Freie Universität Berlin

Adaptive Methods for the Darcy Equation with Stochastic Input Data

Dissertation zur Erlangung des Grades eines Doktors der Naturwissenschaften (Dr. rer. nat.) am Fachbereich Mathematik und Informatik der Freien Universität Berlin

Johannes Neumann

Berlin, 31. März 2016

1. Gutachter	**Prof. Dr. Volker John** Freie Universität Berlin Weierstraß-Institut Berlin
2. Gutachterin	**Dr. Danielle Hilhorst, Directrice de recherche** Université Paris-Sud 11 Centre national de la recherche scientifique
Disputation am	27. Oktober 2016

Weierstraß-Institut für
Angewandte Analysis und Stochastik

Adaptive Methods for the Darcy Equation with Stochastic Input Data

Johannes Neumann

Berlin, March the 31st, 2016

Parts of this thesis have been previously published in

[EMN16] Martin Eigel, Christian Merdon, and Johannes Neumann. An adaptive multilevel monte carlo method with stochastic bounds for quantities of interest with uncertain data. *SIAM/ASA Journal on Uncertainty Quantification*, 4(1):1219–1245, 2016,

[ABE+15a] Felix Anker, Christian Bayer, Martin Eigel, Marcel Ladkau, Johannes Neumann, and John G.M. Schoenmakers. SDE based regression for random PDEs. *WIAS Preprint*, 2192, 2015,

[ABE+15b] Felix Anker, Christian Bayer, Martin Eigel, Johannes Neumann, and John G.M. Schoenmakers. Adaptive SDE based interpolation for random PDEs. *WIAS Preprint*, 2200, 2015.

Abstract

Partial differential equations with stochastic data are used to model insecurities and insufficient knowledge of the properties in the physical domain. In this thesis, the stationary Darcy equation is used as an example as it is only feasible to measure the subsurface domain properties in a limited amount of locations. As such the permeability coefficient is assumed to be stochastic. Sampling methods constitute a major class of algorithms to numerically approximate the solution of such models as they integrate well into existing discrete finite element frameworks and theoretical results can be achieved without regard to the underlying spacial discretization.

Furthermore, the main advantage of the Monte Carlo method is the stochastic dimension independent convergence rate of $1/2$ with respect to the number of samples. This fundamental result from the central limit theorem cannot be improved upon so the computational cost of the samples needs to be reduced. A recent approach is the multilevel Monte Carlo method, which computes the majority of samples on coarser grids in some mesh hierarchy and thus reducing their cost.

We propose mesh adaptive algorithms similar to those in the deterministic setting to further reduce the cost of individual samples. We propose solutions to emerging problems in the determination of optimal control parameters in the Monte Carlo and multilevel Monte Carlo method when used with non uniform meshes in the form of heuristical a posteriori approximations. Additionally, bounds are presented that control the error with a prescribed probability.

For a fully dimension independent approach for higher-dimensional physical domains with rough random input fields we explore an approach using a model representation with some stochastic ordinary equation. Adaptive methods are presented that allow for independent stochastic and spatial adaptivity together with heuristics for optimal control parameters.

I propose we leave math to the machines and go play outside.

Calvin

Acknowledgments

Before we go on with the topic at hand I want to take the time to thank all the people that have helped me to reach this point in my career. First and foremost I want to thank Prof. Dr. Volker John for his guidance and support while I was writing this thesis. His patience is truly remarkable.

I also want to thank all my colleagues at the Weierstrass Institute, in particular Dr. Martin Eigel as he was the major influx of new ideas for this thesis and opened great opportunities for myself. Furthermore, I want to thank my friend Dr. Christian Merdon, whom I know since my time at Humboldt-Universität. It is always a pleasure to work with him. Additionally, I want to thank Ulrich Wilbrandt as he was often able to give insightful thoughts and unconventional suggestions.

Of course I don't want to omit all the people that taught me everything that I know including my former colleagues at Humboldt-Universität zu Berlin, my teachers at Heinrich-Hertz-Gymnasium, my family, and my friends.

Finally, I want to thank my friends Christian B., Christian M., Sara, Stefanie, and Verina who thoroughly proofread this text.

★

Contents

List of Symbols

#	$\cdot^+, \cdot^-$	composite of the corresponding elements from the primal and dual problem linked with weight α, e.g. $u^{\pm} = \alpha u \pm \frac{z}{\alpha}$, cf. Lemma 3.6
	Δ	Laplace operator
	∇	gradient
	$\|\cdot\|$	shortcut for the L^2 norm $\|\cdot\|_{L^2(D)}$
	$\|\cdot\|_{L^2(D)}$	L^2 norm in D
	$\|\cdot\|_{H^1(D)}$	H^1 norm in D
	$\|\!\|\cdot\|\!\|_*$	dual norm, cf. Definition 3.2
	$\|\!\|\cdot\|\!\|_\omega$	energy norm, cf. Equation (1.5)
A	α	**a**) weight between the primal and the dual problem for the guaranteed bounds, cf. Lemma 3.6 **b**) finite element convergence rate, cf. Section 3.4 **c**) nodal interpolation convergence rate, cf. Section 4.3
	α_{opt}	approximation of the optimal α in the deterministic bounds, cf. Remark 3.14
	a	bilinear form in the mixed context, cf. Equation (1.3)
B	b	**a**) left-hand side of the weak formulation, cf. Equation (1.2) **b**) operator in the stochastic representation of the model problem, cf. Section 4.1
	b_ω	stochastic version of the left-hand side in the weak formulation, cf. Equation (1.2)
C	c_1, c_2	parameter functions in the stochastic representation, cf. Section 4.1

	c_η	overestimation factor for some error estimator η
	$\tilde{c}_\eta$	approximation of the overestimation factor for some error estimator η
	c_N	constant in the equations for the number of needed samples
	c_N^ℓ	constant in the equations for the number of needed samples on level ℓ, cf. Section 4.3
	$\dot{c}_N^{L+1}$	extrapolation of the constant c_N^{L+1} to level $L+1$ with information from smaller levels
	C_g	weight function in the goal, cf. Experiment 5.1
	C_P	Poincaré constant in the Poincaré inequality, cf. Lemma 3.7
	$\mathcal{C}_\ell$	computational cost for one finite element solution sample on level ℓ
	$\mathcal{C}_T$	average number of samples on triangle T, cf. Experiment 5.4
	C	covariance function
D	δ	balancing parameter between the sampling error and the time discretization as well as interpolation error, cf. Section 4.3
	Δt	discrete time step, cf. Section 4.1
	D	physical domain
	dist	Euclidean distance
E	$\varepsilon_\cdot$	upper bound for the Monte Carlo estimation error, cf. Lemma 3.15
	$\varepsilon_\cdot^*$	residual sequence for the upper bound of the Monte Carlo estimation error, cf. Lemma 3.15
	ε_η	prescribed tolerance for the pathwise refinement, cf. Section 3.3
	η^+, η^-	upper and lower bounds for the error in the quantity of interest

$\eta^{\oplus}, \eta^{\ominus}$	upper and lower probable bounds for the error in the quantity of interest
η^{loc}	local element-wise error indicator
$\tilde{\eta}^{\mathrm{loc}}$	local element-wise error indicator for the dual problem
$\tilde{\eta}^{\mathrm{loc}}$	local element-wise error indicator for the dual problem
η_h	generic error estimator on the mesh $\mathcal{T}_h$
η_ℓ	generic error estimator on the level ℓ
η_L	generic error estimator on the level L
η_{MS}	local error indicator for the quantity of interest as in [MS09], cf. Section 3.3
η_{OP}	local error indicator for the quantity of interest as in [OP01], cf. Section 3.3
$\eta^{\pm}_{\mathrm{rel}}$	relative efficiency of the guaranteed bounds, cf. Experiment 5.1
η^{Q}_{rel}	relative efficiency of the error estimator, cf. Experiment 5.1
e_k	expected finite element error on level k
$\tilde{e}_k$	approximation of the error on level k
$\dot{e}_{L+1}$	extrapolation of the error to level $L+1$
$\mathsf{e}_{\mathrm{rel}}$	relative error in the quantity of interest, cf. Experiment 5.1
$\tilde{\mathsf{e}}_{\mathrm{rel}}$	reliable relative error in the quantity of interest, cf. Experiment 5.1
E_{rel}	relative error of the probable bounds, cf. Experiment 5.1
$\mathbf{E}[\,\cdot\,]$	Expected value of some random variable, cf. Definition 1.3
$\mathbf{E}^{\mathrm{M}}_{N}[\,\cdot\,]$	Generic estimator for the expected value with the approximation parameter N
$\mathbf{E}^{\mathrm{MC}}_{N}[\,\cdot\,]$	Monte Carlo estimator of some random variable with N number of samples, cf. Definition 2.1
$\mathbf{E}^{\mathrm{ML}}_{\boldsymbol{N}}[\,\cdot\,]$	Multilevel Monte Carlo estimator of some random variable with $\boldsymbol{N} \in \mathbb{R}^L$ number of samples on L levels, cf. Definition 2.6
$\mathbf{E}^{\mathrm{MS}}_{M,N}[\,\cdot\,]$	SDE estimator in point $x \in D$, cf. Section 4.1

	$\mathbf{E}^{\mathrm{MS}}_{\boldsymbol{M},\boldsymbol{N}}[\,\cdot\,]$	SDE estimator in the whole domain using nodal interpolation, cf. Section 4.1
F	f	weight of the right-hand side of the model problem, sinks and wells
	F	right-hand side of the weak formulation, cf. Equation (1.2)
	$\mathcal{F}$	set of events, cf. Definition 1.1
G	γ	upper bound for the dual norm of the residual, cf. Theorem 3.9
	g	weight for the quantity of interest
H	h	mesh element size
	h_T	diameter of triangle T
	$H^1(D)$	Hilbert space
	$H^1_0(D)$	Hilbert space with empty trace, i.e. $\mathrm{tr}(v) \equiv 0$
I	I_c	confidence interval, cf. Experiment 5.1
	I_n	identity matrix of size n
	$\mathcal{I}_h$	nodal interpolation operator for some mesh $\mathcal{T}_h$
K	κ	permeability field
	$\kappa_{\max,T}$	maximum of κ on some triangle T
	$\kappa_{\max,E}$	maximum of κ on some edge E
	K	node of the triangulation $\mathcal{T}$
L	ℓ	index of a level in the mesh hierarchy
	L	**a**) number of levels in a mesh hierarchy **b**) left-hand side in the weak formulation, cf. Section 1.3 **c**) operator in the stochastic representation, cf. Section 4.1

M	M	index for the time discretization
	$\boldsymbol{M}$	vector of indices for the time discretization
	$\mathcal{M}$	set of edges selected for refinement
	mid	midpoint of T
	MC	Monte Carlo
	ML	multilevel Monte Carlo
N	ν_ℓ^i	vertex/node i from mesh $\mathcal{T}_\ell$
	n_E	normal vector for some edge E
	N	number of samples
	N_η	number of samples on each level of the adaptive refinement algorithm, cf. Section 3.3
	N_{init}	initial number of samples to kick-off multilevel Monte Carlo
	$N_{\min}$	minimum number of samples to use
	$\boldsymbol{N}$	vector of number of samples
	$\mathcal{N}(T)$	vertices/nodes of the triangle T
	$\mathcal{N}_h$	vertices/nodes in mesh $\mathcal{T}_h$
	$\mathcal{N}_{\mu,\sigma^2}$	normal distribution with parameters μ and σ
	$\mathcal{N}_{0,1}$	standard normal distribution
	`ndof`	number of degrees of freedom
O	ω	event from the sample space Ω
	Ω	sample space, cf. Definition 1.1
	osc	oscillations, cf. Definition 3.1
P	∂D	boundary of the domain D
	Φ	cumulative distribution function of the standard normal distribution, cf. Lemma 3.15
	p	Darcy flux

	p	prescribed probability
	$\mathsf{p}^*, \mathsf{p}^+, \mathsf{p}^-$	prescribed vectors of auxiliary probabilities for the probable bounds $\eta^{\oplus}$ and $\eta^{\ominus}$, cf. Corollary 3.17
	$\mathbf{p}^*, \mathbf{p}^+, \mathbf{p}^-$	prescribed auxiliary probabilities for the probable bounds $\eta^{\oplus}$ and $\eta^{\ominus}$, cf. Corollary 3.19
	p_{RT}	discrete Raviart–Thomas finite element solution for the Darcy flux
	$\hat{p}_{RT}$	discrete Raviart–Thomas finite element solution for the Darcy flux with right-hand side $f_{\mathcal{T}}$
	P	probability measure, cf. Definition 1.1
	$\mathbf{P}[\,\cdot\,]$	probability
Q	q	flux of the dual problem
	q_{RT}	discrete Raviart–Thomas finite element solution for the flux of the dual problem
	Q	quantity of interest
R	r	control function in the upper and lower bounds for the quantity of interest
	r_g	radius of the support in the goal weight, cf. Experiment 5.1
	Res	residual, cf. Definition 3.2
S	σ	operator in the stochastic representation of the model problem, cf. Section 4.1
T	τ	stopping time, cf. Section 4.1
	θ	bulk criterion parameter, cf. Equation (3.14)
	$\cdot_T$	mean function on the triangle T, cf. Definition 3.1
	$\cdot_{\mathcal{T}}$	element-wise mean function for the triangulation $\mathcal{T}$, cf. Definition 3.1
	t	time index for random processes, cf. Section 4.1

	t_m	discrete time index, cf. Section 4.1
	T	**a**) triangle of some triangulation **b**) upper time limit in the stochastic representation, cf. Section 4.1
	$\mathcal{T}$	triangulation of the domain
	$\mathcal{T}_M$	time discretization, cf. Section 4.1
U	u	solution, describing pressure or hydraulic head
	u_h	discrete P_1 finite element solution
	u_h^M	nodal interpolation in D of the stochastic representation
	u_{CR}	discrete Crouzeix–Raviart finite element solution
	$\hat{u}_{CR}$	discrete Crouzeix–Raviart finite element solution with right-hand side $f_{\mathcal{T}}$
	u_D	Dirichlet boundary function
V	v_A	constructed helper function for the lower bound of the residual, cf. Lemma 3.11
	V	solution space
	V^*	dual space, cf. Definition 3.2
	V_h	discrete conforming polynomial finite element subspace of the solution space V
	V_{CR}	discrete nonconforming Crouzeix–Raviart finite element subspace of the solution space V
	V_{RT}	discrete Raviart–Thomas finite element subspace for the flux
	$\mathbf{Var}[\,\cdot\,]$	Variance of some random variable, cf. Definition 1.3
	$\mathbf{Var}_N^{\mathrm{MC}}[\,\cdot\,]$	Monte Carlo estimator for the variance, cf. Definition 2.5
	$\mathbf{Var}_{\boldsymbol{N}}^{\mathrm{ML}}[\,\cdot\,]$	Multilevel Monte Carlo estimator for the variance, cf. Definition 2.9
W	W	standard Brownian motion

	W_ℓ	quadratic multilevel term for the approximation of the variance, cf. Definition 2.9
X	ξ	lower bound for the dual norm of the residual, cf. Lemma 3.11
	X	generic random variable
	X_t, X_t^x	random process in point $x \in D$, cf. Section 4.1
	X_M	discretization of the random process X_t, cf. Section 4.1
Y	Y_ℓ	multilevel term in the telescoping sum, cf. Equation (2.3)
	Y_t, Y_t^x	random process in point $x \in D$, cf. Section 4.1
Z	z	solution of the dual problem, cf. Equation (1.4)
	z_h	discrete polynomial finite element solution of the dual problem, cf. Equation (1.4)
	z_{CR}	discrete nonconforming Crouzeix–Raviart solution of the dual problem, cf. Equation (1.4)
	Z_ℓ°	multilevel terms for the error estimator, cf. Section 3.4
	Z_ℓ^+, Z_ℓ^-	multilevel terms for the upper and lower bounds, cf. Corollary 3.19
	Z_t, Z_t^x	random process in point $x \in D$, cf. Section 4.1

1 Introduction

1.1 Motivation

One of the main reasons to develop mathematical and numerical tools is the prediction of physical phenomena in various sciences and engineering such as ground water flow, electrical engineering and so forth. For this work, we will consider the Darcy Equation which was conceived experimentally by the French engineer Henry Philibert Gaspard Darcy (1803 – 1858). It describes the flow of fluids through porous media and finds frequent application in the prediction of groundwater flow through aquifers. A modern derivation of Darcy's law is achieved by homogenization of the Navier–Stokes equations.

The general approach in mathematical predictions is to measure real world data which is then processed to generate input data for the model at hand. Numerical evaluation of this model leads to an approximate solution which in turn is used to draw conclusions and initiate necessary actions (Figure 1.1). Inevitably, each

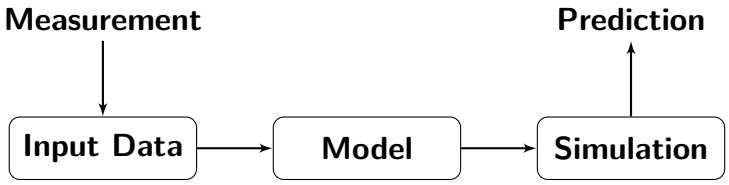


Figure 1.1: From measurement to prediction

of these steps introduces errors. The measurements are inaccurate by definition and often insufficient to provide all necessary input data. As an example, the input data for ground water flow simulations include material properties of the whole three-dimensional domain. Measurements in this area are usually very expensive and only feasible in a few selected spots. As a result, the model input data is only interpolated and thus a rough approximation of the true information. The model itself is by definition a simplified representation of the physical system. Only the most relevant properties and principles are rendered. The missing details result in inaccuracies of the model and, hence, introduce some defect. Further errors are introduced by the numerical simulation of the model and in the post-processing of the simulation result.

In order to obtain reliable results, it is a major challenge to model, control and quantify the errors in this chain. Major advancements in computational power now provide the tools to apply stochastic approaches to several physical phenomena where this has been deemed impossible so far. In the ground water flow example this

includes random fields as model inputs for the material properties of the domain in order to represent the lack of information thereof.

Two new challenges arise in this context. First, the random field needs to be modeled and discrete representations need to be constructed. Then, numerical methods need to be developed or adapted from deterministic models for the computation of the desired quantities. In practice, measurements are taken and Kriging, also known as Gaussian process regression or Wiener–Kolmogorov prediction, is used to approximate the model parameters of the random field such as its mean and correlation function.

These two parameters suffice to characterize a Gaussian random field [Fel68]. As the oscillations in a physical domain are usually much stronger, a log-normal random field is often used as the model. This is a random field of which the logarithm behaves like a Gaussian random field and thus can be characterized as above.

For the numerical computation, a couple of different approaches exist which vary greatly in their performance depending on the characteristics of the problem at hand. An overview of the different methods and their advantages as well as drawbacks are given in [Kee04] and [LK10].

Series expansions can be used to discretize the stochastic domain. Orthogonal decompositions lead to the polynomial chaos methods [GS90, XK02, XK03] using polynomials, like Hermite or Lagrangian, proper orthogonal decompositions or similar approaches. The stochastic Galerkin method [GS93, BTZ04, Kee04, MK05] employs the finite element idea to the stochastic domain. However, Piecewise polynomial ansatz functions result in an exponential dependence on the number of stochastic dimensions represented in the discretization. Sparse or global basis functions might mitigate this problem. The polynomial chaos methods do not exhibit this behavior but still have a polynomial dependence.

On the other hand, direct integration methods for the stochastic dimensions are insensitive to the number of dimensions. Monte Carlo methods use a set of randomly drawn samples to reduce the variance in the desired quantity of interest and thus, compute a discrete approximation. Multilevel Monte Carlo methods [CGST11, TSGU13, CST13] aim to reduce the cost for the computation of these samples through the usage of mesh hierarchies. An introduction in great detail is given in [Tec13].

Another way to control and improve the convergence of these methods is to choose the integration points in a deterministic pseudo random manner such as in the Quasi-Monte Carlo method. Finally, sparse integration methods aim for optimal integration points with the help of projection techniques [XH05, BNT07].

In this work, the focus lays on the multilevel Monte Carlo method as the random fields in the examples are constructed to resemble rough real world applications with short correlation lengths and high variances with strong oscillations. This setting asks for a high number of dimensions in the discretization of the stochastic space and thus, the direct integration methods are most suited for the task.

All direct integration methods have in common that samples need to be generated for the random field. Various methods exist with different computational complexity. As the sampling of random fields is not the focus of this thesis, only some are presented later on.

This thesis consists of six chapters. The rest of Chapter 1 concerns special notations with respect to the stochastic setting followed by a description of the model problem which is used to illustrate the ideas and methods in the remaining chapters.

The direct integration methods considered in this thesis are introduced in Chapter 2. It concludes with an overview of some random field construction procedures of which some are used in the numerical simulations.

Upper and lower bounds for the control of the error in the numerical computations in the deterministic as well as the stochastic setting are presented in Chapter 3. It also includes adaptive algorithms for the reduction of the computational costs and heuristics for the approximation of optimal parameters in the methods.

An alternative approach for the discretization of the physical domain and the probability space is given in Chapter 4. There, a fully stochastic representation of the stochastic problem is presented together with an adaptive interpolation algorithm and parameter heuristic therefor.

In Chapter 5 numerical simulations confirm the effectiveness of the discussed methods and the stability of the heuristics. Chapter 6 gives an outlook on future improvements, achievements and challenges.

1.2 Notation

Before we can continue with the problem definition, it is necessary to introduce basic stochastic terminology. In order to clarify the notation, mathematical objects from the stochastic domain are usually written in sans-serif-style like P and p whereas otherwise serif font is used like P and p. A notable exception is the sample space

Ω which in this work does not represent the spacial domain D as is customary in works with a solely deterministic setting. Furthermore, stochastic operators such as the moments defined below are rendered in bold sans serif fonts such as $\mathbf{E}[\,\cdot\,]$ and $\mathbf{Var}[\,\cdot\,]$. This convention is extended to the approximating estimators of those operators depicted as $\mathbf{E}_N^{\mathrm{M}}[\,\cdot\,]$ and $\mathbf{Var}_N^{\mathrm{M}}[\,\cdot\,]$.

Definition 1.1 A **probability space** is a triplet $(\Omega, \mathcal{F}, \mathsf{P})$ where

1. the **sample space** Ω is a non-empty set,
2. the set of **events** $\mathcal{F} \subseteq \mathcal{P}(\Omega)$ is a sigma algebra with
   a) $\Omega \in \mathcal{F}$,
   b) $\mathcal{F}$ is closed under complements ($A \in \mathcal{F} \implies \Omega \backslash A \in \mathcal{F}$),
   c) $\mathcal{F}$ is closed under countable unions, ($A_i \in \mathcal{F}$ for $i \in \mathbb{N} \Rightarrow \bigcup_{i\in\mathbb{N}} A_i \in \mathcal{F}$),
3. the **probability measure** $\mathsf{P} : \mathcal{F} \to [0,1]$ is a function with
   a) $\mathsf{P}(\Omega) = 1$,
   b) P is countably additive, ($A_i \in \mathcal{F}$ for $i \in \mathbb{N}$, $A_i \cap A_j = \emptyset$ for $i \neq j$ $\implies \mathsf{P}(\dot{\bigcup}_{i\in\mathbb{N}} A_i) = \sum_{i\in\mathbb{N}} \mathsf{P}(A_i)$).

Definition 1.2 Consider the probability space $(\Omega, \mathcal{F}, \mathsf{P})$. Now, $X : \Omega \to E$ is called a **random variable** if X is measurable with respect to P for some measurable space E. Usually it is $E = \mathbb{R}^d$ for $d > 0$. For $E \neq \mathbb{R}$ the function X is also called a random element.

Definition 1.3 For some random variable X and its probability space $(\Omega, \mathcal{F}, \mathsf{P})$ the **expected value** or **mean** $\mathbf{E}[X]$ is defined as

$$\mathbf{E}[X] = \int_\Omega X \,\mathrm{d}\mathsf{P} = \int_\Omega X(\omega)\,\mathsf{P}(\mathrm{d}\omega) = \int_\Omega X(\omega)\mathsf{P}(\omega)\,\mathrm{d}\omega.$$

It is also known as the first raw moment. The second central moment of X is called the **variance** and is defined as

$$\mathbf{Var}[X] = \mathbf{E}\Big[(X - \mathbf{E}[X])^2\Big]$$

which can be expanded to $\mathbf{Var}[X] = \mathbf{E}[X^2] - \mathbf{E}[X]^2$.

Definition 1.4 Consider some stochastic operator $\mathbf{E}[X]$ for the random variables $X : \Omega \to E$ defined in the probability space $(\Omega, \mathcal{F}, \mathsf{P})$. An **estimator** $\mathbf{E}_N^{\mathrm{M}}[\,\cdot\,]$ for this operator is a measurable function $\mathbf{E}_N^{\mathrm{M}}[\,\cdot\,] : \mathcal{F} \to E$. By definition, every estimator is itself a random variable. An estimator is called **convergent** if there exists a sequence

$(N_i)_{i\in\mathbb{N}}$ such that for all $\varepsilon > 0$

$$\lim_{i\to\infty} \mathbf{P}\Big[\Big|\mathbf{E}_{N_i}^{\mathrm{M}}[X] - \mathbf{E}[X]\Big| < \varepsilon\Big] = 1.$$

Such a sequence is called consistent. It is strongly consistent, if it converges almost surely to the true value $\mathbf{E}[X]$.

Definition 1.5 Let X be a random variable over $(\Omega, \mathcal{F}, \mathrm{P})$. The **bias** of a stochastic estimator $\mathbf{E}_N^{\mathrm{M}}[X]$ of the true value $\mathbf{E}[X]$ is defined as

$$\beta \coloneqq \mathbf{E}\Big[\mathbf{E}_N^{\mathrm{M}}[X]\Big] - \mathbf{E}[X]\,.$$

The estimator is called **unbiased** if $\beta = 0$.

1.3 The Darcy Model Problem

Steady-state subsurface flow in saturated conditions is described by the continuity equation as well as Darcy's law for flows in porous media. For some domain D in $\mathbb{R}^2$ or $\mathbb{R}^3$ these are

$$\begin{aligned} \operatorname{div} p &= f && \text{mass conservation,} \\ p &= -\kappa\nabla u && \text{Darcy law} \end{aligned}$$

with appropriate boundary conditions. The right-hand side $f \in L^2(D)$ describes sinks and wells in the domain, κ with $\kappa_{\min} < \kappa < \kappa_{\max}$ for $0 < \kappa_{\min}, \kappa_{\max} < \infty$ gives the permeability in every point of the domain. In general, the latter might be a tensor but for the purpose of simplicity it is assumed to be isotropic from here on. In this work, we set homogeneous Dirichlet boundary conditions in order to simplify the notation later on. Inhomogeneous Dirichlet and Neumann boundaries will alter the spaces in the variational formulations and introduce additional boundary integrals as well as some higher order terms in some of the inequalities. The combination of the above equations leads to the Darcy model problem

$$\begin{aligned} \operatorname{div}(-\kappa\nabla u) &= f && \text{in } D, \\ u &= 0 && \text{on } \partial D. \end{aligned} \tag{1.1}$$

The corresponding weak formulation seeks the solution $u \in H_0^1(D) =: V$ with $b(u,v) = F(v)$ for all $v \in V$ where

$$\begin{aligned} b(u,v) &= \int_D \kappa \nabla u \cdot \nabla v \, \mathrm{d}x, \\ F(v) &= \int_D f v \, \mathrm{d}x. \end{aligned} \tag{1.2}$$

The Hilbert space $H_0^1(D)$ contains all $v \in H^1(D)$ with $\mathrm{tr}(v) \equiv 0$. As the bilinear form b is continuous and weakly coercive for κ satisfying the above conditions, the Babuška–Lax–Milgram theorem [Bab71] states solvability and uniqueness of the solution, the so called Riesz representation.

In order to compute the solution numerically, the infinite-dimensional space V needs to be restricted to some finite-dimensional subspace $V_h \subset V$. For that purpose, the domain D is subdivided by some triangulation $\mathcal{T}_h$. Now, we seek the discrete solution u_h in the subspace

$$V_h = P_k(\mathcal{T}_h) \cap \mathcal{C}^0(D)$$

where $P_k(\mathcal{T}_h)$ denotes the space of piecewise polynomial functions of maximal degree $k > 0$, that is for each $v \in P_k(\mathcal{T}_h)$ it is $v|_T \in P_k(T)$ for all $T \in \mathcal{T}_h$. Nodal hat functions form a basis for the P_1 space. These elements are also called Lagrange or Courant elements. Additionally, we will also consider nonconforming Crouzeix–Raviart elements, in the sense that these form a basis spanning a space V_{CR} which is not a subspace of V. The Crouzeix–Raviart approximation space is defined as the piecewise linear functions that are continuous across the midpoints of all edges $\mathcal{E}(\mathcal{T}_h)$ of the triangulation, that is

$$\begin{aligned} V_{CR}(\mathcal{T}_h) = \{ v_h \mid\ & v_h|_T \in P_1(T) \text{ for all } T \in \mathcal{T}, \\ & v_h \text{ is continuous in the midpoint of all faces} \}. \end{aligned}$$

Figure 1.2 shows illustrative depictions of a P_1 and a CR basis function. The functions in V_{CR} are not continuous in general. Indeed, functions can be discontinuous at the nodes $\mathcal{N}(\mathcal{T}_h)$. The space V_{CR} is called a nonconforming finite element space as it holds $V_{CR} \not\subseteq V$.

The introduction of the intermediate variable $p \coloneqq \kappa \nabla u$ and integration by parts of Problem (1.1) leads to the mixed formulation of the weak Problem (1.2) which allows to focus computational power onto the gradient of the solution and, additionally, can be used for a posteriori error control. It seeks the solution $(u,p) \in L^2(D) \times H(\mathrm{div}, D)$

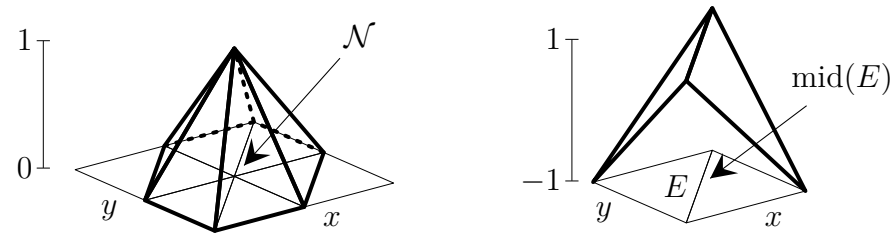


Figure 1.2: The P_1 hat basis function $v_{\mathcal{N}} : D \to [0,1]$ in V_h for node $\mathcal{N}$ of some mesh $\mathcal{T}_h$ (left) and the CR basis function $v_E : D \to [-1,1]$ for edge E in V_{CR} (right).

with $a\Big((u,p),(\mu,\rho)\Big) = L\Big((\mu,\rho)\Big)$ for all $(\mu,\rho) \in L^2(D) \times H(\mathrm{div}, D)$ where

$$\begin{aligned} a\Big((u,p),(\mu,\rho)\Big) &\coloneqq \int_D u \,\mathrm{div}\,\rho \;\mathrm{d}x + \int_D \kappa^{-1} p \cdot \rho \;\mathrm{d}x + \int_D \mathrm{div}\, p\mu \;\mathrm{d}x, \\ L\Big((\mu,\rho)\Big) &\coloneqq -\int_D f\mu \;\mathrm{d}x. \end{aligned} \tag{1.3}$$

Here, it is $H(\mathrm{div}, D) = \{v \in L^2(D;\mathbb{R}^2) \mid \mathrm{div}\, v \in L^2(D)\}$. The discrete subspace V_{RT} is spanned by Raviart–Thomas basis functions spanning a subspace of $L^2(D) \times H(\mathrm{div}, D)$ as follows

$$V_{RT} = \left\{v \in H(\mathrm{div}, D) \mid \forall_{T\in\mathcal{T}_h} \exists_{\alpha_T\in\mathbb{R}^2, \beta_T\in\mathbb{R}} : v|_T(x) = \alpha_T + \beta_T x\right\}.$$

The condition $v \in H(\mathrm{div}, D)$ for $v \in V_{RT}$ needs to be enforced either through the choice of the basis or during the solution process. [BC05] gives an overview of three different approaches.

Usually, the focus of interest is on some property of the solution which a post processing operator $Q : L^2 \to \mathbb{R}$ derives from the solution. In this work, these operators are assumed to be linear. Then, it can be characterized by some function $g : D \to \mathbb{R}$ as

$$Q(v) = \int_D gv \;\mathrm{d}x \qquad \text{for } v \in L^2(D) \tag{1.4}$$

and leads to the dual problem $b(u,v) = Q(v)$ for all $v \in V$. Similar to Problem (1.3), a mixed dual problem can be defined which leads to the dual mixed solution q.

In practical applications, problems arise with this model. The input data κ and f can only be measured in a rather limited number of discrete points in the domain. Thus, they are mostly unknown. The simulation on the other hand needs functions

with supports spanning the whole domain D, that is for each $x \in D$, a value for κ and f should be available. Therefore, the measured data needs to be interpolated in some way, for example using the weighted interpolation Kriging methods. The resulting uncertainty might then be described with stochastic means. Modeling the probabilities is a whole topic on its own and far beyond the scope of this thesis.

Hence, for the numerical simulations we assume a log-normal random field, which exhibits large local spikes and assumes a great range of values. This resembles the behavior observed in the real world and thus, allows testing the algorithms for their practical applicability. Log-normal random fields are defined such that taking its logarithm results in a Gaussian random field. In other words, they can be constructed by characterizing a normal field through its mean and covariance function [Fel68] and then applying the natural exponential function.

For the sake of clarity, we assume the permeability tensor κ to be random whereas the right-hand side f is deterministic. For some probability space $(\Omega, \mathcal{F}, \mathrm{P})$ we assume some log-normal random field $\kappa(\omega)$ for $\omega \in \Omega$ with mean $\kappa_{\min} < \bar{\kappa} < \kappa_{\max}$ such that the conditions for solvability from Section 1.3 are met almost surely. In this setting, the solutions $u(\omega)$ and $u_h(\omega)$ from Equation (1.2) become stochastic and it is the aim to compute the mean of the solution $\int_\Omega u \, \mathrm{dP}$ or the mean of the quantity of interest

$$\mathbf{E}[Q(u)] = \int_\Omega Q(u) \, \mathrm{d}\omega = \int_\Omega \int_D gu \, \mathrm{d}x \, \mathrm{d}\omega.$$

The solvability in the stochastic context follows from some extended Lax–Milgram theorem [GKSS13, CST13]. The bilinear form in Equation (1.2) is dependent on κ and thus, on ω which we will indicate as

$$b_\omega(u(\omega), v) = \int_D \kappa(\omega) \nabla u(\omega) \cdot \nabla v \, \mathrm{d}x.$$

This scalar product induces the energy norm for almost every $\omega \in \Omega$ as

$$\|\!|v|\!\|_\omega^2 = b_\omega(v, v). \tag{1.5}$$

Additionally, $\|v\|_{L^2(D)}^2 = \|v\|^2 = \int_D v^2 \, \mathrm{d}x$ defines the standard $L^2(D)$ norm and $\|v\|_{H^1(D)}^2 = \|v\|^2 + \|\nabla v\|^2$ is the H^1 norm for $v \in V$.

2 Sampling Based Methods

After the approximation by finite elements in the spatial dimension it is necessary to discretize the stochastic dimensions of the solution space. A variety of methods can be achieved. Among the most versatile are the sampling based methods as they combine robustness with ease of adaptation for the problem at hand. Implementation of these does not rely on details of the underlying spatial discretization and thus they can be used as extensions to existing deterministic solver codes.

In the following sampling based estimators for the expected value of some random variable $Q(u)$ as well as its discrete spatial approximation $Q(u_h)$ are presented. Later on these estimators are adopted for the partial differential equation at hand.

2.1 Monte Carlo

In order to approximate the quantity $Q(u)$ with some numerical solution, we have to integrate over the spatial domain D and the stochastic domain Ω simultaneously. The spatial domain is discretized by restricting the solution space V to some finite dimensional subspace V_h in which the discrete solution u_h is sought. In the following, stochastic integration is used to find the solution in Ω.

Definition 2.1 The **Monte Carlo estimator** with $N > 0$ samples for the expected value of some random variable $Q(u)$ is defined as

$$\mathbf{E}_N^{\mathrm{MC}}[Q(u)] \coloneqq N^{-1} \sum_{i=1}^{N} Q(u^i)$$

where $Q(u^i) = Q(u(\omega_i))$ are samples of the random variable $Q(u)$ and ω_i is drawn from Ω for $i = 1, \ldots, N$. This estimator converges and is unbiased.

Remark 2.2 In Definition 2.1, the convergence states that the estimator converges to the expected value of the estimated random variable, that is $\mathbf{E}_N^{\mathrm{MC}}[Q(u)] \to \mathbf{E}[Q(u)]$, for $N \to \infty$. This follows from Lemma 2.3 below. Additionally, the unbiasedness gives the identity of the expected value of the estimator and the expected value of the random variable as $\mathbf{E}\left[\mathbf{E}_N^{\mathrm{MC}}[Q(u)]\right] = \mathbf{E}[Q(u)]$.

All the samples in the Monte Carlo estimator are independently and identically distributed random variables and hence it holds for the variance of the estimator

$$\mathbf{Var}\left[\mathbf{E}_N^{\mathrm{MC}}[Q(u)]\right] = \mathbf{Var}\left[N^{-1} \sum_{i=1}^{N} Q(u^i)\right] = N^{-1}\, \mathbf{Var}[Q(u)]\,.$$

This equation shows that the Monte Carlo estimator, itself a random variable, is a variance reduction method as the estimator's variance is smaller than that of the estimand.

Lemma 2.3 *For some discrete spatial approximation $Q(u_h)$ of $Q(u)$ the decomposition of the mean square error holds as*

$$\mathbf{E}\left[\left(\mathbf{E}_N^{\mathrm{MC}}[Q(u_h)] - \mathbf{E}[Q(u)]\right)^2\right] = N^{-1}\,\mathbf{Var}[Q(u_h)] + \mathbf{E}[Q(u) - Q(u_h)]^2\,. \quad (2.1)$$

That is, the error can be decomposed into a term which is governed by the properties of the stochastic estimator and the variance of the discrete solution as well as a term which depends solely on the expected value of the spatial approximation error.

Proof. Expanding the left-hand side with the help of the properties in Remark 2.2 and the definition for the variance of $\mathbf{E}_N^{\mathrm{MC}}[Q(u_h)]$ gives the result

$$\begin{aligned}
&\mathbf{E}\left[\left(\mathbf{E}_N^{\mathrm{MC}}[Q(u_h)] - \mathbf{E}[Q(u)]\right)^2\right] \\
&\qquad = \mathbf{E}\left[\mathbf{E}_N^{\mathrm{MC}}[Q(u_h)]^2\right] + \mathbf{E}[Q(u)]^2 - 2\,\mathbf{E}\left[\mathbf{E}_N^{\mathrm{MC}}[Q(u_h)]\right]\mathbf{E}[Q(u)] \\
&\qquad = \mathbf{E}\left[\mathbf{E}_N^{\mathrm{MC}}[Q(u_h)]^2\right] - \mathbf{E}\left[\mathbf{E}_N^{\mathrm{MC}}[Q(u_h)]\right]^2 + \mathbf{E}[Q(u_h)]^2 \\
&\qquad\quad + \mathbf{E}[Q(u)]^2 - 2\,\mathbf{E}[Q(u_h)]\,\mathbf{E}[Q(u)] \\
&\qquad = \mathbf{Var}\left[\mathbf{E}_N^{\mathrm{MC}}[Q(u_h)]\right] + \mathbf{E}[Q(u) - Q(u_h)]^2 \\
&\qquad = N^{-1}\,\mathbf{Var}[Q(u_h)] + \mathbf{E}[Q(u) - Q(u_h)]^2\,.
\end{aligned}$$

□

Remark 2.4 The decomposition in Lemma 2.3 gives for some fixed number of samples N the limit

$$\mathbf{E}\left[\left(\mathbf{E}_N^{\mathrm{MC}}[Q(u_h)] - \mathbf{E}[Q(u)]\right)^2\right] \to N^{-1}\,\mathbf{Var}[Q(u)] \qquad \text{for } h \to 0$$

and for a fixed mesh size h this gives

$$\mathbf{E}\left[\left(\mathbf{E}_N^{\mathrm{MC}}[Q(u_h)] - \mathbf{E}[Q(u)]\right)^2\right] \to \mathbf{E}[Q(u_h) - Q(u)]^2 \qquad \text{for } N \to \infty.$$

In other words, it is necessary to balance the two parameters in order to achieve overall convergence.

Definition 2.5 The Monte Carlo approach for the **discrete approximation of the variance Var**$[Q(u)]$ is given by the approximation of the expected values in

Definition 1.3 with Monte Carlo estimators

$$\mathbf{Var}_N^{\mathrm{MC}}[Q(u)] \coloneqq \mathbf{E}_N^{\mathrm{MC}}\left[\left(Q(u) - \mathbf{E}_N^{\mathrm{MC}}[Q(u)]\right)^2\right].$$

2.2 Multilevel Monte Carlo

The main disadvantage of the Monte Carlo method in most situations is the slow convergence rate of $1/2$ with respect to N. To reduce the error by one half it takes four times the amount of samples. As an approach to overcome this obstacle the multilevel Monte Carlo algorithm has come into focus in the past years and was introduced to the finite element context by [CGST11] and [Tec13]. It is loosely related to the control variates variance reduction technique.

The main idea is to use a set of spatial discretizations with varying computational complexity and offload the majority of the computation on cheap approximations. For a given set of discretizations $Q(u_0), \dots, Q(u_L)$ on a set of meshes $\mathcal{T}_0, \dots, \mathcal{T}_L$ with increasing computational complexity and a set $\boldsymbol{N} = (N_0, \dots, N_L) \in \mathbb{N}^L$ the multilevel Monte Carlo estimator is defined by approximating the expected values in the telescoping sum

$$\mathbf{E}[Q(u_L)] = Q(u_0) + \sum_{\ell=1}^{L} \mathbf{E}[Q(u_\ell) - Q(u_{\ell-1})] \tag{2.2}$$

with individual Monte Carlo estimators resulting in the multilevel Monte Carlo estimator. From here on we will use the naming convention

$$Y_0 \coloneqq Q(u_0) \qquad \text{and} \qquad Y_\ell \coloneqq Q(u_\ell) - Q(u_{\ell-1}). \tag{2.3}$$

Definition 2.6 With the naming conventions from Equation (2.3) the **multilevel Monte Carlo estimator** is given by

$$\mathbf{E}_{\boldsymbol{N}}^{\mathrm{ML}}[Q(u_L)] \coloneqq \sum_{\ell=0}^{L} \mathbf{E}_{N_\ell}^{\mathrm{MC}}[Y_\ell] \tag{2.4}$$

for a given set of numbers of samples $\boldsymbol{N} = (N_0, \dots, N_L) \in \mathbb{N}^L$ on each level. It is convergent and unbiased.

Lemma 2.7 *Similar to Lemma 2.3 for the Monte Carlo estimator we get an error*

decomposition for the multilevel Monte Carlo estimator as follows

$$\mathbf{E}\left[\left(\mathbf{E}_{\boldsymbol{N}}^{\mathrm{ML}}[Q(u_L)] - \mathbf{E}[Q(u)]\right)^2\right] = \sum_{\ell=0}^{L} N_\ell^{-1}\,\mathbf{Var}[Y_\ell] + \mathbf{E}[Q(u_L) - Q(u)]^2\,. \tag{2.5}$$

Proof. First, we analyze the variance of the multilevel Monte Carlo estimator with the help of the properties from Remark 2.2 for the Monte Carlo estimator.

$$\begin{aligned}\mathbf{Var}\left[\mathbf{E}_{\boldsymbol{N}}^{\mathrm{ML}}[Q(u_L)]\right] &= \mathbf{Var}\left[\sum_{\ell=0}^{L}\mathbf{E}_{N_\ell}^{\mathrm{MC}}[Y_\ell]\right] = \sum_{\ell=0}^{L}\mathbf{Var}\left[\mathbf{E}_{N_\ell}^{\mathrm{MC}}[Y_\ell]\right] \\ &= \sum_{\ell=0}^{L} N_\ell^{-1}\,\mathbf{Var}[Y_\ell]\,.\end{aligned} \tag{2.6}$$

The rest of the proof is similar to the proof for Lemma 2.3

$$\begin{aligned}\mathbf{E}\left[\left(\mathbf{E}_{\boldsymbol{N}}^{\mathrm{ML}}[Q(u_L)]\right)^2\right] &= \mathbf{E}\left[\mathbf{E}_{\boldsymbol{N}}^{\mathrm{ML}}[Q(u_L)]^2\right] + \mathbf{E}[Q(u)]^2 - 2\,\mathbf{E}\left[\mathbf{E}_{\boldsymbol{N}}^{\mathrm{ML}}[Q(u_L)]\right]\mathbf{E}[Q(u)] \\ &= \mathbf{E}\left[\mathbf{E}_{\boldsymbol{N}}^{\mathrm{ML}}[Q(u_L)]^2\right] - \mathbf{E}\left[\mathbf{E}_{\boldsymbol{N}}^{\mathrm{ML}}[Q(u_L)]\right]^2 \\ &\quad + \mathbf{E}[Q(u_L)]^2 - 2\,\mathbf{E}[Q(u_L)]\,\mathbf{E}[Q(u)] + \mathbf{E}[Q(u)]^2 \\ &= \mathbf{Var}\left[\mathbf{E}_{\boldsymbol{N}}^{\mathrm{ML}}[Q(u_L)]\right] + \mathbf{E}[Q(u_L) - Q(u)]^2\,.\end{aligned}$$

Applying Equation (2.6) to the variance term concludes the proof. □

Remark 2.8 As the spatial error $|Y_\ell|$ vanishes with $\ell \to \infty$, $\mathbf{Var}[Y_\ell] \to 0$ follows. Thus, it is possible to choose $N_\ell \to 1$ for $\ell \to \infty$. This allows us to concentrate the greatest number of samples in the computationally cheapest subspaces V_ℓ.

In fact the optimal number of samples N_ℓ on level $\ell = 0, \ldots, L$ with respect to the overall computational cost is given up to some constant $c_N > 0$ as

$$N_\ell = c_N \sqrt{\mathcal{C}_\ell^{-1}\,\mathbf{Var}[Y_\ell]} \tag{2.7}$$

where $\mathcal{C}_\ell$ is the computational cost of one sample on that level.

Unlike with the Monte Carlo method, the multilevel approach requires the sampling of additional quantities in order to approximate the variance of some random variable.

Definition 2.9 Extending the naming conventions from Equation (2.3) define $W_0 \coloneqq Q(u_0)^2$ and $W_\ell \coloneqq Q(u_\ell)^2 - Q(u_{\ell-1})^2$. The multilevel Monte Carlo approach

for the **discrete approximation of the variance** $\mathbf{Var}[Q(u)]$ is given by

$$\mathbf{Var}_{\boldsymbol{N}}^{\mathrm{ML}}[Q(u_L)] \coloneqq \left| \sum_{\ell=0}^{L} \mathbf{E}_{N_\ell}^{\mathrm{MC}}[W_\ell] - \sum_{\ell=0}^{L} \mathbf{E}_{N_\ell}^{\mathrm{MC}}[Y_\ell]^2 - 2 \sum_{\substack{\ell,k=0 \\ \ell<k}}^{L} \mathbf{E}_{N_\ell}^{\mathrm{MC}}[Y_\ell]\, \mathbf{E}_{N_k}^{\mathrm{MC}}[Y_k] \right|.$$

Lemma 2.10 *The multilevel approximation* $\mathbf{Var}_{\boldsymbol{N}}^{\mathrm{ML}}[Q(u_L)]$ *of the variance in Definition 2.9 is a convergent estimator of the variance* $\mathbf{Var}[Q(u_L)]$.

Proof. Given the variance $\mathbf{Var}[Q(u_L)]$ as the starting point it holds

$$\begin{aligned}
\mathbf{Var}[Q(u_L)] &= \mathbf{E}\Big[(Q(u_L) - \mathbf{E}[Q(u_L)])^2\Big] = \Big|\mathbf{E}\Big[Q(u_L)^2\Big] - \mathbf{E}[Q(u_L)]^2\Big| \\
&= \left| \sum_{\ell=0}^{L} \mathbf{E}[W_\ell] - \left(\sum_{\ell=0}^{L} \mathbf{E}[Y_\ell] \right)^2 \right| \\
&= \left| \sum_{\ell=0}^{L} \mathbf{E}[W_\ell] - \sum_{\ell=0}^{L} \mathbf{E}[Y_\ell]^2 - 2 \sum_{\substack{\ell,k=0 \\ \ell<k}}^{L} \mathbf{E}[Y_\ell]\, \mathbf{E}[Y_k] \right|
\end{aligned}$$

as it is $\mathbf{Var}[Q(u_L)] > 0$. Approximating the expectations with Monte Carlo estimators gives the result. □

2.3 Random Fields

The Monte Carlo and multilevel Monte Carlo methods described in Section 2.1 and Section 2.2 necessitate sampled realizations of the random field κ. A Gaussian random field is fully determined by its mean function $\mathbf{E}[\kappa]$ and its covariance function C. In the application of the Darcy equation it is $\kappa > \kappa_{\min}$ almost everywhere for some fixed $\kappa_{\min} > 0$. Furthermore, the model requires larger variability and thus log-normal fields are better suited for the problem. These fields behave in logarithm like a Gaussian field.

Decomposition of the Covariance Matrix

A straightforward approach for the generation of random fields with prescribed mean $\mathbf{E}[\kappa]$ and covariance function C is a decomposition of the covariance matrix Σ. For

a set of random variables $\boldsymbol{y} = (y_1, \ldots, y_n) \in D$ with covariance function C the covariance matrix is defined as

$$\Sigma = \left(\mathsf{C}\left(y_i, y_j\right)\right)_{i,j=1}^{n}.$$

With its Cholesky decomposition $\Sigma = LL^T$ and the independent identically distributed standard Gaussian random variables $\boldsymbol{x} = (x_1, \ldots, x_n)$ we define $\boldsymbol{y} \coloneqq L\boldsymbol{x}$. Then $\boldsymbol{y}$ is normally distributed with

$$\mathbf{E}\left[\boldsymbol{y}\boldsymbol{y}^T\right] = \mathbf{E}\left[L\boldsymbol{x}\boldsymbol{x}^T L^T\right] = L\,\mathbf{E}\left[\boldsymbol{x}\boldsymbol{x}^T\right] L^T = LL^T = \Sigma.$$

Other decompositions besides Cholesky's are possible and might have advantages for example with Quasi Monte Carlo algorithms. The approximation of the covariance function is exact with respect to machine precision but the decomposition takes $\mathcal{O}(n^3)$ computations. Therefore, it is not feasible for finer domain discretizations or more spacial dimensions.

