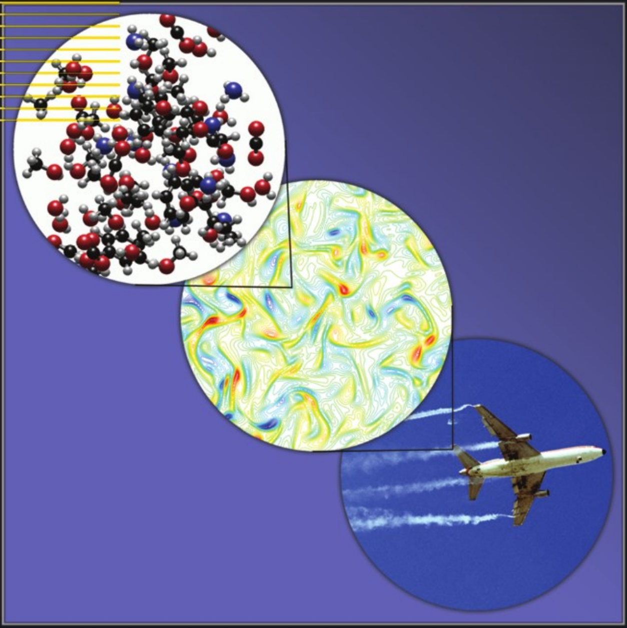
Lecture Notes in Computational Science and Engineering

44



Björn Engquist
Per Lötstedt
Olof Runborg
Editors

Multiscale Methods in Science and Engineering

Springer

Lecture Notes in Computational Science and Engineering

44

Björn Engquist
Per Lötstedt
Olof Runborg
Editors

Multiscale Methods in Science and Engineering

With 85 Figures and 17 Tables

Springer

Editors

Björn Engquist
Olof Runborg

Department of Numerical Analysis
and Computer Science
Royal Institute of Technology
SE-10044 Stockholm, Sweden

engquist@nada.kth.se
olofr@nada.kth.se

Per Lötstedt

Department of Information Technology
Uppsala University
Box 337
SE-75105 Uppsala, Sweden

perl@it.uu.se

Library of Congress Control Number: 2005922827

Mathematics Subject Classification (2000): 65-06, 65M99, 65N99

ISSN 1439-7358
ISBN-10 3-540-25335-1 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-25335-8 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media
springeronline.com

Printed in The Netherlands

Cover production: *design & production*, Heidelberg
Typeset by the authors and TechBooks using a Springer TEX macro package

Printed on acid-free paper 46/3142/sz - 5 4 3 2 1 0

Preface

Multiscale problems naturally pose severe challenges for computational science and engineering. The smaller scales must be well resolved over the range of the larger scales. Multiscale objects must therefore typically be described by a very large set of unknowns. The larger the ranges of scales, the more unknowns are needed and the higher the computational cost. It has been possible to meet many of these challenges by the recent progress in multiscale computational techniques coupled to the capability of the latest generation of computer systems.

This recent progress was presented at the conference *Multiscale Methods in Science and Engineering*, which was held in Uppsala, Sweden on January 26–28, 2004. More than 55 participants from six countries discussed the issues presented in the papers of this proceeding. The conference was sponsored by he Swedish Foundation for Strategic Research (SSF) and by the Swedish Agency for Innovation Systems, Vinnova via the Parallel and Scientific Computing Institute (PSCI).

Challenging multiscale problems are very common. One example can be average airflow, which typically depends on the details of small swirling eddies, which in turn depend on the interaction of molecules on much smaller scales in space and time. One can go further and see how the forces between the molecules depend on the electrons. Typically, a narrow range of scales is modeled by effective equations for that particular range. Turbulence models would then describe the coarsest scales of the phenomena mentioned above. The finer scales could be approximated by the Navies–Stokes equations, the Boltzmann equation and the Schrödinger equation respectively.

When such effective equations for a narrow range of scales can be derived the numerical approximations can be greatly facilitated. These equations should include the influence from other scales in the original multiscale problem. Techniques of this type are presented in this proceeding. In the contributions by Berlyand et al. and Svanstedt and Wellander, new variants of the homogenization technique are described and analyzed. Stochastic differential equations are increasingly common models for multiscale phenomena. New adaptive techniques for stochastic equations are developed by Dzougoutov et al. Stochastic models are also part of the systems studied by Jourdain et al. Sometimes there exist well performing equations for most of the computational domain but a small subdomain contains microscales that are difficult to represent by the numerical method. Special subgrid models need to be developed. Edelvik derives such models for thin wires and slots in electromagnetic simulations. Thin filaments or fibers in fluids are approximated in the contribution by Tornberg. The latter simulations can also be seen as a way of numerically deriving effective equations for suspensions of filaments in fluids. The multiscale discontinuous Galerkin method studied by Aarnes and Heimsund uses multiscale basis functions and is based on homogenization theory.

An important preprocessing step for all numerical multiscale computations is the choice of unknowns. The number of these unknowns should be kept to a minimum. In the two contributions by Larson and collaborators this is achieved by adaptive grid generation based on realistic a posteriori estimates. Runborg uses a wavelet like technique that allows for a hierarchical and efficient representation of geometrical structures.

Computational multiscale methods are of two types. In the more established class of methods the full multiscale problem is discretized and highly efficient numerical methods are then applied to accurately compute the full range of scales. Multigrid, and the fast multipole method are very successful examples of such technique. These algorithms rely on special properties of the solution operator in order to achieve their optimal computational complexity. The smoothing by elliptic operators is one such example. Eberhard and Wittum presents a multigrid method for flow in heterogeneous porous media and a multipole method for electromagnetic scattering is described by Nilsson and Lötstedt.

In the second and more recent class of computational multiscale methods only a fraction of the microscale space is included in order to reduce the number of unknowns. The microscales and the macroscales are coupled in the same simulation exploiting special properties in the original problem, for example, scale separation. The simulation over a wide range of scales can be based on first principles even if effective equations are not known. The techniques discussed by E and Engquist, Jourdain et al., Samaey et al. and Sharp et al. in this proceeding are examples of this type of methods.

There are several active areas of development at the present time for tackling the multiscale challenge and many of the important ones were presented at this conference. The progress will have importance on the whole field of computational science and engineering. Multiscale modeling is emerging as a new computational paradigm.

Stockholm and Uppsala
April 2005

Björn Engquist
Per Lötstedt
Olof Runborg

Contents

List of Contributors

Jørg Aarnes
SINTEF Applied Mathematics
PB 124, N-0314 Oslo, Norway
Jorg.Aarnes@sintef.no

Fredrik Bengzon
Department of Mathematics
Umeå University
90187 Umeå, Sweden
fredrik.bengzon@math.umu.se

Leonid Berlyand
Department of Mathematics and
Materials Research Institute
Pennsylvania State University,
McAllister Bld.
University Park, PA 16802, USA
berlyand@math.psu.edu

Anna Dzougoutov
Department of Numerical Analysis and
Computer Science
KTH
SE-100 44 Stockholm, Sweden
annadz@kth.se

Weinan E
Department of Mathematics
Princeton University
Princeton, NJ 08544, USA
weinan@math.princeton.edu

Jens Eberhard
Simulation in Technology
University of Heidelberg
Im Neuenheimer Feld 368
D-69120 Heidelberg, Germany
eberhard@uni-hd.de

Fredrik Edelvik
Division of Scientific Computing
Department of Information Technology
Uppsala University
SE-75105 Uppsala, Sweden
fredrik.edelvik@it.uu.se

Björn Engquist
Department of Numerical Analysis and
Computer Science
KTH
SE-100 44 Stockholm, Sweden
engquist@nada.kth.se

Yuliya Gorb
Department of Mathematics and
Materials Research Institute
Pennsylvania State University,
McAllister Bld.
University Park, PA 16802, USA
gorb@math.psu.edu

Bjørn-Ove Heimsund
University of Bergen
Allégaten 41, N-5007 Bergen, Norway
Bjorn-Ove.Heimsund@uib.no

August Johansson
Department of Mathematics
Umeå University
90187 Umeå, Sweden
august.johansson@math.umu.se

Benjamin Jourdain
CERMICS
Ecole Nationale des Ponts et Chaussées
6 & 8 Av. Pascal, F–77455 Champs-sur-Marne, France
jourdain@cermics.enpc.fr

Ioannis G. Kevrekidis
Department of Chemical Engineering
PACM and Department of Mathematics
Princeton University
Princeton, NJ 08544, USA
yannis@princeton.edu

Mats G. Larson
Department of Mathematics
Umeå University
90187 Umeå, Sweden
mats.larson@math.umu.se

Claude Le Bris
CERMICS
Ecole Nationale des Ponts et Chaussées
6 & 8 Av. Pascal, F–77455 Champs-sur-Marne, France
lebris@cermics.enpc.fr

Tony Lelièvre
CERMICS
Ecole Nationale des Ponts et Chaussées
6 & 8 Av. Pascal, F–77455 Champs-sur-Marne, France
lelievre@cermics.enpc.fr

Per Lötstedt
Department of Information Technology
Division of Scientific Computing
Uppsala University, SE-75105 Uppsala, Sweden
per.lotstedt@it.uu.se

Axel Målqvist
Department of Mathematics
Chalmers University of Technology
SE-412 96, Göteborg, Sweden
axel@math.chalmers.se

Kyoung-Sook Moon
Department of Mathematics
University of Maryland
College Park, MD 20742, USA
moon@math.umd.edu

Martin Nilsson
Department of Information Technology
Division of Scientific Computing
Uppsala University, SE-75105 Uppsala, Sweden
martin.nilsson@it.uu.se

Alexei Novikov
Department of Mathematics and
Materials Research Institute
Pennsylvania State University,
McAllister Bld.
University Park, PA 16802, USA
anovikov@math.psu.edu

Dirk Roose
Department of Computer Science
K.U. Leuven
Celestijnenlaan 200A, B-3000 Leuven, Belgium
dirk.roose@cs.kuleuven.ac.be

Olof Runborg
Department of Numerical Analysis and
Computer Science
KTH
SE-100 44 Stockholm, Sweden
olofr@nada.kth.se

Giovanni Samaey
Department of Computer Science
K.U. Leuven
Celestijnenlaan 200A, B-3000 Leuven, Belgium
giovanni.samaey@cs.kuleuven.ac.be

Erik von Schwerin
Department of Numerical Analysis and Computer Science
KTH
SE-100 44 Stockholm, Sweden
schwerin@nada.kth.se

Richard Sharp
Program in Applied and Computational Mathematics
Princeton University
Princeton, NJ 08544, USA
rsharp@math.princeton.edu

Nils Svanstedt
Department of Mathematics
Chalmers University of Technology and Göteborg University,
SE-412 96 Göteborg, Sweden
nilss@math.chalmers.se

Anders Szepessy
Department of Numerical Analysis and Computer Science
KTH
SE-100 44 Stockholm, Sweden
szepessy@nada.kth.se

Raúl Tempone
ICES
University of Texas at Austin,
1 Texas Longhorns,
Austin, Texas 78712, USA
rtempone@ices.utexas.edu

Anna-Karin Tornberg
Courant Institute of Mathematical Sciences
New York University
251 Mercer Street, New York, NY 10012-1185, USA
tornberg@cims.nyu.edu

Yen-Hsi Tsai
Department of Mathematics
University of Texas at Austin
1 University Station C1200
Austin, Texas 78712, USA
ytsai@math.utexas.edu

Niklas Wellander
Swedish Defence Research Agency, FOI
SE-581 11 Linköping, Sweden
niklas@foi.se

Gabriel Wittum
Simulation in Technology
University of Heidelberg
Im Neuenheimer Feld 368
D-69120 Heidelberg, Germany
wittum@uni-hd.de

Multiscale Discontinuous Galerkin Methods for Elliptic Problems with Multiple Scales

Jørg Aarnes[1] and Bjørn–Ove Heimsund[2]

[1] SINTEF Applied Mathematics, PB. 124, 0314 Oslo, Norway.
Jorg.Aarnes@sintef.no
[2] University of Bergen, Allégaten 41, 5007 Bergen, Norway.
Bjorn-Ove.Heimsund@cipr.uib.no

Summary. We introduce a new class of discontinuous Galerkin (DG) methods for solving elliptic problems with multiple scales arising from e.g., composite materials and flows in porous media. The proposed methods may be seen as a generalization of the multiscale finite element (FE) methods. In fact, the proposed DG methods are derived by combining the approximation spaces for the multiscale FE methods and relaxing the continuity constraints at the inter-element interfaces. We demonstrate the performance of the proposed DG methods through numerical comparisons with the multiscale FE methods for elliptic problems in two dimensions.

Key words: multiscale methods, discontinuous Galerkin methods, elliptic partial differential equations

1 Introduction

We consider solving the second-order elliptic equation

$$\begin{cases} -\nabla \cdot (a(x)\nabla u) = f, & \text{in } \Omega \subset \mathcal{R}^{\mathrm{d}}, \\ u = 0, & \text{on } \Gamma_{\mathrm{D}} \subset \partial\Omega, \\ -a(x)\nabla u \cdot n = 0, & \text{on } \Gamma_{\mathrm{N}} = \partial\Omega \backslash \Gamma_{\mathrm{D}}, \end{cases} \tag{1}$$

where Ω is bounded, $\partial\Omega$ is Lipschitz, n is the outward unit normal on $\partial\Omega$ and $a(x) = (a_{ij}(x))$ is a symmetric positive definite tensor with uniform upper and lower bounds:

$$0 < \alpha|y|^2 \leq y^T a(x) y \leq \beta|y|^2 < \infty, \quad \forall x \in \Omega, \ \forall y \in \mathcal{R}^d, \ y \neq 0.$$

We will interpret the variable u as the (flow) potential and q as the (flow) velocity. The homogeneous boundary conditions are chosen for presentational brevity. General boundary conditions can be handled without difficulty.

Equation (1) may represent incompressible single-phase porous media flow or steady state heat conduction through a composite material. In single-phase flow, u

is the flow potential, $q = -a(x)\nabla u$ is the Darcy filtration velocity and $a(x)$ is the (rock) permeability of the porous medium. For heat conduction in composite materials, u, q and $a(x)$ represents temperature, heat flow density, and thermal conductivity respectively. These are typical examples of problems where $a(x)$ can be highly oscillatory and the solution of (1) displays a multiscale structure. This leads to some fundamental difficulties in the development of robust and reliable numerical models.

In this paper we introduce a new class of DG methods for solving this particular type of multiscale elliptic problems. Until recently, DG methods have been used mainly for solving partial differential equations of hyperbolic type, see e.g. [10] for a comprehensive survey of DG methods for convection dominated problems. Indeed, whereas DG methods for hyperbolic problems have been subject to active research since the early seventies, it is only during the last decade or so that DG methods have been applied to purely elliptic problems, cf. [5] and the references therein. The primary motivation for applying DG methods to elliptic problems is perhaps their flexibility in approximating rough solutions that may occur in elliptic problems arising from heterogeneous and anisotropic materials. However, to our knowledge, previous research on DG methods for elliptic problems has been confined to solving elliptic partial differential equations with smooth coefficients.

DG methods approximate the solution to partial differential equations in finite dimensional spaces spanned by piecewise polynomial base functions. As such, they resemble the FE methods, but, unlike the FE methods, no continuity constraints are explicitly imposed at the inter-element interfaces. This implies that the weak formulation subject to discretization must include jump terms across interfaces and that some artificial penalty terms must be added to control the jump terms. On the other hand, the weak continuity constraints give DG methods a flexibility which allows a simple treatment of, e.g., unstructured meshes, curved boundaries and h- and p-adaptivity. Another key feature with DG methods is their natural ability to impose mass conservation locally. Moreover, the "local" formulation of the discrete equations allows us us to use grid cells of arbitrary shapes without difficulty. We may therefore choose the gridlines to be aligned with sharp contrasts in, for instance, underlying heterogeneous materials.

The multiscale FE methods (MsFEMs) introduced in [9, 12] have been successfully applied to multiscale elliptic problems, but their accuracy is to some degree sensitive to the selection of the boundary conditions that determine the FE base functions. If, for instance, strong heterogeneous features penetrate the inter-cell interfaces, then simple, e.g. linear, boundary conditions may be inadequate. In such situations, oversampling strategies or other techniques for the generation of adaptive boundary conditions must be used to recover the desired order of accuracy. This sensitivity to the selection of boundary conditions is partly due to the strong continuity requirements at the inter-element interfaces implicit in the FE methods.

Here we propose a class of multiscale DG methods (MsDGMs) for solving elliptic problems with multiple scales. One of the primary motives for developing MsDGMs is to generate multiscale methods that are less sensitive to the selection of boundary conditions for the base functions than is the case for the MsFEMs. Another nice feature with MsDGMs is that they produce solutions for both the potential variable (e.g. pressure or temperature) and the velocity variable (e.g. phase velocity or thermal flux density) that reflect important subgrid variations in the elliptic coefficients. We will demonstrate the benefit of using multiscale methods in comparison with ordinary monoscale numerical methods and perform numerical experiments to display the performance of the MsDGMs relative to the original and mixed MsFEMs. We therefore attempt to reveal that there is a need for multiscale methods, and to demonstrate under what circumstances it may be advantageous to relax the inter-element continuity assumptions implicit in the MsFEMs.

The paper is organized as follows. We give the general mathematical setting for the DG methods in Sect. 2 and show how they are related to the more familiar FE methods. In particular we show that both standard and mixed FE methods may be viewed as special DG methods. This observation allows us to extend this type of FE methods to corresponding DG methods. In Sect. 3 we outline the MsFEMs introduced in [12] and [9] and exploit the relationship between FE methods and DG methods to derive a corresponding class of MsDGMs. Finally, Sect. 4 contains the numerical experiments and we conclude with a discussion of the results in Sect. 5.

2 Mathematical Formulations

In Sect. 2.1 we give the mathematical formulation of the DG methods for (1) and discuss the selection of the so-called numerical fluxes that are used to force weak continuity of the solution across inter-element interfaces. In Sect. 2.2 we show how the conforming and mixed FE methods may be viewed as special DG methods, and describe how such FE methods can be extended to corresponding DG methods.

2.1 Discontinuous Galerkin Methods

To define the DG methods we split (1) into the first order system,

$$\begin{aligned} q &= -a(x)\nabla u, && \text{in } \Omega, \\ \nabla \cdot q &= f, && \text{in } \Omega, \\ u &= 0, && \text{on } \Gamma_{\mathrm{D}}, \\ q \cdot n &= 0, && \text{on } \Gamma_{\mathrm{N}}. \end{aligned}$$

Furthermore, define the following approximation spaces:

$$\begin{aligned} Q_N &= \{p \in (H^1(\varOmega))^d : p \cdot n = 0 \text{ on } \varGamma_N\}, \\ U_D &= \{v \in H^1(\varOmega) : v = 0 \text{ on } \varGamma_D\}. \end{aligned}$$

Upon integration by parts, we now deduce the weak formulation: Find $q \in Q_N$ and $u \in U_D$ such that

$$\int_\varOmega a^{-1} q \cdot p \,\mathrm{d}x = \int_\varOmega u \nabla \cdot p \,\mathrm{d}x \qquad \forall p \in Q_N,$$

$$\int_\varOmega q \cdot \nabla v \,\mathrm{d}x = -\int_\varOmega f v \,\mathrm{d}x \qquad \forall v \in U_D.$$

In the DG methods, a similar set of equations is derived for each grid cell. However, for the grid cell equations it is not natural to impose homogeneous boundary conditions. The boundary conditions are therefore approximated from neighboring values of the unknown solution. Essentially we want to ensure that the potential u and the velocity q are “almost” continuous at the interfaces. Since we do not want to enforce continuity by imposing constraints on the approximation spaces as the FE methods do, we have to penalize the deviation from continuity by introducing an artificial penalty term. To understand the mechanism behind the penalty term, we digress for a moment in order to consider an example that illustrates the basic principle.

Example: Consider the Poisson equation with Dirichlet data,

$$\begin{cases} -\Delta u = f, \text{ in } \Omega, \\ \quad\;\; u = g, \text{ on } \partial\Omega, \end{cases}$$

and for each $\epsilon > 0$, let $u_\epsilon \in H^1(\varOmega)$ be the solution to the regularized problem

$$\int_\varOmega \nabla u_\epsilon \cdot \nabla v \,\mathrm{d}x + \int_{\partial\varOmega} \frac{1}{\epsilon}(u_\epsilon - g) v \,\mathrm{d}s = \int_\varOmega f v \,\mathrm{d}x \quad \forall v \in H^1(\varOmega). \tag{2}$$

Here ds denotes the surface area measure. This problem corresponds to perturbing the boundary data so that instead of $u = g$ we have $u + \epsilon \nabla u \cdot n = g$ on $\partial\varOmega$. One can show that (2) is well posed and that $u_\epsilon \longrightarrow u \in H_0^1(\varOmega)$ as $\epsilon \longrightarrow 0$ [14]. Hence, we see that the extra penalty term is added in order to force, in the limit $\epsilon \longrightarrow 0$, the satisfaction of the boundary conditions.

Just as the satisfaction of the Dirichlet boundary data was imposed weakly in (2), so can inter-element continuity be attained in a similar fashion. It was this observation that originally led to the development of the interior penalty (IP) methods [4, 11, 16]. Arnold et al. [5] recently recognized that the IP methods, along with several other methods with discontinuous approximation spaces, can be classified as DG methods. These methods differ in the flux approximating schemes used to force continuity at the inter element interfaces. We now describe the general framework for the DG methods with respect to the elliptic problem (1).

Let $\mathcal{T}(\varOmega) = \{T \in \mathcal{T}\}$ be a family of elements in a partitioning of $\varOmega$ and define $\partial\mathcal{T} = \cup\{\partial T : T \in \mathcal{T}\}$, $\varGamma = \partial\mathcal{T} \backslash \partial\varOmega$ and $\varGamma_{ij} = \partial T_i \cap \partial T_j$, $T_i, T_j \in \mathcal{T}$. Next, introduce an approximation space $Q^h \times U^h \subset (H^1(\mathcal{T}))^d \times H^1(\mathcal{T})$ where

$$H^k(\mathcal{T}) = \{w \in L^2(\varOmega) : w \in H^k(T), \forall T \in \mathcal{T}\}.$$

The DG method then seeks $q^h \in Q_N^h = Q^h \cap Q_N$ and $u^h \in U_D^h = U^h \cap U_D$ such that

$$\int_T a^{-1} q^h \cdot p \,\mathrm{d}x = \int_T u^h \,\nabla \cdot p \,\mathrm{d}x - \int_{\partial T} \bar{u}\, p \cdot n_T \,\mathrm{d}s \qquad \forall p \in Q_N^h, \tag{3}$$

$$\int_T q^h \cdot \nabla v \,\mathrm{d}x = -\int_T f v \,\mathrm{d}x + \int_{\partial T} v\, \bar{q} \cdot n_T \,\mathrm{d}s \qquad \forall v \in U_D^h, \tag{4}$$

for all $T \in \mathcal{T}$. Here n_T is the outward unit normal on ∂T and $(\bar{q}, \bar{u})$ are the so called numerical fluxes which represent an approximation to (q, u) on ∂T.

The Numerical Fluxes

The perhaps simplest and most natural choice of numerical fluxes is to set

$$(\bar{q}, \bar{u}) = \frac{1}{2}\left[(q^h, u^h)|_{T_i} + (q^h, u^h)|_{T_j}\right] \quad \text{on } \Gamma_{\text{ij}}.$$

We see that this option, which was considered by Bassi and Rebay in [6], does not involve a penalty term and simply computes the fluxes by taking the average of the functional limits on each side of the inter-element interfaces $\mathit{\Gamma}_{ij}$. Though this option seems attractive, the lack of a penalty term renders the method unstable and may lead to a singular discretization matrix on certain grids. It is therefore clear that the stabilization of the DG methods via the inclusion of a penalty term is crucial. In fact, without it, not only stability is affected, but convergence is degraded or lost [5].

To define the numerical fluxes that will be used in this paper, it is convenient to introduce, for $q \in Q^h$, $u \in U^h$, and $x \in \mathit{\Gamma}_{ij}$, the mean value operators

$$\{u\}(x) = \frac{1}{2}(u_i(x) + u_j(x)),$$
$$\{q\}(x) = \frac{1}{2}(q_i(x) + q_j(x)),$$

and the associated jump operators

$$[u]\,(x) = \frac{1}{2}(u_i(x) - u_j(x))n_{ij},$$
$$[q]\,(x) = \frac{1}{2}(q_i(x) - q_j(x)) \cdot n_{ij}.$$

Here $(q_k, u_k) = (q, u)|_{T_k}$ and n_{ij} is the unit normal on $\mathit{\Gamma}_{ij}$ pointing from T_i to T_j. We shall employ the numerical fluxes associated with the method of Brezzi et al. [7], which are

$$\bar{u} = \{u^h\}, \quad \bar{q} = \{q^h\} - \eta[u^h]. \tag{5}$$

These numerical fluxes have been analyzed in [8] in the wider context of LDG (Local Discontinuous Galerkin) methods, and gives a stable, convergent method when $\eta = \mathcal{O}(1/h)$. While there are many other numerical fluxes that has been proposed for DG methods, see e.g., [5], we have chosen to use the Brezzi fluxes (5) because they are simple, stable, and consistent, and give the same rate of convergence (at least for elliptic problems with smooth coefficients) as more elaborate DG methods.

The Primal Formulation

The need to construct approximation spaces for both the potential variable and the velocity variable leads to a relatively large number of degrees of freedom per element. However, it is standard procedure in the literature on DG methods to eliminate the velocity variable from the discretized equations. This elimination leads to the primal formulation:

$$B^h(u^h, v) = \int_\Omega fv \,\mathrm{d}x, \quad \forall v \in U^h, \tag{6}$$

where the primal form $B^h(\cdot,\cdot)$ is defined by

$$\begin{aligned} B^h(u^h, v) := \textstyle\int_\Omega \nabla u^h \cdot a\nabla v \,\mathrm{d}x + \textstyle\int_{\partial\mathcal{T}}([\bar{u} - u^h] \cdot \{a\nabla v\} + \{\bar{q}\} \cdot [v]) \,\mathrm{d}s \\ + \textstyle\int_{\partial\mathcal{T}\setminus\partial\Omega}(\{\bar{u} - u^h\}[a\nabla v] + [\bar{q}]\{v\}) \,\mathrm{d}s, \end{aligned} \tag{7}$$

and $\bar{q} = \bar{q}(u^h, q^h)$ is defined with the understanding that q^h satisfies

$$\begin{aligned} -\textstyle\int_\Omega a^{-1}q^h \cdot p \,\mathrm{d}x = \textstyle\int_\Omega \nabla u^h \cdot p \,\mathrm{d}x + \textstyle\int_{\partial\mathcal{T}}[\bar{u} - u^h] \cdot \{p\} \,\mathrm{d}s \\ + \textstyle\int_{\partial\mathcal{T}\setminus\partial\Omega}\{\bar{u} - u^h\}[p] \,\mathrm{d}s. \end{aligned} \tag{8}$$

If the unknowns associated with the velocity variable q^h are numbered sequentially, element by element, then the matrix block that stems from the term on the left hand side of (8) becomes block diagonal. This allows us to perform a Schur-elimination of the discretization matrix to give the reduced form corresponding to $B^h(\cdot,\cdot)$ at a low cost. Thus, to compute u^h using the primal formulation, we eliminate first the velocity variable by Schur-elimination. The next step is to solve (6) for u^h. Finally one obtains an explicit expression for the fluxes by back-solving for q^h in (8).

For the numerical fluxes considered in this paper, we have $\bar{u} = \{u^h\}$. Thus, since $\bar{q}$ is conservative, i.e., unit valued on $\partial\mathcal{T}$, the integral over $\partial\mathcal{T}\setminus\partial\Omega$ in $B^h(u^h, v)$ vanishes, and the primal form reduces to

$$B^h(u^h, v) := \int_\Omega \nabla u^h \cdot a\nabla v \,\mathrm{d}x - \int_{\partial\mathcal{T}}([u^h] \cdot \{a\nabla v\} - \{\bar{q}\} \cdot [v]) \,\mathrm{d}s. \tag{9}$$

Finally, inserting $\bar{q} = \{q^h\} - \eta[u^h]$ into (9) gives

$$B^h(u^h, v) = \int_\Omega \nabla u^h \cdot a\nabla v \,\mathrm{d}x - \int_{\partial\mathcal{T}} [u^h] \cdot \{a\nabla v\} - (\{q^h\} - \eta[u^h]) \cdot [v] \,\mathrm{d}s. \tag{10}$$

A rigorous analysis of the primal form (10) in the case of polynomial elements can be found in [5]. There it was shown that the bilinear form (10) is bounded and stable, provided that the stabilizing coefficient η is chosen sufficiently large. Hence, the same type of constraint applies to η either we formulate the DG method using the mixed formulation (3)–(4) or the primal formulation (6) and (8) using the primal form (10).

2.2 Finite Element Methods vs. Discontinuous Galerkin Methods

The standard conforming FE discretization of (1) approximates the solution in a finite dimensional subspace V^h of $H^1(\Omega)$. Though $H^1(\Omega)$ is not in general embedded in

$C(\bar{\Omega})$, the discrete FE approximation spaces are. This implies in particular that the corresponding FE methods approximates a possible irregular solution with a continuous one. This continuity assumption can be relaxed, as the non-conforming FE methods do, but they still restrain the solution by putting explicit restrictions on the approximation space. This is in a sense the main difference between FE methods and the DG methods which impose continuity implicitly in a weak sense.

In order to clarify the differences, or perhaps rather the similarities, between FE methods and DG methods for equation (1), we first review the concept behind FE methods. In the standard FE formulation of (1) we define a finite dimensional subspace $U^h \subset H^1(\Omega) \cap C(\bar{\Omega})$ and seek $u \in U^h_D(\Omega) = \{u \in U^h : u = 0 \text{ on } \Gamma_\mathrm{D}\}$ such that

$$\int_\Omega (\nabla u^h)^T a \nabla v \,\mathrm{d}x = \int_\Omega f v \,\mathrm{d}x \quad \forall v \in U^h_D.$$

Now, since $U^h \subset C(\bar{\Omega})$ we know that u^h is continuous. Hence, it makes sense to let $\bar{u} = u^h$ in (7). We then deduce that the primal form (7) reduces to

$$B^h(u^h, v) = \int_\Omega (\nabla u^h)^T a \nabla v \,\mathrm{d}x + \int_{\partial\mathcal{T}\setminus\Omega} [\bar{q}]\{v\}. \tag{11}$$

Thus, if the numerical flux $\bar{q}$ is conservative, i.e. if $\bar{q}$ is single valued on $\partial\mathcal{T}\setminus\Omega$, then the last term on the right hand side of (11) vanishes. Thus, for any approximation space Q^h, the primal formulation of DG methods with a conservative numerical flux for the velocity variable and an approximation space $U^h \subset C(\bar{\Omega})$ reduces to the standard FE variational formulation.

Similarly, in mixed FE methods one seeks a solution (q^h, u^h) of the elliptic problem (1) in a finite dimensional subspace $Q^h_N \times U^h_D$ of $H(\mathrm{div}, \Omega) \times \mathrm{L}^2(\Omega)$. The subscripts N and D indicate that functions in Q^h_N and U^h_D satisfy the homogeneous Neumann and Dirichlet conditions on Γ_N and Γ_D respectively. The mixed FE solution is defined by the following mixed formulation:

$$\begin{aligned} \int_\Omega a^{-1} q^h \cdot p \,\mathrm{d}x &= \int_\Omega u^h \nabla \cdot p \,\mathrm{d}x, && \forall p \in Q^h_N, \\ \int_\Omega \nabla \cdot q^h v \,\mathrm{d}x &= \int_\Omega f v \,\mathrm{d}x, && \forall v \in U^h, \end{aligned}$$

where n is the outward unit normal on $\partial\Omega$.

For many standard mixed FE methods for equation (1), such as the Raviart–Thomas method [15], the approximation space for the velocity consists of functions that are continuous across the interfaces Γ_{ij} in the direction of the coordinate unit normal n_{ij}. For this type of methods we have $\int_{\Gamma_{ij}} [q] ds = 0$ for all $\Gamma_{ij} \subset \Gamma$ and $q \in Q^h$. Thus, by setting $\bar{q} = q^h$ on Γ and $\bar{q} \cdot n = 0$ on Γ_N we find that the second equation above transforms, upon integration by parts, to equation (4). Moreover, if the numerical flux $\bar{u}$ for the potential is single-valued on Γ, then the first equation in the mixed formulation coincides with equation (3). This shows that also mixed FE methods for equation (1) can be viewed as special DG methods for which the numerical fluxes are determined by continuity conditions imposed on the approximation spaces. We may therefore view the DG methods as special FE methods that impose weak continuity of the numerical solution without putting explicit constraints on the approximation space.

3 Multiscale Methods for Elliptic Problems

Many areas of science and engineering face the problem of unresolvable scales. For instance, in porous media flows the permeability of the porous medium is rapidly oscillatory and can span across many orders of magnitude across short distances. By elementary considerations it is impossible to do large scale numerical simulations on models that resolve all pertinent scales down to, e.g., the scale of the pores. The standard way of resolving the issue of unresolvable scales is to build coarse scale numerical models in which small scale variations in the coefficients of the governing differential equations are homogenized and upscaled to the size of the grid blocks. Thus, in this approach small scale variations in the coefficients are replaced with some kind of effective properties in regions that correspond to a grid block in the numerical model.

Multiscale methods have a different derivation and interpretation. In these methods one tries to derive coarse scale equations that incorporate the small scale variations in a more consistent manner. Here we present three different types of multiscale methods; the multiscale FE method (MsFEM) developed by Hou and Wu [12], the mixed MsFEM developed by Chen and Hou [9], and a new class of multiscale DG methods (MsDGM). In these multiscale methods one does not alter the differential coefficients, but instead one constructs coarse scale approximation spaces that reflect subgrid structures in a way which is consistent with the local property of the differential operator. They are therefore more amenable to mathematical analysis, and provides a more flexible approach to solving partial differential equations with multiple scales.

One of the motives for using multiscale methods is reduced complexity. Hence, by introducing a rigorous mathematical formalism where one derives a coarse grid model in which subgrid oscillations in the elliptic coefficients are handled in a mathematical consistent manner, one aims toward a reward in terms of computational efficiency. The computational complexity of the methods proposed below scales linearly with the number of cells in the subgrid model. Hence, the complexity is comparable to the (theoretical) complexity of solving the full system of equations at the subgrid level using an efficient multigrid method. Thus, for these methods it appears that we do not gain much. However, for multiscale problems there are additional considerations.

First, in equations of the form (1) that arise from flows in porous media, the elliptic coefficient function $a(x)$ can vary more than 10 orders of magnitude across short distances. With this extreme span of scales it can be very difficult to obtain linear complexity, or even convergence, using multigrid methods, in particular for linear systems that arise from mixed FE methods. In multiscale methods, like the MsFEM, variations in the coefficients that occur on a subgrid scale appear in the corresponding linear system only as quantities that are integrated over coarse grid blocks. This implies that the impact of the oscillating coefficients on the condition number of the linear system is less severe than for the associated linear system at the subgrid level. In other words, by using a multiscale method we are implicitly doing a preconditioning of the system of subgrid equations. As a consequence, the proposed multiscale methods can be used to obtain quite accurate solutions at the subgrid level for elliptic problems with a range of scales that push the limits, or go beyond the capabilities of multigrid methods.

Another arena where the multiscale methods outlined below can prove useful, also from a computational complexity point of view, is multiphase flow simulation. In a sequential IMPES (IMplicit Pressure Explicit Saturation) formulation of the equations governing incompressible flows in porous media, the pressure equation is basically elliptic, and is non-linearly coupled with a set of (fluid) transport equations. This implies that the pressure equation must be solved repeatedly during a multiphase flow simulation. Fortunately, when simulating fluid flows, for instance flow of oil and water in a heterogeneous oil reservoir, the pressure equation is only weakly coupled to the transport equations. This means that the flow velocity field varies slowly in time away from the propagating saturation front. In such situations the base functions for the proposed multiscale methods need only be generated once, or perhaps a few times during the simulation [1]. In other words, all computations at the subgrid level become part of an initial preprocessing step.

