Robust multi-level Monte Carlo finite volume methods for systems of hyperbolic conservation laws with random input data

Author(s):
Šukys, Jonas

Publication Date:
2014

Permanent Link:
https://doi.org/10.3929/ethz-a-010163877

Rights / License:
In Copyright - Non-Commercial Use Permitted

This page was generated automatically upon download from the ETH Zurich Research Collection. For more information please consult the Terms of use.
Robust multi-level Monte Carlo
Finite Volume methods for
systems of hyperbolic conservation laws
with random input data

A dissertation submitted to
ETH Zürich

for the degree of
Doctor of Sciences

presented by
JONAS ŠUKYS
B.Sc. Math. Jacobs University Bremen
M.Sc. Math. ETH Zürich
born April 19, 1987
citizen of Lithuania

accepted on the recommendation of
Prof. Dr. Siddhartha Mishra, ETH Zürich, examiner
Prof. Dr. Christoph Schwab, ETH Zürich, co-examiner
Prof. Dr. George E. Karniadakis, Brown University, co-examiner
Prof. Dr. Anders Szepessy, Royal Institute of Technology (KTH), co-examiner

2014
Šukys, Jonas
Robust multi-level Monte Carlo Finite Volume methods
for systems of hyperbolic conservation laws with random input data
Diss., ETH Zürich, Nr. 21990
2014
DOI: http://doi.org/10.3929/ethz-a-010163877
ISBN: 978-3-906031-60-6
Acknowledgments

During my work as a PhD student and in the preparation of this thesis, I was constantly supported by several colleagues. In particular, I am grateful to Siddhartha Mishra and Christoph Schwab for their thorough supervision and for teaching me almost everything I know in this field, to George Karniadakis and Anders Szepessy for being my co-examiners, and to Peter Arbenz, Patrick Jenny, and Wesley Petersen for additional guidance. I am also thankful to Andrea Barth, Manuel J. Castro, Robert Gantner (also for proof-reading this thesis), Carlos S. Linares, Daniel Meyer, Florian Müller, Stefan Pauli (also for helping with the Zusammenfassung), and Svetlana Tokareva for productive discussions, to CMA institute in University of Oslo at the Norway for sharing the source code of ASLVID, and to the systems support groups at Swiss National Supercomputing Center and of ETH Zürich parallel compute cluster Brutus for their extensive support.

This work was partly funded by the ETH interdisciplinary research grant CH1-03 10-1 and by the Swiss National Supercomputing Center production project grant ID S366.

I would also like to thank Vilma for her patience and support throughout my endeavors, and for alternative names she has given to some of the plots in this thesis, such as “a butterfly”, “a mushroom”, “an eye”, and “a cabbage”.


Abstract

A mathematical formulation of conservation and of balance laws with random input data is reviewed, specifically with random initial conditions, source terms, flux functions and coefficients. The concept of random entropy solution is specified and its statistical moments such as the mean and the variance are presented. For scalar conservation laws in multi-dimensions, recent results by Mishra and Schwab on the existence and uniqueness of random entropy solutions and their statistical moments are presented; these results are then extended to linear hyperbolic systems of conservation laws. The combination of Monte Carlo sampling with Finite Volume Method discretization in space and time for the numerical approximation of the statistics of random entropy solutions is discussed.

Asymptotic mean square error estimates for combined Monte Carlo Finite Volume Method (MC-FVM) are obtained for scalar and linear hyperbolic systems of conservation laws with random inputs. Following work of Mishra and Schwab, an improved multi-level version of the Monte Carlo Finite Volume Method (MLMC-FVM) is proposed and asymptotic mean square error bounds are presented.

Asymptotic mean square error versus expected work estimates, derived using novel probabilistic computational complexity analysis taking into account the sample path dependent complexity of the underlying FVM solve, indicate superiority of MLMC-FVM over plain MC-FVM, under comparable assumptions on the random input data. In particular, it is shown that approximations of statistical moments converge essentially at the same rate as a single solve of a deterministic problem using FVM. Extensions of the proposed algorithms to nonlinear hyperbolic systems of balance laws are outlined.

Implementation aspects of MLMC-FVM are discussed, including large scale random number generation, numerically stable and computationally efficient statistical estimators, bias-free multi-resolution discretizations of random source terms and coefficients, and novel static/adaptive load balancing techniques for scalability of MLMC-FVM on emerging massively parallel computing platforms.

A new code ALSVID-UQ implementing the MLMC-FVM is described and applied to simulate uncertain solutions of the Euler equations of gas dynamics and ideal magnetohydrodynamics equations of plasma physics with uncertain initial data and equation of state, shallow water equations of oceanography with uncertain bottom topography, acoustic wave equation with log-normally distributed heterogeneous coefficients, and Buckley-Leverett equations of two phase flows with uncertain fluxes. Numerical experiments in one, two and three dimensions with a very large number of uncertainty sources.
are presented, illustrating the theoretical results and verifying the robustness, efficiency
and scalability up to 40 000 cores of the ALSVID-UQ implementation.
Zusammenfassung


Ein neuer MLMC-FVM Code ALSVID-UQ wird beschrieben und angewandt zur Berechnung von statistischen Lösungen der Euler-Gleichungen der Gasdynamik und der idealen
# Contents

## 1 Introduction

1.1 Hyperbolic systems of conservation laws ............................................. 1

1.2 Examples ................................................................................................... 2

1.2.1 Burgers' equation .................................................................................. 2

1.2.2 Buckley-Leverett equation ..................................................................... 3

1.2.3 Acoustic wave equation ......................................................................... 3

1.2.4 Euler equations ....................................................................................... 4

1.2.5 Equations of magnetohydrodynamics ...................................................... 4

1.2.6 Shallow water equations ........................................................................ 5

1.3 General concepts ....................................................................................... 5

1.3.1 Hyperbolicity and nonlinearity ............................................................... 6

1.3.2 Weak solutions and Rankine-Hugoniot condition .................................. 9

1.3.3 Entropy solutions .................................................................................. 10

1.3.4 Existence results ................................................................................... 13

1.4 Finite Volume method ............................................................................. 17

1.4.1 Formulation ........................................................................................... 17

1.4.2 Two-wave approximate Riemann solvers .............................................. 20

1.4.3 Three-wave approximate Riemann solvers .......................................... 22

1.4.4 General framework for high order schemes ......................................... 24

1.4.5 Piecewise linear TVD reconstruction .................................................. 25

1.4.6 Essentially non-oscillatory reconstruction ............................................ 26

1.4.7 Time integration .................................................................................... 33

1.4.8 Well-balanced solvers .......................................................................... 34

1.4.9 Note on boundary conditions ................................................................. 35

1.4.10 Convergence theory for FVM schemes ............................................... 36

1.4.11 Estimates for the computational work of the FVM solver ..................... 40

1.5 Uncertainty ............................................................................................... 41

1.5.1 Uncertainties in input data .................................................................... 41

1.5.2 Probabilistic modeling of uncertainty ................................................... 43

1.6 Random conservation laws ..................................................................... 46

1.6.1 Definitions ............................................................................................ 47

1.6.2 Challenges and methods for uncertainty quantification ....................... 48

1.7 Outline of the dissertation ....................................................................... 50

## 2 Theory for random conservation laws

2.1 Scalar conservation laws with uncertain initial data ............................... 55
Contents

2.1 Random initial data .................................................. 55
2.1.1 Random initial data ............................................... 55
2.1.2 Existence and uniqueness result ................................ 56
2.2 Scalar conservation laws with uncertain flux function ......... 57
2.2.1 Bounded random flux .............................................. 58
2.2.2 Karhunen-Loève expansion of bounded random flux ....... 59
2.2.3 Existence and uniqueness result ................................ 60
2.3 Linear hyperbolic systems with uncertain coefficients ........ 61
2.4 Systems of nonlinear conservation laws ......................... 63

3 Monte Carlo Finite Volume method ................................. 65
3.1 Monte Carlo method .................................................. 65
3.1.1 MC convergence for scalar conservation laws ............... 66
3.1.2 MC convergence for linear hyperbolic systems ............... 67
3.2 Combination of Monte Carlo and Finite Volume Method ....... 69
3.2.1 Assumptions for a Finite Volume scheme ..................... 69
3.2.2 Definition of the MC-FVM scheme ............................. 70
3.3 Convergence analysis of MC-FVM .................................. 72
3.3.1 Error estimates for scalar conservation laws ............... 72
3.3.2 Error estimates for linear hyperbolic systems ............... 73
3.3.3 Number of samples .............................................. 74
3.3.4 Error vs. work estimates ....................................... 75
3.4 Discussion ............................................................ 75

4 Multi-Level Monte Carlo Finite Volume method ................. 79
4.1 Derivation of MLMC-FVM ........................................... 79
4.1.1 Hierarchical nested spatio-temporal discretization ......... 79
4.1.2 Derivation of MLMC-FVM ....................................... 80
4.2 Convergence analysis of MLMC-FVM ............................. 82
4.2.1 Error estimates for scalar conservation laws ............... 82
4.2.2 Error estimates for linear hyperbolic systems ............... 84
4.2.3 Number of samples and error vs. work estimates .......... 86
4.2.4 Sparse tensor MC-FVM and MLMC-FVM ...................... 96

5 Parallel implementation of MLCM-FVM ......................... 99
5.1 Hierarchy of nested meshes ....................................... 99
5.1.1 FVM solver .................................................... 99
5.1.2 Robust pseudo random number generators and seeding.... 100
5.1.3 Stable and efficient statistical estimators ................... 102
5.2 Parallelization with MPI using intra-communicators .......... 103
5.3 Static load balancing .............................................. 107
5.4 Strong and weak scaling ......................................... 109
5.5 Limits of static load balancing .................................. 111
5.6 Adaptive load balancing .......................................... 112
5.6.1 Theory ....................................................... 112
Contents

10.2.2 Normally distributed random coefficients .......................... 176
10.3 Log-normally distributed random material coefficients ............. 179
  10.3.1 Spectral generator using Fast Fourier Transform ................. 180
  10.3.2 Covariance upscaling .............................................. 182
  10.3.3 Exponential covariance kernel for periodic domains ............. 183
10.4 Acoustic wave propagation in random heterogeneous medium .......... 185
  10.4.1 Two dimensional case .............................................. 185
  10.4.2 Two dimensional case with discontinuous layers ................. 189
  10.4.3 Three dimensional case ............................................ 191

11 MLMC-FVM for nonlinear systems with random fluxes ................. 199
  11.1 Euler equations with uncertain equation of state ................. 199
    11.1.1 Uniformly perturbed specific heats in 1-D: Sod shock tube .... 199
    11.1.2 Uniformly perturbed specific heats in 2-D: cloud-shock ....... 200

12 Summary and conclusions .................................................. 203

References ................................................................. 207

List of Publications ....................................................... 217
1 Introduction

A number of problems in physics and engineering are modeled in terms of nonlinear partial differential equations, also termed hyperbolic systems of balance laws. In this chapter, we introduce such systems, provide several relevant examples, and recapitulate well-established theoretical results as well as robust numerical discretization techniques, mainly by following the book of LeVeque [66] and further complimentary well established literature [20, 23, 24, 30, 32, 38, 45, 46, 50, 53, 97, 107] and others. After establishing these preliminaries, we outline potential sources of uncertainty in inputs, leading to random balance laws, which we formulate using the probabilistic framework of Kolmogorov. Finally, we review existing techniques for uncertainty quantification in such random systems, and outline novel theoretical results and numerical discretization schemes, which will be presented and analyzed in this thesis.

1.1 Hyperbolic systems of conservation laws

A system of \( m \) balance laws with \( m \geq 1 \) on a \( d \)-dimensional (\( d = 1, 2, 3 \)) Lipschitz domain \( D \subseteq \mathbb{R}^d \) and time domain \( \mathbb{R}_+ = \{ t \in \mathbb{R} : t \geq 0 \} \) is given by [66]

\[
\begin{aligned}
U_t(x,t) &+ \text{div}(F(c(x,t), U)) = S(x,t,U), \\
L_b U(x,t)|_{\partial D} & = g(x,t), \quad \forall (x,t) \in D \times \mathbb{R}_+. \\
U(x,0) & = U_0(x),
\end{aligned}
\]

(1.1)

Here, \( U_t \) denotes the time derivative \( \partial / \partial t \) of the vector \( U \) denoting the physical variables (which are conserved if \( S = 0 \)), where

\[
U = U(x,t) : D \times \mathbb{R}_+ \rightarrow \mathbb{R}^m, \quad m \geq 1,
\]

and the flux

\[
F : \mathbb{R} \times \mathbb{R}^m \rightarrow (\mathbb{R}^m)^d, \quad F = (F_1, \ldots, F_d),
\]

is a collection of directional vector-valued flux functions, where

\[
F_r(c, U) : \mathbb{R} \times \mathbb{R}^m \rightarrow \mathbb{R}^m, \quad r = 1, \ldots, d.
\]

The divergence is a differential operator corresponding to the spatial domain \( D \) and is given by

\[
\text{div} := \sum_{r=1}^d \frac{\partial}{\partial x_r}.
\]
Introduction

Each flux function \( F \) might depend on a (possibly also vector-valued) coefficient 
\[
c(x, t) : D \times \mathbb{R}_+ \to \mathbb{R},
\]
modeling material properties of the medium. Finally, the source term is given by 
\[
S(x, t, U) : D \times \mathbb{R}_+ \times \mathbb{R}^m \to \mathbb{R}^m.
\]
The partial differential equation is augmented with initial data 
\[
U_0 : D \to \mathbb{R}^m
\]
and boundary data \( g \), imposed through a suitable boundary operator \( L_b \), 
\[
g : \partial D \times \mathbb{R}_+ \to \mathbb{R}^m.
\]
For zero source term \( S \equiv 0 \), the balance law \((1.1)\) is termed a \textit{conservation law}.

In the following subsection, we review several examples of \((1.1)\).

1.2 Examples

Examples for conservation (balance) laws include the Burgers’ equation, the Euler equations of gas dynamics, the Magnetohydrodynamics (MHD) equations of plasma physics, the shallow water equations of oceanography, the acoustic and elastic wave equations, the equations of non-linear elasticity, and the Buckley-Leverett equation of two phase flows in oil reservoirs. We briefly introduce all these examples in the following subsections.

1.2.1 Burgers’ equation

Burgers’ equation \([66]\) is one of the simplest examples of a \textit{non-linear scalar} (i.e. \( m = 1 \) in \((1.1)\)) conservation law. There is also only one conserved variable \( U_1 = u \) and one spatial variable \( x = x_1 \in D \), and the equation is given by 
\[
u_t + f(u)_x = 0, \quad f(u) = \frac{u^2}{2}, \quad x \in D, \ t > 0. \quad (1.2)
\]
Even though equation \((1.2)\) has a very simple form, many phenomena of interest for the multi-dimensional non-linear systems of conservation laws are already present in this case. Hence, Burgers’ equation is often used as a prototypical example for the analysis of \textit{non-linear} conservation laws.
1.2 Examples

1.2.2 Buckley-Leverett equation

Another example of a scalar conservation law (1.1) is the one-dimensional \((d = 1)\) Buckley-Leverett equation [5], modeling the propagation of two phases: oil phase \(S_o\) and water phase \(S_w\). There is only one conserved variable

\[
U_1 = S = S_o = 1 - S_w,
\]

representing the saturation of oil \(S_o\), and the conservation law is as follows,

\[
\begin{cases}
S_t + f(S)_x = 0, \\
f(S) = \frac{qK\lambda_o(S)}{\lambda_w(S) + \lambda_o(S)},
\end{cases}
\]

\((1.3)\)

Here, \(q\) is the total flow rate, \(K\) is the rock permeability and \(\lambda_w, \lambda_o : [0, 1] \to \mathbb{R}\) are the relative permeabilities of the water and oil phases, respectively. The Buckley-Leverett equation is mainly used to model water flooding in an oil or gas reservoir [5].

1.2.3 Acoustic wave equation

An example of a linear system of conservation laws (1.1) is the isotropic acoustic wave equation [33, 53], modeling the propagation of acoustic pressure \(p\) in a heterogeneous medium:

\[
p_{tt}(x, t) - \text{div}(c(x)\nabla p(x, t)) = 0, \quad x \in D, \ t > 0.
\]

\((1.4)\)

Here, the material coefficient \(c : D \to \mathbb{R}_+\) models the (positive) speed of sound in a heterogeneous medium at a given domain point \(x \in D\). For simplicity, only stationary initial data will be considered, i.e. \(p_t(x, 0) \equiv 0\), but \(p(x, 0)\) can be chosen arbitrarily.

The linear acoustic wave equation (1.4) can be rewritten as a linear system of \(m = d + 1\) first order conservation laws. One possibility (out of many) is the following,

\[
U = \begin{bmatrix} p \\ u \end{bmatrix}, \quad \begin{cases} p_t(x, t) - \text{div}(c(x)u(x, t)) = 0, \\
u_t(x) - \nabla p(x) = 0, \quad x \in D, \ t > 0. \end{cases}
\]

\((1.5)\)

To verify the equivalence of (1.5) and (1.4), the first equation of (1.5) is differentiated in time and then the second equation of (1.5) is substituted:

\[
p_{tt} - \text{div}(c(x, \omega)u_t) = p_{tt} - \text{div}(c(x, \omega)\nabla p).
\]

Note that for the wave equation (1.5), the resulting fluxes \(F_r\) in (1.1) are linear:

\[
F_r(c(x), U) = F_r(x, U) = A_r(x)U, \quad A_r : D \to \mathbb{R}^{m \times m}, \quad r = 1, \ldots, d.
\]

\((1.6)\)
Introduction

where all entries \((A_r(x))_{i,j}\) of \(A_r\) for \(i, j = 1, \ldots, m\) are as follows,

\[
(A_r(x))_{i,j} = \begin{cases} 
-c(x) & i = 1, j = r + 1, \\
-1 & i = r + 1, j = 1, \\
0 & \text{else},
\end{cases}
\]

The wave equation is mainly used in structural mechanics and to model sound propagation through geological layers, where the coefficient \(c\) is highly heterogeneous, possibly even discontinuous at the interface of different materials.

1.2.4 Euler equations

A prototypical example of systems of non-linear conservation laws \((1.1)\) is the Euler equations of gas dynamics, which describe the time evolution of mass density, pressure and velocities in compressible fluids. The equations are given by [4, 110]

\[
U = \begin{bmatrix} 
\rho \\
\rho u \\
E
\end{bmatrix}, \quad \begin{cases} 
\rho_t + \text{div}(\rho u) = 0, \\
(\rho u)_t + \text{div}(\rho u \otimes u + pI) = 0, \\
E_t + \text{div}((E + p)u) = 0.
\end{cases}
\]

Here, \(\rho\) is the mass density and \(u\) is the velocity field. The pressure \(p\) and total energy \(E\) are related by the ideal gas equation of state:

\[
E := \frac{p}{\gamma - 1} + \frac{1}{2} \rho |u|^2,
\]

with \(\gamma\) being the ratio of specific heats, which, in the case of a monatomic gas, is \(5/3\). Euler equations are used, for instance, in the design of aircraft, internal combustion engines and gas turbines, among other models.

1.2.5 Equations of magnetohydrodynamics

Another example of \((1.1)\) that we will consider are the equations of Magnetohydrodynamics (MHD) given by [41]

\[
U = \begin{bmatrix} 
\rho \\
\rho u \\
B \\
E
\end{bmatrix}, \quad \begin{cases} 
\rho_t + \text{div}(\rho u) = 0, \\
(\rho u)_t + \text{div}(\rho u \otimes u + (p + \frac{1}{2}|B|^2)I - B \otimes B) = 0, \\
B_t + \text{div}(u \otimes B - B \otimes u) = 0, \\
E_t + \text{div}((E + p + \frac{1}{2}|B|^2)u - (u \cdot B)B) = 0, \\
\text{div}(B) = 0.
\end{cases}
\]
1.3 General concepts

Here, in addition to the density $\rho$ and the velocity field $\mathbf{u}$, $\mathbf{B}$ denotes the magnetic field and the total energy $E$ is given by the following equation of state:

$$E := \frac{p}{\gamma - 1} + \frac{1}{2} \rho |\mathbf{u}|^2 + \frac{1}{2} |\mathbf{B}|^2. \quad (1.11)$$

MHD equations are used to model, for instance, plasmas (e.g. in the sun), liquid metals, various electrolytes and other phenomena where the density, pressure and velocities are interacting with the magnetic field in an electrically conducting fluid.

Note that the Euler equations [1.8] are recovered from the MHD equations [1.10] by setting the magnetic field $\mathbf{B} \equiv 0$.

1.2.6 Shallow water equations

Many interesting flows, for instance in lakes, rivers, irrigation channels, avalanches, landslides, and tsunamis in oceans have the common property that the vertical scale (depth) of the flow is much smaller than the horizontal scales of motion. Hence, the full three-dimensional incompressible Navier-Stokes equations of fluid dynamics can be simplified to the so-called shallow water equations [117], where only two dimensional velocity fields $\mathbf{u} \in \mathbb{R}^2$ are modeled, resulting in

$$\mathbf{U} = \begin{bmatrix} h \\ h \mathbf{u} \end{bmatrix}, \quad \begin{cases} h_t + \text{div}(h \mathbf{u}) = 0, \\ (h \mathbf{u})_t + \text{div} \left( h \mathbf{u} \otimes \mathbf{u} + \frac{1}{2} gh^2 I \right) = -gh \nabla b. \end{cases} \quad (1.12)$$

Here, $h$ denotes the height of the fluid column above the bottom topography

$$b = b(\mathbf{x}) : \mathcal{D} \to \mathbb{R}, \quad \mathcal{D} \subseteq \mathbb{R}^2,$$

over which the fluid flows and $\mathbf{u} = (\mathbf{u}_1, \mathbf{u}_2) \top$ is the vertically averaged (or depth averaged) horizontal fluid velocity field. The constant $g$ denotes the magnitude of the negative vertical acceleration due to gravity. Two main applications of shallow water equations arise in the fields of risk assessment of region flooding (due to tsunami or dambreak) and atmospheric flows for weather prediction and climate dynamics.

1.3 General concepts

For simplicity and ease of exposition, we set $\mathcal{D} = \mathbb{R}^d$ throughout this section. Then, the resulting system of conservation laws (1.1) corresponds to a Cauchy problem, where the boundary condition $L_b \mathbf{U}(\mathbf{x}, t)|_{\partial \mathcal{D}} = \mathbf{g}(\mathbf{x}, t)$ no longer needs to be specified as $\partial \mathcal{D} = \emptyset$. Similarly, bounded periodic Cartesian domains $\mathcal{D} = I_1 \times \cdots \times I_d$ for $I_r \subset \mathbb{R}$ bounded and connected, $r = 1, \ldots, d$ could be considered with periodic boundary conditions.
Introduction

It is well known that, in general, solutions of nonlinear hyperbolic systems (1.1) develop discontinuities or shock waves in finite time even for smooth initial data \[66\]; for an example, refer to the solution of the Burgers’ equation (1.2) on the domain \(D = [0, 2]\) with periodic boundary conditions in Figure 1.1 where an initial smooth sine function develops a discontinuity at \(x = 1\). Hence, in the forthcoming sections, solutions of (1.1) are sought in the sense of distributions and are called weak solutions. Furthermore, additional admissibility criteria or entropy conditions are required to ensure uniqueness of weak solutions.

Figure 1.1: Initial data \(u_0\) (left) and the solution \(u(x, t)\) at \(t = 1\) (right) of the Burgers’ equation (1.2) on periodic domain \(D = [0, 2]\). Propagation (flux) direction \(f'(u) = u\) is rightwards (positive) in the left part of the domain and leftwards (negative) in the right part; hence, even though the initial data \(u_0\) is a smooth sine function, a discontinuity in the solution \(u(x, t)\) at \(x = 1\) develops in finite time.

1.3.1 Hyperbolicity and nonlinearity

We start by defining hyperbolicity for a general system of conservation laws (1.1).

**Definition 1.3.1 (Hyperbolicity).** For all domain points \(x \in \mathbb{R}^d\), consider the Jacobians of the flux functions

\[
DF_r(x, U) : \mathbb{R}^d \times \mathbb{R}^m \to \mathbb{R}^m,
\]

and define an arbitrary convex combination of such Jacobians \(DF_r(x, U)\) by

\[
P^w : \mathbb{R}^d \times \mathbb{R}^m \to \mathbb{R}^m, \quad P^w(x, U) = \sum_{r=1}^{d} w_r DF_r(x, U), \quad w \in S^{d-1}.
\]

Consider the eigen-decompositions of all possible \(P^w\), i.e. for all \(w \in S^{d-1}\),

\[
P^w(x, U) = Q^w(x, U)\Lambda^w(x, U)Q^w(x, U)^{-1},
\]

\[
\Lambda^w(x, U) = \text{diag}(\lambda_1^w, \ldots, \lambda_m^w)(x, U), \quad \text{(1.13)}
\]
1.3 General concepts

where \( \Lambda^w(x, U) \) are the diagonal matrices containing the eigenvalues
\[
\lambda_1^w, \ldots, \lambda_m^w,
\]
and \( Q^w(x, U) \) are the corresponding similarity transformation matrices containing eigenvectors as columns.

The system of conservation laws (1.1) is hyperbolic \[53\] in \( C \in \mathbb{R}^m \) if for each \( x \in \mathbb{R}^d \) and \( U \in C \) all eigenvalues \( \lambda_1^w(x, U), \ldots, \lambda_m^w(x, U) \) are real and there exists \( K < \infty \) such that
\[
\sup_{x \in D, U \in C, w \in \mathbb{Z}^{d-1}} \| Q^w(x, U)^{-1} \| \| Q^w(x, U) \| \leq K. \tag{1.14}
\]

The system (1.1) is strictly hyperbolic \[53\] if all eigenvalues of \( P^w(x, U) \) are different.

We remark that scalar conservation laws (i.e. \( m = 1 \)) are, by Definition 1.3.1, always hyperbolic. As for one-dimensional \( (d = 1) \) systems of conservation laws, the weight vector can be set to \( w = 1 \) and hence the eigendecomposition in (1.13) reduces to
\[
DF(x, U) = Q(x, U)\Lambda(x, U)Q(x, U)^{-1},
\]
\[
\Lambda(x, U) = \text{diag}(\lambda_1, \ldots, \lambda_m)(x, U). \tag{1.15}
\]

Linear hyperbolic systems and characteristic variables

For systems of linear conservation laws with linear fluxes as in (1.6), Definition 1.3.1 is reduced to the requirement that
\[
DF(x, U) = DF(x) = \Lambda(x) : \mathbb{R}^m \to \mathbb{R}^{m \times d} \tag{1.16}
\]
is diagonalizable for all \( x \in \mathbb{R}^d \) and to has distinct real eigenvalues.

Using the characteristic eigenvalue decomposition (1.13), conserved variables \( U \) can be transformed into characteristic variables \( W \) via the change of basis matrix \( Q^{-1}(x) \),
\[
W(x, t) = Q^{-1}(x)U(x, t), \quad x \in \mathbb{R}^d, \quad t > 0. \tag{1.17}
\]

Then (1.1) with linear fluxes (1.6) can be stated in terms of characteristic variables \( W \),
\[
\begin{cases}
W(x, t) + \sum_{r=1}^d \Lambda_r(x) \frac{\partial}{\partial x_r} W(x, t) = 0, \\
W(x, 0) = Q^{-1}U_0(x),
\end{cases} \quad \forall (x, t) \in \mathbb{R}^d \times \mathbb{R}_+. \tag{1.18}
\]
Introduction

The system (1.18) consists of \( m \) completely uncoupled linear scalar conservation laws (also called transport equations)

\[
W_t^p(x, t) + \sum_{r=1}^d \lambda_r^p(x) \frac{\partial}{\partial x_r} W^p(x, t) = 0, \quad \forall (x, t) \in \mathbb{R}^d \times \mathbb{R}_+, \quad p = 1, \ldots, m, \quad (1.19)
\]

for variables

\((W^1, \ldots, W^m) = W \in \mathbb{R}^m,\)

and linear directional fluxes (transport speeds)

\((\lambda_1^r, \ldots, \lambda_m^r) = \Lambda_r, \quad r = 1, \ldots, d.\)

Nonlinear hyperbolic systems

Most natural phenomenon (including gas dynamics, plasma physics, shallow water dynamics) are non-linear, hence we relax the assumption (1.6) of linear flux \( F(U) \).

Definition 1.3.2 (Genuine non-linearity). The \( p \)-th characteristic field \((\lambda_p, q_p)\) is said to be genuinely non-linear in a subset \( K \subseteq \mathbb{R}^m \) of states and on the physical domain \( D \subseteq \mathbb{R}^d \) if

\[ \nabla \lambda_p(x, U) \cdot q_p(x, U) \neq 0, \quad \forall U \in K, \quad x \in D. \quad (1.20) \]

Here, \( \lambda_p \) and \( q_p \) are the eigenvalues and eigenvectors of \( DF(x, U) \) as in (1.13), and

\[ \nabla \lambda_p(x, U) = \left( \frac{\partial \lambda_p}{\partial U_1}, \ldots, \frac{\partial \lambda_p}{\partial U_m} \right)^\top \quad (1.21) \]

denotes the gradient of \( \lambda_p(x, U) \).

Note that for the scalar conservation law (i.e. \( m = 1 \) and setting \( f(u) = F_1(u) \)), the characteristic field is given by \( \lambda_1(u) = f'(u) \) and \( q_1(u) \equiv 1 \), hence condition (1.20) reduces to \( f''(u) \neq 0 \), requiring the flux \( f \) to be either strictly convex or strictly concave.

Definition 1.3.3 (Linear degeneracy). Since monotonicity of the propagation speed \( \lambda_p \) is required through a rarefaction wave [66], genuine non-linearity is a natural assumption. However, a general non-linear hyperbolic system might have one of the characteristic fields with \( \lambda_p \) constant along the integral curves; then the assumption (1.20) is obviously violated, since \( \nabla \lambda_p \equiv 0 \). Such characteristic fields are called linearly degenerate.

It is well known that the Euler equations of gas dynamics (1.8) are strictly hyperbolic and the corresponding eigenvalues and eigenvectors are readily computed [66]. However, the \( p = 2 \) characteristic field of Euler equations (1.8) in the one dimensional case (\( d = 1 \)) is linearly degenerate. Such discontinuities in linearly degenerate characteristic fields are called contact discontinuities; jumps then occur in some of the conserved variables

8
1.3 General concepts

(e.g. in density $\rho$), whereas the remaining variables remain continuous (e.g. momentum $\rho u$ and pressure $p$).

In contrast to the Euler equations, the MHD equations (1.10) are not strictly hyperbolic due to the $\text{div} \mathbf{B} \equiv 0$ constraint. However, like Euler equations, the MHD equations also possess linearly degenerate characteristic fields and contain contact discontinuities.

1.3.2 Weak solutions and Rankine-Hugoniot condition

It is well known that solutions to nonlinear systems of conservation laws (1.1), given arbitrary initial data $u_0$, might develop discontinuities in finite time [66], even for $d = 1$, see the example in Figure 1.1. Hence, the classical pointwise notion of solutions is no longer sufficient. Therefore, (1.1) is recast into its weak (integral) form following the usual techniques: by multiplying (1.1) with smooth, compactly supported test functions $\varphi \in C^\infty_c(\mathbb{R}^d \times \mathbb{R}^+)$, integrating over the space-time domain $\mathbb{R}^d \times \mathbb{R}^+$ and, finally, integrating by parts, we obtain,

$$\int_{\mathbb{R}^d \times \mathbb{R}^+} \left( U \cdot \varphi_t + \sum_{r=1}^d F_r \cdot \frac{\partial}{\partial x_i} \varphi \right) \, dx \, dt + \int_{\mathbb{R}^d} U_0 \cdot \varphi |_{t=0} \, dx = \int_{\mathbb{R}^d \times \mathbb{R}^+} \mathbf{S} \cdot \varphi \, dx \, dt. \tag{1.22}$$

Differential operators $\frac{\partial}{\partial t}$ and $\frac{\partial}{\partial x}$ are applied component-wise,

$$\varphi_t = \left( \frac{\partial}{\partial t} \varphi_1, \ldots, \frac{\partial}{\partial t} \varphi_m \right), \quad \varphi_{x_i} = \left( \frac{\partial}{\partial x_i} \varphi_1, \ldots, \frac{\partial}{\partial x_i} \varphi_m \right),$$

and the usual inner product of vectors is defined by

$$\mathbf{a} \cdot \mathbf{b} = a_1 b_1 + \cdots + a_m b_m, \quad \mathbf{a}, \mathbf{b} \in \mathbb{R}^m.$$

The following notation for Banach spaces of vector-valued functions will be used:

$$C^r(\cdot) = (C^r(\cdot))^m, \quad L^p(\cdot) = (L^p(\cdot))^m, \quad W^{r,p}(\cdot) = (W^{r,p}(\cdot))^m.$$

**Definition 1.3.4 (Weak solution).** A locally integrable function $U \in L^1_{\text{loc}}(\mathbb{R}^d \times \mathbb{R}^+)$ satisfying (1.22) for all smooth test functions $\varphi \in C_c^\infty(\mathbb{R}^d \times \mathbb{R}^+)$ is called a weak solution of the system of conservation laws (1.1) with $D = \mathbb{R}^d$ and the locally integrable initial data $U_0(x) \in L^1_{\text{loc}}(\mathbb{R}^d)$.

The following result provides the necessary and sufficient conditions for weak solutions in the case of piecewise differentiable initial data

$$U_0 \in C^1_{\text{pw}}(\mathbb{R}^d \times \mathbb{R}^+).$$
Theorem 1.3.5 (Rankine-Hugoniot). Consider a system of conservation laws (1.1) on \( \mathbb{R}^d \), complemented by a piecewise differentiable initial data function \( U_0 \in C_{pw}^1(\mathbb{R}^d) \). Then \( U \in C_{pw}^1(\mathbb{R}^d \times \mathbb{R}_+) \) is a weak solution to (1.1) if and only if the following holds

1. \( U \) is a classical solution inside the subdomains of \( D \) where \( U \in C^1(\mathbb{R}^d \times \mathbb{R}_+) \),

2. \( U \) satisfies the Rankine-Hugoniot condition across the discontinuity,

\[
(U_+ - U_-) \eta_t + \sum_{r=1}^{d} (F_r(U_+) - F_r(U_-)) \eta_{x_r} = 0.
\]

where \( \eta = (\eta_t, \eta_{x_1}, \ldots, \eta_{x_d}) \) is a unit normal vector to the discontinuity hyper-surface in \( \mathbb{R}^d \times \mathbb{R}_+ \); \( U_+ \) and \( U_- \) denote the traces of the solution \( U \) at the discontinuity,

\[
U_+(x,t) = \lim_{\varepsilon \to 0} U(x + \varepsilon \eta_{x}, t + \varepsilon \eta_t), \quad U_-(x,t) = \lim_{\varepsilon \to 0} U(x - \varepsilon \eta_{x}, t - \varepsilon \eta_t).
\]

The proof of Theorem 1.3.5 is based on successive elementary analytic manipulations such as integration by parts and application of the chain rule [66].

1.3.3 Entropy solutions

Weak solutions for (1.1) are generally not unique [66]. Hence, some additional criteria are needed to restrict the notion of solutions. Such criteria come from the physical meaning of the evolution equation: information must flow from the initial data (and boundary conditions, if present) only, and no information should be created at the shocks.

Definition 1.3.6 (Entropy pairs). A convex function \( S : \mathbb{R}^m \to \mathbb{R} \) is called an entropy function if for each \( r = 1, \ldots, d \) there exist functions \( Q_r : \mathbb{R}^m \to \mathbb{R} \), called entropy fluxes, such that the following holds:

\[
\frac{\partial}{\partial U} Q_r(U) = \frac{\partial}{\partial U} S(U) \cdot \frac{\partial}{\partial U} F_r(U).
\]

The pair \((S,Q)\) with \( Q = \{Q_1, \ldots, Q_d\} \) is called an entropy pair.

For scalar conservation laws (i.e. \( m = 1 \) in (1.1)), we denote the single conserved scalar variable by \( u(x,t) = U_1(x,t) \), the initial condition by \( u_0(x) = U_0(x) \), and the flux function (for simplicity, not depending on \( c \)) by

\[
f : \mathbb{R} \to \mathbb{R}^d, \quad f_r(u(x,t)) = F_r(\cdot, U(x,t)), \quad r = 1, \ldots, d.
\]
1.3 General concepts

The resulting Cauchy problem \((D = \mathbb{R}^d)\) for a scalar conservation law reads

\[
\begin{align*}
\left\{ \begin{array}{l}
  u_t + \text{div}(f(u)) = 0, \\
  u(x, 0) = u_0(x),
\end{array} \right.
\forall (x, t) \in D \times \mathbb{R}_+.
\]

(1.25)

For such a scalar conservation law \((1.25)\), any convex function \(S : \mathbb{R} \to \mathbb{R}\) is an entropy function with entropy fluxes \(Q_r : \mathbb{R} \to \mathbb{R}\) defined by

\[
Q_r(u) := \int_0^u f'_r(\xi)S'(\xi)\,d\xi, \quad r = 1, \ldots, d.
\]

By definition, such entropy pairs \((S, Q)\) satisfy the scalar version of \((1.24)\),

\[
Q'_r = S'f'_r, \quad r = 1, \ldots, d.
\]

Particularly important are the so-called Kružkov entropy pairs \([61]\),

\[
S = S(u; k) := |u - k|, \quad Q_r(u; k) = \text{sign}(u - k)(f_r(u) - f_r(k)).
\]

(1.26)

For systems of conservation laws \((1.1)\) with \(m > 1\), finding appropriate entropy pairs is more difficult \([24]\).

A classical solution \(U\) of \((1.1)\), satisfies the following conservation law,

\[
S(U)_t + \sum_{r=1}^d \frac{\partial}{\partial x_r} Q_r(U) = 0.
\]

(1.27)

For an arbitrary weak solution \(U\) of \((1.1)\), consider a viscous approximation of \((1.1)\) by adding a small diffusion term,

\[
U^\varepsilon_t + \text{div} F(U^\varepsilon) = \varepsilon \Delta U^\varepsilon, \quad \varepsilon > 0.
\]

(1.28)

This perturbation with second order diffusion term \(\varepsilon \Delta U^\varepsilon\) transforms the system of conservation laws \((1.1)\) into an advection-diffusion equation which is known to possess smooth solutions \(U^\varepsilon \in C^\infty(\mathbb{R}^d \times \mathbb{R}_+)\).

Multiplying \((1.28)\) by \(DS(U^\varepsilon)\), applying \((1.24)\) and using the chain rule successively, we find (analogously as in \([11, 24]\))

\[
\begin{align*}
DS(U^\varepsilon) \cdot \frac{\partial}{\partial t} U^\varepsilon + \sum_{r=1}^d DS(U^\varepsilon) \cdot \frac{\partial}{\partial x_r} F_r(U^\varepsilon) &= \varepsilon DS(U^\varepsilon) \cdot \Delta U^\varepsilon, \\
DS(U^\varepsilon) \cdot \frac{\partial}{\partial t} U^\varepsilon + \sum_{r=1}^d DQ_r(U^\varepsilon) \cdot \frac{\partial}{\partial x_r} U^\varepsilon &= \varepsilon DS(U^\varepsilon) \cdot \Delta U^\varepsilon, \\
\frac{\partial}{\partial t} S(U^\varepsilon) + \sum_{r=1}^d \frac{\partial}{\partial x_r} Q_r(U^\varepsilon) &= \varepsilon \Delta S(U^\varepsilon) - \varepsilon \det(D^2 S(U^\varepsilon)) \sum_{r=1}^d (U^\varepsilon_{x_r})^2.
\end{align*}
\]
Introduction

Since $S$ is convex, we have \( \det(D^2 S(U^\varepsilon)) \geq 0 \) on \( \mathbb{R}^m \), and hence the above simplifies to
\[
S(U^\varepsilon)_t + \sum_{r=1}^d \frac{\partial}{\partial x_r} Q_r(U^\varepsilon) \leq \varepsilon \Delta S(U^\varepsilon).
\]

For sufficiently smooth solutions \( U^\varepsilon \) of (1.28), \( U \) can be expressed as a vanishing viscosity solution, i.e. as the limit of \( U^\varepsilon \) as \( \varepsilon \to 0 \). The entropy equality (1.27) can be generalized to an entropy inequality, defining the notion of entropy solutions to (1.1).

**Theorem 1.3.7** (Vanishing viscosity limit). Assume that for the non-linear system of conservation laws (1.1) with \( m = 1 \) and \( D = \mathbb{R}^d \) there exists an entropy pair \((S,Q)\), and let \( \{U^\varepsilon\}_{\varepsilon > 0} \) be a family of solutions to the viscous version (1.28) of (1.1), such that
\[
\sup_{\varepsilon > 0} \|U^\varepsilon\|_{L^\infty(\mathbb{R}^d \times \mathbb{R}^+)^m} < C < \infty.
\]

The limit \( U = \lim_{\varepsilon \to 0} U^\varepsilon \in L^1_{\text{loc}}(\mathbb{R}^d \times \mathbb{R}^+) \), if it exists, satisfies the entropy inequality
\[
S(U)_t + \sum_{r=1}^d \frac{\partial}{\partial x_r} Q_r(U) \leq 0. \tag{1.29}
\]

For a proof, we refer to [11, 24].

As weak solutions \( U \) of (1.1) are of main interest, pointwise values and hence derivatives might not be available everywhere. Hence, the entropy inequality (1.29) is to be understood in a weak (distributional) sense, i.e.
\[
\int_{\mathbb{R}^d \times \mathbb{R}^+} \left( S(U(x,t))\varphi_t(x,t) + \sum_{r=1}^d Q_r(U(x,t)) \frac{\partial}{\partial x_r} \varphi(x,t) \right) \, dx \, dt \geq 0, \tag{1.30}
\]
for all test functions \( 0 \leq \varphi \in C^\infty_0(\mathbb{R}^d \times \mathbb{R}^+) \).

**Definition 1.3.8** (Entropy solution). For a genuinely non-linear conservation law (1.1) on \( D = \mathbb{R}^d, U \in L^1_{\text{loc}}(\mathbb{R}^d \times \mathbb{R}^+) \) is called the entropy solution if it is a weak solution (1.22) and if it satisfies the entropy inequality (1.29) for all possible entropy pairs \((S,Q)\).

A very important class of systems of conservation laws are considered with initial data consisting of only two different states and an exactly one discontinuity between them; in such cases, an equivalent (much simpler in formulation) definition of the entropy condition is available for \( d = 1 \), called Lax entropy condition [62].

**Definition 1.3.9** (Lax entropy condition). For a genuinely non-linear conservation law (1.1) on \( D = \mathbb{R}^d, U \in L^1_{\text{loc}}(\mathbb{R} \times \mathbb{R}^+) \) is called entropy solution of the Riemann problem
\[
U_0(x) = \begin{cases} U_L & \text{if } x \leq 0, \\ U_R & \text{if } x > 0, \end{cases} \tag{1.31}
\]
1.3 General concepts

if it is a weak solution $\text{(1.22)}$ and if the jump in the $p$-th characteristic field is admissible,

$$
\lambda_p(U_L) \geq s(t) \geq \lambda_p(U_R), \quad s(t) = \frac{\partial}{\partial t} \sigma(t),
$$

(1.32)

for every $p = 1, \ldots, m$. Here, $\sigma, s : \mathbb{R}_+ \to \mathbb{R}$ denote shock location and speed at time $t$.

Remark 1.3.10. Equality in (1.32) is allowed to include linearly degenerate cases, since contact discontinuities occur in vanishing viscosity (entropy) solutions. For the scalar conservation laws (i.e. $m = 1$), (1.32) is also known as Oleinik entropy condition $[84]$.

1.3.4 Existence results

Before stating some existence results for nonlinear conservation laws, we first define the total variation norm, corresponding to the semi-norm on the space of functions with integrable distributional derivatives.

**Total variation**

For scalar conservation laws $\text{(1.25)}$ in space dimension $d \geq 1$, the total variation ($TV$) semi-norm $TV(u) = |u|_{TV(D)}$ of $u \in L^1(D)$ is defined as the integral of the weak (distributional) derivative of $u$,

$$
|u|_{TV(D)} = \sup \left\{ \int_D u(x) \text{div} \varphi(x) dx \mid \varphi \in C^1_c(D, \mathbb{R}^d), \|\varphi\|_{L^\infty(D)} \leq 1 \right\}. \quad (1.33)
$$

The corresponding space of functions with bounded variation ($BV$) is defined as

$$
BV(D) = \{ u \in L^1(D) : \|u\|_{BV(D)} < \infty \}, \quad (1.34)
$$

where the $BV(D)$-norm combines $L^1(D)$-norm and $TV(D)$-semi-norm,

$$
\|u\|_{BV(D)} = \|u\|_{L^1(D)} + |u|_{TV(D)}. \quad (1.35)
$$

**Scalar conservation laws**

For scalar ($m = 1$) conservation laws $\text{(1.25)}$, the notion of weak entropy solutions as defined in $\text{(1.22)} - \text{(1.29)}$ ensures the existence and uniqueness of the solution.

**Theorem 1.3.11 (Kružkov).** For $u_0 \in L^\infty(\mathbb{R}^d) \cap BV(\mathbb{R}^d)$, the scalar conservation law $\text{(1.25)}$ on $D = \mathbb{R}^d$ has an entropy solution $u \in L^\infty(\mathbb{R}^d \times [0, T])$. Moreover, the (nonlinear) data-to-solution operator

$$
S : u_0 \mapsto u(\cdot, t) = S(t)u_0, \quad \forall t > 0,
$$

(1.36)

satisfies the following estimates:
1. \( S(t) : L^1(\mathbb{R}^d) \rightarrow L^1(\mathbb{R}^d) \) is a (contractive) Lipschitz map, i.e.
\[
\| S(t)u_0 - S(t)v_0 \|_{L^1(\mathbb{R}^d)} \leq \| u_0 - v_0 \|_{L^1(\mathbb{R}^d)}, \quad \forall u_0, v_0 \in L^1(\mathbb{R}^d),
\] (1.37)
hence, entropy solutions are unique.

2. \( S(t) \) maps \((L^1 \cap BV)(\mathbb{R}^d)\) into \((L^1 \cap BV)(\mathbb{R}^d)\) and
\[
TV(S(t)u_0) \leq TV(u_0), \quad \forall u_0 \in (L^1 \cap BV)(\mathbb{R}^d).
\] (1.38)

3. For every \( u_0 \in (L^\infty \cap L^1)(\mathbb{R}^d) \),
\[
\| S(t)u_0 \|_{L^\infty(\mathbb{R}^d)} \leq \| u_0 \|_{L^\infty(\mathbb{R}^d)},
\] (1.39)
\[
\| S(t)u_0 \|_{L^1(\mathbb{R}^d)} \leq \| u_0 \|_{L^1(\mathbb{R}^d)}.
\] (1.40)

4. The mapping \( S(t) \) is a uniformly continuous mapping
\[
L^1(\mathbb{R}^d) \rightarrow C_b([0, \infty], L^1(\mathbb{R}^d)),
\] and
\[
\| S(\cdot)u_0 \|_{C_b([0,T],L^1(\mathbb{R}^d))} = \max_{0 \leq t \leq T} \| S(t)u_0 \|_{L^1(\mathbb{R}^d)} \leq \| u_0 \|_{L^1(\mathbb{R}^d)},
\] (1.41)
where for a Banach space \( E \) with norm \( \| \cdot \|_E \) and for \( 0 < T \leq +\infty, C_b([0,T],E) \) denotes the space of bounded and continuous functions from \([0,T]\) into \( E \).

The existence proof is based on the viscous approximation \((1.28)\). The uniqueness proof is based on the ingenious doubling of variables idea by Kružkov \([61]\) and also heavily relies on Kružkov entropies \((1.26)\). We refer to \([49]\) for details.

### Linear hyperbolic systems of conservation laws

For linear hyperbolic systems (i.e. \( m > 1 \)) of conservations laws with linear fluxes \((1.6)\), i.e.
\[
U_t + \sum_{r=1}^{d} \frac{\partial}{\partial x_r} (A_r(x,t)U(x,t)) = S(x,t), \quad \forall (x,t) \in \mathbb{R}^d \times \mathbb{R}_+,
\] (1.42)
classical existence and uniqueness results are also available. Following section 1.3.4 we assume \( D = \mathbb{R}^d \) (for simplicity of exposition) and consider resulting Cauchy problems.

Let \( V \) denote an arbitrary Banach space. The following notation will be used:
\[
\| K, U, S, t \|_V = K(\| U \|_V + t \| S \|_V), \quad K \geq 1, \quad U, S \in V, \quad t \geq 0.
\] (1.43)

The following result recapitulates some of the classical existence and uniqueness results \([53, 66, 119]\) for weak solutions of linear hyperbolic systems \((1.42)\).
1.3 General concepts

**Theorem 1.3.12.** Assume $\mathbf{D} = \mathbb{R}^d$ and that

1. the system of conservation laws (1.42) is strongly hyperbolic with $K < \infty$ in (1.14),
2. there exist $r_0, r_S \in \mathbb{N} \cup \{0, \infty\}$ and $r_A \in \mathbb{N} \cup \{\infty\}$ such that for all $p \in \{2, \infty\}$ there holds
   \[ U_0 \in W^{r_0,p}(\mathbb{R}^d), \quad S \in W^{r_S,p}(\mathbb{R}^d), \quad A_r \in (C^{r_A}(\mathbb{R}^d))^{m \times m}. \] (1.44)

Then, for every finite time horizon $T < \infty$, system (1.42) admits a unique weak solution $U \in L^\infty(\mathbb{R}^d \times [0,T])$. Furthermore, for every $0 \leq t \leq T$, the a-priori estimates

\[ \|U(\cdot,t)\|_{L^2(\mathbb{R}^d)} \leq \|K, U_0, S, t\|_{L^2(\mathbb{R}^d)}, \] (1.45)
\[ \|U(\cdot,t) - V(\cdot,t)\|_{L^2(\mathbb{R}^d)} \leq \|K, U_0 - V_0, S_U - S_V, t\|_{L^2(\mathbb{R}^d)}, \] (1.46)

hold, and the regularity of the input data is inherited by the solution $U$, i.e.

\[ U \in C([0,T], W^{\bar{r},p}(\mathbb{R}^d)), \quad \bar{r} = \min\{r_0, r_S, r_A\}, \quad p \in \{2, \infty\}. \] (1.47)

**Remark 1.3.13.** The assumption of continuously differentiable coefficients $A_r$, i.e.

\[ A_r \in (C^1(\mathbb{R}^d))^{m \times m}, \]

can be, to some degree, relaxed. In particular, we refer to [74] for theoretical analysis and proposed discretization methods in the case of only Hölder continuous coefficients, i.e. for the wave equation (1.4) with material coefficients $c(x)$ in Hölder spaces $C^{0,\alpha}(\mathbb{R}^d)$ for some $\alpha > 0$.

**Non-linear systems of conservation laws**

For non-linear systems of conservation laws, no global well-posedness results are available. The main theoretical results are available only for one-dimensional non-linear systems of conservation laws: for the special case of Riemann initial data, Lax [62] showed existence and stability of entropy solutions; for a general Cauchy problem, existence was obtained by Glimm and is thoroughly introduced in [101, 45]; uniqueness and stability was obtained in [11]. Both results assume that the total variation of the initial data is “sufficiently small”, as outlined in the exact statement of the well-posedness result from Glimm, given below.

**Theorem 1.3.14 (Glimm).** Consider a system of conservation laws in one spatial dimension (1.1) complemented by initial data $U_0$. Assume $\mathbf{D} = \mathbb{R}$ and that:

1. (1.1) is hyperbolic and genuinely non-linear (linear degeneracies are also allowed) in some open non-empty set $\mathbf{K} \subset \mathbb{R}^m$;
2. $\{U_0(x) : x \in \mathbb{R}\} \subset \mathbf{K}$ and its total variation $TV(U_0)$ is sufficiently small.
Introduction

Then (1.1) has a weak entropy solution (according to Definition 1.3.8) for all times $t > 0$. This solution is unique and consists of at most $m + 1$ continuous states separated by shocks, rarefactions and contact discontinuities.

The proof of Theorem 1.3.14 is mainly based on the approximations to (1.1) via the Glimm discretization scheme [101, 45] and checking whether the limit is indeed a weak solution as in (1.22). We remark, that the Glimm discretization scheme already includes randomness, even though the underlying problem is deterministic.

No global (i.e. for large variation $TV(U_0)$ in initial data and long time horizon $T > 0$) well-posedness results (existence and uniqueness/stability) for systems in one or several space dimensions are available. In particular, no $L^\infty$-type bounds analogous to (1.39) and (1.47) are available in general. Furthermore, recent results in [27, 18] provide counterexamples illustrating that weak entropy solutions for multi-dimensional systems of conservation laws are not necessarily unique.

Analogously to non-linear scalar and linear hyperbolic systems of conservation laws, numerical schemes are required to approximate the solution to general systems of conservation laws (1.1) with arbitrarily complicated initial data. Due to lack of theoretical results for such systems, the design of the numerical schemes is based on the properties of the numerical schemes for the scalar and linear hyperbolic systems of conservations laws.

For the remaining part of the thesis, we assume that all nonlinear systems of conservation laws considered in chapters 7, 8 and 11, are well-posed for the specified input data, i.e. entropy solutions exist and are unique. Note, that this is not true in general, i.e. there exist counterexamples of nonlinear systems of conservation laws where solutions are non-unique or are not known to exist for all times $t > 0$.

An alternative notion of solutions to (1.1), extending the notion of entropy solutions (1.30), is the so-called entropy measure valued solutions (EMVS), introduced in [29]. In this framework, the solutions are no longer integrable functions, but parametrized probability measures (or Young measures), representing the limit behavior of the oscillating integrable functions. Recently, EMVS were advocated as an appropriate notion of solution for nonlinear hyperbolic system of conservation laws in several space dimensions, see [37] and references therein. In particular, existence and stability of measure valued solutions to (1.1) with the initial data being a Young measure is studied, together with a number of numerical experiments, confirming the findings of the theoretical results.

Bounded domains and boundary conditions

For bounded smooth domains $D \subset \mathbb{R}^d$ and suitable boundary conditions ($\mathcal{L}_b, g$), well-posedness results analogous to Theorem 1.3.11 and 1.3.12 i.e. for scalar and linear hyperbolic systems of conservation laws, were obtained by [6] (and extended by [32]) and by [53], respectively. In the setting of [6, 32], boundary conditions are enforced either by
means of the entropy boundary inequality, derived using the vanishing viscosity method, or directly by the Riemann problem at the boundary $\partial D$.

### 1.4 Finite Volume method

In the last few decades, finite volume methods (FVM) [66] have emerged as a very popular framework for approximating systems of balance (conservation) laws. We present this method in more detail.

#### 1.4.1 Formulation

For discretization schemes, we assume that the domain $D \subset \mathbb{R}^d$ is bounded and Lipschitz. We note that in such cases, boundary conditions need to be specified; further details on the implementation of certain boundary conditions are provided at the end of this section. For simplicity of exposition, bounded Cartesian spatial domains are considered throughout this thesis, i.e., we set

$$D = I_1 \times \cdots \times I_d \subset \mathbb{R}^d, \quad I_r \subset \mathbb{R}, \quad r = 1, \ldots, d,$$

with $I_r \subset \mathbb{R}$ bounded and connected, for all $r = 1, \ldots, d$. However, the FVM methodology introduced in the present section also extends to systems (1.1) in general polyhedral domains with suitable boundary conditions [77]. We begin by introducing a mesh on a given Cartesian domain $D \subset \mathbb{R}^d$.

#### Mesh

Let $\mathcal{T} = \mathcal{T}^1 \times \cdots \times \mathcal{T}^d$ denote a uniform axiparallel quadrilateral mesh covering the domain $D$; the mesh consists of identical non-overlapping open cells $C_j$.

$$C_j = C_j^1 \times \cdots \times C_j^d \subset I_1 \times \cdots \times I_d \subset \mathbb{R}^d, \quad j = 1, \ldots, \#\mathcal{T}^r,$$

and the cell center of each cell $C_j$ is denoted by $x_j \in \mathbb{R}^d$ with components $(x_{j1}, \ldots, x_{jd})$.

In order to work with fluxes across cell interfaces (boundaries) we will also need coordinates with “half” indices, denoting the midpoint $\Delta x_{j+\frac{1}{2}}$ between two adjacent cell midpoints $x_j$ and $x_{j+e_r}$ aligned with the direction $r$,

$$x_{j+\frac{1}{2}}e_r = \frac{1}{2} (x_j + x_{j+e_r}) = \left( x_{r,j+\frac{1}{2}d}, \ldots, x_{r,j+\frac{1}{2}} \right).$$

Here, $e_r$ denotes the set of canonical basis vectors $\{e_1, \ldots, e_d\}$ of the space $\mathbb{R}^d$.

For simplicity of exposition, we assume that mesh widths are equal in each dimension,

$$\Delta x := \frac{|I_1|}{\#\mathcal{T}^1} = \cdots = \frac{|I_d|}{\#\mathcal{T}^d}.$$
**Introduction**

**Integral form of the conservation law**

Next, we define the piecewise constant approximations $U_T$ and $S_T$ of the solution $U$ and source term $S$ by means of cell averaging on the mesh $T$, i.e.

$$U_T(x,t)\big|_{C_j} = U_j(t) \approx \frac{1}{|C_j|} \int_{C_j} U(x,t) dx,$$

and

$$S_T(x,t)\big|_{C_j} = S_j(t) \approx \frac{1}{|C_j|} \int_{C_j} S(x,t) dx.$$

Since the definition of weak solutions (1.22) only requires the integrability of $U$ and $S$, the cell averages $U_j$ and $S_j$ are well-defined.

Integrating the conservation law (1.1) over $C_j$ we obtain:

$$\int_{C_j} \partial_t U_j \, dx + \sum_{r=1}^{d} \int_{C_j} \frac{\partial}{\partial x_r} F_r(U) \, dx = \int_{C_j} S \, dx,$$

and then using fundamental theorem of calculus and Fubini’s theorem, we obtain

$$\int_{C_j} \partial_t U_j(x,t) \, dx = -\sum_{r=1}^{d} \left( F_r(U(x_{j+\frac{1}{2}}e_r,t)) - F_r(U(x_{j-\frac{1}{2}}e_r,t)) \right) + \int_{C_j} S(x,t) \, dx.$$

Dividing both sides by $|C_j|$ and denoting fluxes in the direction $r$ (associated with the directional flux $F_r$) by

$$F_{j+\frac{1}{2}e_r}^{\Delta x}(t) := F_r(U(x_{j+\frac{1}{2}}e_r,t)), \quad r = 1, \ldots, d,$$

we obtain a semi-discrete finite volume scheme for approximating (1.1),

$$\partial_t U_j(t) = -\frac{1}{|C_j|} \sum_{r=1}^{d} \left( F_{j+\frac{1}{2}e_r}^{\Delta x}(t) - F_{j-\frac{1}{2}e_r}^{\Delta x}(t) \right) + S_j(t).$$

This is exactly the statement of conservation: the rate of change of cell averages is given by the difference of the fluxes across the cell boundaries. Note that $F_{j+\frac{1}{2}e_r}(t)$ as defined in (1.48) requires a-priori knowledge about solution; namely, the trace values at cell boundaries $U(x_{j+\frac{1}{2}}e_r,t)$ are needed for all times $t > 0$. The main goal of the FV scheme is to approximate these fluxes.
1.4 Finite Volume method

Godunov [46] came up with an idea to approximate (1.48). Since we assume that for a fixed intermediate time \( t_0 \geq 0 \) the solution \( U(x, t_0) \) is approximated by cell averages \( U_j(t_0) \) and hence is constant in each cell \( C_j \), for each direction \( x_r \) with \( r \in \{1, \ldots, d\} \) and at each interface \( x_{j+\frac{1}{2}} \), the semi-discrete formulation (1.49) is approximated (in the \( x_r \) direction) by a one-dimensional Riemann problem for \( t \geq t_0 \):

\[
\begin{cases}
\tilde{U}_t + \frac{\partial}{\partial x_r} F_r(\tilde{U}) = 0, \\
\tilde{U}(x_r, t_0) = \begin{cases} U_j(t_0) & \text{if } x_r \leq x_{j+\frac{1}{2}}^r, \\
U_{j+1}(t_0) & \text{if } x_r > x_{j+\frac{1}{2}}^r. \end{cases}
\end{cases}
\tag{1.50}
\]

Hence the solution \( \tilde{U}(x_r, t) \) for \( t \geq t_0 \) can be approximated by the superposition of solutions \( \tilde{U}(x_r, t) \) of Riemann problems (1.50) at each interface \( x_{j+\frac{1}{2}}^r \). The speed of propagation is finite and is bounded by the maximal eigenvalue \( \lambda_{\text{max}} \) from (1.13); this enables us to impose the Courant-Friedrichs-Lewy (CFL) condition,

\[
\max_j |\lambda_{\text{max}}(x_j, U_j(t_0))| \frac{\Delta t}{\Delta x} \leq \frac{1}{2} \tag{1.51}
\]

in order to limit the time step \( \Delta t \) and ensure that waves from (1.50) are not intersecting for \( t < t_0 + \Delta t \). This procedure is repeated iteratively by replacing \( t_0 \) with \( t_0 + \Delta t \). We remark that the CFL condition (1.51) can be independently derived as a necessary and sufficient condition to conserve energy in the system [66].