Karhunen–Loève Expansion

In order to avoid the dimensionality problem the Karhunen–Loève decomposition (KL) can be used [Kar47, Loè78, GS91]. The KL is a series expansion for the random field κ which is optimal regarding the mean square error. It is given by

$$\kappa\left(x, \omega\right) \coloneqq \mathbf{E}[\kappa]\left(x\right) + \sum_{i=1}^{\infty} \sqrt{\lambda_i} f_i \xi_i(\omega)$$

where $\mathbf{E}[\kappa]$ is the prescribed mean function, ξ_i are standard Gaussian random variables as well as λ_i and f_i with $i = 1, 2, \ldots$ are the eigenvalues and eigenfunctions of the following Fredholm equation

$$\int_D \mathsf{C}\left(x, y\right) f\left(x\right)\,\mathrm{d}x = \lambda f\left(y\right).$$

Given the eigenfunctions and eigenvalues in decreasing order of magnitude the KL series can be truncated at a given number of terms resulting in a random field with approximated covariance function.

Smooth Benchmark Field

Similar to the Karhunen–Loève expansion from Section 2.3, the following field will use a series expansion with the goal to provide an easy to construct random field with good adjustability for benchmarking problems and similar properties as the Karhunen-Loève expansion. The version of [EGSZ14] is modified to allow for values closer to real world cases. The coefficients are defined as

$$\begin{aligned} a_m(x) &\coloneqq \alpha_m \cos\left(2\pi\beta_1(m)x_1\right)\cos\left(2\pi\beta_2(m)x_2\right), \\ \alpha_m &\coloneqq Am^{-\sigma_\alpha}, \\ \beta_1(m) &\coloneqq m - k(m)\left(k(m)+1\right)/2, \\ \beta_2(m) &\coloneqq k(m) - \beta_1(m), \\ k(m) &\coloneqq \lfloor -1/2 + \sqrt{1/4 + 2m} \rfloor, \end{aligned}$$

with the parameters $\sigma_\alpha > 0$ and $0 < A < 1/\zeta(\sigma_\alpha)$. Here, ζ refers to the Riemann zeta function. We need to assure that the random field κ fulfills $\kappa(x,\omega) > 0$. Let $\varphi_m : \Omega \to [0,1]$ for $m = 0,\dots,t$ be uniformly distributed random variables. The construction of κ is well defined by

$$\kappa(x) \coloneqq \frac{c_\kappa}{\alpha_{\min}} \left(\sum_{m=0}^{t} a_m(x)\varphi_m + \alpha_{\min} \right) + \varepsilon_\kappa \tag{2.8}$$

for it is $\alpha_{\min} \coloneqq \sum_{m=0}^{t} \alpha_m \leq \min_{x\in D} \sum_{m=0}^{t} a_m(x)\varphi_m$ as $\cos(x) \in [-1,1]$ for $x \in \mathbb{R}$. The scaling factor $c_\kappa/\alpha_{\min}$ with $c_\kappa > 0$ and the offset $\varepsilon_\kappa > 0$ allow to control the codomain of the random field as it is $\kappa : D \to [\varepsilon_\kappa, 2c_\kappa + \varepsilon_\kappa]$.

Practical Methods

Besides the Karhunen-Loève expansion, the above methods are not feasible for practical applications. The synthetic benchmark field only resembles the desired properties on the unit square and the computational cost for the Cholesky decomposition prohibits the use for finer grids. In fact, the domain D might model regions of 10 km^2 and more while the random fields have to account for structures on a much smaller scale. As a result, the cost of sampling the field becomes critical. A comparison of different sampling techniques is given in [DSB11] where the circulant embedding method [DN97] gave suitable results with the smallest cost.

3 Probable Bounds, Adaptivity, and Heuristics

In real world applications the computation of the discrete solution u_h for some partial differential equation is usually only the first step in the estimation of some desired value. Post-processing of the solution leads to derived values, the so called quantities of interest. A major factor in these steps is the proliferation of computational errors from the discrete solution to these quantities, that is errors in the computed solution u_h get carried over and augmented during the computation of the quantity of interest.

Research in the past decades has provided suitable tools to quantify and confine the error in standard FEM analysis, which led to efficient, reliable and guaranteed error bounds in various norms. Furthermore, this was extended to different categories of quantities of interest. The parallelogram identity for the primal and dual problem has been introduced in [BM84a, BM84b, BM84c]. A good overview for optimal control with respect to quantities of interest is given in [AO00] originally published as [AO97], as well as in [BR01]. Introductions to adaptivity, error control and post-processing in general are provided in [BR03] and [Joh12].

In the model description in Section 1.3 we assume, for simplicity, linearity of the quantity of interest such that the dual problem with the right-hand side $Q(v) = \int_D gv \, \mathrm{d}x$ is well defined. This is necessary for the definition of the upper and lower bounds as well as for the goal-driven adaptivity presented in this chapter. Many nonlinear quantities can be linearized and thus fitted to this approach. However, all other parts of the theory, such as the multilevel Monte Carlo algorithm as well as the heuristics, do not depend on this property. No additional restrictions are imposed by the Monte Carlo estimators and any error estimator for the error in the bound $|Q(u) - Q(u_h)|$ will result in similar heuristics.

3.1 Deterministic Bounds for Quantities of Interest

The first step is to establish sample-wise deterministic bounds for the error in the finite element approximation. For this purpose we fix $\omega \in \Omega$ such that Equation (1.2) admits a solution. This is a valid assumption as it holds true almost surely in Ω. We follow the lines of [Mer13, CMN13].

Definition 3.1 The **oscillation** of a function f on a mesh $\mathcal{T}$ is a measure of how well the mesh can represent the roughness of that function. It is defined for

$h_T = \operatorname{diam}(T)$ as $\operatorname{osc}(f, \mathcal{T}) \coloneqq \sum_{T\in\mathcal{T}} \operatorname{osc}(f,T)^2$ where

$$\operatorname{osc}(f,T) \coloneqq h_T \left\| f - f_T \right\|_{L^2(T)} \quad \text{with} \quad f_T \coloneqq |T|^{-1} \int_T f \, \mathrm{d}x.$$

Finally define the **mean function** $f_\mathcal{T} : D \to \mathbb{R}$ on the whole domain as

$$f_\mathcal{T}|_T = f_T \qquad \text{for any } T \in \mathcal{T}.$$

Definition 3.2 The **residual** $\operatorname{Res} \in V^*$ of Equation (1.2) is a common starting point in a posteriori error control. For the discrete solution u_h and its stress $p_h = \kappa\nabla u_h$ it is defined for any $v \in V$ as

$$\operatorname{Res}(v) \coloneqq \int_D fv \, \mathrm{d}x - \int_D p_h \cdot \nabla v \, \mathrm{d}x \tag{3.1}$$

together with its **dual norm**

$$\left\|\!\left| \operatorname{Res} \right|\!\right\|_* = \sup_{v\in V\setminus\{0\}} \frac{|\operatorname{Res}(v)|}{\left\|\!\left| v \right|\!\right\|_\omega}. \tag{3.2}$$

Remark 3.3 The motivation of this ansatz is the equivalence of the residual and the energy norm which allows to control the error in the energy norm $\left\|\!\left| u - u_h \right|\!\right\|_\omega$ with means of the residual. In fact, it holds for every $v \in V$ the identity

$$\operatorname{Res}(v) = \int_D fv \, \mathrm{d}x - \int_D \kappa\nabla u_h \nabla v \, \mathrm{d}x \tag{3.3}$$

$$= \int_D \kappa\nabla u \nabla v \, \mathrm{d}x - \int_D \kappa\nabla u_h \nabla v \, \mathrm{d}x = b_\omega(u - u_h, v)\,. \tag{3.4}$$

Together with the Cauchy–Schwarz inequality this gives for any $v \in V\setminus\{0\}$

$$\begin{aligned}\left\|\!\left| \operatorname{Res} \right|\!\right\|_* &\geq \frac{|\operatorname{Res}(v)|}{\left\|\!\left| v \right|\!\right\|_\omega} = \frac{|b_\omega(u-u_h, v)|}{\left\|\!\left| v \right|\!\right\|_\omega} \\ &\geq \frac{|b_\omega(u-u_h, u-u_h)|^{1/2}\, |b_\omega(v,v)|^{1/2}}{\left\|\!\left| v \right|\!\right\|_\omega} \geq \left\|\!\left| u - u_h \right|\!\right\|_\omega .\end{aligned}$$

Additionally, the choice of $v = (u - u_h) \,/\, \left\|\!\left| u - u_h \right|\!\right\|_\omega$ applied to Equation (3.3) leads to $\operatorname{Res}(v) = \left\|\!\left| u - u_h \right|\!\right\|_\omega$.

Remark 3.4 Though the model problem assumes $u \equiv 0$ on ∂D, the bounds can be extended to inhomogeneous boundary data. For Dirichlet boundary data fulfilling

$u_D \in H^1(D) \cap H^2(\mathcal{E}(\Gamma_D))$ on all faces $\mathcal{E}(\Gamma_D)$ on Γ_D it holds the estimate

$$\left\|\!\left| u - u_h \right|\!\right|_\omega^2 \leq \left\|\!\left| \mathrm{Res} \right|\!\right|_*^2 + C \left\| h^{3/2} \kappa_{\max,T} \frac{\partial^2}{\partial s^2} (u_D - u_h) \right\|_{L^2(\Gamma_D)} . \tag{3.5}$$

Proof. For homogeneous Dirichlet boundary data it follows the identity $\left\|\!\left| \mathrm{Res} \right|\!\right|_* = \left\|\!\left| u - u_h \right|\!\right|_\omega$ as above. Let $V_{\Gamma_D} \coloneqq \{w \in H^1(D) | w = u - u_h \text{ on } \Gamma_D\}$. Under the given constraints for the Dirichlet boundary data it holds the identity

$$\left\|\!\left| u - u_h \right|\!\right|_\omega^2 = \left\|\!\left| \mathrm{Res} \right|\!\right|_*^2 + \inf_{V_{\Gamma_D}} \left\|\!\left| w \right|\!\right|_\omega^2 .$$

A clever choice of w leads to $\inf_{V_{\Gamma_D}} \left\|\!\left| w \right|\!\right|_\omega \leq C \left\| h^{3/2} \kappa_{\max,T} \frac{\partial^2}{\partial s^2} (u_D - u_h) \right\|_{L^2(\Gamma_D)}$ as shown in [BCD04]. Therein, the analysis is further extended to include Neumann boundaries. □

Remark 3.5 The upper bound for the Dirichlet data error in Equation (3.5) constitutes a higher order term and is therefore neglected in the following. Furthermore, the constant C only depends on the shape of the triangles and [Mer13] shows $C \leq 0.4980$ for isosceles triangles.

Lemma 3.6 *Consider for $\alpha \neq 0$ the residuals $\mathrm{Res}^\pm$ of the solution $u^\pm \coloneqq \alpha u \pm \frac{z}{\alpha}$ and its discrete approximation $u_h^\pm \coloneqq \alpha u_h \pm \frac{z_h}{\alpha}$ for the Problem (1.2) with the right-hand side $f^\pm \coloneqq \alpha f \pm \frac{g}{\alpha}$. With these, the error in the quantity of interest can be decomposed as follows*

$$Q(u - u_h) = \frac{1}{4} \left\|\!\left| \mathrm{Res}^+ \right|\!\right|_*^2 - \frac{1}{4} \left\|\!\left| \mathrm{Res}^- \right|\!\right|_*^2 . \tag{3.6}$$

Proof. The starting point is the following parallelogram equality based on the dual problem (1.4) and the Galerkin orthogonality of the solutions. For any $\alpha \neq 0$ it holds

$$\begin{aligned} Q(u) - Q(u_h) &= b_\omega(u - u_h, z - z_h) \\ &= b_\omega\Big(\alpha(u - u_h), \frac{z - z_h}{\alpha}\Big) \\ &\quad + \frac{1}{4} b_\omega(\alpha(u - u_h), \alpha(u - u_h)) - \frac{1}{4} b_\omega(\alpha(u - u_h), \alpha(u - u_h)) \\ &\quad + \frac{1}{4} b_\omega\Big(\frac{z - z_h}{\alpha}, \frac{z - z_h}{\alpha}\Big) - \frac{1}{4} b_\omega\Big(\frac{z - z_h}{\alpha}, \frac{z - z_h}{\alpha}\Big) \\ &= \frac{1}{4} \left\|\!\left| \alpha u + \frac{z}{\alpha} - \Big(\alpha u_h + \frac{z_h}{\alpha}\Big) \right|\!\right|_\omega^2 - \frac{1}{4} \left\|\!\left| \alpha u - \frac{z}{\alpha} - \Big(\alpha u_h - \frac{z_h}{\alpha}\Big) \right|\!\right|_\omega^2 . \end{aligned} \tag{3.7}$$

Equation (3.7) together with Equation (3.5) concludes the proof. □

The next step is to derive upper and lower bounds for $\left\|\mathrm{Res}^{\pm}\right\|_*$ which will allow to bound the error in the quantity of interest via Lemma 3.6.

Lemma 3.7 *For $T \in \mathcal{T}$ and $u \in V$ there exists $C_P(T) > 0$ such that the local Poincaré inequality holds as*

$$\|u - u_T\|_{L^2(T)} \leq C_P(T) h_T \|\nabla u\|_{L^2(T)}$$

with the mean function u_T as in Definition 3.1. The constant C_P depends solely on the shape of the triangle T.

Proof. A proof for general open, bounded and connected domains with $\mathcal{C}^1$ boundary is given in [Eva10, pp. 275]. The triangles $T \in \mathcal{T}$ do not fulfill the requirements. A general proof is presented in [Gal11, II.5.1]. □

Remark 3.8 $C_P(T) = \sup\left\{h_T^{-1} \|v - v_T\|_{L^2(T)} \;\middle|\; v \in H^1(T), \|v\|_\omega = 1\right\}$, defined for each element $T \in \mathcal{T}$, is called the Poincaré constant. Under certain assumptions, it can be explicitly bounded from above. For convex domains, like the triangles $T \in \mathcal{T}$, [PW60, Beb03] claim $C_P(T) \leq 1/\pi$ and the constant is therefore often referred to as the Payne–Weinberger constant. Moreover, for 2D domains [LS10] improves this estimate to $C_P(T) \leq 1/j_{1,1}$ where $j_{1,1} \approx 3.8317$ is the first positive root of the first Bessel function $J_1 = 1/\pi \int_0^\pi \cos(\varphi - x\sin(\varphi))\,\mathrm{d}\varphi$.

Theorem 3.9 *Consider some r with $\operatorname{div} r = -f_{\mathcal{T}}$. The residual Res with right-hand side f and solutions u, u_h is bounded above by*

$$\|\mathrm{Res}\|_* \leq \gamma(r) \tag{3.8}$$

with

$$\gamma(r)^2 \coloneqq \sum_{T\in\mathcal{T}} \left(\left\|\kappa^{-1/2}(p_h - r)\right\|_{L^2(T)} + C_P(T) h_T \kappa_{\min,T}^{-1/2} \|f - f_T\|_{L^2(T)} \right)^2.$$

The constants $C_P(T)$ are induced by Lemma 3.7 and thus share their properties.

Proof. Consider $r \in H(\operatorname{div}, D)$ satisfying $\operatorname{div} r = -f_{\mathcal{T}}$. Starting from the definition

of the residual in Equation (3.1), integration by parts results in

$$\operatorname{Res}(v) = \int_D (f + \operatorname{div} r)v \, \mathrm{d}x - \int_D (p_h - r) \cdot \nabla v \, \mathrm{d}x \tag{3.9}$$

for any $v \in H_0^1(D)$. Splitting the first integral on the right-hand side into a sum over the triangles of the triangulation allows for the local Poincaré inequalities from Lemma 3.7 and local Cauchy–Schwarz inequalities as it holds the orthogonality property $\int_T (f + \operatorname{div} r)v_T \, \mathrm{d}x = 0$ for the local mean function v_T for $v \in V$ and thus the integral can be bounded as follows

$$\begin{aligned}
\int_D (f + \operatorname{div} r)v \, \mathrm{d}x &= \sum_{T \in \mathcal{T}} \int_T (f - f_T)(v - v_T) \, \mathrm{d}x \\
&= \sum_{T \in \mathcal{T}} \int_T \kappa_{\min,T}^{-1/2}(f - f_T)\kappa_{\min,T}^{1/2}(v - v_T) \, \mathrm{d}x \\
&\leq \sum_{T \in \mathcal{T}} \left\| \kappa_{\min,T}^{-1/2}(f - f_T) \right\|_{L^2(T)} \kappa_{\min,T}^{1/2} \left\| v - v_T \right\|_{L^2(T)} \\
&\leq \sum_{T \in \mathcal{T}} \left\| \kappa_{\min,T}^{-1/2}(f - f_T) \right\|_{L^2(T)} C_P(\mathcal{T}) h_T \left\| \kappa^{1/2} \nabla v \right\|_{L^2(T)}.
\end{aligned}$$

Finally, the last integral in Equation (3.9) subjects to the inequality

$$-\int_D (p_h - r) \cdot \nabla v \, \mathrm{d}x \leq \sum_{T \in \mathcal{T}} \left\| \kappa^{-1/2} (p_h - r) \right\|_{L^2(T)} \left\| \kappa^{1/2} \nabla v \right\|_{L^2(T)}.$$

Altogether, these inequalities give a guaranteed estimate for the residual and another Cauchy–Schwarz inequality in $\mathbb{R}^{|\mathcal{T}|}$ together with the requisite $\|\!|v|\!\|_\omega = 1$ proves the claim. □

Remark 3.10 It is possible to extend the above result to Neumann boundaries $\Gamma_N \subset \Gamma = \partial D$ introducing additional constraints for the test function r and resulting in additional terms for the bounds in Equation (3.8). A detailed explanation is given in [Mer13, EMN16].

It remains to construct lower bounds for the dual norm of the residual. The starting point for the following lemma is the definition of this norm.

Lemma 3.11 *Consider the choice of some special function* $v_A \in P_2(\mathcal{T}) \cap C_0(D)$

with $v_A \equiv u_h$ on Γ_D. It holds the lower estimate

$$\|\mathrm{Res}\|_* \geq \xi(v_A - u_h) \quad \textit{where} \qquad \xi(v) \coloneqq \frac{|\mathrm{Res}(v)|}{\|v\|_\omega}.$$

Proof. The claim follows directly from the definition of the dual norm for the residual in Equation (3.2). □

Remark 3.12 The aim is to construct a function v_A fulfilling the required characteristics which at the same time approximates the argument of the maximum in ξ. A proposal from [Ain08, Bra07] incorporates the Crouzeix–Raviart solution $\hat{u}_{CR}$ of Problem (1.2) where the right-hand side f is replaced by its mean function $f_{\mathcal{T}}$. It starts of with the construction of the nonconforming piecewise quadratic function $v^0 \in P_2(D)$ by

$$v^0\big|_T \coloneqq \hat{u}_{CR} - \frac{f_T \varphi_T}{2\kappa_T} \qquad \text{for all } T \in \mathcal{T}, \tag{3.10}$$

where κ_T denotes the mean function of κ and for each $T \in \mathcal{T}$ it is

$$\varphi_T(x) \coloneqq \frac{|x - \mathrm{mid}(T)|^2}{2} - |T|^{-1} \int_T |y - \mathrm{mid}(T)|^2 \,\mathrm{d}y.$$

Let $\hat{p}_{RT}$ be the solution to the mixed formulation Problem (1.3) with the tensor κ replaced by $\kappa_{\mathcal{T}}$. It then holds the identity $\kappa_{\mathcal{T}} \nabla v^0 = \hat{p}_{RT}$ [Ain08] and thus it is possible to set

$$v^0\big|_T \coloneqq \operatorname*{arg\,min}_{v \in P_2(T)} \left\{ \|\kappa_{\mathcal{T}} \nabla v - \hat{p}_{RT}\|_{L^2(T)} \;\middle|\; |T|^{-1} \int_T v \,\mathrm{d}x = \hat{p}_{RT}(\mathrm{mid}(T)) \right\}. \tag{3.11}$$

Both definitions, Equation (3.10) and Equation (3.11), for v^0 require the computation of an additional discrete solution. Note, that in the case of a piecewise constant κ it holds the identity $\hat{p}_{RT} = p_{RT}$ for the mixed solution p_{RT} from Equation (1.3). Either way, v^0 can be used to compute some nearby function $v_A \in P_2(\mathcal{T}) \cap C_0(D)$, for example by means of interpolation of v^0. After enforcing boundary conditions on v_A to conform with u_h on the Dirichlet boundary, it fulfills the requirements of Lemma 3.11 and thus gives a lower bound for the residual.

In the following we will continue to use the naming conventions from Lemma 3.6, that is, we denote the composite functions which combine the primal and the dual problem as $f^\pm = \alpha f \pm \frac{f}{\alpha}$, $u^\pm = \alpha u \pm \frac{z}{\alpha}$ and so forth.

Corollary 3.13 *For any choices $r^{\pm} \in H(\operatorname{div}, D)$ obeying $\operatorname{div} r^{\pm} = -f_{\mathcal{T}}^{\pm}$ and for any choices $v^{\pm} \in V$ as well as $\alpha \neq 0$ from Equation (3.7) there hold the bounds*

$$\eta^{-}(\alpha, r^{-}, v^{+}) \leq Q(u - u_h) \leq \eta^{+}(\alpha, r^{+}, v^{-})$$

with the definition of η^{+} and η^{-} set as

$$\eta^{-} \coloneqq \frac{1}{4}\xi^{+}(v^{+}) - \frac{1}{4}\gamma^{-}(r^{-})^2 \quad \textit{and} \quad \eta^{+} \coloneqq -\frac{1}{4}\xi^{-}(v^{-}) + \frac{1}{4}\gamma^{+}(r^{+})^2.$$

Proof. The claim follows directly from the application of the upper and lower bounds in Theorem 3.9 and Lemma 3.11 to the dual norms of the residuals in Equation (3.6). □

Remark 3.14 A possible setup for $v^{\pm}$ is given in Remark 3.12. The choice $r^{\pm} = p_{RT}^{\pm}$ where $p_{RT}^{\pm}$ are the mixed solutions for the right-hand side $f^{\pm}$ with the notation from Lemma 3.6 is feasible and affordable to acquire. It remains to wisely select an α as poor choices profoundly impact the bounds' performances. Neglecting oscillation terms for the input data it is possible to derive a computable optimal α_{opt} [CMN13] as

$$\alpha_{\mathrm{opt}} \coloneqq \left\| \kappa^{-1/2}(q_h - q_{RT}) \right\|_{L^2(D)}^{1/2} \left\| \kappa^{-1/2}(p_h - p_{RT}) \right\|_{L^2(D)}^{-1/2}.$$

3.2 Probable Bounds for Quantities of Interest

The aim of the following section is the extension of the deterministic bounds derived for every $\omega \in \Omega$ to the stochastic context. It presents upper and lower bounds for the error in the discrete Monte Carlo finite element approximation up to some probability p. The deterministic bounds from Section 3.1 will confine the error in the spatial dimensions whereas the stochastic error is controlled with the help of some central limit theorem.

As a starting point, consider the following probabilistic bounds which confine the error in the Monte Carlo approximation of some random variable X with probability p. That is, we seek an upper bound ε which fulfills

$$\mathbf{P}\Big[\mathbf{E}[X] - \mathbf{E}_N^{\mathrm{MC}}[X] < \varepsilon\Big] \geq \mathrm{p}.$$

Lemma 3.15 *The cumulative distribution function for the standard normal distribution $\mathcal{N}_{0,1}$ is given by $\Phi(x) \coloneqq (2\pi)^{-1/2} \int_{-\infty}^{x} \exp(-t^2/2)\, \mathrm{d}t$. It exists a sequence*

$(\varepsilon_X^*(N,\mathsf{p}))_{N\in\mathbb{N}}$ *such that for every* $\mathsf{p}\in[0,1]$ *it is* $\varepsilon_X^*(N,\mathsf{p})\to 0$ *for* $N\to\infty$ *and it holds the upper bound for the Monte Carlo estimation error*

$$\mathbf{E}[X]-\mathbf{E}_N^{\mathrm{MC}}[X]\leq\varepsilon_X(N,\mathsf{p})\qquad\text{with probability }\mathsf{p}$$

where $\varepsilon_X(N,\mathsf{p})\coloneqq\mathbf{Var}[X]^{1/2}\,N^{-1/2}\Phi^{-1}(\mathsf{p})+\varepsilon_X^*(N,\mathsf{p})$.

Proof. The central limit theorem by Lindeberg and Lévy [Fel68, Kal02] gives

$$\mathbf{Var}[X]^{-1/2}\,N^{1/2}(\mathbf{E}[X]-\mathbf{E}_N^{\mathrm{MC}}[X])\xrightarrow{d}\mathcal{N}_{0,1}\qquad\text{for }N\to\infty$$

where $\xrightarrow{d}$ is the limit in the distributional sense. For any $\mathsf{s}>0$ this gives the limit

$$\mathsf{p}\coloneqq\mathbf{P}\Big[\mathbf{E}[X]-\mathbf{E}_N^{\mathrm{MC}}[X]<\mathbf{Var}[X]^{1/2}\,N^{-1/2}\mathsf{s}\Big]\to\Phi(\mathsf{s})\qquad\text{for }N\to\infty.$$

Define $\varepsilon_X(N,\mathsf{s})\coloneqq\mathbf{Var}[X]^{1/2}\,N^{-1/2}\mathsf{s}$. As we have $\mathsf{p}\to\Phi(\mathsf{s})$ and as Φ is invertible we get $\Phi^{-1}(\mathsf{p})\to\mathsf{s}$ for $N\to\infty$ which gives the result together with an appropriate converging sequence $\varepsilon_X^*(N,\mathsf{p})\to 0$ for $N\to\infty$. □

Remark 3.16 This result can be used to control the error in a Monte Carlo estimation. Lower estimates are derived in the very same manner as the standard normal distribution is even and its cumulative distribution function is odd. Note, that they are governed by the (fixed) variance and the number of samples used for the approximation. An error reduction by the factor of two needs four times the samples, which results in slow convergence.

Corollary 3.17 *Prescribing the probability* $\mathsf{p}\in(0,1)$ *the following constraints for the overall Monte Carlo finite element approximation error can be established. Given that* $\mathsf{p}^*,\mathsf{p}^+,\mathsf{p}^-\in(0,1)$ *subject to* $\mathsf{p}=2\mathsf{p}^*+\mathsf{p}^+-\mathsf{p}^--3$ *the bounds*

$$\begin{aligned}\eta^{\ominus}(\mathsf{p}^*,\mathsf{p}^-,N)\coloneqq{}&\mathbf{E}_N^{\mathrm{MC}}\Big[\eta^-\Big]-\mathbf{Var}\Big[\eta^-\Big]^{1/2}\,N^{-1/2}\Phi^{-1}(\mathsf{p}^-)-\varepsilon_{\eta^-}^*(N,\mathsf{p}^-)\\&-\mathbf{Var}[Q(u_h)]^{1/2}\,N^{-1/2}\Phi^{-1}(\mathsf{p}^*)-\varepsilon_{Q(u_h)}(N,\mathsf{p}^*),\\\eta^{\oplus}(\mathsf{p}^*,\mathsf{p}^+,N)\coloneqq{}&\mathbf{E}_N^{\mathrm{MC}}\Big[\eta^+\Big]+\mathbf{Var}\Big[\eta^+\Big]^{1/2}\,N^{-1/2}\Phi^{-1}(\mathsf{p}^+)+\varepsilon_{\eta^+}^*(N,\mathsf{p}^+)\\&+\mathbf{Var}[Q(u_h)]^{1/2}\,N^{-1/2}\Phi^{-1}(\mathsf{p}^*)+\varepsilon_{Q(u_h)}(N,\mathsf{p}^*)\end{aligned}$$

hold with probability of at least p *in the sense that*

$$\mathbf{P}\Big[\eta^{\ominus}(\mathsf{p}^*,\mathsf{p}^-,N)\leq\mathbf{E}[Q(u)]-\mathbf{E}_N^{\mathrm{MC}}[Q(u_h)]\leq\eta^{\oplus}(\mathsf{p}^*,\mathsf{p}^+,N)\Big]\geq\mathsf{p}.$$

Proof. Direct calculations together with Corollary 3.13 show that the approximation error is governed from above by

$$\begin{aligned}\mathbf{E}[Q(u)] - \mathbf{E}_N^{\mathrm{MC}}[Q(u_h)] &\leq \mathbf{E}\left[\eta^+\right] + \mathbf{E}[Q(u_h)] - \mathbf{E}_N^{\mathrm{MC}}[Q(u_h)] \\ &= \mathbf{E}_N^{\mathrm{MC}}\left[\eta^+\right] + \left(\mathbf{E}\left[\eta^+\right] - \mathbf{E}_N^{\mathrm{MC}}\left[\eta^+\right]\right) + \left(\mathbf{E}[Q(u_h)] - \mathbf{E}_N^{\mathrm{MC}}[Q(u_h)]\right).\end{aligned}$$

Similar calculations give the corresponding lower bound. The next step will apply Lemma 3.15 a total of four times to these upper and lower bounds in order to control the Monte Carlo errors $\mathbf{E}[\eta^{\pm}] - \mathbf{E}_N^{\mathrm{MC}}[\eta^{\pm}]$ and twice $\mathbf{E}[Q(u_h)] - \mathbf{E}_N^{\mathrm{MC}}[Q(u_h)]$ and, thus, the four inequalities

$$\begin{aligned}\mathbf{E}\left[\eta^+\right] - \mathbf{E}_N^{\mathrm{MC}}\left[\eta^+\right] &\overset{\mathrm{I}}{\leq} \mathbf{Var}\left[\eta^+\right]^{1/2} N^{-1/2}\Phi^{-1}(\mathsf{p}^+) + \varepsilon^*_{\eta^+}(N, \mathsf{p}^+), \\ \mathbf{E}[Q(u_h)] - \mathbf{E}_N^{\mathrm{MC}}[Q(u_h)] &\overset{\mathrm{II}}{\leq} \mathbf{Var}[Q(u_h)]^{1/2} N^{-1/2}\Phi^{-1}(\mathsf{p}^*) + \varepsilon_{Q(u_h)}(N, \mathsf{p}^*), \\ \mathbf{E}\left[\eta^-\right] - \mathbf{E}_N^{\mathrm{MC}}\left[\eta^-\right] &\overset{\mathrm{III}}{\geq} \mathbf{Var}\left[\eta^-\right]^{1/2} N^{-1/2}\Phi^{-1}(\mathsf{p}^-) - \varepsilon^*_{\eta^-}(N, \mathsf{p}^-), \\ \mathbf{E}[Q(u_h)] - \mathbf{E}_N^{\mathrm{MC}}[Q(u_h)] &\overset{\mathrm{IV}}{\geq} \mathbf{Var}[Q(u_h)]^{1/2} N^{-1/2}\Phi^{-1}(\mathsf{p}^*) - \varepsilon_{Q(u_h)}(N, \mathsf{p}^*)\end{aligned}$$

hold with the prescribed probabilities $\mathsf{p}^+, \mathsf{p}^*, \mathsf{p}^-, \mathsf{p}^* \in (0, 1]$ respectively. The goal is, that these individual inequalities should hold at the same time with the overall probability p which we want to control from below with the individual probabilities. Consider two events A and B. The combined probability of the event $\mathsf{A} \cap \mathsf{B}$ is constrained by

$$\mathbf{P}[\mathsf{A} \cap \mathsf{B}] = \mathbf{P}[\mathsf{A}] + \mathbf{P}[\mathsf{B}] - \mathbf{P}[\mathsf{A} \cup \mathsf{B}] \geq \mathbf{P}[\mathsf{A}] + \mathbf{P}[\mathsf{B}] - 1. \tag{3.12}$$

Let p^* describe the probability in the Monte Carlo bound from Lemma 3.15 for $Q(u_h)$ and $\mathsf{p}^{\pm}$ the same for the bounds $\eta^{\pm}$. Successive application of Equation (3.12) gives the lower bound for the overall probability p

$$\begin{aligned}\mathsf{p} = \mathbf{P}[\mathsf{I} \cap \mathsf{II} \cap \mathsf{III} \cap \mathsf{IV}] &\geq \mathbf{P}[\mathsf{I}] + \mathbf{P}[\mathsf{II} \cap \mathsf{III} \cap \mathsf{IV}] - 1 \\ &\geq \mathbf{P}[\mathsf{I}] + \mathbf{P}[\mathsf{II}] + \mathbf{P}[\mathsf{III}] + \mathbf{P}[\mathsf{IV}] - 3\end{aligned}$$

and thus to the condition $\mathsf{p} \geq 2\mathsf{p}^* + \mathsf{p}^+ + \mathsf{p}^- - 3$. This concludes the proof. □

Remark 3.18 In numerical applications N has to be chosen large enough such that the influences of the residual terms $\varepsilon^*_{\eta^{\pm}}$ and $\varepsilon^*_{Q(u_h)}$ are negligible. Furthermore, the parameters $\mathsf{p}^*, \mathsf{p}^+, \mathsf{p}^- \in (0, 1]$ have to be chosen large enough according to the constraint $\mathsf{p} = 2\mathsf{p}^* + \mathsf{p}^+ + \mathsf{p}^- - 3$. Numerical aspects are discussed in Remark 3.20 later on.

The same ansatz is extended to the multilevel Monte Carlo context by utilizing the deterministic bounds on each level and through the application of the multilevel idea to these bounds. Therefore, define on each level ℓ the bounds $\eta_\ell^\pm$ as random variables dependent on ω such that $\eta_\ell^- \leq Q(u - u_\ell) \leq \eta_\ell^+$.

Corollary 3.19 *Prescribing the probability* $\mathsf{p} \in (0,1)$ *the following constraints for the overall Monte Carlo finite element approximation error can be established. Given that* $\mathbf{p}^*, \mathbf{p}^+, \mathbf{p}^- \in (0,1)^L$ *with the naming convention* $\mathbf{p}^\bullet = (\mathsf{p}_0^\bullet, \dots, \mathsf{p}_L^\bullet)$ *for* $\bullet \in \{*, +, -\}$ *subject to* $\mathsf{p} = 2\sum_{\ell=0}^L \mathsf{p}_\ell^* + \sum_{\ell=0}^L \mathsf{p}_\ell^+ + \sum_{\ell=0}^L \mathsf{p}_\ell^- - 4L - 3$ *and the random variables* $Y_0 = Q(u_0), Y_\ell = Q(u_\ell) - Q(u_{\ell-1})$ *as well as* $Z_0^\pm = \eta_0^\pm, Z_\ell^\pm = \eta_\ell^\pm - \eta_{\ell-1}^\pm$ *for* $\ell = 1, \dots, L$ *there hold the upper and lower bounds*

$$\begin{aligned}
\eta_{\mathrm{ML}}^\ominus(\mathbf{p}^*, \mathbf{p}^-, \boldsymbol{N}) &\coloneqq \mathbf{E}_{\boldsymbol{N}}^{\mathrm{ML}}\Big[\eta_L^-\Big] - \sum_{\ell=0}^L \mathbf{Var}\Big[Z_\ell^-\Big]^{1/2} N_\ell^{-1/2} \Phi^{-1}(\mathsf{p}_\ell^-) - \varepsilon_{Z_\ell^-}^*(N_\ell, \mathsf{p}_\ell^-) \\
&\quad - \sum_{\ell=0}^L \mathbf{Var}[Y_\ell]^{1/2} N_\ell^{-1/2} \Phi^{-1}(\mathsf{p}_\ell^*) - \varepsilon_{Y_\ell}^*(N_\ell, \mathsf{p}_\ell^*) \\
\eta_{\mathrm{ML}}^\oplus(\mathbf{p}^*, \mathbf{p}^+, \boldsymbol{N}) &\coloneqq \mathbf{E}_{\boldsymbol{N}}^{\mathrm{ML}}\Big[\eta_L^+\Big] + \sum_{\ell=0}^L \mathbf{Var}\Big[Z_\ell^+\Big]^{1/2} N_\ell^{-1/2} \Phi^{-1}(\mathsf{p}_\ell^+) + \varepsilon_{Z_\ell^+}^*(N_\ell, \mathsf{p}_\ell^+) \\
&\quad + \sum_{\ell=0}^L \mathbf{Var}[Y_\ell]^{1/2} N_\ell^{-1/2} \Phi^{-1}(\mathsf{p}_\ell^*) + \varepsilon_{Y_\ell}^*(N_\ell, \mathsf{p}_\ell^*)
\end{aligned}$$

with probability of at least p *in the sense that*

$$\mathbf{P}\Big[\eta_{\mathrm{ML}}^\ominus(\mathbf{p}^*, \mathbf{p}^-, \boldsymbol{N}) \leq \mathbf{E}[Q(u)] - \mathbf{E}_N^{\mathrm{ML}}[Q(u_L)] \leq \eta_{\mathrm{ML}}^\oplus(\mathbf{p}^*, \mathbf{p}^+, \boldsymbol{N})\Big] \geq \mathsf{p}.$$

Proof. Similar to the proof of Corollary 3.17, the introduction of an auxiliary zero and the application of the deterministic bounds from Corollary 3.13 give

$$\begin{aligned}
\mathbf{E}[Q(u)] - \mathbf{E}_{\boldsymbol{N}}^{\mathrm{ML}}[Q(u_L)] &= \mathbf{E}[Q(u)] - \sum_{\ell=0}^L \mathbf{E}_{N_\ell}^{\mathrm{MC}}[Y_\ell] \\
&= \mathbf{E}[Q(u)] - \mathbf{E}[Q(u_L)] + \sum_{\ell=0}^L \Big(\mathbf{E}[Y_\ell] - \mathbf{E}_{N_\ell}^{\mathrm{MC}}[Y_\ell]\Big) \\
&\leq \mathbf{E}\Big[\eta_L^+\Big] + \sum_{\ell=0}^L \Big(\mathbf{E}[Y_\ell] - \mathbf{E}_{N_\ell}^{\mathrm{MC}}[Y_\ell]\Big) \\
&= \sum_{\ell=0}^L \mathbf{E}_{N_\ell}^{\mathrm{MC}}\Big[Z_\ell^+\Big] + \sum_{\ell=0}^L \Big(\mathbf{E}\Big[Z_\ell^+\Big] - \mathbf{E}_{N_\ell}^{\mathrm{MC}}\Big[Z_\ell^+\Big]\Big) + \sum_{\ell=0}^L \Big(\mathbf{E}[Y_\ell] - \mathbf{E}_{N_\ell}^{\mathrm{MC}}[Y_\ell]\Big).
\end{aligned}$$

The lower constraint follows accordingly. For the combined upper and lower bounds,

we have to apply Lemma 3.15 to the Monte Carlo errors $\mathbf{E}\left[Z_\ell^+\right] - \mathbf{E}_{N_\ell}^{\mathrm{MC}}\left[Z_\ell^+\right]$ and $\mathbf{E}\left[Z_\ell^-\right] - \mathbf{E}_{N_\ell}^{\mathrm{MC}}\left[Z_\ell^-\right]$ as well as twice $\mathbf{E}[Y_\ell] - \mathbf{E}_{N_\ell}^{\mathrm{MC}}[Y_\ell]$ for $\ell = 0, \ldots, L$, that is $4L$ times in total. Just like the four estimates in Corollary 3.17 all of these error estimates have to hold with probability p at the same time. Let p_ℓ^* describe the probability for the estimate in Lemma 3.15 with regard to the random variable Y_ℓ, p_ℓ^+ and p_ℓ^- the same for Z_ℓ^+ and Z_ℓ^-. Thus, a successive application of Equation (3.12) to the $4L$ probabilities leads to the condition $\mathsf{p} = 2\sum_{\ell=0}^{L} \mathsf{p}_\ell^* + \sum_{\ell=0}^{L} \mathsf{p}_\ell^+ + \sum_{\ell=0}^{L} \mathsf{p}_\ell^- - 4L - 3$ if we want a guaranteed probability for the upper and lower bounds of at least p and hence to the claim. □

Remark 3.20 The parameters $\mathsf{p}^*, \mathsf{p}^+, \mathsf{p}^-$ and $\mathbf{p}^*, \mathbf{p}^+, \mathbf{p}^-$ in Corollary 3.17 and Corollary 3.19 are free up to the given constraints and an optimal choice can be acquired through numerical means (e.g. [Kra88]). A symbolic computation of the optimal values is not possible as the inverse cumulative distribution function of the normal distribution cannot be described by elementary functions.

Note however, that in practice the variances of the bounds $\eta^\pm$, of the quantity of interest $Q(u_h)$ and $Z_\ell^\pm$, Y_ℓ respectively can vary greatly and hence the optimal choice might lead to numerically unstable results as this can lead to blowups of errors in the approximation of the variances. Choosing the parameters equally circumvents this problem and usually does not result in significantly poorer performance for relatively small L.

Just as in the Monte Carlo case, the number of samples per level N_ℓ has to be large enough such that the limit error terms $\varepsilon_{Z_\ell^\pm}^*$ and $\varepsilon_{Y_\ell}^*$ are small enough to be governed by the other contributions in the bounds.

Nevertheless, the approach in Equation (3.12) becomes less beneficial for larger L. This is to be expected as for large L a smaller tolerance for the error of the individual Monte Carlo estimators on the levels must hold. But then the equal distribution suggested here is disadvantageous. A better bound might be conceived with the help of the normality of the multilevel Monte Carlo estimator similar to [CHAN+14] where this is used to derive the number of levels and the optimal number of samples.

3.3 Goal-Adaptive Mesh Refinement

The aim of this section is to provide local error indicators $\eta_{\ell,\omega}^{\mathrm{loc}}$ for each $\omega \in \Omega$ which give clues to the influence of the mesh structure on the error in the quantity of interest $Q(u) - Q(u_h)$. Based on these indicators a finer mesh is derived which might lead to faster convergence rates than uniform meshes. The indicators used are based

on standard finite element adaptivity.

Mean Error Indicators

Because we want to derive error indicators for the quantity of interest, the influence of the goal functional onto the error has to be respected as shown in [OP01]. Furthermore, [MS09] showed some guaranteed error reduction by using a global Cauchy–Schwarz inequality instead of the local ones. A very similar idea based on wavelets was already earlier presented in [DKV06]. Comparison of the two approaches however is beyond the scope of this work and thus we focus on the former approach as it is completely realized in the finite element framework. In the stochastic context this error reduction cannot be guaranteed anymore in the same sense. Nevertheless, it holds

$$\begin{aligned}\mathbf{E}[|Q(u) - Q(u_\ell)|] = \mathbf{E}[|b_\omega(u - u_\ell, z - z_\ell)|] &\leq \mathbf{E}[\|\!|u - u_\ell|\!\|_\omega \, \|\!|z - z_\ell|\!\|_\omega] \\ &\leq \left(\mathbf{E}\left[\|\!|u - u_\ell|\!\|_\omega^2\right] \mathbf{E}\left[\|\!|z - z_\ell|\!\|_\omega^2\right]\right)^{1/2}.\end{aligned} \tag{3.13}$$

For some fixed $\omega \in \Omega$ the local indicators $\eta_{\ell,\omega}^{\mathrm{loc}}(T)$ and $\tilde{\eta}_{\ell,\omega}^{\mathrm{loc}}(T)$ for the left and right terms on the right-hand side of Equation (3.13) give these error reduction properties. They are defined for each $T \in \mathcal{T}$ as follows

$$\begin{aligned}\eta_{\ell,\omega}^{\mathrm{loc}}(T)^2 &= \frac{h_T^2}{\kappa_{\max,T}} \|f\|_{L^2(T)}^2 + \sum_{E \in \mathcal{E}(T)} \frac{h_T}{\kappa_{\max,E}} \|[\kappa \nabla u_\ell \cdot n_E]\|_{L^2(E)}^2, \\ \tilde{\eta}_{\ell,\omega}^{\mathrm{loc}}(T)^2 &= \frac{h_T^2}{\kappa_{\max,T}} \|g\|_{L^2(T)}^2 + \sum_{E \in \mathcal{E}(T)} \frac{h_T}{\kappa_{\max,E}} \|[\kappa \nabla z_\ell \cdot n_E]\|_{L^2(E)}^2.\end{aligned}$$

In the next step these are used to define the mean error indicators $\mathbf{E}\left[\eta_{\ell,\omega}^{\mathrm{loc}}(T)\right]$ and $\mathbf{E}\left[\tilde{\eta}_{\ell,\omega}^{\mathrm{loc}}(T)\right]$ which in practice we approximate with Monte Carlo estimators $\mathbf{E}_{N_\eta}^{\mathrm{MC}}\left[\eta_{\ell,\omega}^{\mathrm{loc}}(T)\right]$ and $\mathbf{E}_{N_\eta}^{\mathrm{MC}}\left[\tilde{\eta}_{\ell,\omega}^{\mathrm{loc}}(T)\right]$ using N_η samples. As with the deterministic case, some criterion based on the indicators is used to select a set of elements for refinement. With the parameter $0 < \theta \leq 1$ the bulk criterion selects the smallest subsets $\mathcal{M}_{\ell,u} \subseteq \mathcal{T}_\ell$ and $\mathcal{M}_{\ell,z} \subseteq \mathcal{T}_\ell$ fulfilling

$$\sum_{T \in \mathcal{M}} \mathbf{E}\left[\eta^{\mathrm{loc}}(T)\right]^2 \geq \theta \sum_{T \in \mathcal{T}} \mathbf{E}\left[\eta^{\mathrm{loc}}(T)\right]^2 \tag{3.14}$$

with either $\mathcal{M} = \mathcal{M}_{\ell,u}$, $\eta^{\mathrm{loc}} = \eta_{\ell,\omega}^{\mathrm{loc}}$ or $\mathcal{M} = \mathcal{M}_{\ell,z}$, $\eta^{\mathrm{loc}} = \tilde{\eta}_{\ell,\omega}^{\mathrm{loc}}$, respectively. In the deterministic case, both choices will lead to meshes with guaranteed error reduction properties and thus for the error in the quantity of interest in Equation (3.13). It is

therefore admissible to choose the set of marked cells with smaller cardinality.

Based on the selected cells, a new mesh $\mathcal{T}_{\ell+1}$ is generated. In this process, a closure step will extend the selected set of cells such that the refined mesh retains the regularity properties. We will use the Dörfler marking strategy in [Dö96]. In the following its bulk parameter is set to $\theta = 0.5$. A pseudo code depiction of the described algorithm is presented in Algorithm 3.1. We will denote the error indicators that where used for the refinement with $\eta_{\mathrm{MS}}^{\mathrm{loc}}$ and the global error estimator with

$$\eta_{\mathrm{MS}} \coloneqq \left(\sum_{T \in \mathcal{T}_\ell} \left(\eta_{\mathrm{MS}}^{\mathrm{loc}} \right)^2 \right)^{1/2} .$$

As discussed in [MS09], the choice of $\mathcal{M}_{\ell,u} \cup \mathcal{M}_{\ell,z}$ is not optimal asymptotically. However, as the refinement only occurs with respect to the primal or the dual problem in each step, the error reduction property is not guaranteed for each refinement. In fact, it might happen, that the error is momentarily increased between two levels. This effect is clearly visible in the deterministic examples in Chapter 5.

Additionally, we will consider the error estimator η_{OP}. Based on the element-wise Cauchy–Schwarz inequality we define

$$\begin{aligned} \mathbf{E}[|Q(u) - Q(u_\ell)|] &\leq \mathbf{E}\left[\sum_{T \in \mathcal{T}_\ell} \left\| u - u_\ell \right\|_T \left\| z - z_\ell \right\|_T \right] \\ &\leq \mathbf{E}\left[\sum_{T \in \mathcal{T}_\ell} \eta_{\ell,\omega}^{\mathrm{loc}}(T) \tilde{\eta}_{\ell,\omega}^{\mathrm{loc}}(T) \right] =: \eta_{\mathrm{OP}} \end{aligned}$$

by applying the local error estimates. Algorithm 3.2 gives a short overview of this method using a Monte Carlo estimator with N_η samples to approximate the expected value. In the numerical simulations we will compare these refinement methods with uniform meshes as well as the mean indicator for the primal problem

$$\eta_u^2 \coloneqq \mathbf{E}\left[\sum_{T \in \mathcal{T}_\ell} \eta_{\ell,\omega}^{\mathrm{loc}}(T) \right]$$

approximated yet again with N_η samples. However, this might lead to slow or even stalled convergence in the error for the quantity of interest as only the primal solution is considered. In any case, both of these algorithms are comparatively cheap, that is, with respect to the actual calculations. On each level of the algorithm only a very small amount of samples is needed to estimate the mean error indicators. This is due to the self correcting nature of the algorithm. Consider the worst case, that the triangle

In : initial mesh $\mathcal{T}_0$, number of samples N_η, bulk criterion θ
Out : hierarchy of meshes $(\mathcal{T}_\ell)_{\ell=0,1,\dots}$

for $\ell = 0, 1, \dots$ **do**
  solve $(u_\ell^i)_{i=0,\dots,N_\eta}$ and $(z_\ell^i)_{i=0,\dots,N_\eta}$ on $\mathcal{T}_\ell$
  calculate $\left(\eta_{\ell,i}^{\mathrm{loc}}(T)\right)_{i=0,\dots,N_\eta;T\in\mathcal{T}}$ and $\left(\tilde{\eta}_{\ell,i}^{\mathrm{loc}}(T)\right)_{i=0,\dots,N_\eta;T\in\mathcal{T}}$
  approximate mean values $\left(\mathbf{E}_{N_\eta}^{\mathrm{MC}}\left[\eta_\ell^{\mathrm{loc}}(T)\right]\right)_{T\in\mathcal{T}}$ and $\left(\mathbf{E}_{N_\eta}^{\mathrm{MC}}\left[\tilde{\eta}_\ell^{\mathrm{loc}}(T)\right]\right)_{T\in\mathcal{T}}$
  mark $\mathcal{M}_{\ell,u}$ and $\mathcal{M}_{\ell,z}$ by Equation (3.14)
  if $|\mathcal{M}_{\ell,u}| \leq |\mathcal{M}_{\ell,z}|$ **then**
    $\mathcal{T}_{\ell+1} = \mathrm{refine}(\mathcal{T}_\ell, \mathcal{M}_{\ell,u},)$
  else
    $\mathcal{T}_{\ell+1} = \mathrm{refine}(\mathcal{T}_\ell, \mathcal{M}_{\ell,z})$

Algorithm 3.1: Generation of the MLMC mesh hierarchy based on η_{MS}

In : initial mesh $\mathcal{T}_0$, number of samples N_η, bulk criterion θ
Out : hierarchy of meshes $(\mathcal{T}_\ell)_{\ell=0,1,\dots}$

for $\ell = 0, 1, \dots$ **do**
  solve $(u_\ell^i)_{i=0,\dots,N_\eta}$ and $(z_\ell^i)_{i=0,\dots,N_\eta}$ on $\mathcal{T}_\ell$
  calculate $\left(\eta_{\ell,i}^{\mathrm{loc}}(T)\tilde{\eta}_{\ell,i}^{\mathrm{loc}}(T)\right)_{i=0,\dots,N_\eta;T\in\mathcal{T}}$
  approximate mean value $\left(\mathbf{E}_{N_\eta}^{\mathrm{MC}}\left[\eta_\ell^{\mathrm{loc}}(T)\tilde{\eta}_\ell^{\mathrm{loc}}(T)\right]\right)_{T\in\mathcal{T}}$
  mark $\mathcal{M}_\ell$ by Equation (3.14)
  $\mathcal{T}_{\ell+1} = \mathrm{refine}(\mathcal{T}_\ell, \mathcal{M}_\ell)$

Algorithm 3.2: Generation of the MLMC mesh hierarchy based on η_{OP}

with the biggest mean error indicator T^* is not refined because of a bad selection of samples. Then, in the next loop the errors which have been chosen for refinement on the last level have been lowered, as a result of the error reduction property of finite element adaptivity. Now, the probability for triangle T^* being chosen for refinement is even bigger. In fact, it holds for the probability $\mathbf{P}[T^* \in \mathcal{T}_\ell \text{ will be refined}] \to 1$ for $\ell \to \infty$ due to the error reduction properties of the error indicators.