3.1 The Multiscale Finite Element Method

We associate with each element T a set of functions $\mu_T^k \in H^{1/2}(\partial T)$ that play the role of Dirichlet boundary conditions. We then define corresponding multiscale base functions ϕ_T^k by

$$\int_T \nabla\phi_T^k \cdot a\nabla v \,\mathrm{d}x = 0, \quad \forall v \in H_0^1(T), \tag{12}$$

and the associated boundary conditions

$$\begin{aligned} \phi_T^k &= \mu_T^k, && \text{on } \partial\mathrm{T}\backslash\partial\Omega, \\ \phi_T^k &= 0, && \text{on } \Gamma_\mathrm{D} \cap \partial\mathrm{T}, \\ -a\nabla\phi_T^k \cdot n &= 0, && \text{on } \Gamma_\mathrm{N} \cap \partial\mathrm{T}. \end{aligned}$$

To ensure that the base functions are continuous, and hence belong to $H^1(\Omega)$, we require that $\mu_{T_i}^k = \mu_{T_j}^k$ on all non-degenerate interfaces Γ_{ij}. The MsFEM now seeks $u^{\mathrm{ms}} \in U^{\mathrm{ms}} = \mathrm{span}\{\phi^\mathrm{k} : \phi^\mathrm{k} = \sum_\mathrm{T} \phi_\mathrm{T}^\mathrm{k}\}$ such that

$$\int_\Omega \nabla u^{\mathrm{ms}} \cdot a\nabla v \,\mathrm{d}x = \int_\Omega f v \,\mathrm{d}x \quad \forall v \in U^{\mathrm{ms}}. \tag{13}$$

Since the base functions are determined by the homogeneous equation (12), it is clear that the properties of the approximation space U^{ms}, and hence of the accuracy of the multiscale solution u^{ms}, is determined by the boundary data μ_T^k for the multiscale base functions ϕ_T^k.

In [12, 13] it was shown using homogenization theory that for elliptic problems in two dimensions with two-scale periodic coefficients, the solution u^{ms} tends to the correct homogenized solution in the limit as the scale of periodicity tends to zero, $\epsilon \rightarrow 0$. For a positive scale of periodicity, a relation between ϵ and the discretization scale h was established for linear boundary conditions. Moreover, using multiscale expansion of the base functions they showed that with linear boundary conditions μ_T^k at the interfaces, the resulting solution exhibits a boundary layer near the cell boundaries and satisfies

$$\|u - u^{\mathrm{ms}}\|_{L^2(\Omega)} = \mathcal{O}(h^2 + \epsilon/h).$$

This shows that when ϵ and h are of the same order, a large resonance error is introduced. To reduce the resonance effect, which is caused by improper boundary conditions, Hou and Wu introduced also an oversampling technique motivated by the observation that the boundary layer has a finite thickness of order $O(\epsilon)$. However, for further details about this oversampling technique we refer the reader to the article by Hou and Wu [12].

3.2 The Mixed Multiscale Finite Element Method

For each interface Γ_{ij}, define a Neumann boundary condition $\nu_{ij} \in H^{-1/2}(\Gamma_{ij})$ with $\int_{\Gamma_{ij}} \nu_{ij}\,\mathrm{d}s = 1$. Furthermore, for each interface let the corresponding base function ψ_{ij} for the mixed MsFEM be defined by

$$\begin{aligned} \psi_{ij} &= -a\nabla\phi_{ij}, \quad \text{in } \mathrm{T_i} \cup \mathrm{T_j}, \\ \nabla\cdot\psi_{ij} &= \begin{cases} |T_i|^{-1} & \text{in } \mathrm{T_i}, \\ -|T_j|^{-1} & \text{in } \mathrm{T_j}, \end{cases} \end{aligned} \tag{14}$$

and the following boundary conditions:

$$\begin{aligned} \psi_{ij}\cdot n_{ij} &= \nu_{ij}, && \text{on } \Gamma_{\mathrm{ij}}, \\ \phi_{ij} &= 0, && \text{on } \partial(\mathrm{T_i}\cup\mathrm{T_j})\cap\Gamma_{\mathrm{D}}, \\ \psi_{ij}\cdot n &= 0, && \text{on } \partial(\mathrm{T_i}\cup\mathrm{T_j})\backslash(\Gamma_{\mathrm{ij}}\cup\Gamma_{\mathrm{D}}). \end{aligned}$$

Here n_{ij} is the coordinate unit normal to Γ_{ij} pointing from T_i to T_j and n is the outward unit normal on $\partial(T_i \cup \Gamma_{ij} \cup T_j)$. We now define $Q^{\mathrm{ms}} = \mathrm{span}\{\psi_{\mathrm{ij}} : \Gamma_{\mathrm{ij}} \subset \Gamma\}$ and seek $(q^{\mathrm{ms}}, u) \in Q^{\mathrm{ms}} \times \mathcal{P}_0(\mathcal{T})$ which solves

$$\begin{aligned} \int_\Omega a^{-1}q^{\mathrm{ms}}\cdot p\,\mathrm{d}x &= \int_\Omega u\,\nabla\cdot p\,\mathrm{d}x, && \forall p \in Q^{\mathrm{ms}}, \\ \int_\Omega v\nabla\cdot q^{\mathrm{ms}}\,\mathrm{d}x &= \int_\Omega fv\,\mathrm{d}x, && \forall v \in \mathcal{P}_0(\mathcal{T}). \end{aligned}$$

Again we see that the method is determined by the local boundary conditions for the base functions.

It is also possible to choose the right hand side of the equations (14) differently, and in some cases it would be natural not to do so. For instance, in reservoir simulation the right hand side of equation (1) represent wells and wells give rise to source terms that are nearly singular. For simulation purposes it is important that the velocity field is mass conservative. To this end, the right hand side of equation (14) in the well blocks must be replaced with a scaled source term at the well location, see [1] for further details.

A rigorous convergence analysis for the mixed MsFEM has been carried out in [9] for the case of two-scale periodic coefficients using results from homogenization theory. There it was shown that

$$\|q - q^{\mathrm{ms}}\|_{H(\mathrm{div},\Omega)} + \|u - u^{\mathrm{ms}}\|_{L^2(\Omega)} = \mathcal{O}\left(h + \sqrt{\epsilon/h}\right).$$

Hence, again we see that a large resonance error is introduced when $\epsilon/h = \mathcal{O}(1)$. As for the MsFEM method, the possibility of using oversampling as a remedy for resonance errors was explored, and it was shown that oversampling can indeed be used to reduce resonance errors caused by improper boundary conditions. The need for oversampling strategies to reduce resonance errors is, however, a drawback with the MsFEMs since oversampling leads to additional computational complexity.

3.3 A Multiscale Discontinuous Galerkin Method

We now exploit the relationship between DG methods and FE methods that was established in Sect. 2.2. To derive the MsDGM proposed below, we "merge" first the approximation spaces constructed in the original and mixed MsFEMs to create a pair of approximation spaces for the MsDGM. Thus, select suitable boundary conditions $\mu_T^k \in H^{1/2}(\partial T)$ and $\nu_{ij} \in H^{-1/2}(\Gamma_{ij})$ and define base functions ϕ_T^k and ψ_{ij} by (12) and (14) respectively. The approximation spaces for the MsDGM are then defined by

$$U^{\mathrm{ms}} = \mathrm{span}\{\phi^{\mathrm{k}} : \phi^{\mathrm{k}} = \sum_{\mathrm{T}} \phi^{\mathrm{k}}_{\mathrm{T}}\} \quad \text{and} \quad \mathrm{Q}^{\mathrm{ms}} = \mathrm{span}\{\psi_{\mathrm{ij}} : \Gamma_{\mathrm{ij}} \subset \Gamma\}.$$

Thus, the corresponding DG method, henceforth called the MsDGM, reads:

Find $(q^{\mathrm{ms}}, u^{\mathrm{ms}}) \in Q^{\mathrm{ms}} \times U^{\mathrm{ms}}$ so that for each $T \in \mathcal{T}$ we have

$$\int_T a^{-1} q^{\mathrm{ms}} \cdot p \,\mathrm{d}x = \int_T u^{\mathrm{ms}} \nabla \cdot p \,\mathrm{d}x - \int_{\partial T} \bar{u} p \cdot n_T \,\mathrm{d}s, \qquad \forall p \in Q^{\mathrm{ms}}, \tag{15}$$

$$\int_T q^{\mathrm{ms}} \cdot \nabla v \,\mathrm{d}x = -\int_T f v \,\mathrm{d}x + \int_{\partial T} v \bar{q} \cdot n_T \,\mathrm{d}s, \qquad \forall v \in U^{\mathrm{ms}}. \tag{16}$$

In addition to the selection of boundary conditions for the base functions, this method is determined by the choice of numerical fluxes. As indicated in Sect. 2.1 we limit our study to the numerical fluxes (5) of Brezzi et al.

We observe that MsDGMs have much in common with the mortar FE methods. Indeed, here as in the mortar methods, we construct the base functions locally in a manner which corresponds to the underlying partial differential equation, and glue the pieces together using a weak formulation at the interfaces. The new element here is that we derive the above formulation directly from the original and mixed MsFEMs.

Apart for imposing inter-element continuity weakly, the MsDGMs differ from the MsFEMs by using multiscale approximation spaces for both the velocity variable and the potential variable. Another motive for introducing a new multiscale method for elliptic problems is that the accuracy of MsFEMs solutions can be sensitive to the boundary conditions for the base functions. Indeed, previous numerical experience [1, 2] shows that the MsFEMs with simple boundary conditions may produce solutions with poor accuracy when strong heterogeneous features penetrate the inter-element interfaces. Thus, by introducing a MsDGM we aim toward a class of multiscale methods that are less sensitive to resonance errors caused by improper boundary conditions for the multiscale basis functions.

3.4 On the Selection of Boundary Conditions for the MsFEM

Consider the following homogeneous boundary value problem,

$$\begin{aligned} (a(x)u_x)_x &= f, \qquad && \text{in } \Omega = (0,1), \\ u &= 0, \qquad && \text{on } \partial\Omega = \{0,1\}. \end{aligned} \tag{17}$$

and let $\mathbf{x} = \{x_i : 0 = x_0 < x_1 < \ldots < x_N = 1\}$ be a corresponding set of finite element nodal points. Now let $V = H_0^1(\Omega \backslash \mathbf{x})$ and define

$$U = \{u \in H_0^1(\Omega) : (a(x)u_x)_x = 0 \text{ weakly on } \Omega\backslash\mathbf{x},\ u = 0 \text{ on } \partial\Omega\}.$$

Then it is easy to see that $H_0^1(\Omega) = U + V$ and that U and V are orthogonal with respect to the energy norm, i.e.,

$$a(u,v) := \int_\Omega a(x)u_x v_x \,\mathrm{d}x = 0 \quad \forall u \in U, \forall v \in V.$$

Thus, since U^{ms} coincides with U, it follows from the projection property of the FE method that $u^{\mathrm{ms}} = u_I$ where u_I is the interpolant of the exact solution u on $\mathbf{x}$ in $U^{\mathrm{ms}} = U$. This property is due to the fact that there is no resonance error caused by improper boundary conditions and implies that the conforming MsFEM induces an ideal domain decomposition preconditioner for elliptic problems in one spatial dimension.

In higher dimensions the choice of boundary conditions is no longer insignificant. In fact, the MsFEM may be viewed as an extension operator acting on Γ. Hence, the restriction of the solution u^{ms} to Γ must lie in the space spanned by the boundary conditions for the base functions. To clarify the relation between the approximation properties for the MsFEM and the selection of boundary conditions, we consider the following homogeneous boundary value problem: Find $u \in H_0^1(\Omega)$ such that

$$a(u,v) := \int_\Omega \nabla u \cdot a(x)\nabla v \,\mathrm{d}x = \int_\Omega f v \,\mathrm{d}x, \quad \forall v \in H_0^1(\Omega).$$

Now, let $M = H_0^1(\Omega)|_\Gamma$ and define the following extension operator

$$H : M \to H_0^1(\Omega), \quad a(H\mu, v) = 0, \quad \forall v \in H_0^1(\Omega\backslash\Gamma). \tag{18}$$

This extension operator induces a norm on M defined by $\|\mu\|_M^2 = a(H\mu, H\mu)$. Clearly, by definition we find that U^{ms} is a subspace of $W = H(M)$, the space of generalized harmonic functions. In fact, if $M^{\mathrm{ms}} = \mathrm{span}\{\mu_{\mathrm{T}}^{\mathrm{k}}\}$, then $M^{\mathrm{ms}} \subset M$ and $U^{\mathrm{ms}} = H(M^{\mathrm{ms}})$. Thus, since u^{ms} in (13) is the orthogonal projection of the exact solution u onto U^{ms} with respect to $a(\cdot,\cdot)^{1/2}$, it follows that $\mu^{\mathrm{ms}} = u^{\mathrm{ms}}|_\Gamma$ is the orthogonal projection of $\mu = u|_\Gamma$ onto M^{ms} with respect to $\|\cdot\|_M$. This shows that the MsFEM also defines an orthogonal projection onto the space of interface variables with respect to the relevant norm $\|\cdot\|_M$, and hence induces an optimal coarse solver for non overlapping domain decomposition methods. The properties of the MsFEM as a coarse solver in domain decomposition methods has been further analyzed in [2, 3].

4 Numerical Results

Let $\Omega = (0,1) \times (0,1)$ be the computational domain. A uniform finite element mesh $\mathcal{T}$ is constructed by dividing Ω into $N \times N$ squares. The multiscale methods further subdivide each element into $M \times M$ square elements. A reference solution is computed on the full resolved $NM \times NM$ mesh. The global boundary conditions are specified by setting $\Gamma_D = \partial\Omega$ and $\Gamma_N = \emptyset$.

We will test six methods, three monoscale numerical methods and three multiscale methods. The first monoscale method is the FE method (FEM) on quadrilateral grids with bilinear basis functions. The second method is the Raviart–Thomas mixed FEM [15] (MFEM) of lowest order on regular quadrilateral grids. This method uses piecewise constant basis functions for the potential and piecewise linear basis functions for the velocity. The last monoscale method is the DG method (DGM) which uses bilinear basis functions for potential and linear basis functions for the velocity. These monoscale methods will be compared with their multiscale variants defined in Sect. 3, i.e., with the MsFEM, the mixed MsFEM (MsMFEM), and the MsDGM. For these multiscale methods we use the FEM with bilinear basis functions to compute the base functions that span the approximation space U^{ms} for the potential variable u and the Raviart–Thomas mixed FEM to compute the basis functions that span the approximation space Q^{ms} for the velocity variable q. Finally, the reference solution is computed using the Raviart–Thomas mixed FEM. We assess the accuracy of the tested methods with the weighted error measures

$$E(u_h) = \frac{\|u_h - u_r\|_2}{\|u_r\|_2}, \qquad E(q_h) = \frac{1}{2}\left(\frac{\|q_h^x - q_r^x\|_2}{\|q_r^x\|_2} + \frac{\|q_h^y - q_r^y\|_2}{\|q_r^y\|_2}\right).$$

Here $\|\cdot\|_2$ is the $L^2(\Omega)$-norm, the subscript h denotes computed solutions, the subscript r refers to the reference solution and the superscripts x and y signifies velocity components. When comparing velocity fields we do not include the FEMs since these methods are not conservative.

4.1 Subscale Oscillations

We apply the methods to equations (1) with

$$a(x,y) = \frac{2 + P\sin(2\pi x/\epsilon)}{2 + P\sin(2\pi y/\epsilon)} + \frac{2 + P\sin(2\pi y/\epsilon)}{2 + P\cos(2\pi x/\epsilon)},$$
$$f(x,y) = 2\pi^2 \cos(\pi x)\cos(\pi y).$$

This type of coefficients $a(x,y)$ give rise to spurious oscillations in the velocity field, and the source term $f(x,y)$ exerts a low frequent force. We shall fix $P = 1.8$, $NM = 512$ and $\epsilon = 1/64$ for our numerical test cases. We thus get significant subgrid variation for $N < 64$ while the resonance is greatest at $N = 64$. When $N > 64$ the characteristic scale of variation is resolved by the coarse mesh and the use of multiscale methods are no longer necessary.

Table 1. Potential errors for oscillatory coefficients. For the DGM and the MsDGM, the numbers presented correspond to the choice of η that gave the smallest error

N	M	FEM	MFEM	DGM	MsFEM	MsMFEM	MsDGM
8	64	0.9355	1.150	0.6905	0.1161	0.3243	0.08239
16	32	1.043	1.100	0.5674	0.03845	0.1733	0.03776
32	16	1.072	1.086	0.4998	0.02862	0.1127	0.06817
64	8	0.7119	0.712	0.7117	0.04422	0.1508	0.1269

Table 2. Relative velocity errors for oscillatory coefficients. For the DG methods the numbers presented correspond to the choice of η that gave the smallest error

N	M	MFEM	DGM	MsMFEM	MsDGM
8	64	0.5533	0.6183	0.2985	0.4291
16	32	0.5189	0.5388	0.2333	0.2842
32	16	0.5093	0.5144	0.2377	0.2579
64	8	0.5058	0.5079	0.2866	0.3177

Table 1 shows errors $E(u^h)$ in the potential for all the methods. We see that none of the monoscale methods perform particularly well, as they cannot pick up subgrid variations in the coefficients, but the DGM is somewhat more accurate than the other methods. The multiscale methods, on the other hand, generate quite accurate potential fields. The MsFEM is most accurate here, but MsDGM is nearly as accurate, and for very coarse meshes it is the most accurate method. The least accurate multiscale method is the MsMFEM. This is probably due to the fact that piecewise constant functions are used to approximate the potential. The results shown in Table 2 demonstrate that the monoscale methods tend to give more accurate velocity fields than potential fields, but we still see that the multiscale methods give much higher accuracy. We observe also that for this test case the MsMFEM gives more accurate velocity fields than the MsDGM.

The accuracy of the DG methods depend on the parameter η in (5). The results in Table 1 and Table 2 correspond to the value of $\eta = \eta^*$ that produced the best solutions. Since we do not know a priori what η^* is, it is natural to ask how the error behaves as a function of η, and, in particular, how sensitive the DG methods are to the penalty parameter η. We have therefore plotted $E(u_h)$ and $E(q_h)$ for both the DG method and the MsDGM in Figs. 1 and 2 as functions of η. We see that the DG method only converges for a succinct choice of η, except for $N = 64$. In contrast, the MsDGM converges with good accuracy for sufficiently large η. These plots thus demonstrate that; (1): for elliptic problems with oscillating coefficients the MsDGM is less sensitive to η than the monoscale DG methods, and (2): the convergence behavior for the MsDGM seem to be in accordance with the convergence theory for DG methods for elliptic problems with smooth coefficients.

4.2 Random Coefficients

In the second experiment the elliptic coefficient function $a(x, y)$ take random values in the interval $(0, 1)$. Thus, for each grid cell in the fine mesh the value of a is selected at random from this interval. We now only consider the multiscale methods, and compare the MsDGM with the MsFEM and the MsMFEM. The corresponding errors are shown in Table 3. We observe that for this test case the MsDGM generates the by far most accurate potential field, in fact, by almost an order of magnitude. The MsMFEM still produces the most accurate velocity field, but the MsDGM produces a velocity field with comparable accuracy. These results are representative for the results we obtained for a variety of different random coefficient functions. Again the parameter η for the MsDGM numerical flux function $\bar{q}$ was attempted optimized in order to give best results.

Table 3. Potential errors $E(u_h)$ (left) and Velocity errors $E(q_h)$ (right) for an elliptic problem with random coefficients

N	M	MsFEM	MsMFEM	MsDGM	MsMFEM	MsDGM
8	64	0.4458	0.3290	0.1394	0.3375	0.4849
16	32	0.3827	0.1839	0.07163	0.2851	0.3716
32	16	0.3589	0.1510	0.03766	0.2941	0.3472
64	8	0.3335	0.2074	0.02539	0.3346	0.3624

The results in Table 3 indicate that the MsDGM may be more robust than the MsFEM. Unfortunately, for this case the MsDGM was more sensitive to η than what we observed in Sect. 4.1, and choosing η too large can deteriorate the convergence, see Fig. 3. However, the optimal η for potential (~ 13, ~ 34, ~ 83 and ~ 188 for

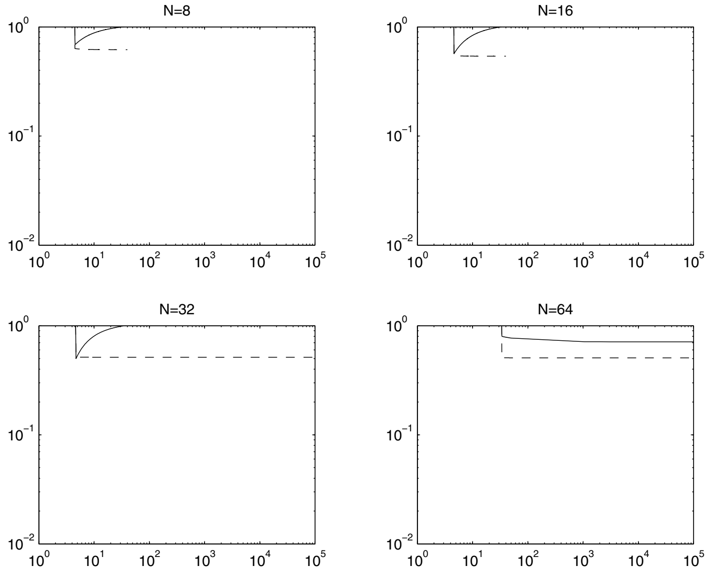


Fig. 1. Errors induced by the DG method as functions of η. The solid line is the potential error, and the dashed line is the velocity error

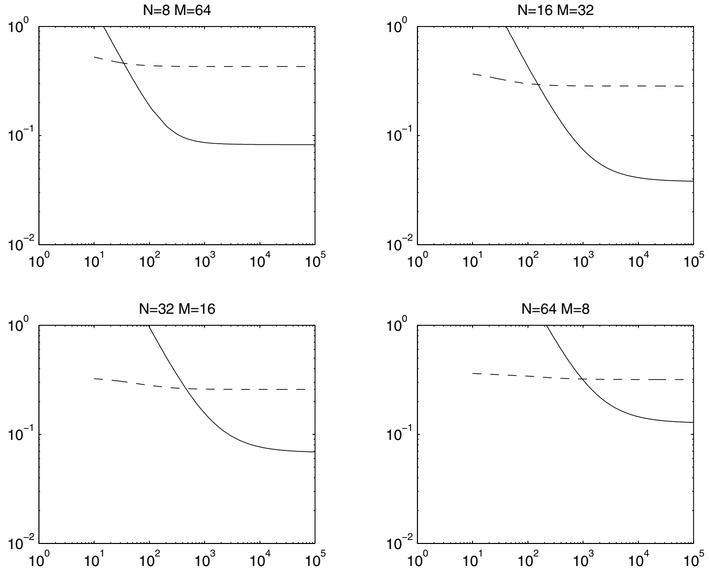


Fig. 2. Errors induced by the MsDGM as functions of η. The solid line is the potential error, and the dashed line is the velocity error. Note that the error decreases monotonically as η increases, and that the method converges for $\eta > \mathcal{O}(1/h)$

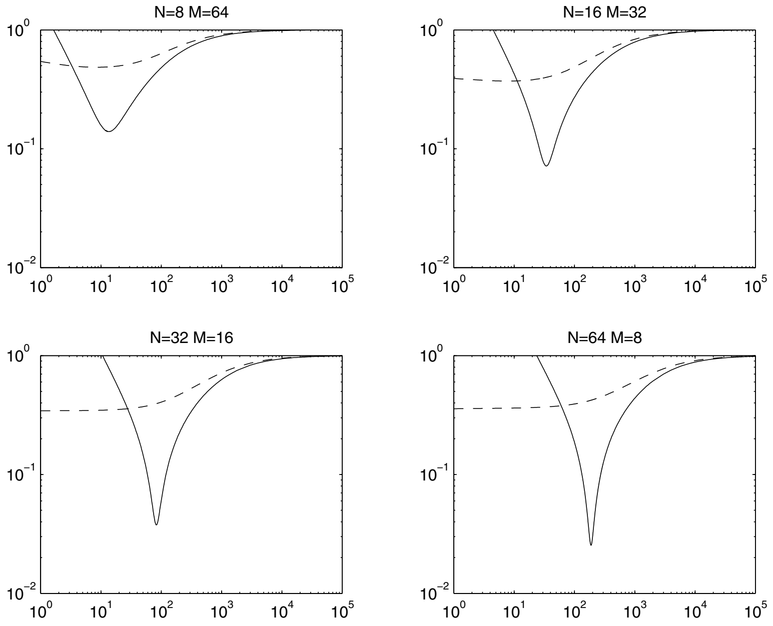


Fig. 3. Logarithmic plot of errors for the MsDGM for random coefficients. The plots should be compared with the results depicted in Fig. 2

$h = 8,\ 16,\ 32$ and 64 respectively) still scales like $\mathcal{O}(1/h)$. This suggest that good accuracy should be obtained by choosing $\eta \sim \alpha/h$ for some fixed $\alpha \sim \mathcal{O}(1)$.

5 Concluding Remarks

In this paper, we have used approximation spaces for two different multiscale finite element methods in order to develop a multiscale discontinuous Galerkin method for elliptic problems with multiple scale coefficients. Unlike the multiscale finite element methods, the multiscale discontinuous Galerkin method introduced in this paper provides detailed solutions for both velocity and potential that reflect fine scale structures in the elliptic coefficients. This makes the multiscale discontinuous Galerkin method an attractive tool for solving, for instance, pressure equations that arise from compressible flows in heterogeneous porous media. Indeed, in compressible flow simulations it is not sufficient to resolve the velocity field well, an accurate pressure field is also needed.

Numerical comparisons with both monoscale- and multiscale methods have been given. The results show that monoscale numerical methods are inadequate when it comes to solving elliptic problems with multiple scale solutions. We have further demonstrated that the multiscale discontinuous Galerkin method produce solutions with comparable or higher accuracy than solutions produced by corresponding multiscale finite element methods. To summarize the results for the multiscale methods, we plot errors for all the multiscale methods in Fig. 4. This figure shows that the velocity solutions obtained with the multiscale discontinuous Galerkin method have comparable accuracy with the velocity solutions obtained with the corresponding mixed multiscale finite element method. The potential solutions produced by the multiscale discontinuous Galerkin method, on the other hand, are equally or more accurate than the potential solutions obtained with both of the multiscale finite element methods.

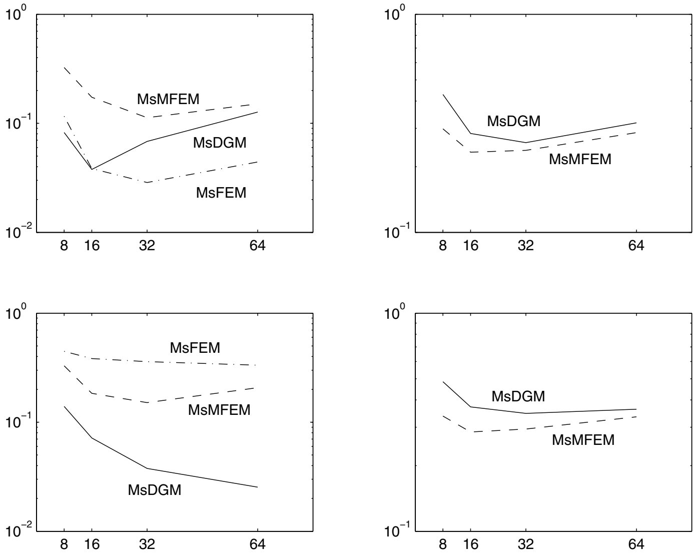


Fig. 4. Plot of the errors as functions of grid size for oscillatory (top) and random coefficients (bottom) respectively. The left figures show potential errors, and the right figures show the velocity errors. The MsDGM is the solid line, the MsMFEM is the dashed line, and the MsFEM is the dashed and dotted line

In the present paper we have not provided any convergence theory, but it is likely that convergence results can be obtained using results from homogenization theory in conjunction with the convergence theory for discontinuous Galerkin methods for elliptic problems with smooth coefficients. However, since the discontinuous Galerkin methods appear to give comparable accuracy to the multiscale mixed finite element methods for which error estimates based on the homogenization theory have been established, one can expect the discontinuous Galerkin methods to enjoy similar error estimates. We have also shown that the multiscale discontinuous Galerkin method appears to converge for values of the penalty parameter η in the numerical flux function that is in accordance with the convergence theory for Galerkin methods for elliptic problems with smooth coefficients. A rigorous convergence analysis of the multiscale discontinuous Galerkin methods is a topic for further research.

Acknowledgments

We would like to thank Kenneth H. Karlsen and Ivar Aavatsmark for useful comments. The second author would also like to thank the Research Council of Norway for financial support.

References

1. J. Aarnes, On the use of a mixed multiscale finite element method for greater flexibility and increased speed or improved accuracy in reservoir simulation, Multiscale Model. Simul., **2**, 421–439, (2004).
2. J. Aarnes, Efficient domain decomposition methods for elliptic problems in heterogeneous porous media, to appear in Comput. Vis. Science.
3. J. Aarnes, T. Y. Hou, Multiscale domain decomposition methods for elliptic problems with high aspect ratios, Acta Math. Appl. Sinica, **18**, 63–76, (2002).
4. D. N. Arnold, An Interior Penalty Finite Element Method with Discontinuous Elements, SIAM J. Numer. Anal., **19**, 742–760, (1982).
5. D. N. Arnold, F. Brezzi, B. Cockburn, L. D. Marini, Unified Analysis of Discontinuous Galerkin Methods, SIAM J. Numer. Anal., **39**, 1749–1779 (2002).
6. F. Bassi, S. Rebay, A High-Order Accurate Discontinuous Finite Element Method for the Numerical Solution of the Compressible Navier Stokes Equations, J. Comput. Phys., **131**, 267–279, (1997).
7. F. Brezzi, G. Manzini, D. Marini, P. Pietra, A. Russo, Discontinuous Finite Elements for Diffusion Problems, in Atti Convegno in onore di F. Brioschi (Milan 1997), Instituto Lombardo, Accademia di Scienze e Lettere, Milan, Italy, 197–217, (1999).
8. P. Castillo, B. Cockburn, I. Perugia, D. Schötzau, An a priori error analysis of the Local Discontinuous Galerkin Method for Elliptic Problems, SIAM J. Numer. Anal., **38**, 1676–1706, (2000).
9. Z. Chen and T. Y. Hou, A mixed multiscale finite element method for elliptic problems with oscillating coefficients, Math. Comp., **72**, 541–576, (2003).
10. B. Cockburn, G. Karniadakis, C.-W. Shu (eds). Discontinuous Galerkin Methods. Theory, Computation and Applications, Lect. Notes Comput. Sci. Eng., **11**, Springer-Verlag, Heidelberg, (2000).
11. D. Douglas, Jr, T. Dupont, Interior Penalty Procedures for Elliptic and Parabolic Galerkin Methods, Lect. Notes Phys., **58**, Springer-Verlag, Berlin, (1976).
12. T. Y. Hou, X.-H. Wu, A Multiscale Finite Element Method for Elliptic Problems in Composite Materials and Porous Media, J. Comput. Phys., **134**, 169–189, (1997).
13. T. Y. Hou, X.-H. Wu, Z. Cai, Convergence of a Multiscale Finite Element Method for Elliptic Problems with Rapidly Oscillating Coefficients, Math. Comp, **68**, 913-943, (1999).
14. J.-L. Lions, Problèmes aux limites non homogènes à donées irrégulières: Une méthod d'approximation, Num. Anal. Part. Diff. Eqn. (C.I.M.E. 2 Ciclo, Ispra, 1967), Edizioni Cremonese, Rome, 283–292, (1968).
15. P. A. Raviart and J. M. Thomas, A mixed finite element method for second order elliptic equations, Mathematical Aspects of Finite Element Methods (I. Galligani and E. Magenes, eds.), Springer–Verlag, Berlin – Heidelberg – New York, 292–315, (1977).

16. M. F. Wheeler, An Elliptic Collocation-Finite Element Method with Interior Penalties, SIAM J. Numer. Anal., **15**, 152–161, (1978).

Discrete Network Approximation for Highly-Packed Composites with Irregular Geometry in Three Dimensions

Leonid Berlyand[1], Yuliya Gorb[2], and Alexei Novikov[2]

[1] Department of Mathematics & Materials Research Institute, Pennsylvania State University, McAllister Bld., University Park, PA 16802, USA,
berlyand@math.psu.edu

[2] Department of Mathematics, Pennsylvania State University, University Park, PA 16802, USA,
gorb@math.psu.edu, anovikov@math.psu.edu

Summary. We introduce a discrete network approximation to the problem of the effective conductivity of a high contrast, densely packed composite in three dimensions. The inclusions are irregularly (randomly) distributed in a host medium. For this class of arrays of inclusions we derive a discrete network approximation for effective conductivity and obtain a priori error estimates. We use a variational duality approach to provide rigorous mathematical justification for the approximation and its error estimate.

Key words: effective conductivity, discrete network, error estimate, variational bounds

1 Introduction

We study composites of highly-conducting identical spherical inclusions embedded into a matrix (host medium) of finite conductivity. Our main objective is to obtain the dependence of the effective conductivity on the irregular (non-periodic or random) geometry of a dense spatial array of inclusions.

Periodic arrays of highly conducting inclusions were analyzed in [12] (for other references see e.g. [6]). For non-periodic arrays of inclusions the geometric connectivity patterns may lead to physical effects, which are not seen in the periodic case. For example, the percolation effects may appear. This issue was addressed in [6] for an analogous two-dimensional problem, where a discrete network approximation for the effective conductivity was developed for irregularly distributed disks (see e.g. [6, 7, 8, 3] for references on other discrete network models). In subsequent work [7] the error estimates of this approximation were obtained. In particular, it was shown there that the approximation and error estimates are valid in the homogenization limit as the radius of the disks tends to zero. The key ingredient in the construction of the error estimates is the $\delta - \mathcal{N}$ close packing condition, where $\mathcal{N}$ is the number of inclusions in the perimeter of a "hole" in the conducting spanning cluster of inclusions [7]. Such "holes" correspond to so-called void spaces in a particulate composite. Loosely speaking, this condition allows for "holes" of size $\mathcal{N}$ in the cluster.

In the present work we further develop the discrete network approach, introduced in [6, 7] for the two dimensional problem, to study the effective conductivity $\widehat{A}$ in three dimensions. Note that the geometry of the three dimensional connectivity patterns is much more complex than in two dimensions. This phenomenon manifests itself in various ways. For example, while the uniqueness of the percolating spanning cluster in two dimensions is immediate, the analogous uniqueness result in three dimensions was a major advance in the early stages of developing mathematical percolation theory [1]. The close packing problem for spheres provides another example. While the periodic array of disks with maximal packing density is unique in two dimensions (hexagonal), there exist two periodic arrays of identical spheres with maximal density (face-centered cubic and hexagonal) in three dimensions [9, 15].