Since the solutions of each Riemann problem (1.50) are self-similar [66], i.e.

\[
\tilde{U}(x_r, t) = \tilde{U}\left(\frac{x_r - x_{j+\frac{1}{2}}^r}{t - t_0}\right),
\]

the flux across the cell interface \( x_r = x_{j+\frac{1}{2}}^r \) is constant,

\[
F_r(\tilde{U}(x_{j+\frac{1}{2}}^r, t)) = F_r(\tilde{U}(x_{j+\frac{1}{2}}^r, t_0)) = F_r(\tilde{U}(x_{j+\frac{1}{2}}^r)).
\]

The flux \( F_r(\tilde{U}(x_{j+\frac{1}{2}}^r)) \) is well-defined since if \( \tilde{U}_{j+\frac{1}{2}}^r(\xi) \) is discontinuous at \( \xi = 0 \) for \( t > t_0 \), then the Rankine-Hugoniot condition (1.23) for zero shock speed implies that

\[
\lim_{\varepsilon \to 0} F_r(\tilde{U}(x_{j+\frac{1}{2}}^r - \varepsilon)) = \lim_{\varepsilon \to 0} F_r(\tilde{U}(x_{j+\frac{1}{2}}^r + \varepsilon)).
\]

We use \( F_r(\tilde{U}(x_{j+\frac{1}{2}}^r)) \) to define the approximation of \( F_{j+\frac{1}{2}e_r}(t) \) in (1.48),

\[
F_{j+\frac{1}{2}e_r}(t) := F(\tilde{U}(x_{j+\frac{1}{2}}^r)), \quad \text{for } t \in [t_0, t_0 + \Delta t]. \tag{1.52}
\]
Then (1.49) together with Riemann problem fluxes (1.52) gives the standard form of a finite volume scheme for conservation laws:

$$\partial_t U_j(t) = -\frac{1}{|C_j|} \sum_{r=1}^{d} \left( F_{j+\frac{1}{2}e_r}(t) - F_{j-\frac{1}{2}e_r}(t) \right) + S_j(t).$$

(1.53)

The remaining ingredient (apart from the time integration) in (1.53) is the quantification of fluxes $F_{j+\frac{1}{2}e_r} := F(\bar{U}(\xi_{j+\frac{1}{2}}))$, where we dropped the notation for time dependence.

For scalar conservation laws (i.e. $m = 1$) and linear and some non-linear one-dimensional systems ($d = 1$, $m > 1$) of conservation laws, Riemann problems (1.50) can be solved exactly, leading to the Godunov flux [66], which, for $m = 1$, is given by

$$F_{j+\frac{1}{2}e_r}^{\text{God}}(U_L, U_R) = \begin{cases} 
\min_{U_L \leq \xi \leq U_R} F_r(\xi) & \text{if } U_L \leq U_R, \\
\max_{U_R \leq \xi \leq U_L} F_r(\xi) & \text{if } U_L > U_R,
\end{cases}$$

(1.54)

where left and right states are denoted by

$$U_L = U_j(t_0), \quad U_R = U_{j+e_r}(t_0).$$

The resulting finite volume scheme with the Godunov flux (1.54) is termed the Godunov scheme. Notice that the implementation of the Godunov flux involves an optimization problem which can be expensive to solve depending on the structure of the flux $F_r$.

Computing exact solutions to (1.50) just to obtain the flux values at the cell interfaces seems too excessive, especially since (1.50) already provides only an approximation of (1.1). Furthermore, an explicit Godunov flux is no longer available for non-linear systems of conservation laws as exact solutions to Riemann problems (1.50) are not known in general.

Since we are already approximating the exact solution $U(x, t)$ by cell averages $U_j(t)$, fluxes $F_{j+\frac{1}{2}e_r}(t)$ could also be estimated by solving Riemann problems (1.50) approximately; this results in the class of finite volume schemes called approximate Riemann solvers (ARS), presented in the next section.

### 1.4.2 Two-wave approximate Riemann solvers

We will consider the so-called HLL (Harten, Lax and van Leer [56]) two wave solvers, where the solution of (1.50) is approximated by two shock waves propagating at the corresponding wave speeds $s_L = s_{j+\frac{1}{2}}^L$ and $s_R = s_{j+\frac{1}{2}}^R$, where we denote $\xi = x_{j+1} - x_{j+\frac{1}{2}}$,

$$\bar{U}(x_{j+\frac{1}{2}}, t) = \begin{cases} 
U_L & \text{if } s_L \geq \frac{\xi}{t-t_0}, \\
U^* & \text{if } s_L < \frac{\xi}{t-t_0} \leq s_R, \\
U_R & \text{if } s_R < \frac{\xi}{t-t_0},
\end{cases}$$

(1.55)
Notice that the middle state \( U^* = U^*_{j+\frac{1}{2}e_r} \) as well as its flux value 

\[
F^* = F^*_{j+\frac{1}{2}e_r} \approx F_r(U^*_{j+\frac{1}{2}e_r})
\]

are not known and equality is not required. Applying the Rankine-Hugoniot condition \((1.23)\) across the shocks given by the speeds \( s_L, s_R \), we obtain the following system:

\[
\begin{align*}
F_L - F^* &= s_L(U_L - U^*), \\
F_R - F^* &= s_R(U_R - U^*),
\end{align*}
\]

where \( F_R := F_r(U_R) \) and \( F_L := F_r(U_L) \) denote the left and the right fluxes.

The system \((1.56)\) can be explicitly solved for \( U^* \) and \( F^* \), obtaining the approximation

\[
F^* = \frac{s_R F_L - s_L F_R + s_R s_L (U_R - U_L)}{s_R - s_L}.
\]

Note that the intermediate flux \( F^* \) in \((1.57)\) can be simplified for opposite wave speeds \( s_L = -s_R \),

\[
F^* = \frac{F_L + F_R}{2} - \frac{s_R}{2} (U_R - U_L).
\]

**Lax-Friedrichs flux**

One of the most straightforward choices for \( s_L \) and \( s_R \) in \((1.57)\) is to consider the maximum wave speeds \( s_L \) and \( s_R \) allowed by the CFL condition \((1.51)\):

\[
s^R_{j+\frac{1}{2}} = \Delta x \Delta t, \quad s_L = -s_R.
\]

The resulting approximation of the flux \( F_{j+\frac{1}{2}e_r} \) is obtained by using \((1.57)\) - \((1.58)\) and is called the Lax-Friedrichs flux,

\[
F^\text{LxF}_{j+\frac{1}{2}e_r}(U_L, U_R) := \frac{F_r(U_L) + F_r(U_R)}{2} - \frac{\Delta x}{2\Delta t} (U_R - U_L).
\]
Introduction

Rusanov flux

The Lax-Friedrichs flux is rather diffusive \[97\]; however, it can be improved by choosing locally adapted wave speeds \(s_L, s_R\) at each cell interface:

\[
s_{j+\frac{1}{2}}^R = \max(|\lambda_{\text{max}}(U_L)|, |\lambda_{\text{max}}(U_R)|), \quad s_{j+\frac{1}{2}}^L = -s_{j+\frac{1}{2}}^R.
\]

The resulting approximation of the flux \(F_{j+\frac{1}{2}e_r}\) is called the Rusanov flux and is given by

\[
F_{j+\frac{1}{2}e_r}^{\text{Rus}}(U_L, U_R) = \frac{F_r(U_L) + F_r(U_R)}{2} - \frac{\max(|\lambda_{\text{max}}(U_L)|, |\lambda_{\text{max}}(U_R)|)}{2} (U_R - U_L).
\]

Remark: different choices of wave speeds \(s_L\) and \(s_R\) result in various different fluxes: Enquist-Osher \[85\], etc.

1.4.3 Three-wave approximate Riemann solvers

One difficulty of the HLL solvers introduced in subsection 1.4.2 is the assumption of the two wave configuration in the Riemann problem solutions. This is no longer valid for systems of conservation laws; multiple rarefactions and shocks are possible. For instance, in one-dimensional \((d=1)\) Euler equations \[1.8\], an initial shock splits into a rarefaction, a shock wave and a contact discontinuity \[109\]. The HLL solver discussed in subsection 1.4.2 replaces the entire middle structure of the wave (in this case, the contact discontinuity) by a constant average value. Hence, approximations of contact surfaces, shear waves and material interfaces can be very inaccurate. As a remedy, Toro, Spruce and Speares \[109\] introduced the HLLC (C stands for contact) solver, which approximates the solution to \(1.50\) with three waves instead of two; the middle wave recovers the structural properties (such as the contact wave) between the fastest left and right going waves.

Analogously to \(1.55\), we begin with the three wave approximation of the solution to \(1.50\); this time with an additional intermediate wave speed \(s_M = s_{j+\frac{1}{2}}^M\), resulting in two intermediate states \(U_L^* = U_{j+\frac{1}{2}e_r}^L\) and \(U_R^* = U_{j+\frac{1}{2}e_r}^R\), where we denote \(\xi = x_r - x_{j+\frac{1}{2}}\),

\[
U(x_r, t) = \begin{cases} 
U_L & \text{if } s_L \geq \frac{\xi}{t-t_0}, \\
U_L^* & \text{if } s_L < \frac{\xi}{t-t_0} \leq s_M, \\
U_R^* & \text{if } s_M < \frac{\xi}{t-t_0} \leq s_R, \\
U_R & \text{if } s_R < \frac{\xi}{t-t_0}.
\end{cases}
\]

(1.62)

In the case of Euler equations \[1.8\], the intermediate wave \(s_M\) in \(1.62\) corresponds to the intermediate eigenvalue of multiplicity equal to the spatial dimension \(d\), implying
\[ \lambda_2 = \lambda_3 = \lambda_4 = \mathbf{u}_s \text{ in the full } d = 3 \text{ case. The Rankine-Hugoniot condition} \text{ (1.23) for shocks across each of the waves } s_L, s_M, s_R \text{ gives the following system of equations,} \]

\[
\begin{align*}
F_L - F_L^* &= s_L(U_L - U_L^*), \\
F_L^* - F_R^* &= s_M(U_M^* - U_R^*), \\
F_R^* - F_R &= s_R(U_R^* - U_R),
\end{align*}
\]

where \( U_L^*, U_R^* \) are intermediate states with the corresponding flux values

\[ F_L^* = F_{Lj+\frac{1}{2}e_r}, \approx F_s(U_{Lj+\frac{1}{2}e_r}^*), \quad F_R^* = F_{Rj+\frac{1}{2}e_r}, \approx F_s(U_{Rj+\frac{1}{2}e_r}^*). \]

The system (1.63) is under-determined since there are four unknown vectors \( U_L^*, U_R^*, F_L^*, F_R^* \), and only three equations, provided that the method to obtain \( s_M \) from given \( s_L \) and \( s_R \) is available. Explicit values for \( U_L^*, U_R^*, F_L^*, F_R^* \) depend on the underlying system of conservation laws; again, we look at the particular case of Euler equations (1.8) and introduce further constraints on \( U_L^*, U_R^*, F_L^*, F_R^* \). The pressure \( p \) and the normal velocity component \( u = \mathbf{u}_n \) are constant across the middle shock, whereas the density is discontinuous; furthermore, the velocity components \( \mathbf{u}_e \) in the remaining directions \( r' \neq r \) coincide with the velocity components at the left-most and right-most states,

\[
\begin{align*}
u_L^* &= u_R^* = u^*, \\
p_L^* &= p_R = p^*, \\
(u_L^*)_{r'} &= (u_R)_{r'}, \quad r' \neq r, \\
(u_L^*)_{r'} &= (u_R)_{r'}, \quad r' \neq r,
\end{align*}
\]

where

\[ \mathbf{U}^* = (\rho^*, \rho^* \mathbf{u}^*, E(\rho^*)). \]

Note that (1.64) is satisfied by the exact solution [109].

The fastest wave speeds \( s_L \) and \( s_R \) can be estimated analogously as in the HLL setting in subsection 1.4.2. The intermediate wave speed \( s_M \) is derived from (1.64),

\[
s_M = \frac{p_R - p_L + \rho_L u_L (s_L - u_L) - \rho_R u_R (s_R - u_R)}{\rho_L (s_L - u_L) - \rho_R (s_R - u_R)}. \]

It is entirely justified [109] (and convenient) to set:

\[ u^* = s_M. \]

Application of (1.64) - (1.65) to (1.63) results in the following estimate for the intermediate vectors \( U_L^*, U_R^* \), where \( \tilde{u}_s^* \) is set by (1.64) and (1.65),

\[ U_s^* = \rho_s \left( \frac{s_s - u_s}{s_s - s_M} \right) \left[ \begin{array}{c} 1 \\
E(\rho_s) + (s_M - u_s) \left( s_M + \frac{p_s}{\rho_s (s_s - u_s)} \right) \end{array} \right], \quad \ast \in \{L, R\}, \]
Introduction

and intermediate fluxes

\[ F^*_\star = F_\star + s_\star(U^*_\star - U_\star), \quad \star \in \{L, R\}. \]

The resulting HLLC (HLL3) approximate flux is given as follows,

\[ F_{j+\frac{1}{2}e_r} = F_{j+\frac{1}{2}e_r}^{\text{HLLC}}(U_L, U_R) = \begin{cases} F_L & \text{if } 0 \leq s_L, \\ F_L + s_L(U^*_L - U_L) & \text{if } s_L < 0 \leq s_M, \\ F_R + s_R(U^*_R - U_R) & \text{if } s_M < 0 < s_R, \\ F_R & \text{if } s_R \leq 0. \end{cases} \] (1.66)

For MHD equations (1.10), several versions of three wave HLL3 solvers, including Linde HLL3L and relaxed HLL3R solvers, and a five wave HLL5 solver are described in detail in [41]. HLL solvers must be further modified to incorporate the Godunov-Powell source term arising in the non-conservative form of the MHD equations.

1.4.4 General framework for high order schemes

The FVM schemes presented in section 1.4.1 with two or three wave HLL solvers are only first order accurate [66], irrespective of the order of time integration method from (1.94) - (1.96). This means that in order to achieve small errors in the approximated solution \( U_T \), a very large number of cells \( C_j \) might be needed.

To address this issue, higher order reconstruction can be obtained by the following generalization of the Godunov method (1.53),

1. Reconstruction. Use cell averages \( U^n_j \) to reconstruct the initial data \( \overline{U}(x_r, t_0) \) of (1.50) with higher order piecewise (cell-wise) polynomials \( p^r(x) : C^r_j \rightarrow \mathbb{R}^m \) instead of piecewise constant values; the cell averages \( C^r_j \) must remain preserved:

\[ \frac{1}{|C^r_j|} \int_{C^r_j} p^r(x) \, dx = U^n_j. \] (1.67)

2. Solution traces at the interface. Compute traces of \( p^r(x) \) at interfaces \( x^r_{j+\frac{1}{2}} \),

\[ U^-_{j+\frac{1}{2}e_r}(t^n) := \lim_{\varepsilon \to 0} p^r(x^r_{j+\frac{1}{2}} - \varepsilon), \quad U^+_{j+\frac{1}{2}e_r}(t^n) := \lim_{\varepsilon \to 0} p^r(x^r_{j+\frac{1}{2}} + \varepsilon). \] (1.68)

3. Flux approximations. Use any flux from subsection 1.4.2 together with the more accurate flux values at the interface \( x^r_{j+\frac{1}{2}} \),

\[ U_L = U^-_{j+\frac{1}{2}}(t^n), \quad U_R = U^+_{j+\frac{1}{2}}(t^n), \quad F_{j+\frac{1}{2}e_r} = F_{j+\frac{1}{2}e_r}(U^-_{j+\frac{1}{2}}, U^+_{j+\frac{1}{2}}). \] (1.69)
1.4 Finite Volume method

1.4.5 Piecewise linear TVD reconstruction

Here we elaborate on each of the steps \(1.67\) - \(1.69\) for the piecewise linear reconstruction. General, higher order polynomial reconstructions will be considered in subsequent sections.

For convenience, we drop the notation for the time \(t_0\):

\[
\bar{U}(x) = U(x_r) = \bar{U}(x_r, t_0), \quad U_j = U^n_j.
\]

Reconstruction. For a given direction \(r = 1, \ldots, d\), denote the linear \(\mathbb{R}^m\)-valued polynomial in the cell \(C_r^j \subset \mathbb{R}\) by \(p^r_j(x)\); then \(1.67\) enforces \(p^r_j(x)\) to be of the form

\[
p^r_j(x) = U_j + \sigma^r_j(x - x^r_j), \tag{1.70}
\]

where \(\sigma^r_j\) denotes the slopes of each component of \(p^r_j(x)\). The local linear reconstructions \(p^r_j\) are then combined into global piecewise linear \(\mathbb{R}^m\)-valued polynomial of degree 1,

\[
p^r(x) = p^r_j(x), \quad \text{for } x \in C^j_r \subset I_r \subset \mathbb{R}. \tag{1.71}
\]

Solution traces. Values for \(U^{\pm}_{j+\frac{1}{2}e_r}\) and \(U^{\pm}_{j-\frac{1}{2}e_r}\) in \(1.68\) can be obtained from \(1.71\),

\[
U^{\pm}_{j+\frac{1}{2}e_r} = p^r_j \left( x^r_{j+\frac{1}{2}} \right) = U_j + \frac{\sigma^r_j \Delta x}{2},
\]

\[
U^{\pm}_{j-\frac{1}{2}e_r} = p^r_{j+1} \left( x^r_{j+\frac{1}{2}} \right) = U_{j+1} - \frac{\sigma^r_{j+1} \Delta x}{2}.
\]

Flux approximations. A second order accurate flux is obtained using the above traces,

\[
F_{j+\frac{1}{2}e_r} = F_{j+\frac{1}{2}e_r} \left( U^{\pm}_{j+\frac{1}{2}e_r}, U^{\pm}_{j-\frac{1}{2}e_r} \right).
\]

The only remaining ingredient is the choice of the slopes \(\sigma^r_j\).

Standard methods, including left, right and central differences (see Definition 1.4.2 ahead), in general will produce oscillatory and numerically unstable solutions near shocks; this is due to the fact that the reconstruction step is not TVD (total variation diminishing, i.e. \(1.38\) holds [66]). Next we discuss the so-called slope limiters that preserve the TVD property of the numerical solution and thus attain numerical stability of the scheme.
Introduction

TVD slope limiters

Ensuring the TVD property of the reconstruction step would ensure the TVD property of the scheme as well. An example of such a limiter is called the MINMOD limiter and is defined as follows.

**Definition 1.4.1 (MINMOD slope limiter).**

\[
\sigma^r_j = \minmod(\sigma^r_L, \sigma^r_R),
\]

(1.72)

where \( \sigma^r_L \) and \( \sigma^r_R \) are left and right differences (as in Definition 1.4.2 ahead) in the \( r \)-direction and the function \( \minmod \) is defined (component-wise for vectors) as [66, 49]

\[
\minmod(a, b) = \begin{cases} 
\text{sign}(a) \min(|a|, |b|) & \text{if} \quad \text{sign}(a) = \text{sign}(b), \\
0 & \text{otherwise}.
\end{cases}
\]

(1.73)

Alternative slope limiters are available, for instance, monotonized central (MC), superbee, van Leer, and many more; all these limiters ensure the TVD property, see [66, 49].

1.4.6 Essentially non-oscillatory reconstruction

One disadvantage of TVD schemes based on slope limiters is that the accuracy necessarily degenerates to first order near smooth extrema [49], and not only in the vicinity of shocks (the latter can not be prevented). Motivated by this, we next look at the so-called essentially non-oscillatory high order schemes; these schemes are not TVD, but the oscillations are small and vanish as the mesh is refined, hence the scheme ends up being numerically stable [97].

Preliminaries

In order to reconstruct \( \bar{U}(x) \) in (1.50) from its cell averages \( U_j \), we extend the reconstruction (1.70) with the approximating \( k \)-th order piecewise polynomial \( p^r(x) \),

\[
p^r(x) = p^r_j(x), \quad \forall x \in C^r_j \subset I_r \subset \mathbb{R}, \quad r = 1, \ldots, d,
\]

(1.74)

where for every cell \( C^r_j \), the \( \mathbb{R}^m \)-valued polynomials \( p^r_j \) are of order \( k \) (or, equivalently, \( \deg p^r_j \leq k - 1 \)),

\[
p^r_j(x) = \bar{U}(x) + \mathcal{O}(\Delta x^k).
\]

The approximation (1.74) provides approximations for trace values (1.68) at interfaces \( x^r_{j+\frac{1}{2}} \),

\[
U^-_{j+\frac{1}{2}e_r} = p^r_j(x^r_{j+\frac{1}{2}}), \quad U^+_{j+\frac{1}{2}e_r} = p^r_{j+1}(x^{r+1}_{j+\frac{1}{2}}),
\]

(1.75)
1.4 Finite Volume method

which for sufficiently smooth $\bar{U}(x)$ are also $k$-th order accurate:

$$
\begin{align*}
U_{j+\frac{1}{2}e_r}^- &= \lim_{\varepsilon \to 0} U\left(x_{j+\frac{1}{2}e_r}^- - \varepsilon, t^n\right) + \mathcal{O}(\Delta x^k), \\
U_{j+\frac{1}{2}e_r}^+ &= \lim_{\varepsilon \to 0} U\left(x_{j+\frac{1}{2}e_r}^+ + \varepsilon, t^n\right) + \mathcal{O}(\Delta x^k).
\end{align*}
$$

(1.76)

For each cell $C^r_j$ and a given order of accuracy $k$, we need to find a stencil consisting of the central cell $C^r_j$, $l$ cells to the left and $g$ cells to the right; we denote such stencil as

$$
S^r_j := \{j-l, \ldots, j+g\} \equiv \{C^r_{j-l}, \ldots, C^r_{j+g}\} \subset T^r.
$$

(1.77)

Denote the unique polynomials

$$
p^r_j(x), \quad \deg p^r_j \leq k-1,
$$

(1.78)

possessing consistent cell averages

$$
\frac{1}{|C^r_i|} \int_{C^r_i} p^r_j(\xi) \, d\xi = U_j, \quad \forall i \in S^r_j.
$$

Then $p^r_j(x)$ is $k$-th order accurate approximation as in (1.74). Approximations (1.68) of $U(x_r, t^n)$ at the cell interfaces are computed as in (1.75),

$$
U_{j+\frac{1}{2}e_r}^- = p^r_j\left(x_{j+\frac{1}{2}}^r\right), \quad U_{j+\frac{1}{2}e_r}^+ = p^r_{j+1}\left(x_{j+\frac{1}{2}}^r\right).
$$

(1.79)

Alternatively, since for all $j$ the dependence of $U^-_{j+\frac{1}{2}e_r}$ and $U^+_{j+\frac{1}{2}e_r}$ on cell averages $\{U_j\}$ is linear, there exist constants $c^r_{l,i}, c^r_{l,i} \in \mathbb{R}$ (here we are dropping the notation for dependence on the direction $r$) depending on the left shift $l$ of the stencil $S^r_j$ in (1.77), on $k$, and on $\Delta x$ but not on the $U_j$’s, such that

$$
U^-_{j+\frac{1}{2}e_r} = \sum_{i=0}^{k-1} c^r_{l,i} U_{j-(l+i)e_r}, \quad U^+_{j+\frac{1}{2}e_r} = \sum_{i=0}^{k-1} c^r_{l,i} U_{j-(l+i)e_r}
$$

(1.80)

are $k$-th order accurate, i.e. the approximation property (1.76) holds. Notice that

$$
c^r_{l,i} = c^r_{l-1,i},
$$

hence, for convenience, we denote $c_{l,i} := c^r_{l,i}$.

Values for the constants $c_{l,i}$ up to $k = 3$ are given in [Table 1.1] a rigorous derivation of these constants using the Lagrange form of the interpolation polynomial together with an extended list up to $k = 7$ and the proof of the accuracy property (1.76) for smooth $U(x_r, t^n)$ can be found in [97].

Since the function $U(x_r, t^n)$ that we aim to approximate is only piecewise smooth and hence might contain finitely many discontinuities, the order of accuracy that is referred
Introduction

Table 1.1: Constants $c_{l,i}$ in (1.80)

<table>
<thead>
<tr>
<th>$k$</th>
<th>1</th>
<th>$i=0$</th>
<th>$i=1$</th>
<th>$i=2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>-1</td>
<td>3/2</td>
<td>-1/2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>1/2</td>
<td>1/2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>-1/2</td>
<td>3/2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>-1</td>
<td>11/6</td>
<td>-7/6</td>
<td>1/3</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>1/3</td>
<td>5/6</td>
<td>-1/6</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>-1/6</td>
<td>5/6</td>
<td>1/3</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1/3</td>
<td>-7/6</td>
<td>11/6</td>
</tr>
</tbody>
</table>

to throughout the text is only formal, i.e. it is determined by local truncation error in the smooth regions of $U(x_r, t^n)$.

Furthermore, if stencils $S^r_j$ are fixed, i.e. the left shift $l$ is fixed for every $j$, then the reconstruction may be inadequate near these discontinuities; oscillations occur if stencils include the discontinuous cell since then the approximation property (1.76) is no longer valid [97]. The last issue motivates us to choose an “adapted stencil” which does not include the discontinuous cells in order to suppress these oscillations. Such reconstructions are called ENO (essential non-oscillatory) and are described in detail in the next section.

Essentially Non-Oscillatory scheme

The ENO idea proposed in [55] was the first successful attempt for a self-similar, uniformly high order accurate and still essentially non-oscillatory interpolation for piecewise smooth functions. By essentially non-oscillatory we mean that (for smooth functions) oscillations decay as $O(\Delta x^k)$ for $k$-th order accurate reconstruction.

In order to choose a stencil $S^r_j = \{j-l, \ldots, j+g\}$, divided differences are used to estimate the smoothness of $\bar{U}(x)$ or $p^r(x)$ inside the corresponding cells $C^r_{j-l}, \ldots, C^r_{j+g}$.

**Definition 1.4.2 (Divided differences).** The first order divided difference (in the direction $r$) of cell averages $U_j$ is defined as

$$U_{[j]} := U_{j_r},$$

and in general, for arbitrary $k \in \mathbb{N}$, the $k$-th order divided differences are defined as

$$U_{[j, \ldots, j+i]} := \frac{U_{[j+1, \ldots, j+i]} - U_{[j, \ldots, j+i-1]}}{i \Delta x}.$$  

(1.81)
1.4 Finite Volume method

The key property of the divided differences (1.81) is that

\[ u_{[j, ..., j+i]} = \begin{cases} \frac{\partial^i u(\xi)}{\partial x^i} & \text{for some } \xi \in (x_{j-1/2}^r, x_{j+i+1/2}^r) \text{ if } u(x) \text{ is smooth}, \\ O(\Delta x^{-i}) & \text{if } u(x) \text{ is discontinuous inside } (x_{j-1/2}^r, x_{j+i+1/2}^r). \end{cases} \]  

(1.82)

Hence, divided differences \( U_{[j-l, ..., j+g]} \) appear to be good smoothness indicators inside the stencils \( S_r^j = \{ j-l, ..., j+g \} \) (i.e. smaller divided difference implies smoother function) and hence can be used to construct an “adequate” locally adapted (depending on location \( j \)) stencil \( S_r^j \) containing \( k \) cells by the following recursive procedure:

1. **Begin with central cell.** Begin with the stencil \( S_r^j = \{ j \} \) containing the cell \( C_r^j \).

2. **Append “smoother” cell.** Given the stencil \( S_r^j = \{ j-l, ..., j+g \} \), there are two choices: append cell \( j-l-1 \) or \( j+g+1 \) to the existing stencil \( S_r^j \); motivated by (1.82), we pick the one with the smaller divided difference corresponding to the smoother reconstruction polynomial \( p_r^j \).

3. **Repeat Step (2) until the stencil is complete.** Keep iterating step (2) until the entire stencil \( S_r^j \) of required size \( k \) is achieved, i.e. until \( l+g+1 = k \).

Since the mesh was assumed to be uniform, i.e. cell sizes \( \Delta x \) are constant for all cells \( C_r^j \), the denominator \( \Delta x \) in (1.81) can be removed resulting in the so-called undivided differences [97], this way increasing the computational efficiency of the procedure.

Once the stencil is computed, the representation (1.80) can be used together with Table 1.1 to obtain the \( k \)-th order accurate trace values \( U_{j+1/2e_r}^\pm \) required in (1.75).

Properties of ENO reconstruction (for proofs, we refer to [97]):

1. **Uniform accuracy.** The accuracy property (1.76) holds in every cell \( C_r^j \) that does not contain any discontinuity; the ENO interpolation procedure recovers full high order accuracy and does not deteriorate near local extrema as TVD reconstructions discussed in subsection 1.4.5.

2. **Essentially non-oscillatory.** For smooth functions \( \hat{U} \), the ENO reconstruction is TVB (total variation bounded), i.e. \( \exists z : \mathbb{R} \rightarrow \mathbb{R} \):

\[ z(x) = p_r^j(x) + O(\Delta x^k), \quad \forall x \in C_r^j \quad \text{and} \quad TV(z) \leq TV(\hat{U}). \]  

(1.83)

3. **Sign property.** The sign property states (we refer to [39, 40]) that at each cell interface \( x_{j+1/2e_r} \), the jump in reconstructed values \( U_{j+1/2e_r}^\pm \) has the same sign as the jump in the corresponding cell averages, i.e.

\[ \text{sign}(U_{j+1/2e_r}^+ - U_{j+1/2e_r}^-) = \text{sign}(U_{j+e_r} - U_j). \]
Furthermore, the following bound holds [39, 40],

\[
\frac{U^+_{j+\frac{1}{2}e_r} - U^-_{j+\frac{1}{2}e_r}}{U_{j+e_r} - U_j} \leq C_k,
\]

with constant \(C_k\) depending only on \(k\).

**Weighted Essentially Non-Oscillatory scheme**

*Motivation.* Despite being uniformly high-order and essentially non-oscillatory due to locally adapted stencils, the ENO reconstruction still has several drawbacks:

1. In the regions of vanishing \(\bar{U}\) and \(\partial_x \bar{U}\), the stencil might change rapidly due to round-off errors caused by subtractive cancellation in differences (1.81), see [54, 99].

2. The resulting flux (1.69) is not smooth in general; stencils \(S^r_j\) and \(S^r_{j+1}\) corresponding to the cell interface \(x^r_{j+\frac{1}{2}}\) might have different values for the left shift \(l\).

3. The choice of \(k\) cells for the stencil out of \(2^k-1\) candidate cells results in \(k\)-th order accuracy; however, incorporating all \(2^k-1\) candidate stencils could potentially lead to \((2k-1)\)-th order accuracy in smooth regions.

4. The ENO reconstruction step (2) involves many logical “if” statements which are not very efficient on certain vector computers such as CRAYs.

We point out that there have been attempts in the literature to remedy the first problem by the “biasing” strategy leading to “preferring” some stencils over the remaining ones, see [97] and references therein.

In this section we describe the recently developed weighted ENO (WENO) reconstruction [70, 58]. WENO reconstruction attempts to improve the four bottlenecks described above. The key idea is simple: contrary to the original ENO scheme where only one particular (smoothest) stencil is picked, WENO scheme incorporates all candidate stencils

\[
S^r_{j,l} := \{j - l, \ldots, j + g\}, \quad \forall l \in 0, \ldots, k - 1,
\]

each stencil producing a different \((k\text{-th order accurate})\) reconstruction of \(U^\pm_{j+\frac{1}{2}e_r}\),

\[
U^\pm_{j+\frac{1}{2}e_r, l} := \sum_{i=0}^{k-1} c^\pm_i \ U_{j-(l+i)e_r}, \quad \forall l \in 0, \ldots, k - 1.
\]

Then, a *convex combination* of all such reconstructions is formed

\[
U^\pm_{j+\frac{1}{2}e_r} = \sum_{l=0}^{k-1} w^\pm_l \ U^\pm_{j+\frac{1}{2}e_r, l}, \quad (1.84)
\]
1.4 Finite Volume method

<table>
<thead>
<tr>
<th></th>
<th>l=0</th>
<th>l=1</th>
<th>l=2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>2/3</td>
<td>1/3</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>3/10</td>
<td>3/5</td>
<td>1/10</td>
</tr>
</tbody>
</table>

Table 1.2: Constants $d_l$ in (1.86)

where the choice of appropriate weights $w_l^\pm$ is the key ingredient.

Firstly, to ensure convexity, we require

$$w_l^\pm \geq 0, \quad \sum_{l=0}^{k-1} w_l^\pm = 1.$$  \hspace{1cm} (1.85)

For smooth $\bar{U}(x)$ inside all the candidate stencils $S_{r,j}^r$, there exist constants $d_l = d_l^-$ and (by symmetry) $d_l^+ = d_{k-1-l}^-$ such that (we refer to [97] for the proof):

$$U_{j+\frac{1}{2}e_r}^- = \sum_{l=0}^{k-1} d_l^- U_{j+\frac{1}{2}e_r,l}^- = U(x_{j+\frac{1}{2}e_r,t^n}) + O(\Delta x^{2k-1}),$$  \hspace{1cm} (1.86)

$$U_{j+\frac{1}{2}e_r}^+ = \sum_{l=0}^{k-1} d_l^+ U_{j+\frac{1}{2}e_r,l}^+ = U(x_{j+\frac{1}{2}e_r,t^n}) + O(\Delta x^{2k-1}).$$

We provide these constants up to $k = 3$ in Table 1.2.

Secondly, for smooth $\bar{U}(x)$, we require that

$$w_l^\pm = d_l^\pm + O(\Delta x^{k-1}),$$  \hspace{1cm} (1.87)

since then the reconstruction (1.84) maintains the order of accuracy $2k-1$ from (1.86).

In addition to requirements (1.85) and (1.87), the weights $w_l$ shall be smooth functions of $U_j$ and the resulting scheme is required to be computationally efficient (polynomials, rational functions). All these considerations lead to the following form of weights [55],

$$w_l^\pm = \frac{\alpha_l}{\sum_{i=0}^{k-1} \alpha_i}, \quad l = 0, \ldots, k-1,$$  \hspace{1cm} (1.88)

where

$$\alpha_l = \frac{d_l^\pm}{(\varepsilon + \beta_l)^2}.$$  \hspace{1cm} (1.89)

Here, $\varepsilon > 0$ prevents vanishing denominators and is usually set to $10^{-6}$. The so-called “smoothness indicators” $\beta_l$ of the stencil $S_{r,j}^r$ are assumed to satisfy

$$\beta_l = \begin{cases} O(\Delta x^2) & \text{if } \bar{U}(x) \text{ is smooth (i.e. } C^1) \text{ inside } S_{r,j}^r, \\ O(1) & \text{if } \bar{U}(x) \text{ contains a discontinuity inside } S_{r,j}^r. \end{cases}$$  \hspace{1cm} (1.90)
Translating (1.90) into the weights \( w_l \) in (1.88), we obtain
\[
 w_l^\pm = \begin{cases} 
  O(1) & \text{if } \bar{U}(x) \text{ is smooth (i.e. } C^1 \text{) inside } S_{j,l}^r, \\
  O(\Delta x^4) & \text{if } \bar{U}(x) \text{ contains a discontinuity inside } S_{j,l}^r.
\end{cases}
\] (1.91)

Notice that by (1.91) most of the cumulative weight is concentrated on the smoothest stencils as \( \Delta x \to 0 \); this way WENO emulates ENO reconstruction near discontinuities. The requirement (1.85) is trivially satisfied by the definition of weights \( w_l^\pm \) in (1.88) and the accuracy requirement (1.87) is proven in [97].

The remaining ingredient are the explicit formulas for “smoothness indicators” \( \beta_l \). Consideration for a smooth flux and the role of higher order variations led to the following candidate [97]:
\[
 \beta_l = \sum_{i=1}^{k-1} \int_{c_j + \frac{1}{2}} \Delta x^{2i-1} \left( \frac{\partial p_{j,l}^r(x)}{\partial x^i} \right)^2 dx, \tag{1.92}
\]
where \( p_{j,l}^r(x) \) corresponds to reconstruction polynomials on the stencils \( S_{j,l}^r \). We note that (1.92) is based on the \( L^2 \)-norm and is smoother than TV measurement based on \( L^1 \) norm; additionally, (1.92) is considered more accurate for \( k = 2 \) and \( k = 3 \).

For \( k = 2 \), the smoothness indicator (1.92) becomes
\[
 \beta_0 = (U_{j+e} - U_j)^2, \\
 \beta_1 = (U_j - U_{j-e})^2. \tag{1.93}
\]

and the resulting WENO reconstruction (1.84) is third order accurate (compare to second order accuracy for ENO reconstruction, \( k = 2 \)). Choosing \( k = 3 \) yields a fifth order accurate scheme. The 7-th order schemes corresponding to \( k = 4 \) have not been extensively tested yet. Note that the accuracy for both ENO and WENO schemes is global (except at discontinuities) and does not deteriorate near local extrema [58].

We finish the review of high-resolution spatial reconstruction methods with a few closing remarks.

**Remark 1.4.3.** The analysis in [97] reveals that for high-order methods, the choice of numerical flux in (1.53) has less influence on the accuracy of the scheme. For first order accurate schemes, Godunov (1.54) and Rusanov (1.61) fluxes result in much less smearing of the solution when compared to Lax-Friedrichs flux (1.60); however, for 2nd order accurate schemes (for instance, ENO with SSP-RK2), this difference becomes much smaller.

**Remark 1.4.4.** For linear systems of conservation laws, the reconstruction of characteristic variables \( W \) instead of conserved variables \( U \) in general produces more accurate approximate solutions [97].
1.4 Finite Volume method

**Remark 1.4.5.** When applied to Euler or MHD equations, ENO and WENO reconstructions might produce negative pressures and densities; to ensure positivity preservation, the so-called “Perthame-Shu clipping” can be applied; refer to [41] for details on these modifications. We denote such modified reconstructions by ENOF and WENOF, respectively.

1.4.7 Time integration

The final ingredient that was missing so far in finite volume scheme (1.53) is the temporal discretization. To this end, we assemble all cell averages $U_j$ of $U$ in the collection \{U\}_T = \{U_j(t)\}_{C_j \in T}$ and rewrite (1.53) in the operator form,

$$\frac{\partial}{\partial t}\{U\}_T(t) = L(\{U\}_T(t)),$$

where the operator $L$ acts on each cell average $U_j(t)$ of $\{U\}_T(t)$:

$$L(U_j(t)) := -\frac{1}{|C_j|} \sum_{r=1}^{d} \left( F_{j+\frac{1}{2}e_r}(t) - F_{j-\frac{1}{2}e_r}(t) \right) + S_j(t).$$

We choose time snapshots $0 = t^0 < t^1 < \cdots < t^n$ with time step $\Delta t = t^i - t^{i-1}$ respecting the CFL condition (1.51) and define $U^n_j := U_j(t^n)$. The simplest time stepping scheme for temporal discretization of (1.53) is Forward Euler (FE):

$$\{U^{n+1}\} = \{U^n\} + \Delta tL(\{U^n\}).$$  \hspace{1cm} (1.94)

The approximation of the solution is given in terms of cell averages $\{U^n_j\}_{C_j \in T}$,

$$U^n_T(x) = U_T(x, t^n) = U^n_j, \quad \forall x \in C_j.$$

The Forward Euler time stepping scheme (1.94) is only first order accurate and hence would spoil the overall asymptotic convergence in the case of high resolution schemes described in subsection 1.4.5 and subsection 1.4.6. A class of higher order schemes are the strong stability preserving Runge-Kutta schemes (SSP-RK). Such second order accurate SSP-RK2 scheme is as follows [98],

$$\{U^n\} = \{U^n\} + \Delta tL(\{U^n\}),$$

$$\{U^{n+1}\} = \frac{1}{2}(\{U^n\} + \{U^{n+1}\}).$$  \hspace{1cm} (1.95)
Introduction

Likewise, the third order accurate scheme SSP-RK3 is given by [98]

\[
\begin{align*}
\{U^*\} &= \{U^n\} + \Delta t \mathcal{L}(\{U^n\}), \\
\{U^{**}\} &= \frac{1}{4}(\{U^*\} + \Delta t \mathcal{L}(\{U^*\})) + 3\{U^*\}, \\
\{U^{***}\} &= \{U^{**}\} + \Delta t \mathcal{L}(\{U^{**}\}), \\
\{U^{n+1}\} &= \frac{1}{3}(\{U^n\} + 2\{U^{***}\}).
\end{align*}
\]

(1.96)

SSP Runge-Kutta time stepping schemes of even higher order are available in [98].

1.4.8 Well-balanced solvers

For the shallow water equations (1.12), the admissible weak solutions satisfy the entropy inequality (in the sense of distributions),

\[
E(U)_t + \left( \frac{1}{2} (hu_1^3 + hu_1u_2^2) + gh u_1(h + b) \right)_{x_1} + \left( \frac{1}{2} (hu_2^3 + hu_2^2) + gh u_2(h + b) \right)_{x_2} \leq 0.
\]

(1.97)

where the total energy \(E(U)\) plays the role of the entropy function,

\[
E(U) = \frac{1}{2}(h||u||^2 + gh^2 + gh b).
\]

Furthermore, the maximal initial perturbation \(\Delta h_0\) in the water column in many problems of engineering interest is very small compared to \(h_0\). Hence, the numerical approximation of fluxes in the space semi-discrete form (1.49) needs special treatment; in particular, we required that the so-called \textit{discrete lake at rest} is preserved, i.e.

\[
u_j \equiv 0, \quad h_j + b_j \equiv \text{constant}, \quad \forall \mathcal{C}_j \in \mathcal{T}.
\]

(1.98)

Such FVM schemes, satisfying the discrete version of (1.97) and the condition (1.98), for all discretizations \(\mathcal{T}\) of \(D\), are called \textit{entropy stable and well-balanced}, respectively. In this section we introduce several examples of such schemes developed in a recent paper [38]. For simplicity, we present the schemes in one space dimension with notation \(u = u_1\).

We will use the following notation for the averages and jumps across cell interfaces,

\[
\bar{a}_{j+\frac{1}{2}} = \frac{a_j + a_{j+1}}{2}, \quad [[a]]_{j+\frac{1}{2}} = a_{j+1} - a_j.
\]

The scheme proposed in [38] is of the space semi-discrete form (1.49),

\[
\frac{\partial}{\partial t} U_j = - \frac{1}{\Delta x} \left( F^{\text{ES1}}_{j+\frac{1}{2}} - F^{\text{ES1}}_{j-\frac{1}{2}} \right) - \frac{g}{2 \Delta x} \begin{bmatrix} 0 \\ \hat{h}_{j+\frac{1}{2}} [[b]]_{j+\frac{1}{2}} + \hat{h}_{j-\frac{1}{2}} [[b]]_{j-\frac{1}{2}} \end{bmatrix},
\]

(1.99)
where the rightmost term is a second order accurate well-balanced discretization of the source term $S_j$. Note that in this example the source term $S = S(x, U)$ does not explicitly depend on time $t$, it only depends on the spatial location $x$ and the (time-dependent) conserved variables $U(x, t)$. For first-order schemes, the *entropy stable* numerical flux is given by

$$F_{ES1 j+\frac{1}{2}} = F_{EC j+\frac{1}{2}} - \frac{1}{2} D_{ES1 j+\frac{1}{2}} [[V]]_{j+\frac{1}{2}}, \tag{1.100}$$

where $V = \partial U E = [g(h+b) - \frac{u^2}{2}, u]^T$ is the vector of entropy variables. In (1.100), the *entropy conservative flux* is given by

$$F_{EC j+\frac{1}{2}} = \begin{bmatrix} \bar{h}_{j+\frac{1}{2}} \bar{u}_{j+\frac{1}{2}} \\ \bar{h}_{j+\frac{1}{2}} (\bar{u}_{j+\frac{1}{2}})^2 + \frac{g}{2} (\bar{h}_{j+\frac{1}{2}})^2 \end{bmatrix},$$

and the numerical diffusion operator is given by

$$D_{ES1 j+\frac{1}{2}} = R_{j+\frac{1}{2}} |\Lambda_{j+\frac{1}{2}}| R_{j+\frac{1}{2}}^T,$$

with

$$R_{j+\frac{1}{2}} = \frac{1}{\sqrt{2g}} \begin{bmatrix} 1 & 1 \\ \lambda_- & \lambda_+ \end{bmatrix}, \quad \lambda_{\pm} = \bar{u}_{j+\frac{1}{2}} \pm \sqrt{g \bar{h}_{j+\frac{1}{2}}}, \quad |\Lambda_{j+\frac{1}{2}}| = \begin{bmatrix} |\lambda_-| & 0 \\ 0 & |\lambda_+| \end{bmatrix}.$$  

A second-order scheme is obtained by replacing the flux $F_{ES1 j+\frac{1}{2}}$ in (1.99) with $F_{ES2 j+\frac{1}{2}}$,

$$F_{ES2 i+\frac{1}{2}} = F_{EC i+\frac{1}{2}} - \frac{1}{2} D_{ES1 i+\frac{1}{2}} \left( V^+_{i+\frac{1}{2}} - V^-_{i+\frac{1}{2}} \right), \tag{1.101}$$

where the $V^\pm$ are obtained from either a MINMOD or ENO reconstruction (see sections 1.4.5 - 1.4.6) of the scaled entropy variables, following the procedure of [39].

Both first and second order schemes satisfy a discrete version of the energy inequality (1.97) and preserve the discrete lake at rest (1.98), see [38]. The extension of this scheme to two space dimensions is detailed in [38].

### 1.4.9 Note on boundary conditions

Fluxes on the boundary of the computational domain are computed using so-called ghost cells [66], denoted by $C_{j+e_r}$ for $j_r = \# \mathcal{T}_r$ and $C_{j-e_r}$ for $j_r = 1$, which are “outside” of the computational domain $D$. For Cartesian domains $D = I_1, \ldots, I_d$ and uniform axiparallel meshes $\mathcal{T} = \mathcal{T}_1, \ldots, \mathcal{T}_d$, periodic and Neumann boundary conditions can be implemented straight-forwardly by copying ghost cell values accordingly.

For *periodic boundary conditions*, we set for index $j$ with $j_r = \# \mathcal{T}_r$,

$$U_{j+e_r} = U_{j-(\# \mathcal{T}_r-1)e_r}, \quad \forall r = 1, \ldots, d, \tag{1.102}$$
and for index $j$ with $j_r = 1$,

$$U_{j-e_r} = U_{j+(#T_r-1)e_r}, \quad \forall r = 1, \ldots, d.$$  \hfill (1.103)

For Neumann (or “transparent”) boundary conditions, we set for index $j$ with $j_r = #T_r$,

$$U_{j+e_r} = U_j, \quad \forall r = 1, \ldots, d.$$  \hfill (1.104)

and for index $j$ with $j_r = 1$,

$$U_{j-e_r} = U_j, \quad \forall r = 1, \ldots, d.$$  \hfill (1.105)

In the case of the acoustic wave equation (1.4), perfectly reflecting boundary conditions will also be used. There the vector $U$ of conserved variables consists of the acoustic pressure $p$ and velocities $u$, and we set for index $j$ with $j_r = #T_r$,

$$p_{j+e_r} = p_j, \quad u_{j+e_r} = -u_j, \quad \forall r = 1, \ldots, d.$$  \hfill (1.106)

and for index $j$ with $j_r = 1$,

$$p_{j-e_r} = p_j, \quad u_{j-e_r} = -u_j, \quad \forall r = 1, \ldots, d.$$  \hfill (1.107)

Ghost cell values for the remaining domain-dependent input data, including initial condition $U_0$, source $S$ and coefficients $c$ are set analogously.

Notice that due to the axiparallel mesh $T$ of the Cartesian domain $D$, different boundary conditions can be easily imposed on different parts of the boundary $\partial D$. Dirichlet boundary conditions, however, shall be handled more carefully, since the imposed boundary values must be consistent with the initial data.

Following the results in [6, 32] and chapter 10 in [66], appropriate boundary conditions (leading to a well-posed problem (1.25) on bounded domains $D \subset \mathbb{R}^d$) can be directly implemented in the setting of approximate Riemann solvers introduced in section 1.4.

### 1.4.10 Convergence theory for FVM schemes

In this section we review the convergence of the FVM schemes (1.53) to the weak entropy solution as in (1.30). In the case of scalar conservation laws, the convergence is proven in [23, 63] for the one dimensional case and in [20] for multiple dimensions using the TVD property of the (first order) monotone schemes. Arbitrarily high order entropy stable FVM schemes are also proven to converge in [39]. As linear hyperbolic systems of conservation laws can be transformed into characteristic variables and this way decoupled to a set of scalar conservation laws (in the absence of boundary conditions), analogous convergence results are also available in this case.

Finally, we address the issue of FVM convergence for nonlinear systems of conservation laws; apart from some stability results for specific cases such as shock and rarefaction waves, no general proofs are currently available even in one space dimension.
1.4 Finite Volume method

Scalar and linear hyperbolic systems of conservation laws

In this section, we provide sufficient assumptions on the (first order accurate) numerical fluxes required to obtain FVM approximations that converge to the weak entropy solutions of scalar conservation laws as $T$ is refined. In particular, first order accurate FVM schemes are assumed to be monotone, conservative and consistent.

**Definition 1.4.6 (Monotone scheme).** Every explicit two level FVM scheme can be written in the update form, where $U_j^{n+1}$ are obtained from $U_j^n$ inside stencils $S_r^j$ for $r = 1, \ldots, d$,

$$U_j^{n+1} = H \left( U_j^{n-(k-1)e_1}, \ldots, U_j^{n+(k-1)e_1}, \ldots, U_j^{n-(k-1)e_d}, \ldots, U_j^{n+(k-1)e_d} \right), \quad (1.108)$$

where $H$ is called the “update” function. The resulting FVM scheme is monotone if the update function $H$ in (1.108) is non-decreasing in each of its arguments.

In order to check whether a given first order FVM scheme is monotone, a simple criterion is available [66] for the two-point numerical flux functions $F_{j+\frac{1}{2}e_r}$, resulting in a stencil containing three cells in each direction (i.e. $k=1$).

**Lemma 1.4.7.** A FVM scheme with two-point flux $F(a, b) = F_{j+\frac{1}{2}e_r}(a, b)$ is monotone if

$$\begin{cases}
  a \mapsto F(a, b) \text{ is non-decreasing for fixed } b, \\
  b \mapsto F(a, b) \text{ is non-increasing for fixed } a,
\end{cases} \quad (1.109)$$

and the following CFL-type condition holds:

$$\left| \frac{\partial F}{\partial a} \right| + \left| \frac{\partial F}{\partial b} \right| \leq \frac{\Delta x}{\Delta t}. \quad (1.110)$$

It is straightforward to check (using Lemma 1.4.7) that first order Lax-Friedrichs and Rusanov fluxes (and the corresponding FVM schemes) are monotone.

**Definition 1.4.8 (Conservative scheme).** Ignoring boundary conditions and source terms, a given FVM scheme is called conservative if it satisfies

$$\sum_{c_j \in T} U_j^{n+1} = \sum_{c_j \in T} U_j^n. \quad (1.111)$$

All schemes described so far are conservative. Actually, (1.111) is equivalent to (1.53) together with any appropriate temporal discretization from subsection 1.4.7.

**Definition 1.4.9 (Consistent scheme).** A first order FVM scheme with two-point numerical flux function $F_{j+\frac{1}{2}e_r}(\cdot, \cdot)$ is consistent if

$$F_{j+\frac{1}{2}e_r}(U, U) = F_r(U), \quad \forall U \in \mathbb{R}^n. \quad (1.112)$$
Introduction

Again, all schemes described so far are consistent. Having all definitions in place, we state the results providing discrete versions of $L^p$ bounds (discrete entropy inequality), $L^\infty$ bounds and TV bounds; such bounds for FVM solutions $U^n_j$ are analogous to the bounds for exact solutions $U$, which hold in the continuous setting of Theorem 1.3.11.

Lemma 1.4.10 (Crandall-Majda). For scalar conservation laws, a FVM solution $U^n_j$ obtained from any consistent monotone scheme satisfies the discrete entropy inequality,

$$|U^n_{j+1} - k| - |U^n_j - k| + \frac{\Delta t}{\Delta x} \sum_{r=1}^d (Q^n_{j+\frac{1}{2}e_r} - Q^n_{j-\frac{1}{2}e_r}) \leq 0, \quad \forall n \in \mathbb{N}_0, \quad C_j \in \mathcal{T}, \quad (1.113)$$

where $Q^n_{j+\frac{1}{2}e_r}$ is the Crandall-Majda numerical entropy flux and is defined as follows:

$$Q^n_{j+\frac{1}{2}e_r} = Q(U^n_j, U^n_{j+e_r}) = F(U^n_j \lor k, U^n_{j+e_r} \lor k) - F(U^n_j \land k, U^n_{j+e_r} \land k). \quad (1.114)$$

Lemma 1.4.11 (Discrete maximum principle). For scalar conservation laws, an FVM solution $U^n_j$ obtained from any consistent monotone scheme with 2-point flux $F_{j+\frac{1}{2}e_r}(. , .)$ satisfies the discrete maximum (and minimum) principle for all $C_j \in \mathcal{T}$, $n \in \mathbb{N}_0$ and all directions $r = 1, \ldots, d$,

$$\min \{U^n_{j-e_r}, U^n_j, U^n_{j+e_r}\} \leq U^{n+1}_j \leq \max \{U^n_{j-e_r}, U^n_j, U^n_{j+e_r}\}. \quad (1.115)$$

In particular, for all $C_j \in \mathcal{T}$ and $n \in \mathbb{N}_0$,

$$\min_{c_j \in \mathcal{T}} U^0_j \leq U^n_j \leq \max_{c_j \in \mathcal{T}} U^0_j. \quad (1.116)$$

Lemma 1.4.12 (Discrete TVD). For scalar conservation laws, any consistent monotone scheme is TVD under the CFL condition (1.110).

The combination of Lemmas (1.4.10), (1.4.11) and (1.4.12) proves that for scalar conservation laws the necessary properties of the exact solution, i.e. $L^p$ bounds (entropy inequality), $L^\infty$ bounds and TV bounds, respectively, are “inherited” by MCC (monotone, conservative, consistent) schemes. It is shown in [23, 63] for the one dimensional case and in [20] for multiple dimensions that for convergent FVM schemes, the MCC assumption implies convergence to the unique weak entropy solution of scalar conservation law (1.1).

Theorem 1.4.13 (Lax-Wendroff). Define the FVM approximations $U_T(x, t)$ obtained from the consistent and conservative FVM scheme by

$$U_T(x, t) = U^n_j, \quad \forall (x, t) \in C_j \times [t^n, t^{n+1}), \quad (1.117)$$

and assume that for $U_0 \in L^\infty(D)$, $U_T(x, t)$

1. is uniformly bounded

$$\|U_T\|_{L^\infty(D \times \mathbb{R}_+)} \leq C \quad (1.118)$$

for some constant $C > 0$ independent of mesh $\mathcal{T}$;
2. converges to some function $\hat{U}(x,t) \in L^1(D \times \mathbb{R}_+)$ and pointwise almost everywhere as $\Delta x \to 0$.

Then $\hat{U}(x,t)$ is a weak solution of scalar conservation law (1.1) with $m = 1$.

For the proofs of Lemmas 1.4.10, 1.4.11, 1.4.12 and Theorem 1.4.13, we refer to the classical books [23, 63, 20].

For MCC schemes, the convergence of the FVM approximation $U_T$ to a limit function $\hat{U}$ in $L^1$ is a consequence of the $L^\infty$ and $BV$ bounds ensured by Lemmas 1.4.11 and 1.4.12. Additionally, $\hat{U}$ satisfies the entropy condition (1.97), since the approximating sequence satisfies the discrete entropy inequality (1.113). Finally, Theorem 1.4.13 establishes that this limit $\hat{U}$ is the exact weak entropy solution to (1.1) for $m = 1$.

Notice that high order time integration (1.95) and (1.96) preserves crucial properties of robust FVM schemes: monotonicity, consistency and conservativeness, provided the Forward Euler scheme (1.94) is MCC. Arbitrarily high order entropy stable FVM schemes were proven to converge in [39].

**Nonlinear systems of conservation laws**

No rigorous convergence results of FVM methods are available for general nonlinear systems of conservation laws, even in one space dimension $d = 1$. Moreover, the numerical stability of most convergent schemes is also not available, see the counterexamples in [37]. So far, only the notion of entropy stability has been rigorously analyzed in [39], requiring the FVM approximations of (1.1) to satisfy a discrete version of the entropy inequality (1.30): such a stability notion, however, does not necessarily ensure the convergence of numerical FVM approximations of (1.1). An example of such numerical scheme was described in subsection 1.4.8, and we refer to [39, 38, 106, 107] for general derivations.

Given the lack of rigorous results on convergence and stability of FVM schemes for systems of nonlinear conservation laws, the convergence of the numerical schemes can only be empirically verified using a handful of well-analyzed benchmarks for nonlinear systems of conservation laws, for which entropy solutions are known to exist and to be unique. An example of such a benchmark would be the radial Sod shock tube analyzed in detail in [37]; a one-dimensional version is also introduced in subsection 7.1.1. We refer to [41, 38, 37] for further benchmark simulations of Euler, MHD and shallow water equations.

A set of numerical algorithms for approximation of entropy measure valued solutions (EMVS) discussed in subsection 1.3.4 are presented in [37]. These methods, however, additionally to the FVM solver, require a Monte Carlo type sampling of the given initial Young measure. Such combined MC-FVM methods, introduced in the forthcoming chapter 3, are proven in [37] to provide stable and convergent approximations of EMVS and its statistical moments.
1.4.11 Estimates for the computational work of the FVM solver

For a given mesh \( \mathcal{T} \) with mesh width \( \Delta x \) and total number of cells \( N = \# \mathcal{T} \), one time step of the FVM scheme (i.e. to obtain \( U_{j}^{n+1} \) from \( U_{j}^{n} \)) consists in evaluating the numerical flux approximations \( F_{j}^{n+1/2} \) at cell interfaces and then applying the time stepping scheme in order to update cell averages \( U_{j}^{n} \) and obtain \( U_{j}^{n+1} \); everything is performed for each cell \( C_{j} \in \mathcal{T} \).

Hence, asymptotically, the total computational work (in terms of FLOPs, i.e. the required number of floating point operations) for one time step of such explicit FVM schemes is given by

\[
\text{Work}_{T}^{\text{step}} = \text{Work}_{T}^{\text{step}}(\Delta x) = \mathcal{O}(N) = \mathcal{O}(\Delta x^{-d}).
\]  

(1.119)

In most explicit FVM schemes \[66\], lower order terms \( \mathcal{O}(\Delta x^{-d}) \) in (1.119) are negligible, even on a very coarse mesh. Hence, we assume a stricter version of (1.119),

\[
\text{Work}_{T}^{\text{step}} = \text{Work}_{T}^{\text{step}}(\Delta x) = K \Delta x^{-d},
\]  

(1.120)

where constant \( K \) depends on the particular FVM scheme that is used and on the time horizon \( t > 0 \), but does not depend on the mesh width \( \Delta x \). To ensure the stability of the FVM scheme, a CFL condition (1.51) is imposed on the time step size \( \Delta t := t^{n+1} - t^{n} \), which forces

\[
\Delta t = \frac{C_{\text{CFL}}}{\lambda} \Delta x, \quad 0 < C_{\text{CFL}} \leq 1, \quad \lambda > 0,
\]  

(1.121)

where the so-called CFL number \( C_{\text{CFL}} \) does not depend on \( \Delta x \) and \( \lambda \) is the absolute value of the maximal wave speed in (1.51). Note that in (1.51), the CFL number \( C_{\text{CFL}} \) is set to 0.5. Hence, the computational work \( \text{Work}_{T}^{\text{det}} \) for one complete deterministic solve using the FVM method on the triangulation \( \mathcal{T} \) with mesh width \( \Delta x \) is given by multiplying the work for one step (1.120) by the total number of time steps \( \Delta t^{-1} \) for the time horizon \( T > 0 \),

\[
\text{Work}_{T} = \text{Work}_{T}^{\text{det}} \cdot \frac{T}{\Delta t} = K \Delta x^{-d} \lambda \frac{T}{C_{\text{CFL}} \Delta x} = B \lambda \Delta x^{-(d+1)},
\]  

(1.122)

where, for the brevity of exposition, we use the notation \( B = (KT)/C_{\text{CFL}} \).

**Remark 1.4.14.** The computational work \( \text{Work}_{T} \), representing the number of required floating points operations (FLOPs), will be assumed to be directly proportional to the actual run-time of the FVM algorithm. One should, however, be aware of possible deviations due to the influence of (computer architecture-dependent) non-uniform memory access and caching effects, which will not be discussed in this thesis.
1.5 Uncertainty

Several types of uncertainty can be considered for (1.1): aleatoric and epistemic [60]. Aleatoric (statistical) uncertainty is present when different results are obtained by repeatedly performing the same experiment, and the resulting uncertainty can not be suppressed by more accurate measurements. Examples of such problems include Kelvin-Helmholtz and Richtmyer-Meshkov instabilities [37], governed by the Euler equations (1.8). In these two examples, the solution is generally non-unique [37]; hence only bulk properties, such as the expected value, could be quantified. Epistemic (systematic) uncertainty is due to uncertain parameters, i.e. quantities that could in principle be known, however, precise values might be unavailable in practice. For instance, existing numerical methods for approximating (1.1) require the initial data $U_0$, boundary data $g$, coefficient $c$, source term $S$, and flux functions $F_r$ as inputs. However, in most practical situations, it is not possible to measure these inputs precisely. In this thesis, we mainly consider only the epistemic (systematic) uncertainty of input data.

1.5.1 Uncertainties in input data

In the following, we provide concise examples for each potential source of randomness in the uncertain input data, including initial data $U_0$, boundary data $g$, coefficient $c$, source term $S$, and flux functions $F_r$.

Uncertainty in initial data

Random initial data arises in many situations, where physical measurement of $U_0$ is not available due to potential technical difficulties, for instance:

- Unreachable physical locations. For instance, initial data measurements for plasma simulation on the solar surface using MHD equations (1.10) are not available. Another example would be the modeling of propagation of tsunami waves with the shallow water equations, where it is not possible to measure the initial water displacement (at tsunami source) with any precision in real time (cp. eg. [67]).

- Short and unpredictable time frames for the measurements. For example, the initial offset of the water column in a tsunami after an underwater earthquake commenced. Such phenomena are often modeled with shallow water equations (1.12).

- No accurate or affordable measurement techniques available. Cheaper measurement techniques could be used and uncertainty in measurements could be accounted for.
Introduction

Uncertainty in sources

Random sources $S$ mainly (but not only) arise in the setting of shallow water equations (1.12), where the topography $b(x)$ directly enters the source term $S$. Randomness in $b(x)$ can originate for several reasons:

- **Old measurement data in dynamic environments.** For example, the bathymetry of ocean and sea beds is measured to rather high precision near coastline; however, due to underwater streams and other dynamical phenomenon, bathymetry is constantly changing. Such changes could be accounted for as additional uncertainty when modeling wave propagation using shallow water equations (1.12).

- **Unreachable physical locations.** For instance, climate modeling in far exoplanets using shallow water equations (1.12) require knowledge of the terrain topography.

- **Scarce and expensive measurements.** For example, the measurements of bathymetry of ocean and sea beds far from coastline are often scarce [35, 10] and not very precise due to the sonar measurement techniques used.

Uncertainty in fluxes

Random fluxes $F_r$ are encountered due to uncertain parameters of the physical model:

- **Unknown exact equations of state.** For example, the gas constant $\gamma$ in the equation of state (1.9) of Euler equations (1.8) is not known precisely in some cases.