As a result, the cost to generate this mesh sequence is dominated by the cost of one solution on the last level as the same fixed number of samples on each level can be chosen. In fact, even one single sample on each level might be considered together with different choices of θ in Equation (3.14). However, as modern computers tend to be multi- or even many-core machines and as Monte Carlo algorithms are easily implemented in parallel it gives almost no computational advantage to choose less samples than processors are available.

It is not optimal to choose all the meshes from the sequence generated by this algorithm as the error reduction between the levels are too small. Instead, we will choose a subsequence such that for each mesh in the uniform sequence this new set of adaptive meshes contains a grid with a similar number of degrees of freedom. In the examples in Chapter 5 we will choose for each uniform mesh the first adaptive grid with at least the same number of degrees of freedom. This choice of meshes will give good results but [HANvST15] might suggest a different subset. For an optimal choice, heuristical approaches have to be developed. An interesting ansatz is the continuation multilevel Monte Carlo algorithm in [CHAN$^+$14] which exploits the normality of the multilevel Monte Carlo estimator for error control.

Pathwise Refinement

As an alternative to the above algorithm we will suggest a different adaptive approach. In the method presented, the adaptivity is employed for the mean error indicators leading to a mesh sequence with an optimal refinement for the average error. This results in less optimal refinement when we consider only a single $\omega \in \Omega$ as the worst case of this method is in fact the uniform mesh sequence.

A suggestion to overcome this situation is to employ pathwise refinement that is specific to ω. For each $\omega \in \Omega$ we define a mesh sequence $\mathcal{T}_0^\omega, \ldots, \mathcal{T}_L^\omega$ according to the algorithm proposed in [MS09] up to a common prescribed tolerance ε_η for the used error indicator. Algorithm 3.3 depicts this approach. In the Monte Carlo case for each ω the quantity of interest is computed with these meshes and the expected

In : initial mesh $\mathcal{T}_0^\omega$, bulk criterion θ, sample ω
Out : hierarchy of meshes $(\mathcal{T}_\ell^\omega)_{\ell=0,1,\dots}$

for $\ell = 0, 1, \dots$ **do**
  solve u_ℓ^ω and z_ℓ^ω on $\mathcal{T}_\ell^\omega$
  calculate $\left(\eta_\ell^{\mathrm{loc}}(\omega, T)\right)_{T\in\mathcal{T}}$ and $\left(\tilde{\eta}_\ell^{\mathrm{loc}}(\omega, T)\right)_{T\in\mathcal{T}}$
  mark $\mathcal{M}_{\ell,u}^\omega$ and $\mathcal{M}_{\ell,z}^\omega$ similar to Equation (3.14)
  if $\left|\mathcal{M}_{\ell,u}^\omega\right| \leq \left|\mathcal{M}_{\ell,z}^\omega\right|$ **then**
    $\mathcal{T}_{\ell+1}^\omega = \mathrm{refine}(\mathcal{T}_\ell^\omega, \mathcal{M}_{\ell,u,})$
  else
    $\mathcal{T}_{\ell+1}^\omega = \mathrm{refine}(\mathcal{T}_\ell^\omega, \mathcal{M}_{\ell,z})$

Algorithm 3.3: Pathwise generation of the MLMC mesh hierarchy based on η_{MS}

value is taken

$$\mathbf{E}[Q(u)] \approx \mathbf{E}_N^{\mathrm{MC}}\left[Q\left(u_{h(\omega)}\right)\right].$$

A similar approach is conceivable for the multilevel Monte Carlo method where the first level is treated as the Monte Carlo method and the latter levels choose two different meshes for each ω from the sequence created with the above algorithm based on the numbers of degrees of freedom or the achieved error reduction.

So far our numerical simulations show an unreliable performance for both methods compared to the mean indicators as the generation of the meshes for each ω is too costly for finer grids. Further tests with different values for θ or other cost improvements and better stopping criterions are necessary to render this approach competitive as indicated in Experiment 5.3.

3.4 Heuristics for the Optimal Number of Samples

The Monte Carlo method as well as the multilevel Monte Carlo method have in common that two approximation parameters have to be carefully chosen in order to assure the best possible convergence with respect to the computational cost. Based on the error decompositions in Lemma 2.3 for Monte Carlo and Lemma 2.7 for multilevel Monte Carlo it is not possible to compute the optimal values a priori. However, given some initial small quantity of samples, sufficient heuristics can be derived.

For the multilevel Monte Carlo method in [TSGU13] the constant c_N in Equation (2.7) is chosen as the minimal value where the expected asymptotic convergence rate of the finite element error is realized. This value has to be provided and thus known a priori or determined heuristically. Alternatively, the error can be approximated with the help of the multilevel correction terms $\mathbf{E}[Y_\ell - Y_{\ell-1}]$ as proposed by [Mike B. Giles, personal communication, September 16, 2015]. This and other more robust approaches are presented below.

For the rest of the section we will assume that there exists an $\alpha > 0$ such that it holds $|\mathbf{E}[Q(u)] - \mathbf{E}[Q(u_h)]| \lesssim h^\alpha$ where h is defined as the mean diameter of the triangles in the triangulation $\mathcal{T}_h$, that is $h = |\mathcal{T}|^{-1} \sum_{T \in \mathcal{T}} h_T$. The first ansatz considers the Monte Carlo method and is based on the error indicators from Section 3.3. A sketch of the idea is given in Figure 3.1.

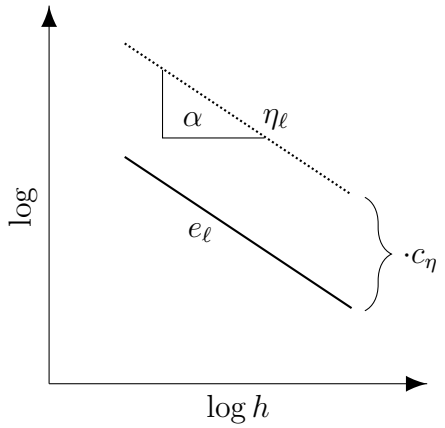


Figure 3.1: Logarithmic sketch of the different variables in the heuristics for the ansatz with error estimators.

Simple Ansatz for the Monte Carlo Method

The idea is to balance the two contributing terms in the error decomposition from Lemma 2.3. The first step is to find good approximations for the two error parts up to the number of samples N which can then be estimated. Some initial number of samples N_{init} will provide us with the discrete variance $\mathbf{Var}^{\text{MC}}_{N_{\text{init}}}[Q(u_h)]$ and an approximate Monte Carlo estimate $\mathbf{E}^{\text{MC}}_{N_{\text{init}}}[\eta_h]$ on that mesh for one of the error indicators $\eta_h = \eta_{MS}$ or $\eta_h = \eta_{OP}$ from Section 3.3. The starting point is the error

decomposition for the Monte Carlo method in Equation (2.1), that is

$$\mathbf{E}\left[\left(\mathbf{E}_N^{\mathrm{MC}}[Q(u_h)] - \mathbf{E}[Q(u)]\right)^2\right] = N^{-1}\,\mathbf{Var}[Q(u_h)] + \mathbf{E}[Q(u) - Q(u_h)]^2\,.$$

The discrete variance sufficiently approximates the first term for relatively small numbers of initial samples N_{init}. For the second term it holds the upper bound

$$\mathbf{E}[Q(u) - Q(u_h)]^2 \leq \mathbf{E}[|Q(u) - Q(u_h)|]^2 \leq \mathbf{E}[c_\eta \eta_h]^2$$

due to the Cauchy–Schwarz inequality in Equation (3.13). The residual error indicators η_u and η_z for the primal and dual problem induce efficiency constants that depend on the angles of the mesh as well as the input data and are combined here into the unknown reliability constant c_η. Unfortunately, this overestimation factor can have arbitrary values which greatly reduces the practicality of this error estimate for the estimation of the optimal number of samples. Note, that any mistake in the error estimate is squared with regard to the number of samples used as seen in Equation (2.1). Thus, further steps need to be taken in order to make this approach feasible.

The idea is to compute some coarse solution $\mathbf{E}_{N_{\mathrm{init}}}^{\mathrm{MC}}[Q(u_H)]$ and its error estimators $\mathbf{E}_{N_{\mathrm{init}}}^{\mathrm{MC}}[c_\eta \eta_H]$ and reconstruct c_η with the help of the finer solution $\mathbf{E}_{N_{\mathrm{init}}}^{\mathrm{MC}}[Q(u_h)]$. Indeed we have $\mathbf{E}[|Q(u) - Q(u_H)|] \leq \mathbf{E}[c_\eta \eta_H]$ and thus

$$c_\eta \geq \frac{\mathbf{E}[|Q(u) - Q(u_H)|]}{\mathbf{E}[\eta_H]} \approx \frac{\mathbf{E}_{N_{\mathrm{init}}}^{\mathrm{MC}}[|Q(u_h) - Q(u_H)|]}{\mathbf{E}_{N_{\mathrm{init}}}^{\mathrm{MC}}[\eta_H]} \coloneqq \tilde{c}_\eta.$$

The approximation $\tilde{c}_\eta$ can now be used to balance the two error contributions in Equation (2.1) which gives the approximation for the number of samples in the Monte Carlo method

$$N \approx \tilde{c}_\eta^2 \frac{\mathbf{Var}_{N_{\mathrm{init}}}^{\mathrm{MC}}[Q(u_h)]}{\mathbf{E}_{N_{\mathrm{init}}}^{\mathrm{MC}}[\eta_h]^2}.$$

Remark 3.21 The trade-off for the extra solutions on the coarse levels is well worth the effort as even relatively small deviations in the number of samples lead to some great impact on the computational cost in order to meet some error tolerance. The same effect leads to insufficient sampling in practice if one ignores the effect of the constant c_η, that is one assumes $c_\eta = 1$.

For the choice of the coarse level $\mathcal{T}_H$ it is prudent to choose it independently of h and rather coarsely. It only needs to be large enough such that the asymptotic phase of the error indicators is already reached. In practice, this has to be done heuristically with numerical tests or expert knowledge. The numerical simulations

in Chapter 5 indicate, that deterministic calculations are enough to determine the estimators' behaviors.

For the coarse levels, where the above approach is not yet feasible, a fixed number of samples N_{init} is used to approximate the solution. We assume that on those coarse levels the spatial error will dominate the stochastic error and hence has little or no influence on the overall error.

Ansatz with Multilevel Correction Terms

As stated in Equation (2.7) the number of samples is chosen up to some constant such that $N_\ell \simeq \sqrt{\mathcal{C}_\ell^{-1} \mathbf{Var}[Y_\ell]}$ where $\mathcal{C}_\ell$ is the cost it takes to compute one sample on level ℓ. Given enough levels in the multilevel Monte Carlo method, that is $L > 1$ we can leverage the correction terms Y_ℓ on the levels to approximate the error on the last level L with some hierarchical estimator. We start of with the error decomposition in Equation (2.5) which reads for the quantity of interest

$$\mathbf{E}\left[\left(\mathbf{E}_{\boldsymbol{N}}^{\mathrm{ML}}[Q(u_L)] - \mathbf{E}[Q(u)]\right)^2\right] = \sum_{\ell=0}^{L} N_\ell^{-1} \mathbf{Var}[Y_\ell] + \left(\mathbf{E}[Q(u_L) - Q(u)]\right)^2 .$$

The aim is to balance the two error contributions. Using the discrete Monte Carlo variance from Definition 2.9 for some initial number of samples N_{init} the first term is known up to the number of samples. In the next step we will reconstruct the error

$$\begin{aligned} e_k &\coloneqq \left|\mathbf{E}[Q(u_k)] - \mathbf{E}[Q(u)]\right| \\ &\leq \left|\mathbf{E}_{\boldsymbol{N}}^{\mathrm{ML}}[Q(u_k)] - \mathbf{E}[Q(u)]\right| + \left|\mathbf{E}_{\boldsymbol{N}}^{\mathrm{ML}}[Q(u_k)] - \mathbf{E}[Q(u_k)]\right| \end{aligned}$$

for multilevel estimators with $k = 0, \dots, L-1$ levels. Neglecting the stochastic error on the coarse levels, we get

$$\begin{aligned} e_k &\lessapprox \left|\mathbf{E}_{\boldsymbol{N}}^{\mathrm{ML}}[Q(u_k)] - \mathbf{E}_{\boldsymbol{N}}^{\mathrm{ML}}[Q(u_L)]\right| + \left|\mathbf{E}_{\boldsymbol{N}}^{\mathrm{ML}}[Q(u_L)] - \mathbf{E}[Q(u)]\right| \\ &\approx \left|\sum_{\ell=k+1}^{L} \mathbf{E}_{N_\ell}^{\mathrm{MC}}[Y_\ell]\right| =: \tilde{e}_k \end{aligned}$$

given $\left|\mathbf{E}_{\boldsymbol{N}}^{\mathrm{ML}}[Q(u_L)] - \mathbf{E}[Q(u)]\right| \ll \left|\mathbf{E}_{\boldsymbol{N}}^{\mathrm{ML}}[Q(u_k)] - \mathbf{E}_{\boldsymbol{N}}^{\mathrm{ML}}[Q(u_L)]\right|$ as we assume that the error on the coarser meshes is dominating. With the assumptions from the beginning of this section we have the asymptotic convergence rate $e_k \lesssim h_k^\alpha$ and hence we can reconstruct this α using linear regression over the points $(\log(h_k), \log(\tilde{e}_k))$ as sketched in Figure 3.1. With the reconstructed errors on the coarse levels and the

convergence rate, we can define an extrapolated approximation of the error e_L as $\dot{e}_L \coloneqq h_L^{\alpha} h_k^{-\alpha} \tilde{e}_k$. Now, the optimal number of samples on each level is given for some constant c_N as

$$N_\ell = c_N \sqrt{\mathbf{Var}[Y_\ell] / \mathcal{C}_\ell} \tag{3.15}$$

where $\mathcal{C}_\ell$ is the computational effort required for one sample on that level. The balancing of the two error contributions as well as inserting Equation (3.15) gives the constraint $\sum_{\ell=0}^{L} c_N^{-1} \sqrt{\mathcal{C}_\ell \, \mathbf{Var}[Y_\ell]} \approx \dot{e}_L^2$ and hence we define

$$\dot{c}_N \coloneqq \dot{e}_L^{-2} \sum_{\ell=0}^{L} \sqrt{\mathcal{C}_\ell \, \mathbf{Var}[Y_\ell]}$$

which, together with Equation (3.15), gives an approximation of N_ℓ.

Ansatz with Probabilistic Bounds

The bounds from Section 3.2 provide the opportunity to choose the number of samples based on variables computed in the initial approximation phase in the multilevel Monte Carlo algorithm. The method presented below is an attempt to reconstruct the necessary information from the computed upper and lower bounds using those for an error estimate. As with the last ansatz, the goal is to balance the two error contributions in Equation (2.5). The variance terms in this decomposition will be approximated with an initial number of samples. It remains to treat the last term. From the deterministic bounds in Section 3.1 we have

$$\mathbf{E}\left[\eta_\ell^-\right] \leq \mathbf{E}[Q(u) - Q(u_\ell)] \leq \mathbf{E}\left[\eta_\ell^+\right].$$

Now consider some arbitrary spatial approximation $\mathbf{E}[\xi]$ of $\mathbf{E}[Q(u)]$. Then, straightforward calculations give

$$\mathbf{E}\left[\eta_\ell^- + Q(u_\ell) - \xi\right] \leq \mathbf{E}[Q(u) - \xi] \leq \mathbf{E}\left[\eta_\ell^+ + Q(u_\ell) - \xi\right]$$

and hence the error bound

$$|\mathbf{E}[Q(u) - \xi]| \leq \max\left\{\left|\mathbf{E}\left[\eta_\ell^+ + Q(u_\ell) - \xi\right]\right|, \left|\mathbf{E}\left[\eta_\ell^- + Q(u_\ell) - \xi\right]\right|\right\}.$$

The linearity of the terms and equilibration gives the minimizer $\xi_{\min} = Q(u_\ell) + \left(\eta_\ell^+ + \eta_\ell^-\right)/2$ and hence it holds

$$\mathbf{E}\left[\left|Q(u) - Q(u_\ell) - \frac{\eta_\ell^+ + \eta_\ell^-}{2}\right|\right] \leq \mathbf{E}\left[\left|\frac{\eta_\ell^+ - \eta_\ell^-}{2}\right|\right].$$

Therefore, $\tilde{e}_\ell \coloneqq \frac{1}{2}c_\eta^{-1}\,\mathbf{E}\left[\eta_\ell^+ - \eta_\ell^-\right]$ is a reasonable a posteriori guess of the mean finite element error $\mathbf{E}[|Q(u) - Q(u_\ell)|]$ for each level $\ell = 0, \dots, L$ with the constant c_η not dependent on the level as it is independent of h. We now want to choose N_ℓ such that $\sum_{\ell=0}^{L} N_\ell^{-1}\,\mathbf{Var}[Y_\ell] = c_\eta^{-2}\,\mathbf{E}\left[\left(\eta_L^+ - \eta_L^-\right)/2\right]^2$ and with Equation (3.15) we get

$$c_N = 4c_\eta^2 \frac{\sum_{\ell=0}^{L}\sqrt{\mathcal{C}_\ell\,\mathbf{Var}[Y_\ell]}}{\mathbf{E}\left[\eta_L^+ - \eta_L^-\right]^2}. \tag{3.16}$$

It remains to obtain a rough approximation of the constant c_η. Under the assumption of $L > 1$, and for this purpose, $\mathbf{E}_{\boldsymbol{N}}^{\mathrm{ML}}[Q(u_L)] \approx \mathbf{E}[Q(u)]$ we get

$$\begin{aligned}\mathbf{E}[Q(u)] - \sum_{\ell=0}^{L-1}\mathbf{E}_{N_\ell}^{\mathrm{MC}}[Y_\ell] &\approx \mathbf{E}_{\boldsymbol{N}}^{\mathrm{ML}}[Q(u_L)] - \sum_{\ell=0}^{L-1}\mathbf{E}_{N_\ell}^{\mathrm{MC}}[Y_\ell] \\ &= \mathbf{E}_{N_L}^{\mathrm{MC}}[Q(u_L) - Q(u_{L-1})]\,.\end{aligned} \tag{3.17}$$

Here, the second term on the left-hand side represents an $L-1$ multilevel Monte Carlo estimate on level $L-1$ and thus we can estimate the error on this level as above. In order to compute the overestimation factor c_η of our guess $\tilde{e}_{L-1}$, we start by comparing the guess and the approximation from Equation (3.17) like

$$\mathbf{E}_{N_L}^{\mathrm{MC}}[Q(u_L) - Q(u_{L-1})] = c_\eta^{-1}\,\mathbf{E}\left[\frac{\eta_{L-1}^+ - \eta_{L-1}^-}{2}\right]$$

and hence we arrive at the rough approximation using the notations for Y_ℓ and $Z_\ell^\pm$ from Equation (2.3) and Corollary 3.19

$$c_\eta = \frac{\sum_{\ell=0}^{L-1}\mathbf{E}\left[Z_\ell^+\right] - \sum_{\ell=0}^{L-1}\mathbf{E}\left[Z_\ell^-\right]}{2\,\mathbf{E}_{N_L}^{\mathrm{MC}}[Y_L]} \tag{3.18}$$

by applying $(L-1)$-level Monte Carlo estimators $\mathbf{E}_{\boldsymbol{N}_{\mathrm{init}}}^{\mathrm{ML}}\left[\eta_{L-1}^\pm\right]$ for the expected values of the bounds with the initial N_{init} samples on each level. Here, we assume, without the introduction of a new notation, the vector $\boldsymbol{N}_{\mathrm{init}}$ to always have the required length identical to the number of levels of the associated multilevel estimator. Finally,

we can combine Equation (3.15) with Equation (3.16) and Equation (3.18) to get an estimate for the optimal number of samples on each level by

$$N_\ell = \frac{\left(\mathbf{E}^{\mathrm{ML}}_{\boldsymbol{N}_{\mathrm{init}}}\left[\eta^+_{L-1}\right] - \mathbf{E}^{\mathrm{ML}}_{\boldsymbol{N}_{\mathrm{init}}}\left[\eta^-_{L-1}\right]\right)^2 \sum_{k=0}^{L} \sqrt{\mathcal{C}_k \, \mathbf{Var}^{\mathrm{MC}}_{N_{\mathrm{init}}}[Y_k]}}{\left(\mathbf{E}^{\mathrm{ML}}_{\boldsymbol{N}_{\mathrm{init}}}\left[\eta^+_{L}\right] - \mathbf{E}^{\mathrm{ML}}_{\boldsymbol{N}_{\mathrm{init}}}\left[\eta^-_{L}\right]\right)^2 \mathbf{E}^{\mathrm{MC}}_{N_{\mathrm{init}}}[Y_L]^2} \sqrt{\mathcal{C}_\ell^{-1} \, \mathbf{Var}^{\mathrm{MC}}_{N_{\mathrm{init}}}[Y_\ell]}$$

through the approximation of the variances.

Ansatz with Error Estimators

In the following, we will adapt the idea to exploit the error estimators from the simple ansatz for the Monte Carlo method to obtain sufficiently accurate numbers of samples $\boldsymbol{N}$ for the multilevel Monte Carlo method. The first step is again to find an adequate approximation of the error $\tilde{e}_k$ on each level. Consider the error estimators $\eta_\ell = \eta_{MS}(u_\ell, z_\ell)$ or $\eta_\ell = \eta_{OP}(u_\ell, z_\ell)$ for the levels $\ell = 0, ..., L$. For the last term in the error decomposition we obtain Equation (2.5) the upper bound

$$\mathbf{E}[Q(u_L) - Q(u)]^2 \leq \mathbf{E}[|Q(u_L) - Q(u)|]^2 \leq \mathbf{E}[c_\eta \eta_L] \,. \tag{3.19}$$

As it is the case for the Monte Carlo ansatz, we need to obtain reliable approximations of the asymptotic constant c_η. In order to do so, we reconstruct hierarchically the error for coarser multilevel estimates with $k = 0, \ldots, L-1$ levels and $\boldsymbol{N}_{\mathrm{init}} = (N_{\mathrm{init}}, \ldots, N_{\mathrm{init}})$ initial samples as follows

$$\begin{aligned} \mathbf{E}[Q(u)] - \mathbf{E}^{\mathrm{ML}}_{\boldsymbol{N}}[Q(u_k)] &\approx \mathbf{E}^{\mathrm{ML}}_{\boldsymbol{N}}[Q(u_L)] - \sum_{\ell=0}^{k} \mathbf{E}^{\mathrm{MC}}_{N_{\mathrm{init}}}[Y_\ell] \\ &= \sum_{\ell=k+1}^{L} \mathbf{E}^{\mathrm{MC}}_{N_{\mathrm{init}}}[Y_\ell] =: \tilde{e}_k. \end{aligned} \tag{3.20}$$

Next, we will define a multilevel Monte Carlo estimate of the mean error estimator $\mathbf{E}[\eta_L]$. Following the naming conventions Equation (2.3) define the terms $Z_0^\circ := \eta_0$ and $Z_\ell^\circ := \eta_\ell - \eta_{\ell-1}$ for $\ell = 1, \ldots, L$. The multilevel Monte Carlo estimator for the expected value of the error estimate on level L is defined as

$$\mathbf{E}^{\mathrm{ML}}_{\boldsymbol{N}}[\eta_L] = \sum_{\ell=0}^{L} \mathbf{E}^{\mathrm{MC}}_{N}[Z_\ell^\circ] \,.$$

The mean overestimation constant c_η is now approximated for each $k = 0, \ldots, L$ by $c_\eta^k := \mathbf{E}^{\mathrm{ML}}_{\boldsymbol{N}_{\mathrm{init}}}[\eta_k] \, \tilde{e}_k^{-1}$ with $\tilde{e}_k$ as defined in Equation (3.20). We define the approximation

of the constant as the mean value over all k as $\tilde{c}_\eta \coloneqq \frac{1}{L-1}\sum_{k=0}^{L-1} c_\eta^k$. The final step is to compute the constant c_N in Equation (2.7) such that the two error contributions are balanced as $\sum_{\ell=0}^{L} N_\ell^{-1}\,\mathbf{Var}[Y_\ell] = c_\eta^{-2}\,\mathbf{E}[\eta_L]^2$. Inserting Equation (2.7) leads to $\sum_{\ell=0}^{L} c_N^{-1}\sqrt{\mathcal{C}_\ell\,\mathbf{Var}[Y_\ell]} = c_\eta^{-2}\,\mathbf{E}[\eta_L]^2$ and thus to the approximation of c_N as

$$\tilde{c}_N \coloneqq \tilde{c}_\eta^2 \sum_{\ell=0}^{L} \frac{\sqrt{\mathcal{C}_\ell\,\mathbf{Var}^{\mathrm{MC}}_{N_{\mathrm{init}}}[Y_\ell]}}{\mathbf{E}^{\mathrm{ML}}_{\boldsymbol{N}_{\mathrm{init}}}[\eta_L]^2}$$

which gives a heuristic for the number of samples on each level.

A Practical Approach

All of the above heuristics rely on the asymptotic behavior of different quantities. For uniform meshes this is usually not a problem as the preasymptotic phase is short enough to be ignored. On the other hand, this phase can be much longer for adaptive algorithms such that extra care needs to be taken for coarse meshes. Studying closely related deterministic problems can provide a good overview for the behavior of the different quantities used in the heuristics.

Furthermore, the adaptive algorithm suggested by [MS09] based on η_{MS} introduces some stepping, as in every step it refines either for the primal or the dual problem. Given an example where these refinements do not coincide locally, a primal refinement step might even lead to a short divergent phase in the quantity of interest. In contrast, a dual refinement might lead to a short phase with a much higher than expected convergence rate. As a consequence strong over- or underestimation of the spatial error might occur in the heuristics resulting in severe over- or undersampling.

Combinations of the above variants for the multilevel Monte Carlo method can mitigate these problems to some extent as the error estimators express different enough behavior compared to the multilevel control terms, as they are derived from different levels. In practice, the minimum N_ℓ predicted by the correction terms and the error estimators proved to be the most reliable. The probabilistic bounds, on the other hand, did not provide good enough results to be practically feasible. This is due to their very long preasymptotic phase as can be seen in Chapter 5.

In the approach with error estimates, it is possible to use the global or element-wise Cauchy–Schwarz inequality, to obtain global estimates. However, numerical testing for the problems in Chapter 5 indicates better performance of the local inequality for the reconstruction of the convergence rate α.

4 An Alternative Approach for Higher Dimensions

4.1 Stochastic Representation of the Stochastic Problem

In general, the solution process for the stochastic version of Problem (1.1) requires the discretization of the spatial dimensions and of the stochastic domain. Stochastic Galerkin methods apply some finite-dimensional basis to both of these. For example, the spatial domain might be discretized by standard finite element functions whereas the stochastic domain is approximated with some polynomial basis functions like Hermite polynomials.

This approach however is severely limited as it suffers from the curse of dimensionality. That is, with increasing number of dimensions in either domain, the computational cost increases exponentially. This might be tolerable for 3D finite element methods but practical applications require a substantial amount of basis functions in the stochastic domain, such that the resulting linear system becomes prohibitively large.

The Monte Carlo methods and its variants from Chapter 2 compensate for this problem by replacing exact integration in the random space with stochastic integration. The following method will extend this idea to both domains with the help of a representation of Problem (1.1) as some stochastic ordinary differential equation (SDE). This is actually a natural approach to this model as the Darcy equation is also the homogenization of a SDE describing the transport of particles in porous media.

The key is to establish a proper connection between the two formulations. The Feynman–Kac formula establishes a link between parabolic partial differential equations and stochastic differential equations. Indeed, the Darcy model problem regarded here resembles a special case. A more general approach is given in [ABE+15a].

This framework allows us to compute the mean of the solution at arbitrary points $x \in D$ in the domain. It allows to focus the computational effort in the most important parts of the product space $D \times \Omega$ giving optimal sparsity with respect to the error density in that space. This is not possible with the methods presented in Chapter 2 as the sampling has to occur on the whole domain. The adaptive meshes and the multilevel Monte Carlo method can counter this drawback up to some degree but are limited to the separate subspaces D and Ω.

In [ABE+15a] this approach is used in conjunction with some stochastic polynomial regression scheme to construct the mean solution on the whole domain D. This allows for a dimension-independent convergence rate but does not exploit the possibility

of local and independent adaptivity for both D and Ω. In this work, we will present an interpolation method which links the adaptive tools from the finite element framework with locally adaptive sampling to construct an optimal representation of the mean solution on an adaptive mesh with minimal sampling cost. However, the advantage of the independence of spacial adaptivity and sampling forfeits the dimension-independent convergence rate.

To achieve this, three steps are necessary. First, a local Monte Carlo estimator for the mean solution and its variance is established. Then, a global mean solution is constructed via interpolation on some mesh $\mathcal{T}$. Finally, an adaptive scheme is conceived to generate a sequence of estimators on adaptively generated meshes which gives the desired solution for a given error tolerance at minimal cost. The same exploratory interpolation approach is presented in [ABE$^+$15b].

Pointwise Approximation with an Ordinary Stochastic Differential Equation

For this approach we assume $D \subset \mathbb{R}^d$ for $d \in \mathbb{N}$. All input data as well as the domain boundary are assumed to be smooth enough, such that the problem in Equation (1.1) yields a unique solution. Furthermore we extend this problem by replacing the homogeneous boundary data with the Dirichlet condition $u = u_D$ on Γ. Neumann boundaries are feasible as well with this ansatz but we omit them for the sake of simplicity in the presentation.

As stated above, the central tool in this approach is the Feynman–Kac formula which connects stochastic processes with partial differential equations. Its application in the context of ground water flow is somewhat natural as the Darcy equation can be considered as a homogenized formulation of some underlying stochastic process in which water particles traverse the domain D on paths with given probabilities. First mentioned in [Kac49], a text book introduction on this topic is given in [MT04]. Finally, note that this method offers a more general approach but it is presented in a limited manner for simplicity. For the complete overview we refer to the aforementioned text book.

We characterize the problem at hand by the following stochastic differential equation and adjoint processes given at some point $x \in D \subset \mathbb{R}^d$ for the processes $X_t =$

$X_t^x, Y_t = Y_t^x$, and $Z_t = Z_t^x$

$$\begin{aligned} \mathrm{d}X_t &= b(X_t)\ \mathrm{d}t + \sigma(X_t)\ \mathrm{d}W_t, \\ X_0 &= x, \\ Y_t &\coloneqq \exp\left(\int_0^t c_1(X_s)\ \mathrm{d}s\right), \\ Z_t &\coloneqq \int_0^t c_2(X_s) Y_s\ \mathrm{d}s. \end{aligned} \tag{4.1}$$

Here, W is a d-dimensional standard Brownian motion in the probability space $(\Omega, \mathcal{F}, \mathsf{P})$. The operators $b : \mathbb{R}^d \to \mathbb{R}^d$ and $\sigma : \mathbb{R}^d \to \mathbb{R}^{d\times d}$ are assumed to be uniformly Lipschitz continuous and $c_1, c_2 : \mathbb{R}^d \to \mathbb{R}$. This results in a unique strong solution process X in $[0,T]$ for every $T > 0$ which is adapted to the filtration generated by W and satisfies

$$\int_0^T \mathbf{E}\left[|X_t|^2\right]\ \mathrm{d}t < \infty \qquad \text{and}$$
$$X_t = x + \int_0^t b(X_s)\ \mathrm{d}s + \int_0^t \sigma(X_s)\ \mathrm{d}W_s,$$

where the operator $\mathrm{d}W_s$ is considered in the Itô-sense. With $a(x) \coloneqq \sigma(x)^T\sigma(x)$ we get the infinitesimal generator L of the problem in Equation (4.1) as the differential operator

$$Lf(x) = \sum_{i=1}^d b_i(x)\frac{\partial}{\partial x_i}f(x) + \frac{1}{2}\sum_{i,j=1}^d a_{ij}(x)\frac{\partial^2}{\partial x_i \partial x_j}f(x). \tag{4.2}$$

With the restriction of $F : [0,T]\times\mathbb{R}^d \to \mathbb{R}$ to be $\mathcal{C}^1$ time-continuous and $\mathcal{C}^2$ space-continuous Itô's formula gives

$$\begin{aligned} F(t, X_t) = F(0,x) &+ \int_0^t \left(\frac{\partial}{\partial t}F(s, X_s) + LF(s, X_s)\right)\ \mathrm{d}s \\ &+ \int_0^t \sum_{i=1}^d \sum_{j=1}^d \frac{\partial}{\partial x_i}F(s, X_s)\sigma_{ij}(X_s)\ \mathrm{d}W_s^j. \end{aligned}$$

Furthermore, $\mathbf{E}\left[\int_0^t u(s)\ \mathrm{d}W_s\right] = 0$ because $t \mapsto \int_0^t u(s)\ \mathrm{d}W_s$ is a martingale on $[0,T]$ for any process $u(s)$ adapted to the filtration generated by W under the assumption $\int_0^T \mathbf{E}\left[|u(t)|^2\right]\ \mathrm{d}t < \infty$. The solution to the stationary Problem (1.1) is derived as the limit $t \to \infty$ from some corresponding time-dependent Cauchy

problem defined with $(t, x) \in [0, \infty[\times \mathbb{R}^d$ as

$$\begin{aligned} \frac{\partial}{\partial t} u(t,x) &= L\, u(t,x) + c_1(x) u(t,x) + c_2(x), \\ u(0,x) &= u_D(x). \end{aligned} \tag{4.3}$$

Given the solution $u \in \mathcal{C}^{1,2}([0, \infty[\times \mathbb{R}^d)$ it holds for $t \geq 0, x \in \mathbb{R}^d$ the identity

$$u(t,x) = \mathbf{E}[u_D(X_t^x) Y_t^x + Z_t^x]\,.$$

For the special case of $c_1 = c_2 = 0$ in Equation (4.3), fixing $T > 0$ gives $Y_t^x \equiv 1$, $Z_t^x \equiv 0$ and for $F(t,x) =: u(T-t,x)$, Itô's formula gives

$$\begin{aligned} u(0, X_T^x) - u(T,x) &= \int_0^T \left(-\frac{\partial}{\partial t} u(T-t, X_t^x) + L\, u(T-t, X_t^x) \right) \mathrm{d}t \\ &\quad + \int_0^T \sum_{i=1}^d \sum_{j=1}^d \frac{\partial}{\partial x_i} u(T-t, X_t^x) \sigma_{ij}(X_t^x)\ \mathrm{d}W_t^j \\ &= \int_0^T \sum_{i=1}^d \sum_{j=1}^d \frac{\partial}{\partial x_i} u(T-t, X_t^x) \sigma_{ij}(X_t^x)\ \mathrm{d}W_t^j. \end{aligned}$$

The general case follows similarly. Taking the expected value of the above equation results in $u(T,x) = \mathbf{E}[u_D(X_T^x)]$.

To compute the actual value of the solution u at the given point x we can utilize the standard Monte Carlo method with N samples to approximate with a rate of $N^{1/2}$. This requires for each sample to approximate the diffusion process X_t^x by some discretization X_M. In this scenario we consider the explicit Euler method for this purpose. Define some time sequence $\mathcal{T}_M$ with $0 = t_0 < \ldots < t_M = t$ with $\Delta t_m \coloneqq t_m - t_{m-1}$ and $\Delta W_{t_m} \coloneqq W_m - W_{t_{m-1}}$ for $m = 1, \ldots, M$ and thus

$$X_m \coloneqq X_{m-1} + b(X_{m-1}) \Delta t_m + \sigma(X_{m-1}) \Delta W_m.$$

This method results in strong convergence with $\mathbf{E}[|X_t - X_M|] \leq C h^{1/2}$ under weak assumptions and C independent on M. Regarding weak convergence, the explicit Euler scheme often gives $|\mathbf{E}[v(X_t)] - \mathbf{E}[v(X_M)]| \leq Ch$ for any test function $v : \mathbb{R}^d \to \mathbb{R}$ and constant C as above. The stationary elliptic problem is represented as the limit for $t \to \infty$ of the parabolic problem discussed above. This results in

$$\begin{aligned} L\, u(x) + c_1(x) u(x) + c_2(x) &= 0 && \text{in } D, \\ u(x) &= u_D(x) && \text{on } \Gamma_D, \end{aligned}$$

which gives the solution identity for $x \in D$

$$u(x) = \mathbf{E}[u_D(X_\tau^x)Y_\tau^x + Z_\tau^x] \tag{4.4}$$

for the random stopping time $\tau = \tau_x \coloneqq \inf\{t \geq 0 \mid X_t^x \in D\}$. The stopping time τ is the time the diffusion process leaves the domain. Besides the error from the discretization of X_t^x by X_M, an additional error is introduced during the approximation of the exit time τ by some discrete value τ_M as the discrete path usually crosses the boundary of the domain in between two grid points of X_M. Once again, the resulting convergence rate is equal to $1/2$. Nevertheless, adaptive time stepping techniques can improve the practically observed behavior back to linear convergence. Here, we use the distance to the closest boundary as the indicator as shown in Algorithm 4.1.

The second step concerns the integration of a random field into the stochastic representation of Problem Equation (1.1). The main idea here is to have two separate random spaces: one for the random diffusion field $(\Omega, \mathcal{F}, \mathsf{P})$ as defined in Section 1.3 and one for the diffusion process $(\Omega_X, \mathcal{F}_X, \mathsf{P}_X)$. We will now consider the product space $(\Omega, \mathcal{F}, \mathsf{P}) \otimes (\Omega_X, \mathcal{F}_X, \mathsf{P}_X)$ of those two. Similar to the deterministic model, we arrive at the representation for $\omega \in \Omega$ and $\omega_X \in \Omega_X$

$$\begin{aligned} u(\omega, x) &= \int_{\Omega_X} u_D\left(\omega, X^x_{\tau(\omega,\omega_X)}(\omega, \omega_X)\right) Y^x_{\tau(\omega,\omega_X)}(\omega, \omega_X)\ \mathsf{P}_X(\mathrm{d}\omega_X) \\ &\qquad + \int_{\Omega_X} Z^x_{\tau(\omega,\omega_X)}(\omega, \omega_X)\ \mathsf{P}_X(\mathrm{d}\omega_X) \\ &= \mathbf{E}[u_D(X_\tau^x)Y_\tau^x + Z_\tau^x \mid \omega]\,. \end{aligned}$$

Compared to the Monte Carlo methods from Chapter 2 this approach adds contribution to the variance with regard to the computation of the expected value in the random product space $\mathbf{E}[u(x)] = \mathbf{E}[u_D(X_\tau^x)Y_\tau^x + Z_\tau^x]$ as it is due to the law of total variance

$$\begin{aligned} \mathbf{Var}[u(x)] &= \mathbf{Var}[u_D(X_\tau^x)Y_\tau^x + Z_\tau^x] \\ &= \mathbf{Var}[\mathbf{E}[u_D(X_\tau^x)Y_\tau^x + Z_\tau^x \mid \omega]] + \mathbf{E}[\mathbf{Var}[u_D(X_\tau^x)Y_\tau^x + Z_\tau^x \mid \omega]] \\ &= \mathbf{Var}[u(x)] + \mathbf{E}[\mathbf{Var}[u_D(X_\tau^x)Y_\tau^x + Z_\tau^x \mid \omega]]\,. \end{aligned}$$

The last term on the right-hand side describes the extra variance and hence requires additional samples in the Monte Carlo method. For a small number of dimensions d it is not worth the extra effort but for larger dimensions the discretization cost per sample grows for the methods in Chapter 2.

The third step in the discretization is the approximation of the expected value in Equation (4.4) with a Monte Carlo estimator. Multilevel Monte Carlo methods are

also possible but require the construction of two related realizations of X_M for each sample on the individual levels as described in [Gil08]. Here, we will limit ourselves to the Monte Carlo estimator which is defined as

$$\mathbf{E}^{\mathrm{MS}}_{M,N}[u(x)] \coloneqq \frac{1}{N}\sum_{i=1}^{N} u_D\Big(X^x_\tau(\omega_i)\Big)Y^x_\tau(\omega_i) + Z^x_\tau(\omega_i), \tag{4.5}$$

where $(\omega_i)_{i=1}^N$ is a set of samples drawn from the random space $(\Omega, \mathcal{F}, \mathsf{P})$ and M describes the number of steps for the discrete diffusion process. Note that this random space represents both the diffusion process X_M and the random field κ from the initial problem definition.

In the case of the Darcy Model Problem (1.1), the parameters in Equation (4.3) can be directly deduced as $c_1 \equiv 0$ and $c_2 \equiv f(x)$ which results in

$$Y^x_\tau \equiv 1 \qquad \text{and} \qquad Z^x_\tau = \int_0^\tau f(X_s)\,\mathrm{d}s.$$

Furthermore it holds $a(x) = 2\kappa(x)\mathrm{I}_n$ and $b(x) = \nabla\kappa(x)$ for the parameters from Equation (4.2) where I_n is the n-dimensional identity matrix. As it holds for the normal distribution $\mathcal{N}_{0,\Delta t} \sim \sqrt{\Delta t}\mathcal{N}_{0,1}$, we get with the time steps $t_0, t_1, \ldots$ for $m = 0, \ldots, M$

$$\begin{aligned} X_0 &= x, \\ X_m &= X_{m-1} + \nabla\kappa(X_{m-1})\Delta t_m + \sqrt{2\kappa(X_{m-1})\Delta t_m}\,\mathcal{N}_{0,1}, \end{aligned}$$

where M is the last index with $X_m \in D$ for $m = 0, \ldots, M$ and $X_{M+1} \notin D$. Here, $\mathcal{N}_{0,1}$ denotes the standard normal distribution in d dimensions. The stopping position and the stopping time t_τ of the diffusion process is now approximated by the projection

$$X_\tau \coloneqq \arg\min\{\|X - X_M\| \mid X \in \partial D, X = X_M + s(X_{M+1} - X_M), s \geq 0\} \tag{4.6}$$

of X_M onto the boundary ∂D in direction of X_{M+1}. Equation (4.5) gives the estimator with a simple one-point integration rule as follows

$$\mathbf{E}^{\mathrm{MS}}_{M,N}[u(x)] \coloneqq \frac{1}{N}\sum_{i=1}^{N}\left(u_D\Big(X_\tau(\omega_i)\Big) + \sum_{m\in(0,\ldots,M,\tau)} f\Big(X_m(\omega_i)\Big)\Delta t_m\right).$$

Simple time adaptivity is applied by choosing $\Delta t_j = \mathrm{dist}(\partial D, X_{j-1})\Delta t_0$ where $\mathrm{dist}(\partial D, x) \coloneqq \min\{\|x_d - x\| \mid x_d \in \partial D\}$ is the Euclidean distance to the boundary of the domain. The whole process is sketched in Algorithm 4.1 where a simple

rectangle integration method is used and Figure 4.1 gives a sketch thereof.

Remark 4.1 During the derivation above we made rather basic and restrictive assumptions, such as uniform ellipticity. In fact, most of the arguments hold with more relaxed conditions, as time-dependent coefficients.

Extension to the whole Domain

In the following, the pointwise approximation of the solution by means of Monte Carlo estimators is extended to the whole domain using interpolation techniques. This allows to apply finite element a posteriori error control. We will define the approximate solution as follows. Consider some given mesh $\mathcal{T}_h$ with vertices $\mathcal{N}_h = (\nu_h^i)_{i=1}^{|\mathcal{N}_h|}$. We will define the discrete solution $\mathbf{E}^{\mathrm{MS}}_{\boldsymbol{M},\boldsymbol{N}}[u_h]$ with $\boldsymbol{M} = (M_i)_{i=1}^{|\mathcal{N}_h|}$ and $\boldsymbol{N} = (N_i)_{i=1}^{|\mathcal{N}_h|}$ as a P_1 function on that mesh, by setting the node values for $i = 1, \dots, |\mathcal{N}_h|$, which resemble also the coefficients in the P_1 basis, as follows

$$\mathbf{E}^{\mathrm{MS}}_{\boldsymbol{M},\boldsymbol{N}}[u_h]\,(\nu_h^i) \coloneqq \mathbf{E}^{\mathrm{MS}}_{M_i,N_i}\Big[u(\nu_h^i)\Big] \qquad \text{for } i = 1, \dots, |\mathcal{N}_h|\,.$$

This introduces three types of errors into the approximation. The first one is the stochastic approximation error originating in the Monte Carlo estimators. It can be controlled by the means from Lemma 3.15 through the numbers of samples N_i for each point. Application of the Inequality (3.12) will be necessary to extend this control to the P_1 approximation. The second error arises from the approximation of the diffusion process and is controlled by the parameters M_i. The third error contribution results from the P_1 interpolation which is approximated by the Monte Carlo estimators and is governed through the interpolation mesh parameter h.

We will consider two error representations. The first one describes a decomposition of the mean square error into three error parts resulting from the interpolation error, the discretization of the ordinary differential equation, and the sampling error in the Monte Carlo method. Define the expected value of the discrete ordinary differential equation as $\mathbf{E}\Big[u_h^M\Big] \coloneqq \mathbf{E}\Big[\mathbf{E}^{\mathrm{MS}}_{\boldsymbol{M},\boldsymbol{N}}[u_h]\Big]$. Now, it holds for the pointwise mean square error in the approximation by binomial formulas

$$\begin{aligned}
\mathbf{E}\Big[\Big(\mathbf{E}^{\mathrm{MS}}_{\boldsymbol{M},\boldsymbol{N}}[u_h] - \mathbf{E}[u]\Big)^2\Big] &= \mathbf{E}\Big[\mathbf{E}^{\mathrm{MS}}_{\boldsymbol{M},\boldsymbol{N}}[u_h]^2\Big] - 2\,\mathbf{E}\Big[\Big(\mathbf{E}^{\mathrm{MS}}_{\boldsymbol{M},\boldsymbol{N}}[u_h]\Big)\Big]\,\mathbf{E}[u] + \mathbf{E}[u]^2 \\
&= \mathbf{E}\Big[\mathbf{E}^{\mathrm{MS}}_{\boldsymbol{M},\boldsymbol{N}}[u_h]^2\Big] - \mathbf{E}\Big[u_h^M\Big]^2 + \Big(\mathbf{E}\Big[u_h^M\Big] - \mathbf{E}[u]\Big)^2 \qquad (4.7) \\
&= \frac{1}{N}\,\mathbf{Var}\Big[u_h^M\Big] + \Big(\mathbf{E}\Big[u_h^M\Big] - \mathbf{E}[u]\Big)^2\,.
\end{aligned}$$

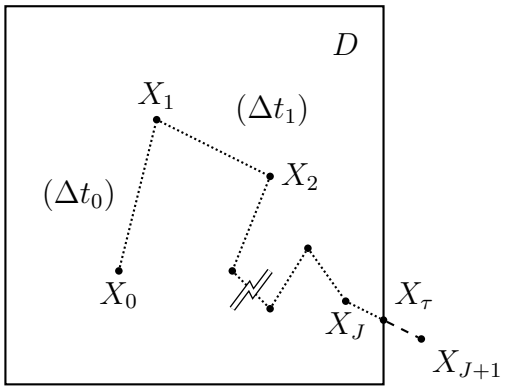


Figure 4.1: Sketch of a discrete diffusion process realization with endpoint projection and indicated step width.

In : point $x \in D$, number of samples N, initial time step Δt_0
Out : $\mathbf{E}^{\mathrm{MS}}_{M,N}[u(x)]$

for $i = 0, \ldots, N$ **do**
  $X_0 = x$
  $F = 0$
  $j = 1$
  sample $\kappa^i = \kappa(\omega_i)$ with $\omega_i \in \Omega$
  while $X_j \in D$ **do**
    $F = F + f(X_{j-1})\Delta t_{j-1}$
    $\Delta t_j = \min\{\mathrm{dist}(\partial D, X), 1\}\, \Delta t_0$
    sample Ξ from $\mathcal{N}_{0,1}$
    $X_j = X_{j-1} + \nabla\kappa^i(X_{j-1})\Delta t_j + \sqrt{2\kappa^i(X_{j-1})\Delta t_j}\Xi$
    $j = j + 1$
  compute X_τ and t_τ according to Equation (4.6)
  $F = F + f(X_\tau)\Delta t_\tau$
  $u^i = u_D(X_\tau) + F$
return $N^{-1} \sum_{i=0}^{N} u^i$

Algorithm 4.1: Point estimate algorithm to compute the estimator $\mathbf{E}^{\mathrm{MS}}_{M,N}[u(x)]$ using the simple one-point rectangle method for integration.