We develop a discrete approximation $\mathcal{I}$ for the effective conductivity $\widehat{A}$ and study the approximation error

$$\text{Error} = |\widehat{A} - \mathcal{I}|.$$

This work contains two main results. The first result concerns an asymptotic relation between the effective conductivity $\widehat{A}$ and the discrete approximation $\mathcal{I}$. We show that $\widehat{A}$ and $\mathcal{I}$ satisfy the following asymptotic relation (Theorem 3):

$$\widehat{A} = \mathcal{I} + O(1), \quad \text{as } \delta \to 0, \tag{1}$$

where the *relative interparticle distance* δ [7] is, roughly speaking, a dimensionless characteristic distance between inclusions in the conducting spanning cluster and $\mathcal{I} = O(|\ln \delta|)$. The second result involves the relative error estimates of the discrete network approximation. In Theorem 4 we prove that

$$\frac{|\widehat{A} - \mathcal{I}|}{\mathcal{I}} \leq \frac{C(\mathcal{D})R}{|\ln \delta|}, \tag{2}$$

for $\delta - \mathcal{D}$ connected distributions of inclusions, where R is the radius of the inclusion. $\mathcal{D}$ is, loosely speaking, the typical relative diameter of a void space or a hole in the conducting cluster if the radius of inclusion is rescaled to be 1 (see Fig. 1). Here the standard definition of a conducting spanning cluster from percolation theory was used with the connectivity defined as follows: two inclusions are connected if the distance between them is less or equal than δR. $C(\mathcal{D})$ is a constant depending on $\mathcal{D}$ for which we obtain an upper bound as an explicit function of $\mathcal{D}$.

The justification of the approximation and its error estimates is based on the variational upper and lower bounds for the effective conductivity $\widehat{A}$. These bounds are obtained by an explicit construction of trial functions for the direct and the dual variational problems. The trial functions are obtained by decomposing the part of the domain complementary to the inclusions (occupied by the conducting matrix) into simple geometric figures: tetrahedra, prisms and hoses. This decomposition is based on a *central projection partition*, introduced in Sect. 3.2.

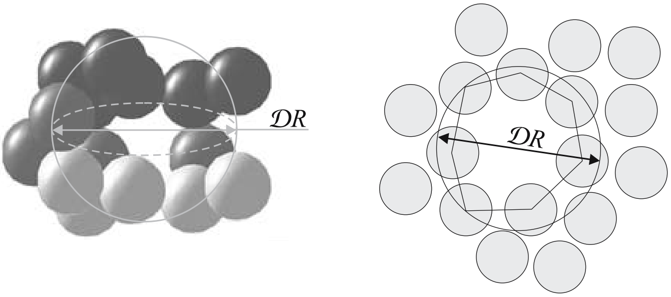


Fig. 1. Holes of the diameter $\mathcal{D}R$ in a composite and 2D cross-section of the part of the composite having a hole

The paper is organized as follows. In Sect. 2 we present a mathematical formulation of the problem. The variational bounds for the effective conductivity $\widehat{A}$ are also given. In Sect. 3 we construct the three dimensional discrete network. In particular, the central projection partition is introduced in Sect. 3.2 and the δ-$\mathcal{D}$ connectedness property of the discrete network is defined in Sect. 3.4. The variational error estimates for the effective conductivity $\widehat{A}$ are then derived in Sect. 4. The trial functions for the lower and upper variational bounds are constructed in Sects. 4.1 and 4.2, respectively. The main results (1) and (2) are proved in Sect. 4.3. In Sect. 5 we present the results of a numerical simulation that show the dependence of $\widehat{A}$ on the volume fraction of the void spaces in a composite. Finally, various technical calculations are included in the Appendices.

2 Formulation of the Problem

Consider a three-dimensional model of a two-phase composite that consists of a matrix of finite conductivity in which a large number of identical, perfectly conducting spherical inclusions are randomly distributed. The composite is modeled by a parallelepiped $Y = [-L_1, L_1] \times [-L_2, L_2] \times [-1, 1]$. The inclusions are modeled by identical non-overlapping balls B_i, $i = 1, \dots N$, of radius R, where N is the number of inclusions. We are concerned with the high concentration regime, that is, the case when the characteristic distance between two neighboring balls is much smaller than their radii. The perforated domain

$$Q = Y \setminus \bigcup_{i=1}^{N} B_i \tag{3}$$

models the matrix of the composite. Let $\partial Q^{\pm} = \{\boldsymbol{x} = (x, y, z) \in \mathbf{R}^3 : \ z = \pm 1\}$ be the upper/lower boundary of the domain Q and $\partial Q_{\text{lat}} = \partial Y \setminus (\partial Q^- \cup \partial Q^+)$ be the lateral boundary of Q.

Let the potential $u(\boldsymbol{x}) = u(x, y, z)$, $\boldsymbol{x} \in \mathbf{R}^3$, be a function from $H^1(Q)$. Introduce the space:

$$V = \left\{ u \in H^1(Q) :\ u(\boldsymbol{x}) = t_i \text{ on } \partial B_i,\ u(\boldsymbol{x}) = \pm 1 \text{ on } \partial Q^{\pm} \right\} \tag{4}$$

where t_i $(i = 1 \dots N)$ are real numbers to be determined in the course of solving the problem, and define a functional:

$$I[u] = \frac{1}{2|Y|} \int_Q |\nabla u|^2 \mathrm{d}\boldsymbol{x} = \frac{1}{16 L_1 L_2} \int_Q |\nabla u|^2 \mathrm{d}\boldsymbol{x}, \quad u \in H^1(Q), \tag{5}$$

where $|Y| = 8L_1L_2$ is the volume of the box Y.

Suppose the function u solves the variational minimization problem:

$$\min_{\widetilde{u} \in V} I[\widetilde{u}] =: I[u]. \tag{6}$$

Then the potential u inside the matrix Q satisfies the Euler-Lagrange equation of (6):

$$\begin{array}{lll} (a) & \triangle u = 0, & \text{in } Q \\ (b) & u(\boldsymbol{x}) = t_i, & \text{on } \partial B_i,\ \ i = 1, \dots, N \\ (c) & u(\boldsymbol{x}) = \pm 1, & \text{on } \partial Q^{\pm} \\ (d) & \displaystyle\int_{\partial B_i} \frac{\partial u}{\partial \boldsymbol{n}} \mathrm{d}\boldsymbol{x} = 0, & i = 1, \dots, N \\ (e) & \displaystyle\frac{\partial u}{\partial \boldsymbol{n}} = 0, & \text{on } \partial Q_{\mathrm{lat}}. \end{array} \tag{7}$$

The effective conductivity $\widehat{a}$ of the composite is defined as the minimum of the functional (5), (6), over the class V given by (4), that is:

$$\widehat{a} = I[u] = \frac{1}{2|Y|} \int_Q |\nabla u|^2 \mathrm{d}\boldsymbol{x}, \tag{8}$$

(see e.g. [4, 5, 11, 13]).

Applying Green's formula to (5) and taking into account (7) we notice that $\widehat{a}$ can also be defined as the total flux per unit length through a horizontal cross-section of the domain. Integrating $\int_Q \triangle u \, \mathrm{d}\boldsymbol{x}$ by parts and using (7a) indicates that the *total fluxes* through the horizontal boundaries ∂Q^- and ∂Q^+ are equal:

$$\int_{\partial Q^+} \frac{\partial u}{\partial \boldsymbol{n}} \mathrm{d}\boldsymbol{x} = - \int_{\partial Q^-} \frac{\partial u}{\partial \boldsymbol{n}} \mathrm{d}\boldsymbol{x}. \tag{9}$$

The integration by parts of $\int_Q u \triangle u \, \mathrm{d}\boldsymbol{x}$ yields

$$\int_{\partial Q^+} \frac{\partial u}{\partial \boldsymbol{n}} \mathrm{d}\boldsymbol{x} = \frac{1}{2} \int_Q |\nabla u|^2 \mathrm{d}\boldsymbol{x}.$$

Therefore for an equivalent definition of the effective conductivity we can take, for example, the total flux through the boundary ∂Q^+ per unit length defined by

$$\widehat{a} = \frac{1}{|Y|}\int_{\partial Q^+} \frac{\partial u}{\partial \boldsymbol{n}}\mathrm{d}\boldsymbol{x} = \frac{1}{|Y|}\int_{\partial Q^+} \frac{\partial u}{\partial z}\mathrm{d}\boldsymbol{x}. \tag{10}$$

For simplicity, we use a rescaled quantity $\widehat{A} = \widehat{a}\,|Y| = 8L_1L_2\widehat{a} = \displaystyle\int_{\partial Q^+} \frac{\partial u}{\partial z}\mathrm{d}\boldsymbol{x}$.

Our goal is to use a variational approach to investigate $\widehat{A}$ when inclusions are close to touching. Besides the direct variational problem (5), (6), (4) we consider a dual variational problem in which $\widehat{A}$ is equivalently defined [10] as a maximum of the functional J:

$$\widehat{A} = \max_{\widetilde{\boldsymbol{v}}\in W} J[\widetilde{\boldsymbol{v}}] =: J[\boldsymbol{v}], \tag{11}$$

where

$$J[\boldsymbol{v}] = -\frac{1}{2}\int_Q \boldsymbol{v}^2\mathrm{d}\boldsymbol{x} + \int_{\partial Q^+} \boldsymbol{v}\cdot\boldsymbol{n}\ \mathrm{d}\boldsymbol{x} - \int_{\partial Q^-} \boldsymbol{v}\cdot\boldsymbol{n}\ \mathrm{d}\boldsymbol{x}, \tag{12}$$

and the class of all fluxes W is given by:

$$W = \left\{\boldsymbol{v}\in \boldsymbol{L}^2(Q) : \boldsymbol{v}(\boldsymbol{x})\cdot\boldsymbol{n} = 0 \text{ on } \partial Q_{\mathrm{lat}},\ \int_{\partial B_i} \boldsymbol{v}\cdot\boldsymbol{n}\ \mathrm{d}\boldsymbol{x} = 0,\ \operatorname{div}\boldsymbol{v} = 0 \text{ in } Q\right\}. \tag{13}$$

The details of the derivation of (11), (12) and (13) can be found for example in [10].

Thus, for any $u \in V$ and $\boldsymbol{v}\in W$ we obtain the following bounds on the effective conductivity $\widehat{A}$:

$$-\frac{1}{2}\int_Q \boldsymbol{v}^2\mathrm{d}\boldsymbol{x} + \int_{\partial Q^+} \boldsymbol{v}\cdot\boldsymbol{n}\ \mathrm{d}\boldsymbol{x} - \int_{\partial Q^-} \boldsymbol{v}\cdot\boldsymbol{n}\ \mathrm{d}\boldsymbol{x} \le \widehat{A} \le \frac{1}{2}\int_Q |\nabla u|^2\mathrm{d}\boldsymbol{x}. \tag{14}$$

The equality in (14) is achieved when $\boldsymbol{v} = \nabla u$.

3 Discrete Network

In this section we construct a discrete network (graph) approximating the continuum problem (10). We use here Keller's observation [12] that the dominant contribution to the effective conductivity comes from the thin gaps (*hoses*) between closely spaced neighboring inclusions, so that the flux outside of these gaps does not change the asymptotics of $\widehat{A}$. We define the notions of neighboring balls and hoses connecting them, in Sect. 3.1. Next we define an algebraic quadratic form:

$$\mathcal{I} = \frac{1}{2}\sum_{\Pi_{ij}} g_{ij}(t_i - t_j)^2, \tag{15}$$

where t_i is the value of the potential in the ball B_i, $(i = 1,\ldots,N)$ and the number g_{ij} is a *specific flux* defined in Sect. 3.3. The sum in (15) is taken over the hoses, defined in Sect. 3.2, connecting the neighboring inclusions. Note that there are hoses connecting inclusions B_i inside the composite with the boundaries of Q. The quadratic form (15) is our discrete network approximation to effective conductivity (8): $\widehat{A} \sim \mathcal{I}$.

In order to determine t_i, $i = 1, \dots, N$ in (15) we set up a *discrete minimization problem* supplemented with appropriate boundary conditions. The obtained minimization problem amounts to solving a system of linear algebraic equations, making it numerically tractable.

Our error estimates of the discrete network rely on the variational bounds (14). Upper and lower trial fields for (14) are constructed using our decomposition of the domain Q into simple geometric regions. This decomposition is obtained by the *central projection partition* which we introduce in Sect. 3.2. Finally, we give useful properties of the discrete network approximation in Sect. 3.4.

3.1 Construction of the Network

Neighbors and necks of the discrete network can be defined using the notion of the *Voronoi diagram* [2] of the domain Q. For the centers of the balls B_i:

$$P = \{\boldsymbol{x}_i \in \mathbf{R}^3,\ i = 1, \dots, N\}, \tag{16}$$

the Voronoi diagram is the partition of the domain Q into non-overlapping *Voronoi cells* $V(\boldsymbol{x}_i)$. Each $V(\boldsymbol{x}_i)$ is the set of all points in $\mathbf{R}^3$ that are closer to $\boldsymbol{x}_i$ than to any other site from P.

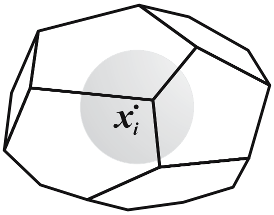


Fig. 2. The Voronoi cell $V(\boldsymbol{x}_i)$

The Voronoi diagram in $\mathbf{R}^3$ can also be defined as follows. Introduce the bisector of two sites $\boldsymbol{x}_i, \boldsymbol{x}_j \in P$, which is the perpendicular plane through the midpoint of the line segment $\boldsymbol{x}_i\boldsymbol{x}_j$. The region $V(\boldsymbol{x}_i)$ of a site $\boldsymbol{x}_i \in P$ is the intersection of half-spaces bounded by bisectors, therefore it is a 3-dimensional convex polyhedron (Fig. 2). The boundary of $V(x_i)$ consists of *faces*, which are the convex polygons, *edges*, which are the line segments formed by intersections of faces, and *vertices*, which are intersections of edges.

Hereafter we assume that P is in *general position* [2], that is, no 5 points lie on a common sphere and no 4 points are cocircular and on a common plane. This assumption implies that each face/edge/vertex is shared by exactly two/three/four Voronoi cells, respectively.

Neighbors are defined to be the balls centered at sites whose Voronoi cells share a common face.

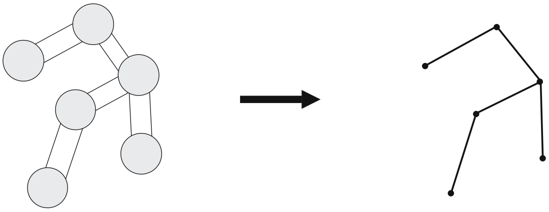


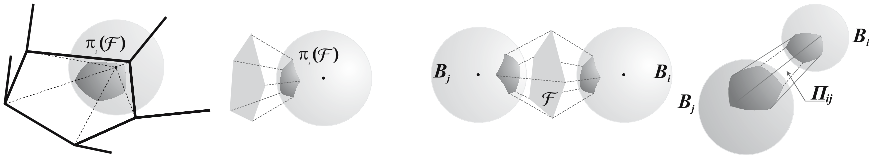
Fig. 3. Balls and hoses are identified with the graph $\mathcal{G}$

For a given array of the balls B_i centered at $\boldsymbol{x}_i$, $i = 1, \ldots, N$, the discrete network is the (Delaunay) graph $\mathcal{G} = (X, E)$, with vertices $X = \{\boldsymbol{x}_i : i = 1, \ldots, K,\ K \geq N\}$ and edges $E = \{e_{ij} : i, j = 1, \ldots, K\}$, connecting each pair of *neighbors* (see Fig. 3). In addition, if one of the Voronoi faces of some site $\boldsymbol{x}_k$ lies on the boundary of the domain Q, then we connect $\boldsymbol{x}_k$ with this boundary by the perpendicular line segment $e_{kk''}$, obtaining a new vertex $\boldsymbol{x}_{k''}$ on the boundary.

Note that $K \geq N$, where N is the number of inclusions, since the graph $\mathcal{G}$ contains the extra vertices $\boldsymbol{x}_{k''}$.

3.2 Central Projection Partition

The *central projection partition* is an elegant algorithm to decompose the domain Q into three types of solids: *hoses, prisms and tetrahedra.*



(a) $\pi_i(\mathcal{F})$, the projection of the Voronoi face $\mathcal{F}$

(b) The bases, $\pi_i(\mathcal{F})$ and $\pi_j(\mathcal{F})$, of the hose Π_{ij} (left). The hose Π_{ij} (right)

Fig. 4.

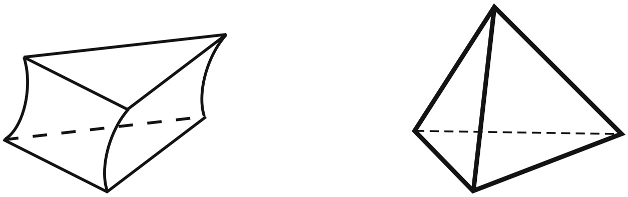
Definition 1. *The central projection is the collection* $\{\pi_i,\ i = 1, \ldots, N\}$ *of maps* $\pi_i : \partial V(\boldsymbol{x}_i) \to \partial B_i$ *given by*

$$\pi_i(\boldsymbol{x}) = R\frac{\boldsymbol{x} - \boldsymbol{x}_i}{|\boldsymbol{x} - \boldsymbol{x}_i|}.$$

Every projection π_i partitions the sphere ∂B_i into non-overlapping curvilinear polygons $\{\pi_i(\mathcal{F})\}$, where $\mathcal{F}$ is a face of the Voronoi cell $V(\boldsymbol{x}_i)$.

For each Voronoi face $\mathcal{F}$ there are two projections $\pi_i(\mathcal{F})$ and $\pi_j(\mathcal{F})$ (one of them is depicted in Fig. 4(a)) onto neighboring spheres ∂B_i and ∂B_j, respectively. These projections $\pi_i(\mathcal{F})$ and $\pi_j(\mathcal{F})$ are the (nonflat) bases of a *hose* Π_{ij} (Fig. 4(b)), a (generalized) cylinder. We remark here that with this construction the edges of a hose are always parallel.

Similarly, the central projections of Voronoi edges and Voronoi vertices give rise to prisms and tetrahedra, respectively.



(a) the prism (b) the tetrahedron

Fig. 5.

In fact, for every Voronoi edge $\mathcal{E}$ there are three projections $\pi_i(\mathcal{E})$, $\pi_j(\mathcal{E})$ and $\pi_k(\mathcal{E})$, which are arcs on the spheres ∂B_i, ∂B_j and ∂B_k, respectively. Connecting the endpoints of the corresponding arcs, we obtain a figure, referred to as a *prism*, shown in Fig. 5(a).

For every Voronoi vertex $\mathcal{V}$ there are four projections $\pi_i(\mathcal{V})$, $\pi_j(\mathcal{V})$, $\pi_k(\mathcal{V})$ and $\pi_m(\mathcal{V})$, which are four points on spheres ∂B_i, ∂B_j, ∂B_k and ∂B_m, respectively. Connecting these points yields a *tetrahedron* (Fig. 5(b)).

The *central projection partition* of Q can alternatively be constructed using four-wise neighbors. We give this alternative construction in Appendix 6.1.

For consistency of the presentation, we introduce the notion of a *quasi-ball*, when we construct the discrete network at the boundary. Suppose that three balls B_m, B_n, B_q, centered at $\boldsymbol{x}_m$, $\boldsymbol{x}_n$ and $\boldsymbol{x}_q$, respectively, lie near the upper boundary ∂Q^+ (see Fig. 6). In order to construct a hose connecting, for example, the ball B_m with ∂Q^+, we consider the reflection $B_{m'}$, centered at $\boldsymbol{x}_{m'}$, of the ball B_m along the plane $z = 1$ and repeat the central projection construction for 4 neighbors B_m, B_n, B_q and $B_{m'}$. The intersection of the hose $\Pi_{mm'}$, connecting the balls B_m and $B_{m'}$, with the upper boundary ∂Q^+ is a curvilinear polygon on ∂Q^+, referred to as a *quasi-ball.* The quasi-ball is centered at the intersection of the line segment $e_{mm'}$, connecting $\boldsymbol{x}_m$ and $\boldsymbol{x}_{m'}$, with ∂Q^+, and denoted by $\boldsymbol{x}_{m''}$. We assume that the quasi-ball has a radius equal to infinity. We assign $+1$ as the value of the potential on this quasi-ball, because it lies on the upper boundary of the composite Q. Prisms $P_{mnm'}$, $P_{m'mq}$, $P_{m'nq}$ and tetrahedron $\Delta_{m'mnq}$ intersected with ∂Q^+ yield the truncated prisms and tetrahedron (which can be obtained by an auxiliary constriction [7]), that we still call the prisms and tetrahedron, respectively, with the values $+1$ of the potential on their

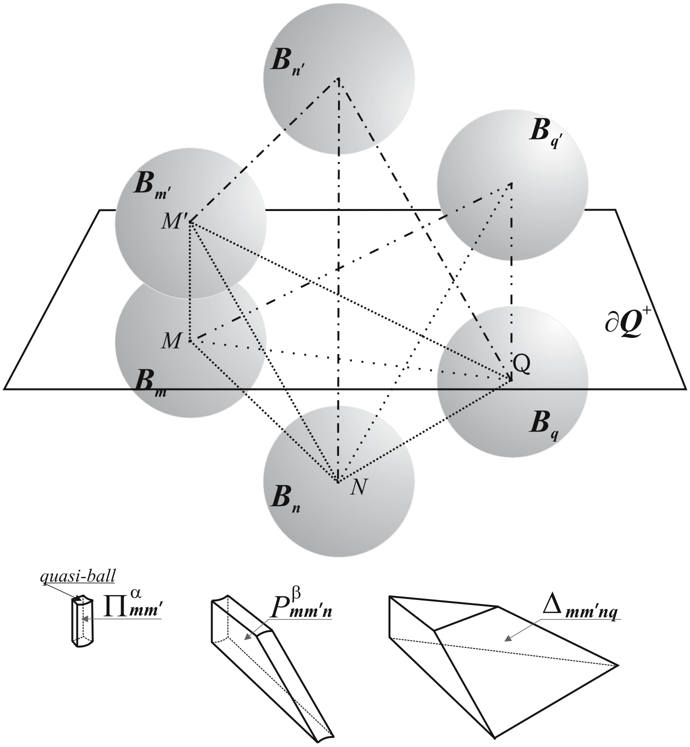


Fig. 6. Auxiliary construction near the upper boundary of Q

upper boundaries (such a tetrahedron and one of the prisms are shown in Fig. 6). The constructions of the quasi-balls on the other boundaries of Q are analogous.

3.3 Discrete Minimization Problem

Using the central projection partition of the matrix Q, it is possible to decompose the value of $\widehat{A}$ (8) into the sum of the Dirichlet's integrals over hoses, prisms and tetrahedra, that is

$$\widehat{A} = \frac{1}{2}\left(\sum_{\Pi_{ij}} \int_{\Pi_{ij}} |\nabla u|^2 \mathrm{d}\boldsymbol{x} + \sum_{P_{ijk}} \int_{P_{ijk}} |\nabla u|^2 \mathrm{d}\boldsymbol{x} + \sum_{\Delta_{ijkm}} \int_{\Delta_{ijkm}} |\nabla u|^2 \mathrm{d}\boldsymbol{x}\right).$$

The asymptotic derivation of the discrete minimization problem is based on three main observations.

First, using [6, 7]:

$$\begin{aligned}\sum_{P_{ijk}} \int_{P_{ijk}} |\nabla u|^2 \mathrm{d}\boldsymbol{x} \quad &\ll \sum_{\Pi_{ij}} \int_{\Pi_{ij}} |\nabla u|^2 \mathrm{d}\boldsymbol{x}, \\ \sum_{\Delta_{ijkm}} \int_{\Delta_{ijkm}} |\nabla u|^2 \mathrm{d}\boldsymbol{x} &\ll \sum_{\Pi_{ij}} \int_{\Pi_{ij}} |\nabla u|^2 \mathrm{d}\boldsymbol{x},\end{aligned} \tag{17}$$

the total Dirichlet integral is approximately the integral over the hoses:

$$\widehat{A} \sim \frac{1}{2} \sum_{\Pi_{ij}} \int_{\Pi_{ij}} |\nabla u|^2 \mathrm{d}\boldsymbol{x}. \tag{18}$$

Second, inside the hoses the potential u is well approximated by the linear interpolation [12, 6, 7] between the values of the potentials on the neighboring balls B_i and B_j. Now we introduce the local system of coordinate (x, y) with origin at $\dfrac{\boldsymbol{x}_i + \boldsymbol{x}_j}{2}$ and y-axis directed along the line segment connecting points $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. Thus, in this coordinate system the local flux in the hose Π_{ij} is approximated by

$$\nabla u \sim \left(0,\ 0,\ \frac{t_i - t_j}{H_{ij}(x,\, y)}\right), \tag{19}$$

where

$$H_{ij}(x,\, y) = \begin{cases} |\boldsymbol{x}_i - \boldsymbol{x}_j| - 2\sqrt{R^2 - x^2 - y^2}, & \text{when } \boldsymbol{x}_i,\, \boldsymbol{x}_j \text{ are centers of balls,} \\ |\boldsymbol{x}_i - \boldsymbol{x}_j| - \sqrt{R^2 - x^2 - y^2}, & \text{either } \boldsymbol{x}_i \text{ or } \boldsymbol{x}_j \text{ is a center of a} \\ & \text{quasi-ball,} \end{cases} \tag{20}$$

is the distance between the inclusions and $|\boldsymbol{x}_i - \boldsymbol{x}_j|$ is the Euclidean distance between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. Hence,

$$\int_{\Pi_{ij}} |\nabla u|^2 \mathrm{d}\boldsymbol{x} = (t_i - t_j)^2 \int_{\Pi_{ij}} \frac{\mathrm{d}\boldsymbol{x}}{H_{ij}^2} = g_{ij}(t_i - t_j)^2, \tag{21}$$

where the specific flux g_{ij} in the hose Π_{ij} is defined by

$$g_{ij} = \int_{\Pi_{ij}} \frac{\mathrm{d}\boldsymbol{x}}{H_{ij}^2} = \int_{\pi_i(\mathcal{F})} \frac{\mathrm{d}x\mathrm{d}y}{H_{ij}(x,\, y)}, \tag{22}$$

and $\pi_i(\mathcal{F})$ is the base of the common Voronoi face of two sites $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ lying on the sphere ∂B_i (shown in Fig. 4(a),(b)). If $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are not neighbors (that is, they are not connected by a common hose Π_{ij}) then $g_{ij} = 0$.

Finally, the specific fluxes g_{ij} are asymptotically large [12] when inclusions B_i and B_j are close to touching:

$$g_{ij} = \pi R\, |\ln \delta_{ij}| + O(1), \quad \text{as} \ \ \delta_{ij} \to 0, \tag{23}$$

where the *relative interparticle distance* is a dimensionless parameter given by

$$\delta_{ij} = \begin{cases} \dfrac{|\boldsymbol{x}_i - \boldsymbol{x}_j|}{R} - 2, & \text{when both } \boldsymbol{x}_i \text{ and } \boldsymbol{x}_j \text{ are centers of balls,} \\ \dfrac{|\boldsymbol{x}_i - \boldsymbol{x}_j|}{R} - 1, & \text{one of } \boldsymbol{x}_i,\ \boldsymbol{x}_j \text{ is a center of a quasi-ball.} \end{cases} \tag{24}$$

Combining (17), (21), (22), (23) we have

$$\widehat{A} \sim \frac{1}{2} \sum_{\Pi_{ij}} g_{ij}(t_i - t_j)^2.$$

This asymptotic derivation, however, does not imply

$$\widehat{A} \ \sim \ \frac{1}{2} \min_{(\tilde{t}_1, \dots, \tilde{t}_N)} \sum_{\Pi_{ij}} g_{ij}(\tilde{t}_i - \tilde{t}_j)^2, \tag{25}$$

because $\bar{\mathbf{t}} = (\bar{t_1}, \dots, \bar{t_N})$, the minimizer of the quadratic form

$$\mathcal{I}(\tilde{\mathbf{t}}) = \frac{1}{2} \sum_{\Pi_{ij}} g_{ij}(\tilde{t}_i - \tilde{t}_j)^2, \tag{26}$$

with $\tilde{\mathbf{t}} = (\tilde{t}_1, ..., \tilde{t}_N)$ and appropriate boundary conditions (defined below), may be different from the values $t_1, \dots, t_N$ of the potentials on the balls $B_1, \dots, B_N$, defined by (7). The value of $\mathcal{I}(\bar{\mathbf{t}})$ (26), defined on the minimizer $\bar{\mathbf{t}}$,

$$\mathcal{I}(\bar{\mathbf{t}}) = \min_{\tilde{\mathbf{t}}} \mathcal{I}(\tilde{\mathbf{t}}) \tag{27}$$

is called [6, 7] the *energy of the discrete system.*

To prescribe the boundary condition for the discrete network approximation on the horizontal boundaries we consider sets $S^{\pm}$ of the centers of the balls that cross or touch boundaries $\partial Q^{\pm}$ and quasi-balls lying on $\partial Q^{\pm}$. Then we define the values of the discrete potentials $\widetilde{t}_i$ on $S^{\pm}$ by

$$\widetilde{t}_i = \pm 1, \quad \text{when } \boldsymbol{x}_i \in S^{\pm}. \tag{28}$$

Then the set $\mathbf{I}$ of the "interior" sites of the discrete system is defined by $\mathbf{I} = \{\boldsymbol{x}_i,\ i = 1, \dots, K\} \setminus \{S^{+} \cup S^{-}\}$.

If $\boldsymbol{x}_{i''} \notin \mathbf{I} \cup (S^{-} \cup S^{+})$, then it is a center of a quasi-ball that lies on the lateral boundary ∂Q_{lat}. The set of such vertices is denoted by S_{lat}. For $\boldsymbol{x}_{i''} \in S_{\text{lat}}$ we assign the value of the discrete potential equal to the potential of the site $\boldsymbol{x}_i$ connected with $\boldsymbol{x}_{i''}$ by the edge $e_{ii''}$:

$$\widetilde{t}_{i''} = \widetilde{t}_i, \quad \text{for } \boldsymbol{x}_{i''} \in S_{\text{lat}},\ \boldsymbol{x}_i \in \mathbf{I}. \tag{29}$$

We comment that the potentials $t_{i''}$ for $\boldsymbol{x}_{i''} \in S_{\text{lat}}$ do not participate in (25) and can be found after solving the problem by using the equality (29). Note that the lateral boundary condition (7e) for the discrete network can be interpreted as follows:

$$\sum_{\Pi_{i''j}, i'' \text{ fixed}} g_{i''j}(t_{i''} - t_j) = g_{i''i}(t_{i''} - t_i) = 0, \quad \text{for } \boldsymbol{x}_i \in S_{\text{lat}}. \tag{30}$$

Let us define the discrete fluxes through the boundaries S^+ and S^- in the following way:

$$P^+ = \sum_{\Pi_{ij},\, \boldsymbol{x}_i \in S^+} g_{ij}(t_i - t_j); \quad P^- = \sum_{\Pi_{ij},\, \boldsymbol{x}_i \in S^-} g_{ij}(t_i - t_j). \tag{31}$$

Then similarly to (10) we have

$$\mathcal{I}(\mathbf{t}) = \frac{1}{2} \sum_{\Pi_{ij}} g_{ij}(t_i - t_j)^2 = \frac{1}{2}\left(P^+ - P^-\right). \tag{32}$$

The derivation of (32) is similar to the two-dimensional case of [7].

3.4 Properties of the Discrete Network

Here we collect some properties of the discrete network approximation, which we use for the variational error estimates presented in Sect. 4.3. Since the effective conductivity $\widehat{A}$ is approximated by the fluxes in the hoses between closely spaced inclusions (18), we define a notion of a δ-subgraph that corresponds to some collection of the densely packed balls in the composite. Then we introduce a so-called *δ-$\mathcal{D}$ partition* $\{\Lambda_\delta\}$ of Q into polyhedra with δ-short edges and triangles as faces. $\mathcal{D}$ can be thought of as the *relative diameter* of "holes" in the composite. The error estimates for our discrete network approximation are determined in terms of the characteristic parameters δ and $\mathcal{D}$.

The discrete minimization problem (27)-(29) amounts to solving a linear algebraic system whose solutions are the discrete potentials t_i at the interior vertices as indicated in the following lemma.

Lemma 1. *There is a unique solution* $\mathbf{t} = \{t_i : \boldsymbol{x}_i \in \boldsymbol{I}\}$ *to the discrete minimization problem (27)-(29), which satisfies the discrete Euler-Lagrange equations:*

$$\sum_{j=1}^{K} g_{ij}(t_i - t_j) = 0, \quad \textit{for fixed } \boldsymbol{x}_i \in \boldsymbol{I}. \tag{33}$$

The proofs of Lemma 1 and the following Lemma 2 are identical to the two dimensional case [7].

For any graph $\mathcal{G}$ and $\delta > 0$ we obtain the *δ-subgraph* $\mathcal{G}_\delta$ by discarding edges e_{ij} if the corresponding $\delta_{ij} > \delta$, that is the length of each edge of $\mathcal{G}_\delta$ is not greater than $2R + \delta R$. For a given $\delta > 0$, consider the set of *small* triangles such that *all* of their edges belong to the δ-subgraph $\mathcal{G}_\delta$:

$$\{\, \text{triangle } \triangle \boldsymbol{x}_i \boldsymbol{x}_j \boldsymbol{x}_k : e_{ij},\, e_{jk},\, e_{ki} \in \mathcal{G}_\delta \,\}. \tag{34}$$

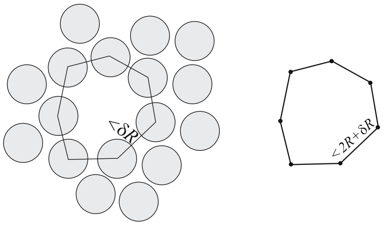


Fig. 7. The two dimensional "hole" and the corresponding polygon

Now we note that not all inclusions are closely spaced, that is, the distance between two neighbors might be of order $O(R)$. In a real composite this situation corresponds to a "hole" or void space (see Fig. 1). Such a "hole" corresponds to an interior of some solid, polyhedron, whose edges are not longer than $2R + \delta R$ in our discrete network. To clarify this idea let us consider a two dimensional "hole" shown in Fig. 7. The centers of disks located at distance less than δR from each other form a polygon in the corresponding graph. The length of each edge of the polygon is less than $2R + \delta R$. Similarly, a three dimensional "hole" (see Fig. 1) provides a polyhedron in the corresponding graph whose edges are shorter than $2R + \delta R$. Note that due to Delauney tetrahedralization each face of such a polyhedron is a triangle (Fig. 8). Below we define the polyhedron and call it a *δ-polyhedron*.

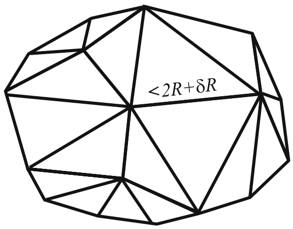


Fig. 8. The δ-polyhedron

We assume that for any given $\delta > 0$ we can partition the domain Y into polyhedra $\{\Lambda_\delta\}$, called δ-polyhedra, whose edges are shorter than $2R + \delta R$. Thus $\bigcup \Lambda_\delta = Y$ and $\partial\Lambda_\delta \in \mathcal{G}_\delta$.

Let the diameter of any Λ_δ be not greater than $\mathcal{D}R$, where R is the radius of inclusion. Then $\mathcal{D}$ is called the *relative diameter* of δ-polyhedra of $\{\Lambda_\delta\}$. If the distance between two points $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ exceeds $\mathcal{D}R$ then they belong to two different δ-polyhedra Λ^i_δ and Λ^j_δ. Thus, if we construct any path connecting $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ this path will contain at least one vertex of some δ-polyhedra Λ_δ and, consequently, a vertex of the δ-subgraph $\mathcal{G}_\delta$. This fact we accept as the definition of a *δ-$\mathcal{D}$ partition*.

Definition 2. *We say that the δ-subgraph $\mathcal{G}_\delta$ induces the δ-$\mathcal{D}$ partition $\{\Lambda_\delta\}$ of Y if for any two vertices $\boldsymbol{x}_i$, $\boldsymbol{x}_j \in \mathcal{G}$, satisfying $|\boldsymbol{x}_i - \boldsymbol{x}_j| > \mathcal{D}R$, any path in $\mathcal{G}$ connecting them contains at least one vertex of the subgraph $\mathcal{G}_\delta$. The partition $\{\Lambda_\delta\}$ is the* *set of non-overlapping δ-polyhedra Λ_δ which are three dimensional solids, whose boundaries consist of small triangles (34).*

Definition 3. *The distribution of balls is δ-$\mathcal{D}$ connected if a δ-$\mathcal{D}$ partition $\{\Lambda_\delta\}$ of Y exists.*

The main use of the δ-$\mathcal{D}$ partition is the following maximum principle.

Lemma 2. *(The Discrete Maximum Principle).*
Suppose $\mathbf{t} = \{t_1, t_2, \ldots, t_M\}$ is the solution to the discrete problem (33). Denote by

$$t_{max} = \max_{\boldsymbol{x}_i \in \partial \Lambda_\delta} t_i, \quad t_{min} = \min_{\boldsymbol{x}_i \in \partial \Lambda_\delta} t_i \tag{35}$$

the maximal and minimal values of the potential on the boundary of any δ-polyhedron Λ_δ. Then the value of the potential t_k at any vertex $\boldsymbol{x}_k$ in the interior of this δ-polyhedron ($\boldsymbol{x}_k \in \Lambda_\delta$) is bounded:

$$t_{min} \le t_k \le t_{max}.$$

An immediate consequence of the maximum principle is a bound on the potential difference of the vertices inside Λ_δ in terms of the potentials of the vertices on $\partial\Lambda_\delta$.

Lemma 3. *Suppose Λ_δ is any δ-polyhedron of the δ-$\mathcal{D}$ partition. The values of the potential on any given vertices $\boldsymbol{x}_k$, $\boldsymbol{x}_l \in$ Int Λ_δ are bounded as follows:*

$$(t_k - t_l)^2 \le \frac{(\mathcal{D}+2)^3}{8} \sum_{\Pi_{ij} \in \partial \Lambda_\delta} (t_i - t_j)^2. \tag{36}$$

Proof. The formula (36) can be proved by applying the discrete maximum principle (35) and the triangle inequality for the values t_i on $\boldsymbol{x}_i \in \Lambda_\delta$:

$$(t_k - t_l)^2 \le (t_{max} - t_{min})^2 \le n \sum_{\Pi_{ij} \in \partial \Lambda_\delta} (t_i - t_j)^2,$$

where *n* is the number of vertices of $\partial\Lambda_\delta$. This number is less than the number M of balls of radius R that can be placed inside the sphere of diameter $\mathcal{D}R$. The number M is bounded by $M \le \frac{(\mathcal{D}+2)^3}{8}$. Hence, we obtain (36). □

The numbers of the hoses, prisms and tetrahedra inside any Λ_δ of the δ-$\mathcal{D}$ partition of Y can be bounded in terms of the parameter $\mathcal{D}$ as follows.

Lemma 4. *For any Λ_δ of the δ-$\mathcal{D}$ partition of Y the number of tetrahedra $\#\triangle_{\Lambda_\delta}$, the number of prisms $\#P_{\Lambda_\delta}$ and the number of hoses $\#\Pi_{\Lambda_\delta}$ in Λ_δ satisfy:*

$$(a) \qquad \#\triangle_{\Lambda_\delta} \le K_1 \mathcal{D}^3, \tag{37}$$

$$(b) \qquad \#P_{\Lambda_\delta} \le K_2 \mathcal{D}^3, \tag{38}$$

$$(c) \qquad \#\Pi_{\Lambda_\delta} \le K_3 \mathcal{D}^3. \tag{39}$$

where constants K_i, $i = 1, 2, 3$, are universal.

The derivation of (37) and the values of the corresponding constants can be found in Appendix 6.2.

4 Variational Error Estimates

Our approach to variational error estimates is to construct trial fields for (14), that give rise to tight upper and lower bounds (constructed in Sects. 4.1 and 4.2, respectively), when inclusions are close to touching. For both lower and upper bounds in (14) the trial functions $u \in V$ and $\boldsymbol{v} \in W$ are constructed in the hoses, prisms and tetrahedra determined by the central projection partition of the matrix Q. The piecewise-differentiable trial function $u \in V$ is chosen to be linear in the hose, a linear interpolation in the tetrahedron, and a linear interpolation in every cross-section of the prism, and so that it takes values ± 1 on $\partial Q^{\pm}$. The trial field $\boldsymbol{v} \in W$ is chosen so that it is equal to zero everywhere except in the hoses. In the hoses it is equal to the local flux defined by (19), where t_i are determined as solutions of the discrete minimization problem (27)-(29).

Our goal is to investigate when the discrete energy $\mathcal{I}$ (27) is a *good* approximation for $\widehat{A}$. For this purpose in Sect. 4.3 the difference between upper and lower bounds is estimated in terms of the parameters δ, $\mathcal{D}$ and $\mathcal{I}$, and the error is obtained.

4.1 Lower Bound

The construction, derivation and justification of the lower bound is identical to that found in [7], where the technical details of this construction can be found.

Theorem 1. *The lower bound on the effective conductivity $\widehat{A}$ is given by*

$$\mathcal{I}(\mathbf{t}) \leq \widehat{A} \tag{40}$$

where $\mathcal{I}(\mathbf{t})$ is defined by (27), $\mathbf{t} = (t_1, \ldots t_N)$ are the solutions of the minimization problem (27)-(29) (or equivalently (33)) and the specific fluxes g_{ij} are defined by (22).

Proof. Consider two neighboring balls B_i and B_j centered at $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, and the values of the potentials t_i and t_j, respectively. The hose Π_{ij} connects them. We choose

$$\boldsymbol{v} = \boldsymbol{v}_{ij} = \begin{cases} \left(0,\, 0,\, \dfrac{t_i - t_j}{H_{ij}(x,\, y)}\right), & \text{in } \Pi_{ij}, \quad i, j = 1, \ldots, K \\ (0,\, 0,\, 0)\,, & \text{otherwise} \end{cases} \tag{41}$$

where the distance between two neighboring balls $H_{ij}(x,\, y)$ is defined by (20). Note that this function is divergence-free since its normal components match along the discontinuity. The potentials t_i and t_j on the balls B_i and B_j satisfy the linear system (33). Then the integral condition $\displaystyle\int_{\partial B_i} \boldsymbol{v} \cdot \boldsymbol{n} \; \mathrm{d}\boldsymbol{x} = 0$, in the definition of the class W, is satisfied, so $\boldsymbol{v} \in W$.

Now we calculate $J[\boldsymbol{v}]$ defined by (12). First, evaluate the integral

$$\frac{1}{2} \int_Q \boldsymbol{v}^2 \mathrm{d}\boldsymbol{x} = \frac{1}{2} \sum_{\Pi_{ij}} g_{ij} (t_i - t_j)^2$$

by the definition of the specific flux g_{ij} and by the choice of the function $\boldsymbol{v}$. Also we note that the fluxes through the boundaries $\partial Q^{\pm}$ are equal to the discrete fluxes $P^{\pm}$ defined by (31). Then

$$\begin{aligned} J[\boldsymbol{v}] &= -\frac{1}{2}\int_Q \boldsymbol{v}^2 \mathrm{d}\boldsymbol{x} + \int_{\partial Q^+} \boldsymbol{v}\cdot\boldsymbol{n}\ \mathrm{d}\boldsymbol{x} - \int_{\partial Q^-} \boldsymbol{v}\cdot\boldsymbol{n}\ \mathrm{d}\boldsymbol{x} \\ &= -\frac{1}{2}\sum_{\Pi_{ij}} g_{ij}(t_i - t_j)^2 + \left(P^+ - P^-\right) = \mathcal{I}(\mathbf{t}). \end{aligned} \tag{42}$$

Then from (14) we obtain (40). □

4.2 Upper Bound

The upper bound on the effective conductivity is given by the following theorem.

Theorem 2. *For any δ-$\mathcal{D}$ connected distribution of balls, the upper bound on the effective conductivity $\widehat{A}$ is*

$$\widehat{A} \le \mathcal{I}(\mathbf{t}) + \widehat{C}R\sum_{\Pi_{ij}}(t_i - t_j)^2, \tag{43}$$

where $\mathbf{t} = (t_1, \ldots, t_N)$ is the solution to the discrete minimization problem (27)-(29), and the constant $\widehat{C}$ depends on the relative diameter $\mathcal{D}$ only:

$$\widehat{C} \le C_0\mathcal{D}^4.$$

The proof of this theorem relies on the following more general variational upper bound given by Lemma 5. In finite element methods [14] for a given distribution of points $\boldsymbol{x}_i$ and Delaunay tetrahedralization $\{T\}$, the quotient $\gamma = \dfrac{\rho}{\ell}$, where ρ is the radius of the circumsphere (circumradius) and ℓ is the length of the shortest edge of the tetrahedron T, is called a mesh quality measure. Similarly, we use here a parameter

$$\gamma_0 = \max_{T\in\{T\}} \gamma \tag{44}$$

to measure the regularity of our (Delaunay) graph $\mathcal{G}$ or, equivalently, the Delaunay tetrahedralization $\{T\}$ induced by $\mathcal{G}$.

Lemma 5. *For any distribution of balls, the upper bound on the effective conductivity $\widehat{A}$ is*

$$\begin{aligned} \widehat{A} \le \frac{1}{2}\sum_{T\in\{T\}} \Bigg(&\sum_{\Pi'_{ij}\subset T} \left[g_{ij} + C_\Pi\right](t_i - t_j)^2 \\ &+ \sum_{P'_{ijk}\subset T} C_P\left\{(t_i - t_j)^2 + (t_i - t_k)^2\right\} \\ &+ \sum_{\triangle_{ijkm}\subset T} C_\triangle\left\{(t_i - t_m)^2 + (t_j - t_m)^2 + (t_k - t_m)^2\right\} \Bigg), \end{aligned} \tag{45}$$

where Π'_{ij} is the part of the hose Π_{ij} lying inside the tetrahedron T, analogously, P'_{ijk} is the part of the prism P_{ijk} lying inside T,

$$C_\Pi = \begin{cases} 2\pi R \ln 2\gamma_0 + 4\pi R\left(1 - \dfrac{1}{2\gamma_0}\right), \textit{for inner hose}, \\ \pi R \ln 2\gamma_0 + 2\pi R\left(1 - \dfrac{1}{2\gamma_0}\right), \ \textit{for boundary hose}, \end{cases}$$

$$C_P = \begin{cases} 8\gamma_0 R, \ \textit{for inner prism}, \\ 16\gamma_0 R, \textit{for boundary prism}, \end{cases} \tag{46}$$

$$C_\triangle = \begin{cases} \dfrac{4}{3}\gamma_0^3 \rho, \textit{for inner tetrahedron}, \\ \gamma_0^3 \rho, \ \ \textit{for boundary tetrahedron}, \end{cases}$$

γ_0 is the the measure of regularity of $\mathcal{G}$, given by (44) and ρ is the circumradius of the tetrahedron $T \in \{T\}$.

We call the hose, prism and tetrahedron the boundary hose, prism and tetrahedron, respectively, if they belong to a tetrahedron $T \in \{T\}$ which has at least one vertex in a quasi-ball. Otherwise we call them inner hose, prism and tetrahedron, respectively. Note that inside every inner tetrahedron T, that is, none of its vertices belongs to a quasi-ball, there are six hoses, four prisms and one tetrahedron (in this case the last sum in (45) contains only one term).

Proof of Lemma 5. The trial function $u \in V$ is piecewise differentiable. Its construction will be done in the hose Π_{ij}, prism P_{ijk} and tetrahedron $\triangle_{ijkm}$.

First, let us consider the hose Π_{ij}. We choose the local coordinate system so the z-axis is directed along the edge connecting two neighbors, as shown in Fig. 9 (points A_B and B_A are the points of the intersection of the line segment connecting the two centers $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ with the spheres ∂B_i and ∂B_j, respectively), and the origin is in $\dfrac{\boldsymbol{x}_i + \boldsymbol{x}_j}{2}$.

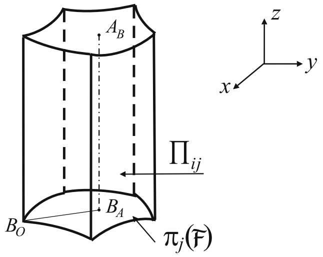


Fig. 9. The hose Π_{ij}

We take a trial function u in Π_{ij} to be linear in z and inversely proportional to the distance $H_{ij}(x, y)$, defined by (20), so that u takes the values t_i and t_j on the spheres ∂B_i and ∂B_j, respectively. Thus in the hose Π_{ij}

$$u(\boldsymbol{x}) = u(x,\, y,\, z) = \frac{t_i - t_j}{H_{ij}(x,\, y)} z + \frac{t_i + t_j}{2}, \quad z \in \left(-\frac{H_{ij}(x,\, y)}{2},\, \frac{H_{ij}(x,\, y)}{2} \right). \tag{47}$$

Then the Dirichlet integral of the function u defined by (47) over the hose Π_{ij} is bounded (see details in Appendix 6.3, Lemma 7) as follows:

$$\int_{\Pi'_{ij}} |\nabla u|^2 \mathrm{d}\boldsymbol{x} \le \int_{\Pi_{ij}} |\nabla u|^2 \mathrm{d}\boldsymbol{x} \le \left(g_{ij} - \pi R \ln\left(\frac{\ell}{2\rho}\right) + \pi R \left(1 - \frac{\ell}{2\rho}\right)\right) (t_i - t_j)^2 \tag{48}$$

where ρ is the circumradius and ℓ is the length of the shortest edge of T.

Next we construct the trial function u in the prism P_{ijk}. Consider the prism $P_{ijk} = A'B'C'C^*B^*A^*$ shown in Fig. 10 and the part of this prism $P'_{ijk} = A'B'C'C''B''A''$ lying in the tetrahedron T shown in Fig. 15(b). We choose the local coordinate system so that the triangle $\triangle A'B'C'$ lies on the xz plane and the y-axis is directed along the altitude of this prism (Fig. 10).

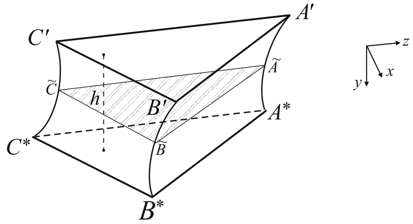


Fig. 10. The prism $P_{ijk} = A'B'C'C^*B^*A^*$

For any cross-section, $\triangle\tilde{A}\tilde{B}\tilde{C}$, perpendicular to the y-axis, the trial function u is a linear function taking values $t_i,\ t_j,\ t_k$ on the arcs $A'A^*$, $B'B^*$, $C'C^*$, respectively. Let h be the length of the altitude of the prism $A'B'C'C^*B^*A^*$: $h \le 2R$. Note that for such a choice of the coordinate system, for some $y = y_0 \in (0,\, h)$ we have $u(x_1,\, y_0,\, z_1) = u(x_2,\, y_0,\, z_2), \forall x_1,\, x_2,\, z_1,\, z_2$.

The central projection partition construction provides the congruence of the cross-section $\triangle\tilde{A}\tilde{B}\tilde{C}$ and $\triangle A'B'C'$ to $\triangle ABC$. To see this one can take a view from the top at prism P_{ijk} depicted in Fig. 11. The concentric circles are the cross-sections of the spheres B_i, B_j, and B_k centered at the points A, B, and C respectively. Points $\widetilde{A}$ and A' are intersections with the sphere B_i of line segments that connect the point A with the point O, which is an intersection of bisectors to AB, BC, and CA (see sections 3.2 and 6.1 for details). Points $\widetilde{B}$, B' and $\widetilde{C}$, C' are defined similarly. Obviously, both $\triangle A'B'C'$ and $\triangle\tilde{A}\tilde{B}\tilde{C}$ is congruent to $\triangle ABC$.

Hence, we get the following estimate

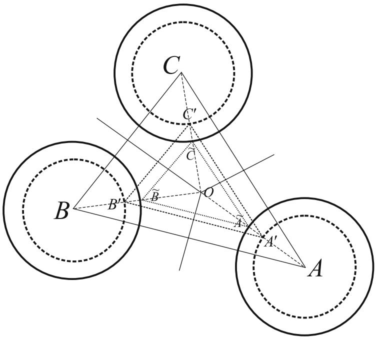


Fig. 11. The prism P_{ijk} from the top

$$\begin{aligned}\int_{P'_{ijk}} |\nabla u|^2 \mathrm{d}\boldsymbol{x} \leq \int_{P_{ijk}} |\nabla u|^2 \mathrm{d}\boldsymbol{x} \leq \int_0^h \mathrm{d}y \int_{\triangle \tilde{A}\tilde{B}\tilde{C}} |\nabla u|^2 \, \mathrm{d}x\mathrm{d}z \\ = \int_0^h \mathrm{d}y \int_{\triangle A'B'C'} |\nabla u|^2 \, \mathrm{d}x\mathrm{d}z \leq 2R \int_{\triangle A'B'C'} |\nabla u|^2 \, \mathrm{d}x\mathrm{d}z.\end{aligned} \tag{49}$$

The proof of equality of the integrals over $\triangle \tilde{A}\tilde{B}\tilde{C}$ and $\triangle A'B'C'$ in (49) is given in Appendix 6.6.

For constructing the gradient of u in the triangle $\triangle A'B'C'$, we use the procedure given in [7], which yields the following bound:

$$\int_{P_{ijk}} |\nabla u|^2 \mathrm{d}\boldsymbol{x} \leq \frac{4R}{\sin\theta} \left\{ (t_i - t_j)^2 + (t_i - t_k)^2 \right\}, \tag{50}$$

where θ is the smallest angle in the triangle $\triangle A'B'C'$ $\left(\theta \leq \frac{\pi}{3}\right)$.

The right hand side of the formula (50) can be expressed in terms of the quotient $\gamma = \frac{\rho}{\ell}$ as follows:

$$\sin\theta \geq \frac{1}{2\gamma}, \tag{51}$$

(see details in Appendix 6.4). Now from (50) and (51) we obtain the following estimate in the prism P'_{ijk}:

$$\int_{P'_{ijk}} |\nabla u|^2 \mathrm{d}\boldsymbol{x} \leq 8\gamma R \left\{ (t_i - t_j)^2 + (t_i - t_k)^2 \right\}. \tag{52}$$

Finally, we construct the trial function u in the tetrahedron $\triangle_{ijkm} = A'B'C'D'$, see Fig. 15(a). Inside the tetrahedron the trial function u is a linear function, taking values t_i, t_j, t_k and t_m at the points A', B', C' and D' respectively. The Dirichlet integral of such a function over $\triangle_{ijkm}$ is bounded as follows:

$$\int_{\triangle_{ijkm}} |\nabla u|^2 \mathrm{d}\boldsymbol{x} \leq \frac{1}{3\,|A'B'C'D'|} \left\{ \frac{1}{4} \left|\overrightarrow{D'B'} \times \overrightarrow{D'C'}\right|^2 (t_i - t_m)^2 \right.$$
$$\left. + \frac{1}{4} \left|\overrightarrow{D'C'} \times \overrightarrow{D'A'}\right|^2 (t_j - t_m)^2 + \frac{1}{4} \left|\overrightarrow{D'A'} \times \overrightarrow{D'B'}\right|^2 (t_k - t_m)^2 \right\},$$

(for the details of the construction of u and evaluating its Dirichlet integral see Appendix 6.5, Lemma 8).

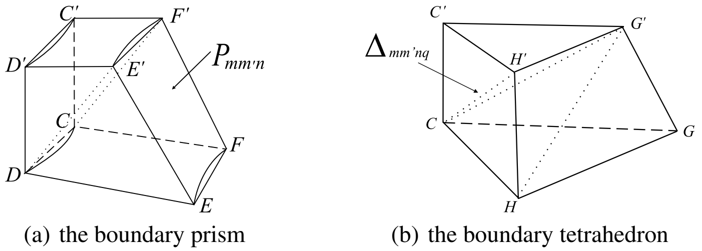
Note that since the tetrahedron $\varDelta_{ijkm}$ is similar to the tetrahedron $T = ABCD \in \{T\}$ (see Fig. 15(a)) then:

$$\frac{\rho'}{\ell'} = \frac{\rho}{\ell} = \gamma,$$

where ρ' is the radius of the circumsphere and ℓ' is the length of the shortest edge of the tetrahedron $\varDelta_{ijkm}$.

Recall that $\frac{1}{2}\left|\overrightarrow{D'B'} \times \overrightarrow{D'C'}\right|$ is the area of the triangle made of the vectors $\overrightarrow{D'B'}$ and $\overrightarrow{D'C'}$. The length of a side of a triangle is less than the diameter of the circumcircle, and since the radius of the circumcircle of any face is less than the radius of the circumsphere ρ' of the tetrahedron, we have $\frac{1}{2}\left|\overrightarrow{D'B'} \times \overrightarrow{D'C'}\right| \leq 2\rho'^2$ (the same is true for the other cross products). Since $|A'B'C'D'| = \det\left[\overrightarrow{D'A'},\, \overrightarrow{D'B'},\, \overrightarrow{D'C'}\right]$ is the volume of the parallelepiped made of the vectors $\overrightarrow{D'A'}$, $\overrightarrow{D'B'}$ and $\overrightarrow{D'C'}$, we have $\det\left[\overrightarrow{D'A'},\, \overrightarrow{D'B'},\, \overrightarrow{D'C'}\right] \geq \ell'^3$. Hence we obtain the following bound:

$$\begin{aligned} \int_{\triangle_{ijkm}} |\nabla u|^2 \mathrm{d}\boldsymbol{x} &\leq \frac{4\,\rho'^4}{3\,\ell'^3} \left\{ (t_i - t_m)^2 + (t_j - t_m)^2 + (t_k - t_m)^2 \right\} \\ &\leq \frac{4}{3}\,\gamma^3 \rho' \left\{ (t_i - t_m)^2 + (t_j - t_m)^2 + (t_k - t_m)^2 \right\} \\ &\leq \frac{4}{3}\,\gamma^3 \rho \left\{ (t_i - t_m)^2 + (t_j - t_m)^2 + (t_k - t_m)^2 \right\}. \end{aligned} \tag{53}$$



(a) the boundary prism (b) the boundary tetrahedron

Fig. 12.

The construction the trial function u in hoses, prisms and tetrahedra connecting the balls and quasi-balls near the upper boundary of Q requires an auxiliary construction (Fig. 6). The construction of u in the hose $\mathit{\Pi}_{mm'}$ is the same as (47), taking into account that $t_{m'} = 1$. Then similar to (48) we have

$$\begin{aligned}\int_{\mathit{\Pi}'_{mm'}} |\nabla u|^2 \mathrm{d}\boldsymbol{x} &\le \int_{\mathit{\Pi}_{mm'}} |\nabla u|^2 \mathrm{d}\boldsymbol{x}\\ &\le \left(g_{mm'} + \pi R \ln \frac{2\rho}{\ell} + 2\pi R \left(1 - \frac{\ell}{2\rho} \right) \right) (t_m - t_{m'})^2,\end{aligned} \tag{54}$$

where ρ is the circumradius and ℓ is the shortest edge of the tetrahedron $MM'NQ$ (here we used the fact that $H_{mm'} = \delta_{mm'} R + R - \sqrt{R^2 - x^2 - y^2}$).

In the boundary tetrahedron $MM'NQ$ (Fig. 6) there are four prisms: $P_{mm'n}$, $P_{m'nq}$, $P_{mm'q}$ and P_{mnq}. The construction of the trial function u in P_{mnq} is the same as for the inner case; for the estimate of the Dirichlet integral over P_{mnq} one can use (52).

To construct the trial function u in the prism $P_{mm'n}$, for example, we divide it into two parts P_1 and P_1 ($P_1 = EFC'D'DC$, $P_2 = EFC'D'E'F'$ in Fig. 12(a)) so that the base of one of prism is the right triangle. Then we construct u in each and estimate the Dirichlet integrals over them using (52). From this we obtain

$$\begin{aligned}\int_{P'_{mm'n}} |\nabla u|^2 \mathrm{d}\boldsymbol{x} \le \int_{P_{mm'n}} |\nabla u|^2 \mathrm{d}\boldsymbol{x} &= \int_{P_1} |\nabla u|^2 \mathrm{d}\boldsymbol{x} + \int_{P_2} |\nabla u|^2 \mathrm{d}\boldsymbol{x}\\ &\le \frac{4R}{\sin\theta_1} \left\{ (t_m - t_{m'})^2 + (t_{m'} - t_{m'})^2 \right\}\\ &+ \frac{4R}{\sin\theta_2} \left\{ (t_n - t_{m'})^2 + (t_m - t_{m'})^2 \right\},\end{aligned} \tag{55}$$

where θ_1 and θ_2 are the smallest angles of the triangles $\triangle DE'D'$ and $\triangle DE'E$, respectively. Then $\sin\theta_1 \ge \dfrac{\ell/2}{\rho}$, similar to the case of the inner prism.

In order to estimate $\sin\theta_2$ we note that if the sphere centered at the point N (Fig. 6) intersects the boundary ∂Q^+ then $t_n = 1$ and all potentials at the vertices of $\triangle DE'E$ are equal to one. If the sphere centered at N does not intersect ∂Q^+ (t_n is not necessarily 1) then in $\triangle DE'E$ so we have $\sin\theta_2 \ge \dfrac{\ell}{2\rho}$ again analogous to the inner prism case. Then continuing (55) we get:

$$\begin{aligned}\int_{P_{mm'n}} |\nabla u|^2 \mathrm{d}\boldsymbol{x} &\le \frac{8R\rho}{\ell} (t_m - t_{m'})^2 + \frac{8R\rho}{\ell} \left\{ (t_n - t_{m'})^2 + (t_m - t_{m'})^2 \right\}\\ &\le \frac{16R\rho}{\ell} \left\{ (t_n - t_{m'})^2 + (t_m - t_{m'})^2 \right\}\\ &\le 16R\gamma \left\{ (t_n - t_{m'})^2 + (t_m - t_{m'})^2 \right\}.\end{aligned} \tag{56}$$

In order to construct the trial function u in the tetrahedron $\triangle_{mm'nq}$ we decompose this truncated tetrahedron into three tetrahedra $\triangle_i$, $i = 1, 2, 3$ ($\triangle_1 = CC'G'H'$, $\triangle_2 = CHH'G'$, $\triangle_3 = CHGG'$ in Fig. 12(b)) and choose u to be a linear function in each. Then one can use the estimate (82) in Appendix 6.5 for each tetrahedron $\triangle_i$:

$$\begin{aligned} \int_{\triangle_{mm'nq}} |\nabla u|^2 \mathrm{d}\boldsymbol{x} &= \int_{\triangle_1} |\nabla u|^2 \mathrm{d}\boldsymbol{x} + \int_{\triangle_2} |\nabla u|^2 \mathrm{d}\boldsymbol{x} + \int_{\triangle_3} |\nabla u|^2 \mathrm{d}\boldsymbol{x} \\ &\leq 3 \frac{4\gamma^3 \rho}{3} \left\{ (t_m - t_{m'})^2 + (t_n - t_{m'})^2 + (t_q - t_{m'})^2 \right\}, \end{aligned} \tag{57}$$

where $t_{m'} = 1$.

Thus the estimates (48), (52), (53) together with (54), (56) and (57) yield (45). □

Now we prove Theorem 2 using the above lemma.

Proof of Theorem 2. The key observation here is that all constants in (45), depending on the radius R of inclusions and the parameter γ_0, can be expressed in terms of $\mathcal{D}$ only. First, we note that for any tetrahedron T the length of its shortest edge ℓ is greater than $2R$: $\ell > 2R$. Note that the circumsphere of any T does not contain vertices of other tetrahedra, and by Definition 3, the diameter 2ρ of this circumsphere is bounded as follows: $2\rho \leq \mathcal{D}R$, for boundary tetrahedron $\rho \leq \mathcal{D}R$. This means that $\gamma_0 < \frac{\mathcal{D}}{4}$, and for boundary tetrahedron $\gamma_0 < \frac{\mathcal{D}}{2}$. Thus, every constant in (46) is bounded from above by some constant depending on $\mathcal{D}$ which is a multiple of the radius of the inclusion R:

$$C_{\Pi} := 2\pi R \ln(2\gamma_0) + 4\pi R \left(1 - \frac{1}{2\gamma_0}\right) \leq C^{11} R \ln \mathcal{D} + C^{12} R \left(1 - \frac{2}{\mathcal{D}}\right),$$

$$C_P := 16\gamma_0 R \leq C^2 R\mathcal{D},$$

$$C_{\triangle} := 4\gamma_0^3 \rho \leq C^3 R\mathcal{D}^4,$$

where constants C^{11}, C^{12}, C^2 and C^3 are universal. Thus, all constants C_{Π}, C_P, and $C_{\triangle}$ bounded from above by $C_0\mathcal{D}^4 R$, where C_0 is universal (that is, does not depend on R, γ_0 or δ). Hence, we have (43). □

4.3 Error Estimate

Here we give two theorems. Theorem 3 is about the leading term of the asymptotics of the effective conductivity $\widehat{A}$ as $\delta \to 0$. This leading term is defined by the discrete energy $\mathcal{I}$ (27), which is of order $|\ln \delta|$ for small δ. In Theorem 4 we obtain the relative error of approximation of the effective conductivity by the discrete energy (27) provided that the discrete network approximating the continuum problem is δ-$\mathcal{D}$ connected (Definition 3).

We assume that there exists at least one *δ-path* connecting the upper boundary ∂Q^+ with the lower boundary ∂Q^-, where the δ-path is a path such that the maximum relative interparticle distance between balls centered at its vertices is bounded by δ. Then the following theorem holds.

Theorem 3. *If there exists at least one δ-path connecting the upper and lower boundaries of Q then the effective conductivity $\widehat{A}$ (8)*

$$\widehat{A} = \frac{1}{2}\int_Q |\nabla u|^2 d\boldsymbol{x}$$

satisfies the following inequality:

$$|\widehat{A} - \mathcal{I}(\mathbf{t})| < \boldsymbol{C}, \quad \mathcal{I}(\mathbf{t}) > K|\ln\delta|, \quad \textit{as} \quad \delta \to 0, \tag{58}$$

where $\mathbf{t} = (t_1, \ldots, t_N)$ is a solution of the discrete minimization problem (33) and g_{ij} are defined by (22)

$$g_{ij} = \int_{\Pi_{ij}} \frac{\mathrm{d}\boldsymbol{x}}{H_{ij}^2} \sim \pi R|\ln\delta_{ij}|, \quad \textit{as} \quad \delta_{ij} \to 0.$$

The constant $\boldsymbol{C}$ in (58) depends on the measure of regularity γ_0 and the number of inclusions N, but is independent of δ. The constant K in (58) depends on the number of inclusions N, the radius R and their locations, but not on δ.

Proof. By Theorem 2 we find that

$$|\widehat{A} - \mathcal{I}(\mathbf{t})| \le \sum_{\Pi_{ij}} \widehat{C}R(t_i - t_j)^2, \tag{59}$$

where the constant $\widehat{C}$ depends on the measure of the regularity of the ball distribution γ_0. By the maximum principle we have $|t_i - t_j| \le 2$, for any i, j. Then from (59) we have

$$|\widehat{A} - \mathcal{I}(\mathbf{t})| \le 4NR\widehat{C}(\gamma_0), \tag{60}$$

which yields the first inequality in (58).

In order to prove the second inequality (58) we assume that there is a δ-path connecting the upper and lower boundaries of length k (the number of vertices in the path) and consider

$$\begin{aligned}\mathcal{I}(\mathbf{t}) &\ge \pi \sum_{\text{short } \Pi_{ij}} R\,|\ln\delta_{ij}|\,(t_i - t_j)^2 \ge \pi R|\ln\delta| \sum_{\text{short } \Pi_{ij}} (t_i - t_j)^2 \\ &\ge \pi R|\ln\delta| \sum_{\delta-\text{path}} (t_i - t_j)^2 \ge \pi R|\ln\delta| \min_{\widetilde{\mathbf{t}}} \sum_{\delta-\text{path}} (\widetilde{t}_i - \widetilde{t}_j)^2 \\ &\ge \pi R|\ln\delta|\frac{1}{k} > K|\ln\delta|,\end{aligned}$$

where “short Π_{ij}” means the hose Π_{ij} corresponding relative interparticle distance δ_{ij} of which is less or equal than δ.

This gives (58). □

The next issue we would like to address here is how large is the constant $\boldsymbol{C}$ in (58). In the following Theorem 4 we obtain an upper bound on this constant $\boldsymbol{C}$ in terms of the relative $\mathcal{D}$ defined in section 3.4 and the radius of inclusion R, provided the discrete network $\mathcal{G}$ of (7) is δ-$\mathcal{D}$ connected.

Theorem 4. *If the discrete network $\mathcal{G}$ for the continuum problem (7) is δ-$\mathcal{D}$ connected, the relative error for the effective conductivity $\widehat{A}$ is*

$$\frac{\left|\widehat{A}-\mathcal{I}(\mathbf{t})\right|}{\mathcal{I}(\mathbf{t})}\leq\frac{\mathcal{C}R\mathcal{D}^{10}}{|\ln\delta|},\tag{61}$$

where $\mathcal{I}(\mathbf{t})$ is the discrete energy defined by (27), and the constant $\mathcal{C}$ does not depend on δ.

Proof. The idea of the proof is to "absorb" *all* the $O(1)$ terms in (43) as "smaller order corrections" into the $O(|\ln\delta|)$ terms and derive the estimate

$$\mathcal{I}(\mathbf{t})\leq\widehat{A}\leq\mathcal{I}(\mathbf{t})\left(1+\frac{\mathcal{C}R\mathcal{D}^{10}}{|\ln\delta|}\right).\tag{62}$$

Then (61) immediately follows from (62).

If the distributions of balls is δ-$\mathcal{D}$ connected, then from Theorems 1 and 2 we obtain the following bounds on the effective conductivity:

$$\mathcal{I}(\mathbf{t})\leq\widehat{A}\leq\frac{1}{2}\left(\sum_{\Pi_{ij}}(g_{ij}+C_1)(t_i-t_j)^2+\sum_{P_{ijk}}C_2[(t_i-t_k)^2+(t_j-t_k)^2]\right.$$
$$\left.+\sum_{\triangle_{ijkm}}C_3[(t_i-t_m)^2+(t_j-t_m)^2+(t_k-t_m)^2]\right).$$

where the constants C_1, C_2, C_3 depend on R, γ_0 but do not depend on δ and can be bounded from above by $\mathcal{C}_0\mathcal{D}^4R$ (see the proof of Theorem 2). Applying this bound we have:

$$\begin{aligned}\mathcal{I}(\mathbf{t})\leq\widehat{A}\leq\frac{1}{2}\Bigg(&\sum_{\Pi_{ij}\notin\partial\Lambda_\delta}g_{ij}(t_i-t_j)^2+\sum_{\Pi_{ij}\in\partial\Lambda_\delta}g_{ij}(t_i-t_j)^2\\&+\mathcal{C}_0\mathcal{D}^4R\Bigg\{\sum_{\Pi_{ij}}(t_i-t_j)^2+\sum_{P_{ijk}}[(t_i-t_k)^2+(t_j-t_k)^2]\\&+\sum_{\triangle_{ijkm}}[(t_i-t_m)^2+(t_j-t_m)^2+(t_k-t_m)^2]\Bigg\}\Bigg),\end{aligned}\tag{63}$$

where $\Pi_{ij}\in\partial\Lambda_\delta$ means that we consider only the hoses along the edges of some δ-polyhedra Λ_δ of the δ-$\mathcal{D}$ partition of Y. Now we use (36) to get estimates in $\Pi_{ij}\in\partial\Lambda_\delta$ only and Lemma 4 for the upper bounds on numbers of vertices, edges and faces of δ-polyhedron. Thus, continuing (63) we obtain

$$\begin{aligned}\mathcal{I}(\mathbf{t})\leq\widehat{A}\leq\frac{1}{2}\Bigg(&\sum_{\Pi_{ij}\notin\partial\Lambda_\delta}g_{ij}(t_i-t_j)^2+\sum_{\Pi_{ij}\in\partial\Lambda_\delta}g_{ij}(t_i-t_j)^2\\&+\mathcal{C}_0\mathcal{D}^4R\frac{(\mathcal{D}+2)^3}{8}\left[\#\Pi_{\Lambda_\delta}+2\#P_{\Lambda_\delta}+3\#\triangle_{\Lambda_\delta}\right]\sum_{\Pi_{ij}\in\partial\Lambda_\delta}(t_i-t_j)^2\Bigg)\end{aligned}$$

$$\leq \frac{1}{2}\left(\sum_{\Pi_{ij} \notin \partial \Lambda_\delta} g_{ij}(t_i - t_j)^2 + \sum_{\Pi_{ij} \in \partial \Lambda_\delta} g_{ij}(t_i - t_j)^2 \right.$$

$$\left. + \mathcal{C}_1 \mathcal{D}^7 R \left[K_1 + 2K_2 + 3K_3\right] \mathcal{D}^3 \sum_{\Pi_{ij} \in \partial \Lambda_\delta} (t_i - t_j)^2 \right)$$

$$\leq \frac{1}{2}\left(\sum_{\Pi_{ij} \notin \partial \Lambda_\delta} g_{ij}(t_i - t_j)^2 + \sum_{\Pi_{ij} \in \partial \Lambda_\delta} (g_{ij} + \mathcal{C}\mathcal{D}^{10} R)(t_i - t_j)^2 \right)$$

$$\leq \frac{1}{2}\left[1 + \frac{\mathcal{C}R\mathcal{D}^{10}}{|\ln \delta|}\right] \sum_{\Pi_{ij}} g_{ij}(t_i - t_j)^2 = \left[1 + \frac{\mathcal{C}R\mathcal{D}^{10}}{|\ln \delta|}\right] \mathcal{I}(\mathbf{t}).$$

Hence, combining all the bounds we obtain (62):

$$\mathcal{I}(\mathbf{t}) \leq \widehat{A} \leq \left[1 + \frac{\mathcal{C}R\mathcal{D}^{10}}{|\ln \delta|}\right] \mathcal{I}(\mathbf{t}).$$

□

Note that the difference between last two theorems is in the following. The hypothesis of Theorem 4 (that is, the δ-$\mathcal{D}$ connectedness of the graph $\mathcal{G}$) is much stronger than one of Theorem 3 therewith it provides stronger results which are as follows. The existence of the δ-$\mathcal{D}$ partition $\{\Lambda_\delta\}$ of Y implies an existence of the spanning cluster. While Theorem 3 gives only an order of magnitude of an error of the discrete network approximation (which is of order $|\ln \delta|$), Theorem 4 provides an explicit error estimate. It states that the leading term of the asymptotics of $\widehat{A}$ is equal to the discrete energy $\mathcal{I}$ for any number of inclusions N in the composite. Furthermore, this theorem provides a finite error in the limiting case as $N \to \infty$, while in Theorem 3 the error grows together with N.

5 Numerical Illustration

We now apply the discrete network approximation (27)-(29) to show numerically that for a fixed volume fraction the sparsity (presence of void spaces) increases the effective conductivity in the densely packed composites. We compare the effective conductivities of two types of composites with same volume fraction of inclusions. The first type corresponds to the periodic arrays of inclusions. The second type corresponds to irregular (non-periodic) arrays with void spaces (holes).

Our numerical experiment is designed for the face centered cubic packing (*FCC*-packing) of spheres (see Fig. 13), which is known to have the densest possible packing of identical spheres in 3D [9]. First we describe such a periodic array.

Consider a cube and let its vertices and the centers of its faces be centers of 14 spheres, packed as shown in Fig. 13 where the periodicity cell for such a construction is presented. This cubic cell contains eight $1/8$-spheres, centered at each cube vertex,

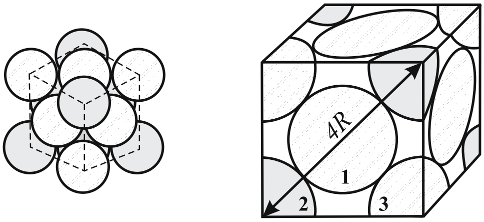


Fig. 13. Periodicity cell for the face-centered cubic array of spheres

and six hemispheres, centered at each center of the cube face. In such an arrangement of spheres the third layer repeats the first layer in the following sense. Consider two plane lattices formed by the centers of the spheres in the first and third layers, respectively. Then the perpendicular projections of these lattices on a horizontal plane are the same.

The *FCC*-packing is a solution to a close packing problem of finding the densest packing of spheres in three dimensions. The packing density of this type is $f = \dfrac{\pi}{3\sqrt{2}} \approx 0.7405$ [15], that is, the volume fraction of inclusions when they are touching.

We choose the minimal relative interparticle distance for this case to be $\delta = 5 \cdot 10^{-8}$ and the radius of the spheres to be $R = 0.2$ (the corresponding minimal distance between two neighboring spheres in the *FCC*-packing is $d = 10^{-8}$). Note that the minimal interparticle distance for the *FCC*-packing is attained between diagonally placed inclusions. For example, neighboring inclusions 1 and 2 are diagonally placed and distance between them is d, whereas neighboring inclusions 2 and 3 in Fig. 13 are not diagonally placed and distance between them is larger than d. Furthermore, d and f are related as follows:

$$d = 2R\left(\sqrt[3]{\frac{\pi}{3\sqrt{2}f}} - 1\right). \qquad (64)$$

Now we are going to describe a model example which illustrates our main point that for highly packed composites, sparsity leads to a very significant increase in the effective conductivity.

We run two sets of experiments.

E1. For the periodic *FCC*-packing of spheres of $R = 0.2$ in the box $Y = [-2,2] \times [-2,2] \times [-1,1]$ with volume fraction f varying from 0.46 to 0.74 with step 0.02, we compute the effective conductivity using the discrete approximation:

$$\widehat{A} \simeq \mathcal{I}(\mathbf{t}),$$

where $\mathbf{t}$ is the solution of the discrete minimization problem (27)-(29). For this experiment the effective conductivity is denoted by A_1 and depicted as Plot 1 in Fig. 14.

E2. (a) For the given relative interparticle distance $\delta = \dfrac{d}{R} = 5 \cdot 10^{-8}$, we compute the volume fraction f_0 of the periodic *FCC*-packing of inclusions from (64).

(b) Here we want to remove inclusions from the periodic *FCC*-packing of step E2(a), *randomly* choosing inclusions to remove. We start from the volume fraction $f_0 \geq f$ and remove at random inclusions so that the total volume fraction of removed inclusions is f_r. We choose f_r so that the total volume fraction of the remaining inclusions $f = f_0 - f_r$ is the same as in the previous experiment (E1). Thus we obtain irregular (non-periodic, with holes) array of inclusions with the same volume fraction as the periodic array. Then we can estimate the effective conductivity due to sparsity (non-periodicity). For the random removal of inclusions we use the Matlab® function *randperm*(N) ("random permutation") that changes randomly the number of an inclusion in a sequence of numbers of inclusions, where N is the number of inclusions in this experiment. Then we remove inclusions of the volume fraction f_r. Our objective is to compare the effective conductivity A_1 for the periodic *FCC*-array of inclusions at volume fraction $f = 0.46 \ldots 0.74$ with the effective conductivity for the non-periodic array, denoted by A_2, when f varies in the same interval.

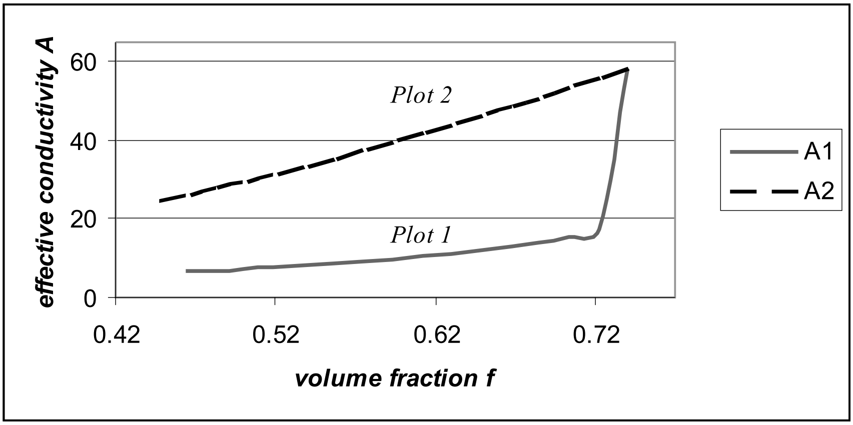


Fig. 14. Comparison of the effective conductivity A_1 for *FCC*, uniform, packing of inclusions, with the effective conductivity A_2 of composites with void spaces (non-uniform array of inclusions)

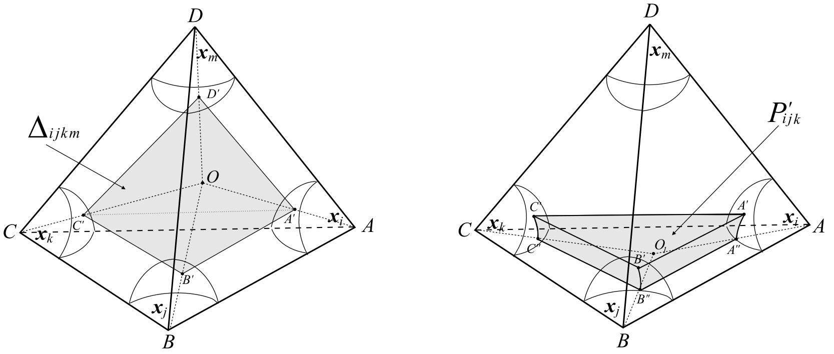
For each fixed f_r we make at least 40 "removal" experiments to collect statistics. For each of 40 configurations we compute the effective conductivity. The average of these values is the effective conductivity A_2 of a non-uniform composite. A_2 is given by Plot 2 in Fig. 14. Notice that the range of the volume fraction f from 0.46 to 0.74 is exactly the same as in the set of experiments E1. In the set of experiments E2 the relative interparticle distance δ between neighbors is either very small or greater than 1, whereas in the experiment E1 the distance between the inclusions is always the same (less than 1) for every fixed volume fraction f. Also note that that the configurations obtained in the "removal" experiments are non-uniform in the sense that they contain holes, described in Introduction (see Fig. 1).

Even though the total volume fraction in periodic and non-uniform configurations are the same the effective conductivity A_2 is approximately 3 times larger than A_1. We also estimate the error (2) of our approximation (27)-(29) by using the upper and lower bounds for both A_1 and A_2. Our preliminary results show that the error for A_2 of the sparse (non-periodic) configuration is larger than for A_1. At the same time the errors for both A_1 and A_2 are significantly smaller than the corresponding error in the analogous two-dimensional problem of randomly distributed disks, estimated in [7]. We are currently working on understanding this observation.

6 Appendices

6.1 Appendix A

Here we present an alternative construction of the hoses, prisms and tetrahedra using four pairwise neighbors $\boldsymbol{x}_i$, $\boldsymbol{x}_j$, $\boldsymbol{x}_k$ and $\boldsymbol{x}_m$, depicted in Fig. 15, 16. The vertices $\boldsymbol{x}_i = A$, $\boldsymbol{x}_j = B$, $\boldsymbol{x}_k = C$ and $\boldsymbol{x}_m = D$ being connected form a tetrahedron $T = ABCD$. The point O is an intersection of the bisector planes to the edges of the tetrahedron. This point O is the Voronoi vertex $\mathcal{V}$. The projections of O on four corresponding spheres are: $\pi_i(\mathcal{V}) = A'$, $\pi_j(\mathcal{V}) = B'$, $\pi_k(\mathcal{V}) = C'$, $\pi_m(\mathcal{V}) = D'$. They form a tetrahedron $A'B'C'D'$ which is one of the tetrahedra in the central projection partition of the domain Q, denoted by Δ_{ijkm}.



(a) the tetrahedron $A'B'C'D'$

(b) the prism $A'B'C'C''B''A''$

Fig. 15. The tetrahedron Δ_{ijkm} and the part of a prism P'_{ijk}

Each prism contains two parts. To obtain one of them, for instance, the prism P'_{ijk} between three neighbors centered at $\boldsymbol{x}_i$, $\boldsymbol{x}_j$, $\boldsymbol{x}_k$ (A, B, C), lying inside T and shown in Fig. 15(b), we consider the Voronoi edge $\mathcal{E}$ that intersects $\triangle ABC$ at the point O_1. The line segment OO_1 is the part of $\mathcal{E}$ lying inside T. The point O_1 is the intersection of the bisectors to the edges of the triangle $\triangle ABC$. The arcs $A'A''$, $B'B''$, and $C'C''$ are the projections of OO_1 on the corresponding spheres ∂B_i, ∂B_j and ∂B_k. The figure $A'B'C'C''B''A''$ obtained is the part of the prism P_{ijk} that lies inside of T.

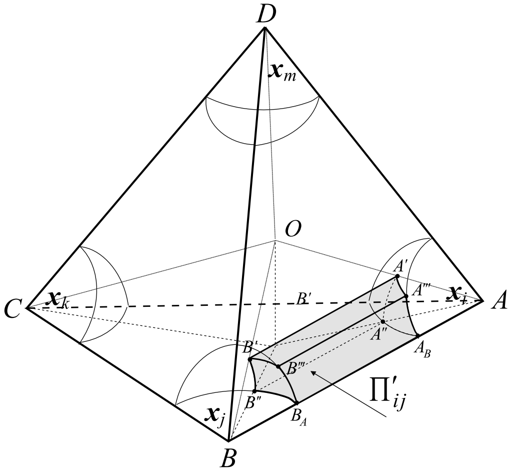


Fig. 16. The part of the hose $\Pi'_{ij} = A'A''A_BA'''B'''B_AB''B'$

Finally, we construct as an example, the hose Π_{ij}, connecting two neighbors centered at $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ (A and B in Fig. 16). The points A_B and B_A are the intersections of the line segment connecting A and B with corresponding spheres ∂B_i and ∂B_j. Similar to the point O_1, the point O_2 is the intersection of the bisectors to the edges of the triangle $\triangle ABD$, and the points A''' and B''' (Fig. 16) are the intersections of O_2A and O_2B with the corresponding spheres ∂B_i, ∂B_j. The obtained figure $A'A''A_BA'''B'''B_AB''B'$ is then the part of the hose Π'_{ij} that lies inside of T.

6.2 Appendix B

Here we estimate the number of the tetrahedra, prisms, and hoses in Λ_δ in terms of the parameter $\mathcal{D}$.

Lemma 6. *For any δ-polyhedron Λ_δ of the δ-$\mathcal{D}$ partition of Y of the δ-subgraph $\mathcal{G}_\delta$, the number of tetrahedra $\#\triangle_{\Lambda_\delta}$, the number of prisms $\#P_{\Lambda_\delta}$, the number of hoses $\#\Pi_{\Lambda_\delta}$ in Int Λ_δ satisfy:*

$$\begin{aligned}(a)\ \#\triangle_{\Lambda_\delta} &\leq \frac{3}{2}\mathcal{D}^3,\\ (b)\ \#P_{\Lambda_\delta} &\leq 3\mathcal{D}^3,\\ (c)\ \#\Pi_{\Lambda_\delta} &\leq \frac{3}{2}\mathcal{D}^3.\end{aligned} \tag{65}$$

Proof. By the $3D$ Euler formula:

$$\#\boldsymbol{x}_{\Lambda_\delta} - \#\Pi_{\Lambda_\delta} + \#P_{\Lambda_\delta} - \#\triangle_{\Lambda_\delta} = 1, \tag{66}$$

where $\#\boldsymbol{x}_{\Lambda_\delta}$ is the number of vertices $\boldsymbol{x}_i$ in Λ_δ. Since each tetrahedron has 4 faces and each face belongs to at most two tetrahedra we have the following bound:

$$\#P_{\Lambda_\delta} \geq 2\#\triangle_{\Lambda_\delta}.$$

Since each edge has 2 vertices and each vertex can belong to not more than 12 edges (due to the "kissing" number in $3D$ [9]) we obtain:

$$\#\boldsymbol{x}_{\Lambda_\delta} \geq \frac{\#\Pi_{\Lambda_\delta}}{6}.$$

From (66) we have

$$\#\boldsymbol{x}_{\Lambda_\delta} + \#P_{\Lambda_\delta} = \#\triangle_{\Lambda_\delta} + \#\Pi_{\Lambda_\delta} + 1 \leq \frac{\#P_{\Lambda_\delta}}{2} + 6\#\boldsymbol{x}_{\Lambda_\delta} + 1,$$

which implies

$$\frac{\#P_{\Lambda_\delta}}{2} \geq 5\#\boldsymbol{x}_{\Lambda_\delta} + 1 < 6\#\boldsymbol{x}_{\Lambda_\delta}.$$

Thus, we obtain

$$\#\Pi_{\Lambda_\delta} \leq 6\#\boldsymbol{x}_{\Lambda_\delta}, \quad \#P_{\Lambda_\delta} \leq 12\#\boldsymbol{x}_{\Lambda_\delta}, \quad \text{and} \quad \#\triangle_{\Lambda_\delta} \leq 6\#\boldsymbol{x}_{\Lambda_\delta}.$$

The volume of Λ_δ is bounded from above by the volume of the ball of diameter $\mathcal{D}R$. Let M be the number of balls of radius R that can be placed into this ball. Then $\frac{4}{3}M\pi R^3 \leq \frac{\pi}{3}\mathcal{D}^3R^3$, from which we find $\#\boldsymbol{x}_{\Lambda_\delta} \leq M \leq \frac{\mathcal{D}^3}{4}$. Then from the above bounds on the number of tetrahedra, prisms and hoses we obtain (65). □

6.3 Appendix C

Here we give the estimates on the Dirichlet integral of the function u defined by (47).

Lemma 7. *Let the function u be given by (47). Then the Dirichlet integral of this function over the hose Π_{ij} is bounded by*

$$\int_{\Pi_{ij}} |\nabla u|^2 \mathrm{d}\boldsymbol{x} \leq \left(g_{ij} - 2\pi R \ln\left(\frac{\ell}{2\rho}\right) + 4\pi R\left(1 - \frac{\ell}{2\rho}\right)\right)(t_i - t_j)^2. \tag{67}$$

Proof. If u is given by (47) then its gradient is

$$\nabla u(\boldsymbol{x}) = \left(\frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}, \frac{\partial u}{\partial z}\right) = (t_i - t_j)\left(-\frac{z(H_{ij})_x}{H_{ij}^2}, -\frac{z(H_{ij})_y}{H_{ij}^2}, \frac{1}{H_{ij}}\right). \tag{68}$$

We need to estimate the Dirichlet integral of u over the hose Π_{ij}. Let $\pi_j(\mathcal{F})$ be the lower base of the hose Π_{ij} (see Fig. 9). Recall that the base of the hose is a convex curvilinear polygon (the projection of the convex polygon on the sphere). Let S be the projection on the xy-plane of the longest line segment connecting the point B_A with all the vertices of this curvilinear polygon, referred to as the *half-width* of the hose (in Fig. 9 the longest line segment connecting B_A with other vertices of $\pi_j(\mathcal{F})$ is B_AB_O, and its projection on xy-plane is equal to S). Then we obtain the following estimate:

$$\begin{aligned}(t_i-t_j)^2\int_{\Pi_{ij}}\left(\frac{\partial u}{\partial z}\right)^2\mathrm{d}\boldsymbol{x}&=(t_i-t_j)^2\int_{\Pi_{ij}}\frac{1}{(H_{ij})^2}\mathrm{d}\boldsymbol{x}\\ =\ (t_i-t_j)^2\int_{\pi_j(\mathcal{F})}\frac{\mathrm{d}x\,\mathrm{d}y}{H_{ij}}&=g_{ij}(t_i-t_j)^2,\end{aligned}\tag{69}$$

using the formula (22).

Next consider

$$\begin{aligned}&(t_i-t_j)^2\int_{\Pi_{ij}}\left[\left(\frac{\partial u}{\partial x}\right)^2+\left(\frac{\partial u}{\partial y}\right)^2\right]\mathrm{d}\boldsymbol{x}\\ =\ &(t_i-t_j)^2\left(\int_{\Pi_{ij}}\frac{z^2(H_{ij})_x^2}{(H_{ij})^4}\mathrm{d}\boldsymbol{x}+\int_{\Pi_{ij}}\frac{z^2(H_{ij})_y^2}{(H_{ij})^4}\mathrm{d}\boldsymbol{x}\right)\\ \le\ &\frac{(t_i-t_j)^2}{4}\left(\int_{\pi_j(\mathcal{F})}\left[\frac{(H_{ij})_x^2}{H_{ij}}+\frac{(H_{ij})_y^2}{H_{ij}}\right]\mathrm{d}x\,\mathrm{d}y\right).\end{aligned}\tag{70}$$

Here we used the fact that $z^2\le\dfrac{H_{ij}^2}{4}$ since $z\in\left(-\dfrac{H_{ij}}{2},\dfrac{H_{ij}}{2}\right)$.

Also,

$$(H_{ij})_x^2=\frac{16\,x^2}{R^2-x^2-y^2},$$

so

$$\begin{aligned}\frac{(H_{ij})_x^2}{H_{ij}}&=\frac{16x^2}{R^2-x^2-y^2}\frac{1}{\delta_{ij}R+2R-2\sqrt{R^2-x^2-y^2}}\\ \le\frac{8x^2}{R^2-x^2-y^2}\frac{1}{R-\sqrt{R^2-x^2-y^2}}&=\frac{8x^2}{R^2-x^2-y^2}\frac{R+\sqrt{R^2-x^2-y^2}}{x^2+y^2}.\end{aligned}$$

The estimate for $\dfrac{(H_{ij})_y^2}{H_{ij}}$ can be obtained Similarly. Continuing (70), we get

$$
\begin{aligned}
&(t_i-t_j)^2\int_{\Pi_{ij}}\left[\left(\frac{\partial u}{\partial x}\right)^2+\left(\frac{\partial u}{\partial y}\right)^2\right]\mathrm{d}\boldsymbol{x}\\
\le\ &\frac{(t_i-t_j)^2}{4}\int_{\pi_j(\mathcal{F})}\frac{8x^2+8y^2}{R^2-x^2-y^2}\,\frac{R+\sqrt{R^2-x^2-y^2}}{x^2+y^2}\,\mathrm{d}x\,\mathrm{d}y\\
=\ &2(t_i-t_j)^2\int_{\pi_j(\mathcal{F})}\frac{R+\sqrt{R^2-x^2-y^2}}{R^2-x^2-y^2}\,\mathrm{d}x\,\mathrm{d}y\\
\le\ &2(t_i-t_j)^2\int_{C_j}\frac{R+\sqrt{R^2-x^2-y^2}}{R^2-x^2-y^2}\,\mathrm{d}x\,\mathrm{d}y,
\end{aligned}
$$

where C_j is a spherical cap of the sphere ∂B_j centered at B_A, the radius of which equals the half-width S of the hose Π_{ij}. Then

$$
\begin{aligned}
&(t_i-t_j)^2\int_{\Pi_{ij}}\left[\left(\frac{\partial u}{\partial x}\right)^2+\left(\frac{\partial u}{\partial y}\right)^2\right]\mathrm{d}\boldsymbol{x}\\
\le\ &4\pi(t_i-t_j)^2\int_0^S\frac{R+\sqrt{R^2-r^2}}{R^2-r^2}r\,\mathrm{d}r \qquad (71)\\
=\ &\left[-2\pi R\ln\left(1-\frac{S^2}{R^2}\right)+4\pi R\left(1-\sqrt{1-\frac{S^2}{R^2}}\right)\right](t_i-t_j)^2.
\end{aligned}
$$

Now our goal is to express the right hand side of the formula (71) in terms of the quotient $\frac{\rho}{\ell}$. Recall that the z-axis is directed along AB. Then the projection $B'\tilde{B}$ (Fig. 17) of the line segment $B'B_A$ on the xy-plane is less or equal to S. We consider a hose in which the length of $B'\tilde{B}$ is exactly S. In the triangle $\triangle BB'\tilde{B}$ the circumradius ρ of the tetrahedron $ABCD$ can be written as $\rho=|BO|=\dfrac{|BM|}{\cos\angle OBM}$, where M is the midpoint of AB. From $\triangle BB'\tilde{B}$ we have:

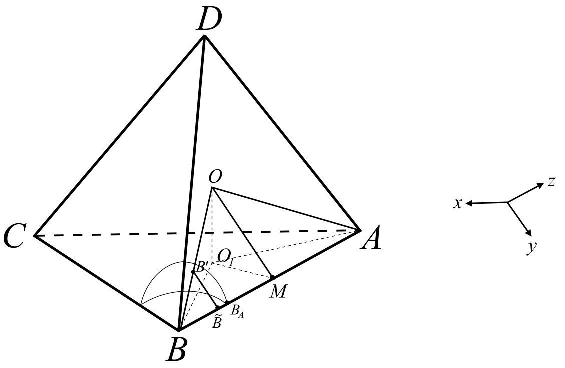


Fig. 17. The construction in the triangle $\triangle BB'\tilde{B}$

$$
|BB'|=R,\quad |B'\tilde{B}|=S;\quad \text{and}\quad \frac{S}{R}=\sin\angle OBM,
$$

so

$$\sqrt{1-\frac{S^2}{R^2}} = \cos\angle OBM = \frac{|BM|}{\rho}.$$

Since $|BM| \geq \frac{\ell}{2}$, where ℓ is the smallest edge of the tetrahedron $ABCD$, we obtain from (71)

$$-2\pi R \ln\left(1-\frac{S^2}{R^2}\right)+4\pi R\left(1-\sqrt{1-\frac{S^2}{R^2}}\right) \leq -2\pi R \ln\left(\frac{\ell^2}{4\rho^2}\right)+4\pi R\left(1-\frac{\ell}{2\rho}\right).$$

Hence

$$\int_{\Pi_{ij}} |\nabla u|^2 \mathrm{d}\boldsymbol{x} \leq \left(g_{ij}+2\pi R \ln\left(\frac{2\rho}{\ell}\right)+4\pi R\left(1-\frac{\ell}{2\rho}\right)\right)(t_i-t_j)^2. \qquad (72)$$

□

6.4 Appendix D

Here we prove the bound (51).

Proof. Denote by θ the smallest angle in the triangle $A'B'C'$. Note that the triangle $A'B'C'$ is similar to the triangle ABC, which is one of the faces of the tetrahedron $T \in \{T\}$ (see Fig. 15). So we find the estimate for $\sin\theta$ from the triangle ABC. Thus θ is the smallest angle of $\triangle ABC$ ($\angle CAB = \theta$ in Fig. 18). Let $\theta = \theta_1 + \theta_2 = \angle CAO_1 + \angle BAO_1$. Recall that the point O_1 is the point of the intersection of the bisectors of $\triangle ABC$. Denote $\angle CBO_1$ by θ_3, and we see that $\theta_1 + \theta_2 + \theta_3 = \frac{\pi}{2}$.

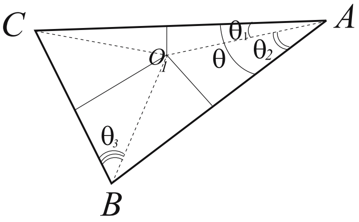


Fig. 18. The triangle ABC

From $\triangle CBO_1$:

$$\cos\theta_3 = \sin(\theta_1+\theta_2) = \sin\theta = \frac{|BC|}{2|BO_1|}.$$

Note that $|BO_1| \leq \rho$ then $\sin\theta \geq \frac{|BC|}{2\rho}$. Since $|BC| \geq \ell$, we get $\sin\theta \geq \frac{1}{2\gamma}$, where $\gamma = \frac{\rho}{\ell}$. Hence, the formula (51) is proved. □

6.5 Appendix E

Lemma 8. *Let four different points $\boldsymbol{A}_i$, $i = 1, \ldots, 4$ form a tetrahedron T'. Let u be a linear function, taking values t_i at $\boldsymbol{A}_i$, respectively, $i = 1, \ldots, 4$. Then its Dirichlet integral over T' is bounded as follows:*

$$\int_{T'} |\nabla u|^2 \mathrm{d}\boldsymbol{x} \le \frac{1}{12\,|T'|} \Big\{ |\boldsymbol{A}_2 \times \boldsymbol{A}_3|^2 (t_1 - t_4)^2 + |\boldsymbol{A}_3 \times \boldsymbol{A}_1|^2 (t_2 - t_4)^2 + |\boldsymbol{A}_1 \times \boldsymbol{A}_2|^2 (t_3 - t_4)^2 \Big\}, \tag{73}$$

where $|T'|$ is the volume of the tetrahedron T'.

Proof. Consider the tetrahedron $T' = \boldsymbol{A}_1\boldsymbol{A}_2\boldsymbol{A}_3\boldsymbol{A}_4$ (see Fig. 19) with values of the potentials t_i at the points $\boldsymbol{A}_i$ $(i = 1, \ldots, 4)$. We choose a coordinate system so that the point $\boldsymbol{A}_4$ is the origin.

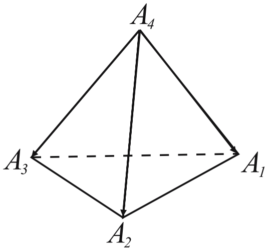


Fig. 19. The tetrahedron T'

Inside the tetrahedron T' the trial function $u(x,\, y,\, z)$ is linear, that is, $u(x,\, y,\, z) = ax + by + cz + d$, where the constants a, b, c and d are to be determined. If we denote $\boldsymbol{K} = (a,\, b,\, c)$ and $\boldsymbol{X} = (x,\, y,\, z)$ then the function $u = \boldsymbol{K} \cdot \boldsymbol{X} + d$. Since $u(0, 0, 0) = t_4$ then $d = t_4$ and

$$\boldsymbol{K} \cdot \boldsymbol{A}_i + t_4 = t_i, \text{ where } \boldsymbol{A}_i = (x_i,\, y_i,\, z_i),\ i = 1,\, 2,\, 3.$$

If we denote

$$\tau_i = t_i - t_4 \ \ (i = 1,\, 2,\, 3),\ \tau_4 = 0 \quad \text{then } \boldsymbol{K} \cdot \boldsymbol{A}_i = \tau_i, \quad i = 1,\, 2,\, 3. \tag{74}$$

From what we find

$$\boldsymbol{K} \cdot (\tau_2 \boldsymbol{A}_1 - \tau_1 \boldsymbol{A}_2) = 0 \quad \text{and} \quad \boldsymbol{K} \cdot (\tau_3 \boldsymbol{A}_2 - \tau_2 \boldsymbol{A}_3) = 0. \tag{75}$$

The formula (75) implies that $\boldsymbol{K} \| \, (\tau_2 \boldsymbol{A}_1 - \tau_1 \boldsymbol{A}_2) \times (\tau_3 \boldsymbol{A}_2 - \tau_2 \boldsymbol{A}_3)$ or

$$(\tau_2 \boldsymbol{A}_1 - \tau_1 \boldsymbol{A}_2) \times (\tau_3 \boldsymbol{A}_2 - \tau_2 \boldsymbol{A}_3) = \alpha \boldsymbol{K}, \tag{76}$$

where α is some constant to be determined. From equation (76) we have

$$\boldsymbol{K} = \frac{\tau_2\tau_3\boldsymbol{A}_1 \times \boldsymbol{A}_2 + \tau_2^2\boldsymbol{A}_3 \times \boldsymbol{A}_1 + \tau_1\tau_2\boldsymbol{A}_2 \times \boldsymbol{A}_3}{\alpha}. \tag{77}$$

Now substitute (77) into, for example, $\boldsymbol{K} \cdot \boldsymbol{A}_1 = \tau_1$ we get

$$\frac{\tau_2\tau_3\boldsymbol{A}_1 \times \boldsymbol{A}_2 + \tau_2^2\boldsymbol{A}_3 \times \boldsymbol{A}_1 + \tau_1\tau_2\boldsymbol{A}_2 \times \boldsymbol{A}_3}{\alpha} \cdot \boldsymbol{A}_1 = \tau_1. \tag{78}$$

We make the definition $\det \mathcal{A} := \det\, [\boldsymbol{A}_1\ \boldsymbol{A}_2\ \boldsymbol{A}_3]$, then simplifying (78) we find that

$$\alpha = \tau_2 \boldsymbol{A}_1 \cdot \boldsymbol{A}_2 \times \boldsymbol{A}_3 = \tau_2 \det \mathcal{A}.$$

Hence

$$\boldsymbol{K} = \frac{\tau_3\boldsymbol{A}_1 \times \boldsymbol{A}_2 + \tau_1\boldsymbol{A}_2 \times \boldsymbol{A}_3 + \tau_2\boldsymbol{A}_3 \times \boldsymbol{A}_1}{\det \mathcal{A}} = (a,\, b,\, c). \tag{79}$$

Recall that our goal here is to calculate the Dirichlet integral of the function $u = \boldsymbol{K} \cdot \boldsymbol{X} + t_4$ over the tetrahedron T'. Then we need to consider $|\nabla u|^2 = |\boldsymbol{K}|^2$:

$$|\boldsymbol{K}|^2 = \frac{|\tau_3\boldsymbol{A}_1 \times \boldsymbol{A}_2 + \tau_1\boldsymbol{A}_2 \times \boldsymbol{A}_3 + \tau_2\boldsymbol{A}_3 \times \boldsymbol{A}_1|^2}{|\det \mathcal{A}|^2}. \tag{80}$$

For any real $x,\ y,\ z$, the following inequality holds: $(x+y+z)^2 \leq 3x^2+3y^2+3z^2$. Continue formula (80) with this inequality gives

$$|\boldsymbol{K}|^2 \leq \frac{3}{|\det \mathcal{A}|^2}\left(\tau_3^2\,|\boldsymbol{A}_1 \times \boldsymbol{A}_2|^2 + \tau_1^2\,|\boldsymbol{A}_2 \times \boldsymbol{A}_3|^2 + \tau_2^2\,|\boldsymbol{A}_3 \times \boldsymbol{A}_1|^2\right).$$

Now we obtain the estimate for the Dirichlet integral of the trial function u in the tetrahedron T':

$$\int_{T'} |\nabla u|^2 \mathrm{d}\boldsymbol{x} \leq \frac{3}{|\det \mathcal{A}|^2}\,|T'|\left(\tau_3^2\,|\boldsymbol{A}_1 \times \boldsymbol{A}_2|^2 + \tau_1^2\,|\boldsymbol{A}_2 \times \boldsymbol{A}_3|^2 + \tau_2^2\,|\boldsymbol{A}_3 \times \boldsymbol{A}_1|^2\right). \tag{81}$$

Note that the volume of the tetrahedron is $|T'| = \dfrac{1}{6}\det \mathcal{A}$, which yields (81):

$$\int_{T'} |\nabla u|^2 \mathrm{d}\boldsymbol{x} \leq \frac{1}{2\,|\det \mathcal{A}|}\left(\tau_3^2\,|\boldsymbol{A}_1 \times \boldsymbol{A}_2|^2 + \tau_1^2\,|\boldsymbol{A}_2 \times \boldsymbol{A}_3|^2 + \tau_2^2\,|\boldsymbol{A}_3 \times \boldsymbol{A}_1|^2\right), \tag{82}$$

and using (82) and the definition of τ_i (74), formula (73) follows. □

6.6 Appendix F

Here we prove that two integrals over $\triangle A'B'C'$ and over $\triangle\widetilde{A}\widetilde{B}\widetilde{C}$ in (49) are equal. Since the potential function is linear in x and z in every cross-section we will construct it in both $\triangle\widetilde{A}\widetilde{B}\widetilde{C}$ and $\triangle A'B'C'$ and compare the Dirichlet integrals of it over these triangles. Let $u = \alpha x+\beta z+\gamma$ in $\triangle\widetilde{A}\widetilde{B}\widetilde{C}$ and $u' = ax'+bz'+c$ in $\triangle A'B'C'$.

The function u takes values t_i, t_j, t_k at points $\widetilde{A} = (x_i, y_i, z_i)$, $\widetilde{B} = (x_j, y_j, z_j)$, $\widetilde{C} = (x_k, y_k, z_k)$, respectively, that is,

$$t_i = \alpha x_i + \beta z_i + \gamma, \quad t_j = \alpha x_j + \beta z_j + \gamma, \quad t_k = \alpha x_k + \beta z_k + \gamma.$$

Analogously, u' takes values t_i, t_j, t_k at points $A' = (x'_i, y'_i, z'_i)$, $B' = (x'_j, y'_j, z'_j)$, $C' = (x'_k, y'_k, z'_k)$, respectively, that is,

$$t_i = ax'_i + bz'_i + c, \quad t_j = ax'_j + bz'_j + c, \quad t_k = ax'_k + bz'_k + c.$$

Then one can find the coefficients α, β, a and b :

$$\begin{aligned}
\alpha &= \frac{t_i(z_j - z_k) + t_j(z_k - z_i) + t_k(z_i - z_j)}{x_i(z_j - z_k) + x_j(z_k - z_i) + x_k(z_i - z_j)}, \\
\beta &= \frac{t_i(x_j - x_k) + t_j(x_k - x_i) + t_k(x_i - x_j)}{x_i(z_j - z_k) + x_j(z_k - z_i) + x_k(z_i - z_j)}, \\
\gamma &= \frac{t_i(z_k x_j - x_k z_j) + t_j(z_i x_k - x_i z_k) + t_k(x_i z_j - x_j z_i)}{x_i(z_j - z_k) + x_j(z_k - z_i) + x_k(z_i - z_j)},
\end{aligned}$$

and

$$\begin{aligned}
a &= \frac{t_i(z'_j - z'_k) + t_j(z'_k - z'_i) + t_k(z'_i - z'_j)}{x'_i(z'_j - z'_k) + x'_j(z'_k - z'_i) + x'_k(z'_i - z'_j)}, \\
b &= \frac{t_i(x'_j - x'_k) + t_j(x'_k - x'_i) + t_k(x'_i - x'_j)}{x'_i(z'_j - z'_k) + x'_j(z'_k - z'_i) + x'_k(z'_i - z'_j)}, \\
c &= \frac{t_i(z'_k x'_j - x'_k z'_j) + t_j(z'_i x'_k - x'_i z'_k) + t_k(x'_i z'_j - x'_j z'_i)}{x'_i(z'_j - z'_k) + x'_j(z'_k - z'_i) + x'_k(z'_i - z'_j)}.
\end{aligned}$$

Therefore, when $x = \dfrac{x'}{k}$ and $z = \dfrac{z'}{k}$ we get $\alpha = ak$ and $\beta = bk$. Thus, $u = akx' + bkz' + c$ and $\nabla u = k\nabla u'$. Since $\triangle A'B'C'$ and $\triangle \widetilde{A}\widetilde{B}\widetilde{C}$ are congruent then $|\widetilde{A}\widetilde{B}| = \dfrac{|A'B'|}{k}$, $|\widetilde{B}\widetilde{C}| = \dfrac{|B'C'|}{k}$, and, $|\triangle \widetilde{A}\widetilde{B}\widetilde{C}| = \dfrac{|\triangle A'B'C'|}{k^2}$.

Hence we obtain

$$\begin{aligned}
&\int_{P_{ijk}} |\nabla u|^2 \mathrm{d}\boldsymbol{x} = \int_0^h \mathrm{d}y \int_{\triangle \widetilde{A}\widetilde{B}\widetilde{C}} |\nabla u|^2 \mathrm{d}x\, \mathrm{d}z \\
&= \int_0^h \mathrm{d}y \int_{\triangle A'B'C'} |\nabla u'|^2 k^2 \,\frac{\mathrm{d}x'}{k}\,\frac{\mathrm{d}z'}{k} = \int_0^h dy \int_{\triangle A'B'C'} |\nabla u'|^2 \mathrm{d}x'\, \mathrm{d}z'. \qquad \square
\end{aligned}$$

7 Acknowledgments

The work of L. Berlyand and Y. Gorb was supported by NSF grant DMS-0204637.

The authors thank A. Panchenko for careful reading of the manuscript and useful suggestions.

References

1. Aizenman, M., Kesten, H., Newman, C.M.: Uniqueness of the infinite cluster and continuity of connectivity functions for short and long range percolation. Comm. Math. Phys., **111**, 505–532 (1987)
2. Aurenhammer, F., Klein, R.: Voronoi Diagrams. In: Sack J. and Urrutia G. (ed) Handbook of Computational Geometry. Chapter V, 201–290. Elsevier Science Publishing (2000)
3. Berlyand, L., Borcea, L., Panchenko, A.: Network approximation for effective viscosity of concentrated suspensions with complex geometries. SIAM J. Math. Anal. (2003)
4. Bakhvalov, N.S., Panasenko, G.P.: Homogenization: Averaging Processes in Periodic Media. Kluwer, Dordrecht/Boston/London (1989)
5. Bensoussan, A., Lions, J.L., Papanicolaou, G. Asymptotic Analysis for Periodic Structure. North-Holland, Amsterdam (1978)
6. Berlyand, L., Kolpakov, A.: Network Approximation in the Limit of Small Interparticle Distance of the Effective Properties of a High Contrast Random Dispersed Composite. Arch. Rat. Math. Anal., **159:3**, 179–227 (2001)
7. Berlyand, L., Novikov, A.: Error of the Network Approximation for Densely Packed Composites with Irregular Geometry. SIAM J. Math. Anal., **34:2**, 385–408 (2002)
8. Borcea, L., Papanicolaou, G.: Network approximation for transport properties of high contrast conductivity. Inverse Problems, **15:4**, 501–539 (1998)
9. Conway , J.H., Sloane, N. J. A.: The kissing number problem, and Bounds on kissing numbers, § 1.2 and Ch. 13. In: Sphere Packings, Lattices and Groups. 2nd ed. Springer-Verlag, New York. 21–24 (1993)
10. Ekeland, I., Temam, R. *Convex Analysis and Variational problems*. North Holland: Amsterdam (1976)
11. Jikov, V.V., Kozlov, S.M., Oleinik, O.A.: Homogenization of Differential Operators and Integral Functionals. Springer-Verlag, Berlin Heidelberg (1994)
12. Keller, J.B.: Conductivity of a medium containing a dense array of perfectly conducting spheres or cylinders or nonconducting cylinders. J. Appl. Phys., **34:4**, 991–993 (1963)
13. Sanchez-Palencia, E.: Non-homogeneous Media and Vibration Theory. In: Lecture Notes in Physics, **127**. Springer-Verlag, Berlin (1980)
14. Shewchuk, J.R.: What Is a Good Linear Finite Element? Interpolation, Conditioning, Anisotropy, and Quality Measures. Unpublished preprint, (2002) available at `http://www-2.cs.cmu.edu/~jrs/jrspapers.html#quality`
15. Steinhaus, H.: Mathematical Snapshots. 3rd ed. Dover, New York. 202–203, (1999)

Adaptive Monte Carlo Algorithms for Stopped Diffusion

Anna Dzougoutov[1], Kyoung-Sook Moon[2], Erik von Schwerin[1],
Anders Szepessy[1,3], and Raúl Tempone[4]

[1] Department of Numerical Analysis and Computer Science, KTH, S–100 44 Stockholm, Sweden
annadz@kth.se, schwerin@nada.kth.se

[2] Department of Mathematics, University of Maryland, College Park, MD 20742, USA,
moon@math.umd.edu

[3] Department of Mathematics, KTH, S–100 44 Stockholm, Sweden,
szepessy@nada.kth.se

[4] ICES, The University of Texas at Austin, 1 Texas Longhorns, Austin, Texas 78712, and IMERL, Facultad de Ingeniería, Julio Herrera y Reissig 565, 11200 Montevideo Uruguay,
rtempone@ices.utexas.edu

Summary. We present adaptive algorithms for weak approximation of stopped diffusion using the Monte Carlo Euler method. The goal is to compute an expected value $E[g(X(\tau),\tau)]$ of a given function g depending on the solution X of an Itô stochastic differential equation and on the first exit time τ from a given domain. The adaptive algorithms are based on an extension of an error expansion with computable leading order term, for the approximation of $E[g(X(T))]$ with a fixed final time $T > 0$ and diffusion processes X in $\mathbb{R}^d$, introduced in [17] using stochastic flows and dual backward solutions. The main steps in the extension to stopped diffusion processes are to use a conditional probability to estimate the first exit time error and introduce difference quotients to approximate the initial data of the dual solutions. Numerical results show that the adaptive algorithms achieve the time discretization error of order N^{-1} with N adaptive time steps, while the error is of order $N^{-1/2}$ for a method with N uniform time steps.

Key words: adaptive mesh refinement algorithm, diffusion with boundary, barrier option, Monte Carlo method, weak approximation

1 Introduction

In this paper, we compute adaptive approximations of an expected value

$$E[g(X(\tau),\tau)] \tag{1}$$

of a given function, $g : D \times [0,T] \to \mathbb{R}$, where the stochastic process X solves an Itô stochastic differential equation (SDE)

$$\mathrm{d}X_i(t) = a_i(X(t))\,\mathrm{d}t + \sum_{l=1}^{l_0} b_i^l(X(t))\,\mathrm{d}W^l(t)\ ,\ i = 1,2,\ldots,d,\ t > 0 \tag{2}$$

and τ is the first exit time

$$\tau := \inf\{0 < t : (X(t),t) \notin D \times (0,T)\} \tag{3}$$

from a given open domain $D \times (0,T) \subset \mathbb{R}^d \times (0,T)$. The functions $a : \mathbb{R}^d \to \mathbb{R}^d$ and $b^l : \mathbb{R}^d \to \mathbb{R}^d$ for $l = 1,2,\ldots,l_0$, are given drift and diffusion fluxes and $W^l(t;\omega)$ for $l = 1,2,\ldots,l_0$, are independent Wiener processes. Such problems arise in physics and finance, for instance when computing the value of barrier options.

In the case when the dimension of the problem is large or when the related partial differential equation is difficult to formulate or to solve, the Monte Carlo Euler method is used to compute the expected value. The main difficulty in the approximation of the stopped (or killed) diffusion on the boundary ∂D is that a continuous sample path may exit the given domain D even though a discrete approximate solution does not cross the boundary of D. This hitting of the boundary makes the time discretization error $N^{-1/2}$ for the Monte Carlo Euler method with N uniform time steps, see [7], while the discretization error is of order N^{-1} without stopping boundary in $\mathbb{R}^d \times [0,T)$. The work [13] and [9] reduce the large $N^{-1/2}$ first exit error to N^{-1}. The idea is to generate a uniformly distributed random variable in $(0,1)$ for each time step and compare it with a known exit probability to decide if the continuous path exits the domain during this time interval. A similar method with N uniform time steps in a domain with smooth boundary is proved to converge with the rate N^{-1} under some appropriate assumptions in [8]. Different Monte Carlo methods for stopped diffusions are compared computationally in [5]. To use these methods, the exit probability needs to be computed accurately.

Inspired by Petersen and Buchmann [16], this work uses the alternative to reduce the computational error by choosing adaptively the size of the time steps near the boundary, which has the advantage that the exit probability does not need to be computed accurately. Section 2 derives an expansion of the error with computable leading order term. Section 3 presents an adaptive algorithm based on the error estimate where the time discretization error is of order N^{-1} with N adaptive time steps.

Using the Monte Carlo Euler method, the expected value (1) can be approximated by a sample average of $g(\overline{X}(\overline{\tau}),\overline{\tau})$, where $(\overline{X},\overline{\tau})$ is an Euler approximation of (X,τ). The global error can then be split into time discretization error and statistical error,

$$\begin{aligned}
&E[g(X(\tau),\tau)] - \frac{1}{M}\sum_{j=1}^{M} g(\overline{X}(\overline{\tau};\omega_j),\overline{\tau})\\
&= \big(E[g(X(\tau),\tau) - g(\overline{X}(\overline{\tau}),\overline{\tau})]\big) + \left(E[g(\overline{X}(\overline{\tau}),\overline{\tau})] - \frac{1}{M}\sum_{j=1}^{M} g(\overline{X}(\overline{\tau};\omega_j),\overline{\tau})\right)\\
&=: \mathcal{E}_T + \mathcal{E}_S
\end{aligned} \tag{4}$$

where M is the number of realizations. The statistical error, $\mathcal{E}_S$ in (4), is asymptotically bounded by $c_0\overline{\sigma}/\sqrt{M}$ using the Central Limit Theorem, where $\overline{\sigma}$ is the sample average of the standard deviation of $g(\overline{X}(\overline{\tau}), \overline{\tau})$ and c_0 is a positive constant for a confidence interval, see Sect. 3.1.

Talay and Tubaro [18] and Bally and Talay [4] prove an a priori error expansion of $E[g(X(T)) - g(\overline{X}(T))]$ for the case without stopping boundary, i.e. for diffusion processes in $\mathbb{R}^d \times [0, T]$. In the same setting without a stopping boundary, the work [17] proves an expansion of the error with computable leading order term, error $\simeq E\left[\sum_{n=1}^N r_n\right]$, using an error density, $\rho = r_n/\Delta t_n^2$, which depends on computable discrete primal and dual solutions. Given this error estimate, consider an algorithm which for each realization refines the solution, $\overline{X}$, by the adaptive time stepping:

```
for all time steps $n = 1, \ldots, N$
    if $\left(r_n \geq \frac{\mathrm{TOL}_T}{E[N]}\right)$ then
        divide $\Delta t_n$ into 2 equal substeps, and generate
        the intermediate value of $W$ by the Brownian bridge (5),
    else let the new step be the same as the old
    endif
endfor,
```

with the stopping criterion:

$$\textbf{if } \left(\max_{1\leq n\leq N} r_n < S\frac{\mathrm{TOL}_T}{E[N]}\right) \textbf{ then } \text{stop.}$$

The intermediate sample points from W are constructed by the Brownian bridge, cf. [10],

$$W^l\left(\frac{t_n + t_{n+1}}{2}\right) = \frac{1}{2}\left(W^l(t_n) + W^l(t_{n+1})\right) + z_n^l \tag{5}$$

where z_n^l are independent random variables in $N(0, (t_{n+1} - t_n)/4)$, i.e. they are normally distributed with mean 0 and variance $(t_{n+1} - t_n)/4$, independent also of previous $W^l(t_j)$. Letting c_0 be the confidence interval parameter, related to the statistical error $c_0\overline{\sigma}/\sqrt{M} \simeq \mathrm{TOL}_S$, in (4), with $\mathrm{TOL} = 3\mathrm{TOL}_T = 3\mathrm{TOL}_S/2$, and assuming $S > C$ are constants such that $C^{-1} \leq \frac{\rho_{parent}}{\rho_{child}} \leq C$, the work [15] proves that the algorithm stops with asymptotically optimal expected number of time steps and the error asymptotically bounded by TOL with large probability (up to problem independent factors):

$$E[N] \lesssim 4CE[N_{optimal}] \text{ and } P(\frac{\text{error}}{\mathrm{TOL}} \leq \frac{S}{3} + \frac{2}{3}) \gtrsim (2\pi)^{-1/2}\int_{-c_0}^{c_0} \mathrm{e}^{-x^2/2}\mathrm{d}x\ .$$

In Sect. 2, we approximate the time discretization error, $\mathcal{E}_T$ in (4), in computable form by extending the error estimate in [17] to weak approximation of stopped diffusion. As in [18] and [17], the first step to derive an error estimate is to introduce a continuous Euler path. Then the error between the exact and continuous Euler path is approximated using stochastic flows and dual backward solutions in Sect. 2.3. The main idea in this extension is to use difference quotients to replace the stochastic flows that do not exist at the boundary. The approximate error between the continuous and the discrete Euler path is derived by a conditional probability using Brownian bridges in Sect. 2.2. Note that the exit probability is used here only to decide the time steps, not to approximate the expected values directly. Therefore the accuracy of the approximation of the exit probability is not crucial.

The computation of the dual solutions may be costly in high dimension. A simplified variant of the algorithm based on the local error is obtained by replacing the dual solutions by 1.

The paper is organized as follows. The computable error estimate for stopped diffusions is derived in the next section and based on this error estimate we develop adaptive algorithms in Sect. 3. Finally some numerical results of adaptive refinements in one and two space dimension are given in Sect.4. This paper is an extension of the preprint paper 5 in [14] where stopped diffusion in one dimension is studied.

2 Error Expansion

Consider a domain $D \subset \mathbb{R}^d$ and assume that the initial position $X(0) = X_0$ lies in D. The goal is to compute the expected value $E[g(X(\tau), \tau)]$ of a given function g which depends on the stochastic process X and the first exit time τ defined in (3).

First discretize the time interval $[0, T]$ into N subintervals $0 = t_0 < t_1 < \ldots < t_N = T$ and let $\overline{X}$ denote the Euler approximation of the process X; start with $\overline{X}(0) = X_0$ and compute $\overline{X}(t_{n+1})$ for $n = 0, 1, \ldots, N-1$ by

$$\overline{X}_i(t_{n+1}) = \overline{X}_i(t_n) + a_i(\overline{X}(t_n))\, \Delta t_n + \sum_{l=1}^{l_0} b_i^l(\overline{X}(t_n))\, \Delta W_n^l, \quad i = 1, 2, \ldots, d, \tag{6}$$

where $\Delta t_n := t_{n+1} - t_n$ denote time increments and $\Delta W_n^l := W^l(t_{n+1}) - W^l(t_n)$ denote Wiener increments. Approximate the first exit time τ with

$$\overline{\tau} := \min_{1 \le n \le N} \{t_n : (\overline{X}(t_n), t_n) \notin D \times [0, T)\} \tag{7}$$

using the Euler approximation path $\overline{X}$ instead of the exact path X.

Introduce, for theoretical purposes only, a continuous Euler path $\overline{X}(t)$ by

$$\overline{X}_i(t) = \overline{X}_i(t_n) + \int_{t_n}^{t} a_i(\overline{X}(t_n))\, \mathrm{d}t + \sum_{l=1}^{l_0} \int_{t_n}^{t} b_i^l(\overline{X}(t_n))\, \mathrm{d}W_t^l, \quad i = 1, 2, \ldots, d, \tag{8}$$

for $t \in [t_n, t_{n+1})$ and denote by

$$\widetilde{\tau} := \inf\{0 < t : (\overline{X}(t), t) \notin D \times [0, T)\} \tag{9}$$

the exit time of the continuous Euler path. Then the time discretization error of the Euler approximation can be split in two parts:

$$\begin{aligned} & E[g(X(\tau), \tau) - g(\overline{X}(\overline{\tau}), \overline{\tau})] \\ & \qquad = E[g(X(\tau), \tau) - g(\overline{X}(\widetilde{\tau}), \widetilde{\tau})] + E[g(\overline{X}(\widetilde{\tau}), \widetilde{\tau}) - g(\overline{X}(\overline{\tau}), \overline{\tau})] \\ & \qquad =: \mathcal{E}_C + \mathcal{E}_D. \end{aligned} \tag{10}$$

In [7], Gobet proves the following a priori error estimate with N uniform time steps:

$$E[f(X(\tau), \tau) - f(\overline{X}(\overline{\tau}), \overline{\tau})] = \mathcal{O}(N^{-1/2}). \tag{11}$$

In order to improve the convergence rate in (11), we adaptively refine the mesh according to computable error estimates. Error estimates for $\mathcal{E}_D$ and $\mathcal{E}_C$ are derived in Theorem 1 and Theorem 2 respectively.

2.1 Notation

In this paper, ∂_i denotes the derivative with respect to x_i, i.e. $\partial_i := \partial/\partial x_i$, and similarly for ∂_{ij} and ∂_{ijk}. If same subscript appears twice in a term, the term denotes the sum over the range of this subscript, e.g., $c_{ik}\partial_k b_j := \sum_k c_{ik}\partial_k b_j$. We use $X_t := X(t)$ and $\overline{X}_t := \overline{X}(t)$ for the continuous cases and $\overline{X}^n := \overline{X}(t_n)$ for the discrete case. The piecewise constant mesh function Δt is defined by

$$\Delta t(s) := \Delta t_n \quad \text{for } s \in [t_n, t_{n+1}) \text{ and } n = 0, 1, \ldots, N-1 \tag{12}$$

and

$$\Delta t_{\max} := \max_{n,\omega} \Delta t_n(\omega).$$

We let $\mathbf{1}_A$ denote the indicator function, i.e. $\mathbf{1}_A(y) = 1$ if $y \in A$, otherwise $\mathbf{1}_A(y) = 0$.

2.2 Expansion of Exiting Error using Probability

Consider the time discretization error between the continuous and the discrete Euler path, denoted by $\mathcal{E}_D$ in (10). In the case when the continuous Euler path ends at time $t = T$, i.e. $\widetilde{\tau} = T = \overline{\tau}$, there is no time discretization error between two Euler paths since $E[g(\overline{X}_{\widetilde{\tau}}, \widetilde{\tau})\mathbf{1}_{\widetilde{\tau}=T}] = E[g(\overline{X}_{\overline{\tau}}, \overline{\tau})\mathbf{1}_{\overline{\tau}=T}]$. On the other hand, if the continuous Euler path is stopped at $\widetilde{\tau} < T$ then it is possible that $\widetilde{\tau} < \overline{\tau}$. Figure 1 shows an illustrative Monte Carlo trajectory where the continuous Euler path $\overline{X}(t) \in \mathbb{R}$ exits

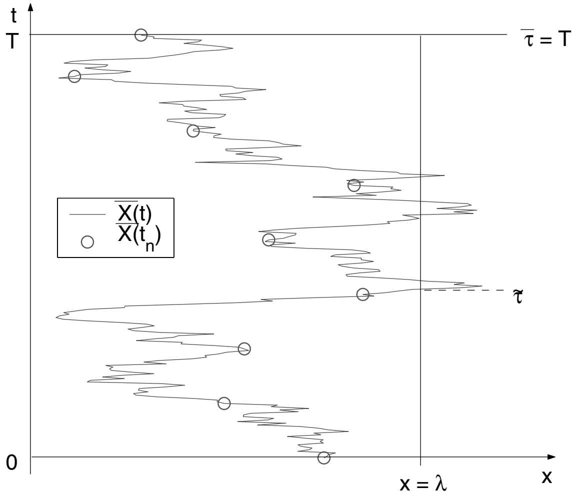


Fig. 1. An illustrative Euler Monte Carlo trajectory when $\widetilde{\tau} < \overline{\tau}$

the domain $D = (-\infty, \lambda)$ at $t = \widetilde{\tau} < T$, but the discrete Euler process $\overline{X}^n$ does not stop until much later, $\overline{\tau} = T$.

Taking the above effect into account, $\mathcal{E}_D$ can be estimated using the probability of the continuous Euler path exiting in a time interval (t_n, t_{n+1}) conditioned on the values of $\overline{X}^n$ and $\overline{X}^{n+1}$ in the discrete Euler process. Consider the particular case of a half space $D = \{x \in \mathbb{R}^d : \langle v, x\rangle_{\mathbb{R}^d} < \lambda\}$ for a constant λ and a constant unit vector v. The probability $P_{\overline{X},n}$ of $\overline{X}(t)$ exiting at some $t \in (t_n, t_{n+1})$ has an explicit expression, see e.g. [12], [1],

$$\begin{aligned} P_{\overline{X},n} &:= \mathbb{P}\left[\max_{t\in[t_n,t_{n+1}]} \langle v, \overline{X}_t\rangle_{\mathbb{R}^d} \geq \lambda \,\middle|\, \overline{X}^n = z^1, \overline{X}^{n+1} = z^2\right] \\ &= \exp\left(-2\frac{(\lambda - \langle v, z^1\rangle_{\mathbb{R}^d})(\lambda - \langle v, z^2\rangle_{\mathbb{R}^d})}{\sigma^2 \Delta t_n}\right) \end{aligned} \tag{13}$$

where $\langle v, z^1\rangle_{\mathbb{R}^d} < \lambda$ and $\langle v, z^2\rangle_{\mathbb{R}^d} < \lambda$ and $\sigma^2 = v_i b(\overline{X}^n)_i^\ell b(\overline{X}^n)_j^\ell v_j$. The work [3] studies estimates for the exit probability of the Brownian bridge in general cases of one dimension, e.g. with time dependent lower and upper boundaries. For a family of non degenerate SDEs in high dimension, including the half space case, the exit probabilities are expressed as asymptotic series in [6], [2]. In the more general case, we can approximate D locally near the boundary by its tangent half space and use the approximation of the exit probability for the half space case, see [7], [8].

We have the following error representation for $\mathcal{E}_D$, formulated for the case $D = \{x \in \mathbb{R}^d : x_1 < \lambda\}$:

Theorem 1. *Let* $\overline{X}(t)$ *and* $\overline{X}(t_n)$ *be the continuous and discrete Euler approximations defined in* (8) *and* (6) *respectively. Let* χ *be the* σ*-algebra generated by*

$\{\overline{X}^n : n = 0, 1, \ldots, N\}$. *Then the error* $\mathcal{E}_D$ *has the representation*

$$E[g(\overline{X}_{\widetilde{\tau}}, \widetilde{\tau}) - g(\overline{X}_{\overline{\tau}}, \overline{\tau})] = E\left[\sum_{n=0}^{N-1} \big(g(\overline{X}_{\xi_n}, \xi_n) - g(\overline{X}_{\overline{\tau}}, \overline{\tau})\big)\, \widehat{P}_{\overline{X},n}\right] \tag{14}$$

for $\xi_n \in (t_n, t_{n+1})$, $\overline{X}_{\xi_n} = (\lambda, \overline{X}_{2,\xi_n}, \ldots, \overline{X}_{d,\xi_n})$ *satisfying*

$$g(\overline{X}_{\xi_n}, \xi_n) = E\big[g(\overline{X}_{\widetilde{\tau}}, \widetilde{\tau}) \mid \chi, \widetilde{\tau} \in [t_n, t_{n+1})\big]$$

and where $\widehat{P}_{\overline{X},n}$ *are conditional first exit probabilities defined by*

$$\begin{aligned} \widehat{P}_{\overline{X},n} &= P_{\overline{X},n} \prod_{k=0}^{n-1} (1 - P_{\overline{X},k}), \qquad n = 1, 2, \ldots, N-1, \\ \widehat{P}_{\overline{X},0} &= P_{\overline{X},0}, \end{aligned} \tag{15}$$

using the conditional exit probabilities from (13).

Proof. Since $E[(g(\overline{X}_{\widetilde{\tau}}, \widetilde{\tau}) - g(\overline{X}_{\overline{\tau}}, \overline{\tau}))\mathbf{1}_{\widetilde{\tau}=T}] = 0$ and $\mathbf{1}_{\widetilde{\tau}<T} = \sum_{n=0}^{N-1} \mathbf{1}_{\widetilde{\tau}\in[t_n,t_{n+1})}$ we obtain

$$E[g(\overline{X}_{\widetilde{\tau}}, \widetilde{\tau}) - g(\overline{X}_{\overline{\tau}}, \overline{\tau})] = E\left[\sum_{n=0}^{N-1} (g(\overline{X}_{\widetilde{\tau}}, \widetilde{\tau}) - g(\overline{X}_{\overline{\tau}}, \overline{\tau}))\mathbf{1}_{\widetilde{\tau}\in[t_n,t_{n+1})}\right],$$

and after smoothing with the σ-algebra χ generated by $\{\overline{X}^n : n = 0, 1, \ldots, N\}$

$$E[g(\overline{X}_{\widetilde{\tau}}, \widetilde{\tau}) - g(\overline{X}_{\overline{\tau}}, \overline{\tau})] = E\left[\sum_{n=0}^{N-1} E\left[\big(g(\overline{X}_{\widetilde{\tau}}, \widetilde{\tau}) - g(\overline{X}_{\overline{\tau}}, \overline{\tau})\big)\mathbf{1}_{\widetilde{\tau}\in[t_n,t_{n+1})} \big| \chi\right]\right]. \tag{16}$$

In the right hand side we have

$$E[g(\overline{X}_{\overline{\tau}}, \overline{\tau})\mathbf{1}_{\widetilde{\tau}\in[t_n,t_{n+1})} \mid \chi] = g(\overline{X}_{\overline{\tau}}, \overline{\tau})\, \mathbb{P}[\widetilde{\tau} \in [t_n, t_{n+1})|\chi] \tag{17}$$

since $g(\overline{X}_{\overline{\tau}}, \overline{\tau}) \in \chi$, and, using the independence of the different coordinate directions in the Brownian bridge and the mean value theorem for integration

$$\begin{aligned} E[g(\overline{X}_{\widetilde{\tau}}, \widetilde{\tau})\mathbf{1}_{\widetilde{\tau}\in[t_n,t_{n+1})} \,|\, \chi] &= E\big[g(\overline{X}_{\widetilde{\tau}}, \widetilde{\tau}) \,|\chi, \widetilde{\tau} \in [t_n, t_{n+1})\big]\mathbb{P}[\widetilde{\tau} \in [t_n, t_{n+1}) \,|\chi] \\ &= g(\overline{X}_{\xi_n}, \xi_n)\mathbb{P}[\widetilde{\tau} \in [t_n, t_{n+1}) \mid \chi], \end{aligned} \tag{18}$$

for some $\xi_n \in (t_n, t_{n+1})$, $\overline{X}_{\xi_n} = (\lambda, \overline{X}_{2,\xi_n}, \ldots, \overline{X}_{d,\xi_n})$. Inserting (17) and (18) into (16) we get

$$E[g(\overline{X}_{\widetilde{\tau}}, \widetilde{\tau}) - g(\overline{X}_{\overline{\tau}}, \overline{\tau})] = E\left[\sum_{n=0}^{N-1} \big(g(\overline{X}_{\xi_n}, \xi_n) - g(\overline{X}_{\overline{\tau}}, \overline{\tau})\big)\, \mathbb{P}[\widetilde{\tau} \in [t_n, t_{n+1}) \,|\, \chi]\right]. \tag{19}$$

To compute the probability in (19), we observe that the event $\{\widetilde{\tau} \in [t_n, t_{n+1})\}$ is equivalent to

$$\big\{\overline{X}_{t\in[t_0,t_1)} \in D, \dots, \overline{X}_{t\in[t_{n-1},t_n)} \in D, \text{ and } \overline{X}_{t\in[t_n,t_{n+1})} \notin D\big\}$$

and that the events $\big\{\overline{X}_{t\in[t_n,t_{n+1})} \in D\big\}$, for $n = 0, 1, \dots, N-1$, are independent with respect to χ. Thus, using the conditional exit probabilities $P_{\overline{X},k}$, we obtain

$$\widehat{P}_{\overline{X},n} := \mathbb{P}[\widetilde{\tau} \in [t_n, t_{n+1}) \mid \chi] = P_{\overline{X},n} \prod_{k=0}^{n-1} (1 - P_{\overline{X},k})$$

and

$$\widehat{P}_{\overline{X},0} := \mathbb{P}[\widetilde{\tau} \in [t_0, t_1) \mid \chi] = P_{\overline{X},0},$$

which together with (19) proves (14).

Remark 1. For uniform time steps we know from [7] that $E[g(\overline{X}_{\widetilde{\tau}}, \widetilde{\tau}) - g(\overline{X}_{\overline{\tau}}, \overline{\tau})] = \mathcal{O}(\sqrt{\Delta t})$. To obtain a computable approximation of $E\big[g(\overline{X}_{\widetilde{\tau}}, \widetilde{\tau}) \bigm| \chi, \widetilde{\tau} \in [t_n, t_{n+1})\big]$ approximate by a linear function

$$g(\overline{X}_{\widetilde{\tau}}, \widetilde{\tau}) = g(\overline{X}(t_n), t_n) + B(\overline{X}_{\widetilde{\tau}} - \overline{X}(t_n)) + \mathcal{O}(|\overline{X}_{\widetilde{\tau}} - \overline{X}(t_n)|^2 + |\widetilde{\tau} - t_n|).$$

The last two terms have expected value $E[\dots|\chi, \widetilde{\tau} \in [t_n, t_{n+1})] = \mathcal{O}(\Delta t)$ and $\overline{X}_{\xi_n} = (\lambda, \overline{X}_{2,\xi_n}, \dots, \overline{X}_{d,\xi_n})$ is based on pinned Brownian motions $Y := (\overline{X}_2, \dots, \overline{X}_d)$ independent of $\widetilde{\tau}$. Hence the expected value $E[Y|\chi, \widetilde{\tau} \in [t_n, t_{n+1})]$ is

$$Y(t_n) + (Y(t_{n+1} - Y(t_n))\frac{E[\widetilde{\tau}|\chi, \widetilde{\tau} \in [t_n, t_{n+1})] - t_n}{t_{n+1} - t_n}.$$

The expected value $E[\widetilde{\tau}|\chi, \widetilde{\tau} \in [t_n, t_{n+1})]$ can be calculated from the explicit probability distribution of the exit time for Brownian bridges in [1].

2.3 Error Expansion Using Dual Solutions

In this subsection, we derive a computable error estimate between the exact and the continuous Euler path, i.e. $\mathcal{E}_C$ in (10). The main result is stated in Theorem 2 and the proof is presented afterwards.

The error estimate uses the discrete dual functions $\varphi(t_n)$, $\varphi'(t_n)$ and $\varphi''(t_n)$, taking values in $\mathbb{R}^d$, $\mathbb{R}^{d^2}$ and $\mathbb{R}^{d^3}$ respectively, defined as follows. For simplicity we describe the case when D is the half space $\{x : x_1 < \lambda\}$; see Remark 2. Introduce the notation

$$c_i(t_n, x) = x_i + \Delta t_n a_i(x) + b_i^l(x)\Delta W_n^l, \qquad i = 1, 2, \dots, d,$$
$$\beta_{ij}(x) = \frac{1}{2} b_i^l(x) b_j^l(x), \qquad i, j = 1, 2, \dots, d.$$

Then the function φ is defined by the dual backward problem

$$\varphi_i(t_n) = \partial_i c_j(t_n, \overline{X}^n)\varphi_j(t_{n+1}), \qquad t_n < \overline{\tau}, \quad i = 1, 2, \dots, d, \tag{20}$$

$$\varphi_i(\overline{\tau}) = \begin{cases} \partial_i g(\overline{X}_{\overline{\tau}}, \overline{\tau}), & \text{if } \overline{\tau} = T, \quad i = 1, 2, \dots, d, \text{ or} \\ & \text{if } \overline{\tau} < T \text{ and } i = 2, \dots, d, \\ -(g(\hat{\overline{X}}_{\hat{\overline{\tau}}}, \hat{\overline{\tau}}) - g(\overline{X}_{\overline{\tau}}, \overline{\tau}))/\Delta x, & \text{if } \overline{\tau} < T \text{ and } i = 1; \end{cases} \tag{21}$$

since $\partial_1 g(\overline{X}_{\overline{\tau}}, \overline{\tau})$ does not exist if $\overline{\tau} < T$ we have introduced the restarted Euler approximation $\hat{\overline{X}}(t_n)$ for $t_n \in [\overline{\tau}, \hat{\overline{\tau}}]$ with initial value $\hat{\overline{X}}(\overline{\tau}) = \overline{X}(\overline{\tau}) + \gamma\Delta x$, where γ is an inward unit normal vector, Δx is a small positive number and $\hat{\overline{\tau}}$ denotes the first exit time of $\hat{\overline{X}}$, i.e. $\hat{\overline{\tau}} := \min\{t_n : \overline{\tau} < t_n \text{ and } \hat{\overline{X}}^n \notin D\}$. The first variation φ' satisfies, cf. [17],

$$\begin{aligned} \varphi'_{ik}(t_n) &= \partial_i c_j(t_n, \overline{X}^n)\partial_k c_m(t_n, \overline{X}^n)\varphi'_{jm}(t_{n+1}) \\ &\quad + \partial^2_{ik} c_j(t_n, \overline{X}^n)\varphi_j(t_{n+1}), \qquad t_n < \overline{\tau}, \end{aligned} \tag{22}$$

$$\varphi'_{ik}(\overline{\tau}) = \delta^2_{ik} g(\overline{X}_{\overline{\tau}}, \overline{\tau}), \tag{23}$$

where we interpret $\delta^2_{ik} g(\overline{X}_{\overline{\tau}}, \overline{\tau})$ as the corresponding second derivatives when possible and make use of difference quotients otherwise. If no simplifying property of the domain D and the drift b^l_i is present we may use additional restarted processes, similar to $\hat{\overline{X}}$, and difference quotients to define the initial values of φ' and φ''. Interpreting $\delta^3_{ikp} g(\overline{X}_{\overline{\tau}}, \overline{\tau})$ analogously to $\delta^2_{ik} g(\overline{X}_{\overline{\tau}}, \overline{\tau})$, the second variation φ'' satisfies

$$\begin{aligned} \varphi'_{ikp}(t_n) &= \partial_i c_j(t_n, \overline{X}^n)\partial_k c_m(t_n, \overline{X}^n)\partial_p c_r(t_n, \overline{X}^n)\varphi''_{jmr}(t_{n+1}) \\ &\quad + \partial^2_{ip} c_j(t_n, \overline{X}^n)\partial_k c_m(t_n, \overline{X}^n)\varphi'_{jm}(t_{n+1}) \\ &\quad + \partial_i c_j(t_n, \overline{X}^n)\partial^2_{kp} c_m(t_n, \overline{X}^n)\varphi'_{jm}(t_{n+1}) \\ &\quad + \partial^2_{ik} c_j(t_n, \overline{X}^n)\partial_p c_m(t_n, \overline{X}^n)\varphi'_{jm}(t_{n+1}) \\ &\quad + \partial^3_{ikp} c_j(t_n, \overline{X}^n)\varphi_j(t_{n+1}), \qquad t_n < \overline{\tau}, \end{aligned} \tag{24}$$

$$\varphi''_{ikr}(\overline{\tau}) = \delta^3_{ikp} g(\overline{X}_{\overline{\tau}}, \overline{\tau}). \tag{25}$$

Remark 2. For more general domains we may approximate ∂D with the tangent plane at the stopping point $(\overline{X}_{\overline{\tau}}, \overline{\tau})$, compute derivatives of g in the directions of the tangent plane and use difference quotients in the normal direction and then transform back to the original coordinate directions.

The time discretization error $\mathcal{E}_C$ in (10) has the following error expansion:

Theorem 2. *Let $X(t)$, $\overline{X}(t)$ and $\overline{X}(t_n)$ be the exact, the continuous Euler and the discrete Euler path defined in* (2), (8) *and* (6) *respectively. Assume that the functions a, b and g are bounded in $\mathcal{C}^6(D)$ and $\mathcal{C}^6(D \times [0, T])$ respectively. Then the time discretization error $\mathcal{E}_C$ has the error expansion*

$$
\begin{aligned}
E[g(X_\tau,\tau) - g(\overline{X}_{\tilde{\tau}},\tilde{\tau})] = {} & E\left[\sum_{n=0}^{N-1} \mathbf{1}_{t_{n+1}\le\tilde{\tau}}\rho_n \mathit{\Delta} t_n^2\right] \\
& + \mathcal{O}\left(\mathit{\Delta} x + \sqrt{\mathit{\Delta} t_{\max}} + \frac{\sqrt{\mathit{\Delta} t_{\max}}}{\mathit{\Delta} x^k}\right) E\left[\sum_{n=0}^{N-1} \mathit{\Delta} t_n^2\right]
\end{aligned}
\tag{26}
$$

where $\mathit{\Delta} x$ is a small positive constant and $k \in \{1,2,3\}$ is the highest order of difference quotient used in (20)–(25) *to define φ, φ', φ'', and*

$$
\begin{aligned}
\rho_n = {} & \frac{1}{2}(\partial_t a_k + a_j\partial_j a_k + \beta_{ij}\partial_{ij}^2 a_k)(\overline{X}^n)\varphi_k(t_{n+1}) \\
& + \frac{1}{2}\left(\partial_t\beta_{km} + 2\beta_{jm}\partial_j a_k + a_j\partial_j\beta_{km} + \beta_{ij}\partial_{ij}^2\beta_{km}\right)(\overline{X}^n)\varphi'_{km}(t_{n+1}) \\
& + (\beta_{jr}\partial_j\beta_{km})(\overline{X}^n)\varphi''_{kmr}(t_{n+1}).
\end{aligned}
\tag{27}
$$

Remark 3. If we do not solve the backward dual problems (20)–(25), but instead set $\varphi \equiv \varphi' \equiv \varphi'' \equiv 1$ we obtain adaptivity based on the local error.

The proof of Theorem 2 has several steps and we present them by following three lemmas. Let us first introduce a solution u of the Kolmogorov backward equation

$$
\begin{aligned}
\partial_t u + a_i\partial_i u + \beta_{ij}\partial_{ij}^2 u &= 0, && (x,t)\in D\times[0,T), \\
u(x,T) &= g(x,T), && x\in D, \\
u(x,t) &= g(x,t), && (x,t)\in\partial D\times[0,T].
\end{aligned}
\tag{28}
$$

Then by the Feynman-Kac formula u can be represented by the expectation

$$
u(x,t) = E[g(X_\tau,\tau) \mid X(t) = x]. \tag{29}
$$

Let $\overline{a_i}$ and $\overline{b_i}$ be the piecewise constant functions defined by $\overline{a_i}(t) = a_i(\overline{X}^n)$ and $\overline{b_i}(t) = b_i(\overline{X}^n)$ for $t\in[t_n,t_{n+1})$. Similarly define $\overline{\beta_{ij}} = \frac{1}{2}\overline{b_i^l b_j^l}$. Then the time discretization error $\mathcal{E}_C$ has the following representation :

Lemma 1. *Let $X(t)$ and $\overline{X}(t)$ be the exact and the continuous Euler path defined by* (2) *and* (8) *respectively and let the function u be defined by* (29). *Suppose that the assumptions in Theorem 2 hold. Then the time discretization error between these two paths has the representation*

$$
E[g(X_\tau,\tau) - g(\overline{X}_{\tilde{\tau}},\tilde{\tau})] = E\left[\int_0^{\tilde{\tau}} \left((a_i - \overline{a_i})\partial_i u + (\beta_{ij} - \overline{\beta_{ij}})\partial_{ij}^2 u\right)(\overline{X}_t,t)\,\mathrm{d}t\right]. \tag{30}
$$

Proof. Apply the Itô formula to the function u in (29) to get

$$
\mathrm{d}u(\overline{X}_t,t) = \left(\partial_t u + \overline{a_i}\partial_i u + \overline{\beta_{ij}}\partial_{ij}^2 u\right)(\overline{X}_t,t)\,\mathrm{d}t + \overline{b_i^l}\partial_i u(\overline{X}_t,t)\,\mathrm{d}W_t^l.
$$

Here the definition of the continuous Euler scheme in (8) is used, i.e. $d\overline{X}_i(t) = \overline{a_i}\,\mathrm{d}t + \overline{b_i^l}\,\mathrm{d}W_t^l$ for $t \in [t_n, t_{n+1})$. Integrate both sides from 0 to $\widetilde{\tau}$ and take the expectation to obtain

$$\begin{aligned} E[u(\overline{X}_{\widetilde{\tau}}, \widetilde{\tau}) - u(\overline{X}_0, 0)] &= E\left[\int_0^{\widetilde{\tau}} (\partial_t u + \overline{a_i}\partial_i u + \overline{\beta_{ij}}\partial_{ij}^2 u)(\overline{X}_t, t)\,\mathrm{d}t\right] \\ &+ E\left[\int_0^{\widetilde{\tau}} \overline{b_i^l}\partial_i u(\overline{X}_t, t)\,\mathrm{d}W_t^l\right]. \end{aligned} \tag{31}$$

Note that the Itô integral in (31) is not adapted to the standard filtration generated by W alone. Instead consider the filtration $\mathcal{G}_t$, the σ-algebra generated by $\{W^l(s), \Delta t(s) : s \leq t, l = 1, 2, \ldots, l_0\}$. Then from Lemma 4.2 in [15] the Itô integral in (31) is a martingale with respect to $\mathcal{G}_t$ and since $\widetilde{\tau}$ is a stopping time, we therefore have

$$E\left[\int_0^{\widetilde{\tau}} \overline{b_i^l}\partial_i u(\overline{X}_t, t)\mathrm{d}W_t^l\right] = 0.$$

In the left hand side of (31) we use the boundary conditions in (28)

$$E[u(\overline{X}_{\widetilde{\tau}}, \widetilde{\tau})] = E[g(\overline{X}_{\widetilde{\tau}}, \widetilde{\tau})],$$

and the Feynman-Kac formula (29)

$$u(\overline{X}_0, 0) \;=\; E[g(X_\tau, \tau) \mid X_0 = \overline{X}_0] = \; E[g(X_\tau, \tau)].$$

Finally we use the Kolmogorov backward equation (28) to eliminate $\partial_t u$ in the first expectation of the right hand side in (31) and conclude (30).

Using the discrete time steps, the error representation (30) can be written

$$\begin{aligned} &E[g(X_\tau, \tau) - g(\overline{X}_{\widetilde{\tau}}, \widetilde{\tau})] \\ &\quad = E\left[\sum_{n=0}^{N-1}\int_{t_n}^{t_{n+1}} \mathbf{1}_{t \leq \widetilde{\tau}}\left((a_i - \overline{a_i})\partial_i u + (\beta_{ij} - \overline{\beta_{ij}})\partial_{ij}^2 u\right)(\overline{X}_t, t)\,\mathrm{d}t\right]. \end{aligned} \tag{32}$$

Lemma 2. *Let* $X(t)$ *and* $\overline{X}(t)$ *be the exact path and the continuous Euler path defined in* (2) *and* (8) *respectively and assume that the assumptions in Theorem 2 hold. Then the time discretization error between these two paths has the following expansion*

$$E[g(X_\tau, \tau) - g(\overline{X}_{\widetilde{\tau}}, \widetilde{\tau})] = E\left[\sum_{n=0}^{N-1} \mathbf{1}_{t_{n+1} \leq \widetilde{\tau}}\widetilde{\rho}_n \Delta t_n^2\right] + \mathcal{O}(\sqrt{\Delta t_{\max}})E\left[\sum_{n=0}^{N-1} \mathcal{O}(\Delta t_n^2)\right] \tag{33}$$

where

$$\begin{aligned}\widetilde{\rho}_n &= \frac{1}{2}(\partial_t a_k + a_j\partial_j a_k + \beta_{ij}\partial^2_{ij}a_k)(\overline{X}^n)\partial_k u(\overline{X}^{n+1}, t_{n+1})\\ &+ \frac{1}{2}\left(\partial_t\beta_{km} + 2\beta_{jm}\partial_j a_k + a_j\partial_j\beta_{km} + \beta_{ij}\partial^2_{ij}\beta_{km}\right)(\overline{X}^n)\partial^2_{km}u(\overline{X}^{n+1}, t_{n+1})\\ &+ (\beta_{jr}\partial_j\beta_{km})(\overline{X}^n)\partial^3_{kmr}u(\overline{X}^{n+1}, t_{n+1}).\end{aligned} \tag{34}$$

Proof. Apply the Itô formula to each term in (32) to get

$$\begin{aligned}a_i(\overline{X}_t) - \overline{a_i}(\overline{X}_t) &= a_i(\overline{X}_t) - a_i(\overline{X}_n)\\ &= \int_{t_n}^t \left(\partial_s a_i + \overline{a_k}\partial_k a_i + \overline{\beta_{jk}}\partial^2_{jk}a_i\right)(\overline{X}_s)\,\mathrm{d}s\\ &+ \int_{t_n}^t \overline{b^l_j}\partial_j a_i(\overline{X}_s)\,\mathrm{d}W^l_s,\end{aligned}$$

and similarly

$$\begin{aligned}&\beta_{ij}(\overline{X}_t) - \overline{\beta_{ij}}(\overline{X}_t)\\ &= \int_{t_n}^t \left(\partial_s\beta_{ij} + \overline{a_k}\partial_k\beta_{ij} + \overline{\beta_{km}}\partial^2_{km}\beta_{ij}\right)(\overline{X}_s)\,\mathrm{d}s + \int_{t_n}^t \overline{b^l_k}\partial_k\beta_{ij}(\overline{X}_s)\,\mathrm{d}W^l_s.\end{aligned}$$

Substitute the above integrals in (32) and use Malliavin derivatives, see [17], for example

$$\begin{aligned}&E\left[\sum_{n=0}^{N-1}\int_{t_n}^{t_{n+1}} \mathbf{1}_{t\le\widetilde{\tau}}\int_{t_n}^t \overline{b^l_j}\partial_j a_i(\overline{X}_s)\partial_i u(\overline{X}_t, t)\,\mathrm{d}W^l_s\,\mathrm{d}t\right]\\ &\qquad = E\left[\sum_{n=0}^{N-1}\int_{t_n}^{t_{n+1}} \mathbf{1}_{t\le\widetilde{\tau}}\int_{t_n}^t 2\overline{\beta_{jm}}\partial_j a_i(\overline{X}_s)\partial^2_{im}u(\overline{X}_t, t)\,\mathrm{d}s\ \mathrm{d}t\right]\end{aligned}$$

to get

$$\begin{aligned}&E[g(X_\tau, \tau) - g(\overline{X}_{\widetilde{\tau}}, \widetilde{\tau})]\\ &= E\Bigg[\sum_{n=0}^{N-1}\int_{t_n}^{t_{n+1}} \mathbf{1}_{t\le\widetilde{\tau}}\Bigg(\int_{t_n}^t \left(\partial_s a_i + \overline{a_k}\partial_k a_i + \overline{\beta_{jk}}\partial^2_{jk}a_i\right)(\overline{X}_s)\,\mathrm{d}s\ \partial_i u(\overline{X}_t, t)\\ &\quad + \int_{t_n}^t \left(\partial_s\beta_{km} + 2\overline{\beta_{jm}}\partial_j a_k + \overline{a_j}\partial_j\beta_{km} + \overline{\beta_{ij}}\partial^2_{ij}\beta_{km}\right)(\overline{X}_s)\,\mathrm{d}s\,\partial^2_{km}u(\overline{X}_t, t)\\ &\qquad + \int_{t_n}^t 2\overline{\beta_{jr}}\partial_j\beta_{km}(\overline{X}_s)\,\mathrm{d}s\ \partial^3_{kmr}u(\overline{X}_t, t)\Bigg)\,\mathrm{d}t\Bigg].\end{aligned} \tag{35}$$

Each term in (35) has the form

$$E\left[\sum_{n=0}^{N-1}\int_{t_n}^{t_{n+1}}\int_{t_n}^t \mathbf{1}_{t\le\widetilde{\tau}}f(\overline{X}_s)\ h(\overline{X}_t, t)\,\mathrm{d}s\,\mathrm{d}t\right] \tag{36}$$

where f is a function of a_i, β_{ij} and their derivatives representing the local error and h is a function of the derivatives of u. Finally apply the a priori error estimate (11) to the expected value (36) to conclude

$$\begin{aligned}
&E\left[\sum_{n=0}^{N-1}\int_{t_n}^{t_{n+1}}\int_{t_n}^{t}\mathbf{1}_{t\leq\bar{\tau}}f(\overline{X}_s)\,h(\overline{X}_t,t)\,\mathrm{d}s\,\mathrm{d}t\right]\\
&=E\left[\sum_{n=0}^{N-1}\frac{1}{2}\mathbf{1}_{t_{n+1}\leq\bar{\tau}}f(\overline{X}^n)h(\overline{X}^{n+1},t_{n+1})\Delta t_n^2\right]+\mathcal{O}(\sqrt{\Delta t_{\max}})E\left[\sum_{n=0}^{N-1}\Delta t_n^2\right]
\end{aligned}$$

which proves (33).

Note that the quantities $\partial_i u, \partial_{ij}^2 u$ and $\partial_{ijk}^3 u$ in (34) not are computable. The adaptive algorithms will use the computable approximations (20)-(25) for these functions. From the construction of u we have

$$\partial_k u(x,t)=E[\partial_i u(X_\tau,\tau)X'_{ik}(\tau;t)\mid X'_{ij}(t)=\delta_{ij},X(t)=x], \tag{37}$$

where δ_{ij} denotes the Kronecker δ–function and $X'_{ij}(s;t):=\partial X_i(s;X(t)=x)/\partial x_j$ is the first variation of $X(s)$ with respect to a perturbation in the initial location at time t, i.e. it satisfies

$$\begin{aligned}
&\mathrm{d}X'_{ij}(s)=\partial_k a_i(X(s))X'_{kj}(s)\,\mathrm{d}s+\partial_k b_i^l(X(s))X'_{kj}(s)\,\mathrm{d}W^l(s),\quad t<s<\tau,\\
&X'_{ij}(t)=\delta_{ij}.
\end{aligned} \tag{38}$$

The goal is to approximate $\partial_k u(\overline{X}^n,t_n)$ in (34) by conditional expected values of the computable quantities φ_k defined in (20)-(21) and similarly to approximate $\partial_{ij}^2 u$ and $\partial_{ijk}^3 u$ by expected values of φ'_{ij} and φ''_{ijk} in (22)-(23) and (24)-(25) respectively.

Note that if the continuous exact path finishes at $\tau=T$ then by the definition of u, we have $\partial_k u(X_T,T)=\partial_k g(X_T,T)$ so that

$$\begin{aligned}
&E[\partial_i u(X_\tau,\tau)X'_{ik}(\tau;t)\mathbf{1}_{\tau=T}\mid X'_{ij}(t)=\delta_{ij},X(t)=x]\\
&\qquad=E[\partial_i g(X_T,T)X'_{ik}(T;t)\mathbf{1}_{\tau=T}\mid X'_{ij}(t)=\delta_{ij},\ X(t)=x].
\end{aligned} \tag{39}$$

However, for $\tau<T$ the first variation $\partial_i g(X_\tau,\tau)$ exists only in the directions tangent to the boundary ∂D, $i=2,\dots,d$. In the direction normal to ∂D we approximate $\partial_1 u(X_\tau,\tau)$ in (37) by the expected value of a difference quotient of g and remove this second expected value. To do this we introduce a small positive constant Δx. Once the continuous exact path crosses the boundary, we start a new realization $\hat{X}$ with the initial value

$$\hat{X}(\tau)=X(\tau)+\gamma\Delta x\in D,$$

where γ denotes an inward unit normal vector. The new realization $\hat{X}_t$ evolves by (2) for $\tau<t<\hat{\tau}$ until it stops with the first exit time $\hat{\tau}\in(\tau,T]$. Then by the Taylor expansion we have

$$\partial_1 u(X_\tau,\tau) = -\frac{u(\hat{X}_\tau,\tau)-u(X_\tau,\tau)}{\Delta x} + \mathcal{O}(\Delta x)$$

and the Feynman-Kac formula (29) gives

$$\partial_1 u(X_\tau,\tau) = -\frac{E[g(\hat{X}_{\hat{\tau}},\hat{\tau})-g(X_\tau,\tau)|\mathcal{G}_\tau]}{\Delta x} + \mathcal{O}(\Delta x)$$

where $\mathcal{G}_t$ is the σ-algebra generated by $\{W^l(s), \Delta t(s) : s \le t, l = 1,2,\dots,l_0\}$. Use the measurability of $X'_{ik}(\tau;t)\mathbf{1}_{\tau<T} \in \mathcal{G}_\tau$ to get

$$\begin{aligned}
&E\left[\partial_1 u\left(X_\tau,\tau\right) X'_{1k}(\tau;t)\mathbf{1}_{\tau<T} \mid X'_{ij}(t)=\delta_{ij},\, X(t)=x\right]\\
&= E\left[E\left[\frac{g(\hat{X}_{\hat{\tau}},\hat{\tau})-g(X_\tau,\tau)}{-\Delta x} \,\middle|\, \mathcal{G}_\tau \right] X'_{1k}(\tau;t)\mathbf{1}_{\tau<T} \,\middle|\, \begin{array}{l} X'_{ij}(t)=\delta_{ij}, X(t)=x,\\ \hat{X}_\tau = X_\tau + \gamma\Delta x \end{array} \right]\\
&\quad + \mathcal{O}(\Delta x)\\
&= E\left[E\left[\frac{g(\hat{X}_{\hat{\tau}},\hat{\tau})-g(X_\tau,\tau)}{-\Delta x} X'_{1k}(\tau;t)\mathbf{1}_{\tau<T} \,\middle|\, \mathcal{G}_\tau \right] \,\middle|\, \begin{array}{l} X'_{ij}(t)=\delta_{ij}, X(t)=x,\\ \hat{X}_\tau = X_\tau + \gamma\Delta x \end{array} \right]\\
&\quad + \mathcal{O}(\Delta x)\\
&= E\left[\frac{g(\hat{X}_{\hat{\tau}},\hat{\tau})-g(X_\tau,\tau)}{-\Delta x} X'_{1k}(\tau;t)\mathbf{1}_{\tau<T} \,\middle|\, \begin{array}{l} X'_{ij}(t)=\delta_{ij}, X(t)=x,\\ \hat{X}_\tau = X_\tau + \gamma\Delta x \end{array} \right] + \mathcal{O}(\Delta x),
\end{aligned}$$

and thus

$$\begin{aligned}
&E[\partial_i u(X_\tau,\tau) X'_{ik}(\tau;t)\mathbf{1}_{\tau<T} \mid X'_{ij}(t)=\delta_{ij}, X(t)=x]\\
&= E\left[-\frac{g(\hat{X}_{\hat{\tau}},\hat{\tau})-g(X_\tau,\tau)}{\Delta x} X'_{1k}(\tau;t)\mathbf{1}_{\tau<T} \,\middle|\, \begin{array}{l} X'_{ij}(t)=\delta_{ij}, X(t)=x,\\ \hat{X}_\tau = X_\tau + \gamma\Delta x \end{array} \right]\\
&+ E\left[\sum_{i=2}^{d} \partial_i g(X_\tau,\tau) X'_{ik}(\tau;t)\mathbf{1}_{\tau<T} \,\middle|\, X'_{ij}(t)=\delta_{ij}, X(t)=x \right] + \mathcal{O}(\Delta x). \qquad (40)
\end{aligned}$$

The expected values in the right hand sides of (39) and (40) can be approximated using Euler approximations and the error in doing so is estimated by repeated use of the a priori error estimate (11). Let thus $(\overline{\hat{X}}, \overline{\hat{\tau}})$ be the Euler approximation of $(\hat{X}, \hat{\tau})$ and gather all X_i, X_{ij} in a stochastic process Y_t, taking values in $\mathbb{R}^{d+d^2}$. Then Y_t satisfies the system of SDEs, (2) and (38), which we write

$$\mathrm{d}Y(t) = A(Y(t))\,\mathrm{d}t + B^l(Y(t))\,\mathrm{d}W^l(t), \qquad t > t_0, \quad Y(t_0) = Y_0. \qquad (41)$$

Similarly define the corresponding Euler approximation $\overline{Y}$ of Y as the solution of

$$\overline{Y}(t_{n+1}) = \overline{Y}(t_n) + A(\overline{Y}(t_n))\Delta t_n + B^l(\overline{Y}(t_n))\Delta W_n^l, \quad n \ge 0, \quad Y(t_0) = Y_0. \qquad (42)$$

Consider first the case $\tau = T$; apply the a priori error estimate (11) to the functions $\check{f}(Y_\tau, \tau) = \partial_i g(X_\tau, \tau) X'_{ik}(\tau; t)\mathbf{1}_{\tau=T}$, for $k = 1, 2, \ldots, d$, to get

$$E[\check{f}(Y_\tau, \tau) - \check{f}(\overline{Y}_{\overline{\tau}}, \overline{\tau})] = \mathcal{O}(\sqrt{\Delta t_{\max}}).$$

When $\tau < T$, the second expected value in the right hand side of (40) is treated similarly as when $\tau = T$. For the first expected value in the right hand side in (40), extend Y_t to $\mathcal{Y}_t$ containing also the d-dimensional process $\hat{X}_t$, which solves (2) for $\tau < t < \hat{\tau}$. Then $\mathcal{Y}_t$ has two exit times $\theta = (\tau, \hat{\tau})^{\mathrm{T}}$. Denote by $\overline{\mathcal{Y}}$ and $\overline{\theta}$ the corresponding Euler approximations and apply (11) to the functions $\check{f}(\mathcal{Y}_\theta, \theta) = -\frac{g(\hat{X}_{\hat{\tau}}, \hat{\tau}) - g(X_\tau, \tau)}{\Delta x} X'_{1k}(\tau; t)\mathbf{1}_{\tau<T}$, for $k = 1, 2, \ldots, d$, to obtain

$$E[\check{f}(\mathcal{Y}_\theta, \theta) - \check{f}(\overline{\mathcal{Y}}_{\overline{\theta}}, \overline{\theta})] = \mathcal{O}\left(\frac{\sqrt{\Delta t_{\max}}}{\Delta x}\right)$$

and consequently

$$\begin{aligned}
\partial_k u(x,t) = {} & E[\partial_i g(\overline{X}_{\overline{\tau}}, \overline{\tau}) X'_{ik}(\overline{\tau}; t)\mathbf{1}_{\overline{\tau}=T} \mid \overline{X}'_{ij}(t) = \delta_{ij},\ \overline{X}(t) = x] \\
& + E\left[\frac{g(\hat{\overline{X}}_{\hat{\overline{\tau}}}, \hat{\overline{\tau}}) - g(\overline{X}_{\overline{\tau}}, \overline{\tau})}{-\Delta x} \overline{X}'_{1k}(\tau; t)\mathbf{1}_{\overline{\tau}<T} \,\middle|\, \begin{array}{l} \overline{X}'_{ij}(t) = \delta_{ij}, \overline{X}(t) = x, \\ \hat{\overline{X}}_{\overline{\tau}} = \overline{X}_{\overline{\tau}} + \gamma\Delta x \end{array} \right] \\
& + E\left[\sum_{i=2}^{d} \partial_i g(\overline{X}_{\overline{\tau}}, \overline{\tau}) \overline{X}'_{ik}(\tau; t)\mathbf{1}_{\overline{\tau}<T} \,\middle|\, \overline{X}'_{ij}(t) = \delta_{ij}, \overline{X}(t) = x \right] \\
& + \mathcal{O}\left(\Delta x + \sqrt{\Delta t_{\max}} + \frac{\sqrt{\Delta t_{\max}}}{\Delta x} \right). \qquad (43)
\end{aligned}$$

This is an expansion of the expected value of φ_k defined in (20)-(21). The higher derivatives $\partial^2_{ij} u$ and $\partial^3_{ijk} u$ can be computed in a similar way and we have the error expansion:

Lemma 3. *Suppose the assumptions in Theorem 2 hold. Then the function u defined by* (29) *and the dual functions φ, φ' and φ'' defined by* (20)-(25) *satisfy, for $\alpha = 1, 2, 3$,*

$$\partial^\alpha u(\overline{X}(t_n), t_n) - E[\varphi^\alpha(t_n) \,|\, \mathcal{F}_n] = \mathcal{O}\left(\Delta x + \sqrt{\Delta t_{\max}} + \frac{\sqrt{\Delta t_{\max}}}{\Delta x^\alpha} \right) \qquad (44)$$

where $\mathcal{F}_n$ denotes the σ-algebras generated by $\{W^l(s), \Delta t(s) : s \leq t_n,\ l = 1, 2, \ldots, l_0\}$, $\varphi^1 = \varphi_i$, $\varphi^2 = \varphi'_{ij}$ and $\varphi^3 = \varphi''_{ijk}$ for some i, j, k and $\partial^\alpha u$ is the corresponding α:th order derivative of u.

Proof. For $\alpha = 1$, the approximation (43) and the definition (20)-(21) yield (44).

Following [17] extend Y to be $(X, X', X'', X''')^{\mathrm{T}}$ satisfying the SDE similar to (41) with $Y(t_0) = (x, I, 0, 0)^{\mathrm{T}}$ where I is the $d \times d$-identity matrix. Here the first variation X' of X is defined in (38) and the other higher variations are defined similarly by taking the derivatives to the right hand side of (38). Introduce the corresponding Euler approximate $\overline{Y} = (\overline{X}, \overline{X}', \overline{X}'', \overline{X}''')^{\mathrm{T}}$ satisfying the SDE similar to (42) and let $(\overline{X}', \overline{X}'', \overline{X}''')$ denote the Euler approximations of (X', X'', X'''). For the case when $\tau = T$ and $\alpha = 2$ or 3, we use the a priori error estimate (11) for the extended systems Y and $\overline{Y}$ with

$$\begin{aligned} \check{f}(Y_\tau, \tau) &= \left(\partial_i g X''{}_{ikn} + \partial^2_{ir} g X'_{ik} X'_{rn}\right) \mathbf{1}_{\tau=T} && \text{if } \alpha = 2, \\ \check{f}(Y_\tau, \tau) &= (\partial_i g X'''{}_{iknm} + \partial^2_{ir} g X'_{ik} X''_{rnm} \\ &\quad + \partial^2_{ir} g X'_{in} X''_{rkm} + \partial^2_{ir} g X'_{im} X''_{rkn} \\ &\quad + \partial^3_{irv} g X'_{ik} X'_{rn} X'_{vm}) \mathbf{1}_{\tau=T} && \text{if } \alpha = 3, \end{aligned}$$

where $\check{f}(Y_\tau, \tau)$ in the case $\alpha = 2$ derives from

$$\begin{aligned} \partial_{kn} u(x,t) = E\big[&\partial_i u(X_\tau, \tau) X''_{ikn}(\tau) + \partial_{ir} u(X_\tau, \tau) X'_{ik}(\tau) X'_{rn}(\tau) \mid \\ & X''_{ikn}(t) = 0,\ X'_{ij}(t) = \delta_{ij},\ X(t) = x\big] \end{aligned}$$

with $u(X_\tau, \tau) = g(X_\tau, \tau)$ if $\tau = T$ and similarly for $\alpha = 3$. The extension to the case $\tau < T$ is similar to the first order derivative treated above; this time second and third order difference quotients appear leading to terms $\mathcal{O}(\sqrt{\Delta t_{max}}/\Delta x^2)$ and $\mathcal{O}(\sqrt{\Delta t_{max}}/\Delta x^3)$ respectively.

Proof of Theorem 2. The measurability of the function f_n depending on the derivatives of a and β, e.g. $f_n = \mathbf{1}_{t_{n+1} \le \overline{\tau}} (\partial_t a_k + a_j \partial_j a_k + \beta_{ij} \partial^2_{ij} a_k)(\overline{X}^n) \Delta t_n^2 \in \mathcal{F}_{n+1}$, proves

$$\begin{aligned} E\left[\sum_{n=0}^{N-1} f_n E[\varphi_k(t_{n+1}) | \mathcal{F}_{n+1}]\right] &= E\left[E\left[\sum_{n=0}^{N-1} f_n \varphi_k(t_{n+1}) \middle| \mathcal{F}_{n+1}\right]\right] \\ &= E\left[\sum_{n=0}^{N-1} f_n \varphi_k(t_{n+1})\right]. \end{aligned} \tag{45}$$

Similar representations hold for the other terms in (34). Consequently, the combination of Lemma 2-3 and the removal of the second expectation (45) prove (26). □

Remark 4. In the case of only first order difference quotients, the optimal size of the constant Δx for the difference quotient in (44) is $\mathcal{O}((\Delta t_{\max})^{1/4})$ and $\Delta x = \mathrm{TOL}_T^{1/4}$ is used for the adaptive algorithm in Sect. 3 where TOL_T is a given time discretization error tolerance. In the one dimensional example in Sect. 4 we use the Kolmogorov equation to replace higher order derivatives on the boundary with lower order terms. For instance $\varphi'(\overline{\tau}) = -\beta^{-1}(\partial_t g(\overline{X}_{\overline{\tau}}, \overline{\tau}) + a(\overline{X}_{\overline{\tau}})\varphi(\overline{\tau}))$ and $\varphi'''(\overline{\tau}) = \beta^{-1}((\partial_t g(\overline{X}_{\hat{\overline{\tau}}}, \hat{\overline{\tau}}) - \partial_t g(\overline{X}_{\overline{\tau}}, \overline{\tau}))/\Delta x + \partial_x a(\overline{X}_{\overline{\tau}})\varphi(\overline{\tau}) + (a + \partial_x \beta)(\overline{X}_{\overline{\tau}})\varphi'(\overline{\tau}))$.

3 Adaptive Algorithms for Stopped Diffusion

This section presents adaptive algorithms for the stopped diffusion problems. As described in Sect. 2, the computational error is separated into the following three terms : the time discretization error between the exact and the continuous Euler path $\mathcal{E}_C$, the time discretization error between the continuous and discrete Euler approximation $\mathcal{E}_D$, and the statistical error $\mathcal{E}_S$, i.e.

$$\begin{aligned}
&E[g(X(\tau),\tau)] - \frac{1}{M}\sum_{j=1}^{M} g(\overline{X}(\overline{\tau};\omega_j),\overline{\tau}) \\
&= E[g(X(\tau),\tau) - g(\overline{X}(\widetilde{\tau}),\widetilde{\tau})] + E[g(\overline{X}(\widetilde{\tau}),\widetilde{\tau}) - g(\overline{X}(\overline{\tau}),\overline{\tau})] \\
&+ \left(E[g(\overline{X}(\overline{\tau}),\overline{\tau})] - \frac{1}{M}\sum_{j=1}^{M} g(\overline{X}(\overline{\tau};\omega_j),\overline{\tau}) \right) \\
&=: \mathcal{E}_C + \mathcal{E}_D + \mathcal{E}_S.
\end{aligned} \tag{46}$$

For a given error tolerance TOL, the goal is to minimize the computational work, which is roughly $\mathcal{O}(M \cdot N) = \mathcal{O}(\mathrm{TOL}_S^{-2}\mathrm{TOL}_T^{-1})$ where TOL_S and TOL_T denote a statistical tolerance and a time discretization tolerance respectively. Thus we obtain

$$\mathrm{TOL}_S = \frac{2}{3}\mathrm{TOL} \quad \text{and} \quad \mathrm{TOL}_T = \frac{1}{3}\mathrm{TOL} \tag{47}$$

by solving

$$\min \mathrm{TOL}_S^{-2}\mathrm{TOL}_T^{-1} \quad \text{subject to} \quad \mathrm{TOL}_S + \mathrm{TOL}_T = \mathrm{TOL}.$$

3.1 Control of the Statistical Error

Let us first introduce some notation. Define the sample average $\mathcal{A}(Y;M)$ and the sample standard deviation $\overline{\sigma}(Y;M)$ of Y by

$$\mathcal{A}(Y;M) := \frac{1}{M}\sum_{j=1}^{M} Y(\omega_j), \quad \overline{\sigma}(Y;M) := \left(\mathcal{A}(Y^2;M) - (\mathcal{A}(Y;M))^2\right)^{\frac{1}{2}}.$$

Then from the Central Limit Theorem, the statistical error $\mathcal{E}_S$ in (46) satisfies

$$|\mathcal{E}_S| \leq \mathrm{E}_S(Y;M) := c_0 \, \frac{\overline{\sigma}(Y;M)}{\sqrt{M}} \tag{48}$$

with probability close to one asymptotically, where $Y = g(\overline{X}_{\overline{\tau}},\overline{\tau})$ and c_0 is a constant corresponding to a confidence interval. For example, $c_0 \geq 1.65$ gives asymptotically the probability greater than 0.90.

3.2 Control of the Time Discretization Error

In this subsection, we present two refinement strategies to control the time discretization error. For a given partition $0 = t_0 < t_1 < \ldots < t_N = T$, the piecewise constant mesh function Δt is defined by (12) and the corresponding number $N(\Delta t)$ of steps is

$$N(\Delta t) := \int_0^T \frac{1}{\Delta t(s)} \mathrm{d}s \, .$$

Then the optimal choice of the time steps is formulated by minimizing the computational work $E[N(\Delta t)]$ such that $\Delta t \in \mathcal{K}$ subject to given accuracy constraints. The feasible set $\mathcal{K}$ for the mesh function Δt is defined by

$$\mathcal{K} := \{\Delta t : \Delta t \text{ is stochastic, positive and piecewise constant on } [0,T] \text{ for each realization } \}.$$

Total Time Discretization Error

The goal is to make the total time discretization error, $\mathcal{E}_T = \mathcal{E}_C + \mathcal{E}_D$ defined in (4), bounded by a given time discretization error tolerance TOL_T in (47). Therefore the accuracy constraint is

$$E\left[\sum_{n=0}^{N-1} r_n\right] \leq \mathrm{TOL}_T \tag{49}$$

where the error indicators r_n are defined for $n = 0, 1, \ldots, N-1$, by

$$\begin{aligned} r_n := \Big| & \mathbf{1}_{t_{n+1} \leq \overline{\tau}} \rho_n \Delta t_n^2 \\ & + \Big(g\big(\mathrm{proj}_{\partial D} \frac{1}{2}(\overline{X}(t_n) + \overline{X}(t_{n+1})), \frac{1}{2}(t_n + t_{n+1})\big) - g(\overline{X}_{\overline{\tau}}, \overline{\tau})\Big) \hat{P}_{\overline{X},n} \Big| \end{aligned} \tag{50}$$

with ρ_n in (27) and $\hat{P}_{\overline{X},n}$ in (15) and $\mathrm{proj}_{\partial D}$ the orthogonal projection to ∂D.

To have as few time steps as possible, we try to make

$$r_n(\omega) = \mathrm{constant}, \quad \forall n \text{ and } \forall \omega$$

and by (49) the natural choice of the constant is then

$$r_n(\omega) = \frac{\mathrm{TOL}_T}{E[N]}, \quad \forall n \text{ and } \forall \omega. \tag{51}$$

The choice (51) is optimal in the case without stopping boundary, see [15], [17], i.e. without the second term in (50). Numerical tests on one dimensional processes show that the error $\mathcal{E}_D$ in (46), corresponding the second term in (50), converges exponentially fast as the number of adaptive steps is increased. Therefore an over-refinement in this part of the error does not seem to cost much. Note that in practice the quantity $E[N]$ is not known and we can only estimate it by the sample average

$\overline{N}[j] := \mathcal{A}(N; M[j])$ of the final number of time steps in the jth batch of $M[j]$ numbers of realizations. Then the statistical error, $|E[N] - \overline{N}[j]|$, is bounded by $\mathtt{E}_S(N; M[j])$, with probability close to one, by the same argument as in (48).

To achieve (51), start with an initial mesh $\Delta t[1]$ and then specify iteratively a new partition $\Delta t[k+1]$ from $\Delta t[k]$, using the following refinement strategy: for each realization in the mth batch and for all time steps $n = 0, 1, \ldots, N[k] - 1$,

```
if $\left( r_n[k] \geq \frac{\mathrm{TOL}_T}{\overline{N}[m-1]} \right)$ then                (52)
        divide $\Delta t_n[k]$ into 2 equal substeps, and
        generate the intermediate value of $W$ by Brownian bridges (5)
else let the new step be the same as the old
endif,
```

with the stopping criterion: for each realization of the mth batch

$$\textbf{if} \left(\max_{1 \leq n \leq N[k]} r_n[k] < S \frac{\mathrm{TOL}_T}{\overline{N}[m-1]} \right) \textbf{ then } \text{stop.} \tag{53}$$

Here S is a given constant, motivated as follows: we want the maximal error indicator to decay quickly to the stopping level $S\mathrm{TOL}_T/\overline{N}$, but when almost all r_n satisfy $r_n \leq \mathrm{TOL}_T/\overline{N}$, the reduction of the error may be slow. The constant S is introduced to cure this slow reduction.

Splitting of the Time Discretization Error

Let us compare the adaptive algorithm (52)-(53) with the following *ad hoc* refinement algorithm. First we split the time discretization tolerance $\mathrm{TOL}_T = \mathrm{TOL}_C + \mathrm{TOL}_D$ by $\mathrm{TOL}_C = \mathrm{TOL}_D = \mathrm{TOL}_T/2$ and define the error indicators r_n^C and r_n^D by

$$\begin{aligned} r_n^C &:= \mathbf{1}_{t_{n+1} \leq \overline{\tau}} |\rho_n| \Delta t_n^2 \\ r_n^D &:= \left| g\big(\mathrm{proj}_{\partial D} \frac{1}{2}(\overline{X}(t_n) + \overline{X}(t_{n+1})), \frac{1}{2}(t_n + t_{n+1})\big) - g(\overline{X}_{\overline{\tau}}, \overline{\tau}) \right| \hat{P}_{\overline{X},n} \end{aligned} \tag{54}$$

with ρ_n in (27) and $\hat{P}_{\overline{X},n}$ in (15). This alternative refinement strategy is to take into account the computational observation that only a few time intervals for each realization have large error indicators r_n^D compared to the others, see Fig. 2, an illustrative Monte Carlo realization of r_n^D for Example 1 in Sect. 4.

Start the algorithm with an initial mesh $\Delta t[1]$ and then specify iteratively a new partition $\Delta t[k+1]$ from $\Delta t[k]$ using following refinement strategy: for each realization in the mth batch and for all time step $n = 0, 1, \ldots, N[k] - 1$,

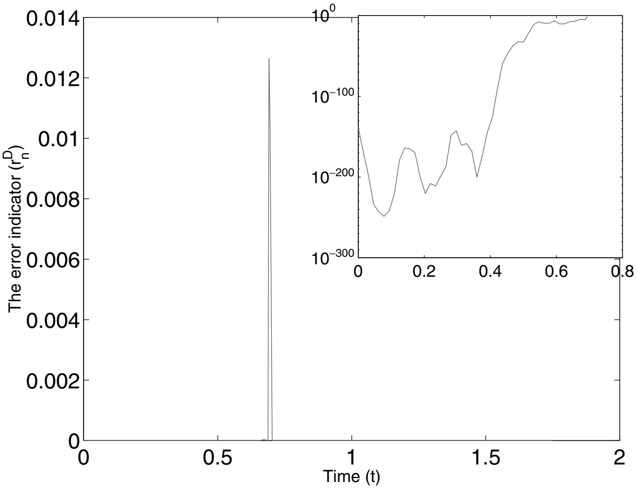


Fig. 2. Example 1: An illustrative Monte Carlo realization of r_n^D with $\mathrm{TOL} = 0.1$

$$\textbf{if } \left(r_n^C[k] \geq \frac{\mathrm{TOL}_C}{\overline{N}[m-1]} \textbf{ or } r_n^D[k] \geq \mathrm{TOL}_D \right) \textbf{ then}, \tag{55}$$

divide $\Delta t_n[k]$ into 2 equal substeps

else let the new step be the same as the old one

endif.

until the following stopping criteria is fulfilled: for each realization of the mth batch

$$\textbf{if } \left(\max_{1 \leq n \leq N[k]} r_n^C[k] < S_C \frac{\mathrm{TOL}_C}{\overline{N}[m-1]} \textbf{ and } \max_{1 \leq n \leq N[k]} r_n^D[k] < S_D \mathrm{TOL}_D \right) \tag{56}$$

then stop.

Here S_C and S_D are given constants to cure the slow reduction when almost all r_n^C or r_n^D satisfy $r_n^C \leq \mathrm{TOL}_C/\overline{N}$ or $r_n^D \leq \mathrm{TOL}_D$.

3.3 The Adaptive Algorithms

The adaptive stochastic time stepping algorithms have structures similar to a basic Monte Carlo algorithm, with an additional inner loop for individual mesh refinement for each realization of a Brownian motion. First we split the specified error tolerance by (47): the outer loop computes the batches of realizations of $\overline{X}$, until an estimate for the statistical error (48) is below the tolerance, TOL_S; then in the inner loop, for each realization, we apply our refinement strategy (52) or (55) to a given initial mesh iteratively until the error indicators satisfy the stopping criteria (53) or (56) with a given time discretization tolerance TOL_T. This procedure, in the inner loop, needs to sample the Wiener process W on finer partitions, given its values on coarser, which is accomplished by Brownian bridge refinements (5).

The adaptive algorithm based on the refinement (52) and the stopping (53) is called `Algorithm A` and the algorithm based on the refinement (55) and the stopping (56) is called `Algorithm B`. We first describe `Algorithm A` in detail and define the additional changes for `Algorithm B` afterwards.

`Algorithm A`

Initialization
Choose:

1. an error tolerance, $\mathrm{TOL} \equiv \mathrm{TOL}_S + \mathrm{TOL}_T$,
2. a number $N[1]$ of initial uniform steps $\Delta t[1]$ for $[0, T]$, with TOL $N[1]$ bounded from above and below by positive constants, and set $\overline{N}[0] = N[1]$,
3. a number $M[1]$ of initial realizations, with TOL^2 $M[1]$ bounded from above and below by positive constants,
4. the stopping constant S in (53),
5. a positive constant c_0 for a confidence interval and an integer MCH$\geq$ 2 to determine the number of realizations in (58),
6. a constant Δx for the difference quotient in (43), see Remark 4.

Set the iteration counter for realization batches $m = 1$ and the stochastic error to $\mathtt{E}_S[m] = +\infty$.

```
Do while ( $\mathtt{E}_S[m] > \mathrm{TOL}_S$ )
    For realizations $j = 1, \ldots, M[m]$
        Set the number of time levels for realization $j$ to $k = 1$ and set the error
        indicator to $r[k] = +\infty$.
        Start with the initial partition $\Delta t[k]$ and generate $\Delta W[k]$.
        Compute for realization $j$, $g(\overline{X}(T))[J]$ and $N[J]$ by calling
        routine Control-Time-Error where $k = J$ is the number of
        final time levels for an accurate mesh of this realization.
    end-for
    Compute the sample average $Eg \equiv \mathcal{A}\left(g(\overline{X}(T)); M[m]\right)$, the sample standard deviation $\mathcal{S}[m] \equiv \mathcal{S}(g(\overline{X}(T)); M[m])$ and the a posteriori bound for
    the statistical error $\mathtt{E}_S[m] \equiv \mathtt{E}_S(g(\overline{X}(T)); M[m])$ in (48).
    if ( $\mathtt{E}_S[m] > \mathrm{TOL}_S$ )
        Discard all old $M[m]$ realizations and determine a larger $M[m+1]$ by
        change_$M$ ($M[m]$, $\mathcal{S}[m]$, $\mathrm{TOL}_S$; $M[m+1]$), in (58), and update
        $\overline{N} = \mathcal{A}\,(N[J]; M[m])$, where the random variable $N[J]$ is the final number of time steps on each realization.
    end-if
    Increase $m$ by 1.
end-do
```

Accept Eg as an approximation of $E[g(X(T))]$, since the estimate of the computational error is bounded by TOL.

```
routine Control-Time-Error$\big(\Delta t[k], \Delta W[k], r[k], \overline{N}[m-1];$
                                    $g(\overline{X}(T))[J], N[J]\big)$

  Do while ( $r[k]$ violates the stopping (53) )
    Compute the Euler approximation $\overline{X}[k]$ in (6) and the error indicator $r[k]$ in
    (50) on $\Delta t[k]$ with the known Wiener increments $\Delta W[k]$.
    if ( $r[k]$ violates the stopping (53) )
      Do the refinement process (52) to compute $\Delta t[k+1]$ from $\Delta t[k]$ and
      compute $\Delta W[k+1]$ from $\Delta W[k]$ using Brownian bridges (57).
    end-if
    Increase $k$ by 1.
  end-do
  Set the number of the final level $J = k-1$.

end of Control-Time-Error
```

At the new time steps $t'_i \equiv (t_i[k] + t_{i+1}[k])/2$, on level $k+1$, the new sample points from W are constructed by the Brownian bridge, cf. [10],

$$W^\ell(t'_i) = \frac{1}{2}\big(W^\ell(t_i[k]) + W^\ell(t_{i+1}[k])\big) + z_i^\ell \tag{57}$$

where z_i^ℓ are independent random variables, also independent of $W(t_j[k])$ for all i, j and ℓ, and each component z_i^ℓ is normal distributed with mean zero and variance $(t_{i+1}[k] - t_i[k])/4$.

```
routine change_$M$ $(M_{in}, \mathcal{S}_{in}, \text{TOL}_S; M_{out})$
```

$$\begin{aligned} M^* &= \min\Big\{\text{integer part } \Big(\frac{c_0\ \mathcal{S}_{in}}{\text{TOL}_S}\Big)^2,\ \text{MCH} \times M_{in}\Big\} \\ n &= \text{integer part } (\log_2 M^*) + 1 \\ M_{out} &= 2^n. \end{aligned} \tag{58}$$

```
end of change_$M$
```

Here MCH ≥ 2 is a positive integer parameter introduced to avoid a large new number of realizations in the next batch due to a possibly inaccurate sample standard deviation $\overline{\sigma}[m]$. Indeed, $M[m+1]$ cannot be greater than MCH × $M[m]$.

Algorithm B

In addition to the **Initialization** of Algorithm A, choose the error tolerances $\text{TOL}_T = \text{TOL}_C + \text{TOL}_D$ and the stopping constants S_C and S_D in (56). Inside the **Do while** loop of Algorithm A, use $(r^C[k], r^D[k])$ in (54) instead of $r[k]$ and the refinement (55) and stopping (56).

4 Numerical Experiments

This section presents numerical results from a one dimensional problem with a `C++` implementation of `Algorithm A` and `Algorithm B` described in Sect. 3 and for a two dimensional problem with a corner singularity with `Matlab` implementation. The numerical results in 1D are obtained using the pseudo-random number generator, `drand48()`, in standard C library functions. The Box-Muller method is used to generate standard Gaussian random variable from the uniformly distributed pseudo-random numbers, see for example [11].

4.1 A One Dimensional Domain

In all computations, the following constants are chosen for the initialization of both `Algorithm A` and `Algorithm B`: the number of time steps in the initial partition, $\Delta t[1]$, of $[0,T]$ is $N[1]=4$; the initial number of realizations is $M[1]=128$; the stopping constant $S=4$ is used in (53) and $S_C=4, S_D=1$ in (56); the constants to determine the number of realizations in (58) are $c_0=1.65$ and $MCH=16$, and the constant $\Delta x = \mathrm{TOL}_T^{1/4}$ is used for the difference quotient in (43).

To describe the behavior of the adaptive algorithm, let us first define some notation. The index Q, which is the ratio between the approximate error and the exact error, is defined by

$$Q := \frac{E_{approx}}{E_{exact}} := \frac{\mathrm{E}_S + |\mathrm{E}_T|}{\left|E[g(X_\tau,\tau)] - \mathcal{A}(g(\overline{X}_{\overline{\tau}},\overline{\tau});M)\right|}. \tag{59}$$

Here the statistical error E_S is defined by (48) and the time discretization error E_T is defined by

$$\mathrm{E}_T := \mathcal{A}\left(\sum_{n=0}^{N-1} \mathbf{1}_{t_{n+1}\le\overline{\tau}}\rho_n \Delta t_n^2 + \left(g(\lambda,\frac{1}{2}(t_n+t_{n+1})) - g(\overline{X}_{\overline{\tau}},\overline{\tau})\right)\hat{P}_{\overline{X},n};M\right),$$

where λ defines the domain $D=(-\infty,\lambda)$.

Example 1. Consider (2) with $d=1$,

$$a(t,x)=\frac{11}{36}x,\quad b(t,x)=\frac{1}{6}x,\quad t\in[0,T],\ \ x\in(-\infty,2)$$

and the initial condition $X(0)=1.6$ and $T=2$. For $g(x,t)=x^3\mathrm{e}^{-t}$ with $x\in\mathbb{R}$, this problem has the exact solution $E[g(X_\tau,\tau)]=u(X(0),0)=X(0)^3$, where the solution u of the Kolmogorov backward equation (28) is $u(x,t)=x^3\mathrm{e}^{-t}$.

To check the behavior of the error expansion described in Sect. 2, Example 1 is constructed such that most of the realizations exit at $\overline{\tau}<T$, for instance, with $\mathrm{TOL}=0.01$, 99% of the paths exit at $\overline{\tau}<T$ and $\mathcal{A}(\overline{\tau};M)\simeq 0.77$.

Table 1. Example 1: Comparisons of the final number of the realizations, M, the sample average of the final number of steps, $\mathcal{A}(N;M)$, the sample standard deviation of the final number of steps, $\overline{\sigma}(N;M)$, and the exact error, E_{exact} for different error tolerances, TOL

		Algorithm A			Algorithm B		
TOL	M	$\mathcal{A}(N;M)$	$\overline{\sigma}(N;M)$	E_{exact}	$\mathcal{A}(N;M)$	$\overline{\sigma}(N;M)$	E_{exact}
0.5	2^7	27	11.7	0.028	24	6.9	0.02
0.1	2^{11}	81	30.6	0.024	84	25.8	0.06
0.05	2^{13}	126	44.0	0.015	158	54.2	0.02
0.01	2^{18}	453	170.7	0.003	700	287.7	0.005

Table 1 shows the comparisons between `Algorithm A` and `Algorithm B` for the computational results of Example 1. As the error tolerance TOL decreases, E_{exact} decreases and is bounded by a given TOL. The sample standard deviation of the number of time steps is around 35% of the average of the number of time steps. The histogram in Fig. 5 indeed shows that highly varying step sizes are used for individual realizations.

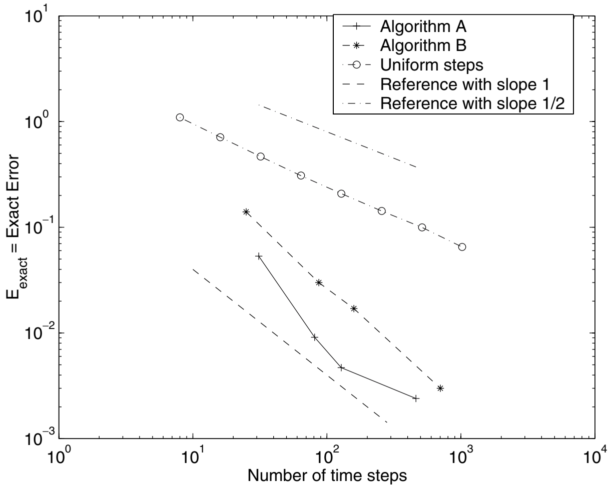


Fig. 3. Example 1: Comparison of the convergence rates with uniform and adaptive meshes. The convergence rate of the adaptive method is of order N^{-1} with N adaptive time steps, while the rate for the uniform method is of order $N^{-1/2}$ with N uniform time steps

To check the accuracy of the error estimate in Sect. 2, choose the number of realizations M sufficiently large so that the total statistical error is small compared to the time discretization error. Here we use $M = 2^{22} = 4,194,304$, which makes the statistical error approximately 0.001. Then the comparison of the convergence between the uniform and the adaptive method is shown in Fig. 3. The x-axis denotes the number of time steps for the uniform method and the sample average of the final number of steps for the adaptive method. The y-axis is the exact error E_{exact} defined by (59). The number of steps $N = 2^k, k = 3, 4, \ldots, 10$ are used for the uniform method and for adaptive method the tolerances $\mathrm{TOL} = 0.5, 0.1, 0.05$ and 0.01 are used. Figure 3 shows that the convergence rate of the adaptive method is of order N^{-1} with N adaptive time steps, while the uniform method converges with the rate $N^{-1/2}$ with N uniform time steps.

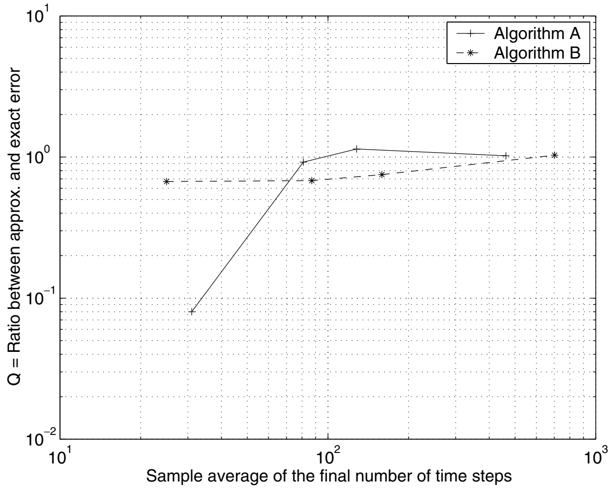


Fig. 4. Example 1: The ratio of the approximate and exact error on adaptive mesh. The ratio tends to 1 as the number of time steps increases

Figure 4 shows the convergence of the ratio Q between the approximate and the exact error in (59), still with $M = 2^{22}$ so that the statistical error is negligible. As predicted by Theorem 1 and 2, Fig. 4 shows that the ratio Q tends to 1 as N increases. From Fig. 3 and 4, `Algorithm B` seems more stable than `Algorithm A` for Example 1, on the other hand `Algorithm A` achieves smaller exact error for the same number of time steps.

Figure 5 shows the histogram of the step sizes depending on the distance from the boundary with $\mathrm{TOL} = 0.05$ and $M = 2^{22}$ realizations of `Algorithm A`. The histogram of `Algorithm B` also has a similar appearance. The x-axis denotes base 2 log-scale of the step size, ranging from 2^{-35} to 2^{-5}, the y-axis denotes base 2 log-scale of the distance from the boundary, ranging from 2^{-20} to 1, and the z-axis denotes base 2 log-scale of the number of steps. To compensate the large error near the boundary, relatively small step sizes are used close to the boundary compared to further away from the boundary.

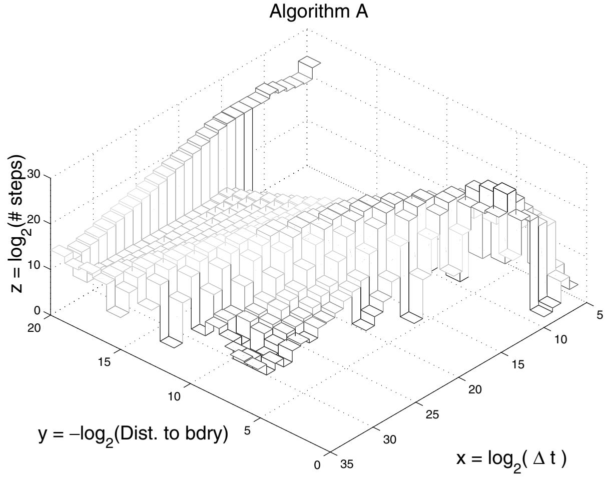


Fig. 5. Example 1: The histogram of the step sizes depending on the distance from the boundary using `Algorithm A`. Relatively small step sizes are used close to the boundary to improve the accuracy

4.2 A Two Dimensional Domain

The methods described in the previous sections are implemented for a two dimensional domain, with a corner, which does not satisfy the conditions of smoothness required in [7]. The idea is to find out if the adaptive method can give some improvements to the standard Euler algorithm even though the approximations of the exit probabilities are somewhat incorrect. The known methods for improving the time discretization error rely on the possibility to locally approximate the boundary by its tangent plane. This is obviously difficult in the case for domains with sharp corners.

The method used by Gobet [7] is strictly dependent on the value of the exit probability of the continuous Euler process between two time levels. When dealing with domains with non smooth boundary, for example corners, this method may give large errors, since it makes use of the assumption that the boundary can be locally approximated by its tangent plane. A domain with a sharp corner, however, cannot be locally well approximated as a tangent plane.

A prerequisite for any adaptive method is some sort of error estimate to decide which regions need refining and which do not. However, one of the advantages of adaptive methods in general is that they do not require a great deal of exactness in this error estimate in order to function in a satisfactory manner. In fact, it is often enough to check that the behavior of the error estimate is qualitatively similar to the real error, i.e. that the estimate increases and decreases similarly as the actual error.

The domain D for our test problem is chosen to be the one shown in Fig. 6 and in this domain we consider the problem

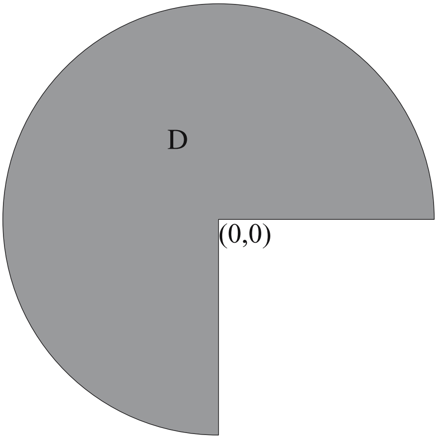


Fig. 6. The computational domain D with a corner at the origin

$$\begin{aligned} &u_t + \frac{1}{2}\Delta u = 0, \quad t < T \\ &u(\cdot, T) = g(\cdot, T) \\ &u(x,t) = g(x,t), \quad x \in \partial D \end{aligned} \tag{60}$$

which is solved by the expectation $u(x,t) = E[F(X_\tau, \tau) \mid X_t = x]$ for a pure Brownian motion $\mathrm{d}X_j(t) = \mathrm{d}W^j(t)$, $j = 1, 2$, where $F = g(\cdot, \tau)$ if the process X exited D first at time τ before T, and $F = g(\cdot, T)$ if no exit occurred before T.

The boundary condition is chosen so that the behavior of the process W near the corner has a much greater impact than the behavior near the arc. Therefore, the boundary condition is chosen as $g(x,t) = 10\mathrm{e}^{-\sqrt{x^2+y^2}-0.1t}$ and we let the process start within D, near the corner at the origin. We also choose a large enough radius, $R = 10$, of the arc boundary and short enough time interval, $T = 1$, so it becomes highly unlikely for the process to reach the arc. The goal is to approximate $u(-0.209, 0.249, 0) = 0.544$.

The algorithm for this type of domain differs from the one for smooth domains only in the approximation of the exit probabilities P_i. To apply the algorithms formally, it is assumed that the corner is slightly 'rounded'. In the quadrant $x_1 < 0$ and $x_2 > 0$ it can then be imagined that the corner is a circular arc with infinitely small radius, in which case an inward pointing normal vector from the boundary to a point X_t is simply given by X_t itself. The tangent plane must then be orthogonal to X_t and pass through the corner at the origin. By proceeding in this way the tangent plane is quite easy to find, but it is obviously not a good approximation of the boundary near the corner. Using this crude estimate for the tangent plane it becomes easy to calculate distances to the tangent planes of the points in the Euler path, and thereby to calculate rough estimates of the exit probabilities. Near the corner, these estimates of the exit probabilities will, however, be quite far from correct. In all three quadrants the algorithm will over-estimate the exit probabilities.

The adaptive algorithm proceeds as described in the previous sections but now using the exit probabilities P_i as described above and

$$r_i = \Big(g\big(\mathrm{proj}_{\partial D}\big(\frac{1}{2}(\overline{X}_{t_i} + \overline{X}_{t_{i+1}})\big), \frac{t_i + t_{i+1}}{2}\big) - g(\overline{X}_{\overline{\tau}}, \overline{\tau})\Big)\widehat{P}_i,$$

where $\mathrm{proj}_{\partial D}$ is the orthogonal projection to the boundary ∂D. A dilemma arises when trying to calculate the error estimate for the case when the discrete process crosses the 'tangent plane' but does not exit the domain D. An example of this is shown in Fig. 7. When calculating the exit probability for such a step, Mannella's and Gobet's method would proceed as earlier, and consider that the process indeed has exited the domain. For the adaptive method however, this seems an unnecessarily erroneous way to proceed, and the exit probability is calculated by reflecting the point which has exited back onto the other side of the tangent plane. This procedure results in a completely incorrect exit probability for some steps, but as this does not occur too often, it seems to be an acceptable way of testing the convergence properties of the adaptive algorithm. It is important to note that it is necessary to limit the refinement, for example by limiting the length of the time steps so that the incorrect behavior of the exit probabilities for these few steps will not cause the algorithm to refine indefinitely.

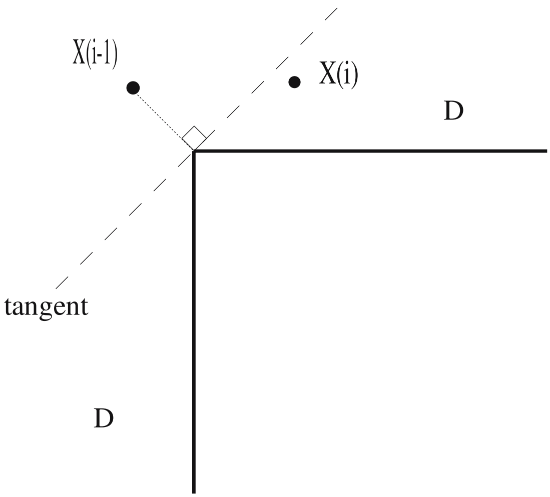


Fig. 7. The discrete process has crossed the 'tangent plane' but is still within D

The solution u of (60) has large derivatives near the corner. This resulted in an even slower convergence rate than $\mathcal{O}(N^{1/2})$ for the standard Euler algorithm, see Fig. 8. Even so, a considerable improvement in the convergence was achieved by using the adaptive algorithm for stochastic differential equations, resulting in a convergence rate which is better than $O(\frac{1}{N})$ and maybe even an exponential rate for this case with $\mathrm{d}X = \mathrm{d}W$, see Fig. 9. As seen in Fig. 9, our implementation of Mannella's and Gobet's method in the corner case gave only a slight improvement to the standard Euler method and was not as effective as the adaptive algorithm. The number of realizations, M, was chosen so that the statistical error was negligible as compared to the time discretization error. For this purpose, $M = 2^{22}$ proved sufficient.

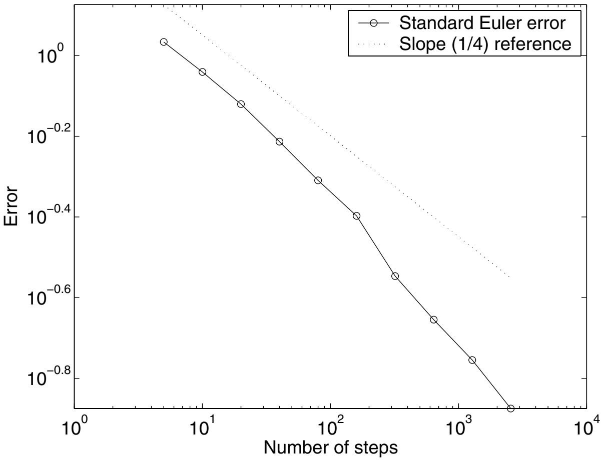


Fig. 8. Error for the Standard Euler method

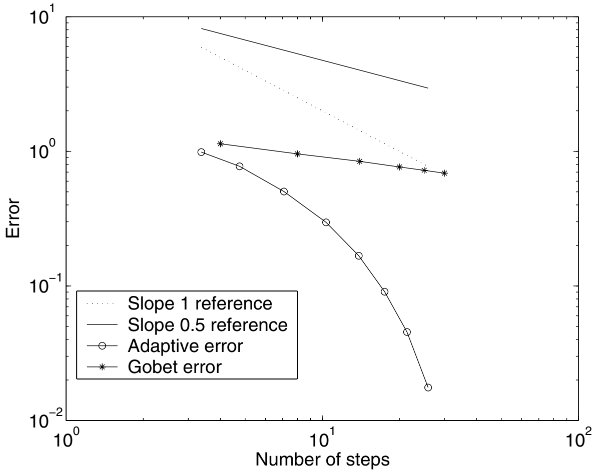


Fig. 9. Error for the adaptive algorithm and Gobet's method

Acknowledgments

The authors thank Michael Tehranchi for the reference [1]. This work is supported by the Swedish Research Council grants 2002-6285 and 2002-4961, the Swedish Foundation for Strategic Research, and the European network HYKE, funded by the EC as contract HPRN-CT-2002-00282.

References

1. Abundo M.: Some conditional crossing results of Brownian motion over a piecewise-linear boundary. Statist. Probab. Lett. **58**, no. 2, 131–145, (2002)
2. Baldi P.: Exact asymptotics for the probability of exit from a domain and applications to simulation. Ann. Probab. **23**, no. 4, 1644–1670, (1995)
3. Baldi P., Caramellino L. and Iovino M.G.: Pricing general barrier options: a numerical approach using sharp large deviations. Math. Finance **9**, no. 4, 293–322, (1999)
4. Bally V. and Talay D.: The law of the Euler scheme for stochastic differential equations, I. Convergence rate of the distribution function. Probab. Theory Related Fields **104**, no. 1, 43–60, (1996)
5. Buchmann F.M.: Computing exit times with the Euler scheme. Research report no. **2003-02**, ETH, (2003).
6. Fleming W.H. and James M.R.: Asymptotic series and exit time probabilities. Ann. Probab. **20**, no. 3, 1369–1384, (1992)
7. Gobet E.: Weak approximation of killed diffusion using Euler schemes. Stochastic Process. Appl. **87**, no. 2, 167–197, (2000)
8. Gobet E.: Euler schemes and half-space approximation for the simulation of diffusion in a domain. ESAIM Probab. Statist. **5**, 261–297, (2001)
9. Jansons K.M. and Lythe G.D.: Efficient numerical solution of stochastic differential equations using exponential timestepping. J. Stat. Phys. **100**, no. 5/6, 1097–1109, (2000)
10. Karatzas I. and Shreve S.E.: Brownian motion and stochastic calculus. Graduate Texts in Mathematics, **113**. Springer-Verlag, New York, (1991)
11. Kloeden P.E. and Platen E.: Numerical solution of stochastic differential equations. Applications of Mathematics, **23**. Springer-Verlag, Berlin, (1992)
12. Lépingle D.: Un schéma d'Euler pour équations différentielles stochastiques réfléchies. C. R. Acad. Sci. Paris Sér. I Math. **316**, no. 6, 601–605, (1993)
13. Mannella R.: Absorbing boundaries and optimal stopping in a stochastic differential equation. Phys. Lett. A **254**, no. 5, 257–262, (1999)
14. Moon K.-S.: Adaptive Algorithms for Deterministic and Stochastic Differential Equations. PhD Thesis, Royal Institute of Technology, Department of Numerical Analysis and Computer Science, Stockholm (2003)
15. Moon K.-S., Szepessy A., Tempone R. and Zouraris G.E.: Convergence rates for adaptive weak approximation of stochastic differential equations. Accepted in Stoch. Anal. Appl. (2005)
16. Petersen W.P. and Buchmann F.M.: Solving Dirichlet problems numerically using the Feynman-Kac representation. BIT **43**, no. 3, 519–540, (2003)
17. Szepessy A., Tempone R. and Zouraris G.E.: Adaptive weak approximation of stochastic differential equations. Comm. Pure Appl. Math. **54**, no. 10, 1169–1214, (2001)
18. Talay D. and Tubaro L.: Expansion of the global error for numerical schemes solving stochastic differential equations. Stochastic Anal. Appl. **8**, no. 4, 483–509, (1990)

The Heterogeneous Multi-Scale Method for Homogenization Problems

Weinan E[1,2] and Björn Engquist[1,3]

[1] Department of Mathematics and PACM, Princeton University, Princeton, NJ 08544, USA
[2] School of Mathematics, Peking University, Beijing 100871, China
weinan@math.princeton.edu
[3] Department of Mathematics, University of Texas at Austin, 1 University Station C1200, Austin, Texas 78712, USA
engquist@math.utexas.edu

Summary. The heterogeneous multi-scale method, a general framework for efficient numerical modeling of problems with multi-scales [15], is applied to a large variety of homogenization problems. These problems can be either linear or nonlinear, periodic or non-periodic, stationary or dynamic. Stability and accuracy issues are analyzed along the lines of the general principles outlined in [15]. Strategies for obtaining the microstructural information are discussed.

Key words: multiscale problems, homogenization, heterogeneous multiscale method

1 Introduction

The heterogeneous multi-scale method (HMM) introduced in [15] is a general methodology for efficient numerical computation of problems with multiple scales and/or multi-levels of physics. