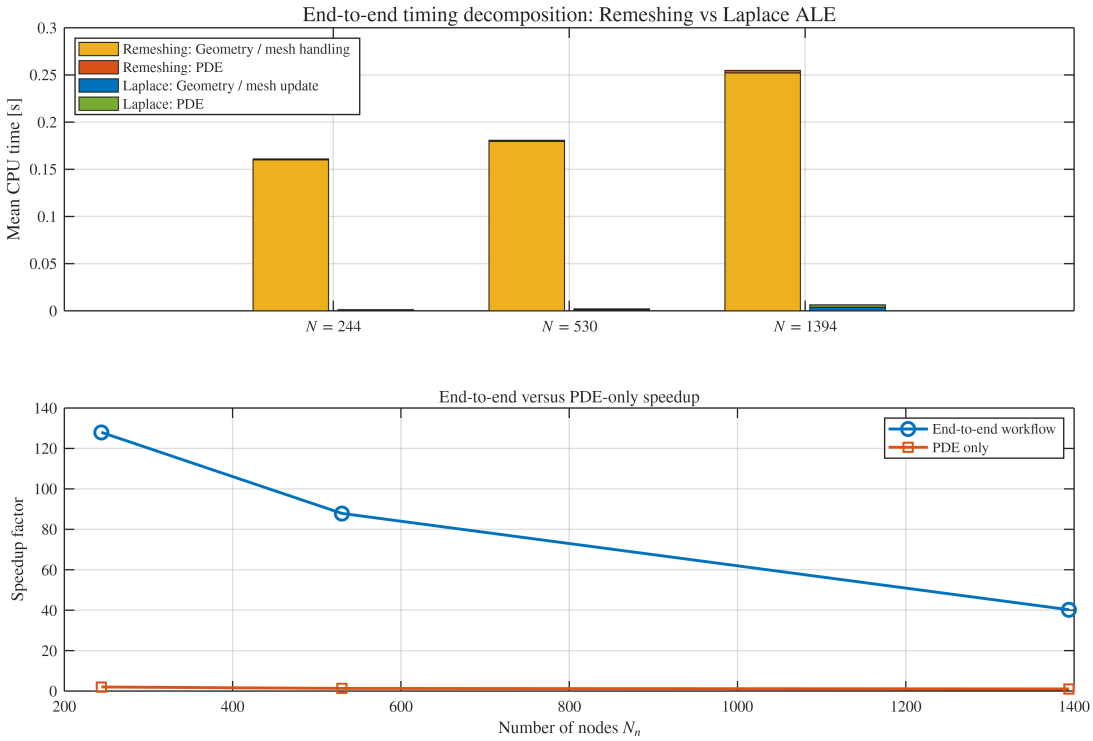


Figure 5.29: Timing decomposition of the remeshing and Laplace ALE workflows. The upper panel compares the mean time per sample for geometry/mesh handling and downstream PDE operations, while the lower panel shows the resulting end-to-end and PDE-only speedup factors as the mesh is refined.

Across the tested mesh resolutions, the Laplace ALE workflow achieved large end-to-end speedups ranging from approximately $41\times$ to $111\times$. The decomposition in Figure 5.29 clarifies the origin of this gain: for the remeshing-based approach, the overwhelming majority of runtime is spent in repeated external mesh generation and mesh import, whereas the downstream physical PDE stage contributes only a small fraction of the total cost. By contrast, the Laplace ALE workflow avoids repeated topology reconstruction and performs the geometric update directly on the reference mesh [40, 6], so that its total cost remains dominated by in-memory linear

Table 5.3: Summary of the computational benchmark. The end-to-end speedup refers to the complete implemented workflow. The PDE-only speedup compares only the downstream physical solve stage after geometry handling has been excluded.

Mesh Level	**Laplace (s)**	**Remesh (s)**	**End-to-End Speedup**	**PDE-Only Speedup**	**Mean $\Delta Q_{\max}$ (%)**
Coarse ($N = 244$)	0.001401	0.155362	110.91x	1.85x	0.78
Medium ($N = 530$)	0.001879	0.180195	95.92x	1.39x	0.23
Fine ($N = 1394$)	0.006029	0.247993	41.14x	1.06x	0.13

algebra operations.

An important observation is that the PDE-only speedup is much smaller than the end-to-end speedup, decreasing from $1.85\times$ on the coarse mesh to $1.06\times$ on the fine mesh. This indicates that the principal computational advantage of the proposed method does not stem from the downstream physical solve itself, since both workflows ultimately solve the same Poisson problem on meshes of similar size, but from avoiding the repeated remeshing overhead associated with geometry generation, mesh import, and mesh reconstruction. This distinction is important for interpreting the benchmark correctly: the reported large speedups should be understood as workflow-level gains for repeated evaluations, rather than as a universal intrinsic factor for the Poisson solve itself.

The end-to-end speedup decreases with mesh refinement. This trend is expected, since the downstream physical solve becomes a larger fraction of the total runtime as the number of unknowns increases. Consequently, the relative influence of geometry-handling overhead is strongest on coarse and medium meshes, while the overall speedup remains substantial even on the finest mesh considered.

At the same time, the discrepancy in the maximum quantity of interest remains small throughout the benchmark. The mean relative difference in $Q_{\max}$ decreases from 0.78% on the coarse mesh to 0.13% on the fine mesh, which indicates that the strong reduction in computational cost is accompanied by close agreement in the selected response measure for the present benchmark configuration.

Remarks on Fixed Topology and Repeated Evaluation The timing decomposition also highlights a structural advantage of the proposed framework. In the remeshing-based pipeline, each perturbed realization produces a new mesh and therefore a new discrete connectivity graph. In the Laplace ALE framework, the mesh topology remains fixed and only the nodal coordinates are updated. This avoids repeated mesh reconstruction and allows all geometric updates to remain within a single reference discretization. In the present implementation, this structural property primarily manifests itself through the elimination of repeated external meshing and import costs. While it is plausible that a fixed sparsity pattern can also reduce some downstream assembly and solver overhead, the benchmark results show that this contribution is secondary compared with the dominant savings obtained from avoiding repeated remeshing.

Remark on Implementation-Specific Overhead The remeshing baseline in the preceding benchmark relies on external invocation of Gmsh as a separate operating system process, which is the standard approach for integrating Gmsh with MATLAB. This workflow involves process creation, file-based mesh export, and subsequent file import, all of which contribute to the measured geometry-handling overhead. In other computational environments, such as Python-based frameworks, the Gmsh library can be called directly through an in-memory API, which eliminates process creation and disk I/O entirely.

To quantify this contribution, an additional micro-benchmark was performed using the Gmsh Python API [14] (version 4.15.1), which generates the mesh entirely in memory on a geometrically equivalent parametric plate-with-hole configuration using the OpenCASCADE kernel. The results are summarized in Table 5.4. For the same mesh parameters, in-memory mesh generation required approximately 17–59 ms per sample, compared to 155–248 ms for the external-process workflow. This indicates that process invocation and disk I/O account for a substantial fraction of the measured geometry-handling cost. Under the assumption that the downstream PDE operations remain comparable, the estimated end-to-end speedup of the Laplace ALE workflow over a hypothetical in-memory remeshing baseline reduces to approximately 9–$13\times$, compared to the 41–$111\times$ measured with external invocation. The structural advantages of the proposed framework, in particular the preservation of mesh topology, the avoidance of repeated connectivity reconstruction, and the elimination of field-transfer operations, remain valid independently of the mesh generation implementation. Further details on the micro-benchmark setup are provided in Appendix E.

Table 5.4: Comparison of geometry-handling overhead between external-process and in-memory Gmsh invocation. The in-memory timings were obtained using the Gmsh Python API (version 4.15.1) on the same machine. Node counts differ slightly due to differences in the geometry kernel (built-in vs. OpenCASCADE).

Mesh Level	**External Gmsh (ms)**	**In-Memory Gmsh (ms)**	**Overhead Factor**	**Est. Speedup (in-memory)**
Coarse ($l_c = 0.8$)	155.4	16.6	9.4$\times$	$\sim$13$\times$
Medium ($l_c = 0.5$)	180.2	26.2	6.9$\times$	$\sim$16$\times$
Fine ($l_c = 0.3$)	248.0	59.2	4.2$\times$	$\sim$12$\times$

It should also be noted that the present benchmark is restricted to a stationary scalar Poisson problem. In more complex settings, such as transient analyses, nonlinear material behavior, or history-dependent constitutive models, a remeshing-based workflow may require repeated transfer of state fields between non-matching meshes. Such projection and interpolation steps introduce additional computational cost and can also affect numerical accuracy [22, 35]. Since the Laplace ALE framework preserves node numbering and connectivity, such transfer operations can in principle be reduced or avoided. However, these additional advantages were not included in the present timing benchmark and therefore should be regarded as a potential extension rather than as a demonstrated result of the current study.

To illustrate the practical consequence of these per-sample differences in a full uncertainty propagation setting, a Monte Carlo simulation with $N_s = 1000$ realizations was executed on the coarse mesh ($l_c = 0.8$, $N = 244$). The total runtime breakdown is shown in Figure 5.30. In this experiment, the remeshing-based workflow required approximately 252 seconds, whereas the Laplace ALE workflow completed the same ensemble in approximately 6.6 seconds, corresponding to an empirical end-to-end speedup of about $38\times$.