The second decomposition seeks to represent the error locally in the L^2 norm. In the following, we will assume $D \subseteq \mathbb{R}^2$. Higher dimensions are possible with different inequalities for the norms. Consider the element $T \in \mathcal{T}_h$ as well as the pointwise P_1 interpolation operator $\mathcal{I}_h$ on the mesh $\mathcal{T}_h$. It holds for the approximation error

$$\begin{aligned}
\left\|\mathbf{E}[u] - \mathbf{E}^{\mathrm{MS}}_{\boldsymbol{M},\boldsymbol{N}}[u_h]\right\|_{L^2(T)} &\lesssim \left\|\mathbf{E}[u] - \mathcal{I}_h\,\mathbf{E}[u]\right\|_{L^2(T)} + \left\|\mathcal{I}_h\,\mathbf{E}[u] - \mathbf{E}^{\mathrm{MS}}_{\boldsymbol{M},\boldsymbol{N}}[u_h]\right\|_{L^2(T)} \\
&\lesssim \left\|\mathbf{E}[u] - \mathbf{E}[u_h]\right\|_{L^2(T)} + \left\|\mathbf{E}[u_h] - \mathcal{I}_h\,\mathbf{E}[u]\right\|_{L^2(T)} \\
&\quad + \left\|\mathcal{I}_h\,\mathbf{E}[u] - \mathbf{E}\left[u_h^M\right]\right\|_{L^2(T)} + \left\|\mathbf{E}\left[u_h^M\right] - \mathbf{E}^{\mathrm{MS}}_{\boldsymbol{M},\boldsymbol{N}}[u_h]\right\|_{L^2(T)} \\
&\lesssim h_T \eta_T + h_T^2\,|u|_2 + |T|\,\Delta t_0 + |T| \max_{K \in \mathcal{N}(T)} \left\{ \mathbf{Var}\left[u_h^M\right]^{1/2} N_K^{-1/2} \right\}. \quad (4.8)
\end{aligned}$$

The error control η_T is the same as in Section 3.3. The first two terms are both governed by the mesh $\mathcal{T}_h$ and can be controlled through refinement as well as the adaptive algorithms from Section 3.3. The third term represents the error in the approximation of the diffusion process in the Euler scheme for the stochastic differential equation. Numerical experiments show that $\Delta t_0 \simeq h_{\min}$ is a reasonable choice in the two-dimensional case. The last term represents the Monte Carlo estimation error. It solely depends on the number of samples used for each node in the element as the variance converges to the variance of the continuous solution $\mathbf{Var}[u]$ for $M \to \infty$ and $h \to 0$.

4.2 Adaptive Algorithms

For convergence of the method presented in Section 4.1 all error components in the decomposition Equation (4.8) have to converge. For optimal convergence they should ideally converge with the same rate with respect to computational effort that has to be invested to gain the error reduction. The main idea of the algorithm is therefore to base the parameter choices on the underlying mesh. Hence, it is the goal in the following parts to define some optimal sequence of meshes $\mathcal{T}_0, \ldots, \mathcal{T}_L$ for the interpolation and then choose matching values for the other discretization parameters.

We start with an initial quasi-uniform triangulation $\mathcal{T}_0$. The first step is to calculate a discrete solution $\mathbf{E}^{\mathrm{MS}}_{\boldsymbol{M_0},\boldsymbol{N_0}}[u_h]$. The parameters $\boldsymbol{M_0}$ and $\boldsymbol{N_0}$ have to be guessed as not enough information is available on the initial level. The next step involves the calculation of the finite element error estimator η_0. As can be seen in Equation (4.8), the estimator only covers the first term $\left\|\mathbf{E}[u] - \mathbf{E}[u_0]\right\|_{L^2(T)}$ and thus the parameters

$\boldsymbol{M_0}$ and $\boldsymbol{N_0}$ have to be chosen such that

$$h_T \eta_0|_T \gg \left\| \mathcal{I}_h \, \mathbf{E}[u] - \mathbf{E}\left[u_h^M\right] \right\|_{L^2(T)} + |T| \max_{K \in \mathcal{N}(T)} \left\{ \mathbf{Var}\left[u_h^M\right]^{1/2} N_K^{-1/2} \right\}.$$

The reason is that the finite element solution is bounded by the interpolator but the discrete solutions' stochastic properties introduce oscillations of length h and amplitude as seen above. These are theoretically unbounded and artificially increase the finite element error estimator. As refinement would not reduce these oscillations we have to limit them by the means of the central limit theorem such that we get reliable mesh refinements. In fact, the stochastic error term from Equation (4.8) for each element has to be smaller than the smallest error estimator $\eta^{\mathrm{loc}}(T)$ chosen for the refinement set $\mathcal{M}$ in Equation (3.14), that is

$$\min_{T \in \mathcal{M}} \eta_{0,T}^{\mathrm{loc}} > |T| \max_{K \in \mathcal{N}(T)} \left\{ \mathbf{Var}\left[u_h^M\right]^{1/2} N_K^{-1/2} \right\} + \left\| \mathcal{I}_h \, \mathbf{E}[u] - \mathbf{E}\left[u_h^M\right] \right\|_{L^2(T)}. \quad (4.9)$$

This constraint also includes the error of the approximation in the Euler scheme but it is deterministic, smooth, and domain-wide. As a result, it alters the error estimator only to some minor extent. The derived refinement $\mathcal{T}_1$ of the mesh $\mathcal{T}_0$ is the basis for the next level and the process is repeated. Heuristics from below allow to balance the parameters based on educated guesses of the error components. This however is only possible if there are at least three meshes, that is $L \geq 3$. Section 4.3 will cover some approaches to this topic.

4.3 Heuristics for the Parameters

The error decomposition in Equation (4.8) results in three components that need to be balanced for guaranteed and optimal convergence. Therefore, we aim to find good estimates for the convergence rates with respect to the relevant parameters and then extrapolate good error estimates to the next level. With these we can determine the necessary parameters that fulfill our balancing requirements. The first goal is the approximation of the convergence rate of the interpolation error introduced by the mesh and controlled through the parameter α. Let h be the minimal inradius over all the triangles of the triangulation $\mathcal{T}_h$ and suppose we have some $\alpha > 0$ such that

$$\left\| \mathbf{E}[u] - \mathbf{E}[\mathcal{I}_h u] \right\|_{L^2(D)} \lesssim h^\alpha.$$

Standard finite element theory gives us $\left\| \mathbf{E}[u] - \mathbf{E}[\mathcal{I}_h u] \right\|_{L^2(D)} \sim \left\| \mathbf{E}[u] - \mathbf{E}[u_h] \right\|_{L^2(D)}$ and thus with some efficient and reliable error estimate $\eta_h \sim \left\| \mathbf{E}[u] - \mathbf{E}[u_h] \right\|_{L^2(D)}$ we get $\eta_h \sim h^\alpha$. We exploit this property to gauge the parameter α as error

estimators have smoothing properties and thus in practice exhibit a behavior closer to a monotonic convergence. For the estimation α_L of α we do a linear regression over the data points $(\log(h_\ell), \log(\eta_\ell))_{\ell=1,\dots,L}$.

Next we estimate the expected spatial error on the next level $L+1$. We exploit the triangle inequality for $\ell = 1, \dots, L$ to get

$$\|\mathbf{E}[u] - \mathbf{E}[u_\ell]\|_{L^2(D)} \leq \|\mathbf{E}[u] - \mathbf{E}[u_L]\|_{L^2(D)} + \|\mathbf{E}[u_L] - \mathbf{E}[u_\ell]\|_{L^2(D)} .$$

The last term on the right-hand side is computable and it remains to estimate the error on level L. We assume $\|\mathbf{E}[u] - \mathbf{E}[u_L]\|_{L^2(D)} \ll \|\mathbf{E}[u_L] - \mathbf{E}[u_\ell]\|_{L^2(D)}$ for the coarser levels $\ell < L$ and thus we conclude

$$\|\mathbf{E}[u] - \mathbf{E}[u_\ell]\|_{L^2(D)} \approx \|\mathbf{E}[u_L] - \mathbf{E}[u_\ell]\|_{L^2(D)}$$

for each level $\ell = 1, \dots, L-1$. As we already have a good approximation of the expected convergence rate α_L we can now find approximations of the error $\|\mathbf{E}[u] - \mathbf{E}[u_L]\|_{L^2(D)}$ with the help of the errors on the coarser levels through

$$\|\mathbf{E}[u] - \mathbf{E}[u_\ell]\|_{L^2(D)} \sim h_\ell^\alpha \qquad \text{and} \qquad \|\mathbf{E}[u] - \mathbf{E}[u_L]\|_{L^2(D)} \sim h_L^\alpha$$

which gives asymptotically for each $\ell = 1, \dots, L-1$ the identity

$$\frac{\|\mathbf{E}[u] - \mathbf{E}[u_L]\|_{L^2(D)}}{\|\mathbf{E}[u] - \mathbf{E}[u_\ell]\|_{L^2(D)}} = \frac{h_L^\alpha}{h_\ell^\alpha}$$

and hence we can construct the approximation

$$\|\mathbf{E}[u] - \mathbf{E}[u_L]\|_{L^2(D)} \approx \frac{h_L^\alpha}{h_\ell^\alpha} \|\mathbf{E}[u_L] - \mathbf{E}[u_\ell]\|_{L^2(D)} .$$

We define now our estimate $\tilde{e}_L$ for the error $\|\mathbf{E}[u] - \mathbf{E}[u_L]\|_{L^2(D)}$ on level L as the arithmetic mean of the different extrapolations from the coarser levels $\ell = 1, \dots, L-1$ as

$$\tilde{e}_L \coloneqq \frac{1}{L-1} \sum_{\ell=1}^{L-1} \frac{h_L^\alpha}{h_\ell^\alpha} \|\mathbf{E}[u_L] - \mathbf{E}[u_\ell]\|_{L^2(D)} .$$

The same technique is used to gauge the expected error on level $L+1$. The mesh $\mathcal{T}_{L+1}$ is already known as the adaptive algorithm from Section 4.2 has realized the refinement. Thus, the parameter h_{L+1} can be used to generate the extrapolation

$\dot{e}_{L+1}$ with the same levels $L = 1, \dots, L-1$

$$\dot{e}_{L+1} \coloneqq \frac{1}{L-1} \sum_{\ell=1}^{L-1} \frac{h_{L+1}^{\alpha}}{h_{\ell}^{\alpha}} \left\| \mathbf{E}[u_L] - \mathbf{E}[u_\ell] \right\|_{L^2(D)} .$$

The next step is to balance the expected Monte Carlo error with the extrapolated spatial error. For the adaptive algorithm in Section 4.2 it is crucial to keep the Monte Carlo error well below the spatial error as otherwise this error gets picked up by the estimator which results in wrong refinement and thus unstable behavior and suboptimal convergence or in no convergence at all. Hence, we introduce the balancing factor δ which describes the desired relation between the two errors in Equation (4.7) as follows

$$\delta^2 = \frac{\left(\mathbf{E}[u] - \mathbf{E}\left[u_h^M\right] \right)^2}{N^{-1} \mathbf{Var}[u_h^M]}. \tag{4.10}$$

A choice of $\delta = 1$ would lead to equality. For some constant c_N^ℓ for each level $\ell = 1, \dots, L$ we set the numbers of samples in the vertices $\mathcal{N}_\ell = (\nu_\ell^i)_{i=1}^{|\mathcal{N}_\ell|}$ as

$$N_\ell^i \coloneqq c_N^\ell \, \mathbf{Var}\left[u_\ell^M(\nu_\ell^i) \right] \qquad \text{for } i = 1, \dots, |\mathcal{N}_\ell| . \tag{4.11}$$

It is now the aim to choose c_N^{L+1} wisely so Equation (4.10) will be fulfilled for $h = h_{L+1}$. In fact, applying Equation (4.11) to Equation (4.10) results in

$$\mathbf{E}[u] - \mathbf{E}\left[u_L^M\right] = \delta \left(c_N^L \, \mathbf{Var}\left[u_L^M\right] \right)^{-1/2} \mathbf{Var}\left[u_L^M\right]^{1/2} .$$

For this purpose we assume $\mathbf{Var}\left[u_\ell^M\right] \approx \mathbf{Var}\left[u^M\right]$ for $\ell = 1, \dots, L$ is enough approximation as only a rough estimate of the variance is needed. Taking the L^2 norm of the last equation and applying the variance approximation we get

$$c_N^L = \frac{|D|^2}{\delta^2} \left\| \mathbf{E}\left[u_L^M\right] - \mathbf{E}[u] \right\|_{L^2(D)}^{-2}$$

and with the extrapolated estimate $\dot{e}_{L+1} \approx \left\| \mathbf{E}[u] - \mathbf{E}\left[u_{L+1}^M\right] \right\|_{L^2(D)}$ we get an estimate for the constant c_N^{L+1} as $\dot{c}_N^{L+1} = \frac{|D|^2}{\delta^2} \dot{e}_{L+1}^{-2}$. The numbers of samples for level $L+1$ are now set according to Equation (4.11) as

$$N_{L+1}^i \coloneqq \dot{c}_N^{L+1} \, \mathbf{Var}\left[u_L^M(\nu_{L+1}^i) \right] \qquad \text{for } i = 1, \dots, |\mathcal{N}_{L+1}| . \tag{4.12}$$

Remark 4.2 It is imperative to ensure a minimum number of samples for each N_L^i in Equation (4.12) as on each level a sufficient approximation of the variance

$\mathbf{Var}\left[u_\ell^M\right]$ needs to be available. This is important as otherwise the algorithm can become unstable through severe undersampling in single points which itself will result in bad spatial error estimation in Section 4.2 and thus suboptimal mesh refinement with reduction of convergence rate or even loss of convergence.

A simple solution is to choose some $N_{\min}$ independent of all parameters and especially independent of the level ℓ. The practical application calls for some crude estimate of the variance which can be computed alongside the expected value and thus set the numbers of samples on level $L+1$ as

$$\dot{N}_{L+1}^i \coloneqq \min\left\{\dot{c}_N^{L+1}\,\mathbf{Var}_{\boldsymbol{M},\boldsymbol{N}}^{\mathrm{MC}}\left[u_L^M(\nu_{L+1}^i)\right], N_{\min}\right\} \quad \text{for } i = 1,\dots,|\mathcal{N}_{L+1}|\,. \tag{4.13}$$

Finally, it remains to choose the parameter Δt for each level. The influence on the error of this parameter is given in the last term of Equation (4.8). As we assume linear pointwise convergence with respect to Δt, we choose the relation $h \sim \Delta t$. In Algorithm 4.2 the overall algorithm is sketched. The computation of $\mathbf{E}_{\boldsymbol{M},\boldsymbol{N}}^{\mathrm{MS}}[u_\ell]$ is depicted in Algorithm 4.1. The experiments three and four in Chapter 5 evaluate the numerical performance of the method.

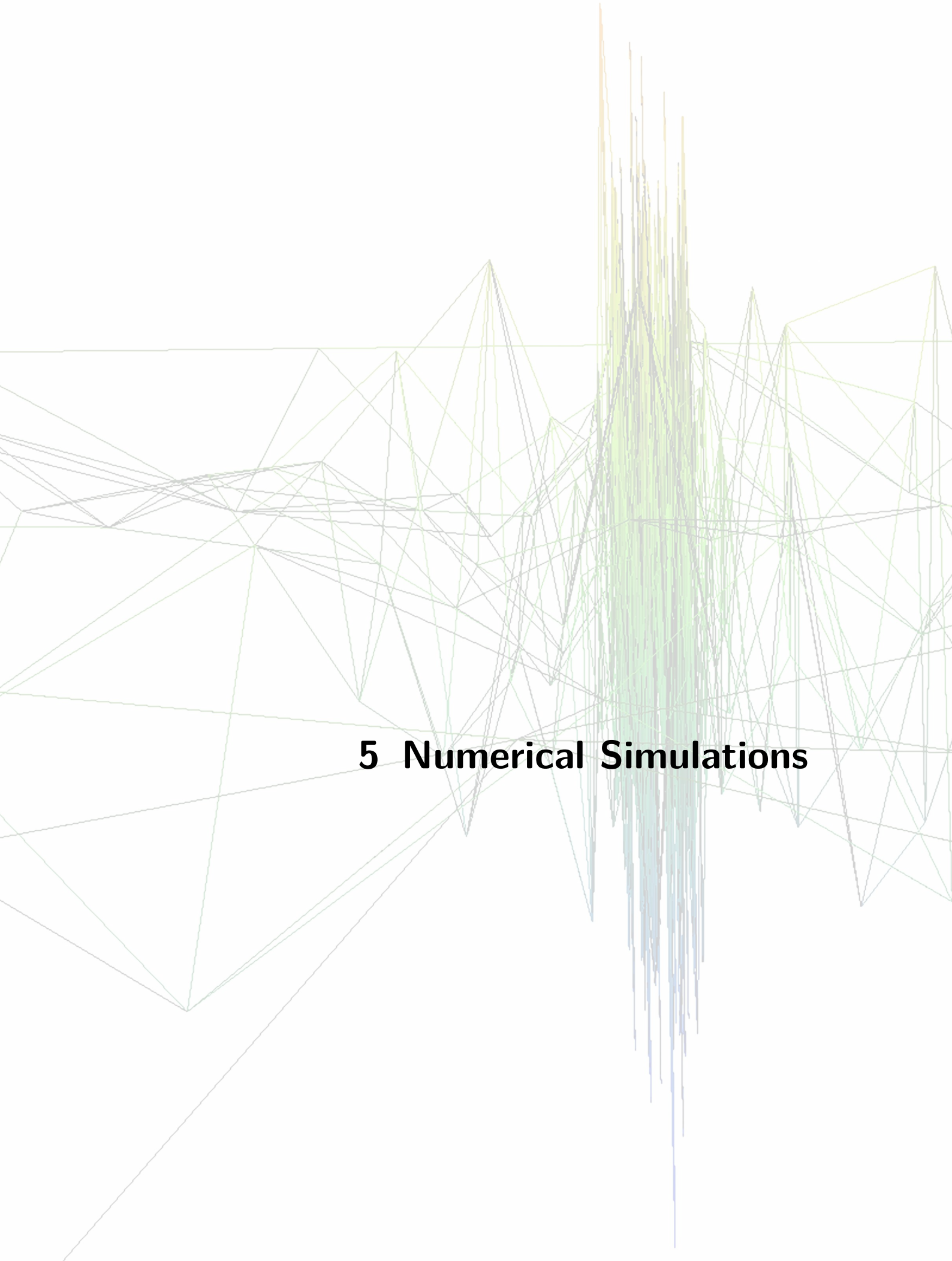
Instead of the variance adaptive local number of samples for each vertex one can choose a common number of samples based on the variance $\mathbf{Var}\left[u_L^M\right]$. This is closer to the methods presented in Chapter 2 as they also have to compute each sample on the whole domain, though the multilevel method uses coarser levels to mitigate this shortcoming. The experiments in Chapter 5 will compare the method presented here with one using uniform meshes and a common number of samples in different combinations.

Remark 4.3 In the numerical calculations one has to impose Equation (4.9) well enough, such that the algorithm becomes stable. This can be achieved by choosing $\delta < 1$. In the experiments in Chapter 5 we choose $\delta = 0.2$. This allows for errors in the error estimation and variance approximation which results in a stable algorithm that only imposes a slight impact on performance.

In : $\mathcal{T}_0$, $\boldsymbol{N}_{init}$, Δt_0
Out : solution $\mathbf{E}^{\mathrm{MS}}_{\boldsymbol{M},\boldsymbol{N}}[u_L]$

```
for ℓ = 1, ..., L do
    if ℓ ≥ 2 then
        compute N_ℓ according to Equation (4.13)
    else
        set N_ℓ to N_0
    set Δt according to h
    compute E^MS_{M,N_ℓ}[u_ℓ] and Var^MS_{M,N_ℓ}[u_ℓ] with Algorithm 4.1
    if ℓ = L then
        break
    compute η_ℓ
    refine T_ℓ with η_ℓ to get T_{ℓ+1}
return E^MS_{M,N}[u_L]
```

Algorithm 4.2: Adaptive Algorithm for the stochastic representation.



5 Numerical Simulations

5.1 Overview

In order to investigate the effect of the different approaches onto the quality of the solution, two problems are considered. The first one exhibits a strong singularity induced by some slit or crack in the domain. The second one, on the other hand, provides a huge bump in one corner whereas it is almost constant in the rest of its domain.

The two quantities of interest are chosen such that an integral over some part of the domain is computed which is some distance away from the singularity and the oscillations of the solution. This constitutes a typical situation where the adaptivity might fail if it is only driven by the problem data instead of taking the quantity of interest into account.

In order to understand the impact of this problem, first, the adaptive meshes are tested in deterministic situations with respect to the domain wide error in the L^2 and H^1 norms. Afterwards, the adaptivity will be goal-driven to assure convergence with respect to the error in the quantity of interest. Finally, the tests are transferred to the stochastic setting in order to assess if the gained advantages can be retained in that setting.

The implementation of the algorithms is realized in Python with the FEniCS toolbox [fen03, ABH+15, LMW+12]. This template based software package provides a highly flexible Python front end to a high performance C++ back-end employing just-in-time compilation. At the same time, it enables easy access to a variety of powerful and wide-spread numerical packages such as PETSc, SLEPc, Trilinos, TetGen, CGAL and UMFPACK.

The Unified Form Language (UFL) [ALO+14, Aln12] provides the basis for the formulation of the variational problem in Python which is then processed by the Unified Framework for Finite Element Assembly (UFC) [ALM+09, ALM12] from which the FEniCS Form Compiler (FFC) [KL06, LORW12, OW10] generates C++ template code. This is compiled on the fly together with an extensive Python interface which is loaded afterwards for immediate use.

In order to facilitate the full potential of multi-core processors the Joblib Python library [V+16] is used as an accessible interface for the Unix fork paradigm. This mechanism allows to share data between processes and gives excellent parallel performance with an easy to use interface. Monte Carlo methods scale very well with the number of threads or processes used as the samples can be computed individually. Hence, it is important to avoid recalculation and multiple copies of data in memory.

The UMFPACK solver [Dav04] is used for the solution of the resulting system of linear equations. It is part of the SuiteSparse package [D$^+$16] which contains a collection of sparse matrix algorithms from [Dav06] and other sources. Additional Python packages used include SciPy [H$^+$16], Matplotlib [Hun07] and others.

The first part of the implementation for the alternate approach is realized in C++ as a Python module computing the single point expectancies in parallel using OpenMP whereas the rest of the adaptive algorithm is implemented in Python with the FEniCS toolbox.

For all the experiments, reference solutions have been acquired on meshes which result from two additional uniform refinements of the best adaptive method's last mesh. For this computation 10^5 samples have been used in the Monte Carlo method to generate the mean function $\mathbf{E}[u^\star] \approx \mathbf{E}[u]$ and hence the reference value of the quantity of interest.

5.2 Monte Carlo and Multilevel Monte Carlo

Experiment 5.1 – Slit Domain

The first experiment incorporates a constant right-hand side $f \equiv 1$ on the unit square domain with a slit $D = [0,1]^2 \setminus [0.5,1] \times \{0.5\}$ together with homogeneous Dirichlet boundary data on the whole boundary Γ_D of D. The weight g from Equation (1.4) for the quantity of interest is defined with the parameters r_g and x_0 as

$$g = \begin{cases} C_g^{-1} r_g^{-2} \exp\left(\frac{-1}{1-\|x_0-x\|^2 r_g^{-2}}\right) & \text{if } \|x_0 - x\| \leq r_g, \\ 0 & \text{else.} \end{cases} \tag{5.1}$$

The weight C_g is defined such that g represents a mollifier, that is

$$C_g = r_g^{-2} \int_D \exp\left(\frac{-1}{1 - \|x_0 - x\|^2 r_g^{-2}}\right) \mathrm{d}x,$$

as g has compact support and $\lim_{\varepsilon \to 0} \varepsilon^{-2} g\left((x_0 - x)/\varepsilon\right) = \delta(x_0)$. The parameters are chosen as $r_g = 0.3$ and $x_0 = (0.3, 0.3)$. For the random field κ, first the smooth experiment random field from Equation (2.8) is used with the parameters $A = 0.6, \sigma_\alpha = 2, t = 5, \varepsilon_\kappa = 5 \cdot 10^{-6}$, and $c_\kappa = 10^{-3}$. Later, the same input data is exposed to a log-normal random field generated with the Cholesky decomposition

as described in Section 2.3 with the vertices of some base mesh $\mathcal{T}_\kappa$ as input. The covariance function used for the generation is defined as

$$\mathsf{C}(x,y) \coloneqq \sigma_\kappa^2 \exp\left(-\lambda_\kappa^{-2} \|x-y\|\right).$$

The normal random field κ_{norm} generated in this way does not abide to the requirements in Problem (1.1). A field which does so and also exhibits behavior closer to practical applications, like high oscillations and fast changing amplitudes, is given for any function $\bar{\kappa} > 0$ in D and some constant $c_\kappa > 0$ by

$$\kappa \coloneqq \bar{\kappa} + c_\kappa \exp\left(\kappa_{\mathrm{norm}}\right).$$

For the experiments, the parameters are chosen as $\bar{\kappa} = 10^{-3}$, $\sigma_\kappa = 1$, $\lambda_\kappa = 0.3$, and $c_\kappa = 10^{-4}$ in order to represent physically plausible values. Visualizations of the mean solution, its variance, the weight of the quantity of interest g, and a realization of the random field are given in Figure 5.2 for the smooth random field and in Figure 5.3 for the rough log-normal random field.

To get a better understanding of the adaptive algorithm's behavior, it is a good idea to evaluate deterministic settings first. Two settings are chosen for this purpose, the constant $\kappa \equiv 1$ and some fixed realization of the rough random field. The resulting meshes from the different adaptive approaches for the first case are presented in Figure 5.4 together with a uniform mesh of similar complexity. The meshes for the rough realization are depicted together with the resulting κ in Figure 5.5.

In these graphics, the main difference between the approaches becomes visibly apparent. While the primal strategy naturally only refines near the singularity, the two goal-driven approaches also care for the quantity of interest. It is apparent, that η_{MS} refines distinctively with respect to the primal or the dual problem resulting in a clear representation of the quantity and the singularity in the refined meshes. On the other hand, the η_{OP} indicator constructs a combined estimator covering also the error contributions connecting the quantity and the singularity for the price of a lower resolution at both. Furthermore, the adaptivity is less pronounced as more refinement occurs further away from these two areas. The rougher κ in the second case introduces additional local refinement resulting in a more diverse mesh. Still, the main features are common for both cases.

The convergence of the error with respect to the quantity of interest, the performance of the bounds as well as the performance of the estimators for both cases is shown in Figure 5.6. There, the relative error $\mathsf{e}_{\mathrm{rel}} = |Q(u) - Q(u_h)| \,/\, |Q(u)|$ is shown together with the relative efficiency of the upper and lower bounds $\eta_{\mathrm{rel}}^{\pm} = |\eta^+ - \eta^-| \,/\, |Q(u)|$. Finally, the error estimators are scaled accordingly as $\eta_{\mathrm{rel}}^{Q} = \eta_\circ / \,|Q(u)|$ where $\eta_\circ$ is

the error estimator used for the refinement. In the case of the uniform meshes η_{OP} is computed and plotted but not used for refinement.

In this setting, all adaptive algorithms exhibit the same much better convergence rate compared to uniform meshes for both cases. This is due to the strong singularity at $x = [0.5, 0.5]$. The two goal-driven adaptive methods improve the convergence slightly as they also refine at the quantity of interest's support. As expected, the convergence for the rough κ is less smooth and due to the nature of the goal-driven adaptivity even non-monotonic. However, after the oscillations are resolved, both cases show the same errors at the same mesh complexity. The goal-adaptive indicator η_{OP} performs the best.

The guaranteed bounds perform suitably well, and are sharpest for the uniform meshes. Note that the convergence of the bounds is much smoother especially for the goal-driven algorithms and that they converge monotonically. This is especially welcome for the development of heuristics in Section 3.4. The efficiency of the bounds, that is $|\eta^+ - \eta^-| \,/\, |Q(u) - Q(u_h)|$, is somewhere around 10 - 70. The estimators indicate the convergence behavior rather well, while they omit the preasymptotic non-monotonic behavior of the true error. This also helps to derive stable and reliable heuristics.

For the better understanding of the error estimators as well as the global convergence behavior, Figure 5.7 presents the error in the L^2 and the H^1 norm alongside the primal error indicator η_u. In both cases, the representation of the errors in the H^1 norm through the estimators is very good. The overestimation constant differs for varying permeability fields κ as the error estimators independence of κ only holds for the weighted energy norm $\|\!|u - u_h|\!\|_\omega$. The uniform meshes show a slower rate of convergence, while all the adaptive refinement methods show approximately the same rate. The primal error indicator η_u and the goal-adaptive indicator η_{MS} show the best performance, closely followed by η_{OP}.

The sharpness of the guaranteed bounds depends greatly on the choice of $\alpha(\omega)$. In Remark 3.14 $\alpha_{\mathrm{opt}}(\omega)$ is suggested as a suitable approximation to the optimal value. The influence of $\alpha(\omega)$ on the bounds is presented in Figure 5.1 for both deterministic examples. Alongside the width of the bounds, the approximation $\alpha_{\mathrm{opt}}(\omega)$ and the value of the bounds with this choice is plotted. It is clear that $\alpha_{\mathrm{opt}}(\omega)$ performs almost perfectly for this setting.

In the stochastic setting, the refinement is driven by mean error estimators and thus, the resulting meshes are less noisy or even smooth as shown for the smooth random field in Figure 5.8 with $N_\eta = 100$ samples for each step in the mesh generation algorithm and in Figure 5.9 with $N_\eta = 10$ samples as well as for the rough random

field in Figure 5.10 with $N_\eta = 100$ samples for each step in the mesh generation algorithm and in Figure 5.11 with $N_\eta = 10$ samples. Additionally to the meshes generated with the primal and goal-driven adaptivity, a mesh resulting from the dual problem's error indicator is presented.

Due to the smoother nature of the mean meshes, the observations from the deterministic cases are even more apparent. The separation of the two areas of interest is clearly visible for the indicator η_{MS}, whereas the singularity and the quantity of interest are clearly connected with a high resolution area in the mesh for η_{OP}. The number of samples used for the refinement only has marginal impact on the general mesh structure. The variant with less samples is a bit more noisy but as the singularity is very strong, this is hardly noticeable. Furthermore, due to the nature of the algorithm, the samples from the different levels add up, such that very few samples are needed to get good results.

To evaluate the quality of the bounds, we take a closer look at two values. The first one describes the width of the frustum between the upper and lower bound relative to the true value. This gives us information about the efficiency of the bounds and hence we define the relative efficiency index as

$$E_{\mathrm{rel}} \coloneqq (\eta^{\oplus} - \eta^{\ominus}) \, |Q(u)|^{-1}.$$

The second quantity will give us insights into the quality of the solution. As the results of the computations are realizations of random variables themselves, it is possible that the error $\mathsf{e} \coloneqq \left| \mathbf{E}[Q(u)] - \mathbf{E}_N^{\mathrm{M}}[Q(u_h)] \right|$ can become arbitrarily small for every mesh and any number of samples though asymptotically the error will converge to the spatial error term in Equation (2.1) and Equation (2.5) for $N \to \infty$ and $\boldsymbol{N} \to \infty$ respectively. In order to mitigate this problem, we will add the confidence intervals derived in Section 3.2 to the error and hence define the reliable relative error as

$$\tilde{\mathsf{e}}_{\mathrm{rel}} \coloneqq (\mathsf{e} + I_c) \, |Q(u)|^{-1}$$

with the confidence intervals $I_c \coloneqq \mathbf{Var}[Q(u_h)]^{1/2} \, N^{-1/2} \Phi^{-1}(\mathsf{p}^*)$ for the Monte Carlo method and $I_c \coloneqq \sum_{\ell=0}^{L} \mathbf{Var}[Y_\ell]^{1/2} \, N_\ell^{-1/2} \Phi^{-1}(\mathsf{p}_\ell^*)$ for the multilevel Monte Carlo method. This will assure that we take the variance of the estimator into account and thus, will give us reliable information about the expected error of the method. Hence, we can compare the different methods reliably.

All multilevel Monte Carlo computations have been performed with $L = 3$ since $L = 2$ did not show enough of an improvement over Monte Carlo methods and $L > 3$ had too big of an initial overhead in order to catch up to Monte Carlo within

reasonable time on the given computer setup and with the used implementation. The effect of adding additional levels is also limited without further effort as shown in [TSGU13]. More research is needed in this area to confirm and extend their findings.

The convergence with respect to the number of degrees of freedom is plotted in Figure 5.12 with $N_\eta = 100$ and Figure 5.13 with $N_\eta = 10$ for the smooth random field. Here, we can see that the adaptive algorithms perform just as well for the Monte Carlo as well as the multilevel Monte Carlo method compared to the deterministic test earlier. Just as in the earlier test the uniform meshes show a slower convergence rate compared to the adaptive meshes. All in all, the results are noisier due to the stochastic nature and the convergence is less smooth most likely due to differences in the performance of the heuristics. The quality of the bounds is also similar to the deterministic settings and the same conclusions hold true. No difference with respect to the number of samples N_η used for the mesh sequence generation can be seen.

However, in contrast to the deterministic case, these graphics do not allow to evaluate the quality of the solutions with respect to the computational effort. Once a balance of the two error contributions in Equation (2.1) and Equation (2.5) is achieved, the number of samples is the driving factor for the computational cost in error reduction. This is due to the lower convergence rate of $\frac{1}{2}$ for sampling methods. So, in order to compare Monte Carlo and multilevel Monte Carlo, we have to examine the cost it takes to achieve a prescribed error. Figure 5.14 for $N_\eta = 100$ and Figure 5.15 for $N_\eta = 10$ allow to do just that for some implementation independent standardized computational cost, which assumes $\mathcal{C}_\ell \sim \texttt{ndof}$, and the actually consumed processor time for our implementation. Note that the standardized computational cost plot does not account for the mesh generation while the processor time includes this effort.

As the overhead for the multilevel Monte Carlo method is bigger compared to the standard Monte Carlo method, it gets a better start on the coarse meshes. Once this advantage is exhausted, the multilevel Monte Carlo method is more efficient. In this experiment, the multilevel Monte Carlo method is approximately two orders of magnitude cheaper to achieve the same error in the solution. The adaptive meshes show a similar effect of two to three orders of magnitude for both sampling methods leading to an overall improvement of four orders of magnitude with respect to the cost for adaptive multilevel Monte Carlo over uniform Monte Carlo. Once all methods are in the asymptotic phase, their convergence rates are the same $N^{1/2}$, as expected. At the beginning, the more expensive mesh generation shows but is insignificant once the sampling error dominates the computational cost. In fact, there is no reason to invest a reasonable amount into the generation of the mesh sequence as it gives slightly higher quality meshes at a very little expense.

A slightly different picture is drawn, when the rough field is considered. In that case, Figure 5.16 for $N_\eta = 100$ and Figure 5.17 for $N_\eta = 10$ once again show similar performance as in the deterministic case. But as before, this does not allow for any conclusions about the cost. Figure 5.18 for $N_\eta = 100$ and Figure 5.19 for $N_\eta = 10$ give this information. The bounds show similar performance as in the smooth case. Multilevel Monte Carlo reduces the cost by approximately one order of magnitude and the adaptive meshes alleviate it by another order of magnitude leading to an overall two orders of magnitude cost advantage of adaptive multilevel Monte Carlo over uniform Monte Carlo. In effect, the rough random field constitutes a much harder problem but still the two presented methods result in a considerable improvement whereas the bounds perform just as well.

As is the case with the smooth field, the choice of the number of samples during the construction of the adaptive mesh sequence N_η shows almost no influence on the error reduction but only a slight performance impact on the coarse levels with a larger N_η. As this becomes insignificant on finer meshes, there is no reason to choose a suitably large N_η.

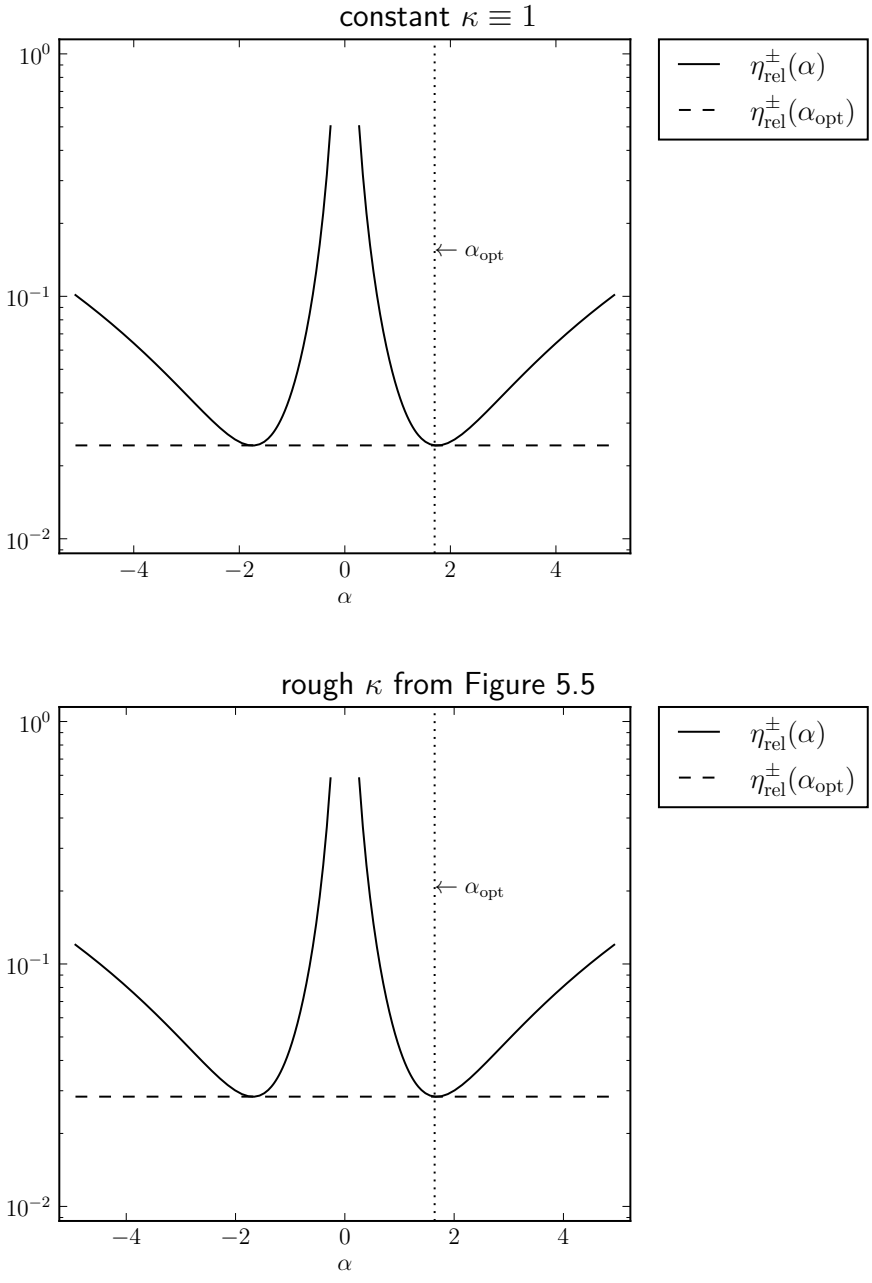


Figure 5.1: Experiment 5.1 Performance of the bounds with respect to the choice of α compared to α_{opt} for the two **deterministic** κ on adaptive meshes using η_{MS} with approximately 1000 degrees of freedom.

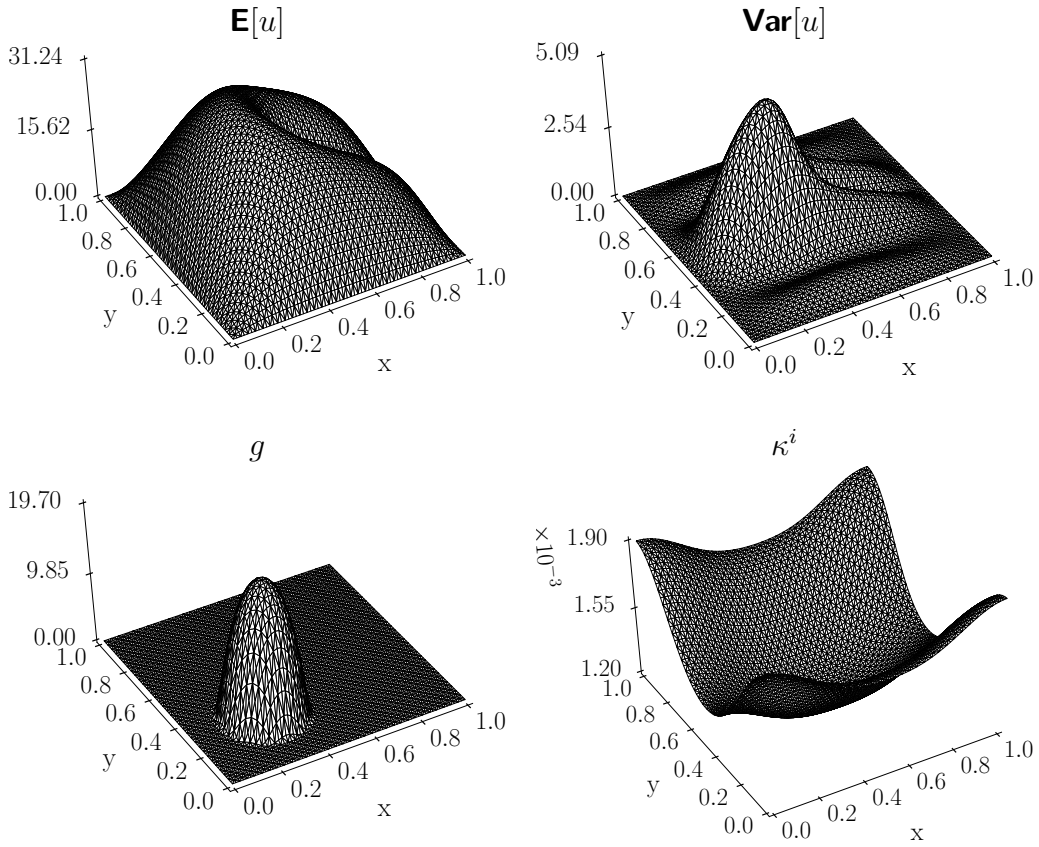


Figure 5.2: Experiment 5.1 The mean solution, the variance, the weight for the quantity of interest and a sample realization of the smooth field κ.

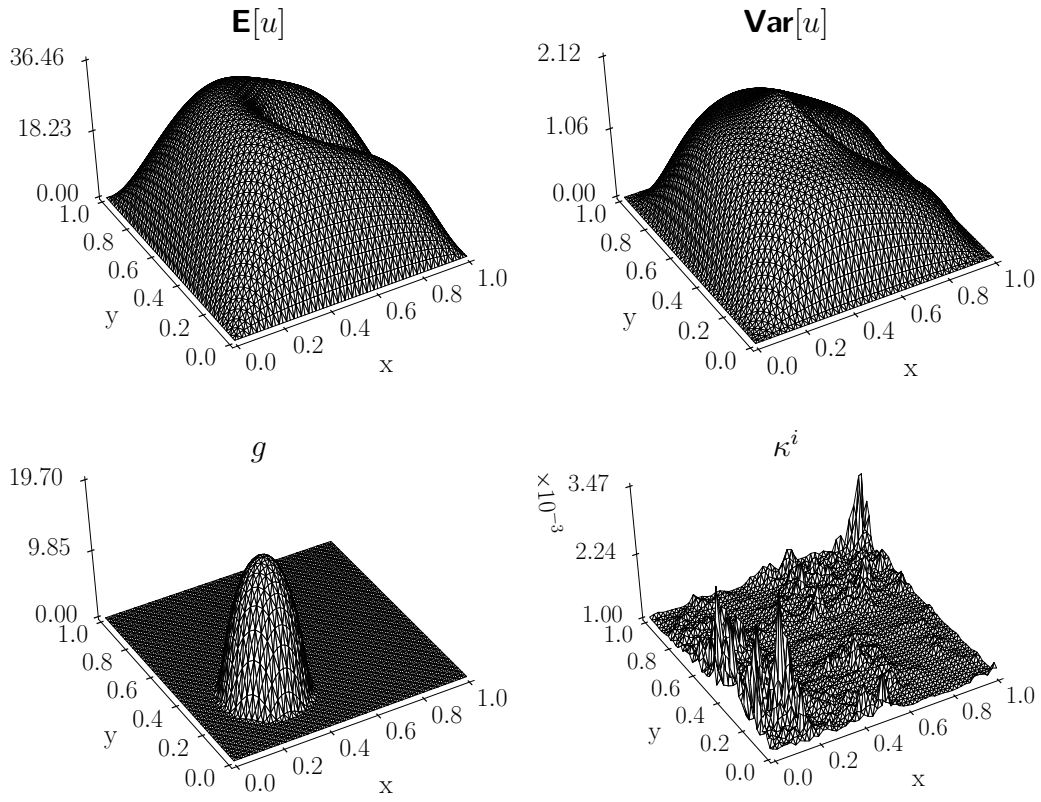


Figure 5.3: Experiment 5.1 The mean solution, the variance, the weight for the quantity of interest and a sample realization of the rough field κ.

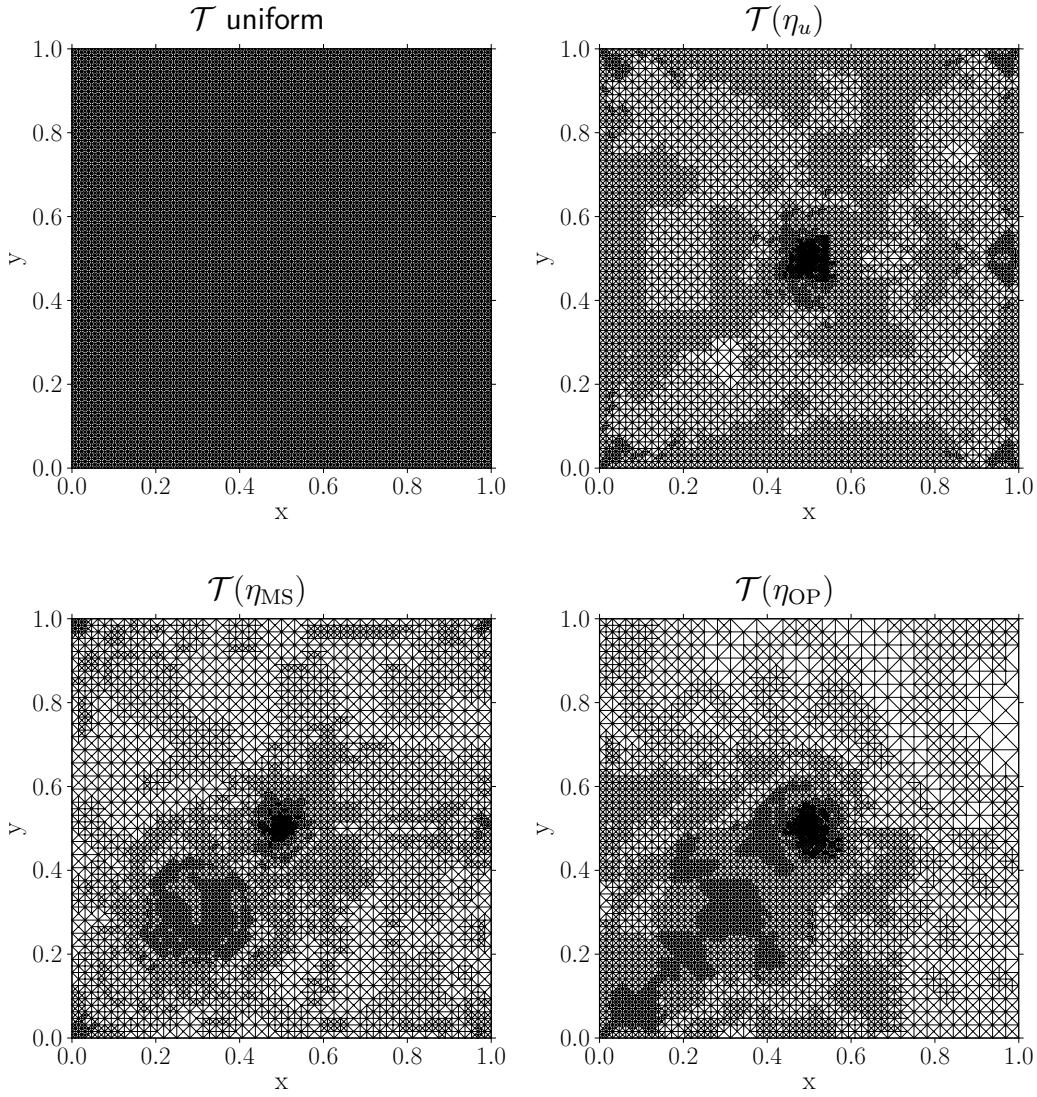


Figure 5.4: Experiment 5.1 The uniform mesh and meshes generated with the refinement indicators η_u, η_{MS}, and η_{OP} for the **constant deterministic** $\kappa \equiv 1$ with approximately 4200 degrees of freedom which result in the first graph in Figure 5.6.