- **Unknown interaction parameters.** In Buckley-Leverett equations (1.3) of two phase flow, modeling the water flooding of an oil and gas reservoir, the rock permeability and the relative permeabilities of each phase with respect to the other need to measured. Again, the measurement process is characterized by uncertainty. Consequently, the inputs (the fluxes) to the underlying two-phase flow equations are uncertain.

Uncertainty in coefficients

Many models (and hence also the resulting governing equations) involve various types of coefficients, which need to be specified and are often given by (possible not sufficiently accurate) measurements or estimated (up to some level of certainty) using available a-priori knowledge. We will consider one special case - the propagation of acoustic waves in the anisotropic layered subsurface media. The scarcity of seismic measurements leads to statistical descriptions of material properties of the medium, which then result in uncertain wave propagation speeds.
1.5.2 Probabilistic modeling of uncertainty

There are many theories for mathematical modeling of the uncertainty; the most prominent ones include fuzzy logic (also known as fuzzy sets) theory [122], Dempster–Shafer (also known as “belief” or “evidence”) theory [28, 96], and Kolmogorov probability theory [59]. For the modeling of uncertainty in inputs and solutions of PDEs, Kolmogorov probability theory is most frequently chosen [26]; this model is also our choice in this thesis. As suggested in [26], the uncertain inputs for (1.1) are then modeled by random fields with prescribed probability laws, which we review within the following sections.

Random fields

Let $\Omega, \mathcal{F}$ be a measurable space, with $\Omega$ denoting the set of all elementary events $\omega \in \Omega$, and $\mathcal{F}$ a $\sigma$-algebra of all possible events in our probability model. A probability measure $\mathbb{P}$ on $(\Omega, \mathcal{F})$ is a $\sigma$-additive set function from $\Omega$ into $[0, 1]$ such that $\mathbb{P}(\Omega) = 1$, and the measure space $(\Omega, \mathcal{F}, \mathbb{P})$ is called probability space. We shall always assume, unless explicitly stated, that $(\Omega, \mathcal{F}, \mathbb{P})$ is complete. Denoting a second measurable space by $(\mathcal{E}, \mathcal{G})$, an $\mathcal{E}$-valued random field (or random variable taking values in $\mathcal{E}$) is any mapping $X : \Omega \to \mathcal{E}$ such that the set $\{\omega \in \Omega : X(\omega) \in A\} \in \mathcal{F}$ for any $A \in \mathcal{G}$, i.e. such that $X$ is a $\mathcal{G}$-measurable mapping $\Omega \to \mathcal{E}$. $\mathcal{L}(X)$ denotes the law of $X$ under $\mathbb{P}$,

$$\mathcal{L}(X)(A) = \mathbb{P}(\{\omega \in \Omega : X(\omega) \in A\}) \quad \forall A \in \mathcal{G}.$$  

The image measure $\mu_X = \mathcal{L}(X)$ on $(\mathcal{E}, \mathcal{G})$ is called law (or distribution) of $X$.

Assume now that $\mathcal{E}$ is a separable Banach space with norm $\|\cdot\|_E$ and (topological) dual $\mathcal{E}^*$, then $\mathcal{B}(\mathcal{E})$ is the smallest $\sigma$-field of subsets of $\mathcal{E}$ containing all sets

$$\{x \in \mathcal{E} : \varphi(x) \leq \alpha\}, \quad \varphi \in \mathcal{E}^*, \quad \alpha \in \mathbb{R}.$$  

Hence for a separable Banach space $\mathcal{E}$, $X : \Omega \to \mathcal{E}$ is an $\mathcal{E}$-valued random field if and only if for every $\varphi \in \mathcal{E}^*$, $\omega \mapsto \varphi(X(\omega)) \in \mathbb{R}$ is an $\mathbb{R}$-valued random variable.

**Definition 1.5.1 (Random field).** For a separable Banach space $\mathcal{E}$, an $\mathcal{E}$-valued random field is a $(\mathcal{F}, \mathcal{B}(\mathcal{E}))$-measurable mapping $\Omega \to \mathcal{E}$,

$$X : (\Omega, \mathcal{F}) \to (\mathcal{E}, \mathcal{B}(\mathcal{E})).$$

**Lemma 1.5.2 (Random field norm is a random variable).** Let $\mathcal{E}$ be a separable Banach space and let $X : \Omega \to \mathcal{E}$ be an $\mathcal{E}$-valued random field on $(\Omega, \mathcal{F})$. Then the mapping $\Omega \ni \omega \mapsto \|X(\omega)\|_E \in \mathbb{R}$ is measurable, i.e. it is a random variable $(\Omega, \mathcal{F}) \to (\mathbb{R}, \mathcal{B}(\mathbb{R}))$.

**Proof.** Since $\mathcal{E}$ is separable, there exists a sequence $\{\varphi_n\} \subset \mathcal{E}^*$ such that for all $x \in \mathcal{E}$,

$$\|x\|_E = \sup_{n \in \mathbb{N}} |\varphi_n(x)| \quad \text{implying} \quad \forall \omega \in \Omega : \|X(\omega)\|_E = \sup_{n \in \mathbb{N}} |\varphi_n(X(\omega))|.$$  

Hence $\omega \mapsto \|X(\omega)\|_E$ is an $\mathbb{R}$-valued random variable.  \qed
Introduction

The random field $X : \Omega \to E$ is called \textit{Bochner integrable} if, for any probability measure $\mathbb{P}$ on the measurable space $(\Omega, F)$,

$$\int_{\Omega} \|X(\omega)\|_E d\mathbb{P}(\omega) < \infty.$$ 

By $L^1(\Omega, E) = L^1((\Omega, F, \mathbb{P}), E)$ we denote the set of all (equivalence classes of) Bochner integrable, $E$-valued random fields $X$, equipped with the norm

$$\|X\|_{L^1(\Omega, E)} = \int_{\Omega} \|X(\omega)\|_E d\mathbb{P}(\omega) = \mathbb{E}(\|X\|_E).$$

More generally, for $1 \leq p < \infty$, we define $L^p(\Omega, E) = L^p((\Omega, F, \mathbb{P}), E)$ as the set of Bochner $p$-integrable random fields taking values $E$, equipped with the norm

$$\|X\|_{L^p(\Omega, E)} := (\mathbb{E}(\|X\|_E^p))^{1/p}, \quad 1 \leq p < \infty.$$ 

For $p = \infty$, we denote by $L^\infty(\Omega, E) = L^\infty((\Omega, F, \mathbb{P}), E)$ the set of all $E$-valued random fields which are $\mathbb{P}$-almost surely bounded, equipped with the norm

$$\|X\|_{L^\infty(\Omega, E)} := \text{ess sup}_{\omega \in \Omega} \|X(\omega)\|_E.$$ 

Expectation of a random field

We will define the mathematical \textit{expectation} $\mathbb{E}[X]$ of an integrable random field $X \in L^1(\Omega, E)$.

A random field $X$ taking values in $E$ is called \textit{simple} if it can take only finitely many values, i.e. if it has the explicit form (with $\chi_A$ the indicator function of $A \in F$)

$$X = \sum_{i=1}^{N} x_i \chi_{A_i}, \quad A_i \in F, x_i \in E, N < \infty.$$ 

We set, for simple random fields $X$ taking values in $E$ and for any $B \in F$,

$$\int_{B} X(\Omega) d\mathbb{P}(\omega) = \int_{B} X d\mathbb{P} := \sum_{i=1}^{N} x_i \mathbb{P}(A_i \cap B).$$ 

By density, for such $X(\cdot)$, and all $B \in F$,

$$\left\| \int_{B} X(\Omega) d\mathbb{P}(\omega) \right\|_E \leq \int_{B} \|X(\Omega)\|_E d\mathbb{P}(\omega).$$ 

For any random variable $X : \omega \to E$ which is Bochner integrable, there exists a sequence $\{X_n\}_{n \in \mathbb{N}}$ of simple random variables such that, for all $\omega \in \Omega$, $\|X(\Omega) - X_n(\Omega)\|_E \to 0$ as $n \to \infty$. Therefore, the above extends by continuity to any $E$-valued random variable.
1.5 Uncertainty

**Definition 1.5.3** (Expectation of a random field). For a Bochner integrable $E$-valued random field $X \in L^1(\Omega, E)$, the expectation $E[X]$ of $X$ is a deterministic element of $E$,  

$$E[X] = \int_\omega X(\omega) \, d\mathbb{P}(\omega) = \lim_{n \to \infty} \int_\omega X_n(\omega) \, d\mathbb{P}(\omega) \in E.$$  

(1.123)

Higher moments

For $k \in \mathbb{N}$ and a separable Banach space $E$, following [75, Section 3.4] we denote by $E^{(k)} = E \otimes \cdots \otimes E$ the Banach space of the $k$-fold tensor product of $k$ copies of $E$, equipped with a cross norm, i.e.  

$$\|X_1 \otimes \cdots \otimes X_k\|_{E^{(k)}} = \|X_1\|_E \cdots \|X_k\|_E, \quad \forall X_1, \ldots, X_k \in E.$$  

(1.124)

For $k \in \mathbb{N}$ and for a Bochner $k$-integrable $E$-valued random field $X \in L^k(\Omega, E)$, we consider the random field $X^{(k)}$ defined by $k$ copies of $X$, given by $X(\omega) \otimes \cdots \otimes X(\omega)$. Then  

$$X^{(k)} := X \otimes \cdots \otimes X \in L^1(\Omega, E^{(k)}),$$  

(1.125)

and, by the definition of the tensor product norm,  

$$\|X^{(k)}\|_{L^1(\Omega, E^{(k)})} = \int_\Omega \|X(\omega)\|_E^k \, d\mathbb{P}(\omega) = \|X\|_{L^k(\Omega, E)}^k < \infty.$$  

**Definition 1.5.4.** The $k$-th moment (a $k$-point correlation function) of $X \in L^k(\Omega, E)$ is a deterministic element $\mathcal{M}^k X$ of $E^{(k)}$, given by  

$$\mathcal{M}^k X := E[X^{(k)}] \in E^{(k)}.$$  

(1.126)

Additionally, define variance $\mathbb{V}[X] \in E$ of an $E$-valued random field $X \in L^2(E)$ as the trace of the centered second moment $\mathcal{M}^2[X] \in E^{(k)}$, well-defined by Definition [1.5.4],  

$$\mathbb{V}[X] = \text{trace}(\mathcal{M}^2[X - E[X]]) = E[XX] - E[X]E[X] \in E.$$  

(1.127)

For time-dependent $E$-valued random fields $X \in L^k(\Omega, C([0, T], E))$, the Definition [1.5.4] of the $k$-th moments $\mathcal{M}^k X$ can be extended to $k$-th moments $\mathcal{M}^k X(t_1, \ldots, t_k)$, i.e. $k$-point correlation functions of $X$ at arbitrary times $(t_1, \ldots, t_k) \in [0, T]^k$. We provide such a generalization of Definition [1.5.4] as Lemma [1.5.5] recapitulated from [75].

**Lemma 1.5.5.** Consider a time-dependent, $E$-valued random field $X : \Omega \to C([0, T], E)$, and assume that for some $k \in \mathbb{N}$ and some real number $r$ such that $k \leq r < \infty$,  

$$X \in L^r(\Omega, C([0, T], E)).$$  

(1.128)

Then, for every $0 < T < \infty$ and every vector  

$$t = (t_1, \ldots, t_k) \in [0, T]^k, \quad 0 < t_1, t_2, \ldots, t_k \leq T < \infty,$$  

(1.129)
the spatial \( k \)-point correlation function

\[
X(t_1, \omega) \otimes \cdots \otimes X(t_k, \omega)
\]  

is well-defined as an element of \( L^{r/k}(\Omega, E^{(k)}) \). Moreover, the \( k \)-th moment

\[
\mathcal{M}^k[X](t) = \mathcal{M}^k[X](t_1, \ldots, t_k) := \mathbb{E}[X(t_1, \omega) \otimes \cdots \otimes X(t_k, \omega)]
\]  

is well-defined for any choice of \((t_1, \ldots, t_k)\) as in (1.129) as an element of \( E^{(k)} \), and

\[
\|\mathcal{M}^k[X](t_1, \ldots, t_k)\|_{E^{(k)}} \leq \left\| \bigotimes_{r=1}^k X(t_r, \cdot) \right\|_{L^1(\Omega, E^{(k)})} \leq \|X\|_{L^r(\Omega, C([0,T], E))}.
\]  

Proof. Since \( 1 \leq k \leq r < \infty \), using (1.125) and (1.128) we obtain for \( 0 < t_1, \ldots, t_k < T \),

\[
X(t_1, \omega) \otimes \cdots \otimes X(t_k, \omega) \in L^{r/k}(\Omega, E^{(k)}),
\]

which is (1.130). Since \( r/k \geq 1 \), the \( k \)-th moment satisfies, by (1.128) and (1.124),

\[
\|\mathcal{M}^k[X](t)\|_{E^{(k)}} \leq \mathbb{E}\left[ \left\| \bigotimes_{r=1}^k X(t_r, \omega) \right\|_{E^{(k)}} \right] \\
\leq \mathbb{E}\left[ \|X(\omega)\|_{C([0,T], E)}^k \right] = \|X\|_{L^k(\Omega, C([0,T], E))}^k.
\]

Hence (1.131) is well-defined as an element in \( E^{(k)} \), and (1.132) follow from Hölder’s inequality since \( r/k \geq 1 \) by assumption. \( \square \)

1.6 Random conservation laws

The inputs to (1.1) are uncertain and this uncertainty propagates to the solution. The modeling and approximation of the propagation of uncertainty in the solution due to uncertainty in inputs constitutes the theme of uncertainty quantification (UQ). The solution of conservation laws (1.1) with random input is also realized as a random field and the statistical moments (1.126) of the solution like the expectation (1.123) and variance are the quantities of interest. For simplicity of exposition, analogously as in section 1.3, all theoretical results will be considered for Cauchy problems with \( D = \mathbb{R}^d \). Analogous results are available for general periodic bounded Cartesian domains \( D = I_1 \times \cdots \times I_d \) introduced in section 1.4.
1.6 Random conservation laws

1.6.1 Definitions

The conservation law (1.1) with random initial data, coefficients, sources and fluxes is

\[
\begin{cases}
\frac{\partial U(x,t,\omega)}{\partial t} + \text{div}(F(c(x,\omega), U, \omega)) = S(x,t, U, \omega), \\
U(x,0,\omega) = U_0(x,\omega),
\end{cases} 
\]

on \( \Omega \times (0,\infty) \) \( \omega \)-a.e., \( \Omega \)-a.e., \( \in \mathbb{R}^d \) \( \in \mathbb{R}^m \), and if it satisfies the entropy condition (corresponding to the entropy formulation for (1.1)) for \( \mathbb{P} \)-a.e. \( \omega \in \Omega \),

\[
F_r : \Omega \times (0,\infty) \ni (\omega,t) \mapsto F_r(U,\omega),
\]

1. **Weak solution.** For \( \mathbb{P} \)-a.e. \( \omega \in \Omega \), \( U(\omega) = U(x,t,\omega) \) satisfies the following,

\[
\begin{aligned}
\int_{\mathbb{R}^d \times \mathbb{R}^+} & \left( U(\omega) \cdot \varphi_t + \sum_{r=1}^d F_r(\omega) \cdot \frac{\partial}{\partial x_i} \varphi \right) dx \, dt \\
&+ \int_{\mathbb{R}^d} U_0(\omega) \cdot \varphi|_{t=0} \, dx = \int_{\mathbb{R}^d \times \mathbb{R}^+} S(\omega) \cdot \varphi \, dx \, dt.
\end{aligned}
\]
Introduction

for all test functions $\varphi \in C^1_c(\mathbb{R}^d \times \mathbb{R}_+).$ The following notation is used in the integrals of (1.134): $F_r(\omega) = F_r(c(x,\omega),U(\omega),\omega)$ and $S(\omega) = S(x,t,U(\omega),\omega).

2. **Entropy condition.** For $\mathbb{P}$-a.e $\omega \in \Omega$ and for any deterministic convex entropy function $S(\cdot)$ and stochastic entropy fluxes $Q_r(\omega,\cdot)$, for directions $r = 1, \ldots, d$, such that (1.27) holds, $U$ satisfies the following integral identity for all test functions $0 \leq \varphi \in C^1_c(\mathbb{R}^d \times \mathbb{R}_+),$

$$
\int_{\mathbb{R}^d \times \mathbb{R}_+} \left( S(U(x,t,\omega))\varphi_t(x,t) + \sum_{r=1}^{d} Q_r(\omega, U(x,t,\omega)) \frac{\partial}{\partial x_r} \varphi(x,t) \right) \, dx \, dt \geq 0. \quad (1.135)
$$

We remark, that the pointwise (in the space $\Omega$) definition of the random entropy solution (as in Definition 1.6.1) can be alternatively defined only weakly in the space $\Omega$; however, since there are no derivatives with respect to $\omega \in \Omega$, the resulting definitions are equivalent.

1.6.2 Challenges and methods for uncertainty quantification

It is highly non-trivial to develop efficient algorithms for quantifying uncertainty in conservation laws with random inputs. The biggest challenge lies in the fact that the discontinuities in physical space (which inevitably arise in solutions of nonlinear hyperbolic conservation laws) may propagate into parametric representations of the probability densities (laws) of the random solutions. A robust numerical method should be able to deal with these discontinuities. Another challenge lies in dealing with the fact that the number of random sources driving the uncertainty may be very large (possibly infinite in the case of random field inputs parametrized by Karhunen-Loève expansions).

The design of efficient numerical schemes for quantifying uncertainty in solutions of partial differential equations has seen a lot of activity in recent years. In the following paragraphs, we briefly review the main approaches.

Among the most popular methods (particularly for elliptic and parabolic PDEs) are the stochastic Galerkin methods based on generalized Polynomial Chaos (gPC for short). An incomplete list of references on gPC methods for uncertainty quantification in hyperbolic conservation laws includes [3, 15, 65, 69, 113, 112, 91, 115, 48] and other references therein. Although these deterministic methods show some promise, they suffer from the disadvantage that they are highly intrusive: existing codes for computing deterministic solutions of balance (conservation) laws need to be completely reconfigured for implementation of the gPC based stochastic Galerkin methods. An alternative class of methods for quantifying uncertainty in PDEs are the stochastic collocation methods, see [120, 71, 118]. Stochastic collocation methods are non-intrusive and easier to parallelize than the gPC based stochastic Galerkin methods. However, the lack of regularity of the solution with respect to the stochastic variables (the solution can be discontinuous in the stochastic variables) impedes efficient performance of both the stochastic Galerkin
1.6 Random conservation laws

as well as the stochastic collocation methods. Yet another set of alternative methods, which heavily rely on the assumed low “effective” number of stochastic dimensions, include adaptive analysis of variance (ANOVA) [121], proper generalized decomposition (PGD) [15] and Fokker-Planck-Kolmogorov type [114] techniques. In addition to the assumption that the “effective” number of sources of uncertainty is low, these methods require very complex representations of the input random fields, which are in practice rarely available exactly for which the numerical estimation can be computationally expensive. The newly developed stochastic Finite Volume [76] method (SFVM) attempts to cure these shortcomings, where the FVM methodology is applied on the product space of physical and stochastic domains. Under sufficient pathwise regularity assumptions of the random solution, and using the adaptivity of the mesh resolution in the stochastic coordinates in order to reduce the impact of the curse of dimensionality for the computational complexity, the SFVM method was shown to provide convergence rates higher than Monte Carlo convergence rate of 1/2 in approximation of the statistical moments. None of these methods are currently able to handle large numbers of sources of uncertainty (stochastic dimensions).

Another class of methods for computational uncertainty quantification in numerical solutions of PDEs are statistical sampling methods, most notably Monte Carlo (MC) sampling. In a MC method, the probability space is sampled and the underlying deterministic PDE is solved for each sample. The MC samples of numerical solutions of the PDE are combined into statistical estimates of expectation and other statistical moments of the random solution which are necessary to quantify uncertainty. MC methods are non-intrusive; they can, therefore, be based on existing, deterministic CFD solvers. In uncertainty quantification for hyperbolic scalar conservation laws with random initial data, MC type methods together with Finite Volume (FV) spatio-temporal discretizations of the PDE were proposed in a recent paper [75] and the estimates of the combined discretization and statistical sampling errors were obtained. In particular, it was shown that MC methods converge at rate 1/2 as the number $M$ of MC samples increases with each “sample” corresponding to a full, deterministic flow simulation. The asymptotic convergence rate $M^{-1/2}$ in terms of the number $M$ of MC samples is set by the central limit theorem. To achieve a sampling error which is of the order of the discretization error, MC Finite Volume Methods therefore require a large number of “samples”, with each sample consisting of the numerical solution of (1.1) for a given draw of the input data. This slow convergence entails high computational costs for MC type UQ methods in the field of computational fluid dynamics (CFD). In particular, accurate quantification of uncertainty by direct MC methods combined with available solvers for hyperbolic systems of conservation or balance laws in several space dimensions becomes very costly. Variance reduction techniques (such as importance sampling, control variates, stratified sampling, correlated sampling, and conditional Monte Carlo) [30] or quasi Monte Carlo methods [31] could be used to possibly improve the plain MC method. Efficient variance reduction, however, requires additional analysis and knowledge about the second moments of the a-priori unknown solution of the random PDE; such knowledge is rarely available. Quasi Monte Carlo methods on the other hand require a parametrization and
smoothness assumptions of the unknown solution field which may not hold for nonlinear problems with discontinuous solutions considered here, possibly resulting in a lack of robustness with respect to the curse of large stochastic dimensions.

In order to address the slow convergence of MC methods, a novel Multi-Level Monte Carlo Finite Volume (MLMC-FVM) algorithm for scalar conservation laws was proposed in [75]. Multi-Level MC methods were introduced by S. Heinrich for numerical quadrature [57] and developed by M. Giles to enhance the efficiency of path simulations for Itô stochastic ordinary differential equations in [42, 43]. More recently, MLMC Finite Element Methods for elliptic problems with stochastic coefficients were introduced by Barth, Schwab and Zollinger in [9]. MLMC methods for various other stochastic PDEs with numerous applications have been recently proposed in [7, 19, 44, 83]. The analysis in these references, in particular in [75], reveals that the MLMC-FVM is able to deliver converged numerical approximations to statistics of uncertain solutions of partial differential equations in computational complexity comparable to that of one numerical solve of a single “path”, i.e. a single realization of the random input data. Moreover, the above result is obtained under minimal regularity assumptions on the solution; in particular, only finite second moments of the random solution are required.

Recent papers [77, 78, 79, 105, 80, 95, 8] extended and analyzed the MLMC algorithm for linear and non-linear systems of conservation laws with random initial data, source terms, fluxes and coefficients. The asymptotic error and complexity analysis for the MLMC method, presented in [79, 105], confirmed the findings of [75], i.e. the method allows the computation of approximate statistical moments with the same accuracy versus computational cost ratio as a single deterministic solve. Efficient static and adaptive load balancing strategies proposed in [104, 103] enabled us to compute realistic solutions of the multi-dimensional random Euler, magnetohydrodynamics, shallow water, acoustic wave and Buckley-Leverett equations.

1.7 Outline of the dissertation

We begin with the definitions for concepts of random entropy solutions for scalar and linear hyperbolic systems of multi-dimensional conservation laws with random inputs in chapter 2. A mathematical framework of well-posedness of such problems is presented, and, in particular, precise statements on the existence and the uniqueness of random entropy solutions for scalar, multi-dimensional conservation laws with random inputs are provided. To this end, the results of recent papers [75, 105, 79] on random entropy solutions for scalar and linear hyperbolic systems of conservation laws with uncertain initial data, sources, fluxes and coefficients are recapitulated. Further details and complete mathematical developments of the results for scalar conservation laws with random fluxes are available in [76]. The corresponding theory will provide a rigorous basis for the design and analysis of Multi-Level Monte Carlo Finite Volume Methods for the ef-
efficient computational quantification of uncertainty in scalar and systems of hyperbolic conservation laws with random input data.

In chapters [3] and [4], the essentials on statistical sampling methods of the Monte Carlo (MC) and Multi-Level Monte Carlo (MLMC) type are outlined, exposing the limitations of the MC-FVM algorithms and discussing the advantages of the MLMC-FVM methods for their use in computational fluid dynamics. We summarize recent results from [75, 105, 77, 79, 76], describe the algorithms, state and prove theorems outlining the asymptotic error convergence results, based on which possible strategies for choosing optimal number of samples for the MLMC-FVM methods are derived. The obtained number of samples equilibrate or optimize the arising statistical and spatio-temporal discretization errors, allowing to achieve maximum accuracy in the statistical estimates of first and higher order moments of the random solution. Finally, novel complexity analysis of mean square error of MC-FVM and MLMC-FVM versus expected computational work is presented, and MLMC-FVM is shown to allow the computation of approximate statistical moments with the same accuracy versus cost ratio as a single deterministic FVM solve.

While offering significantly improved efficiency over standard MC methods, the MLMC-FVM methods are still totally non-intrusive and are as easy to code and parallelize as traditional, single level MC-FVM methods. Hence, chapter [5] addresses several interesting implementation issues: due to the massive computational effort entailed by the accurate numerical solution of multi-dimensional systems of conservation laws, numerical solves of (1.1) for single samples of random input data can not be performed on a single processor. Therefore, in this thesis the MLMC-FVM algorithm is extended towards several forms of parallelism: parallel computation of large numbers of samples on coarse mesh resolutions and parallelism by mesh partitioning (i.e. domain decomposition) of the (few) samples on finest mesh resolutions. To this end, we describe novel static and adaptive load balancing paradigms for a scalable version of the MLMC-FVM on a certain class of massively parallel hardware. Other relevant implementation ingredients, also described in this chapter, include the choice of robust deterministic FVM solver and its required features, parallel pseudo random number generation, and numerically stable efficient bias-free statistical estimators. Finally, all the ingredients are thoroughly tested within the MLMC-FVM implementation ASLVID-UQ, for which strong and weak parallel scaling are verified.

The rest of the thesis presents numerous numerical experiments to demonstrate the efficiency of the proposed algorithms and to analyze the statistical moments of the obtained approximate random fields. The common setup for all numerical experiments, such as notation, terminology, simulation parameter abbreviations and the methods for error computation, is described in [Chapter 6]. Systems of conservation laws with uncertain initial data, uncertain source terms, uncertain flux functions and uncertain coefficients are considered in the numerical examples.

We remark that existence, uniqueness and convergence theory for numerical schemes is obtained only for scalar and linear hyperbolic systems of conservation laws in multiple dimensions; these theoretical results will be verified in numerical experiments throughout
Introduction

The chapters 7 - 10. Theory for nonlinear systems of hyperbolic conservation laws is currently out of reach, as there are no mathematical convergence results for numerical schemes even for systems with deterministic data. Hence we will use the error estimates derived in chapter 4 as a guide to choose the number of samples per sampling level in the MLMC algorithm. The absence of rigorous error estimates makes it imperative to test the MLMC-FVM also for nonlinear systems of conservation laws for a large number of benchmark problems and demonstrate the efficiency of this algorithm. We aim to verify that the basic conclusions drawn based on the numerical analysis of MLMC-FVM for scalar and linear hyperbolic systems of conservation laws appear to be applicable also to algorithm design for a wide range of nonlinear systems of hyperbolic conservation laws.

In chapter 7, systems of nonlinear conservation laws with uncertain initial data are considered. In particular, we consider the Euler equations of gas dynamics and the system of compressible, ideal MHD equations in two space dimensions and show that the MLMC-FVM methods can quantify uncertainty in very complex realistic situations; for instance, the MLMC-FVM is able to handle a large number of sources of uncertainty (reminiscent of a large number of stochastic dimensions in a gPC method) that appear to be beyond the reach of other existing UQ methods for systems of hyperbolic conservation laws. An empirical convergence analysis of the numerical error will be presented for the MHD problem of the Orszag-Tang vortex.

In chapter 8, systems of nonlinear conservation laws with uncertain source terms are presented. In practice, the uncertainty in bottom topography is realized as random (in general, spatially correlated) fluctuations on each node of an underlying mesh. This implies possibly (countably) infinitely many sources of randomness. As the MLMC algorithm (to be introduced in chapter 4) entails that the bottom topography (modeled on a very fine mesh) needs to be sampled on coarse meshes, the very large number of topography samples with high spatial resolution will render the MLMC algorithm inefficient. Therefore, we propose a novel multi-level alias-free representation of random nodal values of the underlying (uncertain) bottom topography in terms of a hierarchical basis. This representation, together with the structure of the spatial discretization, allows us to make the bottom topography samples from a finer mesh redundant when the solution is computed on a given mesh. Hence, the new MLMC-FVM algorithm is considerably faster than plain MC-FVM. The chapter concludes with a description and demonstration of MLMC-FVM approximation of random event probabilities for statistical risk quantification and analysis.

Chapter 9 presents numerical results for scalar nonlinear conservation laws, where the flux function is random. In particular, Burgers’ equation with uncertain flux function and two phase flows in a porous medium modeled by Buckley-Leverett equations with uncertain rock permeability and relative oil and water permeabilities are considered. An empirical convergence analysis of the numerical error will be investigated for both cases, and an additional example exposing the non-monotonicity of the MLMC-FVM estimator is presented for Buckley-Leverett equations.
1.7 Outline of the dissertation

In chapter 10, the acoustic wave equation (1.4) will be considered in one, two and three spatial dimensions, where the random linear fluxes $F_r$ are inferred from the random (uniformly as well as log-normally distributed) highly heterogeneous layered material coefficient $c(x,\omega)$. An empirical convergence analysis of the numerical error will be presented for one- and two-dimensional experiments. The probabilistic nature of the computational complexity estimates for this class of problems is analyzed in detail, as the wave speed (depending on the eigenvalues of the coefficient matrices in (1.42)) is random and can have a large statistical spread. Consequently, for popular explicit time stepping schemes, the time step size, being specified in terms of the wave speed due to the CFL condition, is random and hence the total computational work for the whole simulation is a random quantity. Crucial results of this chapter are the necessary and sufficient conditions on the statistical distribution properties of the random material coefficient, which ensure that the expected work of the MLMC-FVM methods is finite and asymptotically of the same order as of a single deterministic run of the underlying FVM scheme, maintaining the low computational complexity of the MLMC-FVM scheme.

Extending the results of chapter 9 in chapter 11 results for systems of nonlinear conservation laws, where the flux function is random, are presented. In particular, Euler equations with uncertain equation of state are considered, where uncertainty is inferred from the unknown constant of specific heats. Results from one and two dimensional numerical experiments are provided and analyzed.

A summary of the theoretical and numerical results, together with the concluding remarks and discussion, is provided in the closing chapter 12.
Introduction
2 Theory for random conservation laws

In this chapter we present the results for existence and uniqueness of random entropy solutions of (1.133). Such results were obtained by Mishra and Schwab [75, 79, 76] in the case of (linear and nonlinear) scalar \((m = 1)\) conservation laws on \(D = \mathbb{R}^d\) with random initial data and random flux functions, where the random entropy solutions were shown to exist and to satisfy certain stability estimates. The stochastic version of the scalar conservation law (1.25) on \(D = \mathbb{R}^d\) is given by

\[
\begin{align*}
    u_t(x, t, \omega) + \text{div}(f(u, \omega)) &= 0, \\
    u(x, 0, \omega) &= u_0(x, \omega), \\
    x &\in D, \quad t \in \mathbb{R}_+, \quad \omega \in \Omega.
\end{align*}
\]

(2.1)

In the next two sections, we present an overview of the results obtained in [75, 79, 76]. As in subsection 1.3.4, we restrict our analysis to Cauchy problems with \(D = \mathbb{R}^d\).

2.1 Scalar conservation laws with uncertain initial data

Firstly, we describe the assumptions on the random initial data \(u_0\).

2.1.1 Random initial data

Assume we are given \(u_0\) as an \(L^1(\mathbb{R}^d)\)-valued random field, i.e. a measurable map

\[u_0 : (\Omega, \mathcal{F}) \rightarrow (L^1(\mathbb{R}^d), \mathcal{B}(L^1(\mathbb{R}^d))).\]

(2.2)

We assume further that

\[u_0(\cdot, \omega) \in L^\infty(\mathbb{R}^d) \cap BV(\mathbb{R}^d) \quad \mathbb{P}\text{-a.s.},\]

(2.3)

which is to say that

\[\mathbb{P}(\{\omega \in \Omega : u_0(\cdot, \omega) \in (L^\infty \cap BV)(\mathbb{R}^d)\}) = 1.\]

Since \(L^1(\mathbb{R}^d)\) is separable, (2.2) is well-defined (see subsection 1.5.2) and we may impose for \(k \in \mathbb{N}\) the \(k\)-th moment condition

\[\|u_0\|_{L^k(\Omega, L^1(\mathbb{R}^d))} < \infty,\]

(2.4)

where \(L^k(\Omega, L^1(\mathbb{R}^d))\) is the Bochner space with respect to the probability measure \(\mathbb{P}\).
2.1.2 Existence and uniqueness result

By Theorem 1.3.11 the following result from [75] holds:

**Theorem 2.1.1.** Consider the scalar conservation law (2.1) on \( \mathbb{D} = \mathbb{R}^d \) with deterministic flux \( f(u) \) and random initial data \( u_0 : \omega \rightarrow L^1(\mathbb{R}^d) \) satisfying (2.3) and the \( k \)-th moment condition (2.4) for some \( k \in \mathbb{N} \). Then there exists a unique random entropy solution

\[
u : \Omega \ni \omega \rightarrow \mathbb{C}_b([0,T], L^1(\mathbb{R}^d))
\]
given by the pointwise \((\forall \omega \in \Omega)\) application of the solution operator \( S(t) \),

\[
u(\cdot, t, \omega) = S(t)u_0(\cdot, \omega), \quad \forall t > 0, \ \omega \in \Omega,
\]
such that for every \( k \geq m \geq 1 \) and for every \( 0 \leq t \leq T < \infty \),

\[
\|u\|_{L^k(\Omega, C([0,T], L^1(\mathbb{R}^d)))} \leq \|u_0\|_{L^k(\Omega, L^1(\mathbb{R}^d))},
\]

and such that for \( \mathbb{P}\text{-a.s.} \ \omega \in \Omega \), we have

\[
TV(S(t)u_0(\cdot, \omega)) \leq TV(u_0(\cdot, \omega)),
\]

\[
\|S(t)u_0(\cdot, \omega)\|_{L^1(\mathbb{R}^d)} \leq \|u_0(\cdot, \omega)\|_{L^1(\mathbb{R}^d)},
\]

**Proof.** The proof follows [75].

1. For \( \omega \in \Omega \), we define, motivated by Theorem 1.3.11 for \( \mathbb{P}\text{-a.e.} \ \omega \in \Omega \) a random function \( u(x, t, \omega) \) by

\[
u(\cdot, t, \omega) = S(t)u_0(\cdot, \omega), \quad \forall t > 0, \ \omega \in \Omega.
\]

By the properties of the non-linear solution mapping \( (S(t))_{t \geq 0} \), the random field (2.9) is well-defined; for \( \mathbb{P}\text{-a.e.} \ \omega \in \Omega \), \( u(\cdot, \omega) \) is a weak entropy solution of the scalar conservation law (2.1).

2. From Theorem 1.3.11 we obtain that \( \mathbb{P}\text{-a.s.} \), all bounds [1.38] - [1.41] hold. We proceed to check measurability of the mapping \( \Omega \ni \omega \rightarrow u(\cdot, \omega) = S(t)u_0(\cdot, \omega) \).

3. We have to show that for every \( 0 \leq t \leq T \), the mapping \( \Omega \ni \omega \rightarrow u(\cdot, t, \omega) = S(t)u_0(\cdot, \omega) \) is measurable, as an \( L^1(\mathbb{R}^d) \)-valued random variable. This, however, is a consequence of the fact that \( L^1(\mathbb{R}^d) \) is separable and that \( L^1(\mathbb{R}^d)^* \cong L^\infty(\mathbb{R}^d) \). Thus \( \mathbb{B}(L^1(\mathbb{R}^d)) \) is the smallest \( \sigma \)-algebra containing all subsets of \( L^1(\mathbb{R}^d) \) of the form

\[
\{v \in L^1(\mathbb{R}^d) : \varphi(v) \leq \alpha\} : \ \varphi \in L^\infty(\mathbb{R}^d), \ \alpha \in \mathbb{R}.
\]

Let now \( \alpha \in \mathbb{R}, \ \varphi \in L^\infty(\mathbb{R}^d) = L^1(\mathbb{R}^d)^* \) and \( t > 0 \). Consider then

\[
\{u(\cdot, t, \omega) : \varphi(u(\cdot, t, \omega)) \leq \alpha\} = \{S(t)u_0(\cdot, \omega) : \varphi(S(t)u_0(\cdot, \omega)) \leq \alpha\}.
\]
2.2 Scalar conservation laws with uncertain flux function

By continuity (1.40) of $S(t)$ in $L^1(\mathbb{R}^d)$, for every $0 < t \leq T < \infty$, $S(t)$ maps open balls of $L^1(\mathbb{R}^d)$ into open balls of $L^1(\mathbb{R}^d)$. Since $\mathcal{B}(L^1(\mathbb{R}^d))$ is the smallest $\sigma$-field on $L^1(\mathbb{R}^d)$ containing all subsets of the form (2.10), and since $u_0 \in L^0((\Omega, \mathcal{F}, \mathbb{P}), L^1(\mathbb{R}^d))$, we have for every $0 < t \leq T < \infty$ that

$$u(\cdot, t, \cdot) = S(t)u_0(\cdot, \cdot) \in L^0((\Omega, \mathcal{F}, \mathbb{P}), L^1(\mathbb{R}^d)).$$

4. To show (2.6), assume $u_0 \in L^k((\Omega, \mathcal{F}, \mathbb{P}), L^1(\mathbb{R}^d))$ for some $k \in \mathbb{N}$. Then, for every $0 \leq t \leq T < \infty$, we have

$$\int_\omega \|u(\cdot, t, \omega)\|_{L^1(\mathbb{R}^d)}^k \mathbb{P}(d\omega) = \int_\omega \|S(t)u_0(\cdot, \omega)\|_{L^1(\mathbb{R}^d)}^k \mathbb{P}(d\omega) \leq \int_\omega \|u_0(\cdot, \omega)\|_{L^1(\mathbb{R}^d)}^k \mathbb{P}(d\omega) = \|u_0\|_{L^k(\Omega, L^1(\mathbb{R}^d))}^k.$$

Hence

$$\|u_0\|_{L^k(\Omega, C([0,T], L^1(\mathbb{R}^d)))}^k = \int_\omega \left( \max_{0 \leq t \leq T} \|S(t)u_0(\cdot, \omega)\|_{L^1(\mathbb{R}^d)} \right) \mathbb{P}(d\omega) \leq \int_\omega \|S(t)u_0(\cdot, \omega)\|_{L^1(\mathbb{R}^d)}^k \mathbb{P}(d\omega) \leq \int_\omega \|u_0(\cdot, \omega)\|_{L^1(\mathbb{R}^d)}^k \mathbb{P}(d\omega) = \|u_0\|_{L^k(\Omega, L^1(\mathbb{R}^d))}^k.$$

This implies (2.6), (2.7) and (2.8) follow from (1.38), (1.39), (1.40).

Theorem 2.1.1 ensures the existence of (deterministic) $k$-th moments (1.126) of the random entropy solution $u(x, t, \omega)$ provided that $u_0 \in L^k((\Omega, \mathcal{F}, \mathbb{P}), L^1(\mathbb{R}^d))$.

2.2 Scalar conservation laws with uncertain flux function

In this section we extend the uniqueness and existence results for random entropy solutions of random scalar conservation laws (2.1) on $\mathbb{D} = \mathbb{R}^d$ to the cases where not only the initial data $u_0(x, \omega)$, but also the flux function $f = f(U, \omega)$ is random [79] [76].
Theory for random conservation laws

2.2.1 Bounded random flux

Noting that the space $E = C^1(\mathbb{R}, \mathbb{R}^d)$ is separable, we concentrate on the case of spatially homogeneous random flux functions and follow [76]. The definition of random flux for scalar conservation laws (2.1), i.e. for the case $m = 1$ in (1.133) with deterministic coefficients that we shall work with is the following,

**Definition 2.2.1.** A (spatially homogeneous) random flux for a scalar conservation law (2.1) is a $C^1(\mathbb{R}, \mathbb{R}^d)$-valued random field, i.e. a measurable mapping $(\Omega, \mathcal{F}) \rightarrow (C^1(\mathbb{R}, \mathbb{R}^d), \mathcal{B}(C^1(\mathbb{R}, \mathbb{R}^d)))$, $\Omega \ni \omega \mapsto f(\cdot, \omega)$.

A bounded random flux is a random flux whose $C^1(\mathbb{R}, \mathbb{R}^d)$-norm is bounded $\mathbb{P}$-a.s., i.e.

$$\exists 0 < B(f) < \infty : \|f(\omega, \cdot)\|_{C^1(\mathbb{R}, \mathbb{R}^d)} \leq B(f) \quad \mathbb{P}\text{-a.s.} \quad (2.11)$$

We observe that a bounded random flux has finite statistical moments (1.126) of any order $k \in \mathbb{N}$, since (2.11) implies $f \in L^k(\Omega, C^1(\mathbb{R}, \mathbb{R}^d))$, for all $k \in \mathbb{N}$. Of particular interest will be the centered second moment of a bounded random flux, representing the “two-point correlation in state-space” of the random flux $f$.

**Definition 2.2.2.** Let $f \in L^2(\Omega, C^1(\mathbb{R}, \mathbb{R}^d))$ be a bounded random flux as in Definition 2.2.1. Then its covariance function Cov$[f] \in C^1(\mathbb{R} \times \mathbb{R}, \mathbb{R}^{d \times d})$ is defined by

$$\text{Cov}[f](v, v') := \mathbb{E}[(f(v, \cdot) - \mathbb{E}[f(v, \cdot)]) \otimes (f(v', \cdot) - \mathbb{E}[f(v', \cdot)])], \quad \forall v, v' \in \mathbb{R}. \quad (2.12)$$

The two point correlation function of a bounded random flux allows, as is well-known in statistics, for spectral decompositions of the random flux in terms of eigenpairs of its covariance operator $\mathcal{C}_f$,

$$\mathcal{C}_f[\Phi](v) := \int_{\mathbb{R}} \text{Cov}[f](v, v')\Phi(v')dv', \quad \Phi \in L^2(\mathbb{R}),$$

which is a compact and self-adjoint integral operator on the space of square-integrable flux functions $L^2(\mathbb{R})$ with kernel function Cov$[f](v, v')$ defined in (2.12). We remark that our assumption of continuous differentiability of (realizations of) random flux functions entails linear growth of such fluxes as the state variables tend to infinity, i.e. as $|v| \rightarrow \infty$.

At first sight, this precludes considering the covariance operator on the space of square integrable flux functions. In [76], these integrability issues are circumvented for scalar conservation laws by truncating the state space to a bounded interval $[-R, R]$ with sufficiently large $R > 0$. By classical $L^\infty(\mathbb{R}^d)$ bounds on entropy solutions of scalar conservation laws, and for sufficiently large values of the flux cutoff $R$, any realization of the random scalar conservation law will “see” only the flux function for states which (in absolute value) are below the threshold values $R$. Accordingly, it suffices to consider the flux covariance operator only as integral operator on $L^2(-R, R)$, as introduced in [76].
2.2 Scalar conservation laws with uncertain flux function

2.2.2 Karhunen-Loève expansion of bounded random flux

Consider a bounded random flux $f(\omega, u)$ in the sense of Definition 2.2.1 and its covariance function $\text{Cov}[f]$ as defined in 2.2.2. For $0 < R < \infty$ we denote by $C^R_f$ the integral operator with continuously differentiable kernel $\text{Cov}[f](u, v)$, defined on $L^2(-R, R)$ by

$$C^R_f[\Phi](u) := \int_{|v| \leq R} \text{Cov}[f](u, v)\Phi(v)dv.$$  

(2.13)

As explained above, the covariance operator $C^R_f$ describes the covariance structure of the random flux on the set $[-R, R]$ of states of $u$. Given initial data $u_0 \in L^\infty(\mathbb{R}^d)$ with a-priori bound $\|u_0\|_{L^\infty(\mathbb{R}^d)} \leq R$, the unique entropy solution $S(t)u_0$ of the deterministic scalar conservation law (1.25) will, for all $t > 0$, take values in $[-\|u_0\|_{L^\infty(\mathbb{R}^d)}, \|u_0\|_{L^\infty(\mathbb{R}^d)}]$. For random flux and random initial data, therefore, we continue under the assumption

$$R > \text{ess sup}_{\omega \in \Omega} \|u_0(\omega, \cdot)\|_{L^\infty(\mathbb{R}^d)}.$$  

(2.14)

This ensures that $C^R_f$ will “capture” all possible states.

As $\text{Cov}[f] \in C^1(\mathbb{R} \times \mathbb{R}, \mathbb{R}^{d \times d})$, for every $0 < R < \infty$, the integral operator $C^R_f$ is a compact, self-adjoint operator on $L^2(-R, R)$. By the spectral theorem, it admits for every fixed value $0 < R < \infty$ a sequence $(\lambda^R_j, \Phi^R_j)_{j \geq 1}$ of real eigenvalues $\lambda^R_j$ (accumulating only at zero), which are assumed to be enumerated in decreasing magnitude and repeated according to multiplicity, and a corresponding set of eigenfunctions $\Phi^R_j$, 

$$(C^R_f \Phi^R_j)(u) = \lambda^R_j \Phi^R_j(u), \quad |u| \leq R.$$  

(2.15)

To exclude degeneracies, we shall assume throughout that the sequence $(\Phi^R_j)_{j \geq 1}$ is a complete, orthonormal base of $L^2(-R, R)$.

It follows from the continuous differentiability $\text{Cov}[f] \in C^1(\mathbb{R} \times \mathbb{R}, \mathbb{R}^{d \times d})$ and from the eigenvalue equation (2.15), that $\Phi^R \in C^1([-R, R], \mathbb{R}^{d \times d})$ [76], hence, any bounded random flux $f(\omega, u)$ admits, for every fixed $0 < R < \infty$, a Karhunen-Loève expansion

$$f(\omega, u) = \tilde{f}(u) + \sum_{j \geq 1} Y^R_j(\omega) \Phi^R_j(u), \quad |u| \leq R,$$  

(2.16)

which converges in $L^2(\Omega, L^2(-R, R^d))$. In (2.16), $\tilde{f}(u) = \mathbb{E}[f(\cdot, u)]$ is the nominal flux, $(Y^R_j)_{j \geq 1}$ is a sequence of pairwise uncorrelated random variables given by

$$\forall j \in \mathbb{N} : \quad Y^R_j(\omega) := \sqrt{\lambda^R_j} \int_{|v| < R} f(\omega; v)\Phi^R_j(v)dv.$$  

(2.17)

and the principal components of the random flux are given by

$$\forall j \in \mathbb{N} : \quad \Psi^R_j(u) := \frac{1}{\sqrt{\lambda^R_j}} \Phi^R_j(u).$$

We remark that under suitable smoothness conditions on $\text{Cov}[f]$, the convergence of the expansion (2.16) is pointwise with respect to $u$ and the convergence rate increases with increasing smoothness of $\text{Cov}[f]$ (see, e.g. [76]).
2.2.3 Existence and uniqueness result

The following well-posedness result for random entropy solutions of (2.1) holds \[76\], Theorem 2.2.3. Consider the random scalar conservation law (2.1) on \(D = \mathbb{R}^d\) with spatially homogeneous, bounded random flux \(f : \Omega \to C^1(\mathbb{R}, \mathbb{R}^d)\) as in Definition 2.2.1 and with (independent of \(f\)) random initial data \(u_0 : \Omega \to L^1(\mathbb{R}^d)\) satisfying (2.3) and the \(k\)-th moment condition (2.4) for some integer \(k \geq 2\). In particular, then, there exists a constant \(\bar{R} < \infty\) such that

\[
\|u_0(\omega, \cdot)\|_{L^\infty(\mathbb{R}^d)} \leq \bar{R}\ \ \mathbb{P}\text{-a.e. } \omega \in \Omega.
\] (2.18)

Assume moreover that the random flux admits the representation (2.16) with (2.17) where the Lipschitz-continuous scaled flux components \(\frac{\Psi_j}{R}\) have Lipschitz constants \(B^R_j\) such that \(B^R := (B^R_j)_{j \geq 1} \in \ell^1(\mathbb{N})\) with some \(R \geq \bar{R}\) as in (2.18).

Then there exists a unique random entropy solution

\[ u : \Omega \ni \omega \to C_b([0, T], L^1(\mathbb{R}^d)), \]

which is “pathwise”, i.e. for \(\mathbb{P}\text{-a.e. } \omega \in \Omega\), described in terms of a nonlinear mapping \(S(\omega, t)\) depending only on the random flux,

\[ u(\cdot, t, \omega) = S(\omega, t)u_0(\cdot, \omega), \quad t > 0, \ \mathbb{P}\text{-a.e. } \omega \in \Omega, \] (2.19)

such that for every \(k \geq m \geq 1\) and for every \(0 \leq t \leq T < \infty\) holds

\[
\|u\|_{L^k(\Omega, C([0, T], L^1(\mathbb{R}^d)))} \leq \|u_0\|_{L^k(\Omega, L^1(\mathbb{R}^d))}, \quad (2.20)
\]

\[
\|S(\omega, t) u_0(\cdot, \omega)\|_{L^1 \cap L^\infty(\mathbb{R}^d)} \leq \|u_0(\cdot, \omega)\|_{L^1 \cap L^\infty(\mathbb{R}^d)}, \quad (2.21)
\]

and such that we have \(\mathbb{P}\text{-a.s.}\)

\[
TV(S(\omega, t) u_0(\cdot, \omega)) \leq TV(u_0(\cdot, \omega)), \quad (2.22)
\]

and, with \(\bar{R}\) as in (2.18),

\[
\sup_{0 \leq t \leq T} \|u(\cdot, t, \omega)\|_{L^\infty(\mathbb{R}^d)} \leq \bar{R}\ \ \mathbb{P}\text{-a.e. } \omega \in \Omega. \quad (2.23)
\]

The above theorem establishes that random scalar conservation laws (2.1) are well-posed (in several space dimensions) for uncertain initial data as well as for random fluxes. An extension of these definitions and results is also possible to bounded random source terms \(S(u, \omega)\) as well as to spatially inhomogeneous flux functions \(f(\omega, x, u)\), provided their dependence on the spatial coordinate is continuously differentiable: they are measurable mappings from \((\Omega, \mathcal{F})\) into \((E, \mathcal{B}(E))\) where \(E = C^1(\mathbb{R}^{d+1}, \mathbb{R}^d)\).

It is straightforward to extend the principal component representation and the notion of covariance operator to flux functions for hyperbolic systems (1.133) with \(m > 1\): in this
2.3 Linear hyperbolic systems with uncertain coefficients

In the case, the covariance operator is to be interpreted as an abstract, symmetric bilinear form on the space $C^1(\mathbb{R}^m, \mathbb{R}^{m \times d})$ whose kernel coincides with a symmetric, fourth order tensor on the state space $\mathbb{R}^m$. Spectral decompositions analogous to (2.16) for $\mathbb{R}^{m \times d}$ matrix-valued random flux functions which arise in (1.133) can then be defined in an analogous fashion. However, due to the lack of bounds like (2.23), the approach of [76] can not be directly applied for the mathematical investigation of non-linear random hyperbolic systems (1.133) at present. Nevertheless, the spectral expansion (2.16) for (1.133) may be a useful tool to achieve a parsimonious parametric representation of a given random flux also in the numerical treatment of random hyperbolic systems (1.133).

For linear hyperbolic systems of conservation laws, analogous bounds to (2.23) are available, and hence the theory could be extended. For this case, we present an even more general uniqueness and existence result in the next section, where a much larger class of fluxes are allowed, including the random fluxes $F$ where the bound (2.11) is violated, i.e.

$$\|F\|_{L^\infty(\Omega; C^1(\mathbb{R}^m, \mathbb{R}^{m \times d}))} = \infty.$$  

(2.24)

2.3 Linear hyperbolic systems of conservation laws with uncertain coefficients

The stochastic linear system of hyperbolic conservation laws is a special case of (1.133) with linear flux functions $F_r(c(x, \omega), U, \omega) = A_r(x, \omega)U$, initially introduced in (1.6),

$$\left\{ \begin{array}{l}
U_t(x, t, \omega) + \sum_{r=1}^{d} A_r(x, \omega) \frac{\partial}{\partial x_r} U = S(x, t, \omega), \\
L_0 U(x, t)|_{\partial D} = g(x, t), \\
U(x, 0, \omega) = U_0(x, \omega),
\end{array} \right. \quad \forall (x, t) \in D \times \mathbb{R}_+, \quad \forall \omega \in \Omega,$$  

(2.25)

where the matrices $A_r$ are $(C^1(D))^{m \times m}$-valued random fields,

$$A_r : (\Omega, \mathcal{F}) \rightarrow (C^1(D))^{m \times m}, \quad \Omega \ni \omega \mapsto A_r(x, \omega).$$

Based on Theorem 1.3.12 we obtain the following well-posedness result for (2.25).

**Theorem 2.3.1.** In the linear system of conservation laws (2.25) on $D = \mathbb{R}^d$, assume that the following holds for some $k \in \mathbb{N} \cup \{0, \infty\}$:

(A1) (2.25) is strongly hyperbolic with $K(\omega) < \infty$ in (1.14) for $\mathbb{P}$-a.e. $\omega \in \Omega$, and

$$\tilde{K}_k = \|K(\omega)\|_{L^k(\Omega, \mathbb{R})} < \infty,$$  

(2.26)
Theory for random conservation laws

(A2) there exist non-negative integers $r_0, r_S \in \mathbb{N} \cup \{0, \infty\}$ and $r_A \in \mathbb{N} \cup \{\infty\}$ such that for all $p \in \{2, \infty\}$ there holds

$$U_0 \in L^k(\Omega, W^{r_0, p}(\mathbb{R}^d)),
S \in L^k(\Omega, W^{r_S, p}(\mathbb{R}^d)),
A_r \in L^0(\Omega, (C^{r_A}(\mathbb{R}^d))^{m \times m}),$$

where $L^0(\Omega, V)$ denotes measurable $V$-valued random fields.

(A3) each random field $A_r$, $r = 1, \ldots, d$, is independent of $U_0$ and $S$ on $(\Omega, \mathcal{F}, \mathbb{P})$.

Then, for $T < \infty$, (2.25) admits a unique random weak solution

$$U : \Omega \rightarrow C([0, T], L^2(\mathbb{R}^d)), \quad \omega \mapsto U(\cdot, \omega), \quad \forall \omega \in \Omega,$$

where $U(\cdot, \omega)$ is the solution to the deterministic system (1.42). Moreover, $\forall t \in [0, T],
\|U(\cdot, t, \omega)\|_{L^2(\mathbb{R}^d)} \leq \|K(\omega), U_0(\cdot, \omega), S(\cdot, \omega), t\|_{L^2(\mathbb{R}^d)}, \quad \mathbb{P}\text{-}a.s.,
\|K, U_0, S, t\|_{L^k(\Omega, C([0, T], L^2(\mathbb{R}^d)))} \leq \|K, U_0, S, t\|_{L^k(\Omega, L^2(\mathbb{R}^d))},$$

with the notation

$$\|K, U, S, t\|_{L^k(\Omega, V)} = \|K, U, S, t\|_{L^k(\Omega, L^2(\mathbb{R}^d))}.$$ 

Furthermore, the pathwise regularity of $U$ is inherited from Theorem 1.3.13, i.e. for

$$U \in L^k(\Omega, C([0, T], W^{r, p}(\mathbb{R}^d))).$$

Proof. The main ideas for proving well-posedness, measurability and the bounds stem from the proof of Theorem 2.1.1.

1. By Theorem 1.3.12 the random field in (2.28) is well-defined for $\mathbb{P}$-a.e. $\omega \in \Omega$. Furthermore, for $\mathbb{P}$-a.e. $\omega \in \Omega$, $U(\cdot, \omega)$ is a weak entropy solution of (1.42), satisfying (1.22) and the weak version of the entropy equality, obtained by considering the equality in (1.30) instead of the inequality.

2. $\forall t \in [0, T], \forall j = 1, \ldots, m$, we verify the measurability of the component map $\Omega \ni \omega \mapsto U_j(\cdot, t, \omega) \in L^2(\mathbb{R}^d)$. Since $L^2(\mathbb{R}^d)$ is a separable Hilbert space, $\mathcal{B}(L^2(\mathbb{R}^d))$ is the smallest $\sigma$-algebra containing all subsets

$$\{v \in L^2(\mathbb{R}^d) : \varphi(v) \leq \alpha\} : \varphi \in L^2(\mathbb{R}^d), \alpha \in \mathbb{R}.$$

For a fixed $\alpha \in \mathbb{R}$, $\varphi \in L^2(\mathbb{R}^d)$, consider the set

$$\{U_j(\cdot, t, \omega) : \varphi(U_j(\cdot, t, \omega)) \leq \alpha\}.$$
By continuity (1.45) in $L^2(\mathbb{R}^d)$, since

$$U_0, S \in L^0(\Omega, L^2(\mathbb{R}^d))$$

and

$$A_r \in L^0(\Omega, L^2(\mathbb{R}^d)^m),$$

we obtain

$$U_j(\cdot, t, \cdot) \in L^0(\Omega, L^2(\mathbb{R}^d)), \quad \forall 0 \leq t \leq T.$$
Theory for random conservation laws

As discussed in subsection 1.3.4, an alternative notion of solutions to (1.133), potentially capable of extending the notion of random entropy solutions in Definition 1.6.1 is the entropy measure valued solutions (EMVS). Even though EMVS were recently advocated to be the appropriate notion of solution for deterministic nonlinear hyperbolic system of conservation laws in several space dimensions [37], possible extensions of EMVS to random systems (1.133) of hyperbolic conservation laws in several dimensions (i.e. \( m > 1 \) and \( d > 1 \)) have not yet been considered.
3 Monte Carlo Finite Volume method

We aim to design an efficient Monte Carlo (MC) type scheme for approximating solution statistics for the stochastic conservation law (1.133). This entails discretizing spatio-temporal space \( D \times [0, T] \) (for bounded domains \( D \) and finite time horizon \( T < \infty \)) as well as the probability space \((\Omega, \mathcal{F}, \mathbb{P})\). In the first section, we begin with the MC discretization of the probability space and provide convergence results in the case of scalar and linear hyperbolic systems of conservation laws. In the second section, the MC sampling method is combined with FVM methods analyzed in section 1.4 to obtain approximations of MC samples and the bounds for the combined MC and FVM errors are given, again only for the case of scalar and linear hyperbolic systems of conservation laws. In the last section, some computational limitations of MC-FVM method are outlined.

3.1 Monte Carlo method

We view the Monte-Carlo Method as a “discretization” in the stochastic space of the hyperbolic system of conservation laws (1.133) with random input data

\[
\hat{\mathbf{I}}(\omega) := \{\hat{U}_0(\cdot, \omega), \hat{c}(\cdot, \omega), \hat{F}(\cdot, \cdot, \omega), \hat{S}(\cdot, \cdot, \omega)\}.
\]  

(3.1)

Assumption 3.1.1 (Random input data). Assume the following holds for (3.1):

- The random initial data \( u_0(x, \omega) \) for scalar conservation laws (2.1) satisfies (2.2) - (2.3), and in particular, (2.18). For linear hyperbolic systems of conservation laws (2.25), we assume (2.27) for the random initial data \( U_0(x, \omega) \). Additionally, we assume (2.4), i.e. the existence of k-th moments of \( U_0 \) for some \( k \in \mathbb{N} \), which also follows from (2.18).

- The random source term \( S(x, \omega) \) satisfies (1.44) P-a.s. and (2.27).

- For linear systems (2.25), random coefficients \( c(x, \omega) \) and linear deterministic flux functions \( F_r(c, U) \) are such that the resulting matrices \( \mathbf{A}_r(x, \omega) \) in (2.25) define a hyperbolic system as in Definition 1.3.1, i.e. satisfy (1.13) - (1.14) P-a.s. We also assume k-th moment condition (2.26) for \( K(\omega) \) and regularity (2.27).

- For non-linear scalar conservation laws (2.1), random flux functions \( F_r(U, \omega) \) are assumed to be bounded random flux functions as in (2.11), satisfying (2.16) with some \( R \geq \bar{R} \).
Monte Carlo Finite Volume method

The *MC approximation of $E[U]$* is defined as follows: given $M$ independent, identically
distributed samples $\hat{I}_i$ of random input data $\hat{I}$ for $i = 1, \ldots, M$, i.e.
\[
\hat{I}(\omega) := \{\hat{U}_i(\cdot, \cdot, \omega), \hat{c}_i(\cdot, \cdot, \omega), \hat{F}_i(\cdot, \cdot, \cdot, \cdot, \omega), \hat{S}_i(\cdot, \cdot, \cdot, \cdot, \omega)\},
\]
the MC estimate of $E[U(\cdot, t, \cdot)]$ at time $t$ is given by
\[
E_M[U(\cdot, t)] := \frac{1}{M} \sum_{i=1}^{M} U_i(\cdot, t), \quad (3.2)
\]
where $U_i(\cdot, t)$ denotes the $M$ unique entropy solutions of the $M$ Cauchy problems (2.1)
or (2.25) with input data $\hat{I}_i$. Due to lack of existence and uniqueness results, estimate
(3.2) is *not* well-defined for general non-linear hyperbolic random systems (1.133).

Higher moments of $U$ (*$k$*-point correlation functions $M^k[U] = E[U^{(k)}]$) can be estimated
analogously,
\[
E_M[U(\cdot, t)^{(k)}] := \frac{1}{M} \sum_{i=1}^{M} (\bar{U}_i)^{(k)}(\cdot, t). \quad (3.3)
\]

We note that such entropy solutions, as pointed out in chapter 2, might not be available
for a general system conservation laws. However, unique entropy solutions *are* available
for *scalar case* (2.1) and for *linear hyperbolic systems* (2.25). Hence, for a given non-linear
system of conservation laws, we *assume* that for a given problem an entropy solution
exists and bounds analogous to (1.45) - (1.47) are valid.