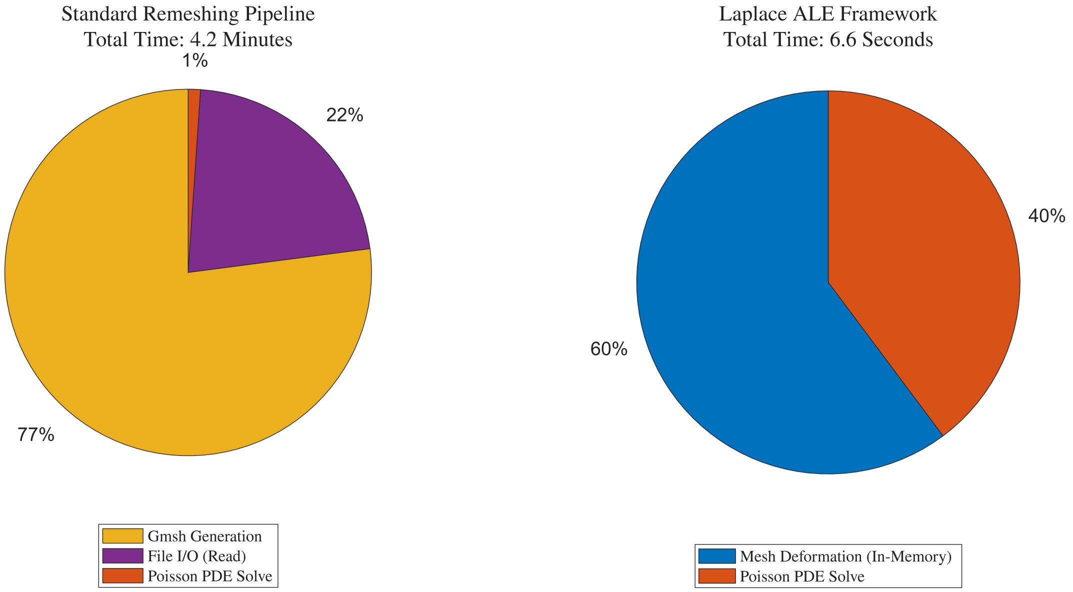


Figure 5.30: Total runtime breakdown for 1000 Monte Carlo evaluations on the coarse mesh ($l_c = 0.8$, $N = 244$). The remeshing-based workflow is dominated by repeated external mesh generation and mesh import, whereas the Laplace ALE workflow remains fully in memory.

For uncertainty quantification, where hundreds or thousands of repeated model evaluations are required, this workflow-level reduction in computational cost is a major practical advantage of the Laplace ALE framework.

5.3.3 Uncertainty Model UM-1: Random Variables (Probabilistic)

The first uncertainty model (UM-1) adopts a probabilistic parametric description of the hole geometry. The uncertain configuration is represented by the random vector

$$\Xi = \{a, b, \Delta x, \Delta y\},$$

where the semi-axes of the hole and the translation of its center are treated as independent random variables. The geometry is defined by

$$\begin{aligned} a = a_0\left(1 + \delta_a \xi_1\right), \qquad b = b_0\left(1 + \delta_b \xi_2\right), \\ \Delta x = \delta_x \xi_3, \qquad \Delta y = \delta_y \xi_4, \end{aligned} \tag{5.12}$$

where $\xi_i \sim \mathcal{U}(-1, 1)$ are independent standard uniform random variables and $\delta_a, \delta_b, \delta_x, \delta_y$ denote the perturbation amplitudes.

The numerical values used in all subsequent UM-1 simulations are summarized in Table 5.5. A total of $N_s = 1000$ Monte Carlo samples were evaluated with a fixed random seed (seed = 2) to ensure reproducibility.

Table 5.5: Parameter values for the probabilistic uncertainty model UM-1 (plate with hole).

Parameter	**Value**	**Description**
a_0	0.5	Nominal semi-axis (horizontal)
b_0	0.5	Nominal semi-axis (vertical)
δ_a	0.10	Relative perturbation amplitude for a
δ_b	0.10	Relative perturbation amplitude for b
δ_x	0.05	Maximum center shift in x
δ_y	0.05	Maximum center shift in y
N_s	1000	Number of Monte Carlo samples

This model generates a family of admissible hole geometries with varying size, eccentricity, and position while preserving the mesh topology through the Laplace-based deformation framework. Figure 5.31 shows four representative realizations selected from the Monte Carlo ensemble. Although all realizations are drawn from the same probabilistic model, the resulting physical fields exhibit visibly different patterns due to changes in the local ligament thickness between the hole boundary and the outer Dirichlet boundary. Since the Poisson operator acts diffusively, the solution tends to attain larger values in regions where the distance to the zero-displacement boundary is locally increased.

To better isolate the effect of geometric uncertainty, the field difference

$$\Delta U^{(s)} = U^{(s)} - U^{\text{ref}}$$

was additionally evaluated on the reference configuration, where U^{ref} denotes the Poisson solution on the nominal geometry. The resulting difference fields, shown in Figure 5.32, are smooth and spatially structured. This suggests that the dominant response variations are geometry-driven and not dominated by visible localized numerical artifacts.

Quantities of Interest and Monte Carlo Statistics

To quantify the stochastic response, three quantities of interest (QoIs) are evaluated for each realization:

1. Q_{max}: the maximum solution value in the domain,
2. Q_{mean}: the area-weighted mean value over the deformed domain $\Omega(\Xi)$,
3. Q_{ring}: an area-weighted near-hole mean over an annular region centered at the perturbed hole center, with inner and outer radii defined by

   $$r_{\text{min}} = \alpha r_{\text{eq}}, \qquad r_{\text{max}} = \beta r_{\text{eq}},$$

   where $r_{\text{eq}} = \sqrt{a_0 b_0}$ is the nominal equivalent radius and $\alpha = 1.10$, $\beta = 1.40$.

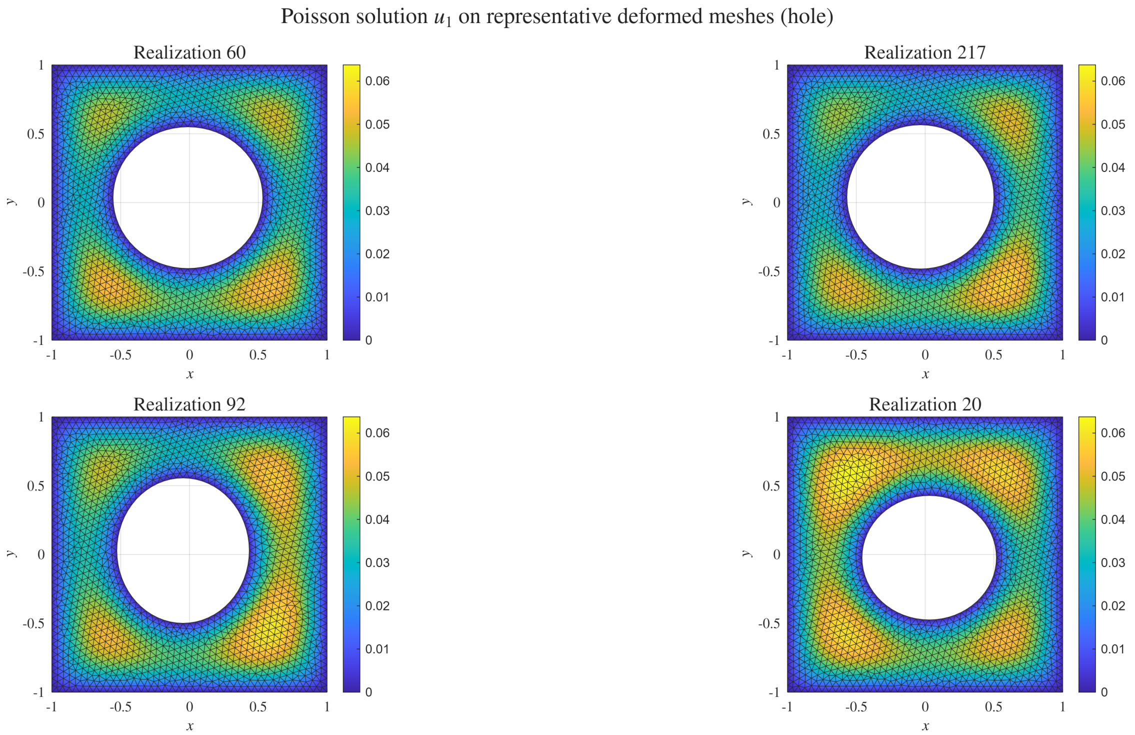


Figure 5.31: Poisson solution u_1 on four representative UM-1 realizations, shown on the corresponding deformed meshes with a shared color scale. The selected realizations illustrate how random variations in hole size and position modify the global response while remaining within the probabilistic uncertainty model.

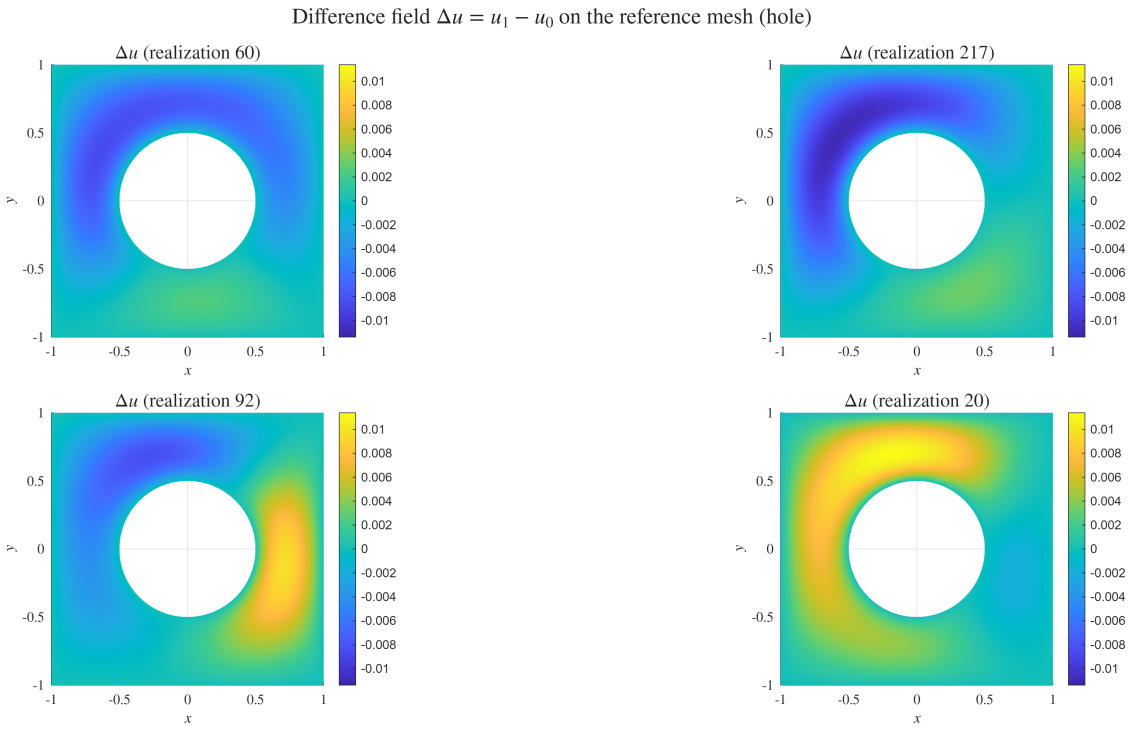


Figure 5.32: Difference field $\Delta U = U^{(s)} - U^{\text{ref}}$ for the same four representative UM-1 realizations, plotted on the reference mesh. The resulting variations are smooth and spatially coherent, indicating that the dominant changes are induced by the geometric perturbations.

The first two QoIs characterize the global response of the domain, whereas Q_{ring} is designed to probe the solution in the immediate neighborhood of the uncertain internal boundary, where the geometric perturbation is expected to have the strongest influence.

A Monte Carlo simulation with $N_s = 1000$ realizations was performed. The cumulative sample means and standard deviations of all three QoIs are shown in Figure 5.33. As the sample count increases, the estimated statistics become progressively more stable. Among the three QoIs, Q_{ring} exhibits the largest standard deviation, which is physically plausible since it samples the response in the region most directly affected by the hole perturbation. By contrast, the domain-wide mean Q_{mean} is less sensitive, as the local geometric effects are averaged over the full domain.

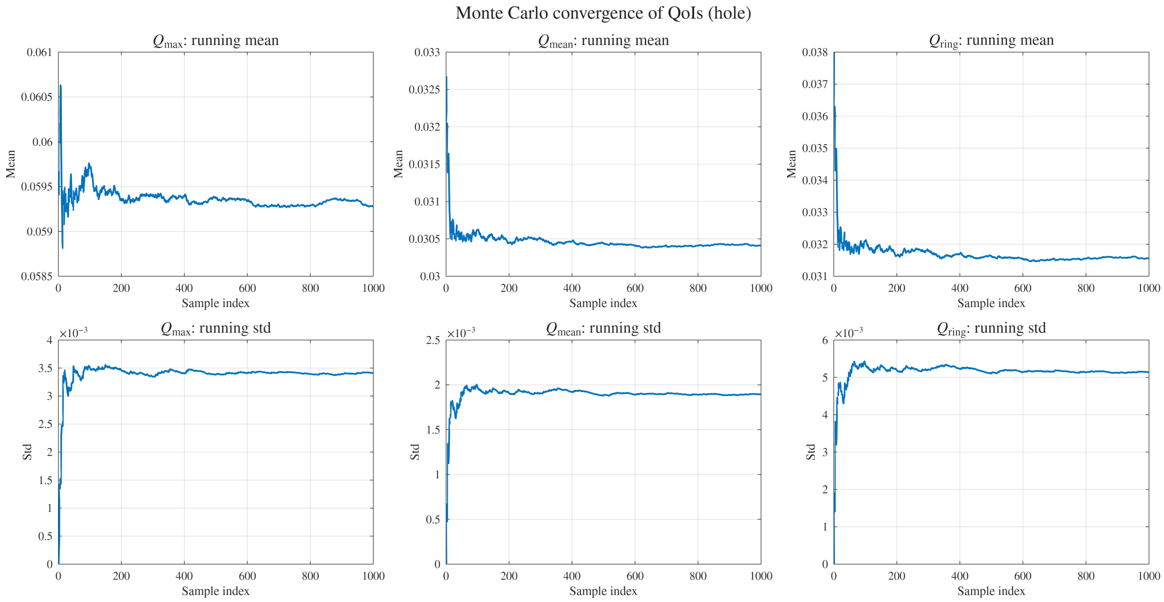


Figure 5.33: Monte Carlo convergence of the three QoIs for UM-1 ($N_s = 1000$). The top row shows the running sample means, and the bottom row shows the running sample standard deviations for Q_{max}, Q_{mean}, and Q_{ring}.

The empirical distributions of the three QoIs are shown in Figure 5.34. All three distributions appear approximately symmetric and bell-shaped, although their spreads differ noticeably. The widest spread is observed for Q_{ring}, again reflecting the fact that the near-hole response is most sensitive to the uncertain geometry. The distribution of Q_{mean} is narrower, which is consistent with the averaging effect of the elliptic solution operator over the full domain.

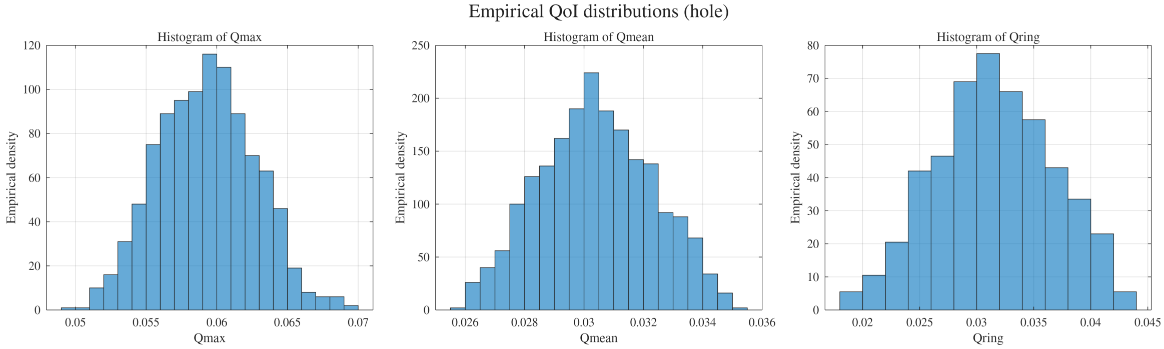


Figure 5.34: Empirical distributions of the UM-1 quantities of interest for $N_s = 1000$. Left: Q_{max}. Center: Q_{mean}. Right: Q_{ring}. The near-hole quantity Q_{ring} shows the largest spread, reflecting its stronger sensitivity to local boundary perturbations.

Figures 5.31–5.34 show that the probabilistic uncertainty model produces smooth and physically interpretable variations in the solution field, while the Monte Carlo statistics remain well behaved. This provides a consistent basis for the subsequent comparison with the interval-based uncertainty model and for the later surrogate-based analysis.

5.3.4 Uncertainty Model UM-2: Interval Variables (Bounded)

In many engineering applications, geometric uncertainty is specified through manufacturing tolerances or admissible design ranges rather than through statistical data. In such situations, assigning a probability density function is not justified. Uncertainty Model UM-2 therefore treats the geometric parameters

$$\boldsymbol{\xi} = (a, b, \Delta x, \Delta y)$$

as bounded interval variables,

$$\xi_i \in [\underline{\xi}_i, \overline{\xi}_i],$$

with the admissible parameter space given by the corresponding four-dimensional hyper-rectangle. In contrast to UM-1, the objective is no longer to estimate statistical moments of the response, but to identify lower and upper response bounds over the full admissible interval box.

Monte Carlo as a Baseline Envelope Estimator

A straightforward baseline approach is to sample the interval box uniformly and track the running minimum and maximum of the resulting response values. However, this procedure is not designed for efficient extremum identification. In a bounded multidimensional parameter space, random sampling explores the domain diffusely, and regions associated with near-extremal responses may occupy only a small fraction of the total volume. As a result, the running envelope obtained from finite Monte Carlo sampling generally converges slowly.

This behavior is illustrated in Figure 5.35 for the quantity of interest Q_{mean}. The light-blue points show the values obtained from $N = 1000$ uniformly distributed Monte Carlo samples, while the blue curves indicate the running minimum and maximum. The red lines denote the lower and upper values identified by the NPBGO procedure. Even after 1000 evaluations, the Monte Carlo sample envelope remains narrower than the bounds found by the surrogate-based optimizer, which illustrates the limited efficiency of uniform random sampling for interval-bound estimation.