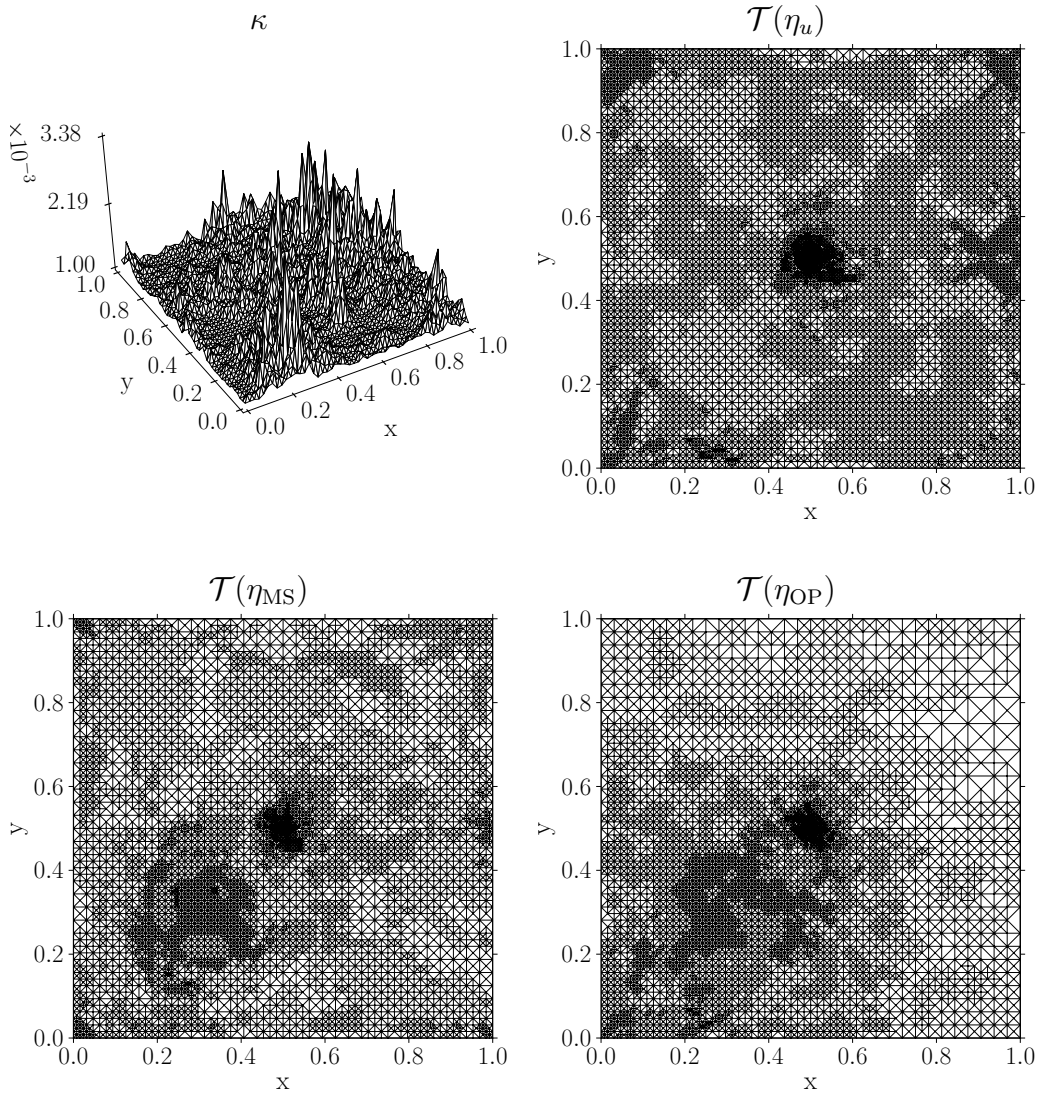


Figure 5.5: Experiment 5.1 The **rough deterministic** κ and the resulting meshes from the refinement indicators η_u, η_{MS}, and η_{OP} with approximately 4200 degrees of freedom which result in the second graph in Figure 5.6.

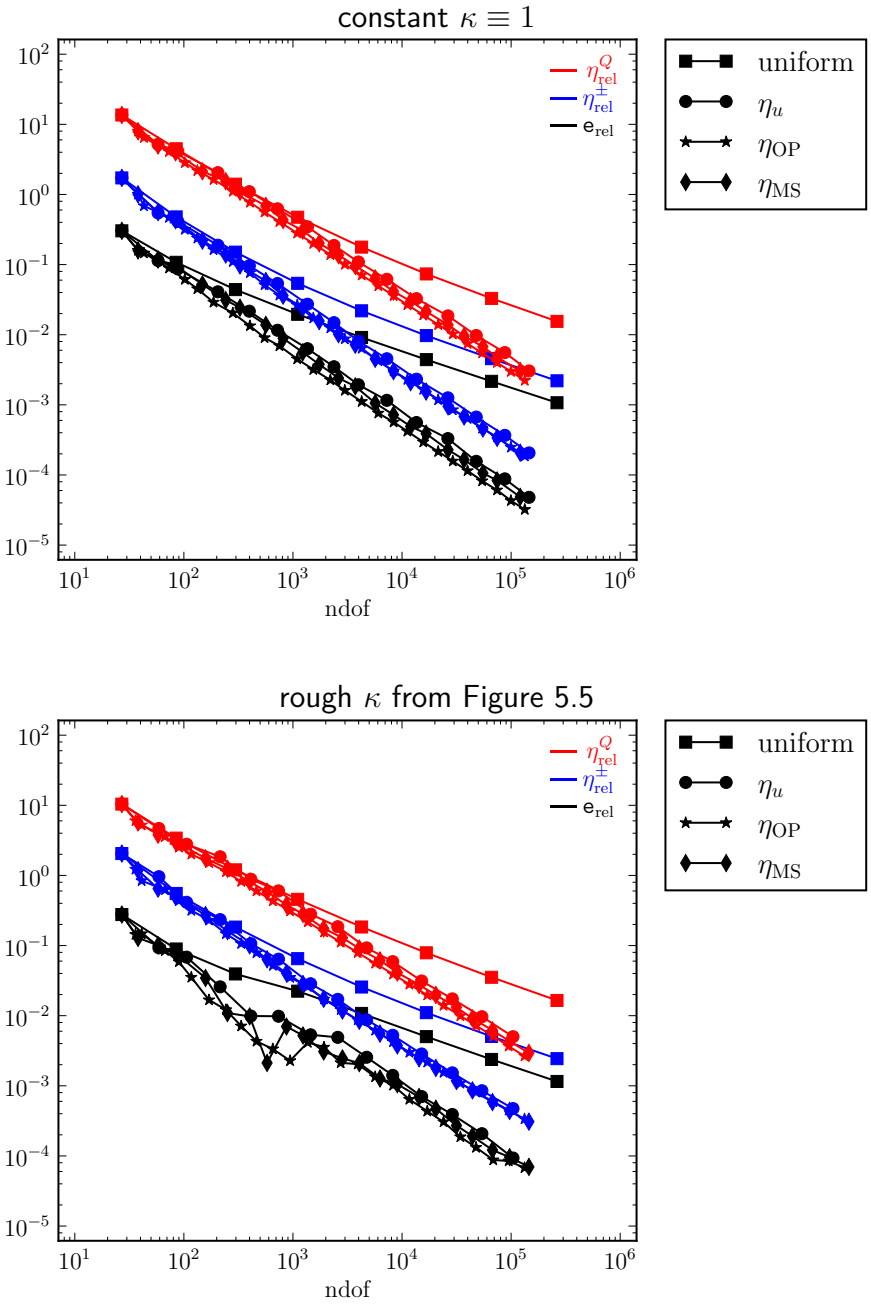


Figure 5.6: Experiment 5.1 Convergence of the error, the bounds, and the estimator with the refinement strategies uniform, η_u, η_{OP}, and η_{MS} for two **deterministic** κ with respect to the number of degrees of freedom. The short notations are $\eta^Q_{\mathrm{rel}} = \eta_\circ / |Q(u)|$ with the refinement indicator $\eta_\circ$, $\eta^{\pm}_{\mathrm{rel}} = |\eta^+ - \eta^-| \,/\, |Q(u)|$, and $\mathsf{e}_{\mathrm{rel}} = |Q(u) - Q(u_h)| \,/\, |Q(u)|$.

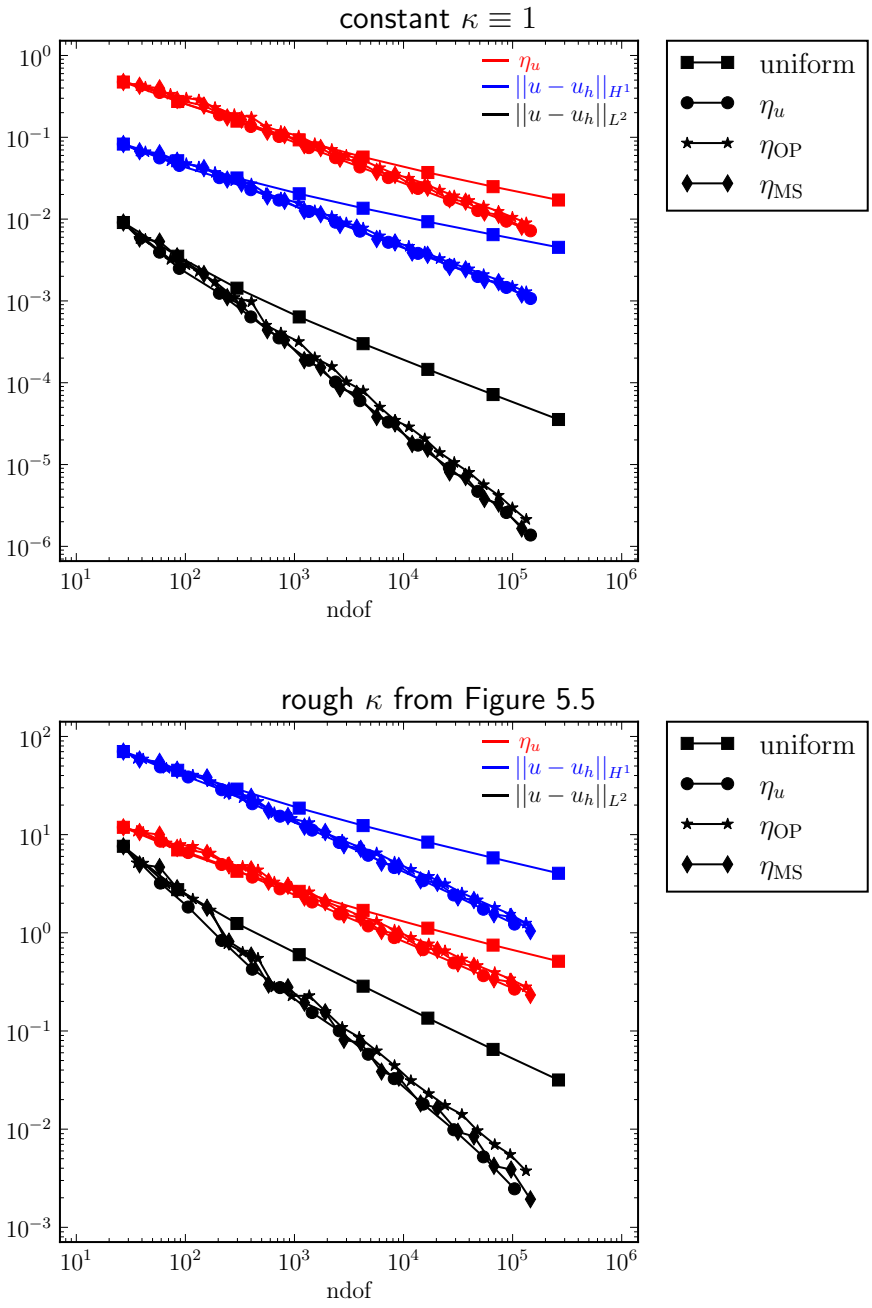


Figure 5.7: Experiment 5.1 Convergence of the global error in the H^1 norm, the L^2 norm, and the estimator with the refinement strategies uniform, η_u, η_{OP}, and η_{MS} for two **deterministic** κ with respect to the number of degrees of freedom.

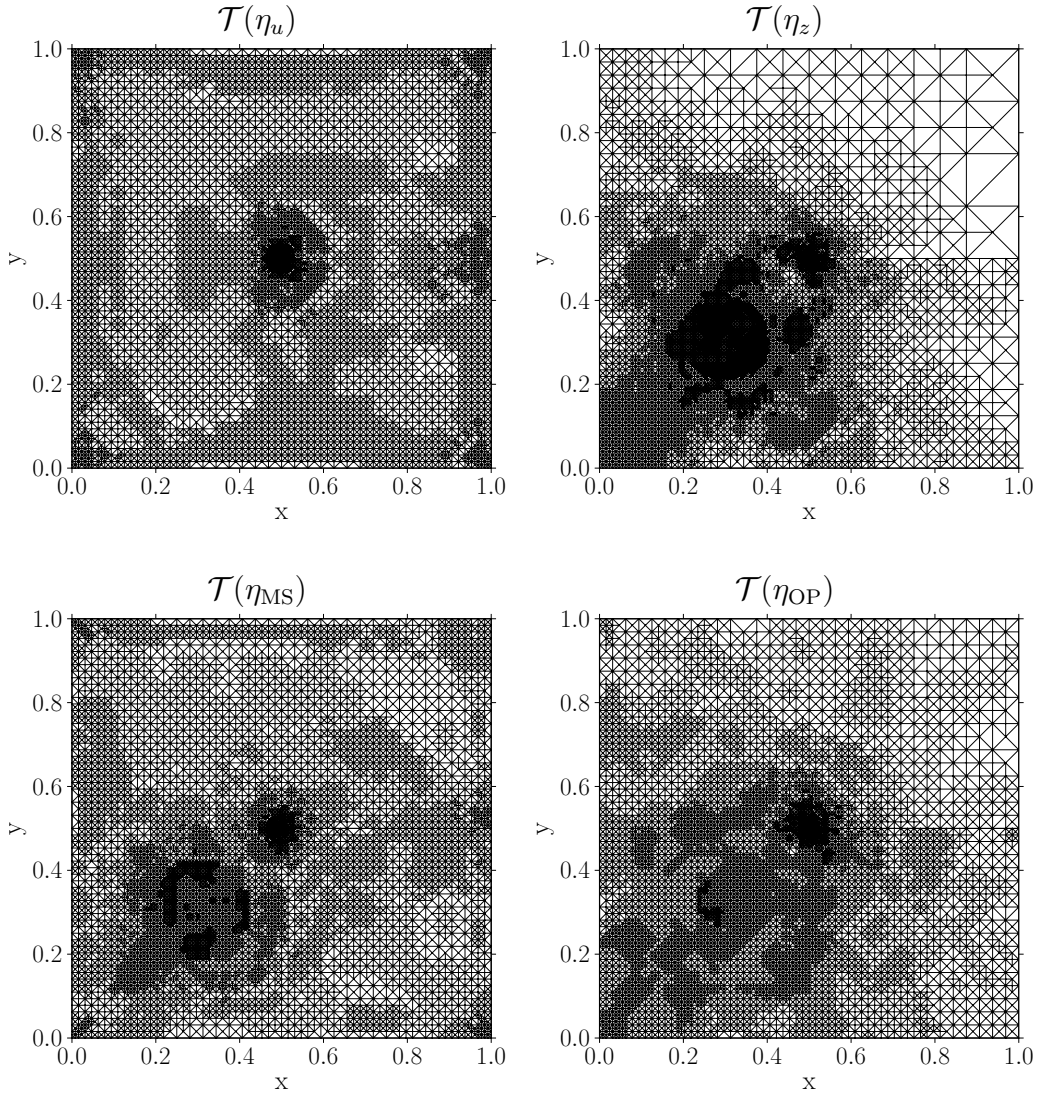


Figure 5.8: Experiment 5.1 The meshes in the stochastic setting generated by different error estimators with approximately 8000 degrees of freedom for the **smooth random field** with **100 samples** for the adaptive mesh generation.

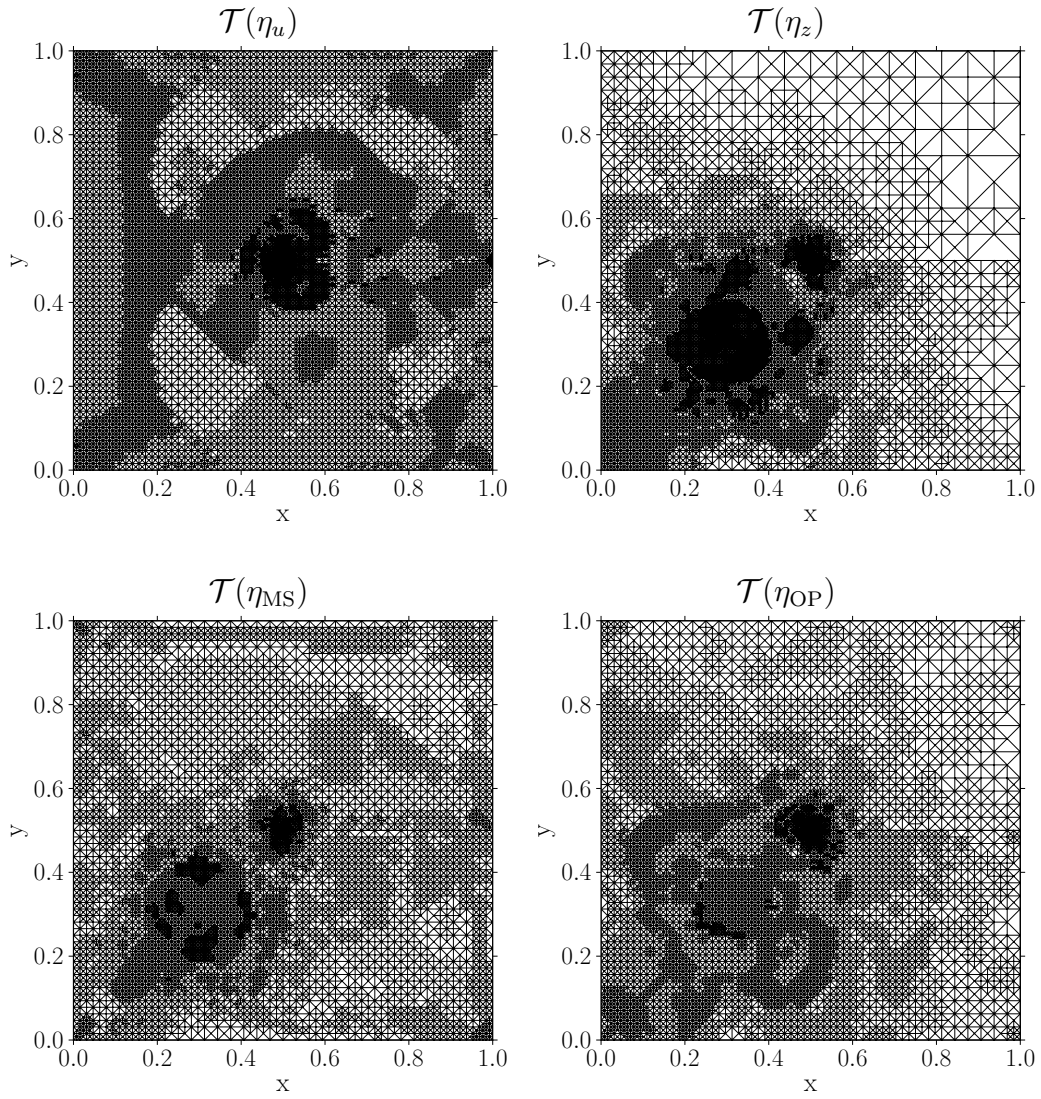


Figure 5.9: Experiment 5.1 The meshes in the stochastic setting generated by different error estimators with approximately 8000 degrees of freedom for the **smooth random field** with **10 samples** for the adaptive mesh generation.

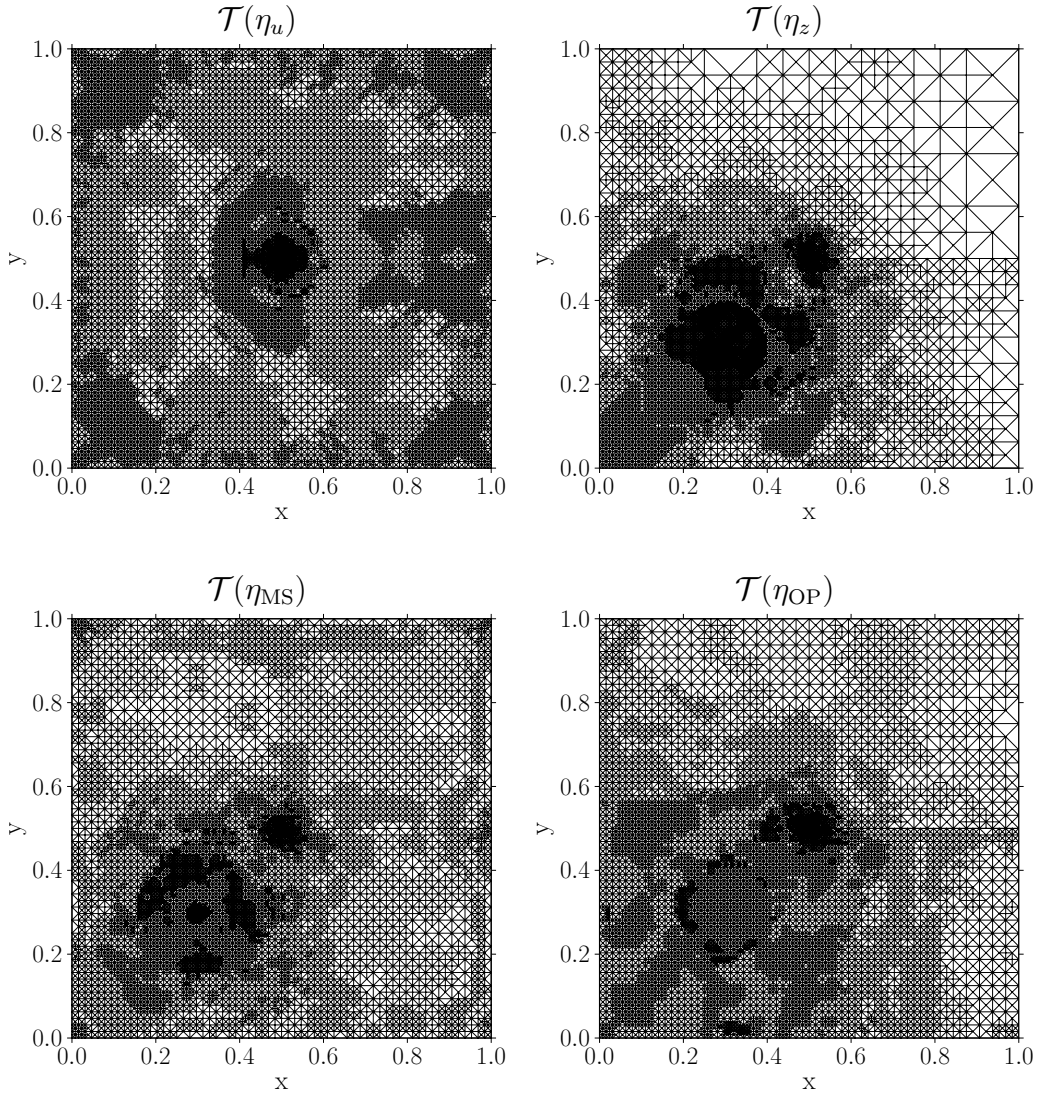


Figure 5.10: Experiment 5.1 The meshes in the stochastic setting generated by different error estimators with approximately 8000 degrees of freedom for the **rough random field** with **100 samples** for the adaptive mesh generation.

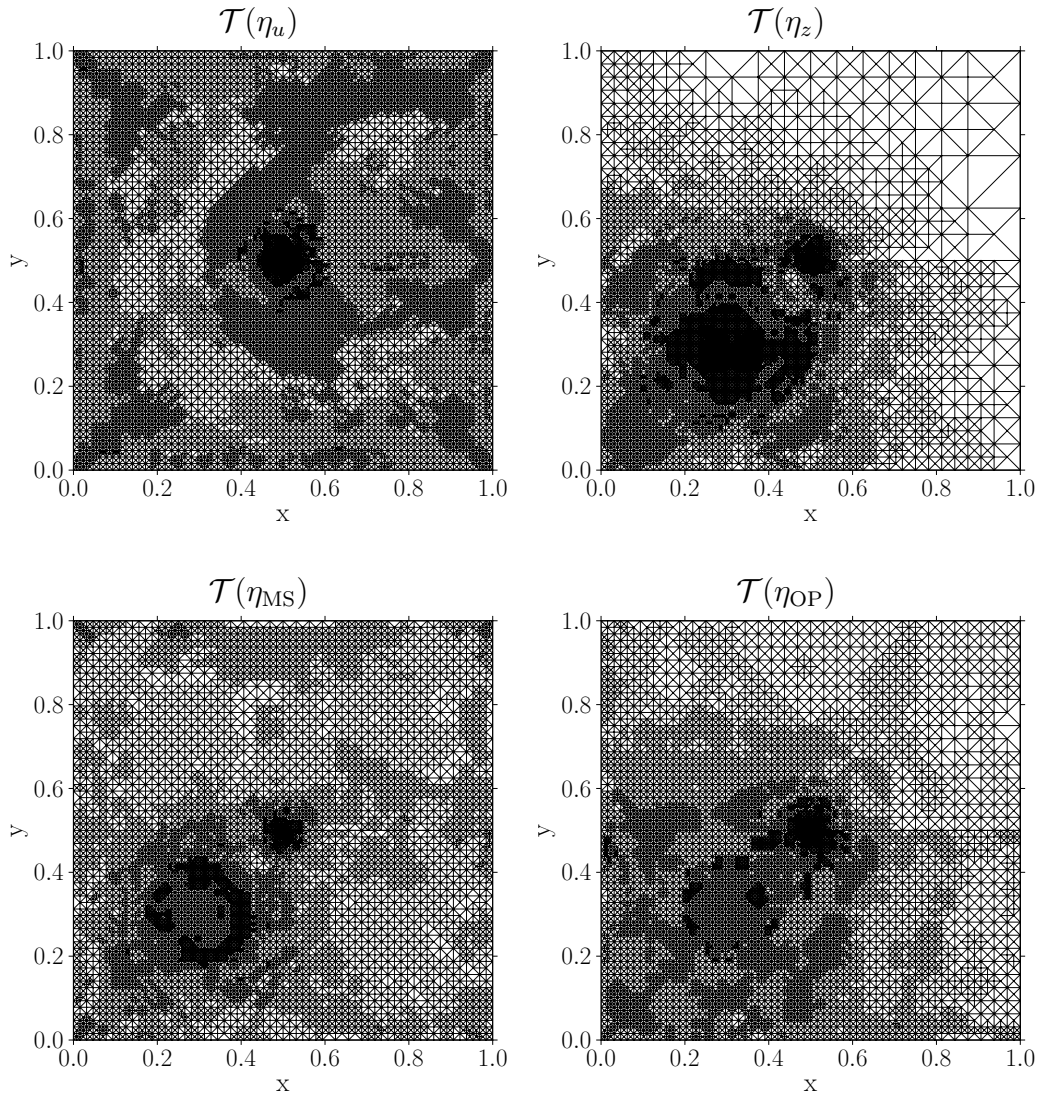


Figure 5.11: Experiment 5.1 The meshes in the stochastic setting generated by different error estimators with approximately 8000 degrees of freedom for the **rough random field** with **10 samples** for the adaptive mesh generation.

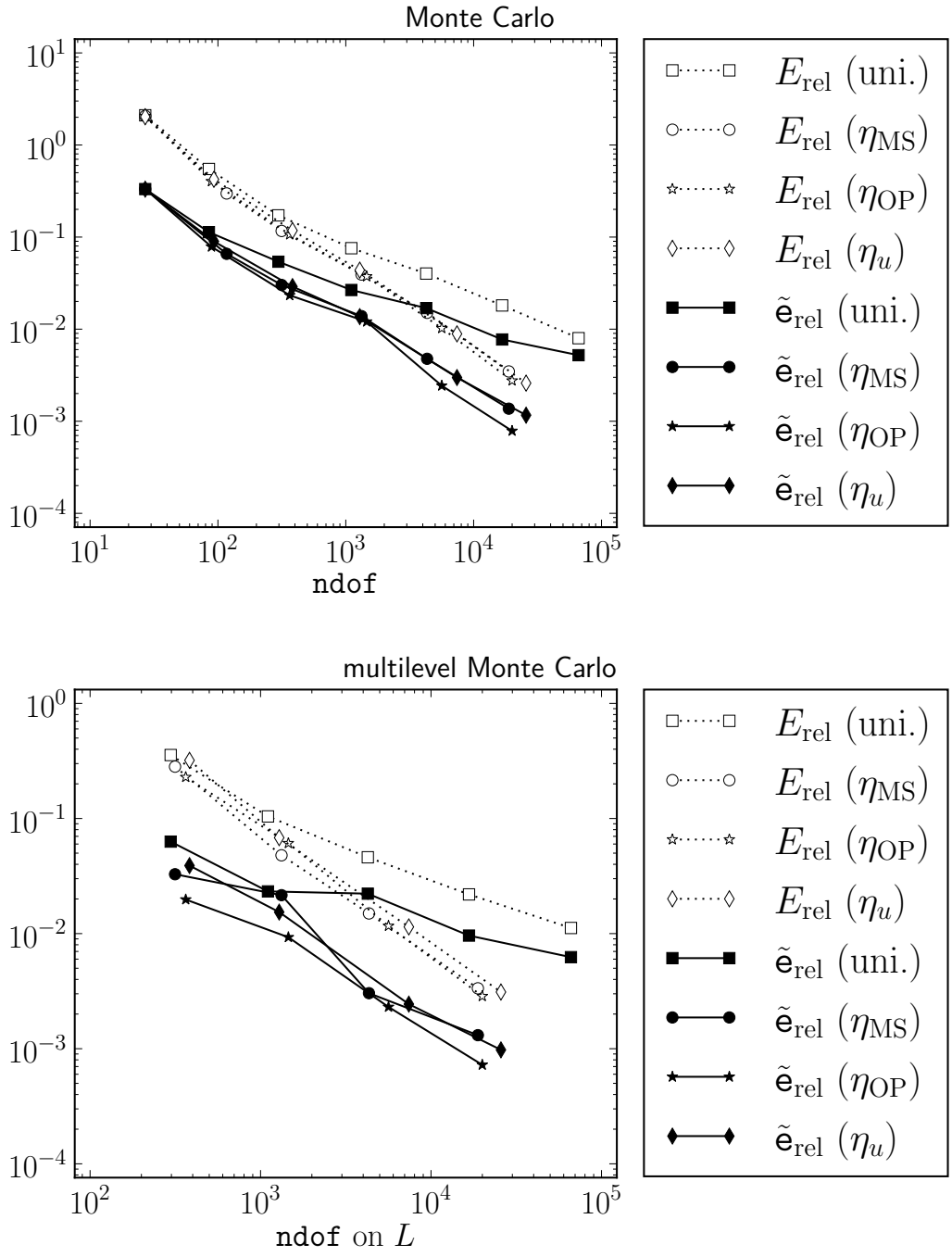


Figure 5.12: Experiment 5.1 Convergence with respect to the number of degrees of freedom for the Monte Carlo and multilevel Monte Carlo method for the **smooth random field** with **100 samples** for the adaptive mesh generation.

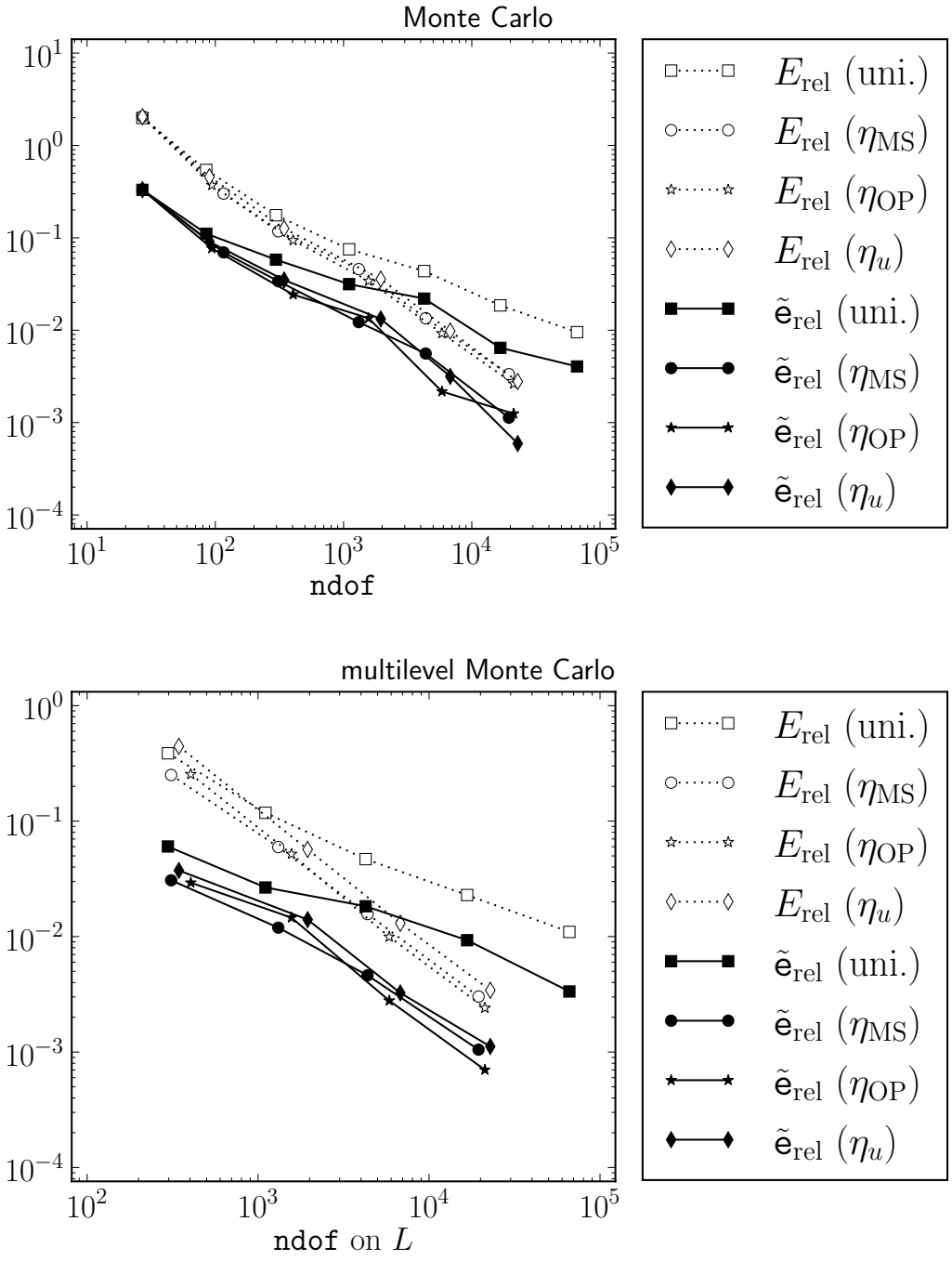


Figure 5.13: Experiment 5.1 Convergence with respect to the number of degrees of freedom for the Monte Carlo and multilevel Monte Carlo method for the **smooth random field** with **10 samples** for the adaptive mesh generation.

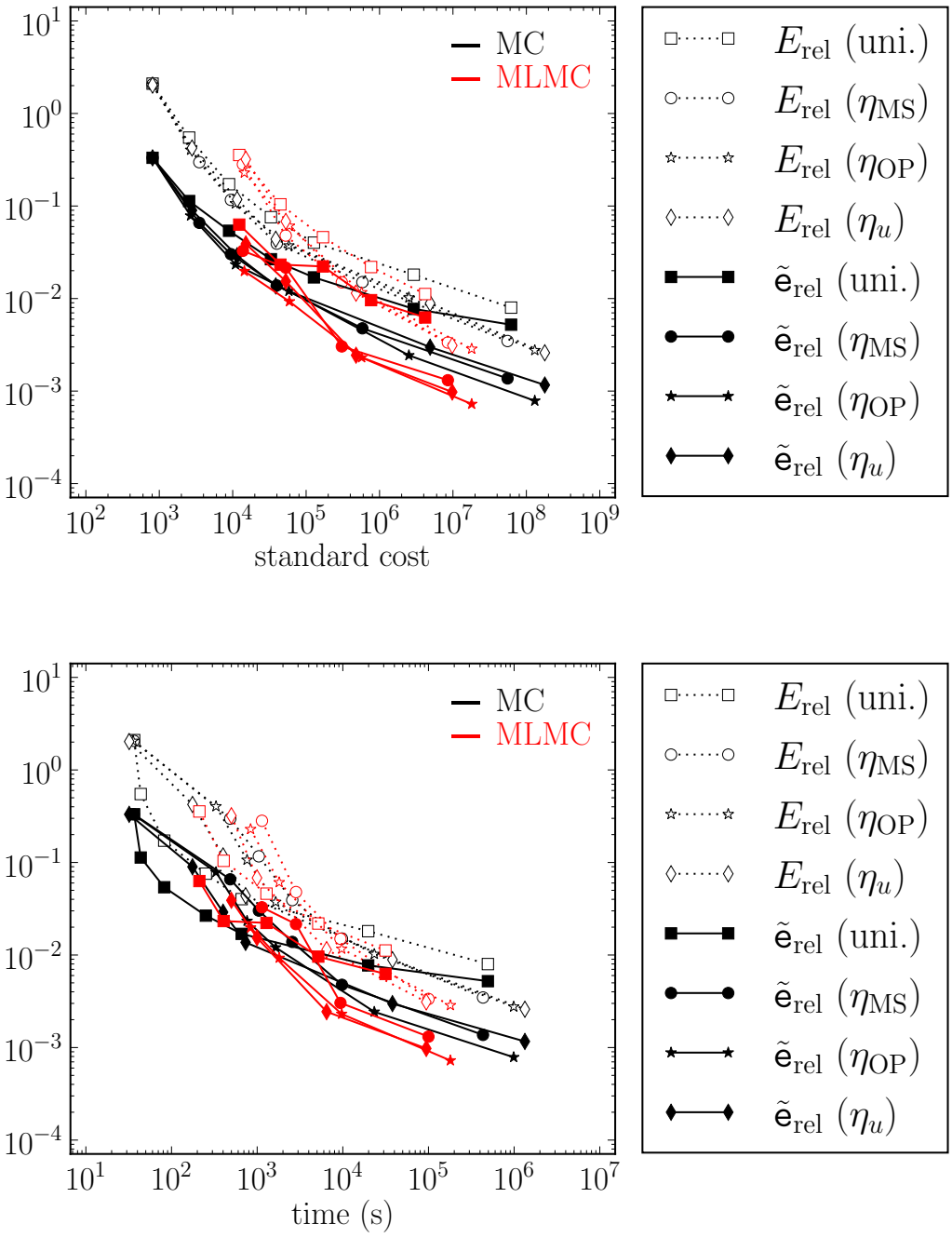


Figure 5.14: Experiment 5.1 Convergence with respect to standardized computational effort (top) and process time (bottom) for the **smooth random field** with **100 samples** for the adaptive mesh generation.

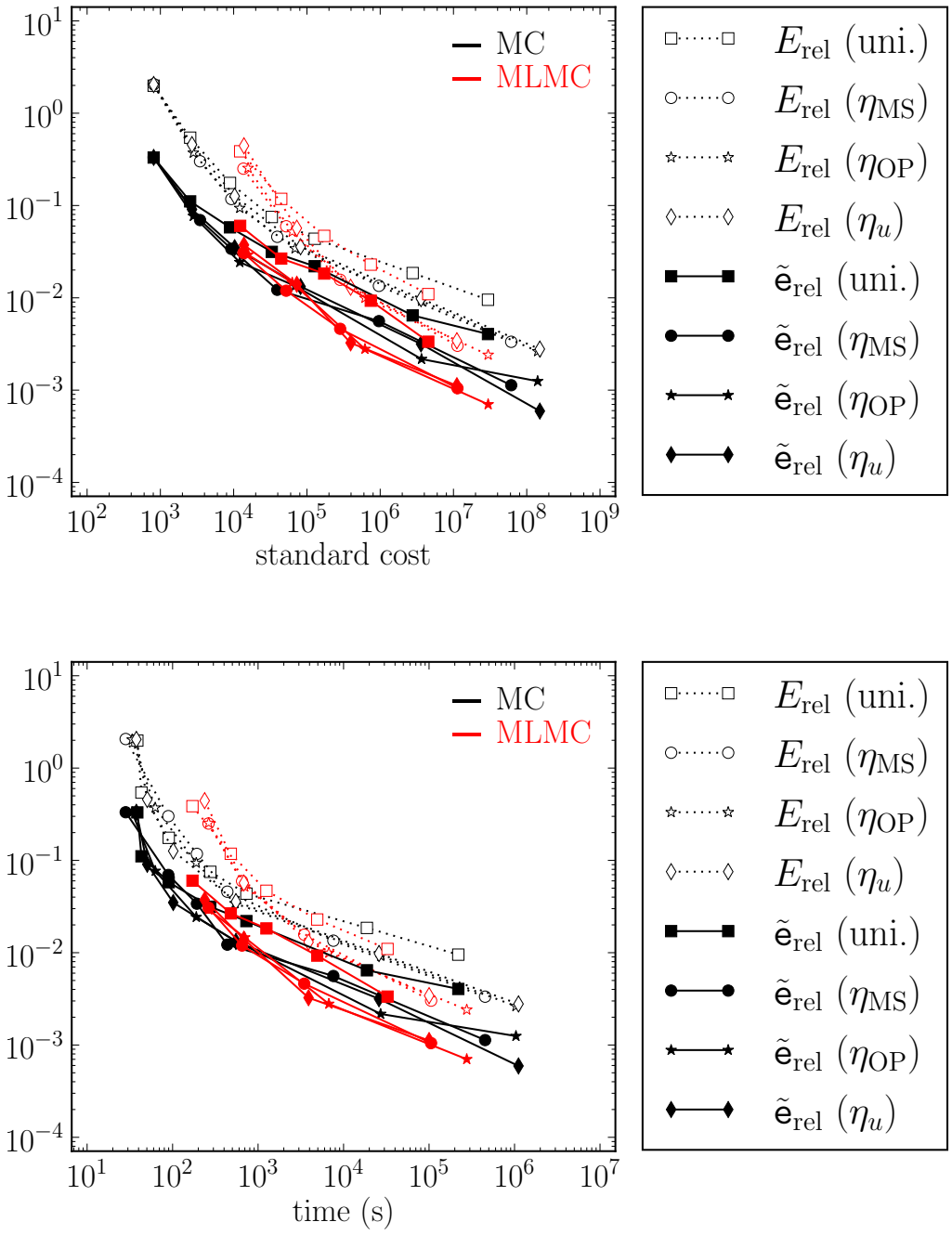


Figure 5.15: Experiment 5.1 Convergence with respect to standardized computational effort (top) and process time (bottom) for the **smooth random field** with **10 samples** for the adaptive mesh generation.

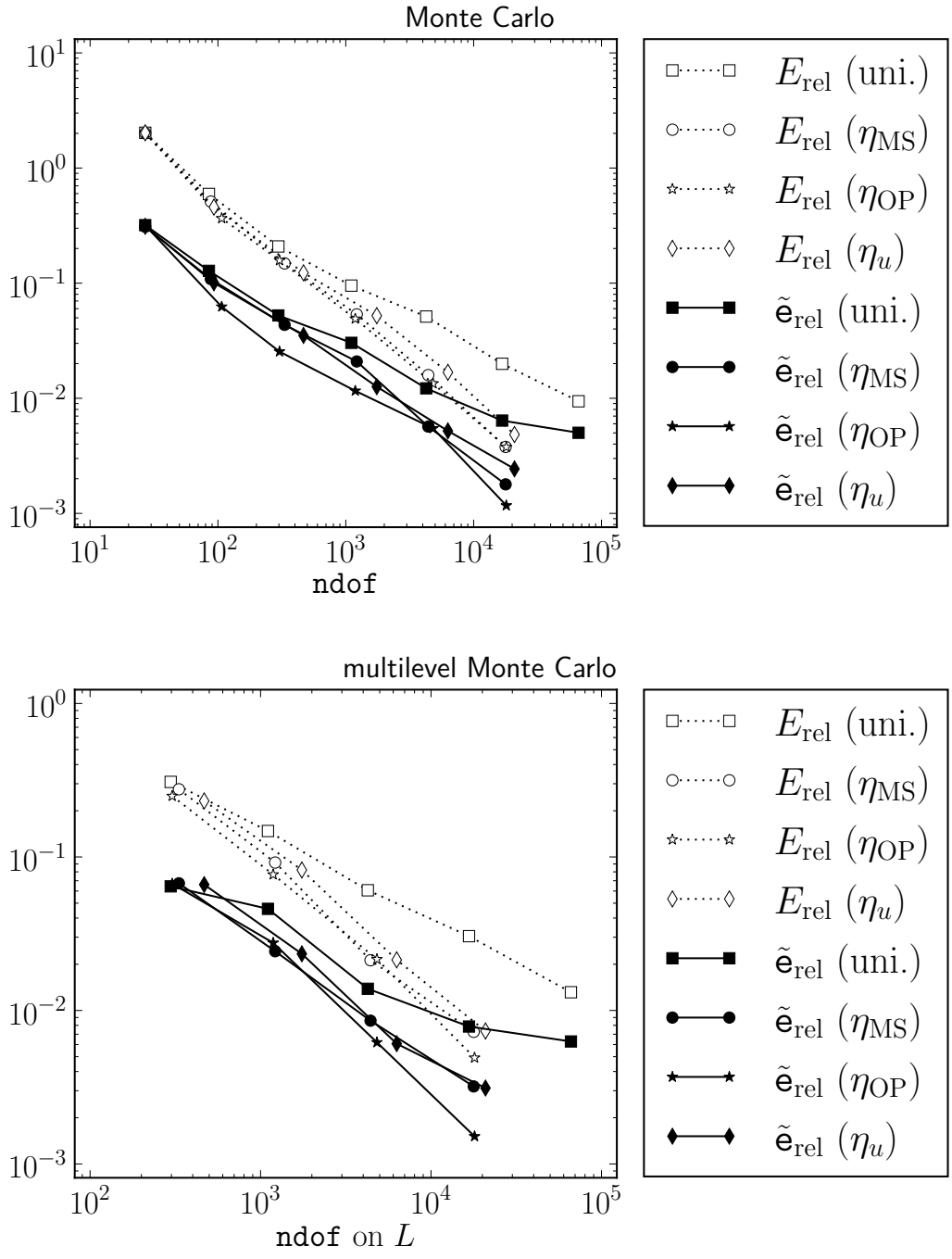


Figure 5.16: Experiment 5.1 Convergence with respect to the number of degrees of freedom for the Monte Carlo and multilevel Monte Carlo method for the **rough random field** with **100 samples** for the adaptive mesh generation.

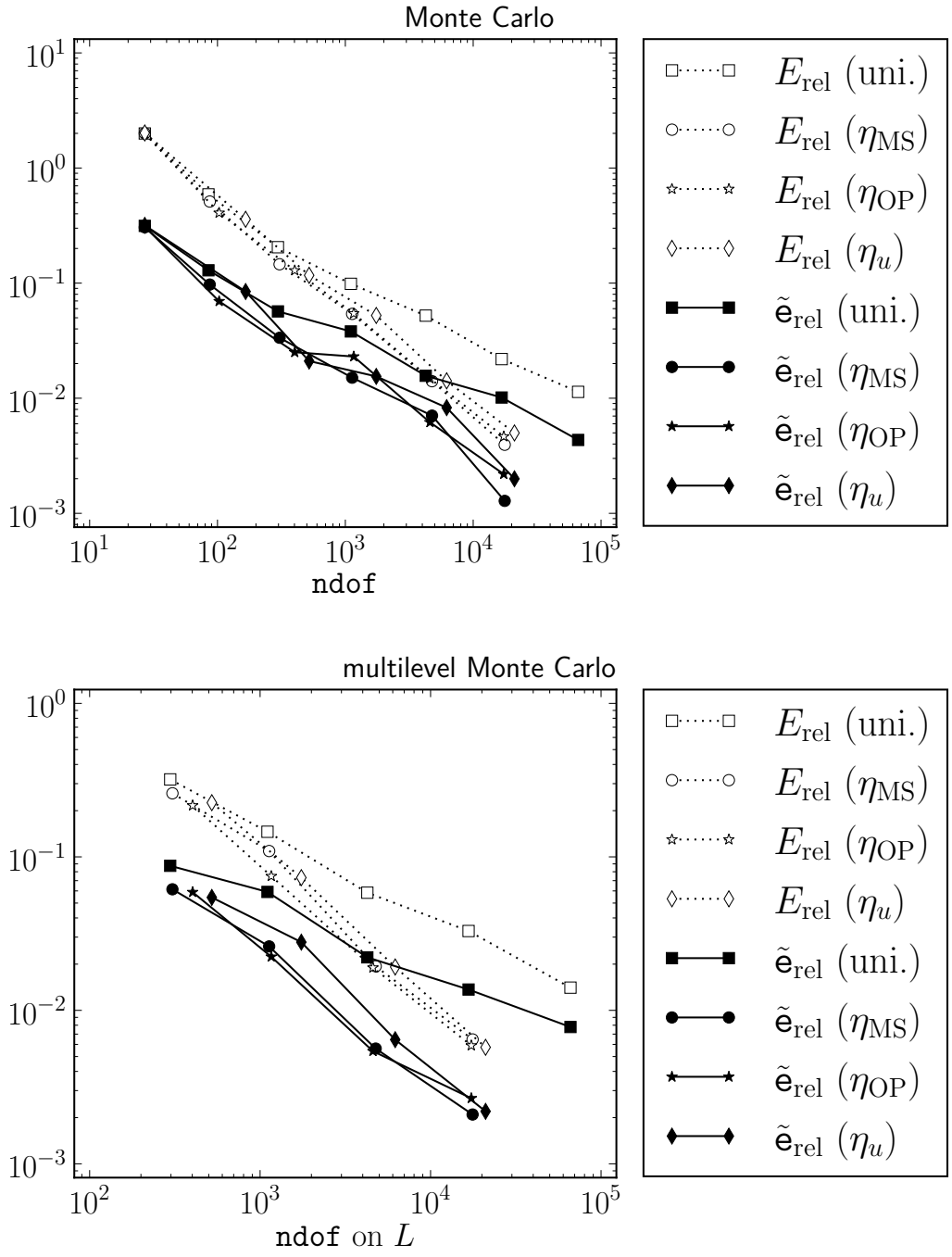


Figure 5.17: Experiment 5.1 Convergence with respect to the number of degrees of freedom for the Monte Carlo and multilevel Monte Carlo method for the **rough random field** with **10 samples** for the adaptive mesh generation.

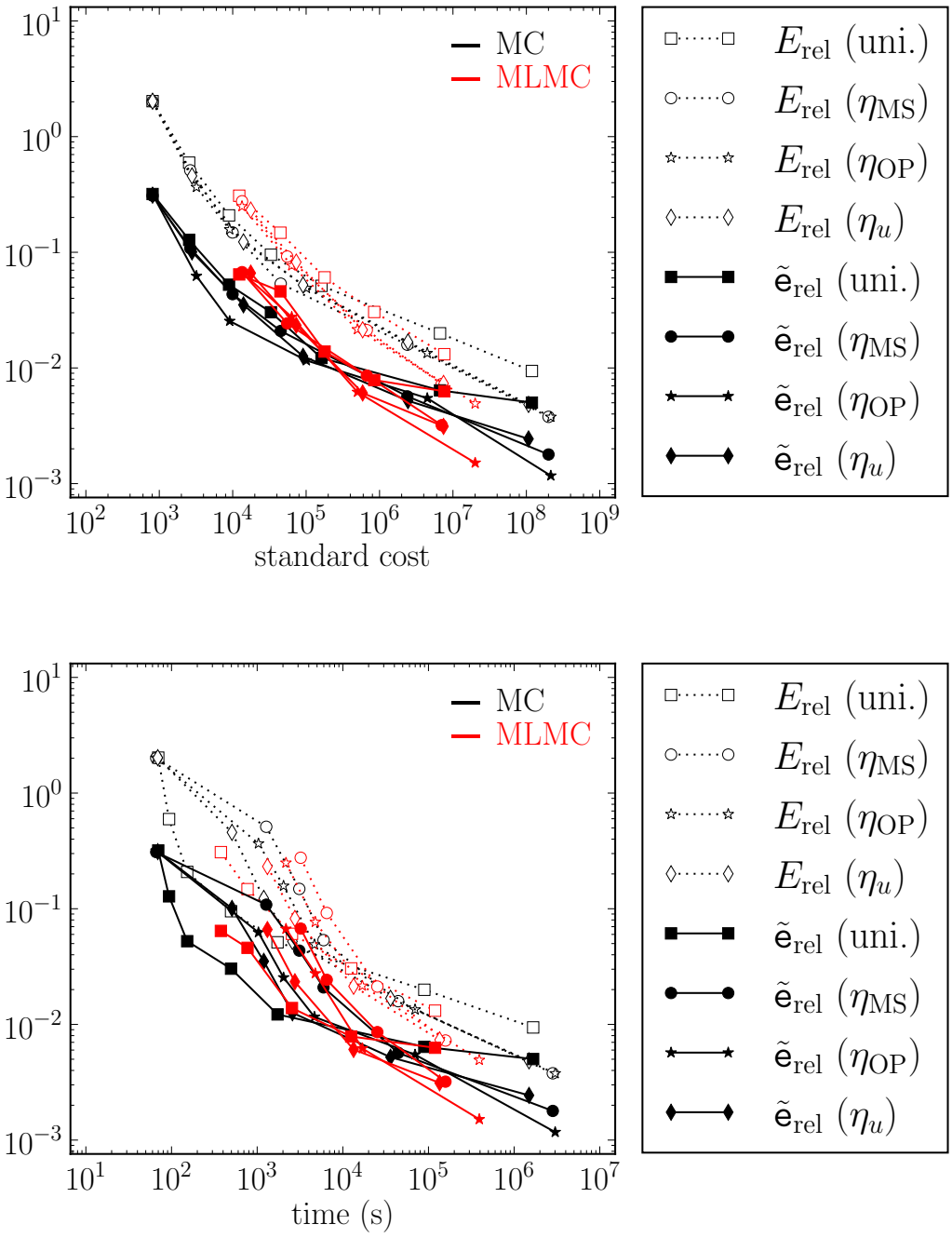


Figure 5.18: Experiment 5.1 Convergence with respect to standardized computational effort (top) and process time (bottom) for the **rough random field** with **100 samples** for the adaptive mesh generation.

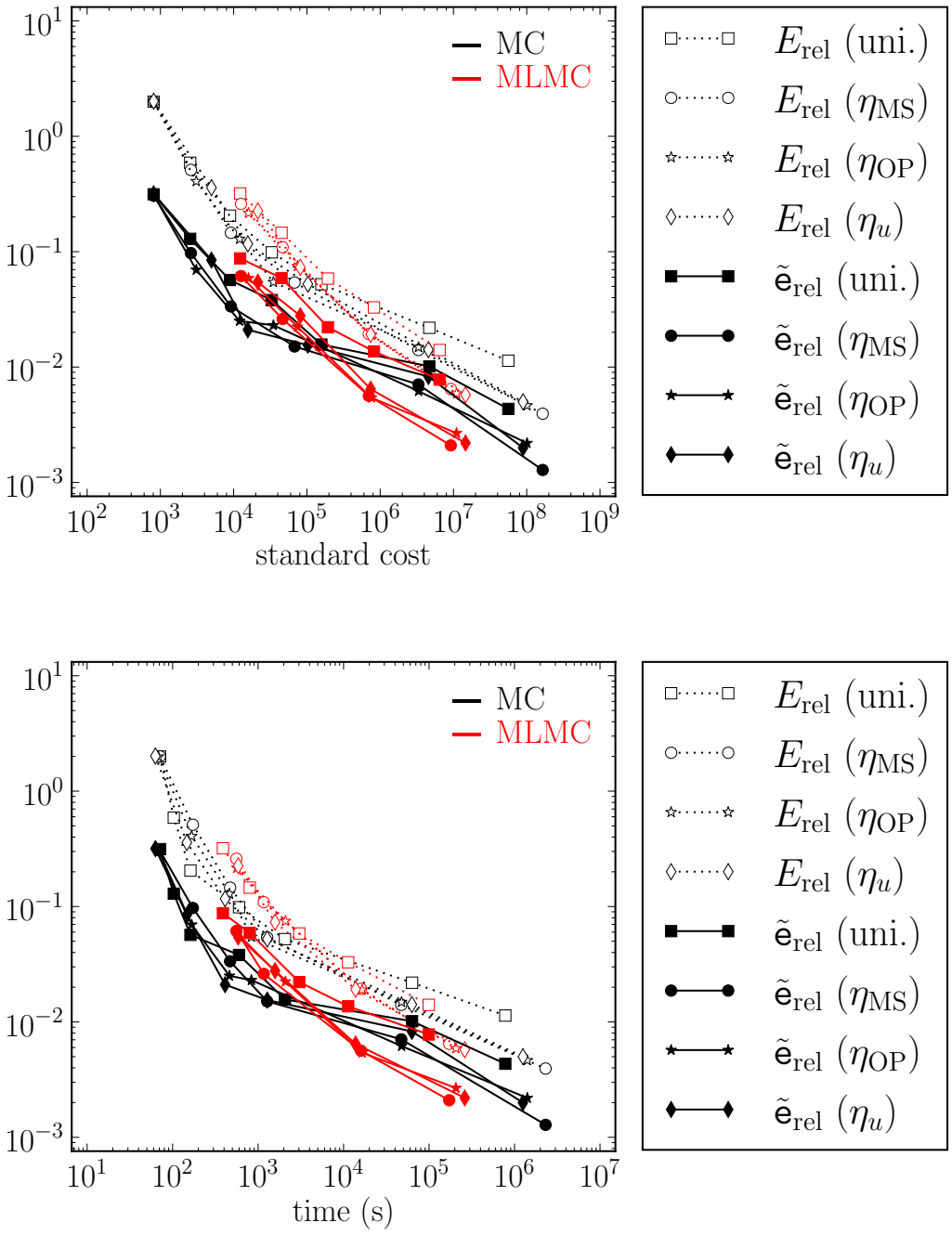


Figure 5.19: Experiment 5.1 Convergence with respect to standardized computational effort (top) and process time (bottom) for the **rough random field** with **10 samples** for the adaptive mesh generation.

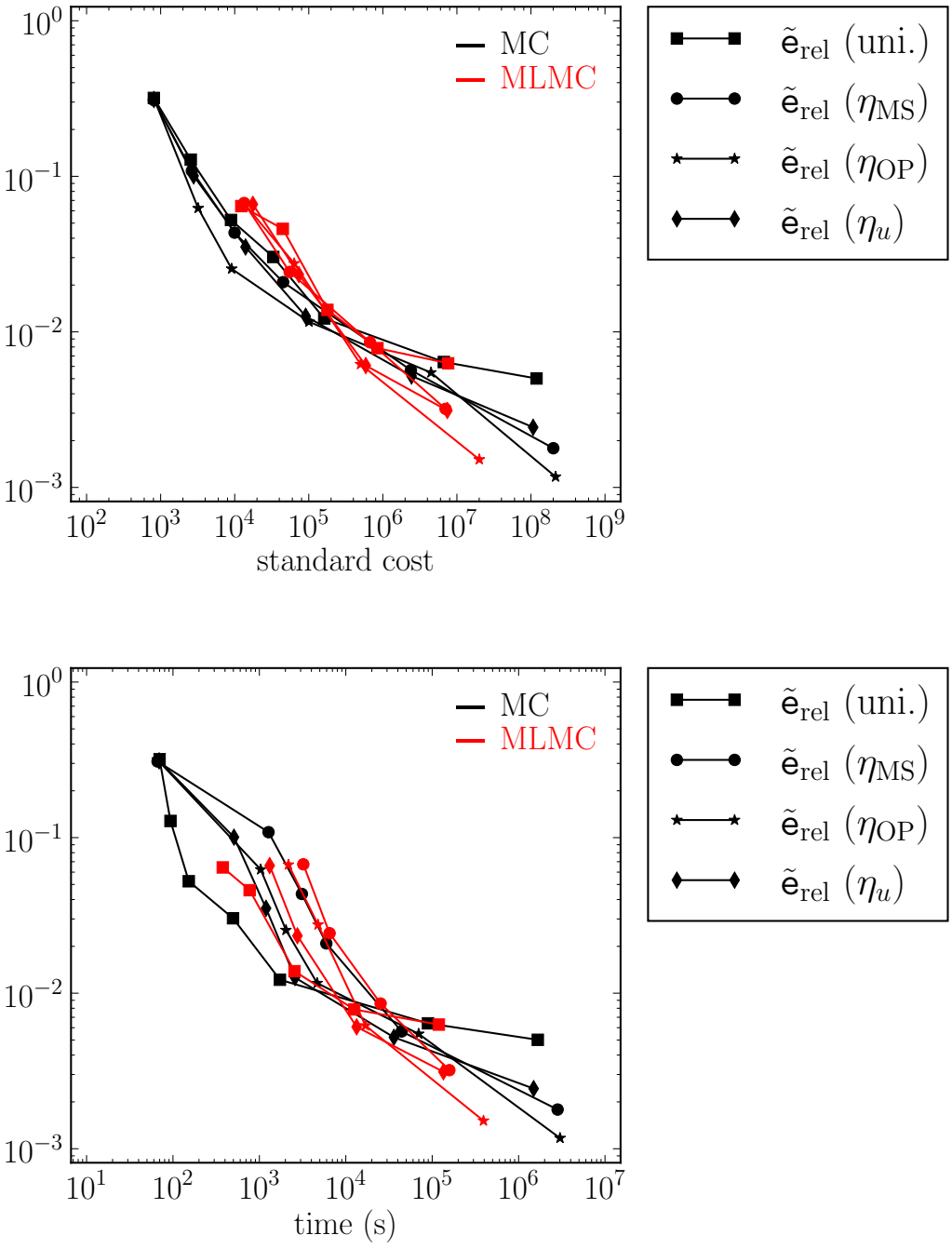


Figure 5.20: Experiment 5.1 Convergence with respect to standardized computational effort (top) and process time (bottom) for the **rough random field** with **100 samples** for the adaptive mesh generation.

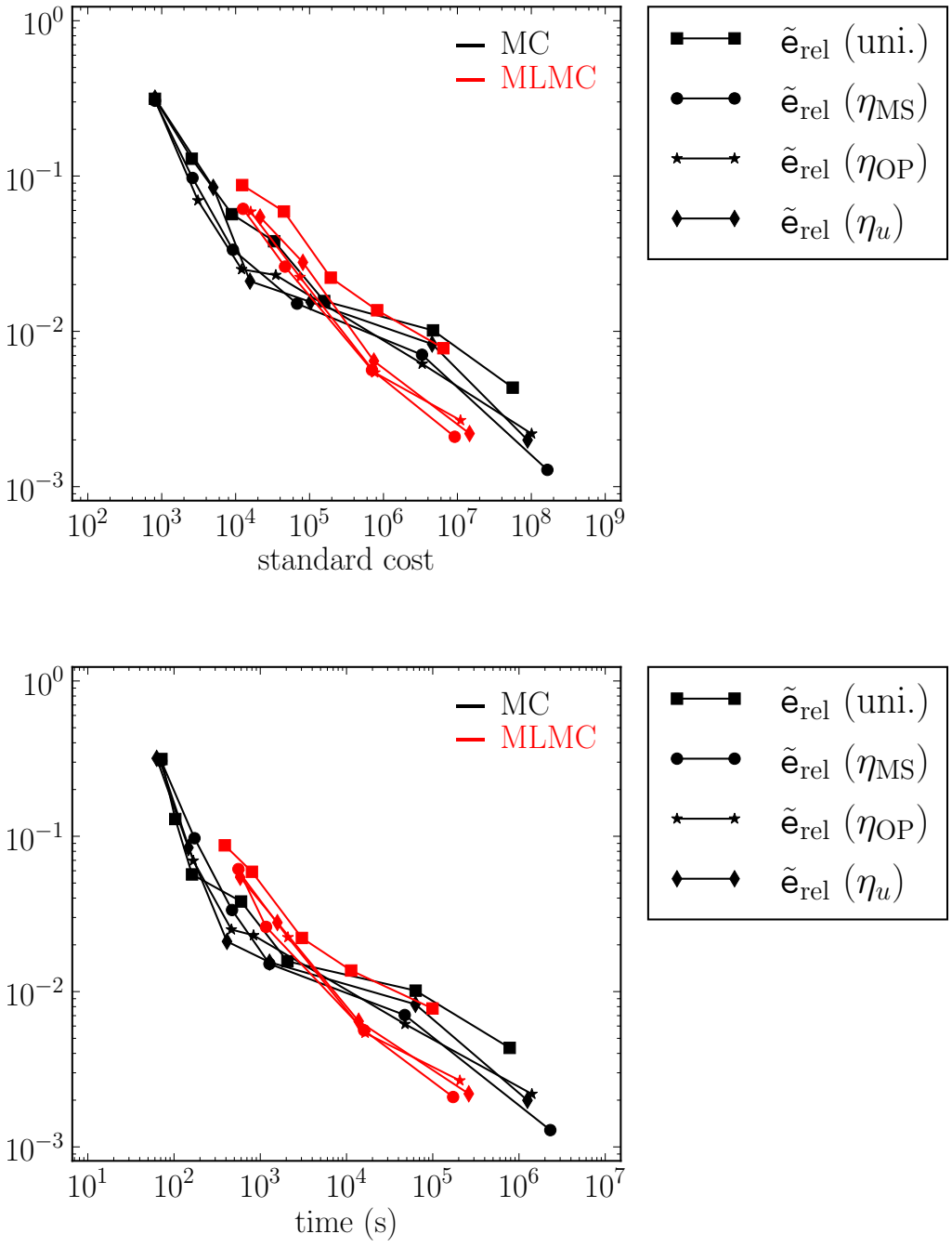


Figure 5.21: Experiment 5.1 Convergence with respect to standardized computational effort (top) and process time (bottom) for the **rough random field** with **10 samples** for the adaptive mesh generation.

Experiment 5.2 – Square Domain with an Exponential Buckle

This experiment adopts a slightly modified variant of Example 8.5 from [AO00] to the stochastic context. It is defined on the domain $D = [0,1]^2$ with homogeneous Dirichlet boundary data $u_D \equiv 0$ on Γ_D. Let us fix

$$u^* = 5x^2(1-x)^2(e^{10x^2}-1)y^2(1-y)^2(e^{10y^2}-1)$$

and define $f = \Delta u^*$. This right-hand side f will give the exact solution u^* for $\kappa \equiv 1$ as in the deterministic setting in [AO00] but, in general, the solution for any $\omega \in \Omega$ as well as the mean solution $\mathbf{E}[u]$ will be different from u^* as $\kappa \not\equiv 1$ almost surely. The random field κ is defined in the same way as the rough field in Experiment 5.1.

This right-hand side oscillates strongly, that is $\operatorname{osc}(f, \mathcal{T}) \gg 0$, which in turn induces large oscillations in the solution. In such a setting, the integration of the right-hand side during the assembly of the load vector F in Equation (1.2) is a challenge and enough care needs to be taking to suppress large numerical errors, which can dominate the overall error. In our numerical calculations, we employ fifth degree Gaussian quadrature with seven integration points according to the Strang and Fix scheme [SF71, Cha99].

The goal uses the same definition of g as in Equation (5.1) but with the location $x_0 = [0.6, 0.4]$ and the radius $r_g = 10^{-3}$. It resembles an approximation of the point evaluation for the solution at x_0. So, in effect, this problem exhibits no domain-induced singularity besides the four corners but high oscillations in the right-hand side f and the mean solution $\mathbf{E}[u]$ shown in Figure 5.22 next to its variance. Furthermore, the goal, presented in the same figure, has a much smaller support and is, thus, much harder in numerical terms.

We performed the same two experiments with a constant $\kappa \equiv 1$ and a deterministic κ. The resulting meshes for the first case are shown in Figure 5.23 and the deterministic κ is shown alongside the meshes for the second case in Figure 5.24. Once again, a clear distinction between the goal adaptivity and the solution adaptivity is apparent for η_{MS}, whereas the mesh generated with η_{OP} exhibits higher resolutions in the area connecting the two. As in the first experiment, the meshes for the rough κ show more noise and less symmetry.

In both cases, we observe a longer preasymptotic phase in Figure 5.25 due to the small support of the quantity of interest which needs to be resolved. Notably the biggest difference of the two cases is the lack of convergence when η_u is used for refinement in the rough case, as it does not account well enough for the error in the quantity of interest.

The bounds and estimators perform differently for the different methods in both cases. While for η_{MS} the bounds and estimators perform almost similarly, the difference for the uniform methods is the biggest. Here, the estimator is much sharper compared to the other methods. Also, in the constant case the uniform method performs just as good at 10^5 degrees of freedom as the goal-adaptive methods.

Adaptivity solely with respect to the primal problem performs the worst however. In the rough case, only the goal-driven adaptive methods perform similar to the constant case. The uniform method achieves an error of about one magnitude worse or for the same error it needs about two orders of magnitude more degrees of freedom. The primal adaptive method is unfeasible as hardly any error reduction is visible after 10^3 degrees of freedom.

The preasymptotic phase for the estimators and bounds lasts as long as 10^4 degrees of freedom. It is not until then that they give a good representation of the true error. This will be a great hindrance for reliable heuristics. A special behavior is exhibited by the guaranteed bounds when applied together with η_{MS}. This is due to the nature of the adaptive algorithm, which in that case chooses to refine for either the primal or the dual error indicator.

As the quantity's support in this experiment is very small, the algorithm will choose to refine almost exclusively with respect to the dual error indicator in the beginning. This, however will lead to slow or no convergence at all for the global error in the L^2 or H^1 norms. As a result, the bounds sharpness is very limited as they rely on norms of the global solutions and tend to give worst case estimates. Subsequently, the bounds will show bad performance as long as the primal problem is not sufficiently resolved by the mesh. With more than 300 degrees of freedom the guaranteed bounds behave similar to the variant with η_{OP}.

The global convergence in Figure 5.26 shows similar behavior as in Figure 5.7 for Experiment 5.1. The overestimation constant for the estimators once again depends on the random field as it is only independent of κ with respect to the energy norm $\|\!|u - u_h|\!|_{\omega}$. In contrast, the convergence for the uniform meshes shows the same error reduction rate as for the adaptive meshes albeit those need up to 10 times less degrees of freedom to achieve the same error.

The performance of $\alpha_{\mathrm{opt}}(\omega)$ from Remark 3.14 as the choice of the parameter α in the guaranteed bounds is shown in Figure 5.35. Yet again, this approximation proves to be almost optimal for both example realizations of the random field.

The differences between the refinement algorithms become obvious in the stochastic case as pictured in Figure 5.27. The mesh sequences were generated using 100

samples in each step. The refinement with η_{MS} will refine just in the same areas as the primal and dual indicators would respectively. However, with the η_{OP} indicator, refinement also occurs in the area connecting the two main refinement spots, while coarser triangles are observed in the support of the quantity and the upper right corner. A big difference, compared to the first experiment, can be seen in Figure 5.29. For both, Monte Carlo and multilevel Monte Carlo, η_{OP} performs among the best. For Monte Carlo, the adaptivity with η_{MS} performs worse than expected due to hardships with the heuristics. This is less of a problem with the multilevel variant if the practical approach from Section 3.4 is used.

The performance of the bounds differs greatly from the first experiment, as this one has much bigger oscillations in the problem data and the solution. Only the η_{MS} method delivers sharp and thus useful bounds. During the preasymptotic phase, it performs worse than η_{OP} due to the bad performance seen in Figure 5.25. Once these are resolved, the bounds are stable and sharp up to one order of magnitude. In contrast, the bounds for η_{OP} meshes are close to 10^3 overestimation in the multilevel Monte Carlo method and the bounds for uniform and primal adaptive meshes overestimate to an extend where they loose any usefulness.

In Figure 5.31 the efficiency of the two algorithms is evaluated with respect to the standardized cost and actual running time. In Figure 5.33, the same data is presented except for the bounds to achieve better scaling. Similar results as in Experiment 5.1 are achieved though it takes longer for the multilevel algorithm to become competitive due to the higher variance of the solution. Moreover, struggling heuristics can be observed for the multilevel method in connection with η_{MS}, which is only overcome for meshes with more than 10^4 degrees of freedom.

Overall, this experiment shows the limits of the presented methods. When considering the bounds, the η_{MS} method is the only feasible mesh strategy out of the presented ones. If only the error is of concern, the η_{OP} meshes have superior characteristics with respect to the error in the quantity of interest. In the asymptotic phase, however, both goal-adaptive error indicators perform equally well.

Notably, goal-adaptive meshes improve the effectiveness of the methods considerably and similar to the first experiment the advantages of multilevel Monte Carlo and goal-adaptive meshes do add up as before, though it takes longer for the multilevel Monte Carlo method to break even. In fact, up to 10^4 degrees of freedom goal-adaptive Monte Carlo outperforms or at least performs just as well as goal-driven multilevel Monte Carlo. For uniform meshes, this break-even point is much earlier with around $3 \cdot 10^3$ degrees of freedom. The advantage of adaptive multilevel Monte Carlo over uniform Monte Carlo on finer grids is approximately three orders of magnitude and, thus, well worth the effort.

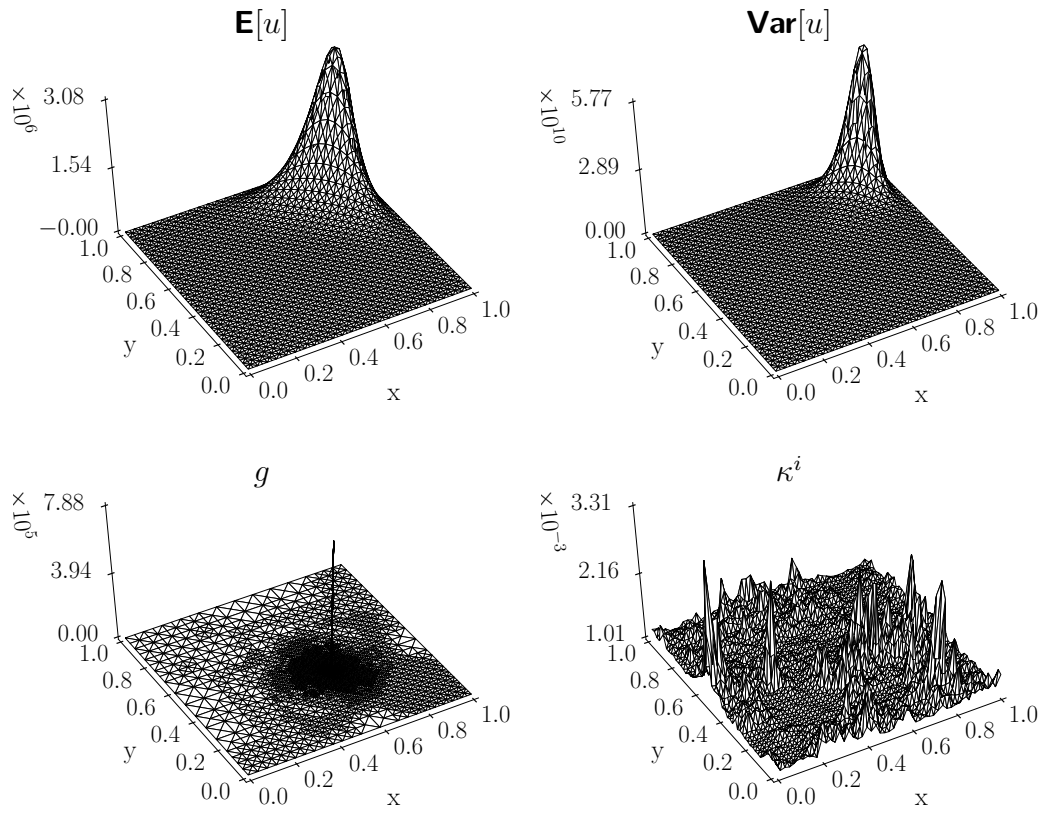


Figure 5.22: Experiment 5.2 The mean solution, the variance, and the weight for the quantity of interest and a realization κ_i of the random field.

Similar to Experiment 5.1 the number of samples used for the generation of the mesh sequence does not have a significant influence on the results. This can be seen in Figure 5.28 for the meshes, in Figure 5.30 for the error with respect to the degrees of freedom, and Figure 5.32 or Figure 5.34 for the error with respect to the computational effort. Please note that, compared to the computations with 100 mesh samples, this experiment stops at less finer meshes with less degrees of freedom due to time constraints during the calculations.

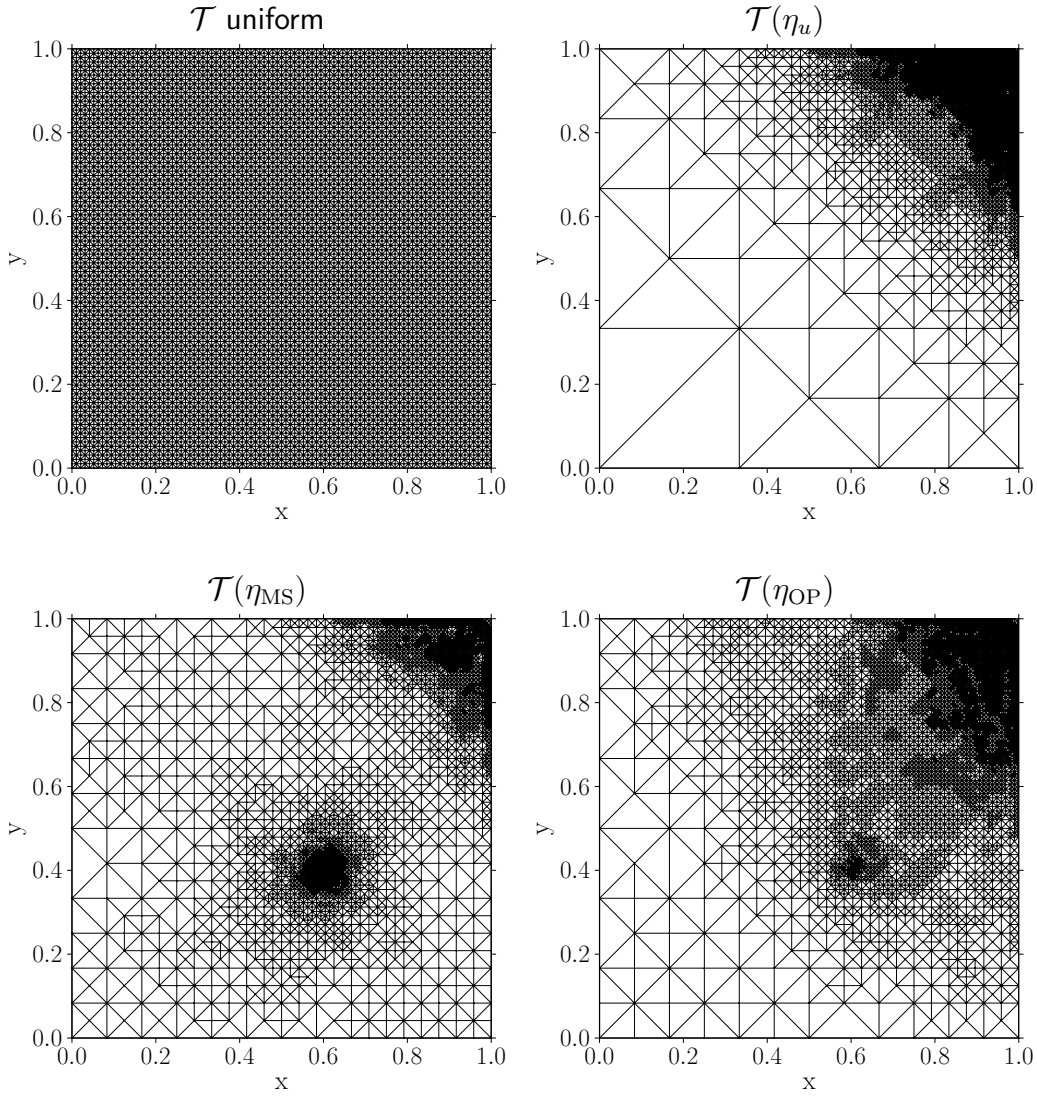


Figure 5.23: Experiment 5.2 The uniform mesh and meshes generated with the refinement indicators η_u, η_{MS}, and η_{OP} for the **deterministic** constant $\kappa \equiv 1$ with approximately 4100 degrees of freedom which result in the first graph in Figure 5.25.

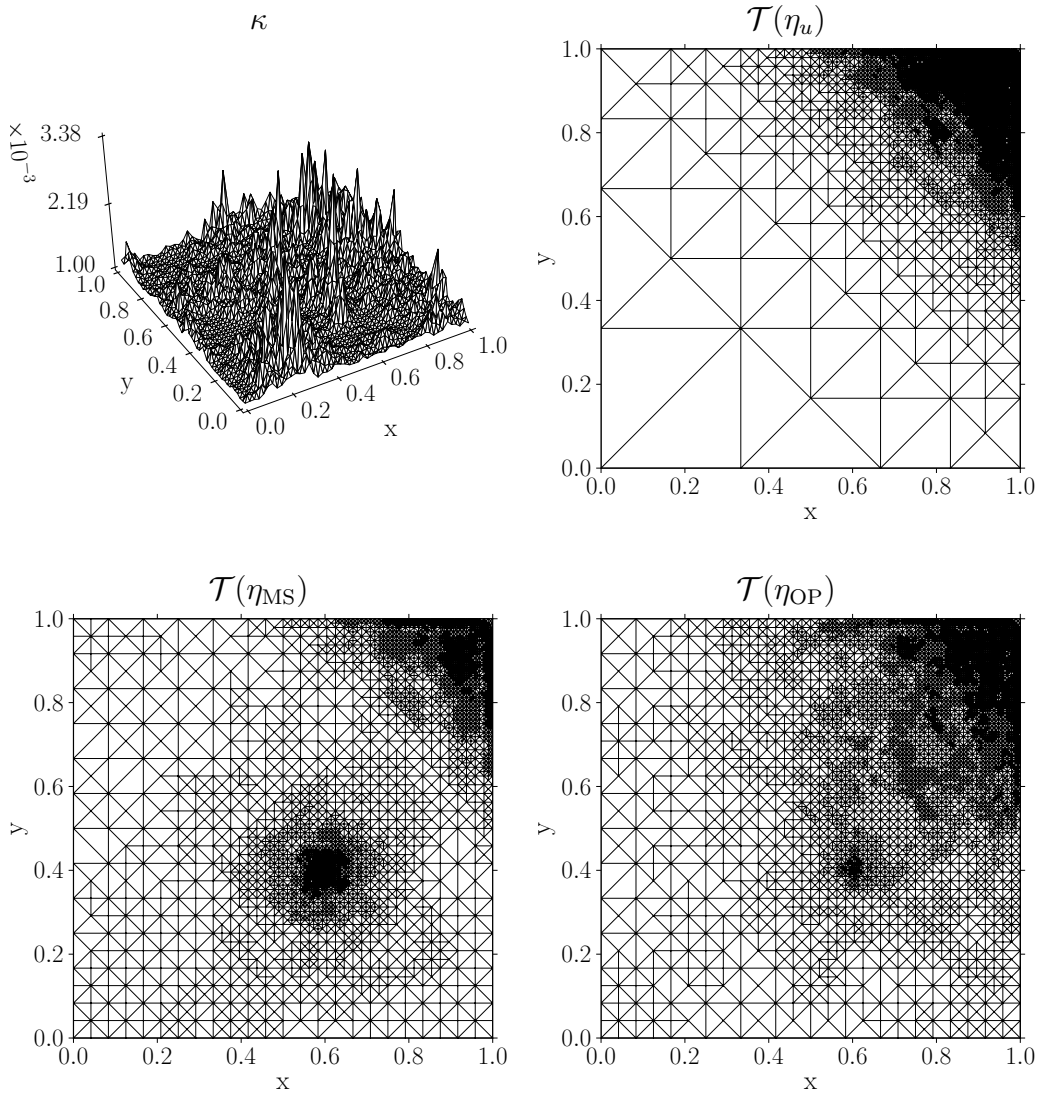


Figure 5.24: Experiment 5.2 The **deterministic** rough κ and the resulting meshes from the refinement indicators η_u, η_{MS}, and η_{OP} with approximately 4100 degrees of freedom which result in the second graph in Figure 5.25.

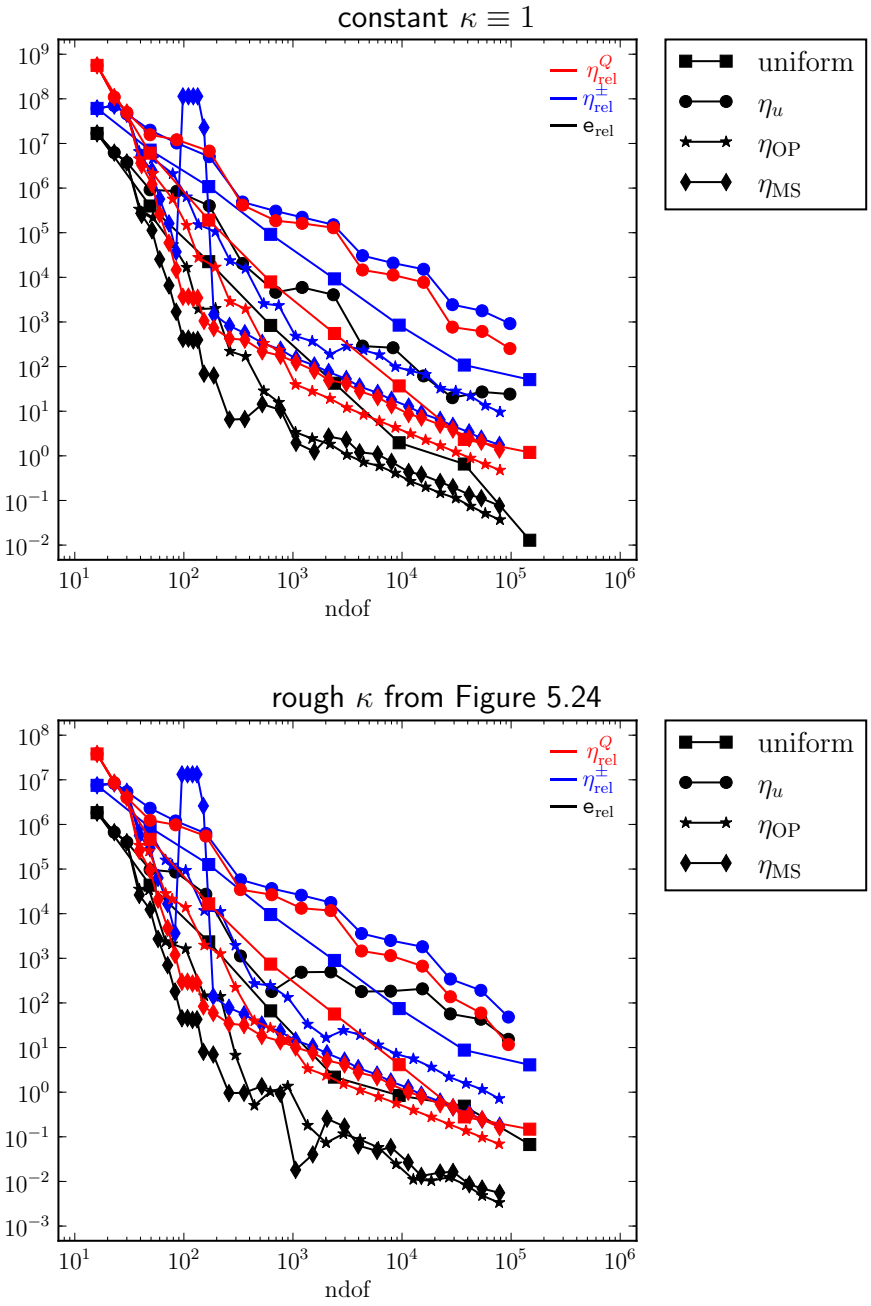


Figure 5.25: Experiment 5.2 Convergence of the error, the bounds, and the estimator with the refinement strategies uniform, η_u, η_{OP}, and η_{MS} for two **deterministic** κ with respect to the number of degrees of freedom. The short notations are $\eta^Q_{\mathrm{rel}} = \eta_\circ / |Q(u)|$ with the refinement indicator $\eta_\circ$, $\eta^{\pm}_{\mathrm{rel}} = |\eta^+ - \eta^-| \,/\, |Q(u)|$, and $\mathsf{e}_{\mathrm{rel}} = |Q(u) - Q(u_h)| \,/\, |Q(u)|$.

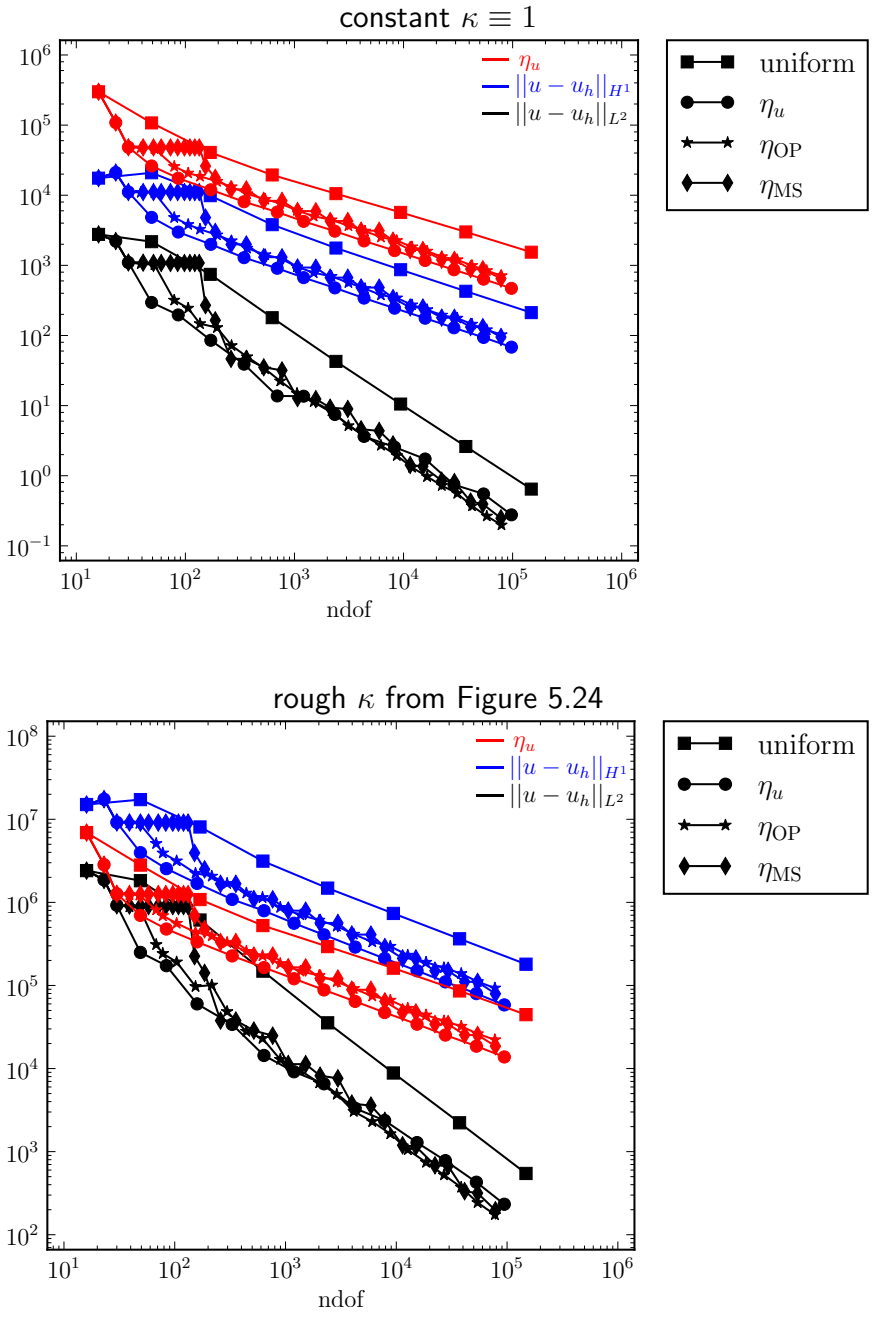


Figure 5.26: Experiment 5.2 Convergence of the global error in the H^1 norm, the L^2 norm, and the estimator with the refinement strategies uniform, η_u, η_{OP}, and η_{MS} for two **deterministic** κ with respect to the number of degrees of freedom.

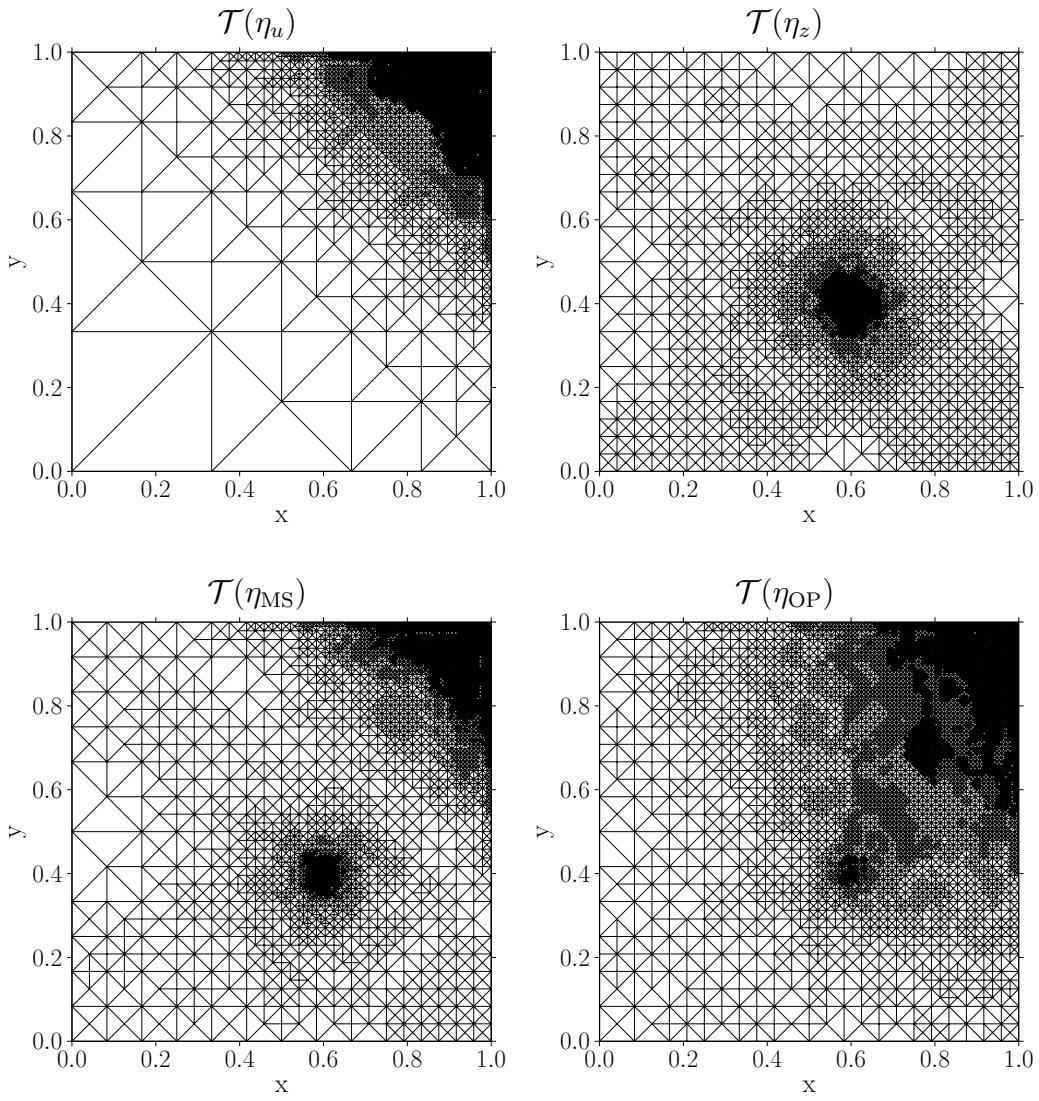


Figure 5.27: Experiment 5.2 The meshes in the stochastic setting generated by different error estimators with approximately 9000 degrees of freedom with **100 samples** for the adaptive mesh generation.

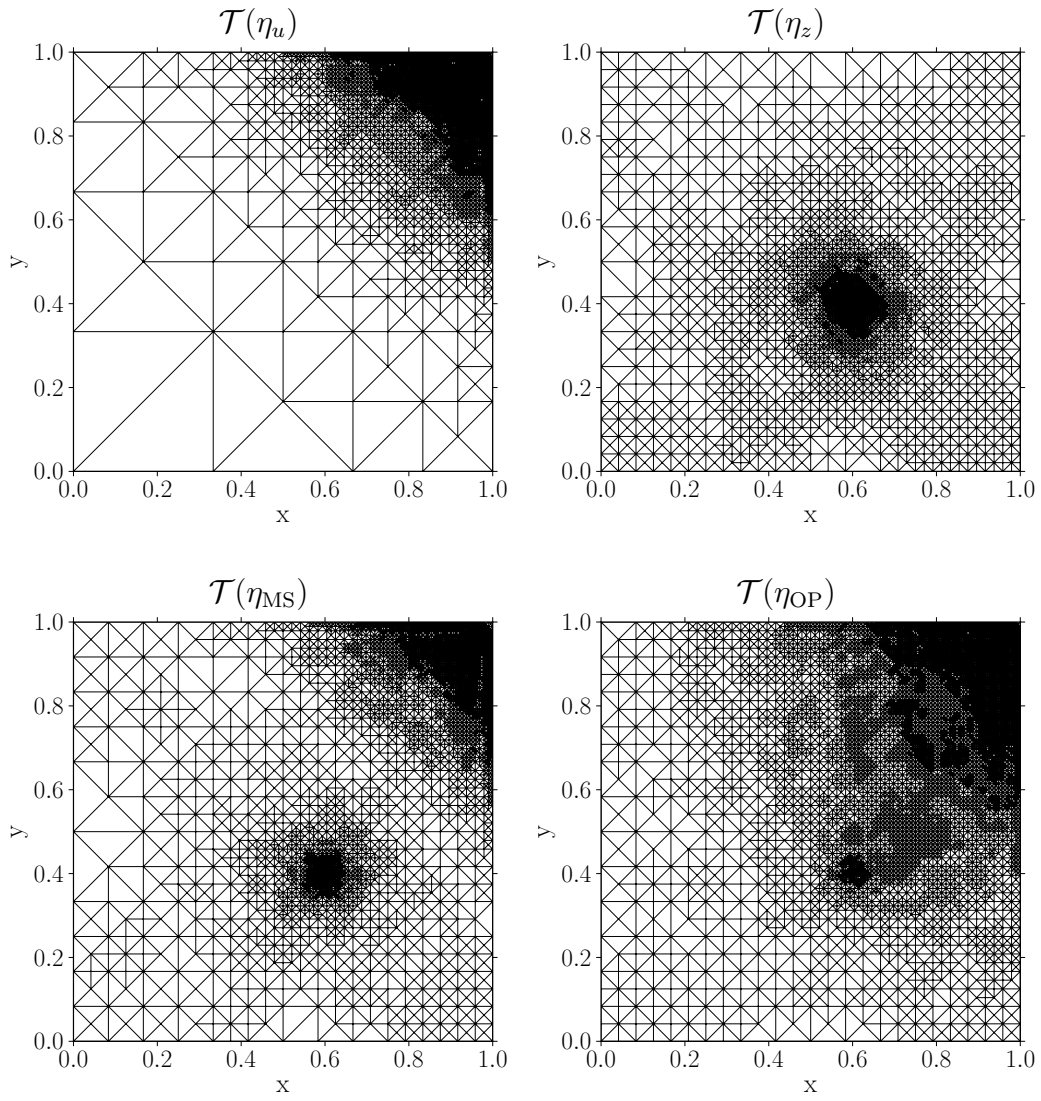


Figure 5.28: Experiment 5.2 The meshes in the stochastic setting generated by different error estimators with approximately 9000 degrees of freedom with **10 samples** for the adaptive mesh generation.

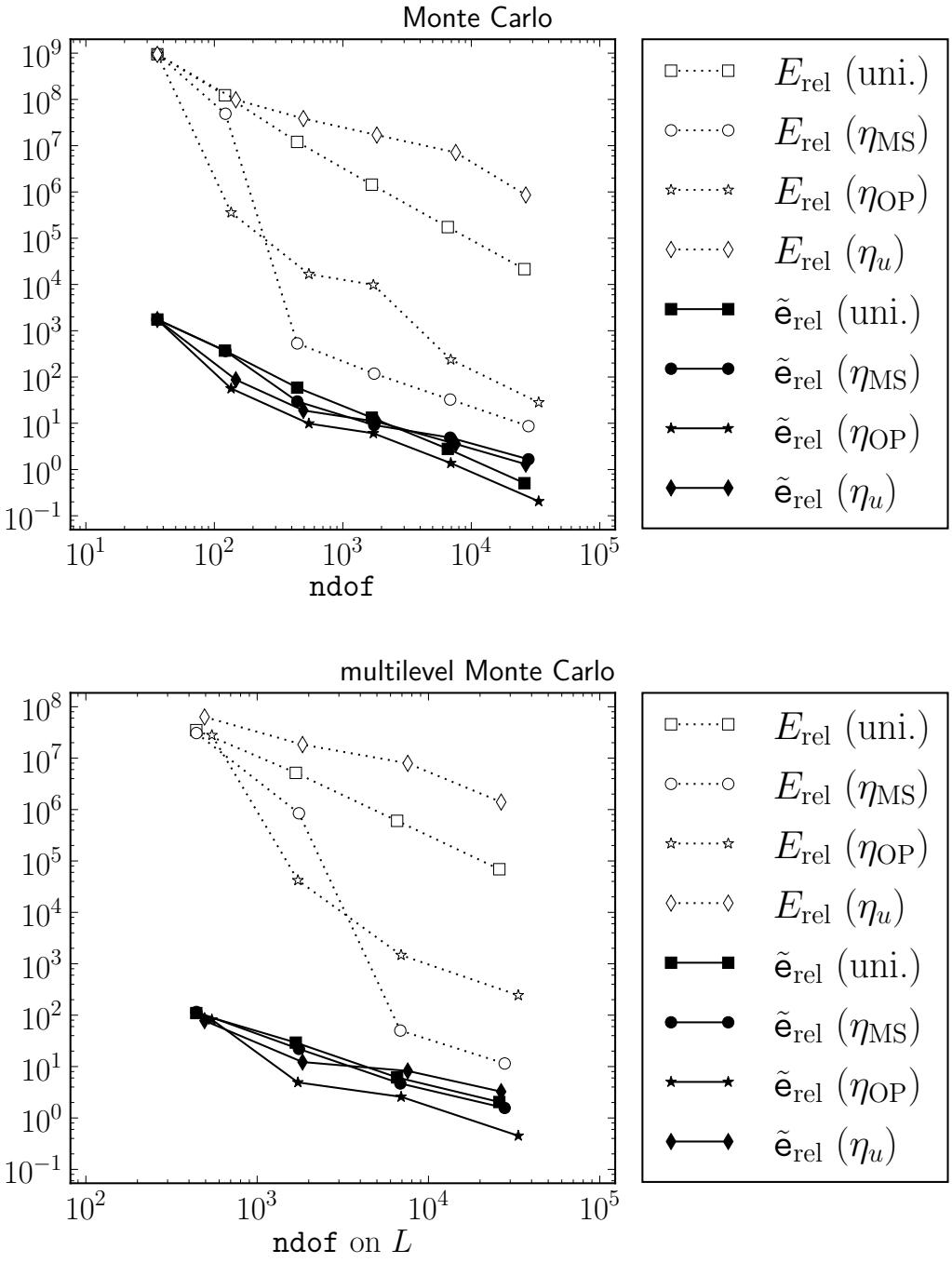


Figure 5.29: Experiment 5.2 Convergence and efficiency of the bounds with respect to the number of degrees of freedom for the Monte Carlo and multilevel Monte Carlo method **100 samples** for the adaptive mesh generation.

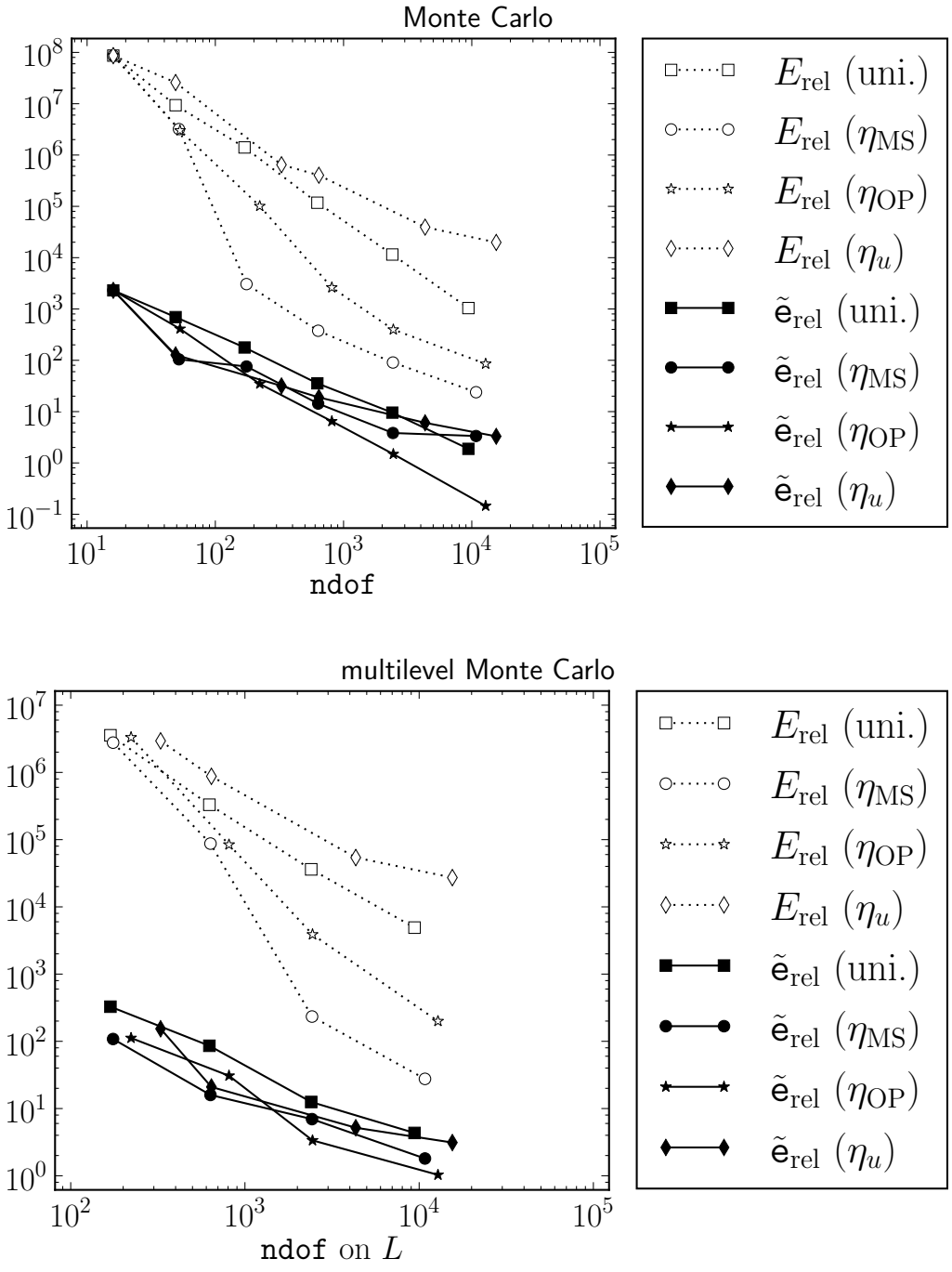


Figure 5.30: Experiment 5.2 Convergence and efficiency of the bounds with respect to the number of degrees of freedom for the Monte Carlo and multilevel Monte Carlo method **10 samples** for the adaptive mesh generation.

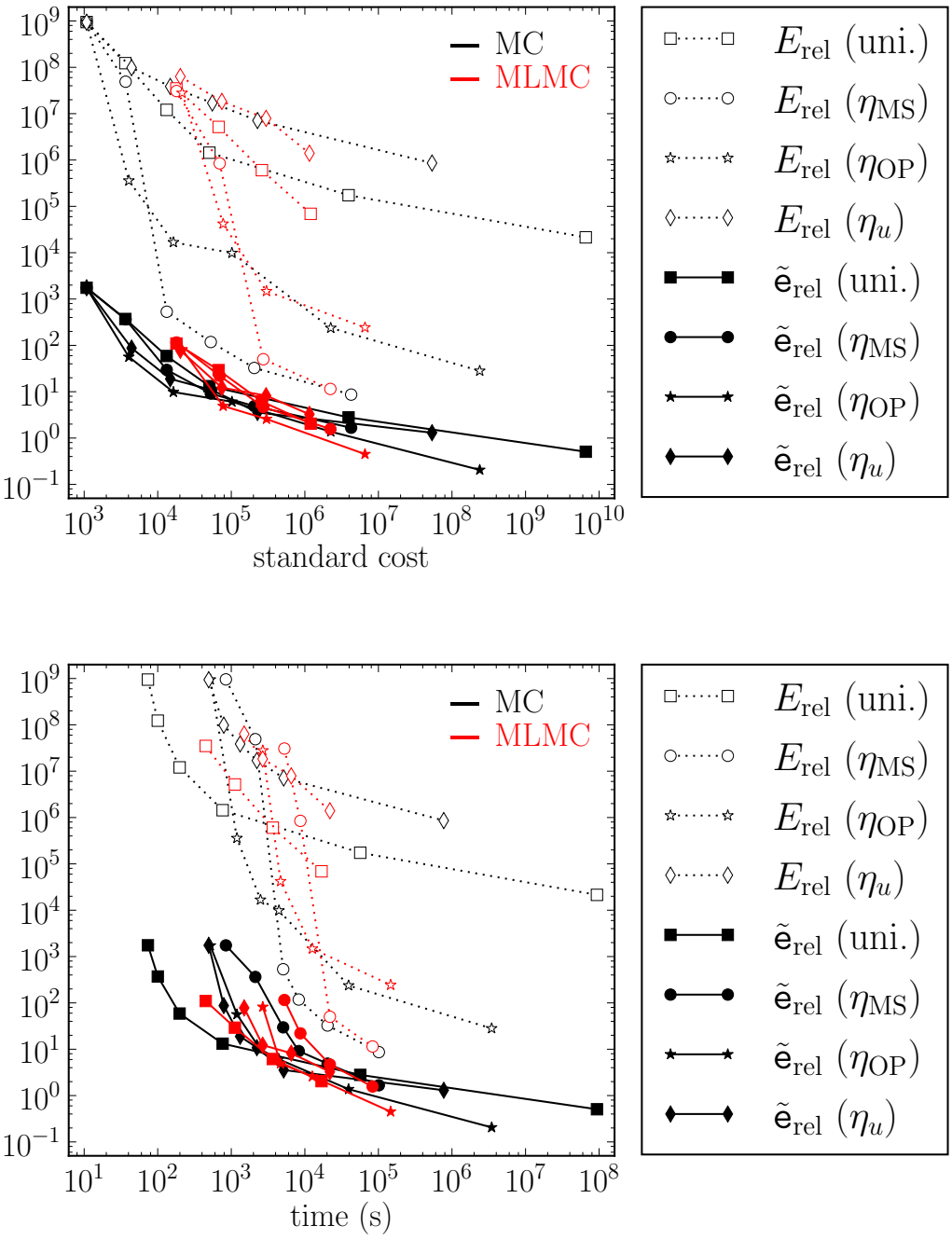


Figure 5.31: Experiment 5.2 Convergence and efficiency of the bounds with respect to standardized computational effort (top) and process time (bottom) **100 samples** for the adaptive mesh generation.

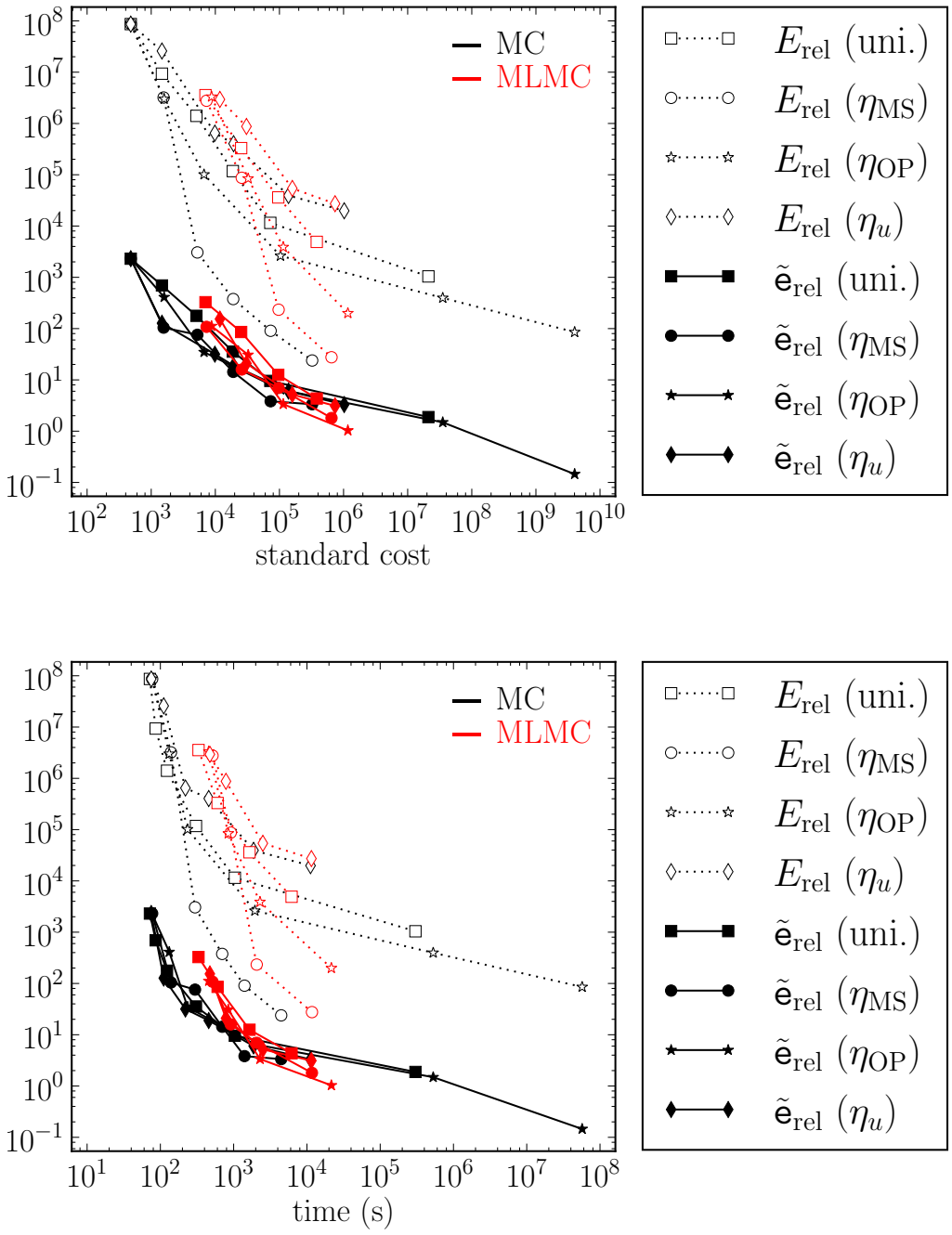


Figure 5.32: Experiment 5.2 Convergence and efficiency of the bounds with respect to standardized computational effort (top) and process time (bottom) **10 samples** for the adaptive mesh generation.

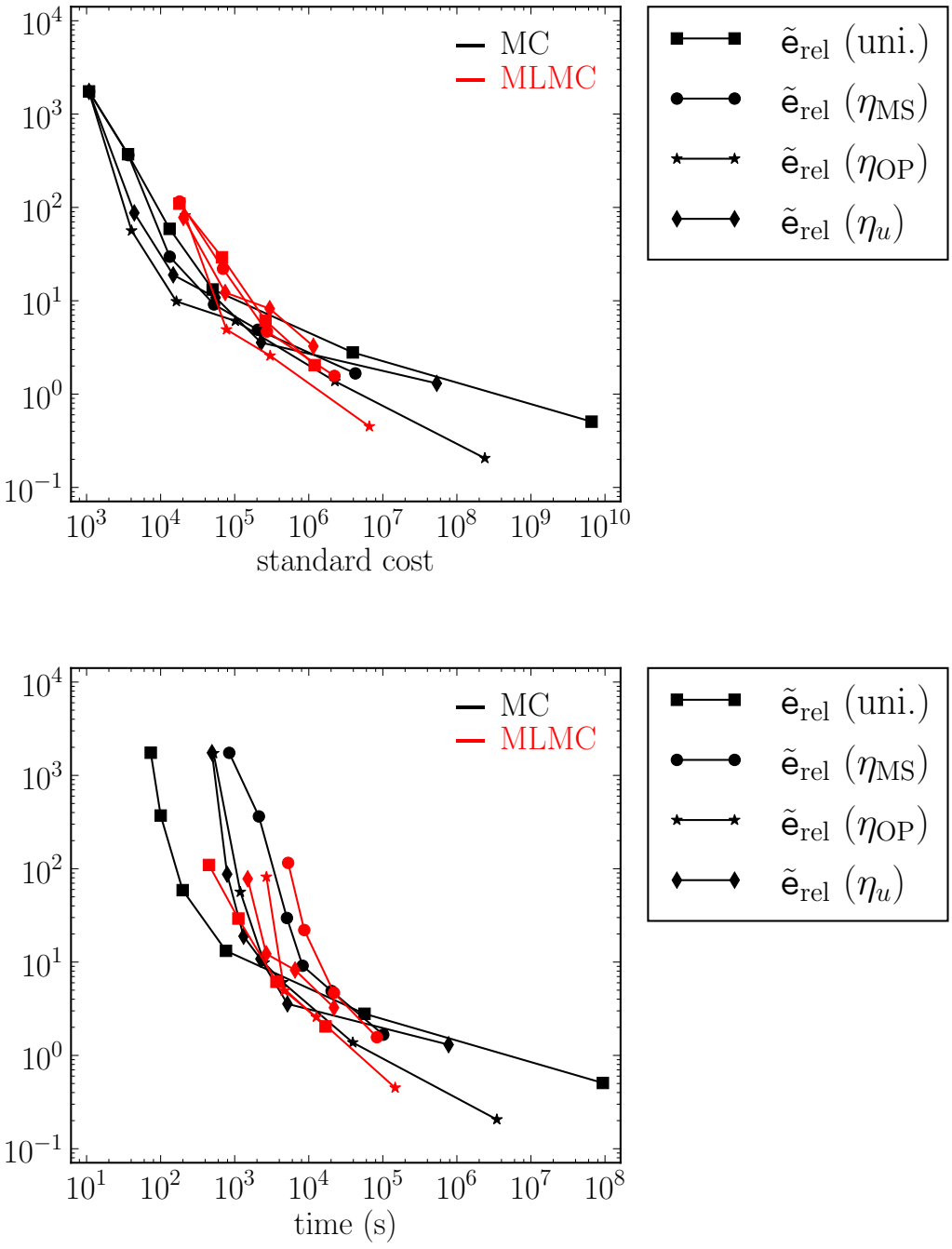


Figure 5.33: Experiment 5.2 Convergence with respect to standardized computational effort (top) and process time (bottom) **100 samples** for the adaptive mesh generation.

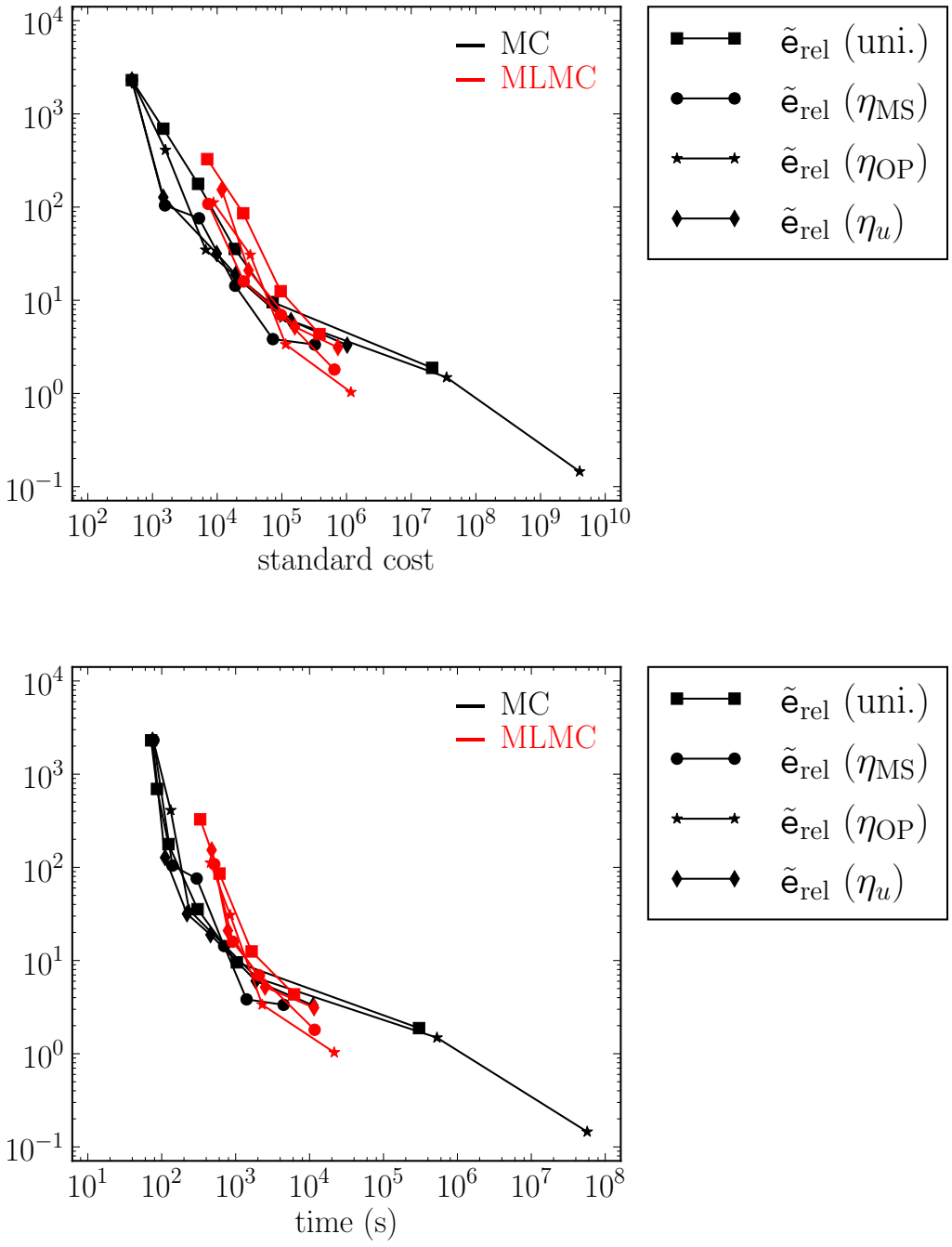


Figure 5.34: Experiment 5.2 Convergence with respect to standardized computational effort (top) and process time (bottom) **10 samples** for the adaptive mesh generation.

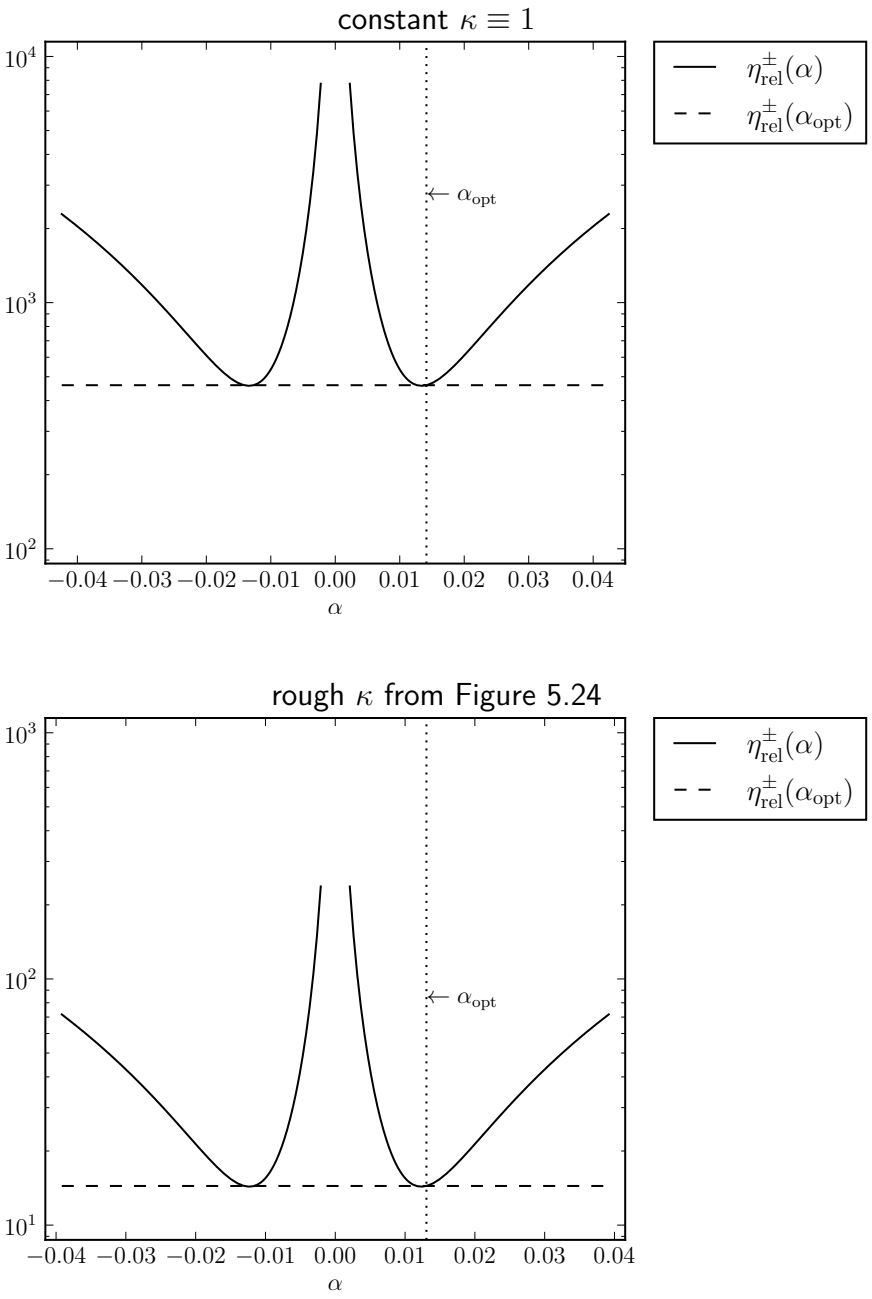


Figure 5.35: Experiment 5.2 Performance of the bounds with respect to the choice of α compared to α_{opt} for the two **deterministic** κ on adaptive meshes using η_{MS} with approximately 1000 degrees of freedom.

Experiment 5.3 – On the Efficiency of the path-wise Approach

In Section 3.3 the additional path-wise refinement was suggested as an alternative to the the mean error indicators. The idea is to resolve sample related oscillations with indicators depending on ω. Preliminary numerical tests could not produce the desired results as shown in Figure 5.40. The deterministic convergence experiments in Figure 5.6, Figure 5.7, Figure 5.25, and Figure 5.26 plotted the error with respect the number of degrees of freedom on the last level. This however only represents the memory usage and ignores the time it takes to build the adaptive mesh sequence.

In stochastic sampling approaches, however, the time it takes to compute a sample is usually more important than the memory it takes to compute that solution. Hence another experiment will show, how the different adaptive algorithms perform with respect to the computational effort for different choices of the bulk parameter θ in Equation (3.14). Besides the time it took to generate the whole adaptive sequence which includes the computation of the dual solution, the error estimators, and the refined meshes we will plot the error with respect to the standardized computational effort for the primal solution defined by the sum over all levels of the number of degrees of freedom.

In Figure 5.36 and Figure 5.37 the convergence is plotted for Experiment 5.1 with the rough deterministic κ from Figure 5.5. The adaptive algorithms show a good advantage in both the goal-driven as well as the global setting but lag the improved convergence rate seen in Figure 5.6. However, the adaptive algorithm resolves a singularity which is geometry induced and, hence, can be sufficiently resolved by the mean meshes of the other methods in Section 3.3. The overhead of the methods employing mean indicators is independent of the number of samples on the levels and thus likely outperform the path-wise adaptive meshes as in Figure 5.40.

In Figure 5.38 and Figure 5.39 the same experiment is repeated for Experiment 5.2. We get sufficient advantage in the global setting but in the goal-driven setting the best adaptive algorithms give only a small advantage compared to the uniform meshes and they take time to compensate for the overhead in the beginning. In the stochastic setting depicted in Figure 5.40 a considerable amount of the adaptive gain can be realized with the mean meshes and hence the path-wise approach is yet again less efficient compared to the mean indicator meshes and the even the uniform meshes. The bulk parameter for those experiments was set to θ for both methods.

Note that different choices of θ might result in more efficient meshes for the cost of smaller steps with regard to the number of degrees of freedom. For the path-wise approach this becomes a problem as shown above. The mean meshes however can employ different θ with only a small time penalty.

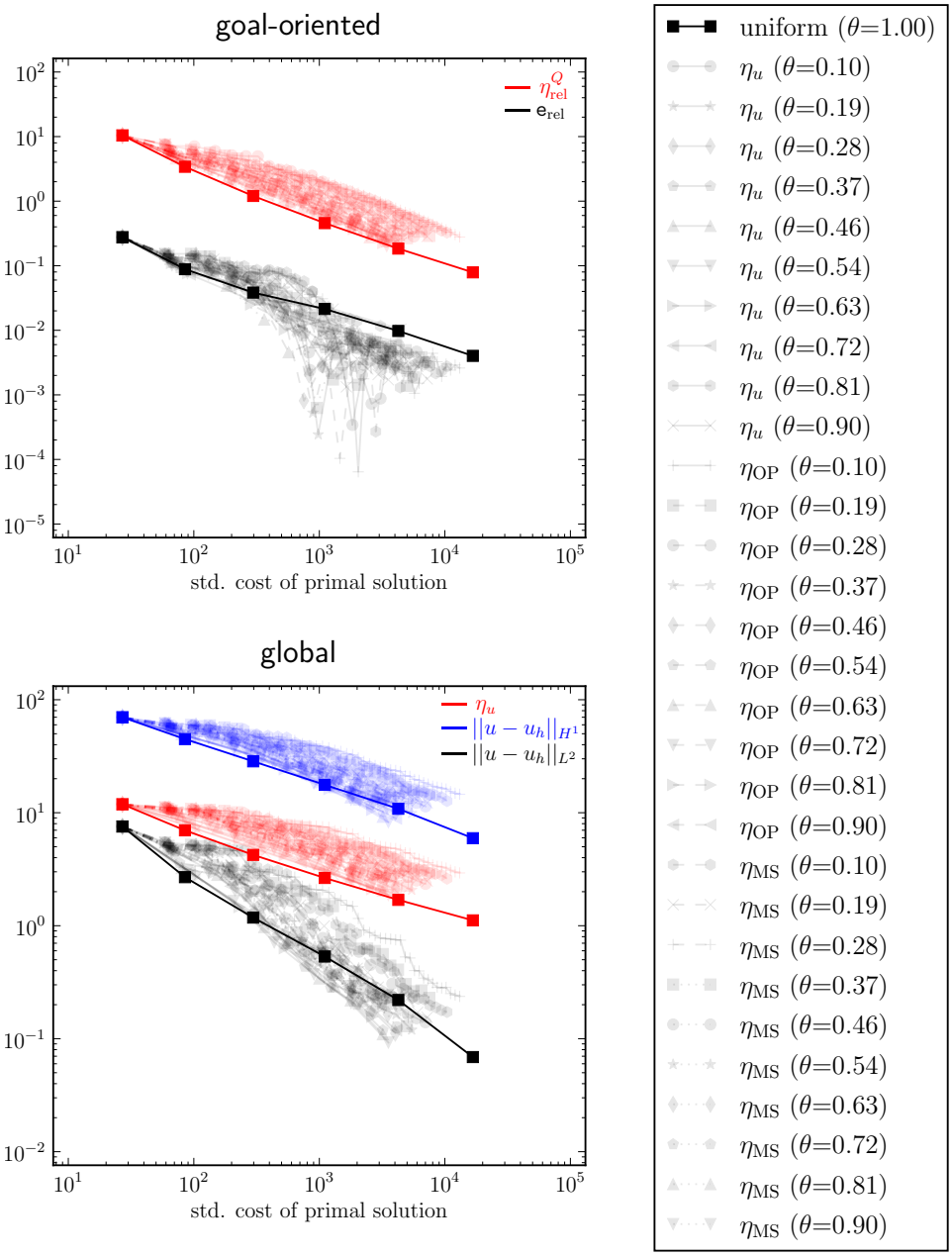


Figure 5.36: Experiment 5.3 Deterministic convergence with respect to the cumulative standardized computational effort for **Experiment 5.1** with the **rough** κ in Figure 5.5 for different choices of θ.

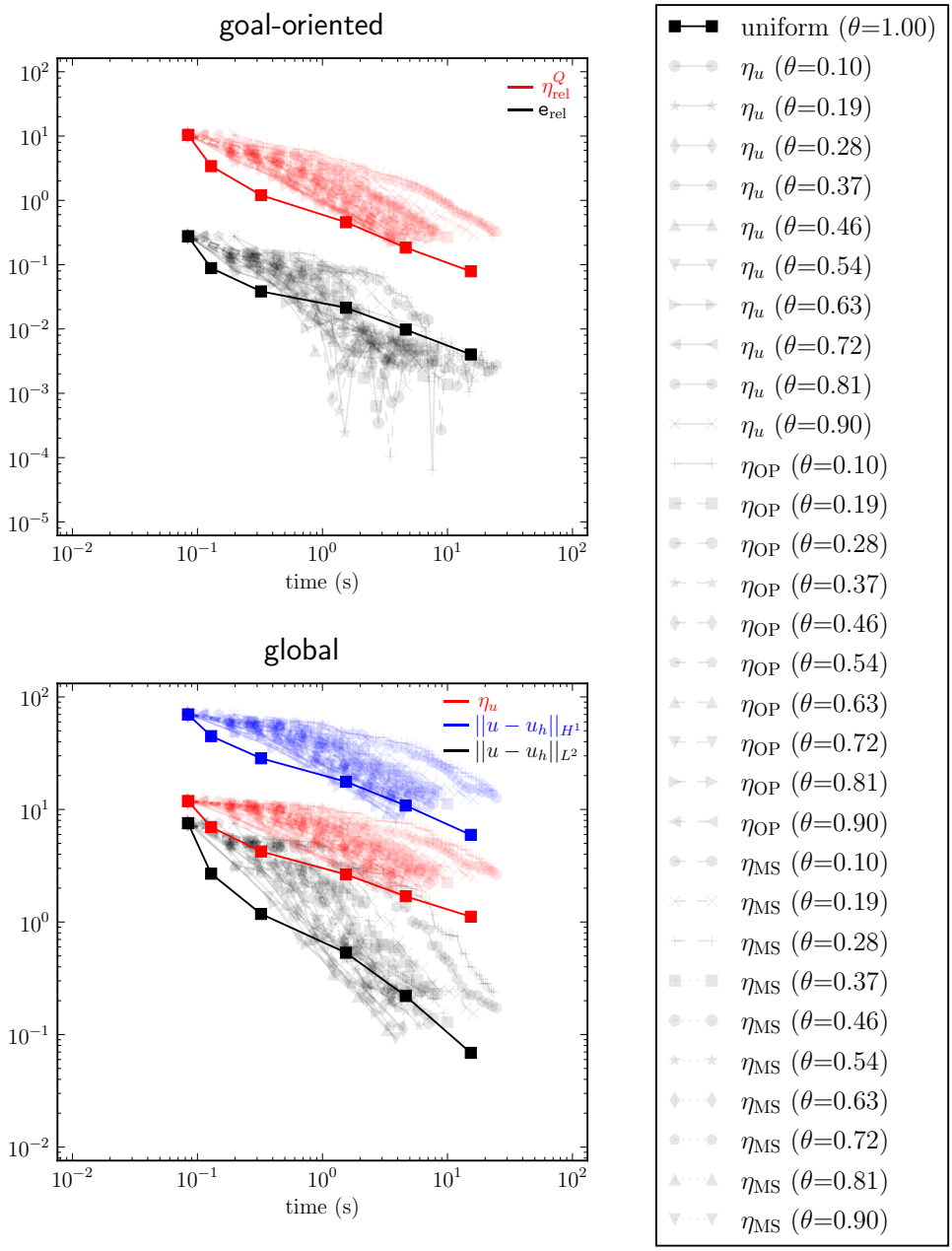


Figure 5.37: Experiment 5.3 Deterministic convergence with respect to the time for **Experiment 5.1** with the **rough** κ in Figure 5.5 for different choices of θ.

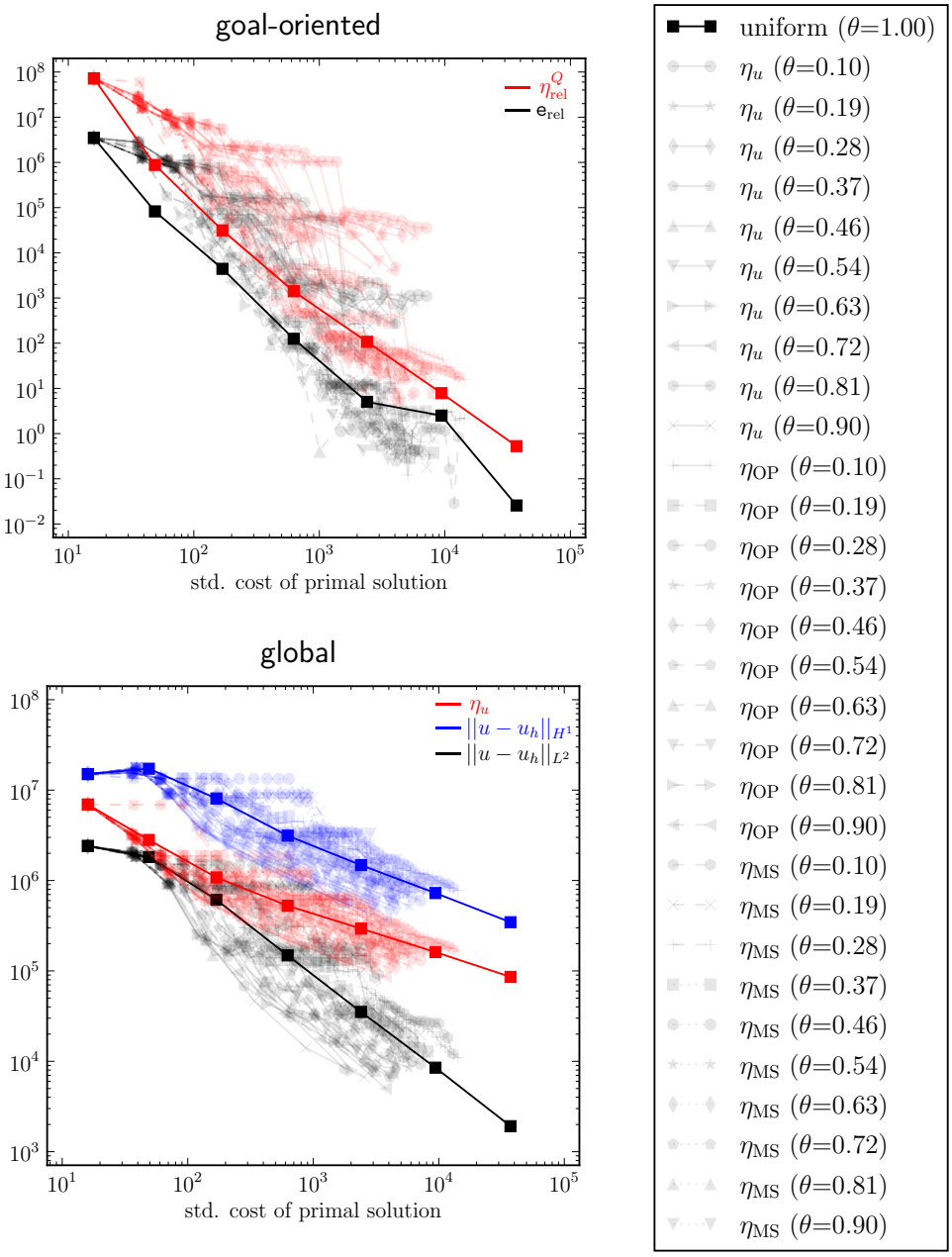


Figure 5.38: Experiment 5.3 Deterministic convergence with respect to the cumulative standardized computational effort for **Experiment 5.2** for different choices of θ.

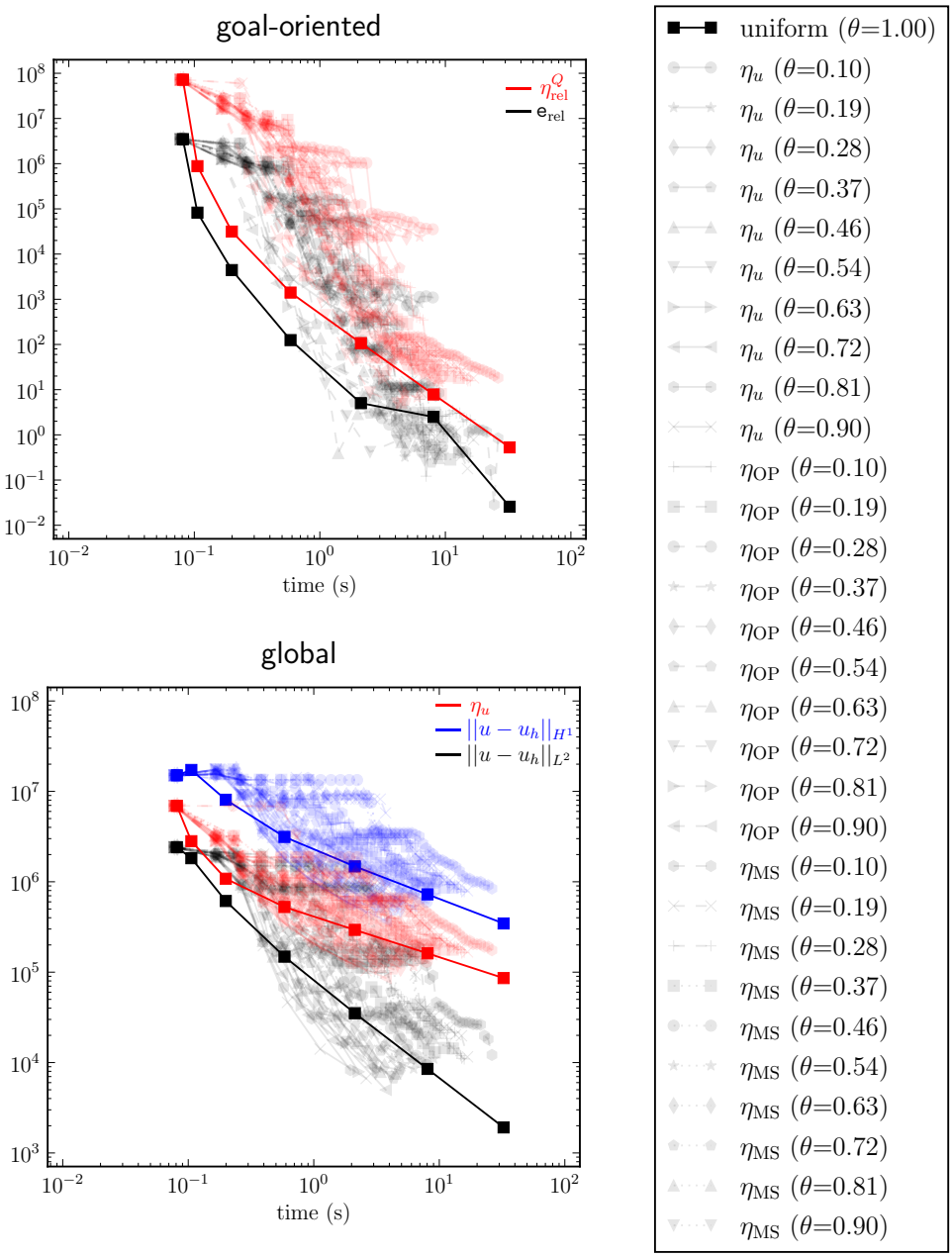


Figure 5.39: Experiment 5.3 Deterministic convergence with respect to the time for **Experiment 5.2** for different choices of θ.

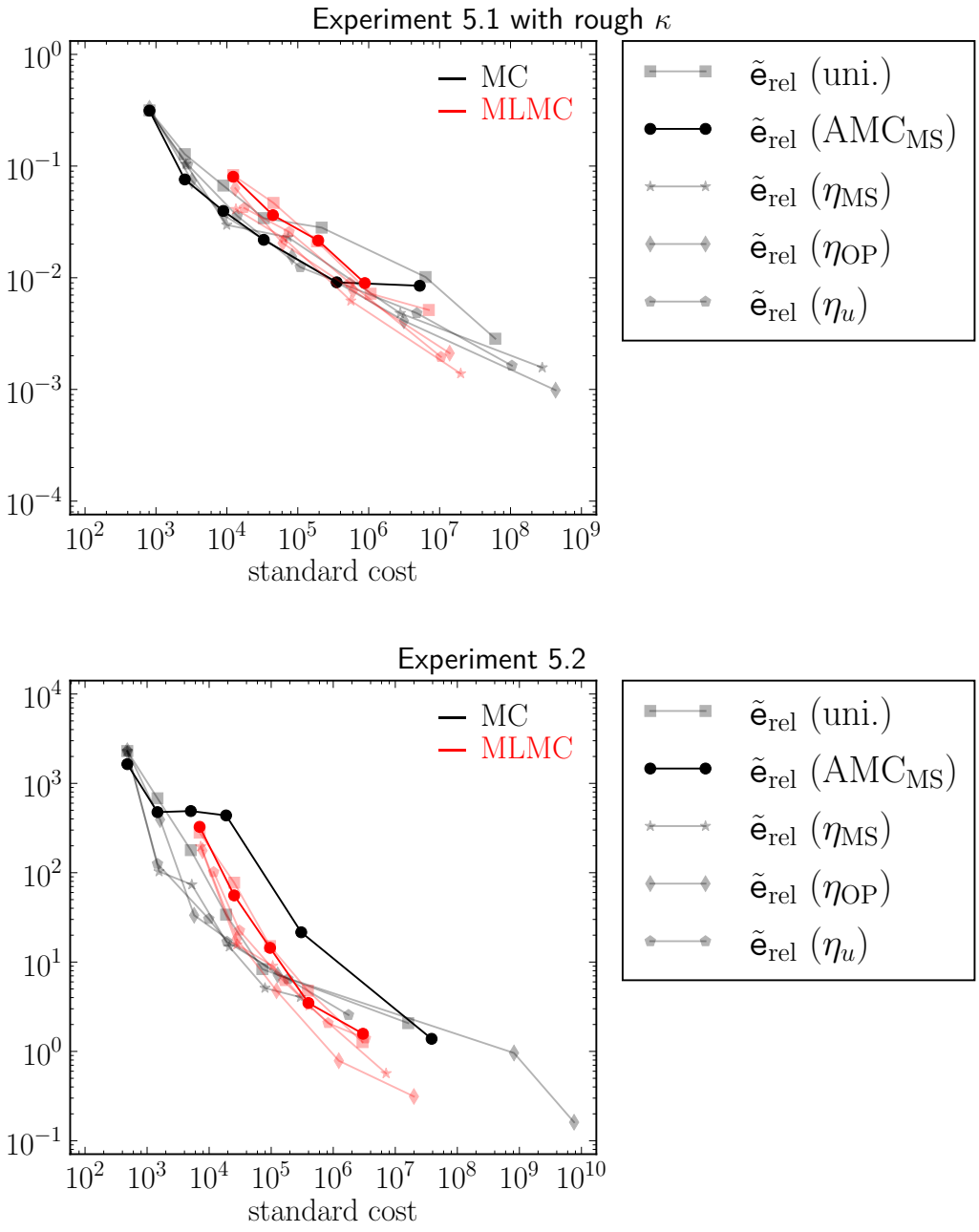


Figure 5.40: Experiment 5.3 Convergence with respect to standardized computational effort for the two previous examples with the rough random field. The $\mathrm{AMC}_{\mathrm{MS}}$ graph depicts the preliminary path-wise algorithm from Section 3.3 with $\theta = 0.5$.

5.3 Alternative Approach

Experiment 5.4 – Smooth Field for the Exponential Buckle

The next experiment will cover the alternative solution method from Chapter 4. The visualization of these methods has to differ from the last experiments as one of the biggest advantages of this method over the Monte Carlo and multilevel Monte Carlo method is its proper separation of the spatial and stochastic domain. This allows to choose the number of samples locally based on the variance in any part of the domain. For comparison purposes we compute the solution for the experiments with a fixed number of samples for all the points in the domain using uniform and adaptive meshes and we use different numbers of samples for the same meshes. This results in four variants of the experiment.

The input data for this experiment is the same as for Experiment 5.2 except for the random field, which is based on the smooth benchmark field in Equation (2.8). Its parameters are $t = 5$, $\varepsilon_\kappa = 0.5 \cdot 10^{-3}$, $\sigma_\alpha = 2$, $A = 0.6$, and $c_\kappa = 1$. The minimum number of samples N_{init} is always set to 100. The mean and variance of the solution, as well as an adaptive finite element mesh from the method in Section 3.3 as well as a realization of the random field κ are shown in Figure 5.42. In Figure 5.43 the number of samples is plotted for each vertex in the domain using the P_1 interpolation.

This, however, gives a wrong impression of the actual computational effort invested in every part of the domain as the density of the adaptive meshes varies greatly in the domain. Instead, Figure 5.44 gives a better overview over the invested time in each part of the domain. It visualizes for each triangle $T \in \mathcal{T}_\ell$ the average number of samples on that triangle divided by its area. Let N_1, N_2, N_3 be the number of samples in the nodes of the triangle, then the values presented in Figure 5.44 are defined as

$$\mathcal{C}_T = \frac{1}{3\,|T|} \sum_{i=1}^{3} N_i. \tag{5.2}$$

Here, it becomes apparent that a uniform distribution of the samples is not efficient. Adaptive sampling, related to stratified sampling, will allocate most samples to the areas with the highest variance, whereas uniformly distributed samples will always assume worst case variances for each point in the domain. In case of adaptive meshes, the value of $\mathcal{C}_T$ is quite small for large triangles even for the uniformly distributed samples but as it does not cater for the local variance, small boundary triangles exhibit a large computational effort per area $\mathcal{C}_T$ despite their small variance. Only the connection of the two methods results in the best approximation behaviors out

of the approaches presented here.

The convergence for these methods is presented in Figure 5.41 for both the L^2 and the H^1 norm. Note that the finite element error estimator optimizes the meshes with respect to the energy norm, that is for the H^1 norm. As a result, the best performance of the mesh adaptive methods can be observed in the H^1 norm. The best result is achieved with the combination of the methods. Almost as good as this is the mesh adaptive only version. Solely, sampling adaptivity performs in third place with almost one order of magnitude more computational effort necessary to achieve the same error compared to the best, while uniform meshes together with a uniform sample distribution performs almost two orders of magnitude worse than the first.

A different picture is drawn for the L^2 norm. The all uniform method is still the worst, while the other methods are comparatively equal with a slight advantage for the two adaptive sampling methods. Suitable error estimators in the L^2 norm should improve these results if necessary. Nevertheless, even in the L^2 norm, the best method is one order of magnitude faster than the all uniform method.

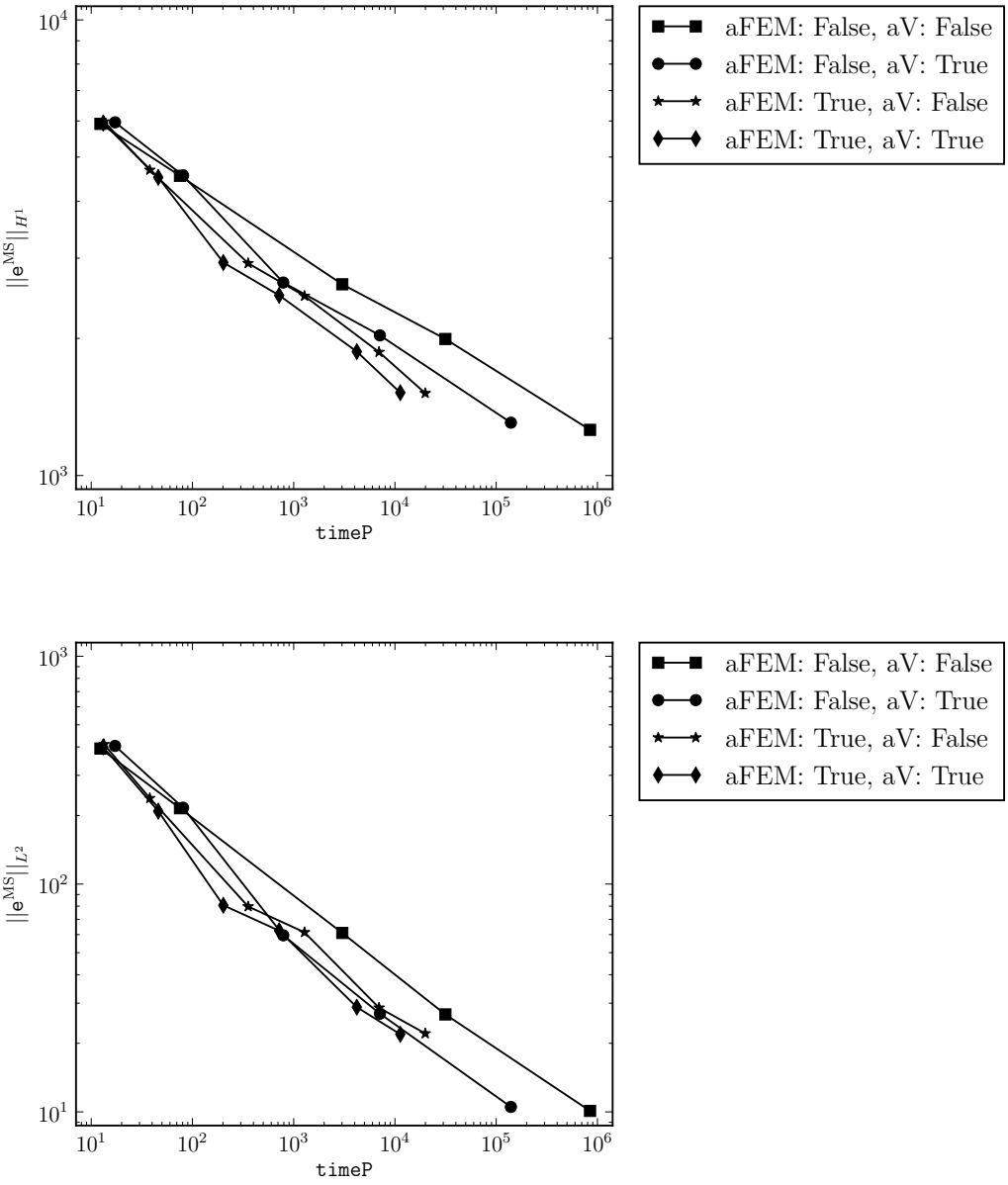


Figure 5.41: Experiment 5.4 Convergence with respect to the process time for different adaptive strategies in the H^1 norm (above) and the L^2 norm (below). The abbreviations are `aFEM` for finite element adaptivity and `aV` for the localized adaptive number of samples.

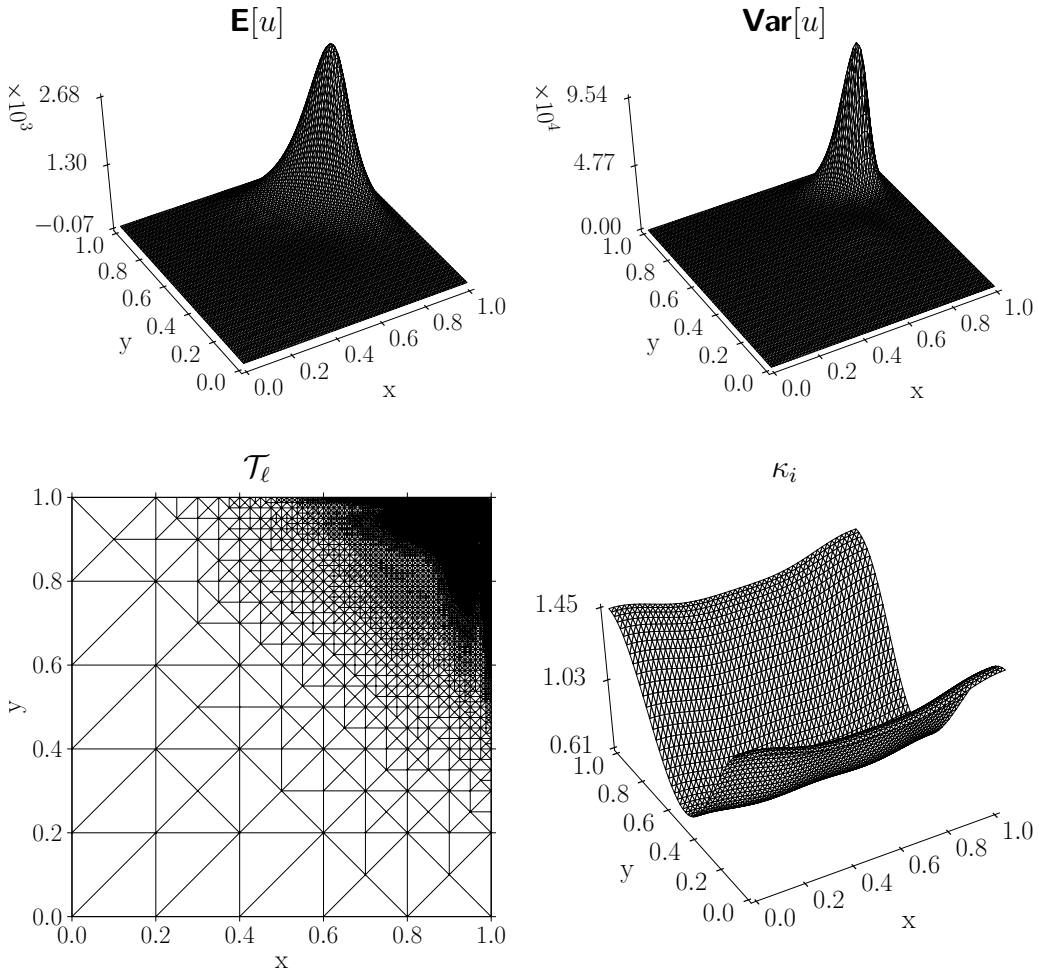


Figure 5.42: Experiment 5.4 The mean solution, its variance, an adaptive finite element mesh generated with the methods from Section 3.3 and a single realization κ_i of the random field κ.

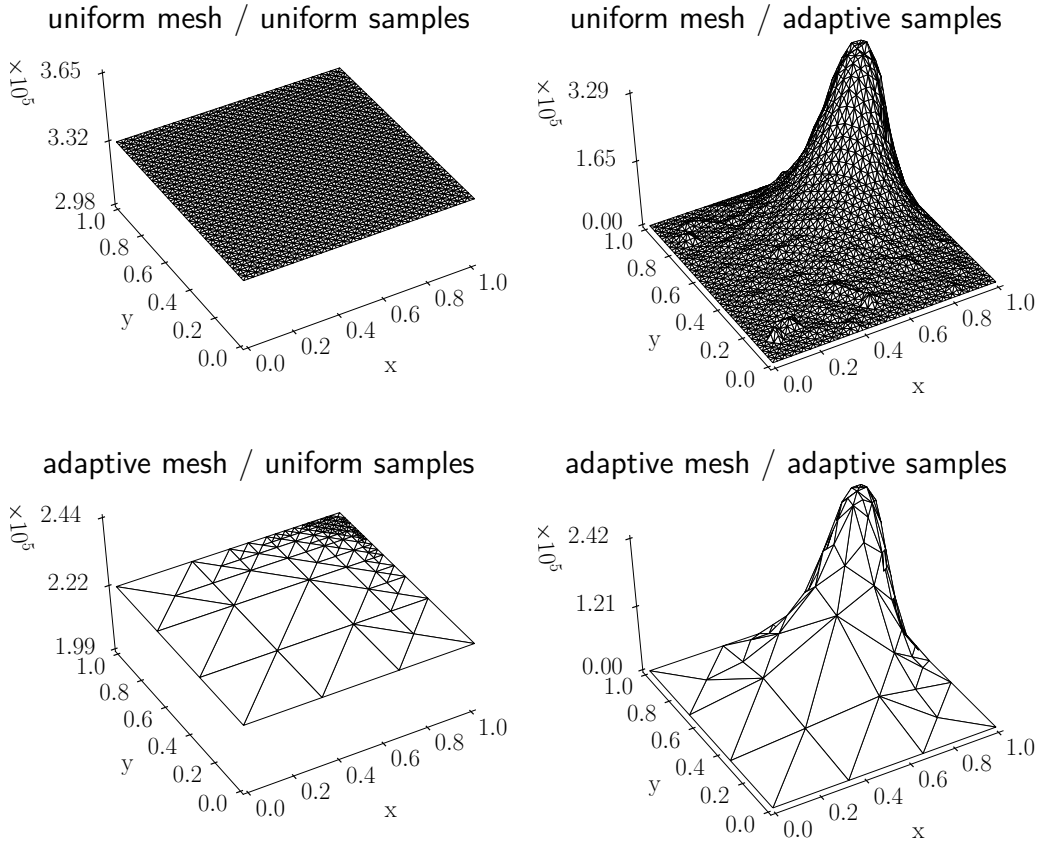


Figure 5.43: Experiment 5.4 The number of samples for each vertex interpolated onto the mesh. A better representation of the computational effort is given in Figure 5.44.

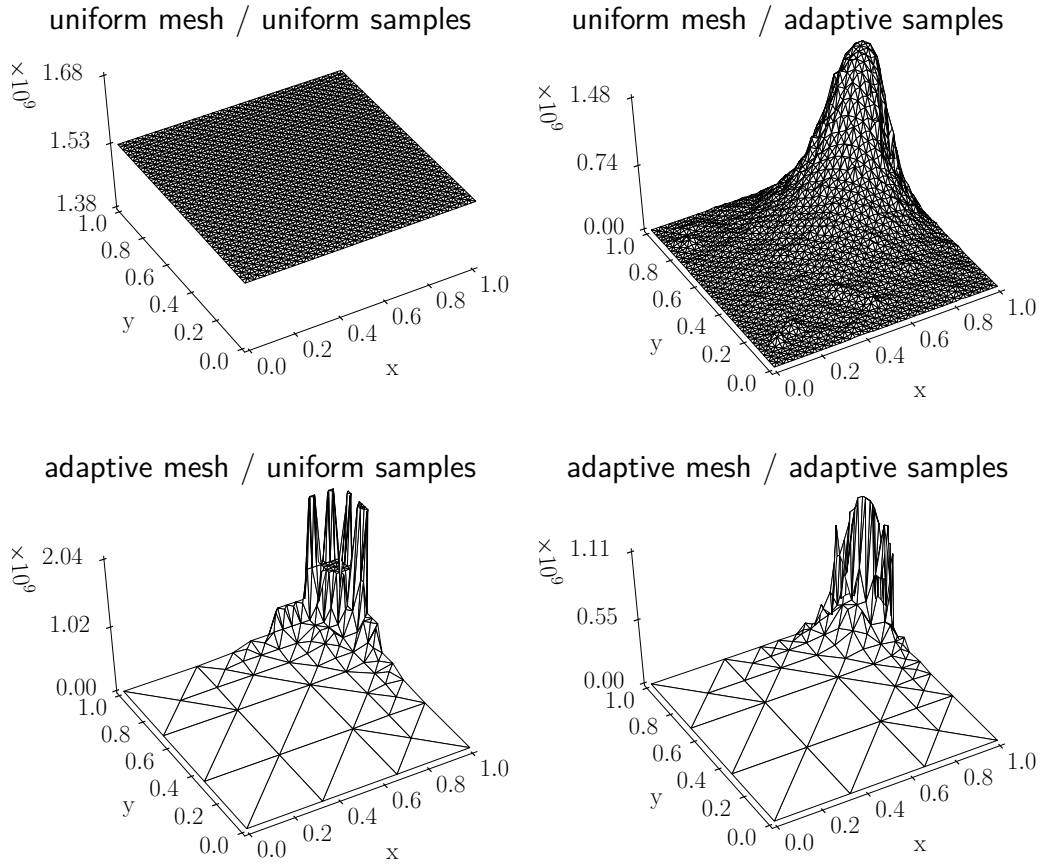


Figure 5.44: Experiment 5.4 The mean number of samples for each triangle divided by its area, defined as $\mathcal{C}_t$ in Equation (5.2), is shown for the different methods in order to visualize the computational effort and its distribution in the domain.

Experiment 5.5 – Smooth experiment

The random field κ is the same as in Experiment 5.4. In contrast to the last experiment, the rest of the input data will be chosen to be very smooth. The right-hand side is set to a constant $f \equiv 1$ and the homogeneous boundary data in Equation (1.1) is replaced with the Dirichlet condition

$$u_D = \sin(\pi x) + \sin(\pi y).$$

As can be seen in the mean solutions plot in Figure 5.46, spatial adaptivity is not needed for this experiment and thus it will create an unnecessary overhead compared to the uniform meshes. On the other hand, the variance, seen in the same figure, is not constant owing to the random field and the domain shape. Furthermore, an adaptive finite element mesh from the method in Section 3.3 as well as a realization of the random field κ is shown.

The distribution of the samples is shown in Figure 5.47. For all the methods, a rather homogeneous distribution is observed, except for the regions close to the boundary. The variance adaptive methods use much fewer samples in those areas. This is, however, compensated in the full adaptive method by the much smaller triangle diameters close to the boundary.

This effect becomes apparent in Figure 5.48 where the $\mathcal{C}_T$ from Equation (5.2) is depicted for this experiment. We can see, that quite a lot of mean samples per area are used in the full adaptive method despite a relatively low variance. This problem is much stronger with the solely spatial adaptive method as nearly all computational effort is concentrated in the low variance boundary regions.

The convergence plot in Figure 5.45 shows almost identical performance with respect to the H^1 norm for all four methods. This comes as no surprise as the described problem is very smooth. With respect to the L^2 norm, the spatial adaptive methods perform slightly worse than those with uniform meshes as the adaptivity optimizes for the H^1 norm. At the end of the preasymptotic phase, the spatial adaptive methods almost manage to catch up but some overhead factor of approximately 3 remains. Yet again, an L^2 norm error indicator might well improve their performance.

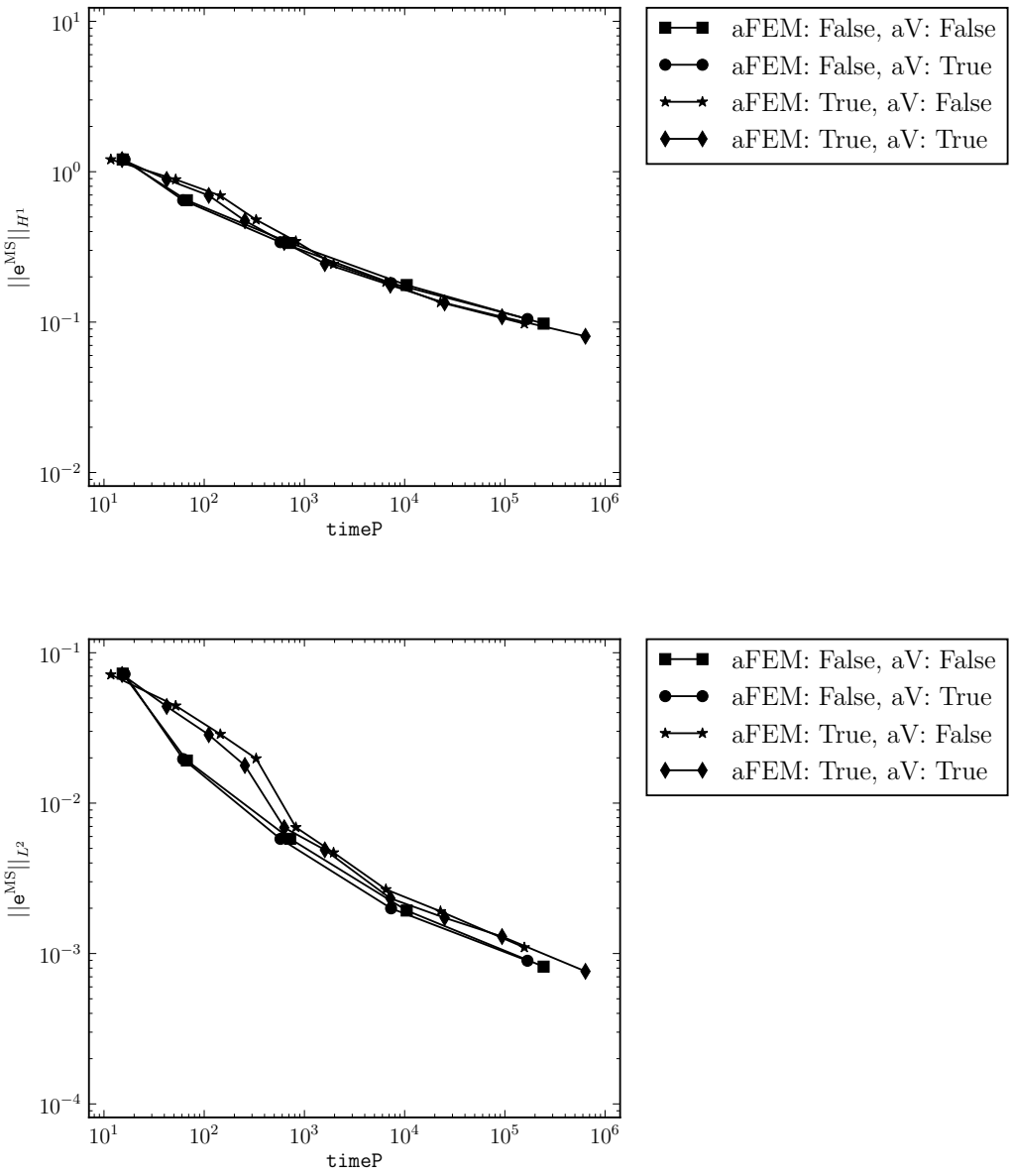


Figure 5.45: Experiment 5.5 Convergence with respect to the process time for different adaptive strategies in the H^1 norm (above) and the L^2 norm (below). The abbreviations are `aFEM` for finite element adaptivity and `aV` for the localized adaptive number of samples.

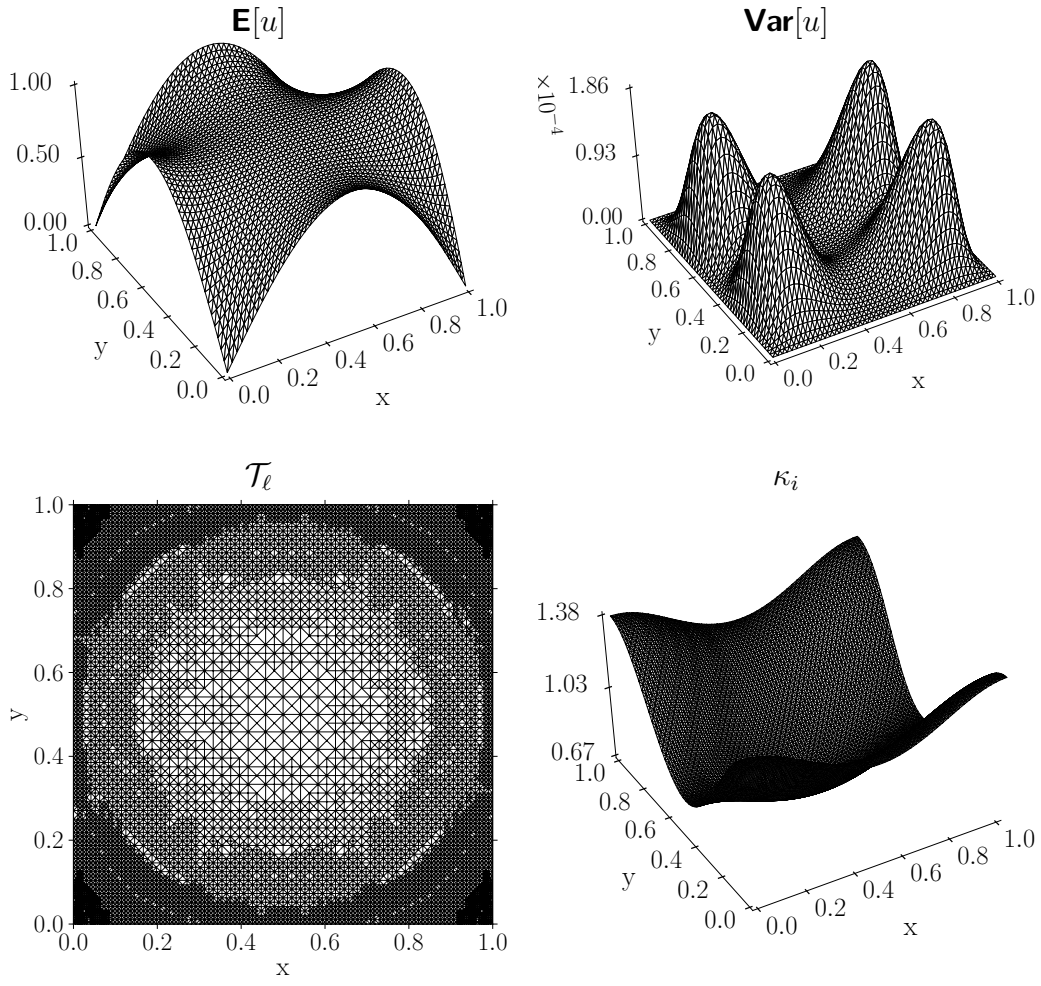


Figure 5.46: Experiment 5.5 The mean solution, its variance, an adaptive finite element mesh generated with the methods from Section 3.3 and a single realization κ_i of the random field κ.

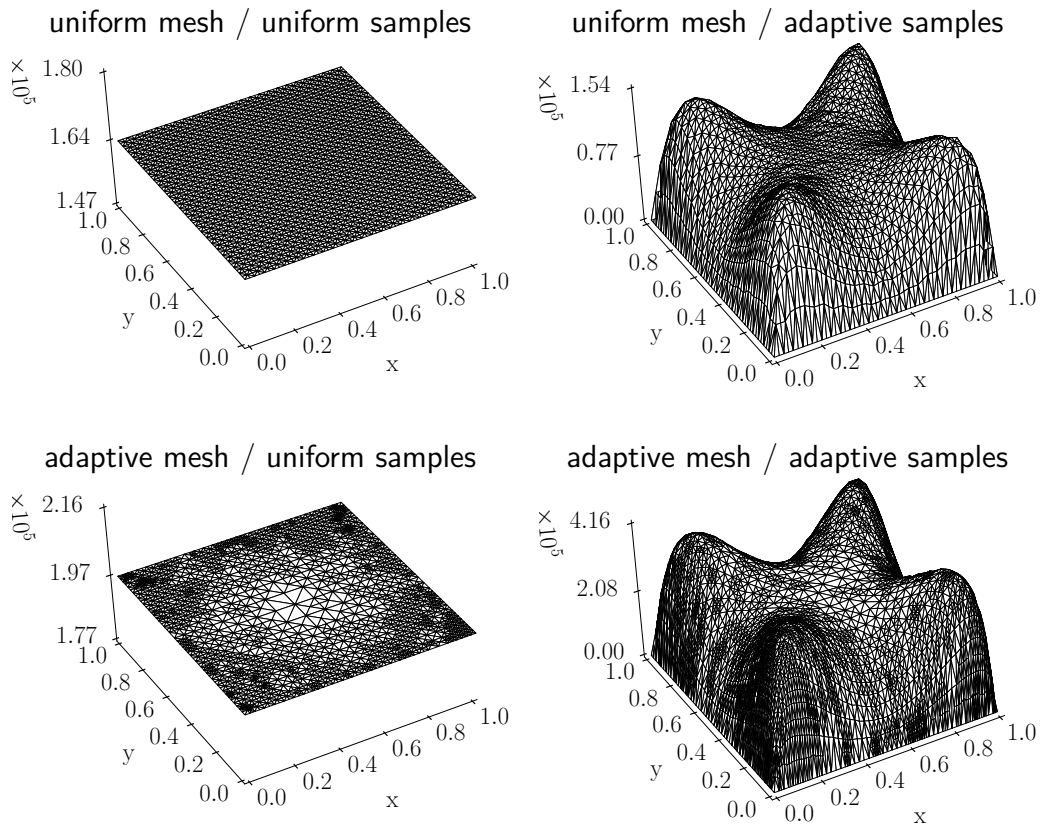


Figure 5.47: Experiment 5.5 The number of samples for each vertex interpolated onto the mesh. A better representation of the computational effort is given in Figure 5.48.

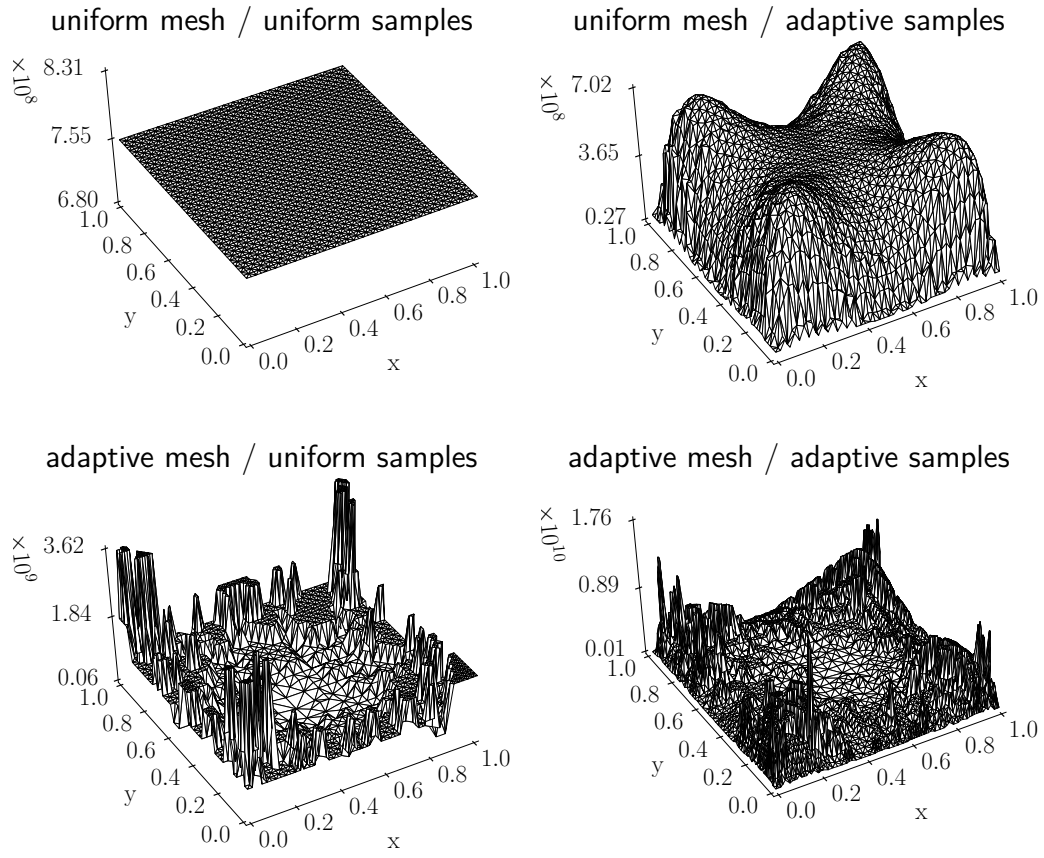


Figure 5.48: Experiment 5.5 The mean number of samples for each triangle divided by its area, defined as $\mathcal{C}_t$ in Equation (5.2), is shown for the different methods in order to visualize the computational effort and its distribution in the domain.

6 Conclusions

Overview

The biggest challenge in finding the solution for partial differential equations with random fields, by far, remains to be the size of the solution space $D \times \Omega$ as Ω usually has infinite dimensions. Also the discretization of D and Ω in some finite-dimensional subspaces still leaves a high-dimensional product space. It is futile to attempt the numerical treatment of real world examples without the exploitation of sparsity patterns in the product space. The aim for all methods must be to focus all computational efforts on those parts with the highest error density. For stochastic Galerkin methods, combined error estimators can be developed [EGSZ14] but those methods still struggle with the high-dimensionality such that even the construction of a sufficiently accurate Karhunen-Loève decomposition can become a hard challenge.

Sampling based methods do not suffer from this problem but instead have to battle a dimension independent slow rate of convergence. This rate is fundamental and cannot be improved by any means. Thus, it is important to find ways to reduce the variance of the random variable that needs to be sampled as well as to reduce the computational effort for each sample that has to be rendered.

Monte Carlo and Multilevel Monte Carlo

The adaptive multilevel Monte Carlo method presented in Chapter 2 addresses both problems. The multilevel method aims to reduce the variance as cheaply as possible through offloading the computation of as many of the samples as possible onto coarse and thus cheap meshes. On finer meshes only correcting random variables are sampled which have a much smaller variance and thus need fewer samples. The cost is reduced even further with adaptive meshes which give the same error but with less computational effort. The method presented here aims to provide adaptive meshes that are cheap to come by and yet offer enough error reduction. In fact, it requires that the mean adaptive meshes offer enough error reduction with respect to the overall error density in $D \times \Omega$.

One of the biggest advantage of the presented method is the ease of implementation, as existing finite element solvers and adjoint adaptive algorithms can readily be used in a black-box style in conjunction with the sampling approaches from Chapter 2. In strong contrast, Galerkin methods need to alter the solver itself with the expected challenges for complex problems. The adaptive meshes come at a very low cost

yet achieve great improvements in cost efficiency for a wide variety of problems. As shown in the numerical studies, the expected gain in speed can be gauged for suitable problems from deterministic situations already at hand. Compared to the other cost contributions, the additional operations in the multilevel algorithm are cheap and the cost improvements easily justify their application. The combination of both approaches can lead to a further reduction in the effort necessary to achieve a given error which leads to a staggering two orders of magnitude cheaper method compared to the uniform Monte Carlo approach.

The worst case scenario for the mean meshes is actually the uniform mesh, which will result in the same performance as the uniform multilevel Monte Carlo method with an additional overhead for the creation of the meshes. As the cost for their generation is dominated by the costs on one level with a small constant number of samples necessary to achieve reliable results, it can be ignored in almost any circumstances. These problems, however, might profit from path-wise adaptivity, that is a different adaptive mesh for every sample. This is much more expensive, because for every sample the adaptive meshes have to be computed so one needs to ensure that the effort for those samples is compensated by the advantage of the adaptive meshes. This situation arises for example with the Lévêque problem, where high velocities demand appropriate anisotropic adaptive meshes [Hol08]. In all the numerical examples in this work, however, the expected improvement as seen in the deterministic case could already be realized with the mean meshes so that no considerable improvement is expected for this class of problems.

SDE Method

The SDE-based interpolation method from Chapter 4 provides a valuable alternative to the finite element approach if combined with proper adaptivity. In case of some quantity of interest this method only considers the support of the quantity and disregards the rest of the domain. This is especially important for higher dimensional problems where finite element approaches suffer from big computational costs.

The exploratory numerical tests in this work already showed encouraging results regarding the adaptive methods. The residual error estimators optimized successfully the mesh structure for the reduction of the H^1 error. More important, however, was the adaptive variance method, which improved the error reliably under cost constraints in a problem-independent approach that is simple to implement.

A direct comparison is not reasonable in this context as the method is expected

to perform the best for higher dimensional spaces D. For the sake of a reliable implementation and verifiability through the Monte Carlo and multilevel Monte Carlo method from Chapter 2 we restrained the number of dimensions to two as in the first methods. This is enough as a proof of concept and to show the robustness of this method even for complex problems with strong oscillations in the input data and a very high variance in the solution.

Outlook

The mean meshes used in this thesis can adapt a great part of the advantages resulting from adaptive meshes for the stochastic context. However, individual adaptivity for each $\omega \in \Omega$ will result in improved error reduction with respect to the number of degrees of freedom. This comes at a higher cost which grows with the number of degrees of freedom and the number of samples as for each sample a sequence of adaptive meshes needs to be computed up to some error tolerance. A combination of the above techniques or a truncated adaptive sequence which uses uniform refinement after some threshold might mitigate this problem.

For the multilevel Monte Carlo method the adaptive meshes were chosen so that they grow with similar computational complexity as the uniform refinement. However, the uniform refinements themselves are not an optimal choice but instead some geometric sequence promises better results [HANvST15, CHAN$^+$14]. The same holds true for the adaptive meshes.

The techniques presented for the Monte Carlo and the multilevel Monte Carlo methods should be readily adaptable for stochastic collocation methods and Quasi-Monte Carlo methods as they start with similar premises. The results should closely resemble those in Chapter 5.

The alternative method showed promising results for these basic examples and further investigation with higher dimensions in the stochastic and physical domain as well as with quantities of interest needs to be done. Other error indicators that are better adapted for the computational goal need to be derived.

Heuristics remain an important part of sampling based methods as non smooth problems with high oscillations in the input data make it hard to extrapolate data in a reliable manner. On the other hand it is common that suboptimal choices of key parameters will lead to a great reduction in the computational efficiencies.

Bibliography

[ABE+15a] Felix Anker, Christian Bayer, Martin Eigel, Marcel Ladkau, Johannes Neumann, and John G.M. Schoenmakers. SDE based regression for random PDEs. *WIAS Preprint*, 2192, 2015.

[ABE+15b] Felix Anker, Christian Bayer, Martin Eigel, Johannes Neumann, and John G.M. Schoenmakers. Adaptive SDE based interpolation for random PDEs. *WIAS Preprint*, 2200, 2015.

[ABH+15] Martin Sandve Alnæs, Jan Blechta, Johan Hake, August Johansson, Benjamin Kehlet, Anders Logg, Chris Richardson, Johannes Ring, Marie Elisabeth Rognes, and Garth N. Wells. The FEniCS Project – Version 1.5. *Archive of Numerical Software*, 3(100), 2015.

[Ain08] Mark Ainsworth. A posteriori error estimation for lowest order Raviart-Thomas mixed finite elements. *SIAM Journal on Scientific Computing*, 30(1):189–204, 2007/08.

[ALM+09] Martin Sandve Alnæs, Anders Logg, Kent-Andre Mardal, Ola Skavhaug, and Hans Petter Langtangen. Unified framework for finite element assembly. *International Journal of Computational Science and Engineering*, 4(4):231–244, 2009.

[ALM12] Martin Sandve Alnæs, Anders Logg, and Kent-Andre Mardal. *UFC: a Finite Element Code Generation Interface*, chapter 16. In Logg et al. [LMW+12], 2012.

[Aln12] Martin Sandve Alnæs. *UFL: a Finite Element Form Language*, chapter 17. In Logg et al. [LMW+12], 2012.

[ALO+14] Martin Sandve Alnæs, Anders Logg, Kristian B. Ølgaard, Marie Elisabeth Rognes, and Garth N. Wells. Unified Form Language: A domain-specific language for weak formulations of partial differential equations. *ACM Transactions on Mathematical Software*, 40(2), 2014.

[AO97] Mark Ainsworth and J. Tinsley Oden. A posteriori error estimation in finite element analysis. *Computer Methods in Applied Mechanics and Engineering*, 142(1):1–88, 1997.

[AO00] Mark Ainsworth and J. Tinsley Oden. *A Posteriori Error Estimation in Finite Element Analysis*. Pure and Applied Mathematics (New York). Wiley-Interscience [John Wiley & Sons], New York, 2000.

[Bab71] Ivo Babuška. Error-bounds for finite eelement method. *Numerische Mathematik*, 16(4):322–333, 1971.

[BC05] Corinna Bahriawati and Carsten Carstensen. Three MATLAB implementations of the lowest-order Raviart-Thomas MFEM with a posteriori error control. *Computational Methods in Applied Mathematics Comput. Methods Appl. Math.*, 5(4):333–361, 2005.

[BCD04] Sören Bartels, Carsten Carstensen, and Georg Dolzmann. Inhomogeneous Dirichlet conditions in a priori and a posteriori finite element error analysis. *Numerische Mathematik*, 99(1):1–24, 2004.

[Beb03] Mario Bebendorf. A note on the Poincaré inequality for convex domains. *Zeitschrift für Analysis und ihre Anwendungen. Journal for Analysis and its Applications*, 22(4):751–756, 2003.

[BM84a] Ivo Babuška and Anthony D. Miller. The post-processing approach in the finite element method – part 1. calculation of displacements, stresses, and other higher derivatives of the displacements. *International journal for numerical methods in engineering*, 20(6):1085–1109, 1984.

[BM84b] Ivo Babuška and Anthony D. Miller. The post-processing approach in the finite element method – part 2: The calculation of stress intensity factors. *International Journal for Numerical Methods in Engineering*, 20(6):1111–1129, 1984.

[BM84c] Ivo Babuška and Anthony D. Miller. The post-processing approach in the finite element method – part 3: A posteriori error estimates and adaptive mesh selection. *International Journal for Numerical Methods in Engineering*, 20(12):2311–2324, 1984.

[BNT07] Ivo Babuška, Fabio Nobile, and Raúl Tempone. A stochastic collocation method for elliptic partial differential equations with random input data. *SIAM Journal on Numerical Analysis*, 45(3):1005–1034, 2007.

[BR01] Roland Becker and Rolf Rannacher. An optimal control approach to a posteriori error estimation in finite element methods. *Acta Numerica*, 10:1–102, 2001.

[BR03] Wolfgang Bangerth and Rolf Rannacher. *Adaptive finite element methods for differential equations*. Birkhäuser, 2003.

[Bra07] Dietrich Braess. *Finite Elements*. Cambridge University Press, Cambridge, 3rd edition, 2007. Theory, Fast Solvers, and Applications in Elasticity Theory, Translated from the German by Larry L. Schumaker.

[BTZ04] Ivo Babuška, Raúl Tempone, and Georgios E. Zouraris. Galerkin finite element approximations of stochastic elliptic partial differential equations. *SIAM Journal on Numerical Analysis*, 42(2):800–825, 2004.

[CGST11] K. Andrew Cliffe, Mike B. Giles, Robert Scheichl, and Aretha L. Teckentrup. Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients. *Computing and Visualization in Science*, 14(1):3–15, 2011.

[Cha99] François Chaplais. The Strang and Fix conditions, 1999. available at `https://www.researchgate.net/publication/237716880_The_Strang_and_Fix_Conditions`.

[CHAN+14] Nathan Collier, Abdul-Lateef Haji-Ali, Fabio Nobile, Erik von Schwerin, and Raúl Tempone. A continuation multilevel monte carlo algorithm. *BIT Numerical Mathematics*, 55(2):399–432, 2014.

[CMN13] Carsten Carstensen, Christian Merdon, and Johannes Neumann. Aspects of guaranteed error control in CPDEs. In *Numerical Solution of Partial Differential Equations: Theory, Algorithms, and Their Applications*, volume 45 of *Springer Proceedings in Mathematics & Statistics*, pages 103–119. Springer New York, 2013.

[CST13] Julia Charrier, Robert Scheichl, and Aretha L. Teckentrup. Finite element error analysis of elliptic PDEs with random coefficients and its application to multilevel Monte Carlo methods. *SIAM Journal on Numerical Analysis*, 51(1):322–352, 2013.

[D+16] Timothy A. Davis et al. Suitesparse, 2006-2016. software available at `http://faculty.cse.tamu.edu/davis/suitesparse.html`.

[Dav04] Timothy A. Davis. Algorithm 832: Umfpack v4.3 — an unsymmetric-pattern multifrontal method. *ACM Transactions on Mathematical Software (TOMS)*, 30(2):196–199, 2004.

[Dav06] Timothy A. Davis. *Direct methods for sparse linear systems*, volume 2. Siam, 2006.

[DKV06] Wolfgang Dahmen, Angela Kunoth, and Jürgen Vorloeper. Convergence of adaptive wavelet methods for goal-oriented error estimation. In *Numerical Mathematics and Advanced Applications*, pages 39–61. Springer, Berlin, 2006.

[DN97] Claude R. Dietrich and Garry Neil Newsam. Fast and exact simulation of stationary gaussian processes through circulant embedding of the covariance matrix. *SIAM Journal on Scientific Computing*, 18(4):1088–1107, 1997.

[DSB11] Xuan-Hung Dang, Bruno Sudret, and Marc Berveiller. Benchmark of random fields simulation methods and links with identification methods. *Applications of Statistics and Probability in Civil Engineering*, page 384, 2011.

[Dö96] Willy Dörfler. A convergent adaptive algorithm for Poisson's equation. *SIAM Journal on Numerical Analysis*, 33(3):1106–1124, 1996.

[EGSZ14] Martin Eigel, Claude Jeffrey Gittelson, Christoph Schwab, and Elmar Zander. Adaptive stochastic Galerkin FEM. *Computer Methods in Applied Mechanics and Engineering*, 270:247–269, 2014.

[EMN16] Martin Eigel, Christian Merdon, and Johannes Neumann. An adaptive multilevel monte carlo method with stochastic bounds for quantities of interest with uncertain data. *SIAM/ASA Journal on Uncertainty Quantification*, 4(1):1219–1245, 2016.

[Eva10] Lawrence Craig Evans. *Partial Differential Equations*, volume 19 of *Graduate Studies in Mathematics*. American Mathematical Society, 2nd edition, 2010.

[Fel68] William Feller. *An Introduction to Probability Theory and its Application*. Wiley, 3rd edition, 1968.

[fen03] The FEniCS project. `http://fenicsproject.org/`, 2003.

[Gal11] Giovanni Paolo Galdi. *An introduction to the mathematical theory of the Navier-Stokes equations: Steady-state problems*. Springer Science & Business Media, 2011.

[Gil08] Michael B. Giles. Multilevel Monte Carlo path simulation. *Operations Research*, 56(3):607–617, 2008.

[GKSS13] Claude Jeffrey Gittelson, Juho Könnö, Christoph Schwab, and Rolf Stenberg. The multi-level Monte Carlo finite element method for a stochastic Brinkman Problem. *Numerische Mathematik*, 125(2):347–386, 2013.

[GS90] Roger Ghanem and Pol Dimitrios Spanos. Polynomial chaos in stochastic finite elements. *Journal of Applied Mechanics*, 57(1):197–202, 1990.

[GS91] Roger Ghanem and Pol Dimitrios Spanos. *Stochastic Finite Elements: A Spectral Approach*. Springer, 1991.

[GS93] Roger Ghanem and Pol Dimitrios Spanos. A stochastic Galerkin expansion for nonlinear random vibration analysis. *Probabilistic Engineering Mechanics*, 8(3):255–264, 1993.

[H+16] Jim Hugunin et al. SciPy, 1995-2016. `http://www.scipy.org/`.

[HANvST15] Abdul-Lateef Haji-Ali, Fabio Nobile, Erik von Schwerin, and Raúl Tempone. Optimization of mesh hierarchies in multilevel monte carlo samplers. *arXiv preprint arXiv:1403.2480v2*, 2015.

[Hol08] Ekkehard Holzbecher. Numerical solutions for the Lévêque Problem of boundary layer mass or heat flux. *Proceedings of the COMSOL Conference*, 2008.

[Hun07] John D. Hunter. Matplotlib: A 2D graphics environment. *Computing in science and engineering*, 9(3):90–95, 2007. software available at `http://matplotlib.org/`.

[Joh12] Claes Johnson. *Numerical solution of partial differential equations by the finite element method*. Courier Corporation, 2012.

[Kac49] Mark Kac. On distributions of certain Wiener functionals. *Transactions of the American Mathematical Society*, 65(1):1–13, 1949.

[Kal02] Olav Kallenberg. *Foundations of Modern Probability*. Springer, New York, 2nd edition, 2002.

[Kar47] Kari Karhunen. *Über lineare Methoden in der Wahrscheinlichkeitsrechnung*. PhD thesis, Suomalaisen Tiedeakatemian, 1947.

[Kee04] Andreas Keese. *Numerical Solutions of Systems with Stochastic Uncertainties: A General Purpose Framework for Stochastic Finite Elements*. PhD thesis, Technische Universität Braunschweig, 2004.

[KL06] Robert C. Kirby and Anders Logg. A compiler for variational forms. *ACM Transactions on Mathematical Software*, 32(3), 2006.

[Kra88] Dieter Kraft. A software package for sequential quadratic programming. *Wiss. Berichtswesen des Deutsche Forschungs- und Versuchsanstalt für Luft- und Raumfahrt e.V. (DFVLR): Forschungsbericht*, 88(28), 1988.

[LK10] Olivier P. Le Maître and Omar M. Knio. *Introduction: Uncertainty Quantification and Propagation*. Springer, 2010.

[LMW+12] Anders Logg, Kent-Andre Mardal, Garth N. Wells, et al. *Automated Solution of Differential Equations by the Finite Element Method*. Springer, 2012.

[Loè78] Michel Loève. Probability theory I/II. In *Graduate Texts in Mathematics*, volume 45/46. Springer-Verlag New York, 1977/1978.

[LORW12] Anders Logg, Kristian B. Ølgaard, Marie Elisabeth Rognes, and Garth N. Wells. *FFC: the FEniCS Form Compiler*, chapter 11. In Logg et al. [LMW+12], 2012.

[LS10] Richard S. Laugesen and Bartłomiej Andrzej Siudeja. Minimizing Neumann fundamental tones of triangles: An optimal Poincaré inequality. *Journal of Differential Equations*, 249(1):118–135, 2010.

[Mer13] Christian Merdon. *Aspects of Guaranteed Error Control in Computations for Partial Differential Equations*. PhD thesis, Humboldt-Universität zu Berlin, 2013.

[MK05] Hermann G. Matthies and Andreas Keese. Galerkin methods for linear and nonlinear elliptic stochastic partial differential equations. *Computer Methods in Applied Mechanics and Engineering*, 194(12):1295–1331, 2005.

[MS09] Mario S. Mommer and Rob Stevenson. A goal-oriented adaptive finite element method with convergence rates. *SIAM Journal on Numerical Analysis*, 47(2):861–886, 2009.

[MT04] Grigori Noah Milstein and Michael V. Tretyakov. *Stochastic Numerics for Mathematical Physics*. Scientific Computation. Springer Berlin Heidelberg, 1st edition, 2004.

[OP01] John Tinsley Oden and Serge Prudhomme. Goal-oriented error estimation and adaptivity for the finite element method. *Computers & mathematics with applications*, 41(5):735–756, 2001.

[OW10] Kristian B. Ølgaard and Garth N. Wells. Optimisations for quadrature representations of finite element tensors through automated code generation. *ACM Transactions on Mathematical Software*, 37, 2010.

[PW60] Lawrence E. Payne and Hans F. Weinberger. An optimal Poincaré inequality for convex domains. *Archive for Rational Mechanics and Analysis*, 5(1):286–292, 1960.

[SF71] Gilbert Strang and George Fix. A fourier analysis of the finite element variational method. In *Constructive aspects of functional analysis*, pages 796–830. Springer, 1971.

[Tec13] Aretha L. Teckentrup. *Multilevel Monte Carlo Methods and Uncertainty Quantication*. PhD thesis, University of Bath, 2013.

[TSGU13] Aretha L. Teckentrup, Robert Scheichl, Mike B. Giles, and Elisabeth Ullmann. Further analysis of multilevel Monte Carlo methods for elliptic PDEs with random coefficients. *Numerische Mathematik*, 125(3):569–600, 2013.

[V+16] Gael Varoquaux et al. Joblib: running python functions as pipeline jobs, 2010-2016. `https://pythonhosted.org/joblib/`.

[XH05] Dongbin Xiu and Jan S. Hesthaven. High-order collocation methods for differential equations with random inputs. *SIAM Journal on Scientific Computing*, 27(3):1118–1139, 2005.

[XK02] Dongbin Xiu and George Em Karniadakis. The Wiener–Askey polynomial chaos for stochastic differential equations. *SIAM journal on scientific computing*, 24(2):619–644, 2002.

[XK03] Dongbin Xiu and George Em Karniadakis. Modeling uncertainty in flow simulations via generalized polynomial chaos. *Journal of computational physics*, 187(1):137–167, 2003.

List of Figures

List of Pictures

Zusammenfassung

Das Stichprobenverfahren, auch bekannt als Monte-Carlo-Methode, ist ein robuster Ansatz zum Lösen von partiellen Differentialgleichungen mit stochastischen Koeffizienten, welcher sich gut für bereits existierende Finite-Elemente- oder Finite-Volumen-Implementationen der entsprechenden deterministischen Differentialgleichung adaptieren lässt. Der Hauptvorteil dieses Verfahrens liegt in der dimensionsunabhängigen Konvergenzrate von $1/2$ bezüglich der verwendeten Stichproben, welche sich aus dem zentralen Grenzwertsatz ergibt. Dies ist sogleich der größte Schwachpunkt. Daher ist es ein zentrales Anliegen aktueller Ansätze die Berechnungskosten der einzelnen Stichproben zu reduzieren. Das Multilevel-Monte-Carlo-Verfahren bietet hierfür einen der vielversprechendsten Ansätze.

Diese Arbeit hat drei Ziele. Zum ersten sollen die Kosten für die Stichproben sowohl im Monte-Carlo-Verfahren als auch im Multilevel-Monte-Carlo-Verfahren weiter durch den Einsatz von adaptiven Gitterhierarchien verbessert werden. Dabei werden Lösungsstrategien für die auftretenden Probleme beim Ermitteln von optimalen Parametern für das Verfahren vorgeschlagen.

Eine fundamentale Eigenschaft des Stichprobenverfahrens ist die Unsicherheit des numerischen Ergebnisses, da es sich wiederum um eine Zufallsgröße handelt. Daher ist das zweite Ziel dieser Arbeit das Finden von Schranken, welche mit einer gewählten Wahrscheinlichkeit den numerischen Fehler beschränken um somit verlässliche Aussagen über die zu berechnende Größe treffen zu können.

Zuletzt soll eine alternative Methode untersucht werden, welche die Dimensionsabhängigkeit von der Finite-Elemente-Diskretisierung des physikalischen Gebietes auflöst, indem sie hier ebenfalls eine Zufallsvariable zur Modellierung mithilfe einer stochastischen gewöhnlichen Differentialgleichung verwendet. Adaptive Methoden werden auch hier die Stichprobenkosten reduzieren.

Numerische Tests belegen die Verbesserungen der vorgeschlagen Methoden.

Lebenslauf

Der Lebenslauf ist in der Online-Version aus Gründen des Datenschutzes nicht enthalten.