### 3.1.1 MC convergence for scalar conservation laws

We observe that by (1.40), for every $M$ and for every $0 < t < \infty$,
\[
\|E_M[u(\cdot, t, \omega)]\|_{L^1(D)} = \left\| \frac{1}{M} \sum_{i=1}^{M} S(t) \bar{u}_i(\cdot, \omega) \right\|_{L^1(D)} \leq \frac{1}{M} \sum_{i=1}^{M} \|S(t) \bar{u}_i(\cdot, \omega)\|_{L^1(D)} \leq \frac{1}{M} \sum_{i=1}^{M} \|\bar{u}_i(\cdot, \omega)\|_{L^1(D)}. \quad (3.4)
\]

Using the i.i.d. property of the samples $\{\bar{u}_i\}_{i=1}^{M}$ of the random initial data $u_0$, Lemma
1.5.2 and the linearity of the expectation $E[\cdot]$, we obtain the bound
\[
E[\|E_M[u(\cdot, t)]\|_{L^1(D)}] \leq E[\|u_0\|_{L^1(D)}] = \|u_0\|_{L^1(\Omega, L^1(D))} < \infty. \quad (3.5)
\]

As $M \to \infty$, the MC estimates (3.2) converge and the following result from [75] holds.
3.1 Monte Carlo method

Theorem 3.1.2. Assume that in (2.1) the random input data satisfies Assumption 3.1.1, i.e. random initial data \( u_0(x,\omega) \) satisfies (2.2) - (2.3), and in particular, (2.18), whereas bounded random flux functions \( f(u,\omega) \) as in (2.11) satisfy (2.16) with some \( R \geq \bar{R} \). Then the MC estimates \( E_M[u(\cdot, t)] \) in (3.2) converge as \( M \to \infty \), to \( M^1(u(\cdot, t)) = E[u(\cdot, t)] \) and, for any \( M \in \mathbb{N}, 0 < t < \infty \), there holds the error bound

\[
\| E[u(\cdot, t, \cdot)] - E_M[u(\cdot, t, \cdot)](\omega) \|_{L^2(\Omega; L^1(D))} \leq M^{-\frac{1}{2}} \| u_0 \|_{L^2(\Omega; L^1(D))}. \tag{3.6}
\]

For a proof of Theorem 3.1.2 we refer to [75].

So far, we addressed the MC estimation of the mean field or first moment. A similar result holds for the MC estimates of the \( k \)-th moment

\[
M^k[u] := E[u^{(k)}] \in (L^1(D))^{(k)}. \tag{3.126}
\]

Proposition 3.1.3. Assume that in (2.1) the random input data satisfies Assumption 3.1.1, i.e. random initial data \( u_0(x,\omega) \) satisfies (2.2) - (2.3), and in particular, (2.18), whereas bounded random flux functions \( f(u,\omega) \) as in (2.11) satisfy (2.16) with some \( R \geq \bar{R} \). Assume further that for some \( k \in \mathbb{N} \) holds \( u_0 \in L^{2k}(\Omega; L^1(D)) \). Then the MC estimates \( E_M[u(\cdot, t)^{(k)}] \) as defined in (3.3) converge to the \( k \)-th moment (or spatial \( k \)-point correlation function) \( M^k[u](t) \) defined in (1.126). Moreover, the error bound holds

\[
\| M^k[u](t) - E_M[(u(\cdot, t, \omega)^{(k)})] \|_{L^2(\Omega; L^1(D^{(k)}))} \leq M^{-1/2} \| u_0 \|^k_{L^{2k}(\Omega; L^1(D))}. \tag{3.7}
\]

For a proof of the Proposition 3.1.3 we refer to [75].

3.1.2 MC convergence for linear hyperbolic systems of conservation laws

We observe that by (2.29), for every \( M \) and for every \( 0 < t < \infty \),

\[
\| E_M[U] \|_{L^2(D)} = \frac{1}{M} \left\| \sum_{i=1}^M \hat{U}^i \right\|_{L^2(D)} \leq \frac{1}{M} \sum_{i=1}^M \| \hat{U}^i \|_{L^2(D)} \tag{3.8}
\]

\[
\leq \frac{1}{M} \sum_{i=1}^M \| K^i, \hat{U}^i_0, \hat{S}^i, t \|_{L^2(D)}.
\]

Using the i.i.d. property of the samples \( \{i\}^M \) of the random input data \( I \), Lemma 1.5.2 and the linearity of the expectation \( E[\cdot] \), we obtain the bound

\[
E[\| E_M[U(\cdot, t, \cdot)] \|_{L^2(D)}] \leq E \left[ K(\omega) \left( \| U_0(\cdot, \omega), S(\cdot, \omega), t \|_{L^2(D)} \right) \right]
\]

\[
= \tilde{K}_1 \| K, U_0, S, t \|_{L^1(\Omega; L^2(D))}
\]

\[
\leq \tilde{K}_1 \left( \| U_0 \|_{L^1(\Omega; L^2(D))} + t \| S \|_{L^1(\Omega; L^2(D))} \right) < \infty.
\]

As \( M \to \infty \), the MC estimates (3.2) converge and the following result holds for (2.25).
Monte Carlo Finite Volume method

**Theorem 3.1.4.** Assume the hypothesis of Theorem 2.3.1 is satisfied with $k \geq 2$, i.e. the second moments of the random initial data $U_0$, source $S$ and $K$ exist. Then, the MC estimates $E_M[U(\cdot,t,\omega)]$ in (3.2) converge to $M^k[U] = E[U]$ as $M \to \infty$. Furthermore,

$$
\|E[U(t)] - E_M[U(t)](\omega)\|_{L^2(\Omega,L^2(D))} \leq M^{-\frac{1}{2}} \|K, U_0, S, t\|_{L^2(\Omega,L^2(D))},
$$

(3.10)

with the corresponding norm defined by (1.43) and (2.31).

**Proof.** As is customary in the convergence analysis of MC methods, the $M$ samples $\{\tilde{I}^i\}_{i=1}^M$ are interpreted as realizations of $M$ independent “copies” of random input fields

$$
I(\omega) = \{U(\cdot,\omega), S(\cdot,\omega), K(\omega)\}
$$
on the probability space $(\Omega, \mathcal{F}, \mathbb{P})$, i.e. $\tilde{I} = \tilde{I}^{i}(\omega)$. The corresponding unique entropy solutions $\hat{U}^i(\cdot,t,\omega), \forall 0 \leq t \leq T$, are also independent in the space $L^2(\Omega, C([0,T],L^2(D)))$ by (2.28): images $\hat{U}^i(\cdot,t,\omega)$ of any two i.i.d. realizations of $I(\omega)$ under (2.28) are strongly measurable $C([0,T],L^2(D))$-valued random functions by $L^2(D)$-contractivity (1.46), and hence are independent random fields. By Lemma 1.5.2 and by continuity (2.29), the mapping $\omega \to \|U(\cdot,t,\omega)\|_{L^2(D)}$ is measurable. Hence,

$$
E \left[ \|E[U] - E_M[U](\omega)\|^2_{L^2(D)} \right] = \frac{1}{M^2} \mathbb{E} \left[ \sum_{i=1}^M \|E[U] - \hat{U}^i(\omega)\|^2_{L^2(D)} \right] \\
= \frac{1}{M} \mathbb{E} \left[ \|E[U] - \hat{U}^i(\omega)\|^2_{L^2(D)} \right] = \frac{1}{M^2} \mathbb{E} \left[ \|E[U]\|^2_{L^2(D)} - \|E[U]\|^2_{L^2(D)} \right] \\
\leq \frac{1}{M} \mathbb{E} \left[ \|E[U]\|^2_{L^2(D)} \right].
$$

Using (2.29), assumption 3 and notation $U(t) = U(\cdot,t,\omega)$, we deduce

$$
E \left[ \|E[U]\|^2_{L^2(D)} \right] \leq E \left[ \|K, U_0, S, t\|^2_{L^2(D)} \right] \leq \tilde{K}^2 \mathbb{E} \left[ \|1, U_0, S, t\|^2_{L^2(D)} \right] < \infty,
$$

hence we finally arrive at

$$
E \left[ \|E[U(t)] - E_M[U(t)\](\omega)\|^2_{L^2(D)} \right] \leq M^{-1} E \left[ \|K, U_0, S, t\|^2_{L^2(D)} \right],
$$

which implies (3.10) upon taking square roots.

Analogous to scalar case considered in subsection 3.1.1, similar result holds for the MC estimates of the $k$-th moment $M^k[U] := E[U^{(k)}] \in (L^2(\mathbb{R}^d))^{(k)}$.

**Proposition 3.1.5.** Assume that for some $k \in \mathbb{N}$, the hypothesis of Theorem 2.3.1 is satisfied with statistical twice higher regularity, i.e. the $2k$-th moments of the random initial data $U_0$, source $S$ and $K$ exist. Then, the MC estimates $E_M[U(\cdot,t,\omega)^{(k)}]$ in (3.2) converge to $M^k[U] = E[U^{(k)}]$ as $M \to \infty$. Furthermore,

$$
\|M^k[U] - E_M[U(t)^{(k)}](\omega)\|_{L^2(\Omega,L^2(D)^{(k)})} \leq M^{-\frac{1}{2}} \|K, U_0, S, t\|_{L^{2k}(\Omega,L^2(D))}^k.
$$

(3.11)
3.2 Combination of Monte Carlo and Finite Volume Method

Proof. The assumption that \( U_0, S \in L^{2k}(\Omega, L^2(D)) \) implies, by Theorem 2.3.1 that \( U \in L^{2k}(\Omega, C([0,T], L^2(D))) \), i.e. the hypothesis (1.128) of Lemma 1.5.5 is satisfied with \( r = 2k \). Then, by Lemma 1.5.5 with \( E = L^2(D) \), we have that for any choice of time instances \( 0 < t_1, ..., t_k < \infty \) and for \( \mathbb{P}\)-a.e. \( \omega \in \Omega \), the spatial \( k \) point correlation

\[
\bigotimes_{r=1}^{k} U(\cdot, t_r, \omega) \in (L^2(D))^k
\]

of the random entropy solution \( U(\cdot, t, \omega) \) is well-defined in \( L^2(\Omega, L^2(D))^k \). Hence the proof of Theorem 3.1.4 directly applies to \( (U(\cdot, t, \omega))^{(k)} \). \qed

3.2 Combination of Monte Carlo and Finite Volume Method

So far, we considered the MC method under the assumption that the entropy solutions \( U^i(x, t, \omega) \) for the system of conservation laws (1.133) with the input data samples \( \hat{I}^i \) are available exactly. In practice, however, numerical approximations of \( \hat{U}^i \) must be computed by FVM as described in section 1.4.

3.2.1 Assumptions for a Finite Volume scheme

Before we proceed with the definition of the MC-FVM scheme, we assume that an abstract FVM scheme (1.49) for a scalar conservation law (1.25) satisfies the following,

**Assumption 3.2.1** (FVM for scalar conservation laws). We assume, that under CFL condition (1.51), the approximate FVM solution \( u_T(x,t) \) of an abstract FVM scheme (1.49) converges to the unique entropy solution \( u \) of the scalar conservation law (1.25). Furthermore, assuming a certain class of boundary conditions that do not introduce additional energy into the solution, FVM approximation \( u_T(x,t) \) satisfies the inequality

\[
\|u_T(\cdot,t)\|_{L^1(D)} \leq \|u_0\|_{L^1(D)},
\]

and the approximation error converges (as \( \Delta x \to 0 \)) with rate \( 0 < s \leq 1 \), i.e., there exist \( C_0, C_s > 0 \) independent of \( \Delta x = \Delta x(T) \) such that, as \( \Delta x(T) \to 0 \), the following holds,

\[
\|u_0 - u_T^0\|_{L^1(D)} \leq C_0 \Delta x^s |u_0|_{TV(D)},
\]

\[
\|u(\cdot,t) - u_T(\cdot,t)\|_{L^1(D)} \leq \|u_0 - u_T^0\|_{L^1(D)} + C_s \Delta x^s |u_0|_{TV(D)}.
\]

Likewise, we assume that an abstract FVM scheme (1.49) for a linear system of conservation laws (1.42) satisfies the following,
Assumption 3.2.2 (FVM for linear hyperbolic systems of conservation laws). We assume, that under CFL condition \((1.51)\), the approximate FVM solution \(U_T^0\) of an abstract FVM scheme \((1.49)\) converges to the unique entropy solution \(U\) of the linear system of conservation laws \((1.42)\). Furthermore, assuming a certain class of boundary conditions that do not introduce additional energy into the solution, FVM approximation \(U_T^0\) satisfies the energy inequality

\[
\|U_T(\cdot, t)\|_{L^2(D)} \leq K\|U_T^0, S_T, t\|_{L^2(D)},
\]

and the approximation error converges (as \(\Delta x \to 0\)) with rate \(s > 0\), i.e., there exists \(C_0, C_1 > 0\) independent of \(\Delta x = \Delta x(T)\) such that, as \(\Delta x(T) \to 0\), the following holds,

\[
\|U_0 - U_T^0\|_{L^2(D)} \leq C_0 \Delta x^s \|U_0\|_{H^s(D)},
\]

\[
\|S - S_T\|_{L^2(D)} \leq C_0 \Delta x^s \|S\|_{H^s(D)},
\]

\[
\|U(\cdot, t) - U_T(\cdot, t)\|_{L^2(D)} \leq \|U_0 - U_T^0, S - S_T, t\|_{L^2(D)} + C_1 \Delta x^s t K \|U_0, S, t\|_{H^s(D)}.
\]

Here, \(H^s(D)\) denotes the Hilbert space \(W^{s,2}(D)^m\) of \(s\)-times weakly differentiable (equivalence classes of) vector functions with (component-wise) \(L^2(D)\)-integrable weak derivatives.

Assumptions 3.2.1 and 3.2.2 are satisfied by many standard FVM (for small \(s\)) schemes with periodic, Neumann (outflow) and reflective boundary conditions, introduced in subsection 1.4.9; we refer to [49, 50, 53, 66, 119] and the references therein for further details. In particular, the convergence estimate \((3.14)\) is known to hold for first-order FVM schemes introduced in the first part of section 1.4 by results of Kusznetsov (see, e.g. [30]) with \(s = 1/2\). We also assume \(s = 1\) for second order schemes introduced in the second part of section 1.4. In general, for \(q\)-th order (formally) accurate schemes, \(q \in \mathbb{N}\), the convergence estimates \((3.14)\) and \((3.17)\) hold \([49, 50, 53, 66, 119]\) with

\[
s = \begin{cases} \min\{q/2, 1\}, & \text{if } \bar{r} = 0, \\ \min\{q, \bar{r}\} & \text{if } \bar{r} \in \mathbb{N}, \end{cases}
\]

meaning that full convergence order \(s = q\) is achieved for sufficiently smooth solutions with \(\bar{r} \geq q\) in \((1.44)\) and \((2.27)\), whereas irregular solutions with shocks (\(\bar{r} = 0\)) converge with order \(s \leq 1\), equal to only half of the formal order \(q\), resulting in \(s = 1/2\) or \(s = 1\).

3.2.2 Definition of the MC-FVM scheme

We next define and analyze the MC-FVM scheme. It is based on the straightforward idea of generating, possibly in parallel, independent samples of the random input data \(I(\omega)\) and then, for each sample \(I^i\) of the random input data, performing one FVM simulation. The MC-FVM scheme for the MC estimation of the mean of the random entropy solutions is the following.
3.2 Combination of Monte Carlo and Finite Volume Method

**Definition 3.2.3 (MC-FVM Scheme).** The Monte Carlo Finite Volume Method, abbreviated by MC-FVM, consists of the following three main steps:

1. **Sample:** We draw \( M \) independent identically distributed (i.i.d.) input data samples \( \hat{I}^i \) with \( i = 1, 2, \ldots, M \) from the random input fields \( I(\omega) \); then approximate the domain dependent random fields, such as initial data \( \hat{U}_0 \), source \( \hat{S}^i \) and coefficient \( \hat{c}^i \) by piecewise constant functions obtained from cell averaging on a given mesh \( T \) of computational domain \( D \), as described in [section 1.4]. Denote the resulting approximated realization of random input data by \( I_T \).

2. **Solve:** For each approximated realization \( I_T \) of random input data \( I(\omega) \), the underlying balance law (1.1) is solved numerically by the Finite Volume Method (1.49) on mesh \( T \) with mesh width \( \Delta x = \Delta x(T) \) up to time \( T = t^n \). We denote the solutions by \( U_{T}^{i,n}(x) = U_T(x,t^n) \).

3. **Estimate Statistics:** We estimate the expectation

\[
E[U(\cdot,t,\omega)] = M^1[U(\cdot,t,\omega)]
\]

of the random solution field \( U(\cdot,t,\omega) \) at time \( t = t^n \) with the sample mean (ensemble average) of the approximate solutions,

\[
E_M[U_T^n] = \frac{1}{M} \sum_{i=1}^{M} U_T^{i,n}(\cdot) . \tag{3.19}
\]

For \( k \geq 2 \), the \( k \)-th moment (or \( k \)-point correlation function)

\[
M^k[U(\cdot,t)] = E[U(\cdot,t)^{(k)}]
\]

defined in (1.126) is estimated by

\[
E_M^{(k)}[U_T^n] = \frac{1}{M} \sum_{i=1}^{M} (U_T^{i,n}(\cdot) \otimes \cdots \otimes U_T^{i,n}(\cdot)) . \tag{3.20}
\]

More generally, for given time instances \( t_1, \ldots, t_k \in [0,T], T < \infty \), the statistical FVM estimate of \( M^k[U](t_1,\ldots,t_k) \) is given by

\[
E_M^{(k)}[U_T](t_1,\ldots,t_k) := \frac{1}{M} \sum_{i=1}^{M} (U_T(\cdot,t_1) \otimes \cdots \otimes U_T(\cdot,t_k)) . \tag{3.21}
\]

The above algorithm is quite simple to implement. We remark that step 1 requires a (pseudo) random number generator. In step 2, any standard (high-order) finite volume scheme can be used. Hence, existing code for FVM can be used and there is no need to rewrite FVM code. Furthermore, the only (data) interaction between different samples is in step 3, when ensemble averages are computed. Thus, the MC-FVM is non-intrusive as well as easily parallelizable.
Monte Carlo Finite Volume method

3.3 Convergence analysis of MC-FVM

The error of the MC-FVM estimate (3.19) is bounded by two contributions: a (statistical) sampling error and a (deterministic) discretization error. We express the asymptotic efficiency of this approach (in terms of overall error versus work). It will be seen that the efficiency of the MC-FVM is, in general, inferior to that of the deterministic scheme, as in (3.14) and (3.17). The present analysis will constitute a key technical tool in our subsequent development and analysis of the multi-level MC-FVM (“MLMC-FVM” for short) which does not suffer from this drawback.

3.3.1 Error estimates for scalar conservation laws

Firstly, we provide rigorous error estimates from [75, 76] for the convergence of $E_M[u^0_T(\cdot)]$ to the $E[u(\cdot,t^n)]$ for arbitrary $k \in \mathbb{N}$ in the case of random scalar conservation laws (2.1).

**Theorem 3.3.1.** Consider the scalar conservation law (2.1) with deterministic flux $f(u)$ and random initial data $u_0 : \omega \rightarrow L^1(D)$ satisfying (2.2) - (2.3). Assume further that we are given a FVM such that (1.51) holds and such that Assumption 3.2.1 is satisfied; in particular, assume that the deterministic FVM scheme converges at rate $s > 0$ in $L^\infty([0,T],L^1(D))$ for every $0 < T < \infty$, as in (3.14). Then, for time $t = t^n$, the MC estimate $E_M[u_T^n(\cdot)](\omega)$ defined in (3.19) satisfies, for every $M$, the error bound

$$
\|E[u(\cdot,t^n)] - E_M[u_T^n(\cdot)](\omega)\|_{L^2(\Omega,L^1(D))} \leq C \Delta x^s \|u_0\|_{L^1(\Omega,TV(D))} + 2M^{-\frac{s}{2}} \|u_0\|_{L^2(\Omega,L^1(D))}.
$$

(3.22)

where $C = C_0 + C_t t^n > 0$ is independent of $M$ and $\Delta x$; constants $C_0, C_t$, as well as the convergence rate $s > 0$, are as in (3.13) and (3.14).

**Proof.** The proof is mostly based on [75, 76], with some minor modifications in order to achieve the $L^1(\Omega,TV(D))$-norm of the first error term instead of the $L^2(\Omega,TV(D))$.

Firstly, we estimate, for arbitrary $t = t^n > 0$,

$$
\|E[u(\cdot,t)] - E_M[u_T(\cdot,t)]\|_{L^2(\Omega,L^1(D))} \leq \|E[u(\cdot,t)] - E[u_T(\cdot,t)]\|_{L^2(\Omega,L^1(D))} + \|E[u_T(\cdot,t)] - E_M[u_T(\cdot,t)]\|_{L^2(\Omega,L^1(D))}
$$

$$
=: \varepsilon_d + \varepsilon_s.
$$

By linearity of the expectation, and since the mathematical expectation $E[\cdot]$ is deterministic and hence $L^2(\Omega,\mathbb{R})$-norm can be dropped, the first term $\varepsilon_d$, corresponding to the spatio-temporal discretization error, is bounded by triangle inequality

$$
\varepsilon_d = \|E[u(\cdot,t^n)] - u_{nT}^{\eta}\|_{L^1(D)} \leq \|u(\cdot,t^n) - u_{nT}^{\eta}\|_{L^1(\Omega,L^1(D))}.
$$

72
3.3 Convergence analysis of MC-FVM

which can be bounded by (3.13), (3.14) and finally by triangle inequality,

\[ \|u(\cdot, t^n) - u^n_{T_L}\|_{L^1(\Omega, L^1(D))} \leq \|u_0 - u^n_T\|_{L^1(D)} + C_1 \Delta x^s t^n \|u_0\|_{TV(D)} \|_{L^1(\Omega, \mathbb{R})} \]

\[ \leq C_0 \Delta x^s \|u_0\|_{TV(D)} + C_1 \Delta x^s t^n \|u_0\|_{TV(D)} \|_{L^1(\Omega, \mathbb{R})} \]

\[ \leq (C_0 + C_1 t^n) \Delta x^s \|u_0\|_{L^1(\Omega, TV(D))}. \]

The second term \( \varepsilon_s \), corresponding to the statistical sampling error, is bounded by (3.10) and then by (3.12),

\[ \varepsilon_s = \|E[u^n_T] - E_M[u^n_T](\omega)\|_{L^2(\Omega, L^1(D))} \leq M^{-\frac{1}{2}} 2 \|u^n_T\|_{L^2(\Omega, L^1(D))} \]

\[ \leq M^{-\frac{1}{2}} 2 \|u_0\|_{L^2(\Omega, L^1(D))}. \]

\[ \square \]

3.3.2 Error estimates for linear hyperbolic systems of conservation laws

We are not able to show rigorously that the MC-FVM algorithm converges for general random systems of conservation laws (1.133). In the special case of the random linear systems of conservation law (2.25), an error estimate for the convergence of \( E_M^{(k)}[U^n_T] \) to the \( \mathbb{E}(U(\cdot, t^n)^{(k)}) \) for arbitrary \( k \in \mathbb{N} \) is obtained in [105]; we review the most important results in this section.

**Theorem 3.3.2.** Consider linear system of conservation laws (2.25) and assume that the hypothesis of Theorem 2.3.7 is satisfied with \( k \geq 2 \), i.e. second moments of the random initial data \( U_0 \), source \( S \) and \( K \) exist. Assume further that we are given a FVM such that (1.51) holds and such that Assumption 3.2.2 is satisfied; in particular, assume that the deterministic FVM scheme converges at rate \( s > 0 \) in \( L^\infty([0, T], L^2(D)) \) for every \( 0 < T < \infty \), as in (3.17). Then, for time \( t = t^n \), the MC estimate \( E_M^{(k)}[U^n_T(\cdot)](\omega) \) defined in (3.19) satisfies, for every \( M \), the error bound

\[ \|\mathbb{E}[U(t^n)] - E_M^{(k)}[U^n_T](\omega)\|_{L^2(\Omega, L^2(D))} \leq \frac{C \Delta x^s}{M^{\frac{1}{2}}} \|K, U_0, S, t^n\|_{L^1(\Omega, H^4(D))} \]

\[ + M^{-\frac{1}{2}} \|K, U_0, S, t^n\|_{L^2(\Omega, L^2(D))}. \]

where \( C = C_0 + C_1 t^n > 0 \) is independent of \( M \) and \( \Delta x \); constants \( C_0, C_1 \), as well as the convergence rate \( s > 0 \), are as in (3.16) and (3.17); norms are as in (1.43) and (2.31).

**Proof.** Firstly, we bound the left hand side of (3.23) using the triangle inequality,

\[ \|\mathbb{E}[U(\cdot, t^n)] - E_M^{(k)}[U^n_T](\omega)\|_{L^2(\Omega, L^2(D))} \leq \|\mathbb{E}[U(\cdot, t^n)] - E_M^{(k)}[U^n_T]\|_{L^2(\Omega, L^2(D))} \]

\[ + \|E_M^{(k)}[U^n_T](\omega)\|_{L^2(\Omega, L^2(D))} \]

\[ =: \varepsilon_d + \varepsilon_s. \]
Monte Carlo Finite Volume method

Using linearity of the expectation, the first term \(\varepsilon_d\), corresponding to the spatio-temporal discretization error, can be bounded by triangle inequality

\[
\varepsilon_d = \|E[U(\cdot, t^n, \omega)] - U^n_T(\cdot, \omega)\|_{L^2(D)} \leq \|U(\cdot, t^n, \omega) - U^n_T(\cdot, \omega)\|_{L^1(\Omega, L^2(D))},
\]

which can be further bounded by (3.17), (3.16), \(K \geq 1\), and finally by triangle inequality

\[
\varepsilon_d \leq \|U(\cdot, t^n, \omega) - U^n_T(\cdot, \omega)\|_{L^1(\Omega, L^2(D))} \\
\leq \|U_0 - U^n_T, S - S_T, t^n\|_{L^2(D)} + C_l \Delta x^s t^n \|K, U_0, S, t^n\|_{H^1(D)} \leq \|C_0 \Delta x^s \|U_0, S, t^n\|_{H^1(D)} + C_l \Delta x^s t^n \|K, U_0, S, t^n\|_{H^1(D)} \leq (C_0 + C_l t^n) \Delta x^s K_2 \bigg(\|U_0\|_{L^1(\Omega, H^1(D))} + t^n \|S\|_{L^1(\Omega, H^1(D))}\bigg) < \infty.
\]

The second term \(\varepsilon_s\), corresponding to the statistical sampling error, is bounded by (3.10) and then by (3.15),

\[
\varepsilon_s = \|E[U^n_T] - E_M[U^n_T](\omega)\|_{L^2(\Omega, L^2(D))} \\
\leq M^{-\frac{1}{2}} \|U^n_T\|_{L^2(\Omega, L^2(D))} \\
\leq M^{-\frac{1}{2}} \|K, U_0, S, t^n\|_{L^2(\Omega, L^2(D))} \\
\leq M^{-\frac{1}{2}} K_2 \bigg(\|U_0\|_{L^2(\Omega, L^2(D))} + t^n \|S\|_{L^2(\Omega, L^2(D))}\bigg).
\]

\[\square\]

3.3.3 Number of samples

To equilibrate statistical and spatio-temporal discretization errors in (3.22) and (3.23), we require the number of Monte Carlo sample to be (asymptotically)

\[
M = O(\Delta x^{-2s}).
\]  

(3.24)

We are unable to prove a version of the error estimates (3.22) and (3.23) for general nonlinear hyperbolic systems of conservation laws. This is primarily on account of the fact that there are no mathematical convergence results for numerical schemes approximating (1.1). However, we will choose the number of MC samples by (3.24) and test numerically whether the above convergence rates and error vs. work estimates hold for particular hyperbolic nonlinear systems.
3.3.4 Error vs. work estimates

Next, we are interested in the asymptotic behavior of the errors (3.22) and (3.23) vs. the (possibly random) computational work of all FVM solves required in (3.19).

We will compute the computational work for required in (3.19) using the results obtained in subsection 1.4.11. Since fluxes \( \mathbf{F} \), coefficients \( c \) and even the initial data \( U_0 \) directly relates to the fastest wave speed \( \lambda \) via the CFL condition (1.51), \( \lambda \) can strongly depend on the particular realizations of the random input data \( I_i \), for \( i = 1, \ldots, M \). For the remaining sections, we assume that the expected maximum wave speed \( \bar{\lambda} = \mathbb{E}[\lambda(\omega)] \) in is finite,

\[
\bar{\lambda} = \mathbb{E}[\lambda(\omega)] < \infty.
\]  

(3.25)

We would also like to remark, that assumption (3.25) holds for bounded random fluxes defined in (2.11), since then

\[
\mathbb{E}[\lambda(\omega)] \leq \mathbb{E} \left[ \left\| \frac{\partial}{\partial u} f(\omega, \cdot) \right\|_{C^0(\mathbb{R}, \mathbb{R}^d)} \right] \leq \text{ess sup}_{\omega \in \Omega} \left\| f(\omega, \cdot) \right\|_{C^1(\mathbb{R}, \mathbb{R}^d)} < \infty.
\]

Under assumption (3.25), the expected computational work for the MC-FVM estimate (3.19) is finite and is given by the product of the number of samples \( M \) and the expected computational work for one sample, leading to

\[
\mathbb{E}[\text{Work}_T^M] = \mathbb{E}[M \cdot \text{Work}_T(\omega)] = \mathcal{O}(\Delta x^{-2s}) \Delta x^{-(d+1)} B \bar{\lambda} = \mathcal{O}(\Delta x^{-(d+1+2s)}).
\]

(3.26)

The resulting error vs. work complexity of the MC-FVM scheme (3.19) with \( L^2(\Omega) \)-type error bounds (3.22) - (3.23) and expected computational work (3.26) is given by

\[
\| \mathbb{E}[U(t^n)] - U_M(t^n) \|_{L^2(\Omega,E)} \lesssim \mathbb{E}[\text{Work}]^{-\frac{s}{d+1+2s}}.
\]

(3.27)

3.4 Discussion

The convergence rate \( s/(d + 1 + 2s) \) of MC-FVM scheme is considerably lower than the convergence rate \( s/(d + 1) \) of the deterministic FVM scheme, which satisfies (with \( C_{\text{Err}}, \lambda \) and \( B \) as in (1.122))

\[
\| U(\cdot, t^n) - U^n_T \|_{L^1(\mathbb{R}^d)} \leq C_{\text{Err}}(\lambda B)^{\frac{s}{d+1}} \text{Work}^{-\frac{s}{d+1}} = C_{\text{FVM}} \text{Work}^{-\frac{s}{d+1}}.
\]

(3.28)

Hence, the MC-FVM is considerably more expensive than the standard FVM for a deterministic conservation law. As an example, a first order scheme \( (s = 1/2) \) leads to a convergence rate of 1/6 for the MC-FVM as compared to a convergence rate of 1/4 for the standard FVM for a deterministic conservation law. Moreover, even for smooth
Monte Carlo Finite Volume method

solutions and high order FV schemes, i.e. as $s \to \infty$, the convergence rate in (3.27) tends very slowly towards the dominating MC convergence rate of $1/2$.

MC methods are non-intrusive and therefore are very easy to code. Existing deterministic PDE solvers are reused in MC codes. The asymptotic convergence rate $M^{-1/2}$ in (3.4) and (3.8) is non-improvable by the central limit theorem. Therefore, MC methods require a large number of “samples” (with each “sample” involving the numerical solution of (1.1) with a given draw of random input data $I(\omega)$) in order to ensure low statistical errors. This slow convergence entails high computational costs for MC type methods. In particular, quantifying uncertainty with MC methods for systems of conservation laws in several space dimensions becomes very costly.

To illustrate the discussion above, we consider a two-dimensional example of shallow water equations (1.12) with random bottom topography, modeling the propagation of a tsunami-like wave with random initial amplitude over a calm surface of an ocean. More detailed description of the problem setting will be provided in the forthcoming chapter 8. The approximated mean $\mathbb{E}[h + b]$ and the variance $\mathbb{V}[h + b]$ of the water surface $h + b$ over bottom topography $b$ obtained by the MC-FVM scheme (3.19) for this test case is provided in Figure 3.1. The mesh resolution is set to $512^2$ cells and the number of MC samples is set to $M = 8192$. The CFL number in (1.121) - (1.122) is set to 0.45 and second order accurate TeCNO scheme from subsection 1.4.8 was used. The run-time of the parallel (see chapter 5) MC-FVM simulation was approximately 7 hours and 30 minutes on 128 cores, resulting in total computational run-time of 960 CPU hours, rendering this simulation almost out of reach for serial implementations of MC-FVM.

<table>
<thead>
<tr>
<th>$M$</th>
<th>grid size</th>
<th>CFL</th>
<th>cores</th>
<th>runtime</th>
</tr>
</thead>
<tbody>
<tr>
<td>8192</td>
<td>512x512</td>
<td>0.45</td>
<td>128</td>
<td>7:27:44</td>
</tr>
</tbody>
</table>

Figure 3.1: MC-FVM simulation of shallow water equations (1.12) with random bottom topography for the propagation of a tsunami-like wave with random initial amplitude. The expectation of the water surface $h + b$ is depicted on the left, and the variance is depicted on the right. The run-time of the MC-FVM simulation was approximately 7 hours and 30 minutes on 128 cores, resulting in total computational run-time of 960 CPU hours.
3.4 Discussion

Even though the parallel implementation of MC-FVM (see chapter 5) significantly reduces the required 960 hours of computational work to only 8 hours on 128 cores (refer to Figure 3.1), the mesh resolution of $512^2$ is only for test purposes; often, much finer mesh resolutions are needed for production runs. As an example of how quickly the required computational cost grows due to low convergence rate of MC-FVM, we estimate the run-time required for MC-FVM scheme for problem as in Figure 3.1 but on slightly finer mesh resolution, i.e. $2048^2$ cells. The required number of samples $M$ according to (3.24) would then be $8192 \cdot 2^4 = 131072$. Using (3.26) with $d = 2$ and $s = 1$, we obtain that

$$E[\text{Work}^M_T] = \mathcal{O}(\Delta x^{-(d+1+2s)}) = \mathcal{O}(\Delta x^{-(2+1+2)}) = \mathcal{O}(\Delta x^{-5}),$$

hence the amount of computational work required for MC-FVM on $2048^2$ mesh would be $2^{5-2} = 1024$ times larger than the amount of computational work required for MC-FVM on $512^2$ mesh, resulting in almost 1 million CPU hours (983 040). Even on the largest CPU-based high performance computing cluster Rosa in Swiss National Supercomputing Center [126], containing 47 872 CPU cores, such simulation would take almost one day, i.e. more than 20 hours. The decrease in the mean square error in (3.27), however, would be only by a modest factor of 4, due to the low MC-FVM convergence rate of $1/5$.

We note that here we presented only a two-dimensional example. Realistic simulations of practical engineering interest for three-dimensional random Euler, MHD and acoustic wave equations are significantly more computationally intensive and hence the uncertainty quantification of such systems using plain MC-FVM is currently out of reach even on the largest high performance computing systems.
Monte Carlo Finite Volume method
4 Multi-Level Monte Carlo Finite Volume method

Given the slow convergence of MC-FVM, we propose the multi-level Monte Carlo finite volume method (MLMC-FVM). The key idea behind MLMC-FVM is to simultaneously draw MC samples on a hierarchy of nested grids [75].

4.1 Derivation of MLMC-FVM

In this section we present a scheme that allows us to achieve asymptotically almost the accuracy versus work bound (3.28) of the deterministic FVM also for the stochastic input data $I(\omega)$, rather than the single level MC-FVM error bound (3.27). The key ingredient in the Multi-level Monte Carlo Finite Volume (MLMC-FVM) scheme is simultaneous MC sampling on different levels of resolution of the FVM, with level dependent numbers $M_\ell$ of MC samples. Before the derivations, we introduce some necessary notation.

4.1.1 Hierarchical nested spatio-temporal discretization

As in section 1.4 we consider here periodic Cartesian physical domains

\[ D = I_1 \times \cdots \times I_d \subset \mathbb{R}^d, \quad I_r \subset \mathbb{R}, \quad r = 1, \ldots, d. \]

However, all results of the present work also extend to systems (1.1) in general, polyhedral domains with suitable boundary conditions.

Assume $\mathfrak{M} = \{\mathcal{T}_\ell\}_{\ell=0}^\infty$ is a family of nested triangulations of $D$,

\[ \{\mathcal{T}_\ell\}_{\ell=0}^\infty = \{\mathcal{T}_\ell^1 \times \cdots \times \mathcal{T}_\ell^d\}_{\ell=0}^\infty, \]

with the mesh widths (for simplicity of exposition, we assume that mesh widths are equal in each dimension), given by

\[ \Delta x_\ell = \Delta x(\mathcal{T}_\ell) := \frac{|I_1|}{\#\mathcal{T}_\ell^1} = \cdots = \frac{|I_d|}{\#\mathcal{T}_\ell^d} = \mathcal{O}(2^{-\ell} \Delta x_0), \quad \ell \in \mathbb{N}_0. \quad (4.1) \]
Multi-Level Monte Carlo Finite Volume method

For each discretization level $\ell \in \mathbb{N}_0$, we denote the FV approximation of $U$ by $U_{T_\ell}(\cdot, t)$ on mesh $T_\ell \in \mathfrak{M}$, with level-dependent time step $\Delta t_\ell$ given by the CFL condition (1.51),

\[ \lambda \frac{\Delta t_\ell}{\Delta x_\ell} \leq C_{CFL}, \quad \ell = 0, 1, 2, \ldots, \tag{4.2} \]

with the maximum wave speed $\lambda$ and CFL constant $C_{CFL}$ independent of level $\ell$.

### 4.1.2 Derivation of MLMC-FVM

As in plain MC-FVM, our aim is to compute, for $0 < t < \infty$, the approximate value of the expectation $E[u(\cdot, t)]$ and higher moments of the random entropy solution of the system of conservation laws (1.133) with random input data $I(\omega)$, $\omega \in \Omega$, satisfying the random input assumption 3.1.1.

As in the previous section, $E[U(\cdot, t)]$ will be estimated by replacing $U(\cdot, t)$ by a FVM approximation. By the stability assumptions 3.2.1 and (3.2.2) of the FVM scheme, for each $\mathbb{P}$-a.s. $\omega \in \Omega$, a sequence of stable FVM approximations $\{U_{T_\ell}(\cdot, t, \omega)\}_{\ell=0}^\infty$ is generated on $T_\ell$ using the time steps $\Delta t_\ell$ adapted to grid $T_\ell \in \mathfrak{M}$ as in (4.2).

Fixing the finest resolution level $L \in \mathbb{N}$, we observe that the telescoping sum holds,

\[ E[U_{T_L}] = E\left[U_{T_0} + \sum_{\ell=1}^L (U_{T_\ell} - U_{T_{\ell-1}})\right], \]

which, by the linearity of the expectation operator, can be rewritten as

\[ E[U_{T_L}] = E[U_{T_0}] + \sum_{\ell=1}^L E[U_{T_\ell} - U_{T_{\ell-1}}]. \tag{4.3} \]

Exact mathematical expectations $E[\cdot]$ for each term in the sum (4.3) are then estimated statistically by a Monte Carlo method with a level-dependent number of samples $M_\ell$, resulting in the MLMC estimator

\[ E^L[U_{T_L}] = E_{M_0}[U_{T_0}] + \sum_{\ell=1}^L E_{M_\ell}[U_{T_\ell} - U_{T_{\ell-1}}], \]

where $E_M[U_T(\cdot, t, \omega)]$ is as in (3.19), and where for each resolution level $\ell = 0, \ldots, L$, the FVM approximation $U_{T_\ell}(\cdot, t, \omega)$ is computed on mesh $T_\ell$ assuming (4.2), i.e. that the time steps $\Delta t_\ell$ are chosen subject to the CFL constraint (4.2).

Based on the derivations above, we present the Multi Level Monte Carlo Finite Volume algorithm (MLMC-FVM for short), consisting of the following four main steps:
4.1 Derivation of MLMC-FVM

1. Hierarchy of space-time discretizations: As described in subsection 4.1.1, consider nested triangulations \( \{T_\ell\}_{\ell=0}^L \) of the spatial domain \( D \) with corresponding mesh widths \( \Delta x_\ell \), satisfying

\[
\Delta x_\ell = \Delta x(T_\ell) = 2^{-\ell} \Delta x_0, \quad \ell = 0, \ldots, L,
\]

where \( \Delta x_0 \) is the mesh width for the coarsest mesh resolution \( T_0 \) and corresponds to the lowest level \( \ell = 0 \), and \( \Delta x_L \) is the mesh width for the finest mesh resolution \( T_L \) and corresponds to the finest level \( \ell = L \).

2. Sample: For each level of resolution \( \ell = 0, \ldots, L \), we draw a level-dependent number \( M_\ell \) of independent, identically distributed (i.i.d) random input samples

\[
\hat{I}_i = \{\hat{U}_{0,\ell}(\omega), \hat{c}_i(\omega), \hat{F}_i(\omega), \hat{S}_i(\omega)\}, \quad i = 1, \ldots, M_\ell,
\]

from the input random fields

\[
I(\omega) = \{U_0(\omega), c(\omega), F(\omega), S(\omega)\}.
\]

Then approximate the domain dependent random fields, such as initial data \( \hat{U}_{0,\ell} \), source \( \hat{S}_i \) and coefficients \( \hat{c}_i \) by piecewise constant functions obtained from cell averaging on mesh \( T_\ell \) of computational domain \( D \), as described in section 1.4.

Denote the resulting approximated realization of random input data by \( I_{T_\ell} \).

3. Solve: For each resolution level \( \ell = 0, \ldots, L \) and for each realization of the random input data \( I_{T_\ell} \) for \( i = 1, \ldots, M_\ell \), the resulting deterministic balance law (1.1) (for this particular realization) is solved numerically by the Finite Volume Method (1.49) with mesh width \( \Delta x_\ell \) and the corresponding time step \( \Delta t_\ell \) given by (4.2).

We denote the resulting ensemble of Finite Volume solutions by \( U_{T_\ell}^{i,n} \), \( i = 1, \ldots, M_\ell \). These constitute vectors of approximate cell averages, i.e. \( U_{T_\ell}^{i,n} = \{U_{C}^{t,n} : C \in T_\ell\} \) of the corresponding realization of the random balance law at time level \( t^n \) and at spatial resolution level \( \ell \).

4. Estimate solution statistics: The estimate of the expectation of the random solution field is given by the following estimator,

\[
E_L[U_{T_\ell}^n(t^n)] := E_{M_0}[U_{T_0}^n] + \sum_{\ell=1}^L E_{M_\ell}[U_{T_\ell}^n - U_{T_{\ell-1}}^n], \quad (4.4)
\]

with \( E_{M_\ell} \) being the MC-FVM estimator defined in (3.19) for the mesh level \( T_\ell \).

Higher moments \( M^k[U(\cdot, t)] = E[U(\cdot, t)^{(k)}] \) of order \( k \geq 2 \) (resp. the \( k \)-th order space-time correlation functions) in (1.131) of the random entropy solution \( U \) can be estimated in the same manner: based on (3.20) in MC-FVM method, the straightforward generalization along the lines of the MLMC estimate (4.4) of the
Multi-Level Monte Carlo Finite Volume method

MC-FVM estimate \([3.21]\) for \(\mathcal{M}^k[U](t)\) leads to the definition of the MLMC-FVM estimator

\[
E^{L,(k)}[U^\ell_t(\cdot)] := E_M \left[ U_{\ell_0}^t(\cdot) \right] + \sum_{\ell=1}^L E_M \left[ U_{\ell_0}^t(\cdot) - U_{\ell-1}^t(\cdot) \right]. \tag{4.5}
\]

More generally, given time instances \(t = (t_1, \ldots, t_k) \in [0,T]^k\) with time horizon \(T < \infty\), the statistical FVM estimate of \(\mathcal{M}^k[U](t)\) is given by

\[
E^{L,(k)}[U^T_{\ell_0}(\cdot)] := E_M \left[ \bigotimes_{r=1}^k U_{\ell_0}^t(\cdot, t_r) \right] + \sum_{\ell=1}^L E_M \left[ \bigotimes_{r=1}^k U_{\ell_0}^t(\cdot, t_r) - \bigotimes_{r=1}^k U_{\ell-1}^t(\cdot, t_r) \right]. \tag{4.6}
\]

**Remark 4.1.1.** In the present work, we assume (for ease of exposition only) the sequence of triangulations \(\{T_\ell\}_{\ell=0}^\infty\) to be nested. This assumption was also made in the proofs of \([75]\). We emphasize here that an inspection of the arguments in \([75]\) reveals that the nestedness assumption on the meshes is not essential for the error bounds to hold. However, in order to execute Step 4, (estimate solution statistics), in the case that the grid hierarchy is non-nested, an efficient inter-grid restriction resp. prolongation must be available. This is often the case, when multilevel discretizations have been employed in the deterministic solver which is used for the discrete solutions.

The MLMC-FVM is **non-intrusive** as any standard FVM (or DG) code can be used in step 3. Furthermore, MLMC-FVM is amenable to **efficient parallelization** as data from different grid resolutions and different samples only interacts in step 4.

### 4.2 Convergence analysis of MLMC-FVM

#### 4.2.1 Error estimates for scalar conservation laws

As for the MC-FVM, we review error estimates from \([75, 76]\) for the convergence of \(E^{L,(k)}[u^n_T]\) to \(E[u(\cdot, t^n)^{(k)}]\) for arbitrary \(k \in \mathbb{N}\) in the case of random scalar conservation laws \((2.1)\).

**Theorem 4.2.1.** Consider the scalar conservation law \((2.1)\) with deterministic flux \(f(u)\) and random initial data \(u_0 : \omega \to L^1(D)\) satisfying \((2.2) - (2.3)\). Assume further that we are given a FVM such that \((1.51)\) holds and such that Assumption 3.2.1 is satisfied; in particular, assume that the deterministic FVM scheme converges at rate \(s > 0\) in \(L^\infty([0,T], L^1(D))\) for every \(0 < T < \infty\), as in \((3.14)\). Then, for time \(t = t^n\), and
for any sequence \( \{M_\ell\}_{\ell=0}^{\infty} \) of sample sizes at mesh level \( \ell \), the MLMC-FVM estimate \( \mathbb{E}^k[u(\cdot,t)] \) in (4.4) satisfies the following error bound,

\[
\left\| \mathbb{E}[u(\cdot,t^n)] - E^k[u^n_T](\omega) \right\|_{L^2(\Omega,L^1(D))} \leq C \Delta x^s \|u_0\|_{L^1(\Omega,TV(D))} + 4C \left( \sum_{\ell=1}^{L} M_\ell^{-\frac{1}{2}} \Delta x_\ell^s \right) \|u_0\|_{L^2(\Omega,TV(D))} + 2M_0^{-\frac{1}{2}} \|u_0\|_{L^2(\Omega,L^1(D))},
\]

where \( C = C_0 + C_1 t^n > 0 \) is independent of \( \ell \), \( M_\ell \) and \( \Delta x_\ell \); constants \( C_0, C_1 \), as well as the convergence rate \( 0 < s \leq 1 \), are as in (3.13) and (3.14).

Assume further that for some \( k \in \mathbb{N} \) holds \( u_0 \in L^{2k}(\Omega,L^1(D)) \). Then the MLMC estimates \( E^{L,(k)}[u^n_T(\cdot,t)] \) as defined in (4.5) converge to the \( k \)-th moment (or spatial \( k \)-point correlation function) \( M^k[u](t) \) defined in (1.126). Moreover, the error bound holds

\[
\left\| M^k[u(\cdot,t^n)] - E^{L,(k)}[u^n_T](\omega) \right\|_{L^2(\Omega,L^1(D)^k)} \leq C \Delta x^s \|u_0\|_{L^k(\Omega,TV(D))} + 4C \left( \sum_{\ell=1}^{L} M_\ell^{-\frac{1}{2}} \Delta x_\ell^s \right) \|u_0\|_{L^{2k}(\Omega,TV(D))} + 2M_0^{-\frac{1}{2}} \|u_0\|_{L^{2k}(\Omega,L^1(D))}.
\]

**Proof.** We provide the proof for \( k = 1 \); the proof for the moments of order \( k > 1 \) is analogous, with an additional argument as in the proof of Theorem 3.1.3.

To estimate the left hand side of (4.7), we write using the triangle inequality, the linearity of the mathematical expectation \( \mathbb{E}[\cdot] \) and the definition (4.4) of the MLMC estimator, with the notation \( u^n_{T-1} \equiv 0 \),

\[
\left\| \mathbb{E}[u(\cdot,t^n)] - E^k[u^n_T](\omega) \right\|_{L^2(\Omega,L^1(D))} \\
\leq \left\| \mathbb{E}[u(\cdot,t^n)] - \mathbb{E}[u^n_T](\omega) \right\|_{L^2(\Omega,L^1(D))} + \left\| \mathbb{E}[u^n_T] - E^k[u^n_T](\omega) \right\|_{L^2(\Omega,L^1(D))} \\
= \| \mathbb{E}[u(\cdot,t^n)] - \mathbb{E}[u^n_T] \|_{L^2(\Omega,L^1(D))} \\
+ \left\| \sum_{\ell=0}^{L} \mathbb{E}[u^n_T] - u^n_{T-1} \right\|_{L^2(\Omega,L^1(D))} \\
=: \varepsilon_d + \varepsilon_s.
\]

We estimate terms \( \varepsilon_d \) and \( \varepsilon_s \) separately. By linearity of the expectation, and since the mathematical expectation \( \mathbb{E}[\cdot] \) is deterministic and hence the \( L^2(\Omega,\mathbb{R}) \)-norm can be dropped, the first term \( \varepsilon_d \), corresponding to the spatio-temporal discretization error, equals

\[
\varepsilon_d = \| \mathbb{E}[u(\cdot,t^n)] - u^n_T \|_{L^1(D)} = \| u(\cdot,t^n) - u^n_T \|_{L^1(\Omega,L^1(D))},
\]
Multi-Level Monte Carlo Finite Volume method

which can be bounded by (3.13) - (3.14),
\[ \| u(\cdot, t^n) - u^n_T \|_{L^1(\Omega, L^1(D))} \leq C \Delta x^\ell \| u_0 \|_{L^1(\Omega, TV(D))}. \]
with constant $C = C_0 + C \ell t^n$. We hence focus on the second term $\varepsilon_s$, corresponding to the statistical sampling error, which is further split and bounded using the triangle inequality,
\[
\varepsilon_s \leq \| \mathbb{E}[u^n_T] - E_{M_0}[u^n_T] \|_{L^2(\Omega, L^1(D))} + \sum_{\ell=1}^L \| \mathbb{E}[u^n_T - u^n_{T_{\ell-1}}] - E_{M_\ell}[u^n_T - u^n_{T_{\ell-1}}] \|_{L^2(\Omega, L^1(D))} =: \varepsilon_0 + \varepsilon_{1...L}.
\]
The first term $\varepsilon_0$ can be bounded directly by the MC bound (3.6) and assumption (3.12),
\[
\varepsilon_0 \leq 2M_0^{-\frac{1}{2}} \| u^n_T \|_{L^2(\Omega, L^1(D))} \leq 2M_0^{-\frac{1}{2}} \| u_0 \|_{L^2(\Omega, L^1(D))}.
\]
The second term $\varepsilon_{1...L}$ is also bounded by the MC bound (3.6),
\[
\varepsilon_{1...L} \leq \sum_{\ell=1}^L 2M_\ell^{-\frac{1}{2}} \| u^n_T - u^n_{T_{\ell-1}} \|_{L^2(\Omega, L^1(D))}.
\]
We estimate for every $\ell \geq 0$ the size of the detail $u^n_T - u^n_{T_{\ell-1}}$ with the triangle inequality
\[
\| u^n_T - u^n_{T_{\ell-1}} \|_{L^2(\Omega, L^1(D))} \leq \| u(\cdot, t^n) - u^n_T \|_{L^2(\Omega, L^1(D))} + \| u(\cdot, t^n) - u^n_{T_{\ell-1}} \|_{L^2(\Omega, L^1(D))}.
\]
Using here (3.13) and (3.14) with (4.2), we obtain for every $\ell \in \mathbb{N}$ the estimate
\[
\| u^n_T - u^n_{T_{\ell-1}} \|_{L^2(\Omega, L^1(D))} \leq 2C \Delta x^\ell \| u_0 \|_{L^2(\Omega, TV(D))}.
\]
with constant $C = C_0 + C \ell t^n$. Summing this error bound over all discretization levels $\ell = 1, ..., L$ and incorporating the bound for the error $\varepsilon_0$, the bound of the MLMC-FVM approximation error (4.7) is obtained.

We remark, that as MC-FVM is a special case of MLMC-FVM (set $L = 0$ and $T_0 = T$ in (4.4) to obtain (3.19)), the MC-FVM error bound (3.22) is also a special case of the MLMC-FVM error bound (4.7).

4.2.2 Error estimates for linear hyperbolic systems of conservation laws

Analogously as for the MC-FVM, it is not possible to show rigorously that the MLMC-FVM algorithm converges for general random hyperbolic systems of conservation laws (1.133). In the special case of the random linear hyperbolic system of conservation laws (2.25), a rigorous error estimate for the convergence of $E^L(k) [U^n_T]$ to the $E(U(\cdot, t^n)^{(k)})$ for arbitrary $k \in \mathbb{N}$ is obtained in [105]; we present the most important results in this section.

84
4.2 Convergence analysis of MLMC-FVM

**Theorem 4.2.2.** Consider linear system of conservation laws (2.25) and assume that the hypotheses (A1) - (A3) of Theorem 2.3.1 are satisfied with \( k \geq 2 \), i.e. second moments of the random initial data \( U_0 \), source \( S \) and \( K \) exist. Assume further that we are given a FVM such that (1.51) holds and such that Assumption 3.2.2 is satisfied; in particular, assume that the deterministic FVM scheme converges at rate \( s > 0 \) in \( L^\infty([0,T],L^2(D)) \) for every \( 0 < T < \infty \), as in (3.17). Then, for time \( t = t^n \), and for any sequence \( \{M_k\}_{k=0}^\infty \) of sample sizes at mesh level \( \ell \), the MLMC estimate \( E^L[U^n_{\ell^k}](\omega) \) defined in (4.4) satisfies the following error bound,

\[
\|E[U(t^n)] - E^L[U^n_{\ell^k}](\omega)\|_{L^2(\Omega;L^2(D))} \leq C \Delta x^k_\ell \|K, U_0, S, t^n\|_{L^1(\Omega;H^s(D))} \\
+ 2C \left( \sum_{\ell=1}^{k} M_{\ell}^{-\frac{1}{2}} \Delta x^k_\ell \right) \|K, U_0, S, t^n\|_{L^2(\Omega;H^s(D))} \\
+ M_0^{-\frac{1}{2}} \|K, U_0, S, t^n\|_{L^2(\Omega;L^2(D))},
\]

where \( C = C_0 + Ct^n > 0 \) is independent of \( \ell, M_0 \) and \( \Delta x^k_\ell \); constants \( C_0, C_1 \), and the convergence rate \( s > 0 \), are as in (3.16) and (3.17); norms are as in (1.43) and (2.31).

Assume further the hypotheses (A1) - (A3) of Theorem 2.3.1 are satisfied with \( k = 2k \) for some \( k \geq 2 \), i.e. the 2k-th moments of the random initial data \( U_0 \) source \( S \) and \( K \) exist. Then, the MLMC estimates \( E^{L,(k)}[U^n_{\ell^k}(\cdot,t,\omega)] \) in (3.2) converge to \( M^k[U] = E[U^{(k)}] \) as \( M \to \infty \). Furthermore,

\[
\|M^k[U(t^n)] - E^{L,(k)}[U^n_{\ell^k}](\omega)\|_{L^2(\Omega;L^2(D))} \leq C \Delta x^{2k}_\ell \|K, U_0, S, t^n\|^k_{L^2(\Omega;H^s(D))} \\
+ 2C \left( \sum_{\ell=1}^{k} M_{\ell}^{-\frac{1}{2}} \Delta x^{2k}_\ell \right) \|K, U_0, S, t^n\|_{L^{2k}(\Omega;H^s(D))} \\
+ M_0^{-\frac{1}{2}} \|K, U_0, S, t^n\|_{L^{2k}(\Omega;L^2(D))}.
\]

**Proof.** We provide a proof for \( k = 1 \); a proof for the remaining case \( k > 1 \) requires a short additional argument as in the proof of Theorem 3.1.5.

Using the triangle inequality, the left hand side of (4.9) is bounded by

\[
\|E[U(\cdot,t^n)] - E[U^n_{\ell^k}](\omega)\|_{L^2(\Omega;L^2(D))} + \|E[U^n_{\ell^k}](\omega) - E^L[U^n_{\ell^k}](\omega)\|_{L^2(\Omega;L^2(D))} = \varepsilon_d + \varepsilon_s.
\]

We estimate terms \( \varepsilon_d \) and \( \varepsilon_s \) separately. By linearity of the expectation, the first term \( \varepsilon_d \), corresponding to the spatio-temporal discretization error, equals

\[
\varepsilon_d = \|E[U(\cdot,t^n,\omega) - U^n_{\ell^k}(\cdot,\omega)]\|_{L^2(D)} = \|U(\cdot,t^n,\omega) - U^n_{\ell^k}(\cdot,\omega)]\|_{L^1(\Omega;L^2(D))},
\]

which can be bounded by (3.16) and (3.17) using \( K \geq 1 \). Using MLMC definition (4.4), the triangle inequality, linearity of the expectation, and the MC bound (3.10), the
second term \( \varepsilon_s \), corresponding to the statistical sampling error, is bounded by (with the notation \( U_{T_l} \equiv 0 \))

\[
\varepsilon_s = \left\| \sum_{\ell=0}^{L} \mathbb{E}\left[U^n_{T_\ell}(\cdot, \omega) - U^n_{T_{\ell-1}}(\cdot, \omega)\right] - E_M\left[U^n_{T_\ell}(\cdot, \omega) - U^n_{T_{\ell-1}}(\cdot, \omega)\right] \right\|_{L^2(\Omega, L^2(D))}
\]

\[
\leq \sum_{\ell=0}^{L} \left\| \mathbb{E}\left[U^n_{T_\ell}(\cdot, \omega) - U^n_{T_{\ell-1}}(\cdot, \omega)\right] - E_M\left[U^n_{T_\ell}(\cdot, \omega) - U^n_{T_{\ell-1}}(\cdot, \omega)\right] \right\|_{L^2(\Omega, L^2(D))}
\]

\[
\leq M_0^{-\frac{1}{2}} \|U^n_{T_\ell}(\cdot, \omega)\|_{L^2(\Omega, L^2(D))} + \sum_{\ell=1}^{L} M_\ell^{-\frac{1}{2}} \|U^n_{T_\ell} - U^n_{T_{\ell-1}}(\cdot, \omega)\|_{L^2(\Omega, L^2(D))}.
\]

The first term is bounded by (3.15); the detail terms \( U^n_{T_\ell} - U^n_{T_{\ell-1}} \) are bounded by

\[
\|U^n_{T_\ell} - U^n_{T_{\ell-1}}\|_{L^2(\Omega, L^2(D))} \leq \|U - U^n_{T_\ell}\|_{L^2(\Omega, L^2(D))} + \|U - U^n_{T_{\ell-1}}\|_{L^2(\Omega, L^2(D))}.
\]

Using (3.16) and (3.17), detail terms can be further bounded by (using \( K \geq 1 \))

\[
\|U^n_{T_\ell} - U^n_{T_{\ell-1}}\|_{L^2(\Omega, L^2(D))} \leq (C_0 + C_1 t^n) \Delta x^n_\ell \|K, U_0, S, t^n\|_{L^2(\Omega, \mathcal{H}^2(D))}.
\]

Summing the above bounds over all levels \( \ell > 0 \), the error bound (4.9) follows. \( \square \)

We remark, that as MC-FVM is a special case of MLMC-FVM (set \( L = 0 \) and \( T_0 = T \) in (4.4) to obtain (3.19)), the MC-FVM error bound (3.23) is also a special case of the MLMC-FVM error bound (4.9).

### 4.2.3 Number of samples for each level and error vs. work estimates

In this section we discuss several possible strategies for choosing the number of samples \( M_\ell \) for each level \( \ell = 0, \ldots, L \) in the MLMC-FVM estimators (4.7) and (4.9). Denoting

\[
E = L^1(D), \quad H = TV(D)
\]

for (4.7), and

\[
E = L^2(D), \quad H = \mathcal{H}^s(D)
\]

for (4.9), we consider a generalized form of the MLMC-FVM errors (4.7) and (4.9),

\[
\left| \mathbb{E}[U(t^n)] - E^L[U^n_{T_\ell}](\omega) \right|_{L^2(\Omega, E)} \leq C_1 \Delta x^n_\ell \|I\|_{L^1(\Omega, H)} + C_2 \left( \sum_{\ell=1}^{L} M_\ell^{-\frac{1}{2}} \Delta x^n_\ell \right) \|I\|_{L^2(\Omega, H)}
\]

\[
+ C_3 M_0^{-\frac{1}{2}} \|I\|_{L^2(\Omega, E)}.
\]

where the constants \( C_1, C_2, C_3, \) and the norms of the random input data \( I \) represent the corresponding constants and norms used in (4.7) and (4.9).
4.2 Convergence analysis of MLMC-FVM

In particular, we are interested in the choice of the sample sizes \( \{M_\ell\}_{\ell=0}^\infty \) such that, for every \( L \in \mathbb{N} \), the MLMC error (4.13) is of order \( \Delta x_L^s \), where \( s \) is the order of convergence as in (3.14) and (3.17). The principal issue in the design of MLMC-FVM is the optimal choice of \( \{M_\ell\}_{\ell=0}^\infty \) such that, for each \( L \), an error (4.13) is achieved with minimal total computational work for MLMC-FVM, which, under the assumption (3.25) of finite expected maximal wave speed \( \bar{\lambda} = \mathbb{E}[\lambda(\omega)] \), is derived from (1.122),

\[
\mathbb{E}\left[ \text{Work}^L_{T_L} \right] = \mathbb{E}\left[ \sum_{\ell=0}^L M_\ell \text{Work}^{T_\ell} (\omega) \right] = \sum_{\ell=0}^L M_\ell \mathbb{E}\left[ \text{Work}^{T_\ell} (\omega) \right]
\]

\[
= \sum_{\ell=0}^L M_\ell B \bar{\lambda} \Delta x_\ell^{-(d+1)} = B \bar{\lambda} \sum_{\ell=0}^L M_\ell \Delta x_\ell^{-(d+1)}.
\]

MLMC-FVM error bound (4.13) is the basis for choosing the numbers \( M_\ell \) of MC samples across the mesh levels.

We present two strategies: asymptotically equilibrated and asymptotically optimized number of samples. We remark that, since only error bounds (instead of strict equalities) of type (4.13) are available with possibly conservative constants \( C_1, C_2, C_3 \) (due to successive application of triangle inequalities in the proofs of Theorems 4.2.1 - 4.2.2), such “equilibration” or “optimization” of errors in (4.13) is only meaningful in the asymptotic regime, i.e., for \( \Delta x \ll 1 \).

Asymptotically equilibrated number of samples

Alternatively, the number of samples \( M_\ell \) in the bound (4.13) can be selected such that all error terms in (4.13) are asymptotically equilibrated. In particular, we require each of these terms to be of order \( O(\Delta x^s_L) \). Consequently, for the last term in (4.13) we obtain

\[
M_0 = O(\Delta x^{-2s}_L) = O(2^{2sL}),
\]

and for each term in the sum of the second error term of (4.13), we obtain

\[
M_\ell = O(\Delta x^s_L/\Delta x^s_\ell)^{-2}) = O(2^{2s(L-\ell)}), \quad \ell = 1, \ldots, L.
\]

Hence, we choose the number of samples according to

\[
M_\ell = M_L 2^{2s(L-\ell)}, \quad \ell = 0, \ldots, L,
\]

where \( M_L \) denotes the number of samples on the finest level and is treated as an additional discretization parameter.

Notice that (4.15) implies that the largest number of MC samples is required on the coarsest mesh level \( \ell = 0 \), whereas only a small fixed number of MC samples are needed on the finest discretization levels.
Multi-Level Monte Carlo Finite Volume method

Choosing sample numbers $M_{\ell}$ in (4.13) according to (4.15) leads to the following asymptotic error bound,

$$
\left\| E[U(t^n)] - E^{L}[U^n_{T_L}](\omega) \right\|_{L^2(\Omega, E)} \leq C_{Err} \Delta x_{L}^s (L + 2) \sim C_{Err} \Delta x_{L}^s \log(\Delta x_{L}^{-1}). \quad (4.16)
$$

MLMC-FVM work estimates. The total amount of expected computational work required for the complete MLMC-FVM estimator (4.4) with the number of samples according to (4.15) can be estimated using (4.14),

$$
\text{Work}_{\text{MLMC}}(\Delta x_L, L) = B \bar{\lambda} \sum_{\ell=0}^{L} M_{\ell} \Delta x_{\ell}^{-(d+1)}
$$

$$
= B \bar{\lambda} \sum_{\ell=0}^{L} M_{\ell} 2^{2s(L-\ell)} \Delta x_{\ell}^{-(d+1)}
$$

$$
= B \bar{\lambda} \sum_{\ell=0}^{L} M_{\ell} 2^{2s(L-\ell)} 2^{-(d+1)(L-\ell)} \Delta x_{L}^{-(d+1)}
$$

$$
= M_{L} B \bar{\lambda} \Delta x_{L}^{-(d+1)} 2^{\alpha L} \sum_{\ell=0}^{L} 2^{-\alpha \ell},
$$

where $\alpha := 2s - (d + 1)$. Next, we bound the sum in (4.17) using geometric series,

$$
2^{\alpha L} \sum_{\ell=0}^{L} 2^{-\alpha \ell} \leq \begin{cases} 
\frac{1}{1-2^{-\alpha}} & \alpha < 0, \\
L + 1 & \alpha = 0, \\
\frac{2^{\alpha L}}{1-2^{-\alpha}} & \alpha > 0.
\end{cases}
$$

MLMC-FVM error vs. work estimates. Using work estimates (4.17) - (4.18) to express $\Delta x_L$ in terms of computation work (which we denote by “Work”) and inserting the expression to (4.16), we obtain the asymptotic error vs. work estimate for (4.4),

$$
\left\| E[U(t^n)] - E^{L}[U^n_{T_L}](\omega) \right\|_{L^2(\Omega, E)} \leq \begin{cases} 
C_{L} \text{Work}^{-s/(d+1)} \log(\text{Work}) & s < (d + 1)/2, \\
C_{L} \text{Work}^{-1/2} \log(\text{Work})^{3/2} & s = (d + 1)/2, \\
C_{L} \text{Work}^{-1/2} \log(\text{Work}) & s > (d + 1)/2.
\end{cases} \quad (4.19)
$$

The second bound in (4.19) is obtained as follows. From (4.18) we deduce that

$$
\Delta x_{L}^{s} \log(\Delta x_{L}^{-1}) = O \left( \left( \frac{\text{Work}}{W(\text{Work})} \right)^{-1/2} \cdot \log \left( \frac{\text{Work}}{W(\text{Work})} \right) \right), \quad (4.20)
$$

where $W(\cdot)$ denotes the inverse of the function $f(W) = W \exp(W)$, also called the Lambert $W$-function [22]. For $y \gg 1$, $y \leq \log(y) \exp(\log(y))$ holds, hence also $W(y) \leq \log(y)$. Replacing $W(\cdot)$ by $\log(\cdot)$ in the first term of (4.20) and removing the $W(\cdot)$ from the logarithmic term, the bound (4.19) is obtained.
4.2 Convergence analysis of MLMC-FVM

The third bound in (4.19) is obtained (for $\alpha > 0$) by rewriting (4.17) - (4.18) as

$$\text{Work}_{\text{MLMC}}(\Delta x_L, L) = O(\Delta x_L^{-\alpha} 2^\alpha L) = O(\Delta x_L^{-\alpha} \Delta x_L^{\alpha} \Delta x_0^\alpha) = O(\Delta x_L^{-2s}),$$

and then expressing $\Delta x_L$ in terms of computational work.

The error estimates in (4.19) show that the MLMC-FVM is superior to the MC-FVM as the asymptotic computational cost for MLMC-FVM scales as $\text{Work}^{-s/(d+1)}$ (disregarding logarithmic term); compare to $\text{Work}^{-s/(d+1+2s)}$ for the MC-FVM scheme as in (3.27).

Hence, the MLMC-FVM is expected to be (asymptotically) considerably faster than the MC-FVM for the same magnitude of error.

Notice that for $s > (d+1)/2$, the asymptotic error convergence rate in (3.27) no longer increases as it is limited by the convergence rate of Monte Carlo sampling which equals 1/2. Moreover, the constant $C_L$ also increases with $s$, see (4.29). However, such high ($s > 2$) convergence rates of the deterministic solver can only be observed in very special cases, where the pathwise regularity of the stochastic solution is high (for instance, for linear hyperbolic systems with smooth initial data, sources and coefficients).

Furthermore, if $s \leq (d+1)/2$ then the error vs. work estimate (4.19) is almost (i.e. up to a logarithmic term) of the same order as the error vs. work of the deterministic finite volume scheme (3.28) which implies that the total amount of work to achieve a certain error level $\varepsilon$, for instance, in approximation of the random entropy solution’s mean field, will (asymptotically) be equal to that of approximating the entropy solution of one deterministic balance law at the same level $L$ of resolution.

**Asymptotically optimized number of samples**

Another approach is to select the number of samples $M_\ell$ in (4.13) such that the sum over all error terms in (4.13) are asymptotically optimized with respect to the required computational work [43, 89]. We note that such “optimization” is understood in terms of minimizing only the error bound in (4.13) and not the error directly; such error bound, of course, is only valid asymptotically. In comparison, the strategy behind the previous choice (4.15) was to asymptotically equilibrate each error term, while completely ignoring the fact that the amount of computational work $\text{Work}_\ell$ for each sample differs significantly among mesh levels $\ell$.

As before, we fix the number of samples $M_L \in \mathbb{N}$ on the finest mesh resolution $\ell = L$. Next, we denote the total computational work for all MC sample differences on a given level $\ell = 1, \ldots L$ by

$$\text{Work}_\ell = \text{Work}_{\ell}^{M_\ell} + \text{Work}_{\ell-1}^{M_\ell} = M_\ell w_\ell, \quad \ell = 1, \ldots L,$$

where $w_\ell$ denotes the computational work needed for one sample difference at level $\ell$,

$$w_\ell = \text{Work}_{\ell-1}^{M_\ell}, \quad \ell = 1, \ldots L.$$
Multi-Level Monte Carlo Finite Volume method

Each $L^2(\Omega, E)$-error term $\varepsilon_\ell$ in the sum of the second error term in (4.13)

$$C_2 \| I \|_{L^2(\Omega, \mathcal{H})} \sum_{\ell=1}^L \varepsilon_\ell, \quad \varepsilon_\ell = M_\ell^{-\frac{1}{2}} \Delta x_\ell^s,$$

can be expressed in terms of the computational work at the corresponding level $\ell$,

$$\varepsilon_\ell = \varepsilon(\text{Work}_\ell) = M_\ell^{-\frac{1}{2}} \Delta x_\ell^s = \left( \frac{\text{Work}_\ell}{w_\ell} \right)^{-\frac{1}{2}} \Delta x_\ell^s = 2^{\beta(L-\ell)} \Delta x_\ell^s w_\ell^\frac{3}{2} \text{Work}_\ell^{-\frac{1}{2}}, \quad (4.21)$$

where $\beta = s - (d + 1)/2$.

In order to find the asymptotically optimal number of samples $M_\ell$, we compute the marginal computational work of error for given $\text{Work}_\ell$ for each level $\ell$,

$$\frac{d\varepsilon_\ell}{d\text{Work}_\ell} = 2^{\beta(L-\ell)} \Delta x_\ell^s w_\ell^\frac{1}{2} \left( -\frac{1}{2} \text{Work}_\ell^{-\frac{3}{2}} \right). \quad (4.22)$$

Then, the optimal computational works $\text{Work}_\ell$ for each level $\ell$ are obtained by equilibrating the marginal computational work of error (4.22) on all levels $\ell = 1, \ldots, L$, i.e. by setting them equal to the marginal computational work of error at the finest level $\ell = L$ with fixed $M_L$,

$$\frac{d\varepsilon_L}{d\text{Work}_L} = \frac{d\varepsilon_\ell}{d\text{Work}_\ell},$$

leading to

$$\Delta x_L^s w_L^\frac{1}{2} \left( -\frac{1}{2} \text{Work}_L^{-\frac{3}{2}} \right) = 2^{\beta(L-\ell)} \Delta x_\ell^s w_\ell^\frac{1}{2} \left( -\frac{1}{2} \text{Work}_\ell^{-\frac{3}{2}} \right),$$

$$\text{Work}_L^{-\frac{3}{2}} = 2^{\beta(L-\ell)} \text{Work}_\ell^{-\frac{3}{2}},$$

$$\text{Work}_\ell = \text{Work}_L 2^{\frac{3}{2} \beta(L-\ell)}.$$

Using definition (4.2.3) and $\beta = s - (d + 1)/2$, the numbers of samples are given by

$$M_\ell = \frac{\text{Work}_\ell}{w_\ell} = \frac{\text{Work}_L 2^{\frac{3}{2} \beta(L-\ell)}}{w_L 2^{-(d+1)(L-\ell)}} = M_L 2^{\frac{3}{2}(s+d+1)(L-\ell)} = M_L 2^{\frac{3}{2}(s+d+1)(L-\ell)},$$

for each level $\ell = 1, \ldots, L$. Setting the number of samples $M_0$ at the coarsest level $\ell = 0$ analogously, all three error terms in the bound (4.13) are asymptotically equilibrated, leading to

$$M_\ell = \left[ M_L 2^{\frac{3}{2}(s+d+1)(L-\ell)} \right], \quad \ell = 0, \ldots, L, \quad (4.23)$$

where $[\cdot]$ denotes the rounding up to integer values for number of samples and $M_L \geq 1$ denotes the number of samples on the finest level and is treated as a parameter of the MLMC-FVM algorithm.

Notice that analogous to (4.15), (4.23) implies that the largest number of MC samples is required on the coarsest mesh level $\ell = 0$, whereas only a small fixed number of MC samples is needed on the finest discretization levels.
4.2 Convergence analysis of MLMC-FVM

**MLMC-FVM error estimates.** Choosing sample numbers \( M_\ell \) in (4.13) according to (4.23) leads to the following second error term in (4.13),

\[
\sum_{\ell=1}^{L} M^{-\frac{1}{2}} \Delta x^s_{\ell} = \sum_{\ell=1}^{L} M_L 2^{-\frac{1}{3}(s+d+1)(L-\ell)} \Delta x^s_{L} 2^{s(L-\ell)} = M_L \Delta x^s_{L} 2^{\alpha L} \sum_{\ell=1}^{L} 2^{-\alpha \ell}, \tag{4.24}
\]

where \( \alpha := \frac{1}{3}(2s - (d + 1)) \). Next, we bound the sum in (4.24) using geometric series,

\[
2^{\alpha L} \sum_{\ell=1}^{L} 2^{-\alpha \ell} \leq \begin{cases} 
\frac{1}{1-2^\alpha} & \alpha < 0, \\
L & \alpha = 0, \\
\frac{2^{\alpha L} 2^{-\alpha}}{1-2^{-\alpha}} & \alpha > 0.
\end{cases} \tag{4.25}
\]

Together with (4.23) for \( M_0 \), (4.25) leads to the following error bound for (4.13),

\[
\| E[U] - E^L[U] \|_{L^2(\Omega, E)} \leq \begin{cases} 
C_{Err} \Delta x^s_{L} & s < (d + 1)/2, \\
C_{Err} \Delta x^s_{L} \log(\Delta x^{-1}_{L}) & s = (d + 1)/2, \\
C_{Err} \Delta x^{\frac{1}{2}(s+d+1)}_{L} & s > (d + 1)/2.
\end{cases} \tag{4.26}
\]

The third bound in (4.26) is obtained (for \( \alpha > 0 \)) by rewriting (4.24) - (4.25) as

\[
M_L \Delta x^s_{L} 2^{\alpha L} \sum_{\ell=1}^{L} 2^{-\alpha \ell} \leq O(\Delta x^s_{L} 2^{\alpha L}) = O(\Delta x^s_{L} \Delta x^\alpha_{L} \Delta x^0_{L}) = O(\Delta x^{\frac{1}{2}(s+d+1)}_{L}).
\]

**MLMC-FVM work estimates.** The total amount of expected computational work required for the complete MLMC-FVM estimator (4.4) with the number of samples according to (4.23) can be estimated using (4.14),

\[
\text{Work}_{\text{MLMC}}(\Delta x_L, L) = B\lambda \sum_{\ell=0}^{L} M_\ell \Delta x^-(d+1) = B\lambda \sum_{\ell=0}^{L} M_L 2^{\frac{1}{3}(s+d+1)(L-\ell)} \Delta x^-(d+1) = B\lambda \sum_{\ell=0}^{L} M_L 2^{\frac{1}{3}(s+d+1)(L-\ell)} 2^{-(d+1)(L-\ell)} \Delta x^-(d+1) = M_L B \lambda \Delta x^-(d+1) 2^{\alpha L} \sum_{\ell=0}^{L} 2^{-\alpha \ell}, \tag{4.27}
\]

where \( \alpha := \frac{1}{3}(2s - (d + 1)) \). Finally, the sum in (4.27) is bounded by (4.18).
Multi-Level Monte Carlo Finite Volume method

MLMC-FVM error vs. work estimates. Using work estimates (4.27) and (4.18) to express $\Delta x_L$ in terms of Work and plugging the expression to (4.26), we obtain the error vs. work estimate for (4.4),

$$\|E[U(t^n)] - E^L[U^L_{t^n}]\|_{L^2(\Omega, E)} \leq \begin{cases} C_L \text{Work}^{-s/(d+1)} & s < (d + 1)/2, \\ C_L \text{Work}^{-1/2} \log \text{Work}^{1/2} & s = (d + 1)/2, \\ C_L \text{Work}^{-1/2} & s > (d + 1)/2. \end{cases}$$

where the logarithmic term in the second case is obtained as in (4.20).

The third bound in (4.28) is obtained (for $\alpha > 0$) by rewriting (4.27) and (4.18) as

$$\text{Work}_{\text{MLMC}}(\Delta x_L, L) = O(\Delta x_L^{-(d+1)/2} \alpha L)$$

$$= O(\Delta x_L^{-(d+1)} \Delta x_L^{-\alpha} \Delta x_0^\alpha) = O(\Delta x_L^{-\frac{s}{d}+(d+1)}),$$

and then expressing $\Delta x_L$ in terms of computational work.

The error estimates (4.28) show that the MLMC-FVM is superior to the MC-FVM as the asymptotic computational cost for MLMC-FVM scales as Work$^{-s/(d+1)}$; compare to Work$^{-s/(d+1)+2s}$ for the MC-FVM scheme as in (3.27). Additionally, logarithmic terms present in (4.19) are absent in (4.28). Hence, the MLMC-FVM with (4.28) is expected to be (asymptotically) slightly faster than MLMC-FVM with (4.19) and considerably faster than the MC-FVM for the same magnitude of error.

Furthermore, if $s < (d + 1)/2$ then the error vs. work estimate (4.28) is of the same order as the error vs. work of the deterministic finite volume scheme (3.28) which implies that the total amount of work to achieve a certain error level $\varepsilon$, for instance, in approximation of the random entropy solution’s mean field, will (asymptotically) be equal to that of approximating the entropy solution of one deterministic balance law at the same level $L$ of resolution.

Remark on constants

The constant $C_L$ in (4.19) and (4.28) is

$$C_L = C_{\text{Err}}(B\lambda)^{s/(d+1)} \begin{cases} (1 - 2^\alpha)^{-s/(d+1)} & s < (d + 1)/2, \\ 1 & s = (d + 1)/2, \\ (1 - 2^{-\alpha})^{-s/(d+1)} & s > (d + 1)/2. \end{cases}$$

For usual convergence rates, i.e. $s \geq 1/2$, $2s \in \mathbb{N}$, the constant in (4.29) is bounded by

$$C_L \leq (2M_L)^{s/(d+1)} \cdot C_{\text{Err}}(B\lambda)^{s/(d+1)}.$$
Based on numerical experiments in subsection 7.1.4 ahead, constants $C_{\text{Err}}$ in (3.28) and (4.16) or (4.26) can be considered equal. Hence, the constant $C_L$ can be bounded in terms of the constant $C_{\text{FVM}}$ as follows,

$$C_L \leq (2M_L)^{s/(d+1)} \cdot C_{\text{FVM}} \leq \sqrt{2M_L} \cdot C_{\text{FVM}}, \quad \text{for } s \leq (d + 1)/2. \quad (4.31)$$

Hence, in addition to optimal asymptotic convergence rates of the MLMC-FVM solver, bounds (4.31) provide estimates on how much further work is required for the stochastic version of the simulation compared to its deterministic version. The constants $C_L$ and $C_{\text{FVM}}$ in asymptotic bounds are of comparable magnitude, i.e. the stochastic simulation is at most $\sqrt{2M_L}$ times less accurate (disregarding logarithmic terms for (4.19)) compared to its deterministic version, where the free parameter $M_L$ is usually chosen to be small, i.e. $O(1) - O(10)$.

In the following, we continue with the discussion on the appropriate number of samples $M_L$ on the finest mesh resolution.

Choosing number of samples on the finest mesh level

In order to obtain an appropriate number of samples $M_L$ on the finest mesh resolution, we further equilibrate (exactly, not only asymptotically) the first error term in (4.13) and the first term (involving $M_L$) in the sum of the second error term, the following must hold:

$$C_1 \Delta x_L^s \|I\|_{L^1(\Omega, E)} = C_2 M_L^{-1/2} \Delta x_L^s \|I\|_{L^2(\Omega, H)},$$

which, when solved for $M_L$, yields

$$M_L = \left( \frac{C_2}{C_1} \right)^{2} \left( \frac{\|I\|_{L^2(\Omega, H)}}{\|I\|_{L^1(\Omega, E)}} \right)^{2}, \quad (4.32)$$

where spaces $E$ and $H$ are as defined in (4.11) and (4.12). For random input fields $I(\omega)$ with relatively small variance compared to the mean, the $L^1(\Omega, \cdot)$ and $L^2(\Omega, \cdot)$ norms of $I(\omega)$ can be considered equal.

Hence, using the values of the constants $C_1$ and $C_2$ as in (4.7), for scalar conservation laws with MLMC-FVM error estimates (4.7) we obtain

$$M_L \approx \left( \frac{4C}{C_1} \right)^{2} = 16, \quad (4.33)$$

and, using the values of the constants $C_1$ and $C_2$ as in (4.9), for linear hyperbolic systems of conservation laws with MLMC-FVM error estimates (4.9) we obtain

$$M_L \approx \left( \frac{2C}{C_2} \right)^{2} = 4. \quad (4.34)$$
Multi-Level Monte Carlo Finite Volume method

We note, however, that factors 4 and 2 in (4.7) and (4.9) appear due to successive application of triangle inequality in the proofs of theorems 4.2.1 and 4.2.2, and hence could in general be rather conservative bounds, meaning that the number of samples $M_L$ on the finest mesh level $L$ as derived in (4.33) and (4.34) could as well be smaller. Analogously, the number of samples $M_L$ on the finest mesh level $L$ in estimates (4.33) and (4.34) could be also larger, if $\|I\|_{L^2(\Omega,H)} \gg \|I\|_{L^1(\Omega,H)}$.

We also note that the results in (4.33) - (4.34) are in agreement with the findings in [75], where optimal values of $M_L$ for several selected scalar conservation laws were estimated using extensive numerical experiments.

Choosing number of samples on the coarsest mesh level

Analogously to the number of samples $M_L$ on the finest mesh level $\ell = L$, the number of samples $M_0$ on the coarsest mesh level $\ell = 0$ can also be derived exactly (not only asymptotically) by equilibrating the last error term in (4.13) and the last term (involving $M_1$) in the sum of the second error term, i.e. the following must hold:

$$C_3 M_0^{-\frac{1}{2}} \|I\|_{L^2(\Omega,E)} = C_2 M_1^{-\frac{1}{2}} \Delta x^s \|I\|_{L^2(\Omega,H)},$$

which, when solved for $M_0$ using the definition of $M_1$ as in (4.15) or (4.23), yields

$$M_0 = M_L 2^{L-1} \Delta x^{-2s} \left( C_3 \right)^2 \left( \frac{\|I\|_{L^2(\Omega,E)}}{\|I\|_{L^2(\Omega,H)}} \right)^2,$$

(4.35)

where spaces $E$ and $H$ are as defined in (4.11) and (4.12). However, contrary to (4.32) for choosing $M_L$, good a-priori estimates for the right hand side of (4.35) (in order to obtain $M_0$) are not available in general and highly depend on particular problem and coarsest mesh $T_0$. We note that since samples on the coarsest mesh resolution are very cheap to compute, the total computational work spent on all samples of mesh level $T_0$ is minuscule, and hence the number of samples on the coarsest mesh level $M_0$ could be safely overestimated by a very large margin. In contrary, the estimation of the number of samples on the finest mesh resolution $M_L$ influences the amount of total computation work significantly. Alternatively, a-posteriori estimates for $M_0$ could be used by estimating (during the MLMC-FVM simulation) by

$$M_0 \approx M_1 \frac{\hat{\sigma}_0^2}{\hat{\sigma}_1^2},$$

(4.36)

where $\hat{\sigma}^2$ are empirical estimates of level difference variances $\sigma^2$,

$$\hat{\sigma}_0 \approx \sigma_0 = \|U^n_{T_0} - \mathbb{E}[U^n_{T_0}]\|_{L^2(\Omega,E)},$$

(4.37)

and, for $\ell > 1$,

$$\hat{\sigma}_\ell \approx \sigma_\ell = \|(U^n_{T_\ell} - U^n_{T_{\ell-1}}) - \mathbb{E}[U^n_{T_\ell} - U^n_{T_{\ell-1}}]\|_{L^2(\Omega,E)}.$$
4.2 Convergence analysis of MLMC-FVM

We refer to [82, 89] for detailed description of empirical variance estimators \( \hat{\sigma}_\ell \) and available “on-line” algorithms to compute \( \hat{\sigma}_0 \) and \( \hat{\sigma}_1 \) at runtime.

The empirical variance estimators \( \hat{\sigma}^2 \) in (4.37) - (4.38), could additionally be used to (empirically) determine the appropriate coarsest mesh resolution \( T_{\ell_0} \), i.e. to find a mesh resolution \( T_{\ell_0} \) such that the variances of level differences start decaying as the level \( \ell \) increases [89],

\[
\ell_0 = \min \{ \ell \geq 0 : \hat{\sigma}_{\ell+1} \sqrt{\text{Work}_{\ell+1}} \ll \hat{\sigma}_\ell \sqrt{\text{Work}_\ell} \}. \tag{4.39}
\]

Empirically estimated number of samples and a posteriori error estimators

In the case when no convergence results as stated in Assumptions 3.2.1 and 3.2.2 are available, one could completely rely on the empirical estimators (4.37) - (4.38). In this case, we use the bias-variance decomposition of the squared MLMC-FVM error

\[
\| E[U] - E^L[U^n_{T_L}] \|_{L^2(\Omega, E)}^2 = \| E[U - U^n_{T_L}] \|_E^2 + \| E[U^n_{T_L}] - E^L[U^n_{T_L}] \|_{L^2(\Omega, E)}^2,
\]

corresponding to the deterministic and statistical errors, where spaces \( E \) and \( H \) are as defined in (4.11) and (4.12). Statistical error can be further decomposed (for details refer to [82]) in terms of level variances \( \sigma_\ell \) as in (4.37) - (4.38),

\[
\varepsilon^2_s = \left\| E[U^n_{T_L}] - E^L[U^n_{T_L}](\omega) \right\|_{L^2(\Omega, E)}^2 = \sum_{\ell=0}^{L} \sigma^2_{\ell} \frac{M_{\ell}}{M_{\ell}}, \tag{4.40}
\]

Minimizing computational work (4.14) of MLMC-FVM under the fixed desired tolerance \( \varepsilon^2_s \), optimal numbers of samples are given (asymptotically) by [82]

\[
M_{\ell} = \frac{1}{\varepsilon^2} \sqrt{\frac{\sigma^2_{\ell}}{\text{Work}_{T_\ell}}} \sum_{k=0}^{L} \sqrt{\sigma^2_{k} \text{Work}_{T_k}} = \mathcal{O} \left( \sqrt{\frac{\sigma^2_{\ell}}{\text{Work}_{T_\ell}}} \right). \tag{4.41}
\]

Alternatively, since the number of levels is small (\( \mathcal{O}(10) \)), the discrete optimization problem of finding optimal \( M_{\ell} \) to minimize the total sampling error (4.40) w.r.t. total work (4.14) can be solved by brute-force algorithms; for further details, we refer to [89]. We also remark, that the discrete optimum, additionally to number of samples \( M_{\ell} \) for each \( \ell \geq 0 \), also provided the optimal required finest mesh discretization level \( L \in \mathbb{N}_0 \).

Note that additional a posteriori error estimators are still required to estimate the remaining term corresponding to the mean discretization error [81],

\[
\varepsilon^2_d = \left\| E[U(t^n) - U^n_{T_L}] \right\|_E^2.
\]

However, reliable and efficient a posteriori error estimators with reliability and efficiency constants close to 1 are rarely available for general nonlinear hyperbolic systems (1.1). For the details of the procedure for estimating \( M_{\ell} \) (4.41) by computing empirical variance estimators (4.37) - (4.38) within the MLMC-FVM simulation from a set of “start-up” samples and then recursively updating \( M_{\ell} \), refer to [81, 82, 89].
4.2.4 Sparse tensor MC-FVM and MLMC-FVM

The error vs. expected computational work estimates \( (3.27) \) and \( (4.19)/(4.28) \) for MC-FVM and MLMC-FVM, respectively, also extend to MC-FVM and MLMC-FVM approximation of variance \( \mathbb{V}[U(t^n)] \) as defined in \( (1.127) \). However, for general \( k \)-th point correlation functions \( k \)-th moments \( \mathcal{M}^k[U] \) as in \( (3.20) - (3.21) \), the asymptotic computational complexity of the assembly of high dimensional tensors \( (U_{T}^{i,n})^{(k)} = U_{T}^{i,n} \otimes \cdots \otimes U_{T}^{i,n} \) is of order \( O(\Delta x^{-kd}) \), which, for \( k > 1 \), is affected by the curse of \( k \)-dimensionality and is significantly more expensive compared to the computational complexity \( O(\Delta x^{-(d+1)}) \) of the FVM solution for \( U_{T}^{i,n} \) on a mesh \( \mathcal{T} \), given by \( (1.122) \).

In order to avoid such curse of dimensionality for assembly of \( (U_{T}^{i,n})^{(k)} \), sparse tensor approximations of full tensors \( (U_{T}^{i,n})^{(k)} \) in \( (3.20) - (3.21) \) were introduced for MC and MLMC methods in \( [79, 75, 9] \). The key idea for sparse tensor approximation is to expand \( U_{T}^{i,n} \) in Haar wavelet basis (see the forthcoming subsection 8.1.1),

\[
U_{T}^{i,n} = \sum_{\ell=0}^{L} W_{T_{\ell}}, \quad W_{T_{\ell}} = \bar{P}_{\ell}(U_{T}^{i,n}) - P_{T_{\ell-1}}(U_{T}^{i,n}),
\]

with \( \bar{T}_{L} = \mathcal{T} \) and \( \bar{P}_{\ell}(U_{T}^{i,n}) \) being cell-average projections of \( U_{T}^{i,n} \) onto nested hierarchical meshes \( \bar{T}_{\ell} \), for \( 0 \leq \ell \leq L \). Then, the sparse tensor approximation \( \bar{P}_{\ell}^{(k)}(U_{T}^{i,n})^{(k)} \) of \( k \)-th order tensor \( (U_{T}^{i,n})^{(k)} \) is given using the \textit{sparse tensor projection}, defined by

\[
\bar{P}_{\ell}^{(k)} = \sum_{|\bar{\ell}| \leq L, j=1}^{k} \bigotimes (P_{\ell_{j}} - P_{\ell_{j-1}}), \quad \bar{\ell} = (\ell_{1}, \ldots, \ell_{k}), \quad |\bar{\ell}|_{1} = |ar{\ell}|_{1} + \cdots + |ar{\ell}|_{k}.
\]

The sparse tensor MC-FVM is obtained by replacing \( (U_{T}^{i,n})^{(k)} \) in \( (3.20) \) with its sparse projection \( \bar{P}_{\ell}^{(k)}(U_{T}^{i,n})^{(k)} \),

\[
\tilde{E}_{M}^{(k)}[U_{T}^{n}] = \frac{1}{M} \sum_{i=1}^{M} \bar{P}_{\ell}^{(k)}(U_{T}^{i,n})^{(k)}.
\]

(4.42)

As a consequence, the sparse tensor MLMC-FVM is given by replacing \( (U_{T_{\ell}}^{n})^{(k)} \) in \( (4.5) \) with its sparse projection \( \bar{P}_{\ell}^{(k)}(U_{T_{\ell}}^{n})^{(k)} \),

\[
\tilde{E}_{M_{\ell}}^{L,(k)}[U_{T_{L}}^{n}] = E_{M_{0}}[\bar{P}_{\ell_{0}}^{(k)}(U_{T_{0}}^{n})^{(k)}] + \sum_{\ell=1}^{L} E_{M_{\ell}} \left[ \bar{P}_{\ell}^{(k)}(U_{T_{\ell}}^{n})^{(k)} - \bar{P}_{\ell-1}^{(k)}(U_{T_{\ell-1}}^{n})^{(k)} \right],
\]

or, equivalently, by replacing full tensor MC-FVM estimators \( E_{M_{\ell}}[(U_{T_{\ell}}^{n})^{(k)}] \) in \( (4.5) \) with sparse tensor MC-FVM estimators \( \tilde{E}_{M_{\ell}}^{L,(k)}[U_{T_{L}}^{n}] \) from \( (4.42) \),

\[
\tilde{E}_{M_{\ell}}^{L,(k)}[U_{T_{L}}^{n}] = \tilde{E}_{M_{0}}^{L,(k)}[U_{T_{0}}^{n}] + \sum_{\ell=1}^{L} \tilde{E}_{M_{\ell}}^{L,(k)} \left[ U_{T_{\ell}}^{n} - U_{T_{\ell-1}}^{n} \right].
\]

(4.43)
4.2 Convergence analysis of MLMC-FVM

The resulting computational complexity for the assembly of $\hat{P}^{(k)}_T(U_{i,n}^{(k)})$ is asymptotically lower than complexity $O(\Delta x^{-(d+1)})$ of a full FVM solve for $U_{i,n}^{(k)}$, and is of order [75, 79]

$$O(\Delta x^{-d}|\log(\Delta x^{-1})|^{k-1}) \ll O(\Delta x^{-(d+1)}), \quad (4.44)$$

which is significantly lower than $O(\Delta x^{-kd})$ for the full tensor $(U_{i,n}^{(k)})$. It is also shown in [75, 79], that the following approximation property holds for $1 \leq p < \infty$,

$$\|U - \hat{P}^{(k)}_T(U)\|_{L^p(\mathbb{R}^{kd})} \lesssim \Delta x^k |\log \Delta x|^{k-1}\|U\|_{W^{s,p}(\mathbb{R}^d)}^{(k)}. \quad (4.45)$$

By (4.44) and (4.45), the convergence rates from the error vs. work estimates (3.27) and (4.19)/(4.28) are not reduced; hence analogous results hold for sparse tensor estimators (4.42) and (4.43), respectively. Exact error bounds and details on the role of the additional logarithmic factor from (4.45) can be found in [75, 79], and numerical results for sparse tensor MLMC-FVM for one-dimensional Euler equations can be found in [79].
Multi-Level Monte Carlo Finite Volume method
5 Parallel implementation of MLCM-FVM

Our parallel implementation of MLMC-FVM scheme is called ALSVID-UQ [2] and was originally derived for the purpose of MLMC-FVM implementation from the FVM solver for MHD equations called ALSVID [1], which was designed by researchers at CMA, University of Oslo and SAM, ETH Zurich.

In this section we describe all implementation aspects, arise in the four main stages of the MLMC-FVM solver, including the choice of the FVM solver, parallelization of FVM computations using domain decomposition methods, generation of pseudo random number for statistical realizations of random input data, stable and efficient statistical estimators and the parallelization of the MLMC-FVM using novel static and adaptive load balancing techniques.

5.1 Hierarchy of nested meshes

We will solve systems of balance laws (1.1) in one, two and three space dimensions. In the numerical results which are reported in following chapters, we will choose Cartesian meshes for simplicity. It is relatively straightforward to choose any hierarchy of nested grids consisting of either triangles/tetrahedra or quad/hexahedral volume in either one, two or three space dimensions.

5.1.1 FVM solver

The FVM solver in the current version of ALSVID-UQ, capable of approximation of the systems of conservation laws (1.1) in one, two and three space dimensions, is based on the following ingredients:

1. **Approximate Riemann solver**: The numerical fluxes in the finite volume scheme (1.49) used in ALSVID-UQ are based on approximate Riemann solvers of the HLL type, described sections 1.4.2 - 1.4.3. In particular, for the Euler equations (1.8), the code uses the HLL three wave solver proposed by Toro et. al [109] and described in 1.4.3. This approximate Riemann solver resolves contact discontinuities and has the same resolution as the standard Roe solver but computationally is less expensive and is known to preserve positive densities and pressures in the absence of round-off errors, see [41]. For the MHD equations (1.10), ALSVID-UQ uses the three and five wave HLL type solvers designed in [41]. These solvers aim
Parallel implementation of MLCM-FVM

at approximating the modified Godunov-Powell form of the MHD equations, see \cite{92} \cite{111}. For shallow water equations (1.12), ALSVID-UQ uses the well-balanced entropy stable numerical fluxes described in subsection 1.4.8.

2. **Divergence constraint.** Design of robust numerical schemes for MHD is quite challenging as the equations (1.10) involve the divergence constraint. This constraint needs to be handled in a suitable manner in order to avoid spurious oscillations, see \cite{111}. The divergence constraint is handled in ALSVID by adding the Godunov-Powell source term to the MHD equations. This source term is proportional to the divergence and allows divergence errors to be swept out of the domain. Numerical stability can only be ensured by a careful upwinding of the source term, see \cite{41}.

3. **Non-oscillatory reconstructions.** ALSVID employs a variety of piecewise polynomial non-oscillatory reconstruction procedures for attaining high order of spatial accuracy. In particular, second order ENO and WENO procedures are employed, see subsection 1.4.6 here and also section 2 of \cite{41}. However, these procedures need to be modified in order to preserve positivity of the density and pressure. Such modifications are termed ENOF and WENOF and are described in section 2 of \cite{41}. For shallow water equations (1.12) and well-balance entropy stable numerical fluxes, the reconstruction is performed in terms of scaled entropy (instead of conserved) variables, as described in \cite{39}.

4. **Time stepping.** High-order accurate time stepping procedures of the SSP Runge-Kutta \cite{47} are employed in ALSVID-UQ, as described in subsection 1.4.7. ALSVID-UQ uses a modular structure in C++ with a Python front end for pre- and post-processing. One and two dimensional visualizations are performed with Matplotlib and three dimensional data sets are visualized using MayaVi2 and parallel visualization tools such as Visit and Paraview. Extensive testing of ALSVID has been performed and reported in \cite{41}.

5.1.2 **Robust pseudo random number generators and seeding**

In this step, for each level $\ell = 0, \ldots, L$, we have to draw $M_\ell$ i.i.d. samples for the random input field $I(\omega)$ corresponding to the underlying probability distribution. Standard random number generators (RNG) can be readily used to draw such samples. For the serial implementation, any reasonable RNG works well in practice.

Random number generation becomes a very sensitive part of Monte Carlo type algorithms on massively parallel architectures. Inconsistent seeding and insufficient period length of the RNG might cause correlations in presumably i.i.d. draws which might potentially lead to biased solutions, see \cite{77}.

For all numerical experiments reported in the following chapters, the WELL-series of pseudo random number generators from \cite{65} \cite{64} were used. These generators have been
5.1 Hierarchy of nested meshes

designed with particular attention towards large periods and good equidistribution. In particular, we found WELL512a to have a sufficiently large period $2^{512} - 1$ and to be reasonably efficient (33 CPU sec for $10^9$ draws). We emphasize that there are plenty of alternatives to WELL512a with even longer periods (which, however, use more memory than WELL512a). To name a few: WELL1024a with period $2^{1024} - 1$, takes 34 sec and WELLRNG44497 with period $2^{44497} - 1$ which takes 41 sec to generate $10^9$ draws; yet another alternative could be 'Marsenne Twister' [73] - it has the same period of $2^{19937} - 1$ as its counterpart WELLRNG19937. We would also like to mention the SPRNG [72] pseudo random number generator, which is specifically targeted to parallel applications and supports independents streams (in addition to seeds) of random numbers. Within our numerical experiments, we did not observe any qualitative difference between the described RNGs, and hence we have not conducted any further quantitative tests.

To deal with the seeding issues in a parallel algorithm, when multiple independent cores are computing partial amount of (different) samples, the required combination of the sample level $\ell$, the run number $k \in \{1, \ldots, K\}$ corresponding to different random solutions needed for error estimates (6.1), and, also possibly the sample index $i \in \{1, \ldots, M_\ell\}$, is injectively (i.e. one-to-one) mapped to some integer seed.

The bijective combination of sample level $\ell$, run number $k$ and sample index $i$ is achieved by the recursive use of the Szudzik’s pairing function [102], given by

$$pair(a, b) = \begin{cases} a^2 + a + b & \text{if } a \geq b, \\ a + b^2 & \text{if } a < b. \end{cases} \quad (5.1)$$

In order improve the quality of the RNG seeding, the obtained seeds could be further mapped to the corresponding elements in a hard-coded array of prime numbers.

An even more efficient methodology, optimizing pseudo random number generation for domain-dependent random fields (such as coefficients and source terms) in presence of domain decomposition parallelization (introduced in the following section 5.2), is given by additionally pairing the sub-domain index, taking values in $\{1, \ldots, D_m\}$.

We note that pseudo random numbers, obtained by additionally pairing the sub-domain index, depend on the parallelization configuration presented in section 5.2 in particular, if the number of sub-domains $D_m$ is different, then different sets of random numbers will be generated for same values of sample level $\ell$, sample index $i$ and run number $k$. Such property is absolutely harmless for production runs, however, might cause significant difficulties in parallel debugging. We point out, however, that the reproducibility of the results is ensured, provided that the parallel configuration remains unchanged.

For pseudo RNG, which support the independent streams (additionally to seeds, such as SPRNG [72]), the pairing of samples level $\ell$ and samples index $i$ specifies the stream, whereas the pairing of run number $k$ and sub-domain index specifies the seed.
Parallel implementation of MLCM-FVM

5.1.3 Stable and efficient statistical estimators

For both MC-FVM and MLMC-FVM algorithms we need to combine individual realizations to compute ensemble averages. It is straightforward to compute the sample mean for the MC-FVM and the estimator (4.4) for MLMC-FVM. A straightforward algorithm to compute an unbiased estimate of the variance for scalar component $u$ of the vector $U = U(x, t)$ with fixed $x, t$ is the following statistical estimator,

$$\text{Var}[u] := \mathbb{E}[u^2] - \mathbb{E}[u]^2 \approx \text{Var}_M[u] := \frac{1}{M-1} \sum_{i=1}^{M} (u^i)^2 - \frac{1}{M(M-1)} \left( \sum_{i=1}^{M} u^i \right)^2$$

(5.2)

where $u^i$ are MC-FVM samples. This way, it suffices to loop over all samples only once; unfortunately, both quantities are almost equal in the regions of vanishing variance which leads to subtractive cancellation and loss of accuracy in floating point arithmetic. In [116], the authors propose an alternative stable “on-line” variance computation algorithm.

Set $\bar{u}^0 = 0$ and $\Phi^0 = 0$; then proceed iteratively,

$$\bar{u}^i = \sum_{j=1}^{i} u^j / i,$$

$$\Phi^i = \sum_{j=1}^{i} (u^j - \bar{u}^i)^2 = \Phi^{i-1} + (u^i - \bar{u}^i)(u^i - \bar{u}^{i-1}),$$

until $i = M$. The unbiased mean and variance estimates are then given by

$$E_M[u] = \bar{u}^M, \quad \text{Var}_M[u] = \Phi^M / (M - 1).$$

(5.3)

Although identical in exact arithmetic, the above algorithm can deal with small cancellation errors.

For parallel simulations, where several cores compute partial amount of the total number of samples $M$, the resulting mean and variance estimates need to be combined in the end of the simulation. Assume we have 2 cores $A$ and $B$ each computing $M_A$ and $M_B$ ($M = M_A + M_B$) number of samples respectively. Then keeping the notation and definitions as in the previous section, the unbiased estimate for mean and variance can be obtained (see [14]) by

$$E_M[u] = \frac{M_A E_{M_A}[u] + M_B E_{M_B}[u]}{M},$$

$$\delta = E_{M_B}[u] - E_{M_A}[u],$$

$$\Phi^M = \Phi^{M_A} + \Phi^{M_B} + \delta^2 \cdot \frac{M_A \cdot M_B}{M},$$

$$\text{Var}_M[u] = \frac{\Phi^M}{M - 1}.$$ 

(5.4)

This algorithm [5.4] is then recursively extended to arbitrary finite number of cores by combining any two of them until only one is left.
5.2 Parallelization with MPI using intra-communicators

We combine a robust pseudo random number generator in step 2, the FVM solver of ASLVID-UQ in step 3 and the above estimators in step 4 to obtain an efficient MLMC-FVM. The resulting code is also written in C++ with Python front and back ends. Note that (for MHD equations) the deterministic code ALSVID is reused without any alterations in the solve step as MC-FVM and MLMC-FVM are non-intrusive. i.e. any standard FVM solver can be used in step 3. Furthermore, MLMC-FVM is amenable to efficient parallelization as most of data interacts only in step 4.

For the sake of consistency, we briefly recapitulate terms frequently used in the following sections,

- **core**: an independent unit running the program (process) in parallel with other such units. We assume one MPI process per every core.
- **node**: multiple cores sharing the same physical memory.
- **sample**: initial data computed for some particular random draw of $\omega \in \Omega$.

In what follows, we assume a homogenous computing environment meaning that all cores are assumed to have identical CPUs and RAM per node, and equal bandwidth and latency to all other cores. For a heterogeneous computing environment, such as heterogeneous computing clusters with accelerators, we refer to a recently developed load balancing technique in [52].

Next, we assume to have a “pool” $\mathcal{G}$ of cores (processing units), consisting of groups $\mathcal{G}_m$ (of arbitrary size) of cores indexed by “multi level” $m = L, L - 1, \ldots, m_0 \geq 0$, which are themselves divided into equal groups $\mathcal{G}_s^m$ of cores indexed by “sampler” $s = 1, \ldots, P_m$. The number of cores in a given sampler $\mathcal{G}_s^m$ is independent on $s$ and denoted by $D_m$. An example of such pool with

$$L = 5, \quad m_0 = 1, \quad \{P_m\} = \{1, 1, 2, 4, 4\}, \quad \{D_m\} = \{1, 1, 1, 1, 2\},$$

is depicted in Figure 5.1. Such configuration of the pool leads to three levels of parallelization: across mesh resolution levels, across MC samples and inside the deterministic solver using domain decomposition (see example in Figure 5.1). We assume, that any of the MC samples from any mesh level $\ell$ can be efficiently computed on any sampler $\mathcal{G}_s^m$ in the pool, in serial or by using domain decomposition if $D_m > 1$. By efficient computation we assume strong scaling of the domain decomposition method.

The total number of cores in the pool is then given by

$$\# \mathcal{G} = \sum_{m=m_0}^{L} \# \mathcal{G}_m = \sum_{m=m_0}^{L} \sum_{s=1}^{P_m} \# \mathcal{G}_s^m = \sum_{m=m_0}^{L} P_m \times D_m = \sum_{m=m_0}^{L} C_m, \quad (5.5)$$

where, for each “multi level” $m = L, L - 1, \ldots, m_0$. 

Parallel implementation of MLCM-FVM

Figure 5.1: An example pool $\mathcal{G}$ of cores (processing units) with 5 “multi-levels” $m$, i.e. $5 = L \geq m \geq m_0 = 1$, each “multi-level” $m$ consisting of $\{P_m\} = \{1,1,2,4,4\}$ samplers denoted by $\mathcal{G}^m$, and each sampler $\mathcal{G}^m$ consisting of $\{D_m\} = \{1,1,1,1,2\}$ cores. The total number of cores in the pool is then given by (5.5), i.e. $\#\mathcal{G} = 16$.

- $D_\ell$ denotes the number of computational sub-domains in domain decomposition,
- $P_\ell$ denotes the number of “samplers”, i.e. groups of cores, where every such group computes some portion of required Monte Carlo samples at a given mesh level $\ell$,
- $C_m = P_m \times D_m$ denotes the total number of cores per “multi level” $m$.

Domain decomposition is used only in the few levels with the finest mesh resolution. On these levels, the number of MC samples is small. However, in most cases these levels require most of the computational effort.

The implementation of the described parallelization paradigm for ALSVID-UQ is using Message Passing Interface (MPI) standard and its particular implementation OpenMPI. Refer to [123] and [124] for detailed descriptions of MPI and OpenMPI, respectively.

In what follows, we assume that each MPI process is running on its own core. Simulation is divided into 3 main phases - initialization, simulation and data collection; key concepts for the implementation of such parallelization for each phase are described below.

**Phase 1 - Initialization:**

- **MPI groups and communicators.** By default, message passing in MPI is done via the main communicator MPI_COMM_WORLD which connects all processes. Each process has a prescribed unique non-negative integer called rank. The process with the rank 0 is called root. Apart from MPI_COMM_WORLD, we use MPI_Comm_split() to create sub-groups and corresponding intra-communicators, this way empowering message passing within particular sub-groups of processes. Such communicators ease the use of collective reduction operations within some particular sub-group of processes. We implemented 3 types of communicators (see Figure 5.2):
5.2 Parallelization with MPI using intra-communicators

1. **Domain** communicators connect processes within each sampler $G_m^s$: these $D_m = \#G_m^s$ processes are used for computational domain decomposition.

2. **Sampler** communicators connect processes that work on the MC samples at the *same* mesh level and the *same* computational sub-domain, i.e. processes with the same sub-domain index within samplers $\{G_m^1, \ldots, G_m^P\}$.

3. **Level** communicators connect the processes (across all multi levels $L, \ldots, m_0$, one process per level) that are roots of sampler communicators, i.e. cores from groups $\{G_L^1, \ldots, G_L^{m_0}\}$; in total there are $D_L$ level communicators, i.e. one for each computational sub-domain on the multi level $L$.

Analogously to `MPI_COMM_WORLD`, every process has a unique rank in each of the communicators 1-3; processes with rank 0 in domain communicators are called *domain roots*, in sampler communicators - *sampler roots*, and in level communicators - *level roots*. `MPI_COMM_WORLD` is used only in `MPI_Init()`, `MPI_Finalize()` and `MPI_Wtime()`. Figure 5.2 depicts all non-trivial communicators and roots for the example setup as in Figure 5.1.

![Communicators/roots:](Figure 5.2: Structure and root processes of the communicators for setup as in Figure 5.1)

“Domain communicators” connect processes within each sampler $G_m^s$ (for domain decomposition), “sampler communicators” connect processes that work on the *same* computational sub-domain and at the *same* mesh level (for parallelization of MC estimates), and “level communicators” connect roots of sampler communicators (for parallelization across all “multi-levels”).

- **Seeding RNG.** To deal with the seeding issues in a parallel algorithm, we *injectively* map the required combination (obtained using the Szudzik’s pairing function \((5.1)\) of the *mesh level* $\ell$ and the sample index $i \in \{1, \ldots, M_\ell\}$ to some corresponding element in the hard-coded array of prime numbers (henceforth, the array of
Parallel implementation of MLCM-FVM

seeds). Then each core generates random vectors with the required number of real numbers for MC samples of random input data \( \{ I^i_{T_\ell} \} \).

Phase 2 - Simulation:

- **FVM solves.** We assume, that all required sample differences in (4.4)
  \[
  \{ U^{n+1,i}_{T_\ell} - U^{n,i}_{T_\ell-1} : \ell = 0, \ldots, L, \ i = 1, \ldots, M_\ell \}
  \]
  are distributed across (assigned to) samplers
  \[\{ G^s_m : m = L, \ldots, m_0, \ s = 1, \ldots, P_m \}\].

On each sampler \( G^s_m \), parallel FVM method (using \( D_m \) cores) is used to solve for all sample pairs difference \( U^{n+1,i}_{T_\ell} - U^{n,i}_{T_\ell-1} \) that are assigned to \( G^s_m \). The obtained FVM approximations of sample differences are combined to partial statistical estimates.

- **Inter-domain communication.** During simulation phase, cell values near interfaces of adjacent computational sub-domains are exchanged asynchronously with MPI_Isend() and MPI_Irecv().

Phase 3 - Data collection and output:

- **MC estimator.** For each level, statistical estimates are collectively reduced with MPI_Reduce() into sampler roots using sampler communicators; then MC estimators (3.2) for mean and variance (see subsection 5.1.3) are obtained.

- **MLMC estimator.** MC estimators from different levels are finally combined via level communicators to level roots to obtain the MLMC estimator (4.4). For parallelization configurations with a very large number of cores in the domain decomposition, the level communicators are not formed in order to avoid too many communicators per core. In such cases, direct point-to-point communication in MPI_COMM_WORLD is used, and in this case it does not introduce any overhead, since no collective communications are present in this step.

- **Parallel data output.** Each process that is both sampler root and level root writes out the final result. Hence, the number of parallel output files is equal to \( D_L \), i.e. the number of computational sub-domains on the finest mesh level.

It is worth noting, that the FVM solver in the current version of ALSVID-UQ has several parallelization-related improvements and extensions over ALSVID, including

- Additional supported systems of conservation laws: Euler, shallow water, acoustic wave, Buckley-Leverett, Burgers’ and linear advection equations were added.

- Well-balanced entropy stable FVM solvers described in subsection 1.4.8 for shallow water equations were implemented.
5.3 Static load balancing

- Communication hiding: numerical fluxes are computed in the internal cells, during the time when non-internal boundary cells wait for the values of the ghost cells from the neighboring computational sub-domain.

- Fully asynchronous DDM communication: non-blocking message passing (sending and receiving) is used for optimal communication among all computational sub-domains.

- Custom MPI data types for transfer of ghost cell values: enables multi-dimensional MPI communication without additional memory buffers and copying of data.

The key issue in the parallel implementation of the solve steps is to distribute computational work evenly among the cores. In what follows, we assume a homogeneous computing environment meaning that all cores are assumed to have identical CPUs and RAM per node, and equal bandwidth and latency to all other cores. Next, we describe two novel load balancing strategies.

5.3 Static load balancing

For the static version of our load balancing algorithm, we enforce a natural condition, that all sample differences of mesh level \( \ell = L, \ldots, 0 \) should be distributed among the samplers \( \{ G^\ell_m, \ldots, G^P_m \} \) of the same multi level, i.e. to samplers from multi level

\[
m = \begin{cases} 
\ell & \text{if } \ell \geq m_0, \\
 m_0 & \text{if } \ell < m_0.
\end{cases}
\]  

(5.6)

Once the distribution of samples to samplers \( G^\ell_m \) is fixed, we need to compute the required number of samplers on each multi level \( m = L, \ldots, m_0 \), denoted by \( P_m \), and the lowest multi level \( m_0 \geq 0 \), such that the total runtime of all samplers is equilibrated.

To this end, for the finest level \( \ell = L \) we fix the number of cores \( C_L \),

\[
 C_L = D_L \times P_L,
\]  

(5.7)

the number of computational sub-domains \( D_L \) and the number of samplers \( P_L \) can be chosen freely, as long as \( C_L = D_L \times P_L \) and \( P_L \leq M_L \). Furthermore, we assume, that for multi mesh level \( m = L, \ldots, m_0 \), all \( M_m \) samples can be distributed evenly among all \( P_m \) samplers, i.e. \( M_m \) is evenly divisible by \( P_m \). In general, this is not a very restrictive assumption.

The total computational work for \( E_{M_{t\ell}}[U^T_{\ell\ell} - U^T_{\ell\ell-1}] \) is then given by (3.26),

\[
\text{Work}_{\ell} = \text{Work}^{M_{t\ell}}_{\ell\ell} + \text{Work}^{M_{t\ell}}_{\ell\ell-1}.
\]  

(5.8)

where we assume that the maximal wave speed \( \lambda \) is deterministic, and hence the work estimate in (3.26) is also deterministic (i.e. expectation \( E[\cdot] \) is not needed).
Parallel implementation of MLCM-FVM

The ratio of computational work for MC estimators \([3.19]\) for levels \(\ell \in \{L - 1, \ldots, 1\}\) can be obtained recursively by inserting \([4.15]\) into \emph{a-priori} work estimates \([3.26]\),

\[
\frac{\text{Work}_\ell}{\text{Work}_{\ell-1}} = \frac{M_L 2^{2(L-\ell)s} K \left( \Delta x_\ell^{-(d+1)} + \Delta x_{\ell-1}^{-(d+1)} \right)}{M_L 2^{2(L-(\ell-1))s} K \left( \Delta x_{\ell-1}^{-(d+1)} + \Delta x_{\ell-2}^{-(d+1)} \right)}
\]

\[= \frac{2^{-2\ell s}}{2^{-2(\ell+1)s}} \frac{\Delta x_\ell^{-(d+1)} \left( 1 + 2^{-(d+1)} \right)}{\Delta x_{\ell-1}^{-(d+1)} \left( 2^{-(d+1)} + 2^{-2(d+1)} \right)} = 2^{d+1-2s}. \tag{5.9}\]

For level \(\ell = 0\), the term \(U^n_{0,T-1}\) in \(E_{M0}[U^n_{0,T} - U^n_{T-1}]\) is known (\(\equiv 0\)), hence \([5.9]\) provides a lower bound rather than an equality, i.e. \(\text{Work}_0 \leq \text{Work}_1/(2^{d+1-2s})\).

Consequently, given the number of cores \(C_L\) on the multi level \(m = L\), the positive integer parameters \(D_L\) and \(P_L \leq M_L\) recursively determine the number of cores needed to equilibrate runtime on each multi level \(m < L\),

\[C_m = \left\lceil \frac{C_{m+1}}{2^{d+1-2s}} \right\rceil, \quad \forall m < L. \tag{5.10}\]

Notice that the denominator \(2^{d+1-2s}\) in \([5.10]\) is a positive integer (a power of 2) provided \(2s \in \mathbb{N}\) and \(s \leq (d + 1)/2\). However, in case \(s < (d + 1)/2\), we have

\[2^{d+1-2s} \geq 2, \tag{5.11}\]

which (when \(L\) is large) leads to inefficient load distribution for multi levels \(m \leq m_0\), where each successive level needs less than one core,

\[m_0 := \min\{0 \leq \ell \leq L : C_{\ell+1} < 2^{d+1-2s}\}, \quad \text{i.e. } C_{m_0} \leq 1/2. \tag{5.12}\]

We investigate the amount of total computational work \((\text{Work}_{\{0,\ldots,m_0\}})\) that is assigned to such “inefficient” multi levels \(m \in \{0, \ldots, m_0\}\):

\[
\text{Work}_{\{0,\ldots,m_0\}} := \sum_{\ell=0}^{m_0} \text{Work}_\ell \tag{5.9} \leq \sum_{\ell=0}^{m_0} \frac{\text{Work}_{m_0}}{2^{(d+1-2s)(m_0-\ell)}} \leq \sum_{\ell=-\infty}^{m_0} \frac{\text{Work}_{m_0}}{2^{(d+1-2s)(m_0-\ell)}} \tag{5.13}
\]

\[= \frac{\text{Work}_{m_0}}{1 - (2^{d+1-2s})^{-1}} \leq \frac{\text{Work}_{m_0}}{1 - \frac{1}{2}} = 2 \cdot \text{Work}_{m_0}.\]

For the sake of simplicity, assume that \(P_L\) and \(D_L\) are nonnegative integer powers of 2. Under this assumption, definition \([5.12]\) of \(m_0\) together with recurrence relation \([5.10]\) without rounding up \(\lceil \cdot \rceil\) implies that \(C_{m_0} \leq 1/2\). Hence, total work estimate \([5.13]\) for all mesh levels \(\ell \in \{0, \ldots, m_0\}\) translates into an estimate for sufficient number of cores, which, instead of \(m_0 + 1\), i.e. one core per multi level \(m \in \{0, \ldots, m_0\}\), turns out to be only 1:

\[\text{Work}_{\{0,\ldots,m_0\}} \leq 2 \cdot \text{Work}_{m_0} \quad \implies \quad C_{\{0,\ldots,m_0\}} \leq 2 \cdot C_{m_0} \leq 2 \cdot \frac{1}{2} = 1. \tag{5.14}\]
5.4 Strong and weak scaling

The implementation of (5.14) (i.e. multiple mesh levels assigned to multi level $m_0$ consisting of only 1 core) is essential to obtain efficient and highly scalable parallelization of MLMC-FVM when $s < \frac{d+1}{2}$.

The example of static load distribution for MLMC-FVM with parameters

\[ L = 5, \quad M_L = 4, \quad d = 1, \quad s = \frac{1}{2}, \quad D_L = 2, \quad P_L = 4, \]

is given in Figure 5.3.

![Static load distribution](image)

Figure 5.3: Static load distribution (5.6) for $L = 5, M_L = 4, d = 1, s = \frac{1}{2}, D_L = 2, P_L = 4$, i.e. for samples $M_5 = 4, M_4 = 8, M_3 = 16, M_2 = 32, M_1 = 64, M_0 = 128$. The static load distribution is obtained by distributing samples according to (5.6), and by computing number samplers $\{P_m\} = \{1, 1, 2, 4, 4\}$ and the number of cores $\{D_m\} = \{1, 1, 1, 1, 2\}$ in each sampler $P_m$ according to (5.10), with $m_0 = 1$ resulting from (5.14). The total number of cores in the pool is then given by (5.5), i.e. $\#G = 16$.

5.4 Strong and weak scaling

The static load balancing algorithm was tested on a series of benchmarks for hyperbolic solvers; refer to [77] for Euler equations of gas dynamics, [77] for Magnetohydrodynamics (MHD) equations of plasma physics, and [78] for shallow water equations with uncertain bottom topography. The runtime of all aforementioned simulations was measured by the wall clock time, accessible by `MPI_Wtime()` routine [123]. Parallel efficiency is defined as a fraction of simulation time (which excludes time spent for MPI communications and idling) over wall clock time,

\[
\text{efficiency} := 1 - \frac{\text{total clock time of all MPI routines}}{(\#\text{cores}) \times \text{wall clock time}}.
\]  

In Figure 5.4 we verify strong scaling (fixed discretization and sampling parameters while increasing #cores) and in Figure 5.5 we verify weak scaling (problem size is equivalent...
Parallel implementation of MLCM-FVM

to #cores) of our implementation. In both cases, we maintained scaling up to almost 4000 cores at high efficiency.

Simulations were executed on Cray XE6 (see [126]) with 1496 AMD Interlagos 2 x 16-core 64-bit CPUs (2.1 GHz), 32 GB DDR3 memory per node, 10.4 GB/s Gemini 3D torus interconnect with a theoretical peak performance of 402 TFlops.

We believe that our parallelization algorithm will scale linearly for a much larger number of cores.

Labels “MLMC” and “MLMC2” in Figures 5.4 - 5.5 indicate that first and second order accurate FVM solvers were used, i.e. \( s = \frac{1}{2} \) and \( s = 1 \) in (4.15), respectively.

Figure 5.4: Strong scaling up to 4000 cores. The inferior scalability of the domain decomposition method (DDM) due to additional networking between sub-domain boundaries has no significant influence on the overall scaling.

Figure 5.5: Weak scaling up to 4000 cores. Analogously as in Figure 5.4, the inferior scalability of DDM has no significant influence on the overall scaling.
5.5 Limits of static load balancing

Even though static load balancing described in section 5.3 performs very well, it also has several disadvantages.

The first disadvantage is that a particular structure for numbers of samples $M_\ell$ on each mesh level $\ell$ is assumed, given by (4.15). This, together with sharp deterministic work estimates (1.122), is the key assumption for the successful strong and weak scaling of static load balancing strategy. For a different choice of sample numbers $M_\ell$ rather than the one given by (4.15), for instance, the more efficient (by a logarithmic term in (4.28) vs. (4.19)) choice given by (4.23), then the number of cores $C_m$ computed in (5.10) would not be integer-valued anymore, for each multi level $m$, not only for $m \leq m_0$. Rounding the estimated number of cores $C_m$ to nearest integer values for implementation could result in severe load imbalance.

The second disadvantage is that despite the estimates (3.26) of average work, the efficiency of the balancing is expected to drop significantly for a single run of the MLMC-FVM due to the non-uniform $\lambda^*_\ell$, which can have a significantly large statistical spread; for instance, the histograms for the fastest wave speed distribution on each mesh resolution level in the case of log-normally distributed material coefficient $c$ in the acoustic wave equation (1.4) are provided in Figure 5.6 (which is presented here for illustration only; more detailed numerical results will be provided in the subsection 10.4.1 ahead). Notice that even for a relatively small (compared to mean $E[c] \approx 1.1$) variance parameter $\sigma^2 = 0.2$ in (10.24) of the coefficient $c$, the deviation from the mean $E[\lambda]$ in this case is as large as 50% throughout all discretization levels, resulting in a spread of the maximal wave speed $\lambda(\omega)$ approximately from 1 to 3.5.

The effect of large statistical spread in maximal wave speed $\lambda(\omega)$ can be clearly seen in Figure 5.7 where the scaling analysis of static load balancing in MLMC-FVM for the wave equation (1.5) with material coefficient given by (10.6) is performed. As expected, the algorithm scales linearly with the number of cores, but the efficiency is low.

To improve the efficiency of load balancing, sample-dependent computational work estimates (1.122) and arbitrarily structured number of samples $M_\ell$ need to be taken into account. The former objective, however, cannot be done statically at the time of compilation, since Work$\gamma(\omega)$ depends on the particular realization. To solve both problems of the static load balancing at once, in the next section we introduce an adaptive load balancing [103], where samples are distributed during run-time, after computing $\lambda(\omega)$ for each required realization, but before actually starting the FVM time stepping of any sample (hence, it is not a dynamic balancing).
5.6 Adaptive load balancing

In the following, we introduce the adaptive load balancing, prove its empirical approximation properties and comment on implementation issues. As in section 5.3, we assume a homogeneous computing environment meaning that all cores are assumed to have identical CPUs and RAM per node, and equal bandwidth and latency to all other cores.

5.6.1 Theory

Define Load\(_i^\ell\) to be the normalized (constants are neglected in (1.122)) required computation time for the \(i\)-th difference of samples between mesh levels \(\ell\) and \(\ell - 1\),

\[
\text{Load}^i_\ell = \lambda_i^\ell \left( \Delta x_\ell^{-(d+1)} + \Delta x_{\ell-1}^{-(d+1)} \right), \quad \ell = 0, \ldots, L, \quad i = 1, \ldots, M_\ell, \tag{5.16}
\]

where all \(\lambda_i^\ell\) are computed in parallel on all \(P_L\) samplers \(G^1_L, \ldots, G^{P_L}_L\), each consisting of \(D_L\) cores, and then broadcast to every core. Computations of \(\lambda_i^\ell\) do not need time stepping and hence are cheaper by a factor \(O(T/\Delta t_\ell) = O(\Delta x_\ell^{-1})\) compared to the full FVM; the required global communication is also small, i.e. of order \(O(2^{2Ls}|G|)\).

The goal of load balancing is to distribute all sample differences indexed by \((\ell, i)\) for \(\ell = 0, \ldots, L\) and \(i = 1, \ldots, M_\ell\) with required computational time \(\text{Load}^i_\ell\) to samplers \(G^*_m\). The quality of empirical load balancing strategies is described by the so-called
Figure 5.7: Inefficient strong scaling of \textit{static} load balancing in case of \textit{random} maximum wave speeds $\lambda(\omega)$ with large relative variance $\frac{\text{Var}[\lambda]}{\text{Exp}[\lambda]}$ resulting from the \textit{random} material coefficients (10.6), introduced in the forthcoming chapter 10.

The greedy algorithm for \textit{identical} samplers has been analyzed in [51] and was proven to be a $4/3$-approximation, i.e. the maximum run-time among all workers is at most $4/3$ times larger than the \textit{optimal} makespan. For not ordered loads, the greedy algorithm is only a $2$-approximation [51]. Here we present a generalization of the greedy algorithm for samplers with \textit{heterogeneous} speeds of execution. The main idea of the algorithm is the \textit{recursive} assignment of the \textit{largest} available load $\text{Load}^i_\ell$ for some level $\ell = 0, \ldots, L$ and sample index $i = 1, \ldots, M_\ell$ to the sampler $G^s_m$ on some multi level $m = L, \ldots, m_0$, such that the total run-time $\mathcal{R}_m$ of the sampler $G^s_m$, including $\text{Load}^i_\ell$ is minimized. The pseudo code of the \textit{adaptive} load balancing is provided as Algorithm 1, where the notation $(\ell, i, \text{Load}^i_\ell) \in G^s_m$ means that $i$-th difference of samples between mesh resolution levels $\ell$ and $\ell - 1$ is assigned to be computed on sampler $G^s_m$ and is expected to have run-time equal to $\text{Load}^i_\ell/D_m$. Additionally, the total run-time of all loads assigned to sampler $G^s_m$ is given by

$$\mathcal{R}(G^s_m) = \frac{1}{D_m} \sum_{(\ell, i, \text{Load}^i_\ell) \in G^s_m} \text{Load}^i_\ell.$$ 

Note that if samplers have \textit{identical} speeds of execution, i.e. $D_m$ are all equal, then the above algorithm [1] reduces to the standard greedy algorithm.
Parallel implementation of MLCM-FVM

Algorithm 1 Greedy load balancing (with non-identical speeds of execution)

\[ \mathcal{L} = \{(\ell, i, \text{Load}_i^\ell) : \ell = 0, \ldots, L, i = 1, \ldots, M_\ell \} \]

while \( \mathcal{L} \neq \emptyset \) do

\[ (\ell, i, \text{Load}_i^\ell) = \max_{(\ell, i, \text{Load}_i^\ell)} \mathcal{L} \]

\[ G^*_m = \arg \min_{G^*_m \in G} \left( R(G^*_m) + \frac{\text{Load}_i^\ell}{D_m} \right) \]

\[ G^*_m = G^*_m \cup (\ell, i, \text{Load}_i^\ell) \]

\[ \mathcal{L} = \mathcal{L} \setminus (\ell, i, \text{Load}_i^\ell) \]

end while

For not ordered loads (replace \( \max \mathcal{L} \) by \( \text{any load from } \mathcal{L} \)), algorithm 1 is only a \( (1 + D_{\max}/D_{\min}) \)-approximation (analogous proof as in [51]). Hence, if samplers \( G^*_m \) have very heterogeneous speeds of execution \( 1/D_m \), algorithm 1 may provide a much longer makespan, compared to the optimal. However, if we assume that loads are ordered and are as heterogeneous as samplers,

\[ \frac{\text{Load}_{\max}}{\text{Load}_{\min}} := \max_{\ell, i} \frac{\text{Load}_i^\ell}{\min_{\ell, i} \text{Load}_i^\ell} \geq \frac{D_{\max}}{D_{\min}} \frac{D_{m}}{\min_m D_{m}}, \tag{5.17} \]

then algorithm 1 is a 2-approximation (see Definition 5.6.1). We present this result as a theorem.

Theorem 5.6.2. Assume that (5.17) holds and that the last load of the bottle-neck sampler is bounded by \( (D_{\min}/D_{\max}) \cdot \text{Load}_{\max} \). Then algorithm 1 is a 2-approximation.

Proof. Let \( R(G^*) \) be the run-time of the bottle-neck sampler \( G^* \) and \( \text{Load}^* \) be the last sample assigned to \( G^* \). Then, according to the distribution procedure,

\[ R(G^*) \leq R(G^*_m) + \frac{\text{Load}^*}{D_m}, \quad \forall m = m_0, \ldots, L, \quad s = 1, \ldots, P_m. \]

Summing the above inequality over all samplers \( G^*_m \), we obtain the bound

\[ R(G^*) \leq \frac{1}{\# \{G^*_m \}} \sum_{m,s} \frac{\text{Load}^*}{D_m} \leq \frac{1}{\# \{G^*_m \}} \sum_{m,s} R(G^*_m) \leq R^o, \]

where \( R^o \) is the optimal timespan, which is certainly not smaller than the average of all runtimes \( R(G^*_m) \). Next, we use (5.17) and \( \text{Load}^* \leq \text{Load}_{\max} D_{\min}/D_{\max} \),

\[ \frac{1}{\# \{G^*_m \}} \sum_{m,s} \frac{\text{Load}^*}{D_m} \leq \frac{1}{\# \{G^*_m \}} \sum_{m,s} \frac{\text{Load}_{\max}}{D_{\max}} \frac{D_{\min}}{D_{\min}} \leq \frac{\text{Load}_{\max}}{D_{\max}} \leq R^o. \]

Combining both bounds, the desired inequality \( R(G^*) \leq 2R^o \) is obtained. \( \Box \)

In case of MLMC-FVM, the assumption (5.17) is often satisfied, since loads \( \text{Load}_i^\ell \) scale asymptotically as \( \text{Work}_{T_\ell} = \mathcal{O}(2^{(d+1)\ell}) \) due to (1.122) and (4.1), and the speeds of execution \( D_m \) using domain decomposition scale only as \( \# T_m \), i.e. \( D_m = \mathcal{O}(2^{d_m}) \).
5.6.2 Implementation remarks

Once the loads have been distributed to samplers \( G_m \), the parallel execution of FVM solves and the final assembly of the MLMC-FVM estimator remained analogous as in [104], i.e. Message Passing Interface (MPI) was chosen, making heavy use of the appropriate local MPI intra-communicators [123]. The new part for the adaptive balancing is the parallel computation (and broadcast) of the maximum wave speeds \( \lambda_i^\ell \), which is problem-specific. For the wave equation (1.5), \( \lambda_i^\ell \) were computed by computing random coefficients \( c_{i,\ell} \) and then using (10.1). Note that in the computation of loads (as well as the samples themselves), the parallelization configuration \((D_m, P_m)\) might need to be adapted to the required mesh level \( \ell \): due to memory limitations, samples on fine meshes use larger \( D_{m,\ell} \) and fewer samplers \( P_{m,\ell} \), and due to inefficiency of DDM, samples on coarse meshes use smaller \( D_{m,\ell} \) and more samplers \( P_{m,\ell} \), keeping \( P_{m,\ell} \cdot D_{m,\ell} = P_m \cdot D_m \).

5.7 Strong and weak scaling

To ensure a fair comparison, the adaptive load balancing algorithm was tested on the same problem as the static load balancing, see (1.5) in section 5.5.

In Figure 5.8 we verify strong scaling of our implementation. We observed scaling to be maintained for up to almost 40 000 cores at high efficiency. Simulations were executed on the same Cray XE6 (see [126]) architecture as in section 5.5. We believe that our parallelization algorithm will scale linearly for a much larger number of cores if the problem size is increased. The computational complexity of the adaptive balancing (computation of loads and distribution of samples) is \( \mathcal{O}(2^{L_{\text{max}}(2s,d)} + |G|2^{2L_s}) \), where both terms are dominated by full simulation.

Figure 5.8: Strong scaling of adaptive load balancing up to 40 000 cores. The efficiency is nearly optimal and is much better if compared to the static load balancing in Figure 5.7
Parallel implementation of MLCM-FVM

For non-linear fluxes and random initial condition, knowledge of the input data (at \( t = 0 \)) might not be sufficient to accurately estimate \( \lambda(\omega) \). In such cases the performance of the adaptive load balancing might be sub-optimal; dynamic load balancing could be used, possibly introducing a large overhead [25].

5.8 Fault tolerant implementation of ALSVID-UQ

In large scale simulations on emerging massively parallel computing platforms, processor failures at run-time are inevitable and occur with increasing frequency as the number of processors increases [13]. Conventional checkpoint/restart mechanisms will no longer work properly, which motivated authors in [87, 88] to propose a fault tolerant multi-level Monte Carlo (FT-MLMC) method, that does not rely on checkpoint/restart or on re-computation: all samples unaffected by failures are used in the computation of the final result, whereas all remaining samples affected by failures are either completely ignored or the unaffected parts in the domain decomposition parallelization of such sample are also incorporated to the final result [90].

The main theoretical result in [87] states, that under sufficient conditions on non-recoverable loss of a random fraction of MLMC samples, in particular, assuming that sufficiently many samples survive on average, asymptotic error vs. work bounds analogous to fault-free MLMC bounds (4.19)/(4.28) are also available for FT-MLMC estimates.

In the subsequent work [88], a fault tolerant version of ALSVID-UQ is implemented, based on the User Level Failure Mitigation [12], a fault tolerant extension of MPI [123]. The theoretical performance of FT-MLMC outlined in [87] is verified for the FT-MLMC-FVM simulations of the two-dimensional stochastic Euler equations of gas dynamics, analogous to simulations considered in subsection 7.1.6. Additionally, range of mean time between failures (MTBF) was determined, where fault tolerance is needed and where FT-MLMC-FVM performs well.


6 Setup for a numerical study of MLMC-FVM

In the remaining chapters, we will test the MLMC-FVM algorithm presented in chapter 4 and demonstrate its robustness, efficiency compared to MC-FVM algorithm presented in chapter 3. We run numerical tests for numerous random conservation laws (1.133), which are grouped into five different problem classes by the origin of uncertainty:

1. **Random initial data.** Euler (1.8) and MHD (1.10) equations will be considered for random initial data with up to 11 sources of randomness in one, two and three space dimensions. An empirical convergence analysis of the numerical error will be presented for the MHD problem of Orszag-Tang vortex. The verification of the derived constants in the asymptotic error bounds (4.30) will be presented for the Sod shock tube of one-dimensional Euler equations.

2. **Random source.** One- and two-dimensional shallow water equations (1.12) will be considered, where randomness enters as a random source from the uncertain bottom topography measurements, resulting in hundreds of independent sources of randomness. An empirical convergence analysis of the numerical error will be presented together with the performance analysis of novel hierarchical representation of bottom topography.

3. **Random fluxes in scalar nonlinear conservation laws.** Burgers’ (1.2) and Buckley-Leverett (1.3) equations will be considered for random, uniformly perturbed flux functions $f$ in (2.1). An empirical convergence analysis of the numerical error will be presented for both cases.

4. **Random coefficients in linear hyperbolic systems of conservation laws.** Acoustic wave equation (1.4) will be considered in one, two and three spatial dimensions, where random linear fluxes $F_r$ are due to random log-normally distributed highly heterogeneous layered material coefficient $c(x, \omega)$. An empirical convergence analysis of the numerical error will be presented for one- and two-dimensional experiments.

5. **Random fluxes in nonlinear hyperbolic systems conservation laws.** One- and two-dimensional Euler equations (1.8) will be considered with uncertain constant $\gamma$ of specifying the relation of pressure and energy in the equation of state (1.9). An empirical convergence analysis of the numerical error will be presented for both cases.
Setup for a numerical study of MLMC-FVM

We remark, that the class distribution is not strict; in particular, random initial data is also present in problem classes 2 and 4.

6.1 Notation, terminology and simulation parameters

Recalling that the discretization of the random conservation law involves discretizing in space-time with a standard Finite Volume Method and the discretizing the probability space with a statistical sampling method, we tabulate various combinations of methods that are to be tested:

- **MC**: Monte Carlo with 1st order \((s = \frac{1}{2})\) FVM \(M = \mathcal{O}(\Delta x^{-1})\),
- **MC2**: Monte Carlo with 2nd order \((s = 1)\) FVM \(M = \mathcal{O}(\Delta x^{-2})\),
- **MLMC**: multi-level MC with 1st order \((s = \frac{1}{2})\) FVM \(M_L = M_1 2^{(L-\ell)}\),
- **MLMC2**: multi-level MC with 2nd order \((s = 1)\) FVM \(M_L = M_1 4^{(L-\ell)}\).

Furthermore, we need the following parameters, which will be specified for every simulation in the form of a table below the corresponding figure:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(L)</td>
<td>number of hierarchical mesh levels</td>
</tr>
<tr>
<td>(M_L)</td>
<td>number of samples at the finest mesh level</td>
</tr>
<tr>
<td>grid size</td>
<td>number of cells in X, Y and Z directions</td>
</tr>
<tr>
<td>CFL</td>
<td>CFL number based on the fastest wave</td>
</tr>
<tr>
<td>cores</td>
<td>total number of cores used in the simulation</td>
</tr>
<tr>
<td>runtime</td>
<td>serial runs: clock-time; parallel runs: wall-time; hrs:min:sec</td>
</tr>
<tr>
<td>efficiency</td>
<td>MPI efficiency, as defined in (5.15)</td>
</tr>
</tbody>
</table>

6.2 Root mean square error estimation in MLMC-FVM

For most of the numerical experiments, an empirical convergence analysis of the numerical error and estimates of the empirical order of convergence (EOC) will be presented. We note, that such empirical error convergence analysis is meaningful not only in the case of scalar and linear hyperbolic systems of conservation laws (where global well-posedness and FVM convergence results are available, see chapter 2) but also in the case of nonlinear hyperbolic systems of conservation laws (where no results of global well-posedness or FVM convergence are available in general), since the empirically estimated FVM convergence rate \(s\) is essential for calibrating the required number of samples \(M_L\) in the MLMC-FVM simulations, as described in subsection 4.2.3. In the following, we specify the error estimator in more detail. In the MC-FVM and MLMC-FVM approximations (3.19) and (4.4) of the expectation \(\mathbb{E}[U]\), the estimators \(E_M[U^n]\) and \(E^L[U^n_{M_L}]\)
are constructed using the random draws of the solution $U$, and hence the estimators themselves are random fields, even though they are approximating deterministic statistical moments. For this reason, as already highlighted in the proofs of Theorems 3.1.4, 3.3.1, 3.3.2, 4.2.1 and 4.2.2, the discretization errors in (3.2), (3.22), (3.23), (4.7) and (4.9) are random quantities as well. For our computational error convergence analysis we therefore compute a statistical estimator by averaging estimated discretization errors from several independent runs. The aforementioned errors of MC-FVM and MLMC-FVM estimators will be computed by approximating the corresponding $L^2(\Omega, E)$ norms with MC quadrature.

Let $E_{\text{ref}}[U(x, t)]$ denote the reference solution, i.e. the exact solution $E[U(x, t)]$ or a very accurate approximation of it, and

$$E_k[U^n_T(x)], \quad k = 1, \ldots, K,$$

be a sequence of independent approximations of $E[U(x, t^n)]$ obtained by running MC-FVM or MLMC-FVM solver $K$ times, corresponding to $K$ realizations of the stochastic space. Then the $L^2(\Omega, E)$-based relative error estimator is defined as in [75],

$$\varepsilon = \sqrt{\frac{1}{K} \sum_{k=1}^{K} \varepsilon_k^2/K}, \quad (6.1)$$

where:

$$\varepsilon_k = \frac{\|E_{\text{ref}}[\cdot (t^n)] - E_k[\cdot (t^n)]\|_E}{\|E_{\text{ref}}[\cdot (t^n)]\|_E} \times 100\%, \quad (6.2)$$

Definitions (6.1) - (6.2) are generalized to any $E$-valued statistical estimators on $U$ by replacing the expectation estimators $E_{\text{ref}}[\cdot]$ and $E_k[\cdot]$ with the required estimators. The extensive analysis for the appropriate choice of $K$ is conducted in [75]; unless indicated otherwise, we choose $K = 30$ which proved to be sufficient in our numerical experiments to remove statistical fluctuations in the convergence plots.

Equipped with the above notation and concepts, we proceed to numerical experiments.
Setup for a numerical study of MLMC-FVM
7 MLMC-FVM for random initial data

7.1 Euler equations

7.1.1 Sod shock tube with uncertain shock location in 1-D

Problem setup

Let \( Y \sim 1 + U(0, \frac{1}{10}) \) be a random variable with \( U(0, \frac{1}{10}) \) being a uniform distribution in interval \([0, \frac{1}{10}]\). We consider one dimensional version of the Euler equations (1.8) in domain \( D = [0, 2] \), with Neumann ("transparent") boundary conditions (1.104) - (1.105), and with random initial shock at uncertain location (near \( x = 1 \)):

\[
U_0(x, \omega) = \{\rho_0(x, \omega), u_0(x, \omega), p_0(x, \omega)\} = \begin{cases} 
3.0, 0.0, 3.0 & \text{if } x < Y(\omega), \\
1.0, 0.0, 1.0 & \text{if } x > Y(\omega).
\end{cases} \tag{7.1}
\]

The initial data (7.1) and the reference solution at time \( t = 0.5 \) are depicted in Figure 7.1. At every point \( x \in [0, 2] \) the solid line represents the mean and the dashed lines represent the mean ± standard deviation of the (random) solution. For each sample the initial shock splits into three waves: a left going rarefaction wave, a right going contact discontinuity and a right going shock wave. Notice the improvement of the regularity in the stochastic solution: deterministic pathwise solutions for each sample are discontinuous due to formation of the shock and the contact; nevertheless, the mean of the solution is continuous.

Remark 7.1.1. The "overshoots" in dashed lines representing mean value ± standard deviation do not imply that the random shock amplitude can exceed the mean depicted in solid line; in fact, the shock amplitude is the same for every sample path.

Numerical convergence analysis in 1-D

Using MLMC-FVM approximation from Figure 7.1 as a reference solution, we run MC-FVM and MLMC-FVM methods on the series of mesh resolutions ranging from 32 cells up to 8192 cells and monitor the convergence behavior. The number of levels for the MLMC-FVM method is chosen so that the coarsest level contains 8 cells. In the convergence analysis we choose \( M_L = 16 \) unless indicated otherwise. Having notations
MLMC-FVM for random initial data

Figure 7.1: Reference solution computed using MLMC-FVM. The solution is computed with a second-order WENO scheme using a HLL solver on a grid size of 32768 cells. The number of levels of refinement is 12 and we use 16 samples at the finest level. Note that the initial shock splits into 3 waves all having uncertain location. The ensemble average is more regular (continuous) than its paths (with shocks).

<table>
<thead>
<tr>
<th>$L$</th>
<th>$M_L$</th>
<th>grid size</th>
<th>CFL</th>
<th>cores</th>
<th>runtime</th>
<th>efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>16</td>
<td>32768</td>
<td>0.475</td>
<td>104</td>
<td>1:49:49</td>
<td>99.1%</td>
</tr>
</tbody>
</table>

Figure 7.2: Convergence of mean of Sod shock tube problem for Euler equations (1.8) with random initial data (7.1). Both MLMC (MLMC2) and MC (MC2) give similar (up to the expected logarithmic term in (4.19)) errors for the same spatial resolution. However, there is a significant difference in runtime: MLMC methods are more than an order of magnitude faster that pure MC.
and definitions in place, we proceed to the convergence plots of the estimated mean and variance.

Dashed lines in Figure 7.2 (and all subsequent figures) indicate expected convergence rate slopes obtained by theory for the scalar case (see (3.22) and (4.7)). We expect them to coincide with the observed empirical convergence rates (see EOC in section 6.2) for hyperbolic systems of nonlinear conservation laws and in this particular case they are actually very similar. Findings coincide with the results published in [75] confirming the robustness of the implementation.

**Remark 7.1.2.** Due to additional log(Work) term in (4.19) for the case \( s = (d + 1)/2 \), the error vs. runtime convergence rate for MLMC2 is achieved only asymptotically.

![Figure 7.3: Convergence of variance of Sod shock tube problem for Euler equations (1.8) with random initial data (7.1). MLMC methods are more than 2 orders of magnitude faster than pure MC.](image)

In Figure 7.3 we show convergence plots for variance. The observed empirical convergence rate for MLMC2 is again slightly smaller than expected. This could be attributed to two reasons. First, as indicated before, the optimal convergence rate is only expected asymptotically when \( s = (d + 1)/2 \) which is the case for formally second order schemes in one space dimension. We assume that second order schemes converge with rate one in the presence of shocks. The second reason could be the amplitude of the fourth moment as the error estimate of the variance relies on the fourth moment. We compute the 4th centered moment in Figure 7.4 and observe that it is of relatively small amplitude.

The performance of the MC and MLMC methods depends on how the number of samples (at each level of resolution) are chosen. Our choices are governed by rigorously obtained error estimates (3.22) and (4.7), respectively, for scalar conservation laws. The only parameter in these estimate is the rate of convergence \( s \) of the underlying finite volume scheme. Here, we again rely on the rigorously proved convergence rate \( s = 1/2 \) for a first-order scheme by the theorem of Kuznetsov [30]. It is well known that this rate might not be optimal in some cases and we obtain better rates of convergence \( (\approx 1) \) numerically. However, there is no proof that this is indeed the case generally. We therefore rely on
MLMC-FVM for random initial data

Figure 7.4: The fourth centered moment of Sod shock tube problem for Euler equations (1.8) with random initial data (7.1). The MLMC-FVM solution is computed with a second-order WENO scheme using a HLL solver on a uniform grid in $D = [0, 2]$ with 4096 cells so that $\Delta x = 1/2048$. The number of levels of refinement is 9 and we use 16 samples at the finest level.

Figure 7.5: MLMC and MC methods comparison with assumed convergence rates $s = 1/2$ and $s = 1$. We observe $s = 1$ leads to much more work for an equivalent error level, hence $s = 1/2$ is a much better choice for this particular problem and both methods.
the error estimate of Kuznetsov. To test whether this choice is a good one, we repeat the above experiment with a different choice of $s = 1$ for a first-order finite volume scheme. Consequently, the number of MC samples given by (3.24) is $M = O(\Delta x^{-2})$ and the number of MLMC samples given by (4.15) is given by $M_\ell = M_L A(L-\ell)$ with $M_L$ being the number of samples at the finest level. The numerical convergence results for this experiment are shown in Figure 7.5. In this figure, we show the error vs. number of cells (left) and error vs. runtime (right) obtained with a first-order finite volume scheme with assumed rates of convergence $s = 1/2$ and $s = 1$. The results show that both choices led to very similar error for any fixed resolution. However, there is a significant difference in computational efficiency. For both MC as well MLMC methods with $s = 1/2$ require considerably less time to compute the same level of error than their $s = 1$ counterparts. This shows that the choice $s = 1/2$ is not only based on the rigorous error estimate of Kuznetsov, but is also computationally efficient in both the MC as well as MLMC contexts. Furthermore, we see that the MLMC method with either $s = 1/2$ or $s = 1$ is at least two orders of magnitude (asymptotically) faster than its MC counterpart with the same choice of $s$ implying that the MLMC method is fairly robust with respect to the assumed rate of convergence.

### 7.1.2 Non-monotonicity of the MLMC-FVM estimator

Notice that the standard MC-FVM estimator (3.2) is clearly monotone, i.e. the sample mean estimator of positive quantities (such as pressure, density) is also positive. This is due to the fact that the MC-FVM estimator is a convex combination of all samples. However, the MLMC-FVM estimator (4.4) is not monotone, i.e. negative undershoots might occur, even if positivity preserving FVM discretizations are used. More generally, if values of FVM approximations of $U^T_n(x)$ are $\mathbb{P}$-a.s. in some connected set $C = [U_{\text{min}}, U_{\text{max}}] \subset \mathbb{R}$, then the MC-FVM estimator in (3.19) will also be $\mathbb{P}$-a.s. in the same set $C$. The same is not true for the MLMC-FVM (4.4), i.e. the multi-level estimator $E^L[U^T_n]$ might have values smaller than $U_{\text{min}}$ or $U_{\text{max}}$.

Next, we provide two examples, numerical and analytical, exposing the reason for the non-monotonicity of MLMC-FVM estimates. An additional example, exposing the decay of the non-monotonic overshoots of the MLMC-FVM estimator, is given in subsection 9.2.2.

#### Numerical example

For the numerical example, consider only two levels and compute the relevant MC terms for the density $\rho_0$ and $\rho_1$ of the FVM solutions $U^T_0$ and $U^T_1$:

$$E_{M_1}[\rho_1(\cdot,t) - \rho_0(\cdot,t)] + E_{M_0}[\rho_0(\cdot,t) - \rho_{-1}(\cdot,t)] \equiv 0 \quad (7.2)$$
Figure 7.6 depicts each individual term. Notice that in the interval \([1.0, 1.1]\) the (blue, 'level1_pos') term \(E_{M_1}[\rho_1(\cdot, t)]\) nearly vanishes; however, the (green, 'level1_neg') term \(E_{M_1}[\rho_0(\cdot, t)]\) significantly dominates the (red, level0_pos) term \(E_{M_0}[\rho_0(\cdot, t)]\) which results in the overall negative combined estimate (7.2). This type of behavior is generic near shocks with uncertain location where the solution is under-resolved on coarser meshes. Since \(M_1 < M_0\), the number of MC samples for the (green, 'level1_neg') term \(E_{M_1}[\rho_0(\cdot, t)]\) is smaller than for (red, level0_pos) term \(E_{M_0}[\rho_0(\cdot, t)]\); the latter approximates vanishing variance more accurately and hence with high probability is dominated by \(E_{M_1}[\rho_0(\cdot, t)]\).

Figure 7.6: MC terms in the telescoping sum of MLMC-FVM solution. Notice that in the interval \([1.0, 1.1]\) the blue term \(E_{M_1}[\rho_1(\cdot, t)]\) nearly vanishes; however, the green term \(E_{M_1}[\rho_0(\cdot, t)]\) significantly dominates the red term \(E_{M_0}[\rho_0(\cdot, t)]\) which results in the overall negative combined estimate (7.2). This type of behavior is generic near shocks with uncertain location where the solution is under-resolved on coarser meshes. Since \(M_1 < M_0\), the number of MC samples for the (green, 'level1_neg') term \(E_{M_1}[\rho_0(\cdot, t)]\) is smaller than for (red, level0_pos) term \(E_{M_0}[\rho_0(\cdot, t)]\); the latter approximates vanishing variance more accurately and hence with high probability is dominated by \(E_{M_1}[\rho_0(\cdot, t)]\).

Notice that since the problem lies essentially in the multi-level MC estimator, the non-preservation of positivity will also show up in the mean field estimates for positive quantities such as density or pressure.

**Analytical example**

For the analytical example, consider two levels, i.e. \(L = 1\), one sample on the level \(\ell = 0\), i.e. \(M_0 = 1\), and two samples on the finer level \(\ell = 1\), i.e. \(M_1 = 2\). The coarsest (one-dimensional) mesh level is assumed to have only one cell, i.e. \(T_0 = \{C_1^0\}\), and the finer mesh level is assumed to have two equal cells, i.e. \(T_1 = \{C_1^1, C_2^1\}\). Then, assume we have random samples \(\omega_0^1, \omega_1^1, \omega_2^1\) for random input data. Furthermore, we assume that the event values for these samples on different mesh resolutions are given by:

\[
\begin{align*}
U_0(C_1^0, \omega_0^0) &= 1, \\
U_1(C_1^1, \omega_1^1) &= 1, \quad U_1(C_2^1, \omega_1^2) = 0, \\
U_1(C_1^2, \omega_1^1) &= 1, \quad U_1(C_2^2, \omega_2^1) = 0, \\
U_0(C_2^1, \omega_1^1) &= 1, \quad U_0(C_2^2, \omega_2^1) = 0.
\end{align*}
\]
7.1 Euler equations

Clearly, the values for all samples of $U$ lie within the interval $[0, 1]$. The corresponding MC-FVM estimates $E_{M_0}[U_0]$, $E_{M_1}[U_1]$ and $E_{M_1}[U_0]$ are

$$E_{M_0}[U_0] = 1, \quad E_{M_1}[U_1(C_1^1)] = 1, \quad E_{M_1}[U_1(C_2^1)] = \frac{1}{2}, \quad E_{M_1}[U_0] = \frac{1}{2} = \frac{1}{2},$$

which lead to the following MLMC-FVM estimate (4.4) for each cell $C_1^1, C_2^1$

$$E^L[U(C_1^1)] = 1 + 1 - \frac{1}{2} = 1 - \frac{1}{2}, \quad E^L[U(C_2^1)] = 1 + \frac{1}{2} - \frac{1}{2} = 1.$$

Clearly $E^L[U(C_1^1)] > 1$. The counter-example resulting in $E^L[U(C_1^1)] < 0$ can be obtained analogously. Notice that even if we reuse samples from the coarsest mesh level $\ell = 0$ on the finer mesh level $\ell = 1$, i.e. we set $\omega_1^1 = \omega_0^1$, the counter-example is still consistent, since $\omega_1^1 \neq \omega_2^1$.

Interpretation of the numerical results of MLMC-FVM

The above considerations do not mean that the MLMC-FVM approximation of $E^L[U^0_L]$ is wrong or inconsistent with the exact value, i.e. the non-monotonicity of MLMC-FVM is not related to numerical instabilities (taking into account the small magnitude of the undershoot in variance, one could suspect this to be some numerical instability artifact), bias of the estimators, inappropriate FVM schemes or other implementation errors. On the contrary, the MLMC-FVM approximation $E^L[U^0_L]$ converges to the exact value $E[U]$, as proved for multiple settings in section 4.2. In order to avoid confusion, in plots throughout this manuscript, the approximated values of $E^L[U^0_L]$ were clipped to the problem-specific interval $[U_{\min}, U_{\max}]$. Analogously, in variance plots throughout this manuscript, the variance was clipped to $[0, \infty]$. The consequences of such clipping are numerically negligible; however, the awareness of this non-monotonicity of the MLMC-FVM estimator is essential for the interpretation of the numerical results. In particular, it is important to realize that undershoots might occur, which are not related to numerical instability, bias of the estimators, inappropriate FVM schemes or other implementation errors.

7.1.3 Empirical probability density functions

The MC-FVM numerical method presented in chapter 3 for approximation of statistical moments of random entropy solution of (1.133) is exactly the same as the method described in [37] for the approximation of entropy measure valued solutions (EMVS) of nonlinear systems of conservation laws, where the given initial data is a Young measure and hence can be sampled using MC. Such combined MC-FVM methods were proved to provide stable and convergent approximations of EMVS and its statistical moments, see [37]. As plain MC-type methods entail prohibitively large computational costs of the simulation, we consider and numerically analyze the applicability of multi-level MC-FVM
MLMC-FVM for random initial data

methods (introduced in chapter 4) for the estimation of empirical probability density functions of random solution U for well-posed systems (1.133) as well as of EMVS for general systems (1.133) with initial data as Young measure. We note, however, the convergence of MLMC-FVM to EMVS, unlike of MC-FVM in [37], has not yet been considered and is still an open problem.

Following [37], we construct MLMC-FVM approximations of the empirical probability density function of the EMVS (or, equivalently, random solution U(\cdot, \cdot, \omega)) for Sod shock tube in one spatial dimension (see subsection 7.1.1 for detailed problem statement).

The probability distribution (probability measure) \( f[\rho(x, t)] \) of the physical density variable \( \rho(x, t) \) in (1.8) at a given space-time point \( (x, t) \) is given by the distributional derivative of the cumulative distribution function (CDF) \( F[\rho(x, t)] \),

\[
  f[\rho(x, t)](\xi) = \frac{\partial}{\partial \xi} F[\rho(x, t)](\xi), \quad F[\rho(x, t)](\xi) = \mathbb{P}(\rho(x, t) < \xi), \quad \xi \in \mathbb{R},
\]

where differentiation is considered in the distributional sense, and \( f[\rho(x, t)](\xi) \) is a distribution (measure) over \( \mathbb{R} \).

Given a finite a-priori range \( R = [\rho_{\min}, \rho_{\max}] \) of possible values for \( \rho \) and a corresponding partition of \( R \) consisting of intervals \( B_1, \ldots, B_N \), for \( N \in \mathbb{N} \), considered as “bins”, the MC-FVM approximation \( f_M[\rho^n_T(x)] \) of \( f[\rho(x, t^n)] \) with \( M \) Monte Carlo samples is defined using the histogram approximation of \( f[\rho(x, t^n)] \) to bins \( B_1, \ldots, B_N \),

\[
  f_M[\rho^n_T(x)](\xi) = E_M[\chi_{B_j}(\rho^n_T(x, \omega))] = \frac{1}{M |R|} \sum_{i=1}^{M} \chi_{B_j}(\rho^n_T(x, \omega)), \quad \forall \xi \in B_j, \quad (7.4)
\]

where \( \chi_{B_j}(\cdot) \) denote characteristic functions of the sets \( B_j, j = 1, \ldots, N \). Notice that \( f_M[\rho^n_T(x)](\xi) \) is well-defined only in the a-priori prescribed range \( R = [\rho_{\min}, \rho_{\max}] \).

The MLMC-FVM estimator \( f^L[\rho^n_T(x)] \) of \( f[\rho(x, t^n)] \) is defined analogously as in (4.4),

\[
  f^L[\rho^n_T(x)](\xi) = f_{M_0}[\rho^n_{T_0}(x)](\xi) + \sum_{\ell=1}^{L} f_{M_\ell}[\rho^n_{T_\ell}(x) - \rho^n_{T_{\ell-1}}(x)], \quad (7.5)
\]

where \( f_{M_\ell}[\rho^n_{T_\ell}(x)] \) and \( f_{M_\ell}[\rho^n_{T_{\ell-1}}(x)] \) are MC-FVM estimators from (7.4).

Empirical approximations of probability distributions using the MLMC-FVM scheme, are provided in Figure 7.7 for initial time \( t = 0 \) and in Figure 7.8 for final time \( t = 0.5 \). The number of “bins” used is \( B = 512 \), and the range was set to \([\rho_{\min}, \rho_{\max}] = [0.5, 3.5] \).

In the left plots of both figures 7.7 - 7.8, the initial (at \( t = 0 \)) and the evolved (at \( t = t^n = 0.5 \)) empirical density functions \( f^L[\rho^n_T(x)](\xi) \) and \( f^L[\rho^n_T(x)](\xi) \) are depicted, respectively, with \( x \) on the abscissa (x-axis) and \( \xi \) on the ordinate (y-axis), following the structure of the plots in Figure 7.1. The gray-scale color of the line indicates the value of \( f^L[\rho^n_T(x)](\xi) \), representing the probabilities of all possible values for random density field.
\( \rho(x, t^n, \omega) \). The white color indicates \( f^L[\rho^n(x)](\xi) = 0 \) and represents the remaining set of impossible (i.e. with probability equal to 0) values for \( \rho(x, t^n, \omega) \).

In the right plots of both figures 7.7 - 7.8 the empirical density functions \( f^L[\rho^n(x)](\xi) \) and \( f^L[\rho^n_2(x)](\xi) \) are depicted, respectively, with spatial argument \( x \in [0, 2] \) fixed to the value indicated by the red dotted line in the corresponding left plot. For easier comparison between the left and right plots, variable \( \xi \) is again on the ordinate (y-axis), whereas values of \( f^L[\rho^n(x)](\xi) \) are on the abscissa (x-axis), representing the probabilities of all possible values for random density field \( \rho(x, t, \omega) \) at a specified spatial location \( x \in [0, 2] \) and time \( t = t^n \). Additionally, for visualization purposes, in the right plots we depict the approximated probability mass \( |B_j| \cdot f^L[\rho^n(x)](B_j) \) of several selected bins \( B_j \) with largest probability mass, which potentially could contain an atomic part of \( f[\rho(x, t^n)] \), i.e. a Dirac distribution \( \delta_\xi \) with \( \xi \in B_j \).

![Figure 7.7: Initial empirical probability density](image)

At \( t = 0 \), the initial distribution \( f^L[\rho^0_1(x)](\xi) \) specified by \( 7.1 \) consists of atomic probability measures away from the interval \([1, 1.1]\), and it consists of two Dirac \( \delta \)-functions inside the interval \([1, 1.1]\).

At \( t^n = 0.5 \), the empirical distribution \( f^L[\rho^n(x)](\xi) \) approximated by MLMC-FVM...
MLMC-FVM for random initial data

Figure 7.8: MLMC-FVM estimates of probability density \( f_L[\rho(x)](\xi) \) of \( \rho \) at \( t^n = 0.5 \) for all \( x \in [0, 2] \) (left plots) and for fixed \( x = 0.5, x = 1.25 \) and \( t = 1.85 \) (right plots). Uniform distribution is observed in the rarefaction region, whereas regions near the contact discontinuity and shock have di-atomic Dirac distributions.

\[
\begin{array}{|c|c|c|c|c|c|}
\hline
L & M_L & \text{grid size} & CFL & \text{cores} & \text{runtime} \\
8 & 8 & 4096 & 0.475 & 1 & 0:44:53 \\
\hline
\end{array}
\]
7.1 Euler equations

scheme consists of four (approximate) regions

\[ [0, 0.35], \ [0.75, 1.2], \ [1.3, 1.8] \text{ and } [1.9, 2.0] \]

of domain \( D = [0, 2] \) with the monatomic Dirac distributions \( f_L^a(\rho^a_T(x))(\xi) = \delta_0(\xi) \) for \( a \in \{3, a_1, a_2, 1\} \) with \( 3 > a_1 > a_2 > 1 \), as depicted in the left plots of Figure 7.8.

In-between these four regions with monatomic Dirac distributions, there are three regions, corresponding to rarefaction, contact discontinuity and shock. In the right plots of Figure 7.8 approximations of \( f_L^a(\rho^a_T(x))(\xi) \) using MLMC-FVM at three fixed spatial parameters \( x \in [0, 2] \) are presented, corresponding to the three regions with non-monatomic distributions \( f_L^a(\rho^a_T(x))(\xi) \).

In the rarefaction region (first plot in Figure 7.8 with \( x = 0.5 \)), the empirical distribution \( f_L^a(\rho^a_T(0.5))(\xi) \) resembles the uniform distribution with finite support \([2.5, 2.9]\).

In the contact discontinuity region (second plot in Figure 7.8 with \( x = 1.25 \)), the empirical distribution \( f_L^a(\rho^a_T(1.25))(\xi) \), analogously to initial distribution \( f_L^a(\rho^a_T(x, 0))(\xi) \) in the region \( x \in [1, 1.1] \), consists of two Dirac \( \delta \)-functions,

\[
f_L^a(\rho^a_T(x))(\xi) = c_1(x)\delta_{a_1}(\xi) + c_2(x)\delta_{a_2}(\xi), \quad x \in [1.2, 1.3],
\]

with \( c_1(x) \) linearly decreasing from 1 to 0 and \( c_1(x) \) linearly increasing from 0 to 1 as functions of \( x \) in \([1.2, 1.3] \).

Finally, in the shock region (third plot in Figure 7.8 with \( x = 1.85 \)), the empirical distribution \( f_L^a(\rho^a_T(1.85))(\xi) \) again consists of two Dirac \( \delta \)-functions,

\[
f_L^a(\rho^a_T(x))(\xi) = c_1(x)\delta_{a_1}(\xi) + c_2(x)\delta_{1}(\xi), \quad x \in [1.8, 1.9],
\]

with \( c_1(x) \) linearly decreasing from 1 to 0 and \( c_1(x) \) linearly increasing from 0 to 1 as functions of \( x \) in \([1.8, 1.9] \).

7.1.4 Verification of the derived constants in the asymptotic error bounds

In this section, we would like to verify the results derived in subsection 4.2.3, i.e. that there is only a small (of order \( \sqrt{2M_L} \)) difference in constants of the asymptotic error convergence bounds \((3.28)\) and \((4.19)\) or \((4.28)\). To this end, consider the deterministic and the stochastic one-dimensional Sod shock tube problems presented in \((4.2.1)\). Four different configurations are considered:

<table>
<thead>
<tr>
<th>Initial data</th>
<th>Solver</th>
<th>Reconstruction</th>
<th>CFL</th>
</tr>
</thead>
<tbody>
<tr>
<td>DET</td>
<td>deterministic</td>
<td>HLL 3-wave</td>
<td>-</td>
</tr>
<tr>
<td>MLMC</td>
<td>stochastic</td>
<td>MLMC + 3-wave HLL</td>
<td>-</td>
</tr>
<tr>
<td>DET2</td>
<td>deterministic</td>
<td>HLLC 3-wave</td>
<td>WENO</td>
</tr>
<tr>
<td>MLMC2</td>
<td>stochastic</td>
<td>MLMC + HLL 3-wave</td>
<td>WENO</td>
</tr>
</tbody>
</table>
MLMC-FVM for random initial data

The convergence results, i.e. error vs. number of cells and error vs. computational runtime are depicted in Figure 7.9. In the table below the figure, the approximated constants $C_{FVM}$ and $C_L$ from asymptotic error bounds (3.28) and (4.19)/(4.28) are provided. The values of the constants were obtained by the least squares fitting of the linear polynomial to the obtained logarithmic error convergence data. The measured differences in the constants are small (approximately $1.4$ and $1.75$ times) and hence support the derived estimate (4.31) presented in subsection 4.2.3.

![Figure 7.9](image)

Figure 7.9: Error convergence for deterministic and stochastic Sod shock tube problem.

For some fixed computational time, errors of the stochastic (MLMC(2)) runs are only approximately $1.4 - 1.75$ times larger than errors of the deterministic (DET(2)) runs.

### 7.1.5 Shock-vortex interaction in 2-D

In this standard test case for deterministic solvers (see [97]), the computational domain is taken to be $D = [0, 1] \times [0, 1]$, with Neumann (“transparent”) boundary conditions (1.104) - (1.105). Let $Y \sim \frac{1}{2} + U(0, \frac{1}{10})$. The initial random stationary Mach 1.1 shock is normal to $x$ axis and has uncertain location with mean at $x_1 = \frac{1}{2} + \frac{1}{20}$. The resulting initial data are:

$$\{\rho_0(x, \omega), u_0(x, \omega), p_0(x, \omega)\} = \begin{cases} 
\{1, (\sqrt{\gamma}, 0)^\top, 1\} & \text{if } x_1 < Y(\omega), \\
\{\frac{1}{1.1}, (1.1\sqrt{\gamma}, 0)^\top, 1 - \frac{\gamma}{10}\} & \text{if } x_1 > Y(\omega).
\end{cases} \quad (7.6)
$$

This base flow is superposed with the a vortex centered at $(x_c, y_c)$:

$$\bar{u}_0(x, \omega) = (\epsilon \tau e^{\alpha(1-\tau^2)} \sin \theta, -\epsilon \tau e^{\alpha(1-\tau^2)} \cos \theta)^\top, \quad (7.7)$$

$$\bar{p}_0(x, \omega) = -(\gamma - 1) \frac{\epsilon^2 e^{\alpha(1-\tau^2)}}{4\alpha \gamma \rho}, \quad (7.8)$$
7.1 Euler equations

where:

\[ r = \sqrt{(x_1 - x_c)^2 + (x_2 - y_c)^2}, \quad \tau = \frac{r}{r_c}, \quad \sin \theta = \frac{x_2 - y_c}{r}, \quad \cos \theta = \frac{x_1 - x_c}{r}. \]  

(7.9)

and we choose the parameters to be:

\[ \epsilon = 0.3, \quad r_c = 0.05, \quad \alpha = 0.204, \quad x_c = 0.25, \quad y_c = 0.5. \]  

(7.10)

![Image of mean of rho and variance of rho at t=0.35](image)

Figure 7.10: Shock-vortex interaction solution at time \( t = 0.35 \) using MLMC. The mean of the pressure shows that the stationary shock (with uncertain location) is smoothed out and the vortex is resolved quite sharply. The initial variance (concentrated on the shock) is redistributed by the flow - there is a clear signature of the vortex in the variance although the vortex has already crossed the shock.

The location of the initial stationary supersonic shock is uncertain in this experiment. For each realization of the initial data, the pathwise solution consists of the vortex moving to the right, interacting with the shock and emerging out of the shock. Since the shock location is uncertain, we expect the mean to be smoother than the pathwise solutions. A-priori, it is unclear how the variance is going to be redistributed.

The results of uncertain shock-vortex interaction simulation at time \( t = 0.35 \) are given in Figure 7.10. At this time instant, the vortex has just emerged out of the shock. We present results for a run with a second-order WENO discretization in physical space. The HLLC solver is used for computing the numerical fluxes. The results are computed on 9 nested levels of resolution \( (L = 8) \) with the finest resolution being on a 2048 × 2048 mesh and with timesteps reduced accordingly in order to maintain the same CFL constant over all discretization levels. The simulation is run on 128 cores and 16 samples are taken for the finest mesh resolution.
MLMC-FVM for random initial data

The results in Figure 7.10 show that the MLMC-WENO scheme is quite robust. The mean of the pressure shows that the stationary shock (with uncertain location) is smoothed out. Similarly, the vortex (in mean) has emerged from the shock at this time instant. The vortex is resolved quite sharply. The plot for the variance of the pressure is more interesting. Although the initial variance was concentrated on the shock, we see that the variance is redistributed by the flow. A large proportion of the variance is still concentrated on the shock. However, the profile is distorted near the point where the vortex was incident at the shock. Furthermore, there is a clear signature of the vortex in the variance although the vortex (in mean) has already crossed the shock. The above experiment reveals that the variance (and possibly higher moments) can have a much more complex behavior than the mean flow field. It requires a very efficient numerical method to be able to resolve such complex features.

7.1.6 Cloud shock in 2-D

So far, we have presented numerical experiments for the Euler equations with only one source of uncertainty - uniformly distributed random initial shock location. In reality, the number of uncertainty sources can be quite large. Stochastic Galerkin methods based on gPC expansions will be deficient at resolving such problems as the computational complexity grows exponentially with the number of uncertainty sources (number of stochastic dimensions or terms in the gPC expansion). On the other hand, methods that are based on MC-sampling methods are more suitable for such high dimensional problems. As an example, consider uncertain initial data with a large number of sources for uncertainty. There is a very negligible increase of computational cost over the case of a single source of uncertainty as a random vector (with the number of components corresponding to the number of uncertainty sources) is drawn instead of a random number at each mesh point and only at the initial time step. For uncertain sources or boundary conditions, the computational cost increases only linearly w.r.t. the number of uncertain parameters.

We put the above hypothesis to test on a problem with a high number of uncertainty sources.

We consider the so-called cloud-shock interaction problem. The computational domain is taken to be \( D = [0, 1] \times [0, 1] \), with Neumann (“transparent”) boundary conditions (1.104) - (1.105). Let \( Y \sim \mathcal{U}(0, \frac{1}{25}) \) and let \( Y_1, ..., Y_7 \sim \mathcal{U}(0, 1) \) denote i.i.d. random variables independent of \( Y \).

The initial data consists of an initial shock with uncertain amplitude and uncertain location given by:

\[
\begin{align*}
\{ \rho_0(x, \omega), \ u_0(x, \omega), \ p_0(x, \omega) \} &= \begin{cases} 
3.86859 + \frac{1}{10} Y_0(\omega), \ (11.2536, 0)^\top, \ 167.345 + Y_7(\omega) & \text{if } x_1 < Y(\omega), \\
1, \ (0, 0)^\top, \ 1 & \text{if } x_1 > Y(\omega). 
\end{cases}
\end{align*}
\]
Furthermore, a high density cloud or bubble with uncertain amplitude and uncertain shape of the form

$$\rho_0(x, \omega) = 10 + \frac{1}{2} Y_1(\omega) + Y_2(\omega) \sin(4(x_1 - \frac{1}{4})) + \frac{1}{2} Y_3(\omega) \cos(8(x_2 - \frac{1}{2}))$$

if \( r \leq 0.13 + \frac{1}{50} Y_4(\omega) \sin \theta + \frac{1}{100} Y_5(\omega) \sin(10\theta) \),

where

$$r = \sqrt{(x_1 - 0.25)^2 + (x_2 - 0.5)^2}, \quad \theta = \frac{x_1 - 0.25}{r},$$

lies to the right of the shock. The mean and the variance of the initial data are depicted in Figure 7.11(a). Note that there are 8 sources of uncertainty in the above problem. A parametric representation of the initial data results in a 11 dimensional problem consisting of two space, one time and eight stochastic dimensions. The mean and variance of the solution at time \( t = 0.06 \) is shown in Figure 7.11(b). The results are from a MLMC-WENO run with 10 nested levels of resolution (\( L = 9 \)) and the finest resolution is set to 4096 × 4096 mesh. The number \( M_L \) of MC samples at the finest resolution is 8 and the number of cores for this run is 1023.

The physics of the flow in this case consists of the supersonic initial shock moving to the right, interacting with the high density bubble and leading to a complex flow pattern that consists of a leading bow shock, trailing tail shocks and a very complex region (near the center) possessing sharp gradients as well as turbulent like smooth features. The mean flow (for the density) consists of the bow shock, tail shocks and a complex region with sharp gradients as well as smooth regions. The variance is concentrated in the smooth region at the center; it is significantly smaller at the tail shocks and almost vanishing at the bow shock. The initial uncertainty in the shape of the bubble seems to lead to a more complex distribution of the variance.

### 7.1.7 Cloud shock in 3-D

We consider three-dimensional Euler equations in domain \( D = [0, 1]^3 \), with Neumann (“transparent”) boundary conditions (1.104) - (1.105), and the so-called cloud-shock initial data with 11 sources of uncertainty, i.e. with random initial shock at random location (near \( x = 0.1 \)) heading towards high density cloud with uncertain shape of its boundary and uncertain inner density. One sample of the 3-D cloud-shock obtained using deterministic FVM is provided in Figure 7.12. The mean and variance for the density of the MLMC-FVM solution at initial time \( t = 0 \) as well as at the time \( t = 0.06 \) are shown in Figures 7.13(a) - 7.13(b). The results are from a MLMC-WENO run with 7 nested levels of resolution (\( L = 6 \)) and the finest resolution is set to 1024^3 mesh. The flow in this case consists of the supersonic initial shock moving to the right, interacting with the high density bubble and leading to a complex flow pattern that consists of a leading bow shock, trailing tail shocks and a very complex center region possessing sharp gradients as well as turbulent like smooth features. Runtime was approximately 4 and a half hours on 21 844 cores.
MLMC-FVM for random initial data

Figure 7.11: Cloud shock at $t = 0$ and $t = 0.06$ using MLMC-FVM. The mean flow (for the density) consists of the bow shock, tail shocks and a complex region with sharp gradients as well as smooth regions. The variance is concentrated in the smooth region at the center; it is significantly smaller at the tail shocks and almost vanishing at the bow shock. The initial uncertainty in the shape of the bubble seems to lead to a more complex distribution of the variance.
7.1 Euler equations

<table>
<thead>
<tr>
<th>$L$</th>
<th>$M_L$</th>
<th>grid size</th>
<th>CFL</th>
<th>cores</th>
<th>runtime</th>
<th>efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1024x1024x1024</td>
<td>0.475</td>
<td>4096</td>
<td>4:29:44</td>
<td>95.7%</td>
</tr>
</tbody>
</table>

Figure 7.12: One sample at $t = 0$ (left) and $t = 0.06$ of density in the 3-D cloud-shock. The flow in this case consists of the supersonic initial shock moving to the right, interacting with the high density bubble and leading to a complex flow pattern that consists of a leading bow shock, trailing tail shocks and a very complex center region possessing sharp gradients as well as turbulent like features.
MLMC-FVM for random initial data

Figure 7.13: Mean (left) and variance (right) of density in the 3-D cloud-shock estimated with MLMC-FVM at $t = 0.06$. The results are similar to the analogous 2-D simulation in 7.11(b): the variance is concentrated in the smooth region at the center; it is significantly smaller at the tail shocks and almost vanishing at the bow shock.
7.2 Magnetohydrodynamics equations in 2-D

7.2.1 Problem setup

Next, we consider the MHD equations (1.10) on domain $D = [0, 2] \times [0, 2]$, with periodic boundary conditions (1.102) - (1.103), and random initial data being a parametric version of the celebrated Orszag-Tang vortex, randomly perturbed in two different ways:

1. **2 sources of uncertainty.** Let $Y_1, Y_2 \sim U(0, 1)$. The phases of the velocities are uncertain and depend on the scaled random variables $Y_1, Y_2$:

   \[
   \{\rho_0(x, \omega), p_0(x, \omega)\} = \{\gamma^2, \gamma\},
   \]

   \[
   u_0(x, \omega) = \left(-\sin\left(\pi x_2 + \frac{1}{20} Y_1(\omega)\right), \sin\left(\pi x_1 + \frac{1}{10} Y_2(\omega)\right)\right)^T,
   \]

   \[
   B_0(x, \omega) = (-\sin(\pi x_2), \sin(2\pi x_1))^T.
   \] (7.14)

2. **8 sources of uncertainty.** Let $Y_i \sim U(-1, 1)$, $i = 1, \ldots, 8$. The amplitudes of the initial density and pressure are uncertain

   \[
   \rho_0(x, \omega) = \gamma^2 \left(1 + \frac{1}{20} Y_1(\omega)\right),
   \]

   \[
   p_0(x, \omega) = \gamma \left(1 + \frac{1}{20} Y_4(\omega)\right),
   \] (7.15)

   and, additionally, the phases of the initial velocities and the phases with the amplitudes of the initial magnetic fields are also uncertain,

   \[
   u_0(x, \omega) = \left(-\sin\left(\pi x_2 + \frac{1}{20} Y_2(\omega)\right), \sin\left(\pi x_1 + \frac{1}{10} Y_3(\omega)\right)\right)^T,
   \]

   \[
   B_1(x, \omega) = -\left(1 + \frac{1}{20} Y_6(\omega)\right) \sin\left(\pi x_2 + \frac{1}{25} Y_5(\omega)\right),
   \]

   \[
   B_2(x, \omega) = \left(1 + \frac{1}{20} Y_8(\omega)\right) \sin\left(2\pi x_1 + \frac{1}{20} Y_7(\omega)\right).
   \] (7.16)

Here, as in the setup for cloud-shock interaction problem in Figure 7.11(a), a parametric representation of the initial data results in a 11 dimensional problem consisting of two space dimensions, one time dimension and eight stochastic dimensions.

The MLMC-FVM solution is then considered for both versions of the initial data, i.e. with 2 sources (7.14) and with 8 sources (7.15) of uncertainty. The mean field and the variance (for the plasma density) of the solutions are shown in Figures 7.14 and 7.15 respectively.

The computation is performed using the MLMC-FVM scheme with second-order WENO reconstruction, and with the HLL three wave solver of [41]. The code uses an upwind
MLMC-FVM for random initial data

Figure 7.14: Uncertain Orszag-Tang vortex solution at $t = 1.0$ using MLMC-FVM (2 sources of uncertainty). Variance is very large near discontinuities of the pathwise solutions as well a (smooth) current sheet at the center of the domain. We observe that the variance is a very good indicator of sharp gradients.

<table>
<thead>
<tr>
<th>$L$</th>
<th>$M_L$</th>
<th>grid size</th>
<th>CFL</th>
<th>cores</th>
<th>runtime</th>
<th>efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>4</td>
<td>2048x2048</td>
<td>0.475</td>
<td>128</td>
<td>5:02:14</td>
<td>98.4%</td>
</tr>
</tbody>
</table>

Figure 7.15: Uncertain Orszag-Tang vortex solution at $t = 1.0$ using MLMC-FVM (8 sources of uncertainty). Again, the largest variances appear near discontinuities of the pathwise solutions as well a (smooth) current sheet at the center of the domain. We observe that the variance is a very good indicator of sharp gradients.

<table>
<thead>
<tr>
<th>$L$</th>
<th>$M_L$</th>
<th>grid size</th>
<th>CFL</th>
<th>cores</th>
<th>runtime</th>
<th>efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>4</td>
<td>4096x4096</td>
<td>0.475</td>
<td>2044</td>
<td>3:17:18</td>
<td>97.0%</td>
</tr>
</tbody>
</table>
7.2 Magnetohydrodynamics equations in 2-D

discretization of the Godunov-Powell source term. The results shown in these figures are from a computation with 8 levels of refinement ($L = 7$) and the finest mesh resolutions of $2048 \times 2048$ mesh cells and $4096 \times 4096$ mesh cells for the problem with two sources of uncertainty (7.14) and with eight sources of uncertainty (7.15), respectively. The number of MC samples at the finest resolution for both problems is 4. The problems have more than $10^9$ degrees of freedom per time step and the total number of time steps is about $10^4$ amounting to an overall computational cost of this simulation between $10^{12}$ and $10^{13}$ FLOPS. These numbers show that the simulations are extremely challenging and requires massively parallel architectures. In fact, the problem with 2 sources of uncertainty took about 6 hours (wall-clock) on 128 cores (simulated on ETH’s parallel cluster Brutus [127]) and the problem with 8 sources of uncertainty took about 3.5 hours (wall-clock) on 2040 cores (simulated on Palu, CSCS [126]). We also observe that the variance for the problem with eight sources of uncertainty is more diffused than the variance for the problem with two sources of uncertainty.

It is well known (see [41]) that stable computation of numerical solutions of the Orszag-Tang problem on highly refined meshes (which, by the CFL condition (1.51), entails a correspondingly large number of timesteps) is quite challenging. Since our spatial resolution at mesh level $L = 7$ is very fine, we need an extremely robust code like ALSVID for the solve step in MLMC-FVM in order to resolve this problem.

The mean density is quite complicated with shocks along the diagonals of the domain as well a (smooth) current sheet at the center of the domain. The solution consists of discontinuities interspersed within interesting smooth features. Our simulations show that the variance is concentrated at shocks as well as at the current sheets and other interesting smooth regions. From this problem as well as the results of the previous section, we observe that the variance is a very good indicator of where the discontinuities and sharp gradients of the solution are concentrated and would serve as a good a posteriori error indicator for adaptive mesh refinement.

7.2.2 Numerical convergence analysis in 2-D

We analyze these particular two dimensional numerical experiments (Orszag-Tang vortex with 2 and 8 sources of uncertainty) in greater detail. Again, we use the high-resolution MLMC-FVM simulations from Figures 7.14 and 7.15 as the reference solutions, respectively. We investigate convergence of error vs. work in Figure 7.16 and Figure 7.18 for 2 sources of uncertainty and in Figure 7.17 and Figure 7.19 for 8 sources of uncertainty. The error in the mean field converges at expected rates. At comparable numerical resolution and accuracy, the MLMC(2) is about two orders of magnitude faster than the MC(2) method for both problems. We observe a slight deterioration in the estimated convergence rates for the variance. This could well be a pre-asymptotic effect. As seen in Figures 7.18 and 7.19 the curves are steepening which seems to indicate better rates with further refinement. Again, the MLMC(2) appears considerably faster than the
MLMC-FVM for random initial data

corresponding MC(2) method in delivering variance estimates of comparable numerical accuracy.

Figure 7.16: Convergence of mean in the uncertain Orszag-Tang vortex simulation (2 sources of uncertainty). The error in the mean field converges at expected rates. At comparable numerical resolution and accuracy, the MLMC(2) is about two orders of magnitude faster than the MC(2) method.

Figure 7.17: Convergence of mean in the uncertain Orszag-Tang vortex simulation (8 sources of uncertainty). The results are analogous to results with 2 sources of uncertainty in Figure 7.16. i.e. at comparable numerical resolution and accuracy, the MLMC(2) is about two orders of magnitude faster than the MC(2) method.

Remark 7.2.1. Our aim in computing the Orszag-Tang vortex with two and with eight sources of uncertainty in the initial data is to compare the robustness of the MLMC method with respect of an increase in the number of sources of uncertainty. To this end, we plot the error vs. resolution and the error vs. runtime for the MLMC(2) FVM with both two and with eight sources of uncertainty in Figure 7.20. The results in this figure show that the runtime for a fixed level of error is nearly identical whether there are two or eight sources of uncertainty in the initial data. This shows that the MLMC method is quite robust with respect to large number of sources of uncertainty in the data: an increase in the number of sources of uncertainty does not appear to lead to a
7.2 Magnetohydrodynamics equations in 2-D

Figure 7.18: Convergence of variance in the uncertain Orszag-Tang vortex simulation (2 sources of uncertainty). We observe a slight deterioration in the estimated convergence rates for the variance. This could well be a pre-asymptotic effect. Furthermore, the MLMC(2) again appears considerably faster than the corresponding MC(2) method.

Figure 7.19: Convergence of variance in the uncertain Orszag-Tang vortex simulation (8 sources of uncertainty). The results are analogous to results with 2 sources of uncertainty in Figure 7.18, i.e. the MLMC(2) again appears considerably faster than the corresponding MC(2) method.
MLMC-FVM for random initial data

Figure 7.20: Error convergence sensitivity to the number of sources of uncertainty. For both 2 and 8 sources of uncertainty, the convergence of error vs. computational work is almost identical. This shows that the MLMC method is quite robust with respect to large number of sources of uncertainty in the data: an increase in the number of sources of uncertainty does not appear to lead to a deterioration in the computational efficiency.

7.2.3 Isothermal blast wave in 2-D

In the numerical experiments presented so far, we have considered the initial randomness to be uniformly distributed. The MLMC-FVM algorithm allows, however, initial data with any probability distribution which can be sampled numerically. We demonstrate its versatility with respect to the initial probability distribution in the following experiment where the initial uncertainty is distributed normally.

The computational domain for this isothermal blast wave experiment is taken to be $D = [0, 1] \times [0, 1]$, with Neumann (“transparent”) boundary conditions $\text{(1.104) - (1.105)}$. For the initial condition, we use the standard setup for the deterministic isothermal blast wave, with the magnitude of the magnetic field additionally perturbed by a normally distributed random variable $Y \sim N(0, 1)$:

$$\rho_0(x, \omega) = \begin{cases} 100 & \text{if } (x_1 - \frac{1}{2})^2 + (x_2 - \frac{1}{2})^2 < \left(\frac{1}{20}\right)^2, \\ 0 & \text{otherwise.} \end{cases}$$

$$\{u_0(x, \omega), \rho_0(x, \omega), B(x, \omega)\} = \{(0, 0)^\top, \rho_0(x, \omega), \left(\frac{5}{\sqrt{\pi}} + \frac{1}{5}Y(\omega), 0\right)^\top\}.$$  

The resulting solution is shown in Figure 7.21. We present results of a simulation performed with five levels of refinement and with the mesh at the finest level consisting
7.2 Magnetohydrodynamics equations in 2-D

of 1024 \times 1024 cells. The number of samples on the finest level is 4 and the computation is performed on 508 cores.

The results show that the mean plasma density has a rotating profile with the initial blast wave spreading out from the center of the domain. Furthermore, the variance is concentrated at the outer blast wave.

We point out that even a single deterministic run for the isothermal blast wave at mesh level $L = 6$ is known to be challenging, (see, e.g. [41]). Additionally, as the MC and MLMC are based on randomly generated initial scenarios, robustness of the pathwise solver ALSVID is crucial to handle each run. Furthermore, given the time scales of the problem, only a fast method like MLMC-FVM can handle such complex problems.

Figure 7.21: Isothermal blast wave solution at $t = 0.09$ using MLMC-FVM with normally distributed perturbation of initial magnetic field. The results show that the mean plasma density has a rotating profile with the initial blast wave spreading out from the center of the domain, and the variance of the density is concentrated at the outer blast wave.
MLMC-FVM for random initial data
8 MLMC-FVM for random source terms

8.1 Multi-level alias-free representation of uncertain bottom topography

An approximation to the exact bottom topography \( b(x) \in W^{1,\infty}(D) \) is often obtained from the measurements. For instance \cite{10,35}, in the two-dimensional case, nodal measurements \( b_{i+\frac{1}{2},j+\frac{1}{2}} := b(x_{i+\frac{1}{2},j+\frac{1}{2}}) \) are obtained at locations \( x_{i+\frac{1}{2},j+\frac{1}{2}} = (x_i + \frac{1}{2}, y_j + \frac{1}{2}) \), i.e. at vertices of an axiparallel quadrilateral topography mesh \( \mathcal{T} \) on the rectangular two-dimensional domain \( D \). Since each measurement \( b_{i+\frac{1}{2},j+\frac{1}{2}} \) is prone to uncertainty \cite{35}, all measured values are treated as random variables with some prescribed probability distribution; we choose

\[
b_{i+\frac{1}{2},j+\frac{1}{2}}(\omega) := b(x_{i+\frac{1}{2},j+\frac{1}{2}}) + Y_{i,j}(\omega), \quad Y_{i,j} \sim \mathcal{U}(-\varepsilon_{i,j}, \varepsilon_{i,j}), \quad \varepsilon_{i,j} > 0,
\]

i.e. \( b_{i+\frac{1}{2},j+\frac{1}{2}}(\omega) \in L^2(\Omega, \mathbb{R}) \) are random variables (not necessarily independent, see \cite{35}), which deviate from the measurements \( b_{i+\frac{1}{2},j+\frac{1}{2}} \) by \( \pm \varepsilon_{i+\frac{1}{2},j+\frac{1}{2}} \) with \( \mathcal{U} \) being the uniform distribution (other distributions can be considered analogously.) Thus, (8.1) provides an approximation to the uncertain topography \( b(x,\omega) \in L^2(\Omega, W^{1,\infty}(D)) \).

In two space dimensions, if we assume that the bottom topography \( b(x,\omega) \) is a continuous piecewise linear function, the energy conservative well-balanced discretization of the source term \( S_{i,j} \) was described in subsection 1.4.8 and is given by \cite{35}

\[
S_{i,j}^{EC} := \left[ \begin{array}{c}
\frac{\Delta x}{2} \left( \bar{h}_{i+\frac{1}{2},j}([b])_{i+\frac{1}{2},j} + \bar{h}_{i-\frac{1}{2},j}([b])_{i-\frac{1}{2},j} \right) \\
\frac{\Delta y}{2} \left( \bar{h}_{i,j+\frac{1}{2}}([b])_{i,j+\frac{1}{2}} + \bar{h}_{i,j-\frac{1}{2}}([b])_{i,j-\frac{1}{2}} \right)
\end{array} \right],
\]

which amounts to computing the averages of \( S_{i+\frac{1}{2},j}(\omega) \) and \( S_{i,j+\frac{1}{2}}(\omega) \),

\[
S_{i,j}(\omega) := \left[ \begin{array}{c}
\frac{\Delta x}{2} \left( S_{i-\frac{1}{2},j}(\omega) + S_{i+\frac{1}{2},j}(\omega) \right) \\
\frac{\Delta y}{2} \left( S_{i,j-\frac{1}{2}}(\omega) + S_{i,j+\frac{1}{2}}(\omega) \right)
\end{array} \right],
\]

where \( S_{i+\frac{1}{2},j}(\omega) \) and \( S_{i,j+\frac{1}{2}}(\omega) \) are the integrals of the weak directional derivatives of a continuous piecewise linear bottom topography \( b(x,\omega) \) over “shifted” cells \( C_{i+\frac{1}{2},j}, C_{i,j+\frac{1}{2}} \).
MLMC-FVM for random source terms

of the mesh \( T \),

\[
S_{i+\frac{1}{2},j}(\omega) := \int_{C_{i+\frac{1}{2},j}} \bar{h}_{i+\frac{1}{2},j}(\omega) \partial_x b(x, \omega) \, dx,
\]

\[
S_{i,j+\frac{1}{2}}(\omega) := \int_{C_{i,j+\frac{1}{2}}} \bar{h}_{i,j+\frac{1}{2}}(\omega) \partial_y b(x, \omega) \, dx.
\]

In the one-dimensional case, the last term in (8.3) is dropped and the first integral in (8.4) is taken over cells \( C_{i+\frac{1}{2}} = (x_i, x_{i+1}) \), resulting in the one-dimensional energy conservative source discretization in [1.99]. Since MLMC methods require that a deterministic problem is solved by the means of such FVM on the coarsest mesh level for a very large number (often around \( 10^{10} \)) of samples, the evaluation of the full bottom topography for each sample becomes computationally infeasible.

To remedy this, we consider a hierarchical multi-level representation of the bottom topography. The key to an efficient MLMC simulation of the uncertain random topography (with very large numbers of sources of uncertainty) is that only some of the hierarchical bottom topography levels need to be evaluated; in particular, only these levels, which are coarser than or coincide with the active FVM discretization level, are necessary. An active level is that level of resolution on which the FVM computation is being performed for a given sample.

8.1.1 Preliminaries

To introduce the multi-level topography representation, we recall some notation: levels \( \ell = 0, \ldots, L \) enumerate nested grids \( T_0, \ldots, T_L \) that are used in the MLMC-FVM solver. Apart from \( T_0, \ldots, T_L \), we consider an additional hierarchical structure, that will be used in the multi-level representation of the bottom topography. More precisely, assume a nested sequence \( \{ \bar{T}_\ell = \bar{T}_1^d \times \cdots \times \bar{T}_d^d, \ \ell = 0, \ldots, L \} \) of isotropic regular \( d \)-dimensional axiparallel quadrilateral meshes for the physical bounded domain \( D = I_1 \times \cdots \times I_d \subset \mathbb{R}^d, \ I_r \subset \mathbb{R}, \ d = 1, 2 \), each of them obtained by \( \ell \) uniform refinements of some initial, regular mesh \( \bar{T}_0 \) (of domain \( D \)) consisting of the cells \( C_k = C_{k_1}^1 \times \cdots \times C_{k_d}^d, \ k \in \mathbb{N}_0 \). Note that a-priori we do not assume any relation between \( L \) and \( L \). However, for the sake of consistency, we assume

\[
\bar{T}_\ell = T_\ell, \quad \text{provided } \ell = \ell.
\]

For \( p \in \mathbb{N}_0 \), define \( \mathcal{Q}^p(D, \bar{T}) \) to be the space of piecewise multivariate tensor product polynomials of degree \( p \) on a mesh \( \bar{T} \) of a bounded domain \( D \) having essentially bounded weak derivatives up to order \( p \), i.e.

\[
\mathcal{Q}^p(D, \bar{T}) := \{ f \in W^{p,\infty}(D) : f|_C \in \mathcal{Q}_p(C), \ \forall C = C_1 \times \cdots \times C_d \in \bar{T} \},
\]

where \( \mathcal{Q}_p(C) \) is the space of multivariate tensor product polynomials on cell \( C \),

\[
\mathcal{Q}_p(C) := \{ x \mapsto p_1(x_1) \cdots p_d(x_d) : p_r \in \mathcal{P}_d(C_r), \ \forall r = 1, \ldots, d \}.
\]
Haar wavelets in 1-D

For any interval $I \subset \mathbb{R}$, there is an orthogonal decomposition of $Q^0(I, \hat{T}_L)$,

$$Q^0(I, \hat{T}_L) = \bigoplus_{\ell=0}^\infty R_\ell, \quad R_\ell := Q^0(I, \hat{T}_\ell) \cap Q^0(I, \hat{T}_{\ell-1})^\perp. \quad (8.5)$$

An $L^2(I)$-orthogonal basis of $Q^0(I, \hat{T}_L)$ can be explicitly constructed from the so-called *mother-wavelets* $\{\hat{\psi}\}$, refer to \[21\]. For each $\ell \in \mathbb{N}_0$, every cell $C_k^\ell \in \hat{T}_\ell$ is affinely equivalent to the reference cell $\hat{C} = (0, 1)$, i.e. for all $k = 1, \ldots, \#\hat{T}_\ell$, there are affine mappings

$$F^\ell_k : \hat{C} \ni x \mapsto x \in C_k^\ell \in \hat{T}_\ell, \quad \|DF^\ell_k\| = |C_k^\ell| = O(2^{-\ell}).$$

Let $\hat{T}_0 = \{\hat{C}\}$ and define $\hat{T}_1 = \{\hat{C}_1, \hat{C}_2\}$ to be the set of two cells $\hat{C}_1 = (0, 1/2)$, $\hat{C}_2 = (1/2, 1)$ that are obtained by uniform subdivision of the reference cell $\hat{C}$. Then

$$\dim \left(Q^0(\hat{C}, \hat{T}_0)\right) = 1, \quad \dim \left(Q^0(\hat{C}, \hat{T}_1) \cap Q^0(\hat{C}, \hat{T}_0)^\perp\right) = 1.$$

Denote by $\{\hat{\varphi}\}$ the $L^2(\hat{C})$-normalized basis of $\hat{W}_0 = Q^0(\hat{C}, \hat{T}_0)$ and by $\{\hat{\psi}\}$ the $L^2(\hat{C})$-normalized basis of $\hat{W}_1 = Q^0(\hat{C}, \hat{T}_1) \cap Q^0(\hat{C}, \hat{T}_0)^\perp$. Next we define $\Psi_0$ by

$$\Psi_0 := \left\{\psi^0_k : \psi^0_k \circ F^0_k = \hat{\varphi} \in \hat{W}_0, \ k = 1, \ldots, \#\hat{T}_0\right\},$$

and, for every $\ell \geq 1$, we define $\Psi_\ell$ to be the set of affine images of the (mother-wavelets) $\hat{\psi}_n$ under the affine mappings $F^{\ell-1}_k$, i.e.

$$\Psi_\ell := \left\{\hat{\psi}_k : \psi^\ell_k \circ F^{\ell-1}_k = \hat{\psi} \in \hat{W}_1, \ k = 1, \ldots, \#\hat{T}_{\ell-1}\right\}, \quad \ell \geq 1.$$

By construction, $\psi^\ell_k$ forms an $L^2(I)$-orthogonal system and $R_\ell = \text{span}(\Psi_\ell)$ in \(8.5\).

In the remainder of this thesis, we confine ourselves to the so-called Haar wavelets in one dimension, i.e. $\hat{\varphi}(x) \equiv 1$ and the mother-wavelet $\hat{\psi}$ is given by

$$\hat{\psi}(x) = \chi_{[0, \frac{1}{2})}(x) - \chi_{[\frac{1}{2}, 1]}(x), \quad \forall x \in \hat{C}.$$

and hence an $L^2(I)$-orthogonal basis $\{\psi^\ell_k\}$ of $Q^0(I, \hat{T}_L)$ is formed by (see Figure 8.1)

$$\hat{\psi}^0_1 \equiv 1, \quad \psi^\ell_k(x) = \hat{\psi}(2^\ell x - |I|2^{-\ell}(k-1)), \quad \ell = 1, \ldots, L, \ k = 1, \ldots, \#\hat{T}_{\ell-1}.$$
MLMC-FVM for random source terms

Figure 8.1: Hierarchical Haar wavelets $\psi_{\ell}^k(x)$ for the first 4 levels.

Figure 8.2: The uncertain measurements $b_{i+1/2}(\omega)$ of the exact bottom topography $b(x)$ at locations $x_{i+1/2}$ are treated as nodal values and then are linearly interpolated on the finest mesh level $\bar{L}$.

8.1.2 Multi-level representation of bottom topography in 1-D

For $d = 1$ (i.e. $D = I_1$), we will assume that uncertain measurements $b_{i+1/2}(\omega) := b(x_{i+1/2}, \omega)$ of the exact bottom topography $b(x)$ are available, as described in (8.1). Then $b_{i+1/2}(\omega)$ are treated as nodal values and are linearly interpolated, see Figure 8.2.

Furthermore, we use the nodal hierarchical “hat” basis interpolation operator $I_{\bar{L}}$,

$$I_{\bar{L}} b(x, \omega) = \sum_{\ell=0}^{\bar{L}} b^\ell(x, \omega), \quad b^\ell := I_{\ell} b - I_{\ell-1} b, \quad I_{-1} \equiv 0,$$

(8.6)

where $I_{\ell}$ denotes linear nodal interpolation operator on the mesh $\bar{T}_\ell$. The finest level $\bar{L} < \infty$ corresponds to the “pixel level” of a terrain imaging device.

Each $b^\ell(x, \omega) \in L^2(\Omega, Q^1(I_1, \bar{T}_\ell))$ is a linear combination of the hierarchical “hat” basis (“Schauder basis”, see Figure 8.3) functions $\varphi_{\ell}^k(x)$, i.e.

$$b^\ell(x, \omega) = \sum_{k=1}^{N_{\ell}} b_{k}^\ell(\omega) \varphi_{\ell}^k(x), \quad b_{k}^\ell \in L^2(\Omega, \mathbb{R}),$$

(8.7)
8.1 Multi-level alias-free representation of random bottom topography

where

\[ \hat{N}_\ell := \dim(Q^1(I_1, \bar{T}_\ell)) - \sum_{\ell'=0}^{\ell-1} \dim(Q^1(I_1, \bar{T}_{\ell'})), \]

i.e. \( \hat{N}_0 = 2 \) and \( \hat{N}_{\ell} = \#\bar{T}_{\ell-1} = 2^{\ell-1}, \forall \ell \geq 1. \)

Figure 8.3: Hierarchical “hat” basis functions \( \varphi^\ell_k(x) \) for the first 4 levels.

The interpolated bottom topography belongs to the space

\[ I_{\bar{L}}b(x, \omega) \in L^2(\Omega, Q^1(I_1, \bar{T}_L)), \]

and the weak spatial derivative of the interpolated topography belongs to the space

\[ \partial_x(I_{\bar{L}}b)(x, \omega) \in L^2(\Omega, Q^0(I_1, \bar{T}_L)). \]

Since \( \partial_x \varphi_k^\ell(x) = \frac{1}{\Delta x^\ell} \psi_k^\ell(x) \), (see Figure 8.4), we have

\[ \partial_x b^\ell(x, \omega) = \frac{1}{\Delta x^\ell} \sum_{k=1}^{N_\ell} b_k^\ell(\omega) \psi_k^\ell(x). \]

Figure 8.4: The derivative of a one-dimensional hierarchical basis function \( \varphi_k^\ell(x) \) is a scaled Haar wavelet \( \frac{1}{\Delta x^\ell} \psi_k^\ell(x) \).

Consequently, \( B := \partial_x(I_{\bar{L}}b) \) admits the Haar wavelet representation,

\[ B(x, \omega) := B^L(x, \omega) := \sum_{\ell=0}^{\bar{L}} \sum_{k=1}^{\#\bar{T}_{\ell-1}} B_k^\ell(\omega) \psi_k^\ell(x), \quad (8.8) \]
MLMC-FVM for random source terms

where the “coefficients” $B^\ell_k(\omega) \in L^2(\Omega, \mathbb{R})$ are random variables given by

$$B^\ell_k(\omega) = \frac{1}{\Delta x_0} \left( b^\ell_0(\omega) + b^\ell_1(\omega) \right), \quad B^\ell_k(\omega) = \frac{1}{\Delta x_\ell} b^\ell_k(\omega), \quad \forall \ell > 0.$$  

Let $h^\ell_i(\omega)$ be cell averages of water level $h(x,\omega)$ above bottom topography at the mesh level $0 \leq \ell \leq L$, i.e. $h^\ell(x,\omega) \in L^2(\Omega, Q^\ell(I_1, T_\ell))$, $h^\ell(x,\omega) := h^\ell_i(\omega)$, $\forall x \in C^\ell_i$. The terms

$$S^\ell_{i+\frac{1}{2}}, \quad \forall \ell \geq 1,$$

needed in one-dimensional analogue of (8.4) are given by integrating over “shifted” cells $C^\ell_{i+\frac{1}{2}} = (x^\ell_i, x^\ell_{i+1})$ of the mesh $T_\ell$,

$$S^\ell_{i+\frac{1}{2}}(\omega) = \int_{C^\ell_{i+\frac{1}{2}}} \bar{h}^\ell_i(\omega) B^\ell_k(\omega) \psi^\ell_k(x) dx, \quad i = 0, \ldots, \#T_\ell. \quad (8.9)$$

The first superscript $\ell$ of $S^\ell_{i+\frac{1}{2}}$ denotes the FVM mesh level. The second superscript $\bar{L}$ denotes “pixel level” in the multi-level topography representation (8.8).

**Lemma 8.1.1.** Assume $d = 1$ and that $B(x,\omega)$ is given in the form (8.8). By $B^L(x,\omega)$ denote the hierarchical representation (8.8) truncated up to level $\bar{L} \leq L$. Then, for all $\bar{L} \geq \bar{L} \geq \ell + 1$, we have

$$S^\ell_{i+\frac{1}{2}}(\omega) = S^\ell_{i+\frac{1}{2}}(\omega), \quad i = 0, \ldots, \#T_\ell. \quad (8.10)$$

Lemma 8.1.1 states that it is sufficient to compute only terms up to level $\ell + 1$ in the hierarchical bottom topography representation (8.6) when the underlying FVM solver is on the mesh level $\ell$. This way, computational work is significantly reduced on the coarsest levels $\ell \ll \bar{L}$.

**Proof.** Notice that due to the vanishing moments of the wavelet basis functions,

$$\int_{C^\ell_{i+\frac{1}{2}}} \psi^\ell_k(x) dx = 0 \quad \text{for} \quad \bar{\ell} > \ell + 1.$$

Since $\bar{h}^\ell_{i+\frac{1}{2}}(\omega)$ is constant and equals $(h^\ell_i(\omega) + h^\ell_{i+1}(\omega))/2$ in each cell $C^\ell_{i+\frac{1}{2}}$,

$$\int_{C^\ell_{i+\frac{1}{2}}} \bar{h}^\ell_{i+\frac{1}{2}}(\omega) B^\ell_k(\omega) \psi^\ell_k(x) dx = B^\ell_k(\omega) \bar{h}^\ell_{i+\frac{1}{2}}(\omega) \int_{C^\ell_{i+\frac{1}{2}}} \psi^\ell_k(x) dx = 0 \quad \text{for} \quad \bar{\ell} > \ell + 1.$$  

The terms for levels $\bar{\ell}$ in the sum (8.8)-(8.9) vanish provided $\bar{\ell} > \ell + 1$, hence these levels ($\bar{\ell} > \ell + 1$) can be disregarded entirely. \hfill $\Box$
8.1 Multi-level alias-free representation of random bottom topography

8.1.3 Multi-level representation of bottom topography in 2-D

For two-dimensional case (i.e. \( d = 2 \), and Cartesian domain \( \mathbf{D} = I_1 \times I_2 \)), we assume that uncertain measurements \( b_{i+\frac{1}{2},j+\frac{1}{2}}(\omega) := b(x_{i+\frac{1}{2}}, y_{j+\frac{1}{2}}, \omega) \) of the exact bottom topography \( b(x, y) \) are available, as in (8.1). Then \( b_{i+\frac{1}{2},j+\frac{1}{2}}(\omega) \) are treated as nodal values and are linearly interpolated in each dimension using the bilinear hierarchical interpolation operator,

\[
\mathcal{I}^L b(x, y, \omega) = \sum_{\ell=0}^{I_\ell} b^\ell(x, y, \omega), \quad b^\ell := \mathcal{I}_{\ell,\ell} b - \mathcal{I}_{\ell-1,\ell-1} b, \quad \mathcal{I}_{-1, -1} \equiv 0, \tag{8.11}
\]

where \( \mathcal{I}_{\ell,\ell} \) denotes bilinear nodal interpolation operator on the mesh \( \mathcal{T}_\ell = \mathcal{T}_1^1 \times \mathcal{T}_1^2 \). The finest level \( L < \infty \) corresponds to the “pixel level” of a terrain imaging device.

In order to construct \( \mathcal{I}^L b \), we consider isotropic tensorization of the hierarchical “hat” ("Schauder") basis functions, i.e. each \( b^\ell(x, \omega) \in L^2(\Omega, Q^1(I_1 \times I_2, \mathcal{T}_\ell)) \) is a linear combination of the multivariate tensor products of such basis functions,

\[
b^\ell(x, \omega) = \sum_{k,k'=1}^{N_\ell} c_{k,k'} b^\ell_k(x) \varphi_{k'}(y) + \sum_{k=1}^{N_\ell} \sum_{k'=1}^{N_{\ell-1}} b^\ell_{k,k'}(\omega) \varphi^:\ell_k(x) \varphi^:\ell_{k'+\frac{1}{2}}(y)
\]

\[
+ \sum_{k=1}^{N_{\ell-1}} \sum_{k'=1}^{N_\ell} b^\ell_{k,k'}(\omega) \varphi^:\ell_{k+\frac{1}{2}}(x) \varphi^:\ell_{k'}(y), \quad c_{k,k'}, b^\ell_{k,k'}, b^\ell_{k,k'} \in L^2(\Omega, \mathbb{R}),
\]

(8.12)

where \( \varphi^:\ell_{k+\frac{1}{2}}(x) := \varphi^\ell_k(x - \frac{1}{2} \Delta x_\ell) \) and \( \varphi^:\ell_{k'+\frac{1}{2}}(y) := \varphi^\ell_{k'}(y - \frac{1}{2} \Delta y_\ell) \).

The interpolated bottom topography belongs to the space

\[
\mathcal{I}^L b(x, \omega) \in L^2(\Omega, Q^1(I_1 \times I_2, \mathcal{T}_L)).
\]

For \( p_1, p_2 \in \mathbb{N}_0 \), define the tensor product of mixed degree polynomial spaces \( Q^{p_1,p_2}(I_1 \times I_2, \mathcal{T}^1 \times \mathcal{T}^2) := Q^{p_1}(I_1, \mathcal{T}^1) \otimes Q^{p_2}(I_2, \mathcal{T}^2) \).

Using this notation, the weak gradient of \( \mathcal{I}^L b \) belongs to the space

\[
\mathbf{B}(x, \omega) := \nabla(\mathcal{I}^L b)(x, \omega) \in L^2 \left[ \Omega, Q^{0,1}(I_1 \times I_2, \mathcal{T}_L) \times Q^{1,0}(I_1 \times I_2, \mathcal{T}_L) \right].
\]

Each component of the weak gradient \( \mathbf{B} := (B_1, B_2)^\top \) is a tensor product of piecewise constant and piecewise linear continuous functions; in particular,

\[
B_1(\cdot, y, \cdot) \in L^2(\Omega, Q^0(I_1, \mathcal{T}_L^1)), \quad B_2(x, \cdot, \cdot) \in L^2(\Omega, Q^0(I_2, \mathcal{T}_L^2)).
\]
MLMC-FVM for random source terms

Hence, $B_1$ admits a one-dimensional Haar wavelet representation in $x$ variable and $B_2$ admits one-dimensional Haar wavelet representation in $y$ variable, i.e.

$$B_1(x, y, \omega) := B_1^L(x, y, \omega) := \sum_{\ell=0}^{L} \sum_{k=1}^{\#\mathcal{T}_\ell} B_{1,k}^\ell(y, \omega) \psi_k^\ell(x),$$

$$B_2(x, y, \omega) := B_2^L(x, y, \omega) := \sum_{\ell=0}^{L} \sum_{k' = 1}^{\#\mathcal{T}_{\ell'}^L} B_{2,k'}^\ell(x, \omega) \psi_{k'}^\ell(y),$$

where the coefficients $B_{1,k}^\ell(y, \cdot), B_{2,k'}^\ell(x, \cdot) \in L^2(\Omega, \mathbb{R})$ depend on $c_{i,j}^\ell, b_{i,j}^\ell, b_{i,j}^\ell$.

Let $h_{i,j}^\ell(\omega)$ be cell averages of water level $h(x, \omega)$ above bottom topography at the mesh level $0 \leq \ell \leq L$, i.e. $h^\ell(x, \omega) \in \mathcal{Q}^\ell(I_1 \times I_2, \mathcal{T}_\ell)$, $h^\ell(x, \omega) := h_{i,j}^\ell(\omega)$, $\forall x \in C_{i,j}^\ell$. The terms $S_{i+\frac{1}{2},j}^{\ell,L}, S_{i,j+\frac{1}{2}}^{\ell,L} \in L^2(\Omega, \mathbb{R}^2)$ in (8.14) are given by integrating over “shifted” cells

$$C_{i+\frac{1}{2},j} = (x_i^\ell, x_{i+1}^\ell) \times (y_j, y_{j+1}), \ C_{i,j+\frac{1}{2}} = (x_i^\ell, x_{i+1}^\ell) \times (y_j^\ell, y_{j+1}^\ell)$$

of $\mathcal{T}_\ell$, i.e.

$$S_{i+\frac{1}{2},j}^{\ell,L}(\omega) = \int_{C_{i+\frac{1}{2},j}} h_{i+\frac{1}{2},j}^\ell(\omega) B_1^\ell(x, \omega) \, dx,$$

$$S_{i,j+\frac{1}{2}}^{\ell,L}(\omega) = \int_{C_{i,j+\frac{1}{2}}} h_{i,j+\frac{1}{2}}^\ell(\omega) B_2^\ell(x, \omega) \, dx.$$  

(8.14)

The first superscript $\ell$ of $S_{i,j}^{\ell,L}$ denotes the FVM mesh level. The second superscript $L$ denotes the “pixel level” in the multi-level topography representations (8.13).

**Lemma 8.1.2.** Assume $d = 2$ and that the weak gradient $\mathbf{B}(x, \omega)$ is given in the form (8.13). By $B_{1}^\ell(x, \omega), B_{2}^\ell(x, \omega)$ denote the hierarchical representations (8.13) truncated up to level $\bar{L} \leq L$. Then, for all $L \leq \bar{L} \leq L + 1$, we have

$$S_{i+\frac{1}{2},j}^{\ell,L}(\omega) = S_{i+\frac{1}{2},j}^{\ell,L}(\omega), \ i = 0, \ldots, \#\mathcal{T}_\ell^1, \ j = 1, \ldots, \#\mathcal{T}_\ell^2,$$

$$S_{i,j+\frac{1}{2}}^{\ell,L}(\omega) = S_{i,j+\frac{1}{2}}^{\ell,L}(\omega), \ i = 1, \ldots, \#\mathcal{T}_\ell^1, \ j = 0, \ldots, \#\mathcal{T}_\ell^2.$$  

(8.15)

Lemma 8.1.2 states that it is sufficient to compute only terms up to level $\ell + 1$ in the hierarchical bottom topography representation (8.13). Hence, the computational work is significantly reduced on the coarsest mesh levels where $\ell \ll L$.

**Proof.** The proof follows the main ideas presented in the proof of Lemma (8.1.1). Since the wavelet basis functions $\psi_k^\ell(x), \psi_{k'}^\ell(y)$ have vanishing moments and since

$$\hat{h}_{i+\frac{1}{2},j}^\ell(\omega), \ \hat{h}_{i,j+\frac{1}{2}}^\ell(\omega)$$

154
are constant in cells $C_{i+\frac{1}{2},j}$, $C_{i,j+\frac{1}{2}}$, respectively, we obtain

$$
\int_{C_{i+\frac{1}{2},j}} \bar{h}_{i+\frac{1}{2},j}^\ell B_{1,k}^\ell (y,\omega) \psi_k^\ell (x) dx = \bar{h}_{i+\frac{1}{2},j}^\ell \int_{C_{i+\frac{1}{2}}} \psi_k^\ell (x) dx \int_{C_{j+\frac{1}{2}}} B_{1,k}^\ell (y,\omega) dy = 0
$$

and, analogously,

$$
\int_{C_{i,j+\frac{1}{2}}} \bar{h}_{i,j+\frac{1}{2}}^\ell B_{2,k'}^\ell (x,\omega) \psi_{k'}^\ell (y) dy = \bar{h}_{i,j+\frac{1}{2}}^\ell \int_{C_{j+\frac{1}{2}}} \psi_{k'}^\ell (y) dy \int_{C_{i+\frac{1}{2}}} B_{2,k'}^\ell (x,\omega) dx = 0
$$

provided $\tilde{\ell} > \ell + 1$. The terms for levels $\tilde{\ell}$ in the sums (8.13) vanish provided $\tilde{\ell} > \ell + 1$, hence these levels can be discarded.

\section{8.2 Shallow water equations}

We tested the proposed MC and MLMC algorithms extensively for the shallow water equations with uncertain bottom topography, in one and two space dimensions. In particular, we are interested in verifying the following claims; i) MC and MLMC algorithms can handle a very large number of sources of uncertainty which are beyond the reach of existing deterministic methods like stochastic Galerkin and stochastic collocation, ii) the MLMC method is considerably superior (faster) than the corresponding MC algorithm, iii) the hierarchical multi-level representation of the bottom topography (see the previous section) speeds up the MLMC algorithm and iv) an (efficient) massively parallel version of the MLMC algorithm can simulate complex and physically relevant shallow water flows with uncertain initial data and bottom topography.

In real applications, random variables $b_{i+\frac{1}{2}} (\omega), b_{i+\frac{1}{2},j+\frac{1}{2}} (\omega)$ would need to be converted to (possibly correlated) hierarchical random variables $b_{k}^\ell (\omega), b_{k,k'}^\ell (\omega)$ using (8.6) or (8.11), respectively. However, in this thesis, we consider a prototype problem by generating a synthetic data set, i.e. we directly choose coefficients $b_{k}^\ell (\omega), b_{k,k'}^\ell (\omega)$ to represent a synthetic data set, which models two bumps and a valley in-between, all with uncertain elevations.

As second-order high-resolution schemes are the basis of the production code [67], we concentrate on the second-order entropy stable schemes of [38]. In particular, we follow the second-order TeCNO implementation suggested in a recent paper [39].
8.2.1 A 1-D dam break problem

The computational domain is $D = [0, 1]$ with Neumann (“transparent”) boundary conditions \((1.104) - (1.105)\). We have an initial dam:

$$\begin{align*}
\{h_0(x, \omega), \ u_0(x, \omega)\} &= \begin{cases} 
0.0 + Y(\omega) - b(x, \omega), & \text{if } x < 1.0, \\
1.5 - b(x, \omega), & \text{if } x > 1.0,
\end{cases}
\end{align*}$$

with $Y(\omega) \sim U(0, \frac{1}{10})$. Thus we have a dambreak problem with uncertain initial dam height. The uncertain bottom topography $b(x, \omega)$ is represented in terms of the hierarchical “Schauder” basis \((8.6) - (8.7)\) with 10 levels (i.e. $L = 9$, $\ell = 0, \ldots, 9$) where the coefficients $b_{\ell}^k(\omega)$ are given by mean values $\mu_{\ell}^k$ which are perturbed by independent uniformly distributed centered random variables, i.e.

$$b_{\ell}^k(\omega) = 0.7 + \mu_{\ell}^k + Y_{\ell}^k(\omega) \sim \frac{1}{5} U(-\varepsilon_{\ell}, \varepsilon_{\ell}), \quad \varepsilon_0 = 0, \quad \varepsilon_{\ell} = \frac{1}{\ell^{1.5}}, \quad \forall \ell \geq 1,$$

where all coefficients $\mu_{\ell}^k$ are zero except (modeling a bump, a valley and a bump)

$$\begin{align*}
\mu_2^3 &= 0.2, \quad \mu_4^4 &= -0.16, \quad \mu_7^{11} &= 0.06.
\end{align*}$$

In \((8.17)\), perturbations $\varepsilon_{\ell}$ were chosen such that the total (from all levels) deviation $\|I_L b - E[I_L b]\|_{L^{\infty}(\Omega, P)}$ is around $5\% - 10\%$ of the mean $E[I_L b]$; we refer to \cite{10, 35}.

A realization of the bottom topography is shown in \textbf{Figure 8.5} and the mean (and variance) of the topography are shown in \textbf{Figure 8.6}. To the best of our knowledge, such high dimensional problems (with $2^9 = 512$ sources of uncertainty) have not been considered in the literature.

A single realization of the reference height is shown in \textbf{Figure 8.5} and the solution mean and variance are shown in \textbf{Figure 8.6}. The solution clearly has a left-moving rarefaction...
8.2 Shallow water equations

Figure 8.6: Mean and standard deviation of the reference solution for the dambreak problem \((8.16)\) computed using MLMC-FVM with ES flux \((1.101)\). Variance is concentrated around the shock.

wave and a right-moving shock wave. Notice that the standard deviation is uniformly distributed only in the interval \([0,0.5]\). There is a significant decrease of the standard deviation at both ends of the rarefaction wave \((x \approx 0.5\) and \(x \approx 0.7)\), and a significant increase around the shock wave \((at x \approx 1.4)\). For the FVM and MLMC-FVM simulations in Figures 8.5 and 8.6, strong stability preserving second order Runge Kutta (SSP-RK2) time stepping scheme was used.

Numerical convergence analysis

Using MLMC-FVM approximation from Figure 8.6 (computed on 9 levels of resolution with the finest resolution being on a mesh of 8192 cells) as a reference solution \(U_{\text{ref}}\), we run MC-FVM and MLMC-FVM on the series of mesh resolutions ranging from 32 cells up to 2048 cells and monitor the convergence behavior. The number of levels for the MLMC-FVM method is chosen so that the coarsest level always contains 8 cells. As all the simulations are performed using the massively parallel ALSVID-UQ \([2]\), the runtime of the parallel algorithm is obtained by measuring the so-called wall clock time, i.e. the total time passed during the simulation. It is accessible as \texttt{MPI\_Wtime()} routine in \texttt{MPI2.0\}. In the convergence plots we use the cumulative wall clock time (obtained by multiplying wall clock time by number of cores); this way the dependence on the used number of cores is reduced allowing for straightforward comparison of the runtimes.

Dashed lines in Figure 8.7 (and all subsequent figures) indicate expected convergence rate slopes expected from the theory for the scalar hyperbolic conservation law (see \((3.19)\) and \((4.7)\)). Assuming validity of analogous convergence results in the case of hyperbolic systems under consideration here, we expect the rates obtained in \([75]\) to coincide with the presently observed convergence rates for systems of balance laws. In this particular
MLMC-FVM for random source terms

Figure 8.7: Convergence of the estimated mean for 1-D dambreak (8.16). Both MLMC2 and MC2 give similar errors for the same spatial resolution. However, there is a significant difference in runtime: MLMC methods are 2 orders of magnitude faster than MC.

In this case we find that they are actually very similar. Our findings coincide with the results published in [75, 77] confirming the robustness of the implementation.

Figure 8.8: Convergence of the estimated variance for 1-D dambreak (8.16). MLMC methods are 2 orders of magnitude faster than MC.

In Figure 8.8, we show convergence plots for variance. Both figures show that MLMC methods are two orders of magnitude faster than MC methods in computing the mean as well as in computing the variance. This numerical experiment clearly illustrates the superiority of the MLMC algorithm over the MC algorithm.

Remark 8.2.1. Since we are using the same MPI parallelization setup for all runs in the convergence analysis, runtimes for small problems (which take only a couple of seconds) are not very representative and should be disregarded (see Figures 8.7 - 8.8).
8.2 Shallow water equations

8.2.2 Random perturbation of lake at rest in 2-D

We consider (1.12) with $d = 2$ in a computational domain $D = [0, 2] \times [0, 2]$ with Neumann (“transparent”) boundary conditions (1.104) - (1.105), and investigate the evolution of an uncertain perturbation of the lake at rest coupled with outflow boundary conditions.

The uncertain bottom topography $b(x, \omega)$ is represented in terms of the nodal, bivariate hierarchical basis (8.11) - (8.12) with random amplitudes. Notice that, formally, this bilinear basis can be obtained by tensorizing the univariate Schauder basis of $C^0([0, 2])$.

Notice also that we used in the present study only isotropically supported product functions. The bottom topography was resolved to 6 levels (i.e. $L = 5$, $\ell = 0, \ldots, 5$) where coefficients $c_{k,k'}^\ell(\omega), \tilde{b}_{k,k'}^\ell(\omega), \tilde{b}_{k,k'}^\ell(\omega)$ are given by mean values $\mu_{k,k',k,k'}^\ell, \hat{\mu}_{k,k',k,k'}^\ell, \hat{\mu}_{k,k,k,k'}^\ell$, respectively. Mean values are then perturbed by independent uniformly distributed centered random variables with decaying variances,

$$
c_{k,k'}^\ell(\omega) = \mu_{k,k'}^\ell + \tilde{Y}_{k,k'}^\ell(\omega) \sim \frac{2}{5} U(-\varepsilon_r, \varepsilon_r),$$

$$
\tilde{b}_{k,k'}^\ell(\omega) = \hat{\mu}_{k,k'}^\ell + \tilde{Y}_{k,k'}^\ell(\omega) \sim \frac{2}{5} U(-\varepsilon_r, \varepsilon_r),$$

$$
\tilde{b}_{k,k'}^\ell(\omega) = \hat{\mu}_{k,k'}^\ell + \tilde{Y}_{k,k'}^\ell(\omega) \sim \frac{2}{5} U(-\varepsilon_r, \varepsilon_r),$$

where all $\mu_{k,k',k,k'}^\ell, \hat{\mu}_{k,k',k,k'}^\ell, \hat{\mu}_{k,k,k,k'}^\ell$ are zero except (modeling two bumps and a valley)

$$
\mu_{2,2}^2 = 0.4, \quad \mu_{6,6}^4 = -0.32, \quad \mu_{11,11}^6 = 0.12, \quad (8.20)
$$

and the magnitudes of the uncertainties are set by $\varepsilon_r = 0$, $\varepsilon_r = 2^{-\ell}$, $\forall \ell \geq 1$. As in subsection 8.2.1, perturbations $\varepsilon_r$ in (8.19) were chosen such that the total (from all levels) deviation $\|I_L b - \mathbb{E}[I_L b]\|_{L^\infty(\Omega, \mathbb{P})}$ is around $5\% - 10\%$ of the mean $\mathbb{E}[I_L b]$.

A realization of the uncertain bottom topography and the corresponding mean and variance are shown in figure 8.9.

Next, we consider the initial data $U_0$ to be a random perturbation of a lake-at-rest. Let $Y \sim \frac{1}{\sigma Y} + \frac{1}{\mu Y} U(-1, 1)$ be a random variable independent of $\{\tilde{Y}_{k,k'}^\ell, \tilde{Y}_{k,k'}^\ell, \tilde{Y}_{k,k,k,k'}^\ell\}$. An initial perturbation around $x_0 = (x_0, y_0) = (1.0, 0.7)$ with a radius $r = \frac{3}{10}$ reads

$$
h_0(x, y, \omega) = \begin{cases} 1.0 + Y(\omega) - b(x, y, \omega) & \text{if } |x - x_0| < r, \\ 1.0 - b(x, y, \omega) & \text{if } |x - x_0| > r, \end{cases} \quad (8.21)
$$

with $b(x, \omega)$ as defined in (8.19) and the initial layer velocities set to zero, i.e.

$$
\{u_0(x, y, \omega), v_0(x, y, \omega)\} = \{0.0, 0.0\}. \quad (8.22)
$$

Note that here we have even more sources of uncertainty $((2^5 - 1)^2 - 1 = 962)$ than in one-dimensional case (8.17).
MLMC-FVM for random source terms

Figure 8.9: Uncertain bottom topography (8.19) with 9 hierarchical levels ($\tilde{L} = 8$), modeling two bumps of uncertain height and a valley of uncertain depth. The total deviation $\|I_{\tilde{L}} b - E[I_{\tilde{L}} b]\|_{L^\infty(\Omega, P)}$ is around 5% - 10% of the mean $E[I_{\tilde{L}} b]$.

Reference solution at time $T = 0.1$, computed with the second-order entropy stable TeCNO scheme [38, 39] and strong stability preserving second order Runge Kutta (SSP-RK2) time stepping, is depicted in Figure 8.10. The results are computed on 9 nested levels of resolution ($L = 8$) with the finest resolution being on a $4096 \times 4096$ mesh and with time steps reduced accordingly in order to maintain the same CFL constant over all discretization levels. The simulation was run on 2044 cores, lasted 2 hours 14 minutes, and 16 samples were taken for the finest mesh.

The above problem is quite involved due to large number of sources of uncertainty as well as the underlying difficulty of simulating small perturbations of steady states. The reference solution show that the wave (in mean) spreads out of the initial source. The variance is distributed in a non-linear and complicated manner with large amount of variance corresponding to the uncertainties in the bottom topography.

Numerical convergence analysis

We investigate the estimated empirical error vs. work in Figure 8.11 and Figure 8.12. Here we use the MLMC-FVM simulation from Figure 8.10 with 9 levels of resolution with the finest resolution being on a $4096 \times 4096$ mesh as the reference solution $U_{\text{ref}}$. Again, in both cases (MC2 and MLMC2), strong stability preserving second order Runge Kutta (SSP-RK2) time stepping scheme was used. The error in the mean field converges at expected rates. At comparable numerical resolution and accuracy, the MLMC2 is about two to three orders of magnitude faster than the MC2 method in delivering mean and variance estimates of comparable numerical accuracy for this problem.
8.2 Shallow water equations

![Image of mean of h+b at t=0.10 and variance of h+b at t=0.10]

<table>
<thead>
<tr>
<th>$L$</th>
<th>$M_L$</th>
<th>grid size</th>
<th>CFL</th>
<th>cores</th>
<th>runtime</th>
<th>efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>16</td>
<td>4096x4096</td>
<td>0.45</td>
<td>2044</td>
<td>2:13:51</td>
<td>97.5%</td>
</tr>
</tbody>
</table>

Figure 8.10: The reference solution for perturbed steady-state (8.21) using MLMC-FVM with ES flux (1.101). Initial perturbation evolves into asymmetric ribbon wave with uncertain amplitude, with large amount of variance corresponding to the uncertainties in the bottom topography.

**Speed up due to hierarchical topology representation**

We test the gain in efficiency due to the (truncated) multi-level hierarchical representation of the uncertain bottom topography (8.11) by comparing with a simulation that uses the classic (full) MLMC algorithm. In other words, the MLMC2 (full) simulation uses the underlying bottom topography (at the resolution of the underlying topography mesh) for all shallow water samples. In particular, simulations at the coarsest level of the FVM mesh use the topography at the finest level of the underlying topography mesh. We compare MLMC2 (full) with MLMC2 (truncated) which uses the representation (8.11) on the perturbations of lake at rest steady state problem in Figure 8.13.

![Image of convergence of the estimated mean in the 2-D simulation (8.21)]

Figure 8.11: Convergence of the estimated mean in the 2-D simulation (8.21). MLMC methods are approximately 3 orders of magnitude faster than MC.
MLMC-FVM for random source terms

As suggested by the theory of section 8.1, the two methods should lead to an identical order of the error for a given space-time resolution. We verify this in Figure 8.13. On the other hand, the MLMC2 (truncated) is at least an order of magnitude faster than the MLMC2 (full) showing that the multi-level representation of the uncertain bottom topography really provides a significant gain in efficiency.

Figure 8.12: Convergence of the estimated variance in the 2-D simulation (8.21). MLMC methods are approximately 2 orders of magnitude faster than MC.

Figure 8.13: Convergence of the estimated mean for 2-D steady-state (8.21) with full \((L = 8)\) and truncated \((\ell + 1)\) number of levels in the hierarchical representation (8.19) of bottom topography. For a given mesh resolution, both estimators coincide, verifying statements in Proposition 8.1.2. The implementation with the truncated number of levels is more than 10 times faster on a mesh of 256 \times 256 cells.
8.3 MLMC approximation of probabilities

8.3.1 MLMC Estimation of Probabilities

So far, in these notes, we addressed the analysis and implementation of MLMC-FVM solvers for the efficient solution of conservation laws with random inputs, and we focused on the efficient computation of first and higher order statistical moments of the random solution(s), such as mean field and variances.

Often, in applications, one is interested in probabilities of certain extremal events \( E \in \mathcal{F} \), rather than statistical moments, conditioned on the given random input data. Denoting by \( \chi_E(\omega) \) the indicator function of \( E \in \mathcal{F} \), the probability of interest is

\[
P(E) = \int_{\omega \in \Omega} \chi_E(\omega) dP(\omega) .
\]  

(8.23)

One of the problems of interest (in order to assess the risk) would be the following: given a fixed sub-domain \( C \subset D \subset \mathbb{R}^d \), for a fixed time \( t \geq 0 \), find the probability \( p(U) \in [0, 1] \) that a certain event \( E \in \mathcal{F} \) will occur. Here, we assume that the event \( E \) takes the generic form \( E = \{ \omega \in \Omega : \chi(U(\cdot, t; \omega)) = 1 \} \), where \( \chi(\cdot) : \mathbb{R}^m \to \{0, 1\} \) is a measurable function. Then, the probability of interest can be expressed as

\[
p(U) := P(\{ \chi_E = 1 \}) = E[\chi(U(\cdot, t, \omega))] .
\]

(8.24)

Rather than developing a general theory for the multi-level MC computation of such probabilities, we exemplify the main ideas for the Shallow Water Equations with uncertain bottom topography, i.e. (1.12). Here, one is often interested in the event “average water level \( \frac{1}{|C|} \int_{x \in C} h(x, t, \omega) \) at time \( t \) in sub-domain \( C \subset D \) (e.g. a neighborhood of the shoreline) exceeds some given threshold \( h_{\text{max}} \)”. Then

\[
\chi_E(U(\cdot, t, \omega)) := \chi(h_{\text{max}}, \infty) \left( \frac{1}{|C|} \int_{x \in C} h(x, t, \omega) dx \right)
\]

and \( P(E) \) is given by

\[
P(E) = E[\chi(U(\cdot, t, \omega))] = \int_{\Omega} \chi(h_{\text{max}}, \infty) \left( \frac{1}{|C|} \int_{x \in C} h(x, t, \omega) dx \right) dP(\omega) .
\]  

(8.25)

The probability of interest \( P(E) \) in (8.25) is an integral w.r.t. the probability measure \( P \). Hence, the integral in (8.25) could be approximated numerically by a Monte-Carlo FVM estimator, i.e. by Monte-Carlo integration of an approximate Finite Volume solution at mesh level \( \ell \), denoted by \( U_\ell \). The single-level Monte-Carlo Finite Volume estimator for \( P(E) \) with \( M \) i.i.d. input data samples based on the FVM on mesh level \( \ell \) is given by

\[
p_M(U_\ell) := \frac{1}{M} \sum_{i=1}^{M} \chi(U_\ell^i(\cdot, t)) = \frac{1}{M} \sum_{i=1}^{M} \chi(h_{\text{max}}, \infty) \left( \frac{1}{|C|} \int_{x \in C} h_\ell^i(x, t) dx \right) ,
\]

(8.26)
MLMC-FVM for random source terms

where \( h_i^\ell (\cdot, t), \ i = 1, \ldots, M \) are the MC samples approximated using the FVM scheme at mesh level \( \ell \). The MLMC-FVM estimator combines MC-FVM estimators (8.26) on the nested family \( \{ T_\ell \}_{\ell \geq 0} \) of FVM meshes, as before, and is given by:

\[
p_L^L (U_L) := p_{M_0} (U_0) + \sum_{\ell = 1}^L \left( p_{M_\ell} (U_\ell) - p_{M_{\ell-1}} (U_{\ell-1}) \right).
\]  

(8.27)

8.3.2 Application to random perturbation of lake at rest in 2-D

We consider the setup as in subsection 8.2.2, i.e. we are interested in the two-dimensional shallow water equations where the uncertain initial perturbation (8.21) of the water surface is propagating over the uncertain bottom topography described in (8.11) - (8.12) and in (8.19). Our aim is to numerically approximate the probabilities as in (8.25) that the cell averaged water level \( h + b \) will exceed the preset threshold \( h_{\text{max}} = 1.002 \) in a subdomain \( C \) where \( C \in T_L \) denotes a Finite Volume cell.

The results of the numerical simulation using the MLMC-FVM estimator (8.27) for the probability integral (8.25) are given in Figure 8.14. There are 9 levels \( (L = 8) \) of FVM mesh resolution with the finest resolution (at the finest level \( \ell = L \)) being 2048 cells. Rusanov flux with second order accurate well-balanced TECNO [38] reconstruction was used.

![Figure 8.14: MLMC-FVM approximation of the probabilities (8.25) that the random water level \( h(\cdot, t, \omega) + b(\cdot, \omega) \) will exceed (at time \( t = 0.1 \)) the preset maximal threshold \( h_{\text{max}} = 1.002 \).](image)

<table>
<thead>
<tr>
<th>( L )</th>
<th>( M_L )</th>
<th>grid size</th>
<th>CFL</th>
<th>cores</th>
<th>runtime</th>
<th>efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>16</td>
<td>2048x2048</td>
<td>0.45</td>
<td>128</td>
<td>1:42:33</td>
<td>94.7%</td>
</tr>
</tbody>
</table>

164
9 MLMC-FVM for nonlinear conservation laws with random fluxes

9.1 Burgers’ equation with uniformly perturbed flux

We consider a random version of Burgers’ equation \([1.2]\) on domain \(D = [0, 2]\) with periodic boundary conditions \([1.102] - [1.103]\) and with the random flux

\[
f(u, \omega) = \frac{u \cdot p(\omega)}{p(\omega)}, \quad p \sim U(1.5, 2.5).
\] (9.1)

It is straightforward to verify that the random flux \(f\) defined above is a bounded random flux with each realization \(f(\omega) \in C^1(\mathbb{R}, \mathbb{R})\) with \(R = \bar{R}\).

We consider deterministic initial data of the form

\[
u_0(x) = \sin(\pi x).
\] (9.2)

Notice that \(\|u_0\|_{L^\infty(D)} = 1\), hence one can choose \(\bar{R} = 1\) in (2.18) of Theorem 2.2.3.

The initial data (9.2) and the reference solution (obtained by MLMC-FVM) at time \(t = 4\) are depicted in Figure 9.1. There are 13 levels \((L = 12)\) of FVM mesh resolution with the finest resolution (at the finest level \(\ell = L\)) being 32768 cells. The Rusanov numerical flux with a second order accurate WENO reconstruction was used. At every point \(x \in [0, 2]\) the solid line represents the mean and the dashed lines represent the mean \(\pm\) standard deviation of the (random) solution. For each sample (realization) of the random flux (9.1), the smooth initial data evolves into (as expected) discontinuity in the physical space and a shock forms at \(x_1 = 1\). Given the fact that the flux function is random, the variance is high over the entire physical domain and is not just concentrated at the discontinuity.

Next, we use this high-resolution MLMC-FVM simulation from Figure 9.1 as the reference solution. We investigate the convergence of error vs. work in Figure 9.2. The error in the mean field converges at expected rates; furthermore, the MLMC2 method is almost two orders of magnitude faster than the MC2 method (for the same numerical accuracy).
MLMC-FVM for nonlinear conservation laws with random fluxes

Figure 9.1: MLMC-FVM solution of Burgers’ equation with uniformly perturbed flux \((9.1)\). As the flux function is random, the variance is high over the entire physical domain and is not concentrated only at the discontinuity.

9.2 Two phase flows in a porous medium with uncertain permeabilities

Many interesting phenomena (such as water flooding) in an oil and gas reservoir can be modeled by using two phase flows in a porous medium. For simplicity, we consider the flow of two phases (oil and water) in a one dimensional reservoir \([5]\). The model reduces to a one dimensional scalar conservation law (with the Buckley-Leverett flux):

\[
S_t + f(S)_x = 0,
\]

\[
f(S) = \frac{qK\lambda_w(S)}{\lambda_w(S) + \lambda_o(S)}.
\]

(9.3)

Here, the variable \(S\) represents the saturation of oil, \(q\) is total flow rate, \(K\) the rock permeability and \(\lambda_w, \lambda_o : [0, 1] \rightarrow \mathbb{R}\) are the relative permeabilities of the water and oil phases, respectively. In practice, the rock permeability needs to be measured and is prone to uncertainty. Similarly, the relative permeabilities are measured in laboratory experiments and are characterized by uncertainty. In this example, we focus on the case of uncertain relative permeabilities, which are frequently taken to be of the form

\(\lambda^o = S^2, \quad \lambda^w(S) = (1 - S)^2\).

We add random perturbations to \(\lambda^o\) and \(\lambda^w\), i.e.

\[
\lambda^o(S) = S^2 + \varepsilon_o Y_o(\omega)S^2(1 - S),
\]

\[
\lambda^w(S) = (1 - S)^2 + \varepsilon_w Y_w(\omega)(1 - S)^2S,
\]

(9.4)

with

\(\varepsilon_o = 0.3, \quad \varepsilon_w = 0.2, \quad Y_o, Y_w \sim \mathcal{U}[-1, 1]\).

Uncertain relative permeabilities defined in (9.4) are depicted in Figure 9.3.
9.2 Two phase flows in a porous medium with uncertain permeabilities

Figure 9.2: Convergence of mean for MLMC-FVM solution of Burgers’ equation with uniformly perturbed flux (9.1). The error in the mean field converges at expected rates and the MLMC2 method is almost two orders of magnitude faster than the MC2 method.

Figure 9.3: The mean and the upper/lower bounds for the uncertain relative oil and water permeabilities as defined in (9.4).

9.2.1 1-D numerical experiment: deterministic initial shock

We set $K = 1.0$ and $q = 1.0$. The initial data is given by a deterministic shock at $x = 1$ in a finite domain $D = [0, 2]$ with Neumann (“transparent”) boundary conditions (1.104) - (1.105),

\[ S_0(x) = \begin{cases} 
0.25 & \text{if } x < 1.0, \\
0.85 & \text{if } x > 1.0.
\end{cases} \quad x \in D. \tag{9.5} \]

Notice that $\|u_0\|_D = 0.85$, hence one can choose $\bar{R} = 0.85$ in (2.18) of Theorem 2.2.3. Furthermore, the random flux defined in (9.3) with random permeabilities (9.4) is a bounded random flux with each realization $f(\omega) \in C^1(-R, R)$ with $R = \bar{R}$.

The initial data (9.5) and the reference solution at time $t = 0.4$ are depicted in Figure 9.4. There are 13 levels ($L = 12$) of FVM mesh resolution with the finest resolution (at
MLMC-FVM for nonlinear conservation laws with random fluxes

the finest level \( \ell = L \) being 32768 cells. At every point \( x \in [0, 2] \), the solid line represents the mean and the dashed lines represent the mean \pm standard deviation of the (random) solution. For each sample of the random permeabilities \( \lambda^o, \lambda^w \), the initial shock splits into a compound shock, consisting of right going rarefaction that is immediately followed by a right moving shock wave. Notice the improvement of the regularity in the stochastic solution: deterministic pathwise solutions for each sample are discontinuous due to formation of the shock; nevertheless, the mean of the solution appears to be continuous. Furthermore, the uncertainty seems to be concentrated on the compound shock in this case.

Figure 9.4: MLMC-FVM solution of Buckley-Leverett equation (9.3) with uncertain permeabilities (9.4) and deterministic initial shock (9.5). Notice the improvement of the regularity in the stochastic solution: deterministic pathwise solutions for each sample are discontinuous due to formation of the shock; nevertheless, the mean of the solution appears to be continuous. Furthermore, the uncertainty seems to be concentrated on the compound shock in this case.

Next, we use the high-resolution MLMC-FVM simulation from Figure 9.4 as the reference solution. We investigate convergence of error vs. work in Figure 9.5 and Figure 9.6. The error in the mean field converges at expected rates. Comparing the MC2 method (for the same numerical accuracy) with the MLMC2, we find that the latter is approximately an order of magnitude faster for the approximation of the mean field and approximately two orders of magnitude faster for the approximation of variance.

9.2.2 Example of the non-monotonicity of the MLMC-FVM estimator

In this section we present two additional examples of MLMC-FVM estimated mean and variance fields with much smaller discretization parameters (number of mesh cells and mesh levels) in order to expose the non-monotonicity of the MLMC-FVM estima-
9.2 Two phase flows in a porous medium with uncertain permeabilities

Figure 9.5: Convergence of mean in Buckley-Leverett equation (9.3) with uncertain permeabilities (9.4). Compared to the MC2 method, the MLMC2 method is approximately an order of magnitude faster for the approximation of the mean field.

Figure 9.6: Convergence of variance in Buckley-Leverett equation (9.3) with uncertain permeabilities (9.4). The MLMC2 method is approximately two orders of magnitude faster than the MC2 method for the approximation of the variance field.
MLMC-FVM for nonlinear conservation laws with random fluxes

tor described in subsection 7.1.2. In particular, we consider the same problem as in subsection 9.2.1 only with substantially smaller space and time step size.

Two cases are considered: mesh size of 128 with 5 levels and mesh size 1024 with 7 levels. The results are presented in Figures 9.7 - 9.8 respectively.

For the first case, where the resolution is coarse, small overshoots in the approximated mean and a significant negative overshoot to the left of the peak in the approximated variance appear. These phenomena are, indeed, discretization artifacts which are due to the MLMC-FVM estimation. For a more detailed explanation, we refer to subsection 7.1.2.

For the second case, where the resolution is fine, the approximations of the mean and the variance have almost converged, i.e. the numerical overshoots have substantially smaller amplitude as compared to results presented in Figure 9.7. In fact, these overshoots are not significant at all and are expected to eventually vanish due to the convergence of the MLMC-FVM method.

![Graphs showing mean and variance of So at t=0.40](image)

**Figure 9.7**: MLMC-FVM solution as in Figure 9.4, with only 128 cells and 5 levels \((L = 4)\). Notice the small overshoot in the approximated mean and a significant negative overshoot to the left of the peak in the approximated variance. Such overshoots are discretization artifacts and are explained by the non-monotonicity of the MLMC-FVM estimator.
9.2 Two phase flows in a porous medium with uncertain permeabilities

Figure 9.8: MLMC-FVM solution as in Figure 9.4 with only 1024 cells and 7 levels ($L = 6$). The approximations of the mean and the variance have almost converged, i.e. the overshoots are much smaller compared to results presented in Figure 9.7. In fact, these overshoots are not significant at all and eventually vanish due to the convergence of the MLMC-FVM method.
MLMC-FVM for nonlinear conservation laws with random fluxes
10 MLMC-FVM for linear systems of conservation laws with random coefficients

Before stating the numerical results, we briefly investigate the hyperbolicity of the wave equation (1.4) as a random linear system of conservation laws (2.25) in order to ensure that the hypothesis (2.26) in Theorem 2.3.1 holds. Furthermore, we derive a condition on the material coefficient $c$ in order to attain finite expected maximum wave propagation speeds, i.e. which ensure that (3.25) holds.

10.1 Hyperbolicity of the random wave equation

We analyze the structure of the matrices $A_r$ in (2.25), which define a strongly hyperbolic linear system of conservation laws under some conditions on the statistical properties of the material coefficient $c$. For brevity of exposition, we consider the one-dimensional case $d = 1$. Then, for each $\omega \in \Omega$ and $x \in D$, there exists an invertible matrix $Q(x, \omega)$,

$$Q(x, \omega) = \frac{1}{\sqrt{2}} \begin{bmatrix} \frac{1}{\sqrt{c(x, \omega)}} & -\frac{1}{\sqrt{c(x, \omega)}} \end{bmatrix},$$

diagonalizing the matrix $A(x, \omega)$ defined in (1.7),

$$Q(x, \omega)A_1(x, \omega)Q(x, \omega)^{-1} = \begin{bmatrix} -\sqrt{c(x, \omega)} & 0 \\ 0 & \sqrt{c(x, \omega)} \end{bmatrix}.$$  

The maximum wave speed $\lambda(\omega)$ is then given by

$$\lambda(\omega) = \| \sqrt{c(\cdot, \omega)} \|_{C(D)}.$$

Since

$$\|Q_x(\omega)\|_{L^\infty(\Omega, C(D))} = \max\{c^{\frac{1}{2}}, c^{-\frac{1}{2}}\} \leq c^{\frac{1}{2}} + c^{-\frac{1}{2}},$$

the uniform boundedness

$$c, c^{-1} \in L^\infty(\Omega, C(D))$$

(10.2)
ensures \( \bar{K}_\infty < \infty \). However, assumption (10.2) is overly strict and can be further relaxed. In particular, for \( 1 \geq k < \infty \),

\[
c^{\frac{1}{2}}, c^{-\frac{1}{2}} \in L^k(\Omega, C(D))
\]  
(10.3)

implies

\[
K^k(\omega) = \| \max\{c^{\frac{1}{2}}(\cdot, \omega), c^{-\frac{1}{2}}(\cdot, \omega)\}\|_{C(D)}^k \leq \| c(\cdot, \omega) \|_{C(D)}^k + \| c^{-1}(\cdot, \omega) \|_{C(D)}^k.
\]

Hence \( c^{\frac{1}{2}}, c^{-\frac{1}{2}} \in L^k(\Omega, C(D)) \) is sufficient to ensure (A1) of Theorem 2.3.1

\[
\bar{K}_k^k = \mathbb{E}[K^k(\omega)] \leq \| c(\cdot, \omega) \|_{L^k(\Omega, C(D))}^k + \| c^{-1}(\cdot, \omega) \|_{L^k(\Omega, C(D))}^k < \infty. 
\]  
(10.4)

Since the non-zero eigenvalues of \( A_r \in \mathbb{R}^{m \times m} \) are \( \pm \sqrt{c}(x, \omega) \), the assumption of finite expected maximum wave speed \( \bar{\lambda} \) in (3.25) holds, provided \( \sqrt{c} \in L^1(\Omega, C(D)) \),

\[
\bar{\lambda} := \mathbb{E}[\lambda] = \| \sqrt{c} \|_{L^1(\Omega, C(D))} \infty, \quad \text{for} \quad \sqrt{c} \in L^1(\Omega, C(D)).
\]  
(10.5)

Finally, assumption (A2) in the Theorem 2.3.1 holds with \( r_A = r_c \), provided

\[
c \in L^0(\Omega, C^{r_c}(D)).
\]

### 10.2 Acoustic wave equation in 1-D

For this section (i.e., one dimensional simulations), we assume that the random wave speed \( c \) is given by its Karhunen-Loève (KL) expansion

\[
\log c(x, \omega) = \log \bar{c}(x) + \sum_{m=1}^{\infty} \sqrt{\lambda_m} \Psi_m(x) Y_m(\omega),
\]  
(10.6)

with eigenvalues \( \{\lambda_m\}_{m=1}^\infty \) satisfying \( \{\sqrt{\lambda_m}\}_{m=1}^\infty \in \ell^4(\mathbb{N}) \), eigenfunctions \( \Psi_m \) satisfying \( \|\Psi_m\|_{L^2(D)} = 1 \), deterministic part \( \bar{c} \in L^2(D) \), and independent random variables \( Y_m \) with zero mean and finite variance.

#### 10.2.1 Smooth wave with uniformly distributed random coefficients

For physical domain \( D = [0, 2] \) with periodic boundary conditions (1.102) - (1.103), consider deterministic, smooth \((r_0 = \infty)\) initial data

\[
p_0(x, \omega) := \sin(\pi x), \quad p_1(x, \omega) \equiv 0,
\]  
(10.7)

and a random coefficient \( c(x, \omega) \) that is given in terms of its KL expansion (10.6) with independent uniformly distributed \( Y_m \sim \mathcal{U}[-1, 1] \). We choose eigenvalues \( \lambda_m = m^{-2.5} \), eigenfunctions \( \Psi_m(x) = \sin(\pi mx) \) and \( \bar{c}(x) \equiv 0.1 \). Then both \( c \) and \( c^{-1} \) are uniformly bounded in \( \Omega \): \( c(x, \omega), c^{-1}(x, \omega) \in L^\infty(\Omega, C(D)) \). Hence (10.4) and (10.5) holds with
10.2 Acoustic wave equation in 1-D

any \( k \in \mathbb{N}_0 \cup \{\infty\} \). For simulations, the KL expansion (10.6) is truncated after the first 10 terms: \( \lambda_m = 0, \forall m > 10 \). Since \( r_0 = \infty, r_c \geq 1 \), by Theorem 1.3.12 the solution has bounded weak derivatives of first order \( \mathbb{P}\)-a.s., i.e.

\[ U(\cdot, \cdot, \omega) \in W^{r,\infty}(D) \] with \( \bar{r} = 1 \). A first order accurate FVM scheme \((q = 1, \text{HLL Rusanov flux} [66], \text{FE time stepping})\) will be used, hence, in (3.18), \( s = \min\{1, \max\{1/2, 1\}\} = 1 \). Higher order schemes \((s > 1)\) for the case \( d = 1 \) are inefficient since \( s/(d+1) > 1/2 \) in (4.28). Results of the MLMC-FVM simulation at \( t = 2.0 \) are presented in Fig. 10.1.

![Figure 10.1: One sample (left) and mean and variance (right) of the acoustic pressure \( p(x, \omega) \) as in (1.4). We observe very complex distribution of the variance, resulting from highly heterogeneous random material coefficient \( c(x, \omega) \).](image)

Using the MLMC-FVM approximation from Fig. 10.1 (computed on 12 levels of resolution with the finest resolution having 16384 cells) as a reference solution \( U_{\text{ref}} \), we run MC-FVM and MLMC-FVM (with \( \Delta x_0 = 1/4 \)) on a series of mesh resolutions from 32 cells up to 1024 cells and monitor the convergence behavior.

In Figure 10.2 we compare the MC-FVM scheme with \( M = O(\Delta x^{-2s}) \) and the MLMC-FVM scheme with \( M_L = M_L 2^{2s(d-1)} \), where \( M_L = 16 \) is chosen as suggested in [75]. Dashed lines indicate convergence rate slopes proved in Theorems 3.3.2 and 4.2.2. Theoretical and numerically observed convergence rates coincide, confirming the robustness of our implementation. The MLMC method is observed to be three orders of magnitude faster than the MC method. This numerical experiment clearly illustrates the superiority of the MLMC algorithm over the MC algorithm (for \( q = 1, s = 1 \)).

In Fig. 10.3 we show convergence plots for variance. Both figures show that MLMC methods are two to three orders of magnitude faster than MC methods in computing the mean as well as in computing the variance. This numerical experiment clearly illustrates the superiority of the MLMC algorithm over the MC algorithm (for \( q = 1, s = 1 \)).
MLMC-FVM for linear systems of conservation laws with random coefficients

Figure 10.2: Convergence of the estimated mean for (10.7). Both MLMC and MC give similar errors for the same spatial resolution. However, the MLMC method is 3 orders of magnitude faster than MC.

Figure 10.3: Convergence of the estimated variance for (10.7). The MLMC method is 2 orders of magnitude faster than MC.

10.2.2 Discontinuous wave with normally distributed random coefficients

For the domain $D = [0, 2]$ with periodic boundary conditions (1.102) - (1.103), consider deterministic, discontinuous ($r_0 = 0$) initial data

\[ p_0(x, \omega) := 2\chi_{(0.5,1.5)}(x) - 1.0, \quad p_1(x, \omega) \equiv 0. \tag{10.8} \]

and a stochastic coefficient $c(x, \omega)$ that is given by a KL expansion (10.6) with independent normally distributed $Y_m \sim N[0, 1]$. We choose eigenvalues $\lambda_m = m^{-2.5}$, eigenfunctions $\Psi_m(x) = \sin(\pi mx)$ and the mean field $\bar{c}(x) \equiv 0.1$. Then, unlike in the uniform case before, $c, c^{-1} \notin L^\infty(\Omega, C(D))$, i.e. there is positive probability that $\lambda(\omega)$ attains any arbitrarily large or arbitrarily small value.

However, (10.4) and (10.5) hold by

**Proposition 10.2.1.** Assume $\{\sqrt{\lambda_m}\} \in \ell^1(\mathbb{N})$. Then $c, c^{-1} \in L^k(\Omega, C(D))$, $\forall k \in \mathbb{N} \cup \{0\}$.
10.2 Acoustic wave equation in 1-D

Proof. Using the triangle inequality and \( \| \Psi_m \|_{L^\infty(D)} = 1 \), we obtain the following bound,

\[
\frac{\| c(\cdot, \omega) \|_{C(D)}}{\| c \|_{C(D)}} \leq \exp \left( \sum_{m=1}^{\infty} \sqrt{\lambda_m} |Y_m(\omega)| \right) =: \tilde{c}(\omega).
\]

Next, we bound \( \mathbb{E}[\tilde{c}(\omega)^k] \). Since \( Y_m \) are independent and normally distributed,

\[
\mathbb{E}[\tilde{c}(\omega)^k] = \prod_{m=1}^{\infty} \mathbb{E} \left[ \exp \left( k \sqrt{\lambda_m} |Y_m(\omega)| \right) \right]
= \prod_{m=1}^{\infty} \exp \left( \frac{k^2 \lambda_m}{2} \right) \left( 1 + \text{erf} \left( \frac{k \sqrt{\lambda_m}}{\sqrt{2}} \right) \right),
\]

where the error function is defined by \( \text{erf}(a) = \frac{2}{\sqrt{\pi}} \int_0^a \exp(-t^2) \, dt \).

Using the inequalities \( \text{erf}(a) \leq \frac{2}{\sqrt{\pi}} a \) and \( 1 + a \leq \exp(a) \), for any real \( a \geq 0 \),

\[
\prod_{m=1}^{\infty} \left( 1 + \text{erf} \left( \frac{k \sqrt{\lambda_m}}{\sqrt{2}} \right) \right) \leq \prod_{m=1}^{\infty} \left( 1 + \frac{2}{\sqrt{\pi}} \frac{k \sqrt{\lambda_m}}{\sqrt{2}} \right) \leq \exp \left( \sum_{m=1}^{\infty} \frac{2}{\sqrt{\pi}} \frac{k \sqrt{\lambda_m}}{\sqrt{2}} \right).
\]

Hence, \( \| c \|_{L^k(\Omega, C(D))} \leq \| \tilde{c} \|_{C(D)} \mathbb{E}[\tilde{c}(\omega)^k]^{\frac{1}{k}} \) is bounded further using

\[
\mathbb{E}[\tilde{c}(\omega)^k]^{\frac{1}{k}} \leq \exp \left( \frac{k}{2} \| \lambda_m \|_{\ell^1(\mathbb{N})} + \sqrt{\frac{2}{\pi}} \left\| \sqrt{\lambda_m} \right\|_{\ell^1(\mathbb{N})} \right) < \infty.
\]

The proof of \( c^{-1} \in L^k(\Omega, L^\infty(D)) \) is analogous due to symmetry of \( Y_m \). \qed

Since \( r_0 = 0 \), by Theorem 13.12, the solution \( U(\omega) \in W^{r, \infty}(D) \) is \( \mathbb{P} \)-a.s. discontinuous (\( \bar{r} = 0 \)). First order accurate \((q_1 = 1, \text{HLL Rusanov flux} \ [66], \text{Forward Euler time stepping})\) and second order accurate \((q_2 = 2, \text{HLL Rusanov flux, WENO reconstruction, SSP-RK2 time stepping} \ [66])\) FVM schemes will be used; hence, in (3.18), \( s_1 = 1/2 \) and \( s_2 = 1 \). For simulations, the KL expansion is truncated after the first 10 terms: \( \lambda_m = 0, \forall m > 10 \). Results of the MLMC-FVM simulation at \( t = 2.0 \) are presented in Fig. 10.5.
MLMC-FVM for linear systems of conservation laws with random coefficients

Figure 10.4: One sample (left) and mean and variance (right) of the coefficient $c(x, \omega)$ as in (10.6) with truncation after the first 10 terms.

Figure 10.5: One sample (left) and mean and variance (right) of the acoustic pressure $p(x, \omega)$ as in (1.5). We observe the improved regularity (discontinuities are absent) of the mean acoustic pressure field and a very complex distribution of the variance, resulting from highly heterogeneous random material coefficient $c(x, \omega)$.

<table>
<thead>
<tr>
<th>$L$</th>
<th>$M_L$</th>
<th>grid size</th>
<th>CFL</th>
<th>cores</th>
<th>runtime</th>
<th>efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>16</td>
<td>16384</td>
<td>0.475</td>
<td>192</td>
<td>0:17:54</td>
<td>93.8%</td>
</tr>
</tbody>
</table>

The MLMC-FVM approximation from Fig. 10.5 (computed on 12 levels of resolution with the finest resolution being on a mesh of 16384 cells) is used as a reference solution $U_{\text{ref}}$. Additionally to MC and MLMC schemes with $s = s_1$, we consider MC2 and MLMC2 schemes with $s = s_2$. In Figures 10.6-10.7, we show convergence plots for the mean and the variance, respectively. MLMC methods appear to be two orders of magnitude faster than MC methods in approximating the variance, however, no significant speed-up is observed in the approximation of the mean, possibly due to a pre-asymptotic
10.3 Log-normally distributed random material coefficients

In most applications of practical interest, many terms are needed in the Karhunen-Loève expansion to approximate the random material coefficient well. The computation of so many Karhunen-Loève terms is very costly, especially on the coarsest mesh levels of MLMC-FVM method.

Hence, we propose an alternative method which allows the generation of random material coefficient samples with almost optimal (up to logarithmic terms) computational complexity on all MLMC mesh levels, including the coarsest ones.

Figure 10.6: Convergence of the estimated mean for (10.8). Both MLMC(2) and MC(2) give similar errors for the same spatial resolution. Furthermore, there is no significant difference in runtime: MLMC methods are only slightly faster than MC methods (possibly due to a pre-asymptotic effect).

Figure 10.7: Convergence of the estimated variance for (10.8). Both MLMC(2) and MC(2) give similar errors for the same spatial resolution. However, MLMC methods are 2 orders of magnitude faster.
10.3.1 Spectral generator using Fast Fourier Transform

In the following, the spectral generator to compute log-normally distributed random field realizations based on \[83, 17, 86, 94\] is described. We will assume that the distribution of the random material coefficient is completely determined by the positive definite covariance operator of the random field \(c(x, \omega)\). Furthermore, we will assume that the covariance is stationary, i.e. that the covariance of the (random) values at two given points \(x, y \in D\) in a periodic domain \(D\) depends only on the (component-wise) distance vector \(\tau = |y - x|\) between these two points,

\[
\text{Cov}(\log c(x, \omega), \log c(y, \omega)) := k(\tau), \quad \tau \in D = [0, p_1) \times \cdots \times [0, p_d), \tag{10.9}
\]

where \(k : \mathbb{R}^d \rightarrow \mathbb{R}_+\) is called an anisotropic covariance kernel.

For a given mesh \(T\), define a multi-dimensional array of cell mid-points \(x^{(i_1, \ldots, i_d)}\),

\[
X \in \mathbb{R}^{#T_1} \times \cdots \times \mathbb{R}^{#T_d}, \quad X_{i_1, \ldots, i_d} = x^{(i_1, \ldots, i_d)}, \quad i_r = 1, \ldots, #T_r. \tag{10.10}
\]

We are interested in approximating the values of one realization of the random material coefficient \(c(x, \omega)\) at the mesh discretization points \(x^{(i_1, \ldots, i_d)}\). We define an analogous multi-dimensional array for these values:

\[
c \in \mathbb{R}^{#T_1} \times \cdots \times \mathbb{R}^{#T_d}, \quad c_{i_1, \ldots, i_d} = c(x^{(i_1, \ldots, i_d)}, \omega), \quad i_r = 1, \ldots, #T_r. \tag{10.11}
\]

Define the covariance matrix, corresponding to the discrete version of the symmetric positive definite covariance operator, to be

\[
C \in \left(\mathbb{R}^{#T_1} \times \cdots \times \mathbb{R}^{#T_d}\right)^2, \quad C_{i_1, \ldots, i_d; j_1, \ldots, j_d} = \text{Cov}(\log c_{i_1, \ldots, i_d}; \log c_{j_1, \ldots, j_d}). \tag{10.12}
\]

By \[93, \text{section 4.1}\], the covariance matrix \(C\) is symmetric positive definite, hence the following decomposition of \(C\) exists,

\[
L \in \left(\mathbb{R}^{#T_1} \times \cdots \times \mathbb{R}^{#T_d}\right)^2, \quad \text{such that} \quad C = LL^T. \tag{10.13}
\]

For instance, one choice for a matrix \(L\) could be the square root matrix of the matrix \(C\); as \(C\) is symmetric positive definite, \(L\) would also be symmetric and hence \(L = L^T\) would hold. Another possible choice for a matrix \(L\) could be the Cholesky factor \(L\) of the corresponding Cholesky decomposition \(C = LL^T\).

Using the decomposition \[10.13\], a realization of the random material coefficient \(c\) can be obtained by the following steps:

1. generate a Gaussian vector with i.i.d. Gaussian (standard normal) random variables as its components,

\[
g \in \mathbb{R}^{#T_1} \times \cdots \times \mathbb{R}^{#T_d}, \quad g_{i_1, \ldots, i_d} \sim \mathcal{N}(0, 1), \tag{10.14}
\]

180
10.3 Log-normally distributed random material coefficients

2. given a Gaussian vector $\mathbf{g}$, compute a realization of the correlated coefficient $\mathbf{c}$ by

$$\mathbf{c} = \exp(\mathbf{Lg}), \quad (10.15)$$

where the exponential function of a vector is applied component-wise.

Then, the logarithm of the realizations of the vector $\mathbf{c}$, generated using the above methodology, have the desired symmetric positive definite covariance matrix $\mathbf{C}$,

$$\text{Cov}(\log \mathbf{c}) = \mathbb{E}[\log \mathbf{c} \log \mathbf{c}^\top] = \mathbf{L} \mathbb{E}[\mathbf{g} \mathbf{g}^\top] \mathbf{L}^\top = \mathbf{LL}^\top = \mathbf{C}. \quad (10.16)$$

Notice that the covariance $\mathbf{C}$ is much larger than the required material coefficient array $\mathbf{c}$, hence, without further improvements, such an algorithm would be computationally extremely expensive. However, as the covariance matrix is determined by a stationary covariance kernel $k(\tau)$ on a periodic domain $\mathbf{D}$, see (10.9), the resulting covariance matrix $\mathbf{C}$ is circulant. Hence, the “matrix-vector” multiplication in (10.15) is actually a multi-dimensional “vector-vector” convolution \cite{83, 17, 86, 94} which can be performed much faster using the Fast Fourier Transform (FFT). The algorithm has three main steps:

**Step 1:** $d$-dimensional Fourier transforms of the evaluated kernel

$$\mathbf{k} \in \mathbb{R}^{T_1 \times \cdots \times T_d}, \quad \mathbf{k}_{i_1, \ldots, i_d} = k(|\mathbf{x}_{(i_1, \ldots, i_d)} - \mathbf{x}^{(1, \ldots, 1)}|), \quad (10.16)$$

and a random Gaussian vector $\mathbf{g}$ with i.i.d. entries are needed:

$$\hat{\mathbf{k}} = \mathcal{F}\mathbf{k} \in \mathbb{R}^{T_1 \times \cdots \times T_d}, \quad \mathbf{\hat{g}} = \mathcal{F}\mathbf{g} \in \mathbb{C}^{T_1 \times \cdots \times T_d}. \quad (10.17)$$

Since $\mathbf{k}$ is real and periodic, $\hat{\mathbf{k}}$ is also real and periodic (in each dimension). Moreover, the Fourier basis diagonalizes the circulant positive semi-definite matrix $\mathbf{C}$, hence $\hat{\mathbf{k}}$ is the vector of the eigenvalues of $\mathbf{C}$, i.e. all entries in $\hat{\mathbf{k}}$ are non-negative.

**Step 2:** Computing the square root matrix $\mathbf{L}$ as in (10.13) (with $\mathbf{L} = \mathbf{L}^\top$) corresponds to taking the element-wise square root $\hat{\mathbf{1}}$ of $\hat{\mathbf{k}}$:

$$\hat{\mathbf{1}} \in \mathbb{R}^{T_1 \times \cdots \times T_d}, \quad \hat{1}_{i_1, \ldots, i_d} = \sqrt{\hat{k}_{i_1, \ldots, i_d}}. \quad (10.18)$$

**Step 3:** “matrix-vector” multiplication in (10.15) is equivalent to multiplying $\hat{\mathbf{1}}$ and $\mathbf{\hat{g}}$ element-wise and performing $d$-dimensional inverse Fourier transform $\mathcal{F}^{-1}$:

$$\mathbf{c} = \exp(\mathcal{F}^{-1}(\hat{\mathbf{1}} \mathbf{g})) \in \mathbb{R}^{T_1 \times \cdots \times T_d}, \quad (10.19)$$

where the vector-vector multiplication and the exponential are applied element-wise.

For the implementation of this generator, $d$-dimensional Fourier transforms $\mathcal{F}$ and $\mathcal{F}^{-1}$ were computed using the parallel version of the Fast Fourier Transform library FFTW \cite{125}. 181
MLMC-FVM for linear systems of conservation laws with random coefficients

We remark that \( \hat{k} \) and \( \hat{1} \) in steps 1 and 2 need to be computed only once for a given mesh \( \mathcal{T} \), whereas \( \hat{g} \) in step 1 and the resulting sample \( c \) in step 3 need to be computed for each realization of random coefficient \( c \). We would also like to note that none of the above mentioned Fourier transforms are available in complex arithmetic, i.e. \( C \to \mathbb{C} \). In particular, since \( k \) is real and even, the transform \( k \to \hat{k} \) is \( \mathbb{R} \to \mathbb{R} \) with \( \hat{k} \) even; the transform \( g \to \hat{g} \) is \( \mathbb{R} \to \mathbb{C} \) with \( \hat{g} \) even; the final transform \( \hat{lg} \to \mathcal{F}^{-1}(l\hat{g}) \) is \( \mathbb{C} \to \mathbb{R} \), since both \( l \) and \( \hat{g} \) are even, resulting in even \( l\hat{g} \). Performing such transforms with a standard \( C \to \mathbb{C} \) FFT is inefficient (though possible). In the numerical experiments reported ahead, the appropriate (hardware) optimized \( \mathbb{R} \to \mathbb{R}, \mathbb{R} \to \mathbb{C} \) and \( \mathbb{C} \to \mathbb{R} \) FFT transforms from the FFTW library \cite{125} were used instead.

### 10.3.2 Covariance upscaling

The MLMC-FVM algorithm requires MC estimates of the differences \( U_\ell - U_{\ell-1} \), i.e. solutions obtained on two consecutive mesh levels,

\[
E_{M_\ell}[U_\ell - U_{\ell-1}].
\]  
(10.20)

Notice that the same realization of the random material coefficient \( c(\cdot, \omega) \) is required on different mesh resolutions, \( \ell \) and \( \ell - 1 \). The required coupling of the MLMC methods with the generation of log-normal fields using spectral FFT method is not straight-forward.

One idea was recently introduced in [83], where the coupling is achieved by drawing a realization \( c_{\ell} \in \mathbb{R}^{\mathcal{T}_\ell} \) of \( c(\cdot, \omega) \) on the finer mesh \( \mathcal{T}_\ell \), and then computing the averages of \( c_{\ell} \) on the coarser mesh to obtain the upscaled realization \( \bar{c}_{\ell-1} \in \mathbb{R}^{\mathcal{T}_{\ell-1}} \). One must, however, ensure that the distribution of such “averaged” discrete random fields \( \bar{c}_{\ell-1}(\omega) \) coincides with the distribution of the “non-averaged” discrete random fields \( c_{\ell-1}(\omega) \). In order to achieve this and at the same time maintain the efficiency of the method, the covariance \( C_{\ell} \in \mathbb{R}^{\mathcal{T}_{\ell} \times \mathcal{T}_{\ell}} \) from level \( \ell \) also needs to be upscaled to the coarser mesh resolution \( \ell - 1 \). In the context of stationary covariance (as described in subsection 10.3.1), with given kernel, only the kernel needs to be upscaled from \( k_{\ell} \in \mathbb{R}^{\mathcal{T}_{\ell} \times \mathcal{T}_{\ell}} \) to \( k_{\ell-1} \in \mathbb{R}^{\mathcal{T}_{\ell-1} \times \mathcal{T}_{\ell-1}} \). However, in order to obtain the upscaled covariance kernel \( k_{\ell} \) (and hence also \( C_{\ell} \)) for any arbitrary level \( 0 \leq \ell < L \), a recursive computation needs to be performed using the covariance kernel \( k_{\ell} \) on the finest mesh resolution. Such overhead is very inefficient, for instance, in parallel applications, where only samples from several coarsest meshes need to be computed on the same core. Moreover, since the domain decomposition method is not used for samples on the coarsest mesh resolutions, the amount of memory available might be significantly smaller than required for the computation of \( k_{\ell} \).

Here we present a different coupling strategy, which directly uses the spectrum \( k \) of the kernel \( k \). Let \( g_{\ell-1}(\omega) \) be defined as the scaled multi-dimensional average of the vector \( g_{\ell}(\omega) \),

\[
g_{\ell-1}^{\ell}(\omega) = \frac{1}{\sqrt{2^d}} \sum_{j_r \in \{0,1\}} g_{2i_1+j_1,...,2i_d+j_d}(\omega), \quad i_r = 1, \ldots, \#\mathcal{T}_{\ell-1}. \]  
(10.21)
10.3 Log-normally distributed random material coefficients

The upscaled realization $\mathbf{c}^{\ell - 1}(\omega)$ of $\mathbf{c}^{\ell}$ is defined according to (10.19), using the averaged $\mathbf{g}^{\ell - 1}$,

$$
\bar{c}^{\ell - 1}(\omega) = \exp(\mathcal{F}^{-1}i^{\ell - 1}\mathcal{F}\mathbf{g}^{\ell - 1}).
$$

(10.22)

Since the (scaled) average of Gaussian i.i.d. random variables is again the same Gaussian random variable, the upscaled coefficient $\bar{c}^{\ell - 1}(\omega)$ is log-Gaussian with the same covariance kernel $k$. Note, that in order to have a standard normal distribution for $\bar{g}^{\ell - 1}$, the averages are multiplied by an additional scaling factor of $\sqrt{2}d$. Moreover, the Fourier transform $\mathcal{F}\bar{g}^{\ell - 1}$ of the averaged $\mathbf{g}^{\ell}$ will be an approximation of $\mathcal{F}\mathbf{g}^{\ell}$, since the averaging (10.21) corresponds to the low-pass filtering of the vector $\mathbf{g}^{\ell}$.

In comparison to the upscaling strategy presented in [83], the above method operates only on the mesh resolution that is one level finer, and not on every level with finer mesh resolution.

10.3.3 Anisotropic exponential covariance kernel for periodic boundary conditions

Given variance $\sigma^2 > 0$ and correlation lengths $\eta_1, \ldots, \eta_d > 0$, the standard exponential covariance kernel is given by

$$
k(\tau) = \sigma^2 \exp \left( -\sum_{1}^{d} \frac{\tau_r^2}{\eta_r^2} \right), \quad \tau = \mathbf{D}. \tag{10.23}
$$

Since the domain $\mathbf{D} = [0, p_1] \times \cdots \times [0, p_d]$ is assumed to be periodic, the resulting kernel $k(\tau)$ is not smooth at hyper-planes $\tau_r = p_r/2$, $r = 1, \ldots, d$. The exponential covariance kernel $k(\tau(x))$ with $p_1 = 2$ and $\eta_1 = \frac{1}{2}$ is shown in Figure 10.8. It was observed in [83], that for long correlation lengths $\eta_r$, the resulting kernel in (10.23) defines a covariance matrix $\mathbf{C}$ that is not positive semi-definite [83]. Hence, following [93, section 4.2.3], we define a periodic exponential covariance kernel $k_P(\tau)$ by superimposing kernel (10.23) and a smooth periodic sine function,

$$
k_P(\tau) = \sigma^2 \exp \left( -\sum_{1}^{d} \frac{(p_1 \sin(\pi/p_1 \tau_1))^2}{\eta_1^2} + \cdots + \frac{(p_d \sin(\pi/p_d \tau_d))^2}{\eta_d^2} \right), \quad \tau = \mathbf{D}. \tag{10.24}
$$

The derivation of $k_P(\tau)$ in (10.24) (as a generalization of derivation in [93, section 4.2.3]) is as follows. Define an auxiliary kernel $\bar{k}(\tau)$, obtained by tensorizing the kernel $k(\tau)$ in (10.23), resulting in a kernel defined on the $2d$-dimensional space $\mathbf{D} \times \mathbf{D}$,

$$
\bar{k}(\tilde{\tau}) = \sigma^2 \exp \left( -\sum_{1}^{d} \frac{\tau_{1r}^2}{\eta_r^2} + \cdots + \frac{\tau_{2d}^2}{\eta_{2d}^2} \right), \quad \tilde{\tau} = \mathbf{D} \times \mathbf{D}, \tag{10.25}
$$

where the correlation lengths $\eta_1, \ldots, \eta_d$ are extended to the indices $1, \ldots, 2d$ by

$$
\eta_{d+1} = \eta_1, \ldots, \eta_{2d} = \eta_d.
$$
MLMC-FVM for linear systems of conservation laws with random coefficients

As the kernel \((\tau)\) in (10.23) defines a positive definite covariance matrix, so does the tensorized kernel \(\bar{k}(\bar{\tau})\), but only on a tensorized domain \(D \times D\). Then, the kernel \(k_P(\tau)\) in (10.24) is obtained by restricting \(\bar{k}(\bar{\tau})\) to the \(d\)-dimensional sphere in \(D \times D\), parametrized as follows,

\[
\bar{x}_r = \frac{1}{2} p_r \cdot \begin{cases} 
\cos(2\pi/p_r x_r) & \text{if } r = 1, \ldots, d, \\
\sin(2\pi/p_{r-d} x_{r-d}) & \text{if } r = d, \ldots, 2d.
\end{cases} \quad x \in D, \quad \bar{x} \in D \times D. \quad (10.26)
\]

Using the following trigonometrical identity for each component \(x_j\) of \(\tau = x - x' \in D\), \(j = 1, \ldots, d\),

\[
\frac{1}{4} \left( \cos(2x_j) - \cos(2x'_j) \right)^2 + \frac{1}{4} \left( \sin(2x_j) - \sin(2x'_j) \right)^2 = \sin^2(x_j - x'_j),
\]

the parametrization (10.26) of the tensorized kernel (10.25) gives the periodic kernel (10.24). Since \(\bar{k}(\bar{\tau})\) defines a positive definite covariance matrix, so does its restriction \(k_P(\tau)\).

Notice that the periodic exponential covariance kernel \(k_P(\tau)\) depends explicitly on the domain geometry (periods \(p_1, \ldots, p_d\)) and is smooth with minimum at the center of \(D\). An example of \(k_P(\tau(x))\) for \(d = 1\) with period \(p_1 = 2\) and correlation length \(\eta_1 = \frac{1}{2}\) is shown in Figure 10.8.

![Figure 10.8](image-url)

**Figure 10.8:** Conventional (10.23) and periodic (10.24) covariance kernels \(k(\tau(x))\) and \(k_P(\tau(x))\) for \(d = 1\) with period \(p_1 = 2\) and correlation length \(\eta_1 = \frac{1}{2}\). The latter is smooth at \(x = p_1/2\), the former is not.
10.4 Acoustic wave propagation in random heterogeneous layered medium

10.4.1 Two dimensional case

The physical domain is $D = [0, 2] \times [0, 2]$ with $d = 2$. We assume a stationary periodic covariance kernel \(^{(10.24)}\) with variance $\sigma^2 = 0.2$ and correlation lengths $\eta_1 = 2.0, \eta_2 = 0.1$ in \(^{(10.24)}\). The initial data is chosen to be deterministic and set to zero, i.e. $p_0(x, \omega) \equiv 0, u_0(x, \omega) \equiv 0$. Identical periodic (in time) acoustic pressure pulses are injected into two locations of the domain through the deterministic source term $f$,

$$
f(x, t, \omega) = \begin{cases} A_p \left( \exp \left( -\frac{\|x-x^c_1\|^2}{2\sigma^2} \right) + \exp \left( -\frac{\|x-x^c_2\|^2}{2\sigma^2} \right) \right) & \text{if } \{3t\} < 0.02, \\
0 & \text{else.} \end{cases} \tag{10.27}$$

where $A_p = 5000$, $x^c_1 = (0.5, 2.0)$, $x^c_2 = (1.5, 2.0)$, $\sigma^2 = 0.04$ and $\{ \cdot \}$ denotes the fractional part. Perfectly reflecting boundary conditions \(^{(1.106)} - (1.107)\) are assumed at the top and the bottom, and the periodic boundary conditions \(^{(1.102)} - (1.103)\) are assumed at the sides of the domain.

Results of the deterministic FVM simulation up to $t = 1.0$ are presented in Fig. \(^{10.9}\), where the approximated sample of the random material coefficient $c(x, \omega)$ and the acoustic pressure $p(x, t, \omega)$ at different time instances are provided. The computation is performed using the HLL two wave Rusanov solver and a first order accurate piecewise constant reconstruction on the mesh resolution of $4096 \times 4096$ cells, and took 7 minutes on 256 cores. Notice that since the correlation length is long in $x$-dimension and short in $y$-dimension, the resulting realization of random material coefficient $c(x, \omega)$ exhibits layered structures: many layers with very heterogeneous wave propagation speeds are visible.

The structure of the acoustic pressure wave propagation consists of close-to-circular wave fronts interfering in the center of the domain and getting distorted by the heterogeneity of the underlying physical domain.

Results of the stochastic MLMC-FVM simulation up to $t = 1.0$ are presented in Figures \(^{10.10} - 10.11\) where the approximated mean and variance of the random material coefficient $c(x, \omega)$ and the acoustic pressure $p(x, t, \omega)$ at different time instances are provided. The computation is performed using an HLL two wave Rusanov solver and second order accurate piecewise linear WENO reconstruction. The number of levels is set to 9, i.e. $L = 8$, and the mesh resolution on the finest mesh level is $8192 \times 8192$ cells. The number of MC samples at the finest resolution is 8. We also note that the number of uncertainty sources in this simulation is very large: the material coefficient $c(x, \omega)$ was sampled on the resolution of $128 \times 128$, resulting in 16 384 sources of uncertainty in coefficients. The simulation took almost 5 hours (wall-clock) on 8176 cores (simulated on CSCS production cluster Rosa \(^{[126]}\)).
MLMC-FVM for linear systems of conservation laws with random coefficients

Figure 10.9: Top left plot: one sample of the coefficient $c(x,y,\omega)$ with variance $\sigma^2 = 0.2$ and correlation lengths $\eta_1 = 2.0, \eta_2 = 0.1$. Remaining plots: time snapshots of the approximated acoustic pressure $p(x,y,\omega)$. Since the correlation length is long in $x$-dimension and short in $y$-dimension, the resulting random material coefficient $c(x,y,\omega)$ exhibits layered structures: notice many layers with very heterogeneous wave propagation speeds.
10.4 Acoustic wave propagation in random heterogeneous medium

As expected, the mean and the variance of the material coefficient $c(x, y, \omega)$ are homogeneous within the entire domain, since a stationary covariance kernel is used. Since statistical moments of $c(x, \omega)$ are known, the results in Figure 10.10 are also used as a “self-test” of the MLMC-FVM, including the spectral generation of the samples of the log-normally distributed material coefficient $c(x, \omega)$ presented in subsection 10.3.1 and the upscaling presented in subsection 10.3.2.

The structure of the propagation of the mean acoustic pressure waves shown in Figure 10.11 resembles the mean behavior of the circular interfering waves seen in the deterministic simulation of one sample, given in the previous Figure 10.9. The largest variances are observed at the top of the domain, i.e. at the regions where the interference of the waves is strongest.

Finally, in Figure 10.12 the distributions of random maximal wave speeds across all levels are depicted. Notice that the spread is approximately from 1 to 3.5, strongly violating the assumption of equal computational work for samples at the same mesh resolution level, needed for the design of static load balancing in section 5.3. Hence, the simulation was executed using the adaptive load balancing technique introduced in section 5.6. Furthermore, as an additional advantage of adaptive load balancing, asymptotically optimized numbers of samples $M_\ell$ as in (4.23) were used.

Next, we use the high-resolution MLMC-FVM simulations from Figure 10.11 as the reference solution $U_{ref}$ and investigate the convergence of the error vs. computational work in Figure 10.13 and Figure 10.14. We consider both strategies for choosing the number of samples $M_\ell$ in (4.4), i.e. the asymptotically equilibrated $M_\ell$ as in (4.15) and asymptotically optimized $M_\ell$ as in (4.23), labeled with suffixes 'equ' and 'opt' in the legends. We note that as expected from the comparison of the error estimates

<table>
<thead>
<tr>
<th>$L$</th>
<th>$M_L$</th>
<th>grid size</th>
<th>CFL</th>
<th>cores</th>
<th>runtime</th>
<th>efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>8</td>
<td>8192x8192</td>
<td>0.475</td>
<td>8176</td>
<td>4:28:57</td>
<td>97.4%</td>
</tr>
</tbody>
</table>

Figure 10.10: MLMC-FVM estimates for mean and variance of the coefficient $c(x, y, \omega)$ with variance $\sigma^2 = 0.2$ and correlation lengths $\eta_1 = 2.0, \eta_2 = 0.1$. Stationary covariance kernel resulted in homogeneous moments of the coefficient.
MLMC-FVM for linear systems of conservation laws with random coefficients

Figure 10.11: MLMC-FVM estimates for mean (left) and variance (right) of time snapshots of the approximated acoustic pressure $p(x,y,\omega)$. The structure of the mean acoustic pressure is symmetric and consists of smooth circular wave fronts, i.e. it resembles the mean behavior of the (distorted) circular interfering waves seen in the deterministic simulation in Figure 10.9. The largest variances are observed at the top of the domain.

<table>
<thead>
<tr>
<th>$L$</th>
<th>$M_L$</th>
<th>grid size</th>
<th>CFL</th>
<th>cores</th>
<th>runtime</th>
<th>efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>8</td>
<td>8192x8192</td>
<td>0.475</td>
<td>8176</td>
<td>4:28:57</td>
<td>97.4%</td>
</tr>
</tbody>
</table>
10.4 Acoustic wave propagation in random heterogeneous medium

Figure 10.12: Distributions of random maximal wave speeds of samples for all resolution levels in the MLMC-FVM simulation reported in Figures 10.10 - 10.11. The spread is approximately from 1 to 3.5, leading to very heterogeneous computational work for samples at the same mesh resolution level.

(4.28) and (4.19), the asymptotically optimized choice (4.23) provided slightly higher accuracy (lower error) for the same amount computational work when estimating the mean in Figure 10.13 and the variance in Figure 10.14. Independently of which of the two strategies for $M_\ell$ is chosen, the errors in the mean and variance fields converge at the expected rates. At comparable numerical resolution and accuracy, the MLMC(2) is approximately one order of magnitude faster than the MC(2) method.

10.4.2 Two dimensional case with discontinuous layers

Next, in order to demonstrate the generality of the MLMC-FVM methods for very complex random material coefficients, we consider a setup analogous to subsection 10.4.1, however, this time we assume that the random material coefficient $c$ is given by independent log-normal distributions within four layers $D_1, D_2, D_3, D_4$ partitioning the domain $D$,

$$D_1 = [0, 2] \times [0, 0.5], \quad D_2 = [0, 2] \times [0.5, 1],$$
$$D_3 = [0, 2] \times [1, 1.5], \quad D_4 = [0, 2] \times [1.5, 2].$$

In each of the layers $D_1, \ldots, D_4$, we assume that the material coefficient is log-normally distributed with stationary periodic covariance kernels (10.24) and layer-dependent variances $\sigma^2$ and correlation lengths $\eta_1, \eta_2$, given by

$$\sigma^2 = \begin{cases} 
0.2 & \text{in } D_1 \text{ and } D_3, \\
0.4 & \text{in } D_2 \text{ and } D_4, 
\end{cases} \quad (\eta_1, \eta_2) = \begin{cases} 
(1.0, 0.5) & \text{in } D_1 \text{ and } D_3, \\
(2.0, 0.1) & \text{in } D_2 \text{ and } D_4. 
\end{cases}$$

(10.28)

189
MLMC-FVM for linear systems of conservation laws with random coefficients

Figure 10.13: Error convergence of the mean of the acoustic pressure $p(x, y, \omega)$. The asymptotically optimized number of samples (4.23) provided slightly higher accuracy for the same computational work. Both MLMC2 methods are approximately two orders of magnitude faster than the MC2 method.

Figure 10.14: Error convergence of the variance of the acoustic pressure $p(x, y, \omega)$. The asymptotically optimized number of samples (4.23) provided slightly higher accuracy for the same computational work. Both MLMC(2) methods are approximately one order of magnitude faster than the MC(2) method.
The initial data is chosen to be deterministic and set to zero, i.e. $p_0(x, \omega) \equiv 0$, $u_0(x, \omega) \equiv 0$. Identical periodic (in time) acoustic pressure pulses are injected into two locations of the domain through the deterministic source term $f$ given in (10.27). Perfectly reflecting boundary conditions are assumed at the top and the bottom, and periodic boundary conditions are assumed at the sides of the domain. Results of the deterministic FVM simulation at $t = 1.0$ are presented in Fig. 10.15, where the layers $D_1, \ldots, D_4$ are depicted in bottom to top order. The computation is performed using the HLL two wave Rusanov solver and a second order accurate piecewise linear WENO reconstruction on the mesh resolution of $2048 \times 2048$ cells, and took 15 minutes on 64 cores. Notice that in layers $D_2$ and $D_4$, the correlation length is long in $x$-dimension and short in $y$-dimension, resulting in even finer layered structures. On the other hand, in layers $D_1$ and $D_3$, the correlation lengths are similar in both directions, and hence no evident layered structures are present in the resulting random material coefficient. At the interfaces of layers $D_1, D_2, D_3, D_4$, each realization as well as the mean and variance of the random material coefficient $c$ is discontinuous; this, however, is expected, as the distributions of $c$ within each layer were assumed to be independent.

Results of the stochastic MLMC-FVM simulation up to $t = 1.0$ are presented in Figures 10.17–10.18 where the approximated mean and variance of the random material coefficient $c(x, \omega)$ and the acoustic pressure $p(x, t, \omega)$ at different time instances are provided. The computation is performed using the HLL two wave Rusanov solver and second order accurate piecewise linear WENO reconstruction. The number of levels is set to 5, i.e. $L = 4$, and the mesh resolution on the finest mesh level is $2048 \times 2048$ cells. The number of MC samples at the finest resolution is 16, i.e. the same as in the previous simulation. The simulation took almost 1 hour (wall-clock) on 992 cores (simulated on CSCS production cluster Rosa [126]).

The distributions of random maximal wave speeds across all levels are depicted in Figure 10.16. The spread is again large, approximately from 1.5 to 5.0, hence, adaptive load balancing was used, together with asymptotically optimized numbers of samples $M_\ell$ as in (4.23).

The mean and the variance of the material coefficient $c(x, \omega)$ in Figure 10.10 consist of interchanging regions, discontinuous in $y$-direction; such heterogeneity in statistical moments is due to different covariance kernels in (10.28).

The structure of the propagation of the mean acoustic pressure waves in Figure 10.18 again resembles the mean behavior of the circular interfering wave seen in the deterministic simulation of one sample, depicted in Figure 10.15.

**10.4.3 Three dimensional case**

The physical periodic (in each dimension) domain is $D = [0, 2] \times [0, 2] \times [0, 2]$ with $d = 3$. We assume a stationary periodic covariance kernel (10.24) with variance $\sigma^2 = 0.2$ and correlation lengths $\eta_1 = 2.0, \eta_2 = 0.1, \eta_3 = 2.0$ in (10.24). The initial data is chosen to
MLMC-FVM for linear systems of conservation laws with random coefficients

Figure 10.15: Top left: one sample of the coefficient $c(x, y, \omega)$ with layer-dependent variances and correlation lengths as specified in (10.28). Remaining plots: time snapshots of the approximated acoustic pressure $p(x, y, \omega)$. Interchanging layers of isotropic and anisotropic material coefficient regions $c(x, y, \omega)$ are present; at the interfaces, each realization of the random material coefficient $c(x, y, \omega)$ is discontinuous in $y$-direction.
10.4 Acoustic wave propagation in random heterogeneous medium

Figure 10.16: Distributions of random maximal wave speeds of samples for all resolution levels in the MLMC-FVM simulation reported in Figures 10.17 - 10.18. The spread is approximately from 1.5 to 5.0, leading to very heterogeneous computational work for samples at the same mesh resolution level.

Figure 10.17: MLMC-FVM estimates for mean and variance of the coefficient $c(x, y, \omega)$ with variances and correlation lengths as specified in (10.28). Alternating regions, discontinuous in $y$-direction, of mean and variance of the material coefficient $c(x, y, \omega)$ are due to different covariance kernels in (10.28).
Figure 10.18: MLMC-FVM estimates for the mean (left) and the variance (right) of time snapshots of the approximated acoustic pressure $p(x, y, \omega)$. The structure of the mean acoustic pressure is symmetric and consists of smooth circular wave fronts, i.e. it resembles the mean behavior of the (distorted) circular interfering waves seen in the deterministic simulation in Figure 10.15. Largest variances are observed at the top of the domain.
10.4 Acoustic wave propagation in random heterogeneous medium

be deterministic and set to zero, i.e. \( p_0(x, \omega) \equiv 0, u_0(x, \omega) \equiv 0 \). Analogously, identical periodic (in time) acoustic pressure pulses are injected into two locations of the domain through the deterministic source term \( f \),

\[
f(x, t, \omega) = \begin{cases} 
A_p \left( \exp \left( -\frac{\|x-x_1\|}{2\sigma} \right) + \exp \left( -\frac{\|x-x_2\|}{2\sigma} \right) \right) & \text{if } \{3t\} < 0.02, \\
0 & \text{else.}
\end{cases}
\]

where \( A_p = 5000, x_1 = (0.5, 2.0), x_2 = (1.5, 2.0), \sigma = 0.04 \) and \( \{\cdot\} \) denotes the fractional part. Perfectly reflecting boundary conditions (1.106) - (1.107) are assumed at the top and the bottom, and the periodic boundary conditions (1.102) - (1.103) are assumed at the sides of the domain.

Results of the deterministic FVM simulation up to \( t = 1.0 \) are presented in Fig. 10.19, where the approximated sample of the random material coefficient \( c(x, \omega) \) and the acoustic pressure \( p(x, t, \omega) \) at different time instances are provided. The computation is performed using the HLL two wave Rusanov solver and a second order accurate piecewise linear WENO reconstruction on the mesh resolution of \( 1024 \times 1024 \times 1024 \) cells, and took almost 3 hours on 4096 cores. Analogously to the two-dimensional experiment in Sect. 10.4.1, the long correlation length in \( xz \)-plane combined with short correlation length in \( y \)-dimension results in a random material coefficient exhibiting layered structures. For a slice taken parallel to the \( xz \)-plane, the “entire” layer is obtained. The structure of acoustic pressure wave propagation consists of close-to-circular wave fronts interfering in the center of the domain and getting distorted by the heterogeneity of the underlying physical domain.

Results of the stochastic MLMC-FVM simulation up to \( t = 1.0 \) are presented in Figure 10.20, where the approximated mean and variance of the random material coefficient \( c(x, \omega) \) and the acoustic pressure \( p(x, t, \omega) \) at different time instances are provided. The computation is performed using the HLL two wave Rusanov solver and a second order accurate piecewise linear WENO reconstruction. The number of levels is set to 7, i.e. \( L = 6 \), and the mesh resolution on the finest mesh level is \( 1024 \times 1024 \times 1024 \) cells. The number of MC samples at the finest resolution is 8. We also note that the number of uncertainty sources in this simulation is even larger than for the 2-D case: the material coefficient \( c(x, \omega) \) was sampled on the resolution of \( 128 \times 128 \times 128 \), resulting in more than 2 million (2 097 152) sources of uncertainty in coefficients. The simulation took almost 3 hours (wall-clock) on 43680 cores (simulated on CSCS production cluster Rosa [126]).

The distributions of random maximal wave speeds across all levels are depicted in Figure 10.21. The spread is again large, approximately from 1 to 3, hence, adaptive load balancing was used, together with optimal numbers of samples \( M_\ell \) as in (4.23).
Figure 10.19: Top left: one sample of the coefficient \(c(x, \omega)\) with \(\sigma^2 = 0.2\) and \(\eta_1 = \eta_3 = 2.0, \eta_2 = 0.1\). Remaining plots: time snapshots of the approximated acoustic pressure \(p(x, \omega)\). Due to anisotropic correlation lengths, coefficient \(c(x, \omega)\) exhibits layered structures, distorting the circular wave fronts of the acoustic pressure.
Figure 10.20: MLMC-FVM estimates for mean (left) and variance (right) of time snapshots of the approximated acoustic pressure \( p(x, y, \omega) \). The structure of the mean acoustic pressure is symmetric and consists of smooth circular wave fronts, i.e. it resembles the mean behavior of the (distorted) circular interfering waves seen in the deterministic simulation in Figure 10.19. Largest variances are again observed at the top of the domain.
MLMC-FVM for linear systems of conservation laws with random coefficients

Figure 10.21: Distributions of random maximal wave speeds of samples for all resolution levels in the MLMC-FVM simulation reported in Figure 10.20. The spread is approximately from 1 to 3, leading to very heterogeneous computational work for samples at the same mesh resolution level.
11 MLMC-FVM for nonlinear systems with random fluxes

11.1 Euler equations with uncertain equation of state

For the final set of numerical results, we analyze a nonlinear system of conservation laws with random fluxes.

In particular, consider a random version of the Euler equations (1.8) with the random constant of specific heats in (1.9):

\[ \gamma = \gamma(\omega), \quad \gamma \sim U(5/3 - \varepsilon, 5/3 + \varepsilon), \quad \varepsilon = 0.1. \] (11.1)

11.1.1 Uniformly perturbed specific heats in 1-D: Sod shock tube

We consider the one-dimensional version of the Euler equations (1.8) with (11.1) in the domain \( D = [0, 2] \) with Neumann (“transparent”) boundary conditions (1.104) - (1.105). The initial data consists of the following shock at \( x = 1 \):

\[ U_0(x, \omega) = \{ \rho_0(x, \omega), u_0(x, \omega), p_0(x, \omega) \} = \begin{cases} 
\{3.0, 0.0, 3.0\} & \text{if } x < 1, \\
\{1.0, 0.0, 1.0\} & \text{if } x > 1.
\end{cases} \] (11.2)

The initial data (11.2) and the reference solution at time \( t = 0.5 \) are depicted in Figure 11.1. There are 9 levels (\( L = 8 \)) of FVM mesh resolution with the finest resolution (at the finest level \( \ell = L \)) being 2048 cells. Rusanov flux with second order accurate positivity preserving WENO reconstruction was used. At every point \( x \in [0, 2] \) the solid line represents the mean and the dashed lines represent the mean ± standard deviation of the (random) solution. For each sample of the random constant \( \gamma \) of specific heats, the initial shock splits into three waves: a leftward moving rarefaction wave, a rightward moving contact discontinuity and a rightward moving shock wave. The variances are relatively small (compared to the mean) with a concentration of the variance near the shock.
MLMC-FVM for nonlinear systems with random fluxes

Figure 11.1: MLMC-FVM solution of the Sod shock tube (11.2) with random specific heats constant (11.1). The variances are relatively small (compared to the mean) with a concentration of the variance in the vicinity of the shock.

11.1.2 Uniformly perturbed specific heats in 2-D: cloud-shock

The MLMC-FVM algorithm is tested on a problem that is analogous to the one presented in subsection 7.1.6. The computational domain is again taken to be $D = [0, 1] \times [0, 1]$ with Neumann (“transparent”) boundary conditions (1.104) - (1.105). Compared to subsection 7.1.6, we assume a random constant of specific heats (11.1) and we consider the deterministic initial data for the cloud shock problem:

\[
\{\rho_0(x, \omega), u_0(x, \omega), p_0(x, \omega)\} = \begin{cases} 
3.86859, (11.2536, 0)^T, 167.345 & \text{if } x_1 < 0.05, \\
1, (0, 0)^T, 1 & \text{if } x_1 > 0.05,
\end{cases}
\]

(11.3)

with a high density cloud (or bubble) lying to the right of the shock

\[
\rho_0(x, \omega) = 10, \quad \text{if } \sqrt{(x_1 - 0.25)^2 + (x_2 - 0.5)^2} \leq 0.15.
\]

(11.4)

The mean and variance of the solution at time $t = 0.06$ obtained by the numerical simulation using MLMC-FVM are given in Figure 11.2. The results are from a MLMC-WENO run with 9 nested levels of resolution ($L = 8$) and the finest resolution is set to $2048 \times 2048$ mesh. Rusanov flux with second order accurate positivity preserving WENO reconstruction was used. The number $M_L$ of MC samples at the finest resolution is 8 and number of cores for this run is 128. Although the flow and uncertainty appear to be similar to the one discussed in subsection 7.1.6, there are important differences. In particular, the variance near the bow and tail shocks appears to be spread over a larger region compared to the case of random initial data. Furthermore, the smooth regions after the bow shock have a very different distribution of uncertainty in this case.
11.1 Euler equations with uncertain equation of state

Figure 11.2: MLMC-FVM solution of the cloud-shock (11.3) - (11.4) with random specific heats constant (11.1). Compared to the numerical experiment in sub-section 7.1.6, the variance near the bow and tail shocks appears to be spread over a larger region. Furthermore, the smooth regions after the bow shock have a very different distribution of uncertainty in this case.
MLMC-FVM for nonlinear systems with random fluxes
12 Summary and conclusions

The issue of uncertainty quantification for physical and engineering applications which are modeled by hyperbolic systems of balance laws with random input data has received increasing attention in recent years. The inputs to hyperbolic systems of balance laws such as initial data, boundary conditions, source terms, fluxes and coefficients, are in general uncertain due to inherent uncertainties in the measurement process. Input data uncertainty, modeled using probabilistic mathematical descriptions, propagates into the solution, and hence methods for efficient and robust numerical quantification of uncertainty (UQ) are required.

In this thesis, recent results on existence and uniqueness of the random entropy solutions as well as the design, analysis and implementation of efficient statistical sampling methods of the Monte Carlo (MC) and Multi-Level Monte Carlo (MLMC) type for quantifying uncertainty in the solutions of systems of random balance laws with random input data were presented.

Uncertain inputs such as random initial data, sources, fluxes and coefficients are modeled in terms of random fields, i.e. within the classical probabilistic framework of Kolmogorov. The corresponding notion of random entropy solutions was introduced and the problem of uncertainty quantification in scalar and linear hyperbolic systems of multi-dimensional conservation laws with random inputs was shown to be well-posed. Furthermore, statistical regularity (in particular, existence of k-th statistical moments) of the random entropy solutions was shown.

The MC-FVM and MLMC-FVM algorithms for systems of balance laws were presented. Efficient high-resolution finite volume schemes were used for the spatio-temporal discretizations of the balance laws and were combined with the MC and MLMC algorithms. The mean square error convergence analysis and a novel probabilistic complexity analysis of the mean square error vs. expected computational cost of the resulting schemes (in scalar and linear hyperbolic systems case) were presented, leading to several strategies for the optimal number of samples for each resolution level in MLMC-FVM. In particular, we showed that the MLMC-FVM has the same asymptotic complexity (up to a logarithmic term) as a single deterministic finite volume solve.

Details of our implementation ALSVID-UQ of the MC-FVM and MLMC-FVM for systems of balance laws were described, including parallel robust statistical estimators, pseudo random number generators and the choice of the underlying problem-specific deterministic FVM solver. Static and, by design much more widely applicable, adaptive load balancing strategies for homogeneous MIMD hardware were proposed. Strong and weak
Summary and conclusions

scaling was experimentally verified up to 40 000 cores on the high performance clusters Brutus in ETH Zürich [127] and Rosa in Swiss National Supercomputing Center [126].

A key challenge in designing efficient MLMC algorithms is the multi-resolution representation of random spatially inhomogeneous coefficients and source terms. Multi-level hierarchical representations and multi-resolution sample generation techniques were proposed, leading to significant speed up over standard MLMC-FVM algorithm: a novel multi-resolution FFT spectral generator with level-coupled random number generation for statistical sampling of stochastic material coefficients in acoustic wave equations, and a multi-level representation methodology for general MLMC-FVM treatment of bias-free multi-level sampling of random source terms, such as the uncertain bottom topography in shallow water equations.

In addition to mean and variance estimators, the MLMC-FVM framework was applied as a novel non-intrusive technique for computing the approximate probabilities of statistical events of engineering interest based on random entropy solutions, such as probabilities of extreme (rare) events in engineering risk and failure analysis.

A large number of numerical experiments supporting the findings from the theory and illustrating the efficiency of the MLMC-FVM method were presented. These experiments included Euler and MHD equations with uncertain initial data and uncertain equation of state (uncertain flux), shallow water equations with uncertain initial data and uncertain bottom topography, Buckley-Leverett equations with uncertain relative permeabilities, and the acoustic wave equation with uncertain heterogeneous log-normally distributed material coefficients. The role of the non-linearity in the evolution of the variance, as well as a gain in regularity of the mean due to ensemble averaging, are described.

The MLMC-FVM-based statistical sampling method was demonstrated to be a robust, efficient and powerful tool in the context of computational uncertainty quantification for very complex flows governed by systems of nonlinear hyperbolic balance laws with uncertain initial conditions. The MLMC methodology is completely non-intrusive, as it can be readily used in conjunction with any spatio-temporal discretization of the underlying system of conservation laws. Additionally, this methodology is very flexible, as different types of uncertain inputs, such as random initial data, source terms, flux functions and coefficients, can be used, even with very low regularity assumptions (presence of discontinuities is allowed) on the underlying random input data and entropy solutions.

MLMC-FVM can deal with a very large number of sources of uncertainty. For instance, the computation for shallow water equations with uncertain bottom topography involved approximately 1000 sources of uncertainty, and the simulation of acoustic wave propagation with uncertain material coefficient involved 2 million sources of uncertainty. To the best of our knowledge, no other method (particularly deterministic methods such as quasi Monte Carlo, stochastic Galerkin, stochastic collocation, or stochastic FVM) is able to efficiently handle this many sources of uncertainty (i.e., high “stochastic dimensions”),

204
Summary and conclusions

in particular with solutions of low regularity and possibly non-smooth dependence on random input fields.

Many advantageous properties of the MLMC-FVM methods are directly inherited from the standard Monte Carlo estimates, based on convex combinations of sampled numerical solutions and naturally inheriting non-oscillation, discrete maximum principle and monotonicity properties of the underlying FV discretization schemes. Since MLMC-FVM estimators are not based on convex combinations of numerical FVM solutions, these (desirable) properties of discretization schemes can no longer be ensured. However, our error convergence analysis for scalar and linear hyperbolic systems of balance laws indicates that MLMC-FVM methods offer a significantly improved asymptotic rate of convergence, which was shown to be even optimal (i.e. leads to the same error vs. work complexity as a single FVM solve of an equivalent deterministic problem), provided that the FVM convergence rate is sufficiently large. In our numerical experiments, we observed that already in the engineering range of accuracy, MLMC-FVM methods are several orders of magnitude faster than plain MC methods. Such efficiency gains afforded by the multi-level methodology outweigh the lack of monotonicity in MLMC-FVM. Nevertheless, awareness of these effects is warranted in the interpretation of simulation results.

A potential disadvantage of the MLMC-FVM method arises due to the sampling on coarse mesh resolutions: in some cases, the FVM approximations can be very inaccurate at coarse mesh levels, with asymptotic error convergence rates available only for very fine meshes. For instance, in order to resolve small scale heterogeneities in the realizations of the log-normally distributed material coefficients, a sufficiently fine mesh resolution needs to be used in FVM, limiting the number of hierarchical mesh levels in MLMC.

In [100, 76, 108], the MLMC-FVM was compared to stochastic Galerkin and stochastic Finite Volume (SFVM) methods. Even for a moderate number of sources of uncertainty, the MLMC-FVM method was found to provide significantly smaller errors than SFVM (for the same amount of computational work). In the cases where the regularity of the statistical moments of the solution is high (for examples we refer to [100, 108]), the stochastic Galerkin method was shown to converge with asymptotically higher rates than MLMC-FVM. However, for non-linear hyperbolic systems of random conservation laws, statistical moments of high regularity are present only in special cases [100, 108].

Given these advantages, the Multi-level Monte Carlo Finite Volume Method, with carefully chosen number of hierarchical mesh levels and efficient (possibly fault-tolerant) implementation on massively parallel (possibly heterogeneous) hardware architectures, appears to be a powerful general purpose technique for quantifying uncertainty in solutions of complex flow problems governed by systems of balance laws with uncertain inputs.
Summary and conclusions
References


References


References


References


References


References


References


References


[126] *Rosa and Palü (Cray XE6)*, Swiss National Supercomputing Center (CSCS), Lugano, www.cscs.ch

[127] *Brutus*, ETH Zürich, de.wikipedia.org/wiki/Brutus_(Cluster)
List of Publications