Non-Parallel Bayesian Global Optimization (NPBGO)

To estimate the interval bounds more efficiently, the Non-Parallel Bayesian Global Optimization (NPBGO) strategy of Dang is employed. The finite element model is treated as a black-box map

$$\boldsymbol{\xi} \mapsto Q(\boldsymbol{\xi}),$$

where Q denotes the selected quantity of interest. For each QoI, the algorithm first generates an initial Latin hypercube design of $N_0 = 30$ samples over the admissible parameter box. Based on these evaluations, a Gaussian process surrogate is constructed and iteratively refined.

The optimization procedure is carried out in two sequential phases: one for the lower bound and one for the upper bound. In each phase, new evaluation points are selected by maximizing an Expected Improvement criterion tailored to minimization or maximization, respectively. The procedure is terminated using the stopping criterion implemented in the original NPBGO code, which monitors the normalized improvement indicator over successive iterations.

Figure 5.36 compares the evolution of the Monte Carlo running bounds with the bounds identified by NPBGO for all three quantities of interest. In each case, the optimizer reaches tighter lower and upper bounds using only a small fraction of the model evaluations required by the Monte Carlo baseline.

The quantitative comparison is summarized in Table 5.6. For all three QoIs, the NPBGO procedure identifies a lower minimum and a higher maximum than the Monte Carlo baseline, despite using only 34–43 total evaluations (of which $N_0 = 30$ correspond to the initial Latin hypercube design) instead of 1000. This demonstrates that optimization-based bound estimation is substantially more efficient than uniform random sampling for the present interval uncertainty problem.

The strongest relative gain is observed in the localized quantity Q_{ring}, which is particularly sensitive to the hole geometry and therefore represents the most challenging response measure from the perspective of bound

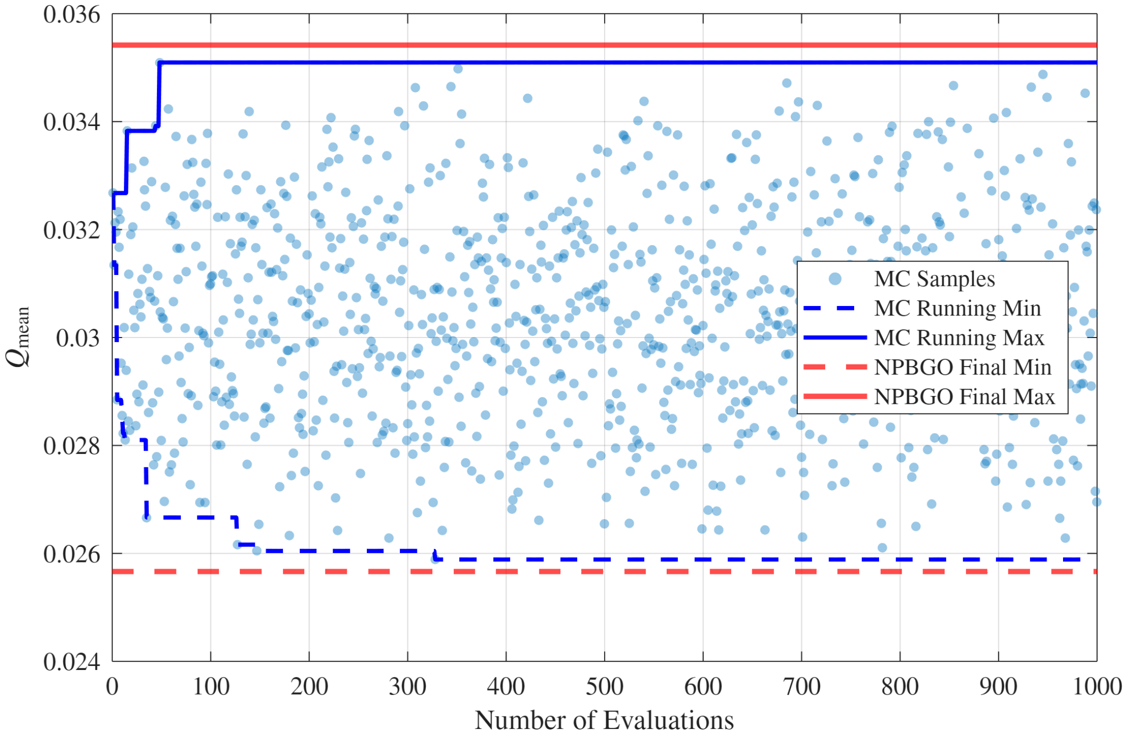
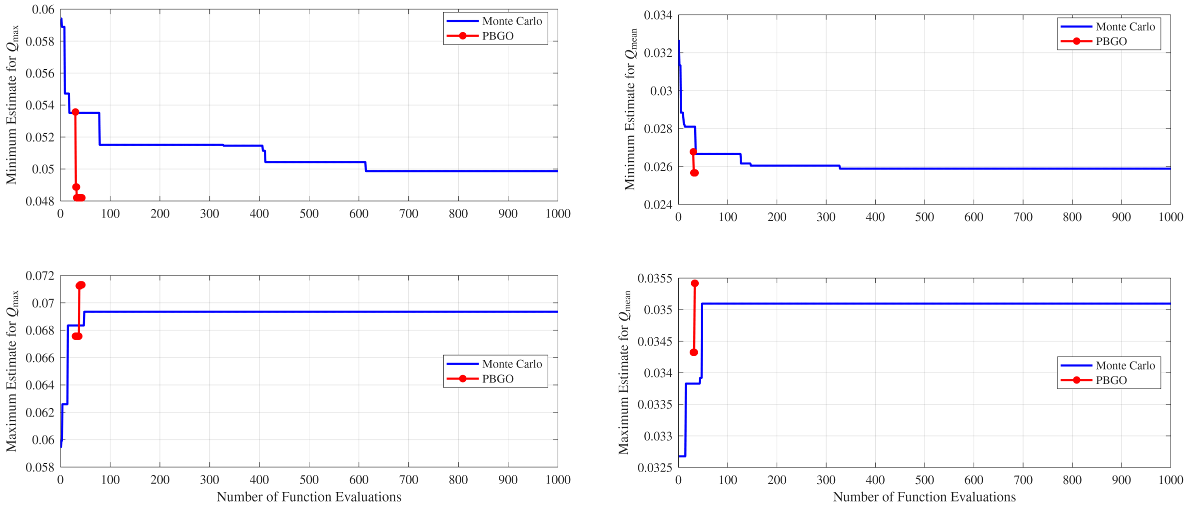


Figure 5.35: Uniform Monte Carlo sampling of the interval box for Q_{mean} ($N = 1000$). The light-blue points show the sampled responses, the blue curves show the running minimum and maximum, and the red lines show the lower and upper values identified by the NPBGO procedure. The Monte Carlo envelope expands only gradually and remains narrower than the optimizer-based bounds.

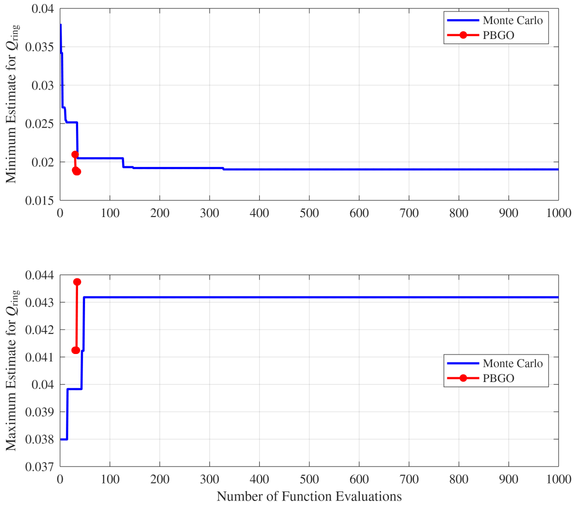
Table 5.6: Comparison of interval bounds obtained by uniform Monte Carlo sampling and NPBGO for the three quantities of interest.

QoI	**MC Min**	**MC Max**	**NPBGO Min**	**NPBGO Max**	**MC Evals**	**NPBGO Evals**
Q_{max}	4.9869×10^{-2}	6.9348×10^{-2}	4.8204×10^{-2}	7.1314×10^{-2}	1000	43
Q_{mean}	2.5888×10^{-2}	3.5096×10^{-2}	2.5664×10^{-2}	3.5418×10^{-2}	1000	34
Q_{ring}	1.9028×10^{-2}	4.3182×10^{-2}	1.8717×10^{-2}	4.3740×10^{-2}	1000	35



(a) Q_{max}

(b) Q_{mean}



(c) Q_{ring}

Figure 5.36: Comparison of Monte Carlo running bounds and NPBGO-identified bounds for the three quantities of interest. In all cases, the surrogate-based optimization identifies lower minima and higher maxima than the Monte Carlo baseline using substantially fewer function evaluations.

identification. Nevertheless, the same qualitative behavior is obtained for the global quantities $Q_{\max}$ and Q_{mean}: the Monte Carlo envelope expands slowly, whereas NPBGO rapidly identifies tighter lower and upper response values.

Overall, these results show that the proposed ALE-based finite element framework can be coupled effectively with surrogate-based global optimization for interval uncertainty propagation. The mesh deformation and physical solve can be evaluated sufficiently efficiently to support repeated black-box optimization queries, enabling practical estimation of response bounds for geometric interval uncertainty.

Chapter 6

Conclusion and Outlook

6.1 Conclusion

This thesis addressed the efficient treatment of geometric boundary uncertainty in finite element analysis by means of a Laplace-based mesh deformation strategy within an Arbitrary Lagrangian-Eulerian (ALE) framework. The central objective was to develop and validate a non-intrusive computational methodology that avoids repeated remeshing by deforming a fixed reference mesh according to prescribed boundary perturbations, while preserving mesh connectivity and discretization consistency across uncertainty realizations.

The proposed framework was formulated in a fully modular and non-intrusive manner. For each realization of the uncertain geometry, only the nodal coordinates were updated, whereas the finite element topology remained unchanged. This separation between geometric update and physical solve made it possible to treat the deterministic solver as a black box and to compare responses across realizations without introducing topological inconsistencies caused by regenerated meshes.

The findings of this thesis are summarized with respect to the three research questions formulated in Section 1.2.

RQ 1: Accuracy of Laplace-based mesh deformation. The methodology was assessed through a sequence of increasingly demanding benchmark problems. In the one-dimensional validation example, the deformed-mesh solution reproduced the analytical reference with errors at machine precision level, confirming the correctness of the mapping in the simplest setting. In the two-dimensional rectangular benchmark, highly irregular synthetic boundary perturbations were propagated smoothly into the interior, demonstrating the harmonic smoothing character of the Laplace operator. In the plate-with-hole benchmark, a refined-reference comparison showed that the Laplace ALE approach preserves the physical response with good accuracy. The field-based error norms remained of the same order of magnitude as those of a standard remeshing workflow, and the relative error in the peak response Q_{max} stayed below 0.1% in all three deliberately severe perturbation cases.

RQ 2: Computational efficiency within Monte Carlo propagation. The timing decomposition showed that the dominant advantage of the proposed method arises from the elimination of repeated geometry generation, mesh import, and mesh reconstruction, rather than from a fundamentally different physical solve. Across the tested mesh resolutions, end-to-end speedups ranged from approximately $41\times$ on the fine mesh to $111\times$ on the coarse mesh. In a full Monte Carlo demonstration with $N_s = 1000$ realizations, the Laplace ALE workflow completed the ensemble in a fraction of the time required by the remeshing-based pipeline. These speedups include the overhead of external process invocation and file I/O inherent to the MATLAB-based workflow; a supplementary micro-benchmark using the Gmsh Python API showed that with in-memory mesh generation the estimated speedup reduces to approximately 9–$13\times$, confirming that the advantage remains substantial but is partly implementation-specific. The principal gain is therefore a workflow-level reduction in computational cost for repeated evaluations under geometric variability.

RQ 3: Performance under severe perturbations and varying mesh densities. Under deliberately extreme perturbation cases (strong translation, eccentricity, and expansion of the internal boundary), the Laplace ALE solution remained close to the refined remeshed reference. The mean relative difference in Q_{max} between the Laplace ALE and remeshed solutions decreased from 0.78% on the coarse mesh to 0.13% on the fine mesh, indicating that the deformation-based approach benefits from mesh refinement in the same manner as the remeshing baseline. The end-to-end speedup decreased with mesh refinement, which is expected since the downstream physical solve becomes a larger fraction of the total runtime as the number of unknowns increases.

The uncertainty-propagation studies further confirmed the flexibility of the framework. Under the probabilistic uncertainty model UM-1, the method produced smooth, physically interpretable field variations and well-behaved Monte Carlo statistics for the quantities of interest Q_{max}, Q_{mean}, and Q_{ring}. Under the interval uncertainty model UM-2, the same deformation-based finite element model could be coupled with a surrogate-assisted global optimization strategy. The results showed that uniform Monte Carlo sampling over the interval box provides only a slowly expanding response envelope, whereas the NPBGO procedure identified tighter lower and upper response bounds using far fewer function evaluations. This demonstrates that the proposed framework is compatible not only with sampling-based probabilistic analysis, but also with optimization-based interval bound estimation.

The results of this thesis show that Laplace-based mesh deformation provides an accurate, robust, and computationally efficient alternative to repeated remeshing for moderate geometric boundary perturbations that preserve domain topology. By maintaining nodal correspondence and a fixed mesh connectivity, the method enables consistent uncertainty propagation while reducing the geometry-handling overhead that would otherwise dominate repeated finite element evaluations.

At the same time, the conclusions must be interpreted within the scope of the present study. The investigation was deliberately restricted to a scalar elliptic model problem (the Poisson equation) under linear static conditions, and to perturbation magnitudes for which the mesh remains valid under harmonic deformation. The Laplace operator acts as a smoothing mechanism, but it does not explicitly optimize element quality. For sufficiently large perturbations, local element distortion and eventually inversion may occur, especially near strongly deformed boundaries. Moreover, nonlinear material behavior, large deformations, dynamics, and topology changes were not considered. The efficiency and accuracy claims established here therefore apply primarily to problems in which geometric variations remain smooth and the mesh topology can be preserved.

This thesis demonstrates that Laplace-based mesh deformation constitutes a practically effective and numerically consistent strategy for the treatment of geometric boundary uncertainty in finite element analysis. Within the considered scope, it provides a strong foundation for non-intrusive uncertainty quantification on variable domains while avoiding the major computational penalties associated with repeated remeshing.

6.2 Outlook

Several directions for future research arise naturally from the present work.

A first extension concerns the physical model. The methodology should be transferred from the scalar Poisson problem to vector-valued systems such as linear elasticity, and subsequently to nonlinear constitutive models. In such settings, the interaction between geometric uncertainty, stress concentration, and material nonlinearity may lead to richer response behavior and more demanding requirements on the mesh-deformation strategy.

A second direction concerns the mesh-motion model itself. Although the Laplace operator proved robust and computationally inexpensive for the perturbation ranges considered here, larger geometric changes could benefit from alternative or enhanced deformation operators, such as elasticity-based, biharmonic, or quality-aware mesh-motion strategies. Hybrid approaches that combine smooth deformation with local remeshing in critical regions may offer a useful compromise between robustness and computational efficiency.

A third direction concerns uncertainty propagation in higher-dimensional parameter spaces. In the interval setting considered here, the NPBGO strategy already showed a clear advantage over uniform random sampling. Future work could investigate more scalable Bayesian optimization variants, including parallel or batch strategies, more sophisticated acquisition functions, and adaptive surrogate-management schemes for problems with a larger number of uncertain geometric descriptors.

A fourth direction concerns the source of the geometric uncertainty description itself. In this thesis, the uncertain boundary was parameterized through prescribed distributions (UM-1) or bounded intervals (UM-2) on a small number of shape parameters. In practice, geometric deviations arising from manufacturing processes, assembly tolerances, or operational wear are typically characterized through measurement data, such as coordinate measuring machine (CMM) readings, 3D surface scans, or statistical process control records. A natural extension would be to construct data-driven geometric uncertainty models, for example by fitting random field representations or principal-component-based shape parameterizations to measured boundary deviations, and to propagate these through the Laplace ALE framework. Because the proposed methodology accepts arbitrary boundary displacement fields and operates non-intrusively, it is well suited for integration with such data-driven uncertainty descriptions.

A fifth direction concerns the class of quantities of interest. In this thesis, the analysis focused on representative scalar outputs such as $Q_{\max}$, Q_{mean}, and Q_{ring}. For practical engineering applications, it would be valuable to extend the framework to more specialized response measures, such as stress-based indicators, compliance-type quantities, or reliability-relevant limit-state functions.

Finally, an important long-term direction is the integration of the proposed methodology into high-fidelity industrial workflows. Because the approach preserves topology and operates non-intrusively with an existing solver, it offers a promising basis for incorporating geometric boundary uncertainty into established simulation pipelines without a complete reformulation of the deterministic finite element code.

Overall, the present work establishes a consistent computational foundation for further developments in uncertainty-aware finite element analysis on variable domains. The combination of reference-domain modeling, Laplace-based mesh deformation, and non-intrusive uncertainty propagation provides a flexible starting point for future methodological advances and practical engineering applications.

Appendix A

MATLAB Implementation Structure

The numerical framework described in Chapter 4 is implemented in MATLAB and organized into the directory structure shown below. All source files are collected under a common project root.

```
matlab/
   config/
      cfg_2D_probabilistic.m
      cfg_2D_interval.m
   experiments/
      run_1D_beam.m
      run_2D_probabilistic.m
      run_2D_interval.m
      run_2D_interval_compare.m
      run_benchmark_timing.m
      run_benchmark_1000.m
      run_validation_extreme.m
   meshes/
      (Gmsh .msh and .geo reference mesh files)
   src/
      fem/
         assembleLaplaceKm_T3.m
         assemblePoisson_T3.m
         applyDirichletStrong.m
         buildKmVector2D.m
         buildMeshBCs.m
         minSignedTwoA_T3.m
         readGmsh22.m
      setup/
         setupCase.m
         getBoundaryNodesFromLines.m
         boundaryNodes_rectangleByCoords.m
         estimateHoleCenter.m
         getPhysicalId.m
      perturbation/
         holeEllipseBC.m
         holeEllipsePerturbationAngle.m
         sampleUM1_hole_uniform.m
         sampleUM1_rect_uniform.m
      uq/
         runMC_2D_probabilistic.m
         runIntervalMC_2D.m
         runNPBGO_2D.m
         evaluateQoI_intervalHole.m
         computeDomainMean_T3.m
         computeRingMean_T3.m
      pbgo/
         Non_parallel_Bayesian_global_optimization.m
```

```
    LHS.m
    TLBO.m
post/
    (plotting and visualization routines)
beam1d/
    assembleBeamEB_Hermite.m
    assembleLaplace1D.m
    applyDirichletBC_dofs.m
    readMsh1D.m
```

Table A.1 maps the main driver scripts to the corresponding thesis sections and numerical experiments.

Table A.1: Mapping of MATLAB driver scripts to thesis sections.

Script	**Purpose**	**Thesis Section**
`run_1D_beam.m`	1D beam validation	Section 5.1
`run_2D_probabilistic.m`	UM-1 Monte Carlo	Section 5.3.3
`run_2D_interval.m`	UM-2 NPBGO optimization	Section 5.3.4
`run_2D_interval_compare.m`	UM-2 MC vs NPBGO comparison	Section 5.3.4
`run_benchmark_timing.m`	Timing benchmark	Section 5.3.2
`run_validation_extreme.m`	Extreme-case validation	Section 5.3.2

The core finite element routines in `src/fem/` implement the element-level operations described in Chapter 4: `assembleLaplaceKm_T3.m` computes the scalar Laplace stiffness matrix $\mathbf{K}_m$ (Eq. 4.4), `buildKmVector2D.m` constructs the block-diagonal system blkdiag$(\mathbf{K}_m, \mathbf{K}_m)$, and `applyDirichletStrong.m` enforces Dirichlet boundary conditions via row/column elimination (Section 4.2.2). The boundary perturbation models in `src/perturbation/` define the geometric uncertainty parameterizations, while the uncertainty propagation loops in `src/uq/` wrap these routines within the Monte Carlo and NPBGO workflows described in Section 4.5.

Appendix B

Laplace Mesh Deformation Algorithm

This appendix provides detailed pseudocode for the Laplace-based mesh deformation procedure described in Section 4.2. The procedure computes the interior mesh displacement field $\mathbf{U}_m$ by solving the discrete Laplace equation

$$\nabla^2 \mathbf{u}_m = \mathbf{0} \quad \text{in } \Omega_0, \tag{B.1}$$

subject to prescribed displacements on the perturbed boundary Γ_{pert} and zero displacement on the fixed boundary Γ_{fixed}.

Key implementation detail. The scalar Laplace stiffness matrix $\mathbf{K}_m \in \mathbb{R}^{N_n \times N_n}$ depends only on the reference mesh coordinates **X** and is therefore assembled once during initialization, before the uncertainty propagation loop. The two-dimensional mesh displacement is solved as a single block-diagonal system $\text{blkdiag}(\mathbf{K}_m, \mathbf{K}_m)\, \mathbf{U}_m = \mathbf{0}$ with appropriate Dirichlet boundary conditions, as described in Section 4.2.

The complete deformation procedure for a single realization is given in Algorithm 2.

Algorithm 2 Laplace Mesh Deformation for a Single Realization

```
Require: Reference mesh $(\mathbf{X}, \mathcal{T}_h)$, preassembled block-diagonal stiffness $\mathbf{K}_{m,2D} = \text{blkdiag}(\mathbf{K}_m, \mathbf{K}_m)$, boundary node sets $(\Gamma_{\text{pert}}, \Gamma_{\text{fixed}})$, geometric parameters $\boldsymbol{\Xi}$
Ensure: Deformed coordinates $\mathbf{x}^*$, mesh validity flag
 1: Evaluate boundary perturbation model: $(\mathbf{U}_x, \mathbf{U}_y) \leftarrow \mathbf{g}(\mathbf{X}, \boldsymbol{\Xi})$ on $\Gamma_{\text{pert}}$
 2: Assign zero displacement on $\Gamma_{\text{fixed}}$
 3: Collect constrained DOF indices and values: (dofs, vals) $\leftarrow$ buildMeshBCs$(\Gamma_{\text{pert}}, \Gamma_{\text{fixed}}, \mathbf{U}_x, \mathbf{U}_y, N_n)$
 4: Apply Dirichlet BCs via row/column elimination: $(\mathbf{K}_{mod}, \mathbf{F}_{mod}) \leftarrow$ applyDirichletStrong$(\mathbf{K}_{m,2D}, \mathbf{0}$, dofs, vals$)$
 5: Solve for mesh displacement: $\mathbf{U}_m \leftarrow \mathbf{K}_{mod} \backslash \mathbf{F}_{mod}$
 6: Update coordinates: $\mathbf{x}^* \leftarrow \mathbf{X} + [\mathbf{U}_m(1{:}N_n),\ \mathbf{U}_m(N_n{+}1{:}2N_n)]$
 7: Mesh validity check: compute $\min_e(2A_e^*)$ via minSignedTwoA_T3
 8: if $\min_e(2A_e^*) \leq 0$ then
 9:     return invalid
10: end if
11: return $\mathbf{x}^*$, valid
```

For the plate-with-hole benchmark, the boundary perturbation in Step 1 is computed using an angle-based elliptical mapping (implemented in `holeEllipseBC.m`). Each boundary node on Γ_{hole} is assigned a displacement that maps its reference position from the nominal ellipse (a_0, b_0) to the perturbed ellipse (a, b) centered at $(x_c^0 + \Delta x,\ y_c^0 + \Delta y)$, using the angular coordinate of the node relative to the nominal hole center.

Appendix C

Monte Carlo Simulation Procedure

This appendix provides the detailed pseudocode for the probabilistic uncertainty propagation pipeline described in Section 4.5. The procedure combines the Laplace mesh deformation (Appendix B) with the physical Poisson solve within a standard Monte Carlo sampling loop. The implementation corresponds to `runMC_2D_probabilistic.m`.

Algorithm 3 Probabilistic Monte Carlo Pipeline (UM-1)

```
Require: Reference mesh $(\mathbf{X}, \mathcal{T}_h)$, configuration parameters (Table 5.5), number of samples $N_s$, random seed
Ensure: QoI arrays $\mathbf{Q}_{\text{max}}, \mathbf{Q}_{\text{mean}}, \mathbf{Q}_{\text{ring}}$
 1: Initialization (executed once):
 2: Set random seed: rng(seed)
 3: Identify boundary node sets: $\Gamma_{\text{pert}}, \Gamma_{\text{fixed}}$, Dirichlet nodes
 4: Assemble Laplace stiffness: $\mathbf{K}_m \leftarrow$ assembleLaplaceKm_T3$(\mathbf{X}, \mathcal{T}_h)$
 5: Build block-diagonal system: $\mathbf{K}_{m,2D} \leftarrow \text{blkdiag}(\mathbf{K}_m, \mathbf{K}_m)$
 6: Solve reference Poisson problem: $\mathbf{U}_0 \leftarrow$ assemble and solve on $(\mathbf{X}, \mathcal{T}_h)$
 7: Preallocate QoI arrays as NaN
 8: for $s = 1, \ldots, N_s$ do
 9:     Sample geometric parameters: $\boldsymbol{\Xi}^{(s)} \sim$ prescribed distribution
10:     Compute Laplace deformation $\rightarrow \mathbf{x}^*$ (Algorithm 2)
11:     if mesh invalid then
12:         Skip realization (QoI entries remain NaN); continue
13:     end if
14:     Assemble physical system on deformed mesh: $(\mathbf{K}_{phys}, \mathbf{F}_{phys}) \leftarrow$ assemblePoisson_T3$(\mathbf{x}^*, \mathcal{T}_h, f)$
15:     Apply physical Dirichlet BCs and solve: $\mathbf{U}_{phys} \leftarrow \mathbf{K}_{phys,mod} \backslash \mathbf{F}_{phys,mod}$
16:     $Q^{(s)}_{\text{max}} \leftarrow \max_i(U_{phys,i})$
17:     $Q^{(s)}_{\text{mean}} \leftarrow$ computeDomainMean_T3$(\mathbf{x}^*, \mathcal{T}_h, \mathbf{U}_{phys})$
18:     $Q^{(s)}_{\text{ring}} \leftarrow$ computeRingMean_T3$(\mathbf{x}^*, \mathcal{T}_h, \mathbf{U}_{phys}, \text{center}, r_{\text{min}}, r_{\text{max}})$
19: end for
20: Compute sample statistics (mean, variance, histograms) from non-NaN entries
```

Treatment of invalid realizations. If the Laplace deformation produces a mesh with $\min_e(2A_e^*) \leq 0$ (element inversion), the realization is skipped entirely: the corresponding QoI entries remain as NaN and are excluded from the subsequent statistical evaluation. This ensures that the Monte Carlo estimators are computed only over physically meaningful realizations.

Note on the reference solution. The Poisson problem is also solved on the undeformed reference mesh (Step 6) to obtain $\mathbf{U}_0$. This reference solution is used to compute the difference field $\Delta\mathbf{U}^{(s)} = \mathbf{U}^{(s)}_{phys} - \mathbf{U}_0$ for visualization purposes (Section 5.3.3), but does not enter the QoI evaluation.

Appendix D

Non-Parallel Bayesian Global Optimization

This appendix describes the Non-Parallel Bayesian Global Optimization (NPBGO) procedure used for interval bound estimation in the UM-2 uncertainty model (Section 5.3.4). The core optimization algorithm was developed and provided by Dr. Chao Dang [5]; the implementation effort of this thesis concerns the integration of the finite element evaluation pipeline as a black-box objective function, as described in Section 4.5.

Problem formulation. Given the deterministic mapping $\boldsymbol{\xi} \mapsto Q(\boldsymbol{\xi})$ from the geometric parameter vector to the quantity of interest, the interval uncertainty problem seeks

$$Q_{\text{min}} = \min_{\boldsymbol{\xi} \in [\mathbf{L}, \mathbf{U}]} Q(\boldsymbol{\xi}), \qquad Q_{\text{max}} = \max_{\boldsymbol{\xi} \in [\mathbf{L}, \mathbf{U}]} Q(\boldsymbol{\xi}),$$

where $[\mathbf{L}, \mathbf{U}]$ denotes the admissible hyper-rectangular parameter domain.

Algorithm overview. The NPBGO procedure operates in three stages: an initial space-filling design, followed by a minimum search phase and a maximum search phase. The detailed procedure is given in Algorithm 4.

Algorithm 4 Non-Parallel Bayesian Global Optimization (NPBGO)

```
Require: Objective function $Q(\boldsymbol{\xi})$, dimension $d$, bounds $[\mathbf{L}, \mathbf{U}]$
Ensure: Estimated bounds $Q_{\text{min}}$, $Q_{\text{max}}$
Stage 1: Initial Design
Generate $N_0 = 30$ samples via Latin Hypercube Sampling over $[\mathbf{L}, \mathbf{U}]$
Evaluate $Q(\boldsymbol{\xi}_i)$ for $i = 1, \dots, N_0$
Initialize sample database: $\mathcal{D} \leftarrow \{(\boldsymbol{\xi}_i, Q_i)\}_{i=1}^{N_0}$
Stage 2: Minimum Search
repeat
    Fit Gaussian process surrogate $\hat{Q}(\boldsymbol{\xi})$ with ARD squared-exponential kernel to $\mathcal{D}$
    Compute Expected Improvement for minimization:
        $\text{EI}_{\text{min}}(\boldsymbol{\xi}) = (Q^*_{\text{min}} - \hat{\mu})\, \Phi\left(\frac{Q^*_{\text{min}} - \hat{\mu}}{\hat{\sigma}}\right) + \hat{\sigma}\, \phi\left(\frac{Q^*_{\text{min}} - \hat{\mu}}{\hat{\sigma}}\right)$
    where $Q^*_{\text{min}} = \min(\mathcal{D})$, $\hat{\mu}$ and $\hat{\sigma}$ are the GP predictive mean and standard deviation
    Maximize $\text{EI}_{\text{min}}$ via Teaching-Learning-Based Optimization (TLBO) with population size $5d$ and $15d$ iterations
    Compute normalized improvement: $\Delta_j = \text{EI}_{\text{min}}(\boldsymbol{\xi}_{\text{new}})/(|Q^*_{\text{min}}| + 10^{-6})$
    if $\Delta_j < \kappa$ for $D$ consecutive iterations then
        Terminate minimum search
    else
        Evaluate $Q(\boldsymbol{\xi}_{\text{new}})$ and augment $\mathcal{D}$
    end if
until convergence
Stage 3: Maximum Search
Repeat Stage 2 with the Expected Improvement adapted for maximization:
    $\text{EI}_{\text{max}}(\boldsymbol{\xi}) = (\hat{\mu} - Q^*_{\text{max}})\, \Phi\left(\frac{\hat{\mu} - Q^*_{\text{max}}}{\hat{\sigma}}\right) + \hat{\sigma}\, \phi\left(\frac{\hat{\mu} - Q^*_{\text{max}}}{\hat{\sigma}}\right)$
using the same stopping criterion
return $Q_{\text{min}} = \min(\mathcal{D})$, $Q_{\text{max}} = \max(\mathcal{D})$
```

Algorithmic parameters. The stopping criterion uses a normalized improvement threshold $\kappa = 0.002$ with a delay counter $D = 2$, meaning the search terminates when two consecutive iterations fail to produce a normalized improvement exceeding κ. These values follow the defaults of the original implementation [5].

Integration with the finite element pipeline. In the present framework, the objective function $Q(\boldsymbol{\xi})$ is implemented as a MATLAB function handle that encapsulates the complete evaluation sequence: boundary perturbation, Laplace mesh deformation, mesh validity check, Poisson assembly and solve, and QoI extraction (Section 4.5). If the deformed mesh is invalid ($\min_e(2A_e^*) \leq 0$), the function returns a large penalty value ($Q = 10^{30}$) to discourage the optimizer from exploring parameter regions that lead to mesh inversion.

Appendix E

In-Memory Gmsh Mesh Generation Benchmark

This appendix documents the supplementary micro-benchmark used to quantify the overhead of external process invocation and file I/O in the remeshing baseline of the computational efficiency benchmark (Section 5.3.2).

Motivation

The MATLAB-based remeshing benchmark invokes Gmsh as a separate operating system process via `system()`, writes the generated mesh to a `.msh` file, and reads it back into MATLAB. This workflow includes costs that are not intrinsic to mesh generation itself: process creation, disk write, disk read, and file parsing. To isolate the contribution of these implementation-specific costs, the mesh generation step was repeated using the Gmsh Python API, which operates entirely in memory.

Setup

The micro-benchmark was implemented as a standalone Python script (`experiments/gmsh_inmemory_benchmark.py`) using the Gmsh Python API (version 4.15.1) with the following configuration:

- **Geometry:** Geometrically equivalent parametric plate-with-hole configuration, constructed using the OpenCASCADE (OCC) kernel. The MATLAB benchmark uses the Gmsh built-in kernel via `hole_template.geo`; the OCC kernel produces geometrically identical domains but may yield slightly different node counts due to differences in boundary discretization.
- **Plate dimensions:** 10×10, centered at the origin.
- **Nominal hole:** Semi-axes $a_0 = 2.0$, $b_0 = 1.5$, centered at $(0, 0)$.
- **Uncertain parameters:** $a = a_0(1 + 0.1\,U(-1, 1))$, $b = b_0(1 + 0.1\,U(-1, 1))$, $\Delta x = 0.2\,U(-1, 1)$, $\Delta y = 0.2\,U(-1, 1)$.
- **Mesh algorithm:** Frontal-Delaunay (algorithm 6), matching the `.geo` template.
- **Characteristic lengths:** $l_c \in \{0.8, 0.5, 0.3\}$.
- **Samples:** $N_s = 100$ per mesh level.

Timing Decomposition

For each sample, three components were timed separately:

1. **Geometry construction + synchronize:** OCC geometry definition (`addRectangle`, `addDisk`, `cut`) and `occ.synchronize()`.
2. **Mesh generation:** `gmsh.model.mesh.generate(2)`.
3. **Data extraction:** `getNodes()` and `getElements()`, which transfer the mesh data from the Gmsh engine to Python arrays.

No `.msh` file was written or read at any point. The entire pipeline operates in memory.

Reconstruction

The measured in-memory generation time replaces the external-process components (`tRem_gmsh` + `tRem_read`) from the MATLAB timing decomposition, while the downstream PDE components (boundary detection, assembly, Dirichlet enforcement, linear solve, and QoI extraction) are retained from the MATLAB measurements. The reconstructed in-memory remeshing time is:

$$t_{\text{remesh, in-mem}} = t_{\text{Python}} + t_{\text{setup}} + t_{\text{assembly}} + t_{\text{BC}} + t_{\text{solve}} + t_{\text{QoI}},$$

and the estimated in-memory speedup is:

$$S_{\text{in-mem}} = \frac{t_{\text{remesh, in-mem}}}{t_{\text{Laplace, total}}}.$$

This approach avoids a cross-platform solver comparison (MATLAB vs. Python) and instead isolates the single component, mesh generation overhead, that differs between external-process and in-memory invocation. The PDE-related timings, which are solver-dependent, remain consistently measured within the MATLAB environment.

Limitations

- The in-memory timings were measured in a different runtime environment (Python) than the MATLAB benchmark. While the Gmsh meshing engine is identical (same C++ library), minor differences in API binding overhead may exist.
- The OCC kernel was used instead of the built-in kernel, which may produce slightly different meshes. The node counts are comparable but not identical.
- Run-to-run variability of approximately 1–5 ms was observed due to operating system scheduling. The reported values represent averages over $N_s = 100$ samples.

Bibliography

[1] Pierre Alliez, Giuliana Ucelli, Craig Gotsman, and Marco Attene. Recent advances in remeshing of surfaces. Shape analysis and structuring, pages 53–82, 2008.

[2] Nicolas Barral, Edward Luke, and Frédéric Alauzet. Two mesh deformation methods coupled with a changing-connectivity moving mesh method for cfd applications. Procedia Engineering, 82:213–227, 2014.

[3] John T Batina. Unsteady euler algorithm with unstructured dynamic mesh for complex-aircraft aerodynamic analysis. AIAA journal, 29(3):327–333, 1991.

[4] Seung-Kyum Choi, Robert A Canfield, and Ramana V Grandhi. Reliability-based structural design. Springer, 2007.

[5] Chao Dang, Pengfei Wei, Matthias GR Faes, Marcos A Valdebenito, and Michael Beer. Interval uncertainty propagation by a parallel bayesian global optimization method. Applied Mathematical Modelling, 108:220–235, 2022.

[6] Jean Donea and Antonio Huerta. Finite element methods for flow problems. John Wiley & Sons, 2003.

[7] Jean Donea, Antonio Huerta, J-Ph Ponthot, and Antonio Rodríguez-Ferran. Arbitrary l agrangian–e ulerian methods. Encyclopedia of computational mechanics, 2004.

[8] Qiang Du, Vance Faber, and Max Gunzburger. Centroidal voronoi tessellations: Applications and algorithms. SIAM review, 41(4):637–676, 1999.

[9] David A Field. Laplacian smoothing and delaunay triangulations. Communications in applied numerical methods, 4(6):709–712, 1988.

[10] Luca Formaggia, Fabio Nobile, et al. A stability analysis for the arbitrary lagrangian eulerian formulation with finite elements. East-West Journal of Mathematics, 7:105–132, 1999.

[11] Alexander Forrester, Andras Sobester, and Andy Keane. Engineering design via surrogate modelling: a practical guide. John Wiley & Sons, 2008.

[12] MS Gadala and J Wang. A practical procedure for mesh motion in arbitrary lagrangian-eulerian method. Engineering with Computers, 14(3):223–234, 1998.

[13] Paul L George. Automatic mesh generation: applications to finite element methods. John Wiley & Sons, Inc., 1992.

[14] Christophe Geuzaine and Jean-François Remacle. Gmsh: A 3-d finite element mesh generator with built-in pre-and post-processing facilities. International journal for numerical methods in engineering, 79(11):1309–1331, 2009.

[15] Mohammad Amin Hariri-Ardebili, Siamak Sattar, and Christopher L Segura Jr. Beyond the sum: Multi-source uncertainty and its impact on structural fragility and risk. Engineering Structures, 343:121106, 2025.

[16] Cyrill W Hirt, Anthony A Amsden, and JL Cook. An arbitrary lagrangian-eulerian computing method for all flow speeds. Journal of computational physics, 14(3):227–253, 1974.

[17] Thomas JR Hughes, John A Cottrell, and Yuri Bazilevs. Isogeometric analysis: Cad, finite elements, nurbs, exact geometry and mesh refinement. Computer methods in applied mechanics and engineering, 194(39-41):4135–4195, 2005.

[18] Thomas JR Hughes, Wing Kam Liu, and Thomas K Zimmermann. Lagrangian-eulerian finite element formulation for incompressible viscous flows. Computer methods in applied mechanics and engineering, 29(3):329–349, 1981.

[19] Amaury Johnen, J-F Remacle, and Christophe Geuzaine. Geometrical validity of curvilinear finite elements. Journal of Computational Physics, 233:359–372, 2013.

[20] Donald R Jones, Matthias Schonlau, and William J Welch. Efficient global optimization of expensive black-box functions. Journal of Global optimization, 13(4):455–492, 1998.

[21] Patrick M Knupp. Achieving finite element mesh quality via optimization of the jacobian matrix norm and associated quantities. part ii—a framework for volume mesh optimization and the condition number of the jacobian matrix. International Journal for numerical methods in engineering, 48(8):1165–1185, 2000.

[22] Rolf Krause and Patrick Zulian. A parallel approach to the variational transfer of discrete fields between arbitrarily distributed unstructured finite element meshes. SIAM Journal on Scientific Computing, 38(3):C307–C333, 2016.

[23] Nataly A Manque, Jan Liedmann, Franz-Joseph Barthold, Marcos A Valdebenito, and Matthias GR Faes. Interval isogeometric analysis for coping with geometric uncertainty. Computer Methods in Applied Mechanics and Engineering, 437:117773, 2025.

[24] M. J. Martin-Burgos, D. González-Juárez, and E. Andrés-Pérez. A laplacian mesh deformation technique for simulation-driven design optimization. In ECCOMAS Congress 2016 – VII European Congress on Computational Methods in Applied Sciences and Engineering, Crete Island, Greece, 2016.

[25] Piotr Minakowski and Thomas Richter. Finite element error estimates on geometrically perturbed domains. Journal of Scientific Computing, 84(2):30, 2020.

[26] David Moens and Dirk Vandepitte. A survey of non-probabilistic uncertainty treatment in finite element analysis. Computer methods in applied mechanics and engineering, 194(12-16):1527–1555, 2005.

[27] Ramon E Moore. Methods and applications of interval analysis. SIAM, 1979.

[28] U Nackenhorst. The ale-formulation of bodies in rolling contact: Theoretical foundations and finite element approach. Computer methods in applied mechanics and engineering, 193(39-41):4299–4322, 2004.

[29] Habib N Najm. Uncertainty quantification and polynomial chaos techniques in computational fluid dynamics. Annual review of fluid mechanics, 41(1):35–52, 2009.

[30] Arnold Neumaier. Interval methods for systems of equations. Number 37. Cambridge university press, 1990.

[31] Matteo Pozzi, Giacomo Bonaccorsi, Hyunsun Alicia Kim, and Francesco Braghin. Robust structural optimization in presence of manufacturing uncertainties through a boundary-perturbation method. Structural and Multidisciplinary Optimization, 66(6):120, 2023.

[32] Gangan Prathap. The finite element method in structural mechanics: principles and practice of design of field-consistent elements for structural and solid mechanics, volume 24. Springer Science & Business Media, 2013.

[33] Carl Edward Rasmussen. Gaussian processes in machine learning. In Summer school on machine learning, pages 63–71. Springer, 2003.

[34] Christian P Robert, George Casella, and George Casella. Monte Carlo statistical methods, volume 2. Springer, 1999.

[35] Roman Sartorti, Wadhah Garhuom, and Alexander Düster. Remeshing and eigenvalue stabilization in the finite cell method for structures undergoing large elastoplastic deformations. Archive of applied mechanics, 94(9):2745–2768, 2024.

[36] Keith Stein, Tayfun Tezduyar, and Richard Benney. Mesh moving techniques for fluid-structure interactions with large displacements. J. Appl. Mech., 70(1):58–63, 2003.

[37] R. Stocki, P. Tauzowski, and M. Kleiber. Efficient sampling techniques for stochastic simulation of structural systems. Computer Assisted Mechanics and Engineering Sciences, pages 77–97, 2007.

[38] Timothy John Sullivan. Introduction to uncertainty quantification, volume 63. Springer, 2015.

[39] Tayfun E Tezduyar. Computation of moving boundaries and interfaces and stabilization parameters. International Journal for Numerical Methods in Fluids, 43(5):555–575, 2003.

[40] Thomas Wick. Fluid-structure interactions using different mesh motion techniques. Computers & Structures, 89(13-14):1456–1467, 2011.

[41] Zhibao Zheng, Marcos Valdebenito, Michael Beer, and Udo Nackenhorst. A stochastic finite element scheme for solving partial differential equations defined on random domains. Computer Methods in Applied Mechanics and Engineering, 405:115860, 2023.

[42] VS Ziel, H Bériot, O Atak, and Gwenael Gabard. Comparison of 2d boundary curving methods with modal shape functions and a piecewise linear target mesh. Procedia Engineering, 203:91–101, 2017.

[43] Olgierd C Zienkiewicz and Robert Leroy Taylor. The finite element method set. Elsevier, 2005.

Eidesstattliche Versicherung

(Affidavit)

Daher, Mohamad

Name, Vorname
(surname, first name)

222429

Matrikelnummer
(student ID number)

☐ Bachelorarbeit
(Bachelor's thesis)

☒ Masterarbeit
(Master's thesis)

Titel
(Title)

Efficient Treatment of Boundary Uncertainty in FEM via Mesh Deformation without Remeshing

Ich versichere hiermit an Eides statt, dass ich die vorliegende Abschlussarbeit mit dem oben genannten Titel selbstständig und ohne unzulässige fremde Hilfe erbracht habe. Ich habe keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie wörtliche und sinngemäße Zitate kenntlich gemacht. Die Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

I declare in lieu of oath that I have completed the present thesis with the above-mentioned title independently and without any unauthorized assistance. I have not used any other sources or aids than the ones listed and have documented quotations and paraphrases as such. The thesis in its current or similar version has not been submitted to an auditing institution before.

Dortmund, 17.03.2026

Ort, Datum
(place, date)

Unterschrift
(signature)

Belehrung:
Wer vorsätzlich gegen eine die Täuschung über Prüfungsleistungen betreffende Regelung einer Hochschulprüfungsordnung verstößt, handelt ordnungswidrig. Die Ordnungswidrigkeit kann mit einer Geldbuße von bis zu 50.000,00 € geahndet werden. Zuständige Verwaltungsbehörde für die Verfolgung und Ahndung von Ordnungswidrigkeiten ist der Kanzler/die Kanzlerin der Technischen Universität Dortmund. Im Falle eines mehrfachen oder sonstigen schwerwiegenden Täuschungsversuches kann der Prüfling zudem exmatrikuliert werden. (§ 63 Abs. 5 Hochschulgesetz - HG -).

Die Abgabe einer falschen Versicherung an Eides statt wird mit Freiheitsstrafe bis zu 3 Jahren oder mit Geldstrafe bestraft.

Die Technische Universität Dortmund wird ggf. elektronische Vergleichswerkzeuge (wie z.B. die Software „turnitin") zur Überprüfung von Ordnungswidrigkeiten in Prüfungsverfahren nutzen.

Die oben stehende Belehrung habe ich zur Kenntnis genommen:

Official notification:
Any person who intentionally breaches any regulation of university examination regulations relating to deception in examination performance is acting improperly. This offense can be punished with a fine of up to EUR 50,000.00. The competent administrative authority for the pursuit and prosecution of offenses of this type is the Chancellor of TU Dortmund University. In the case of multiple or other serious attempts at deception, the examinee can also be unenrolled, Section 63 (5) North Rhine-Westphalia Higher Education Act (*Hochschulgesetz, HG*).

The submission of a false affidavit will be punished with a prison sentence of up to three years or a fine.

As may be necessary, TU Dortmund University will make use of electronic plagiarism-prevention tools (e.g. the "turnitin" service) in order to monitor violations during the examination procedures.

I have taken note of the above official notification:*

Dortmund, 17.03.2026

Ort, Datum
(place, date)

Unterschrift
(signature)

***Please be aware that solely the German version of the affidavit ("Eidesstattliche Versicherung") for the Bachelor's/ Master's thesis is the official and legally binding version.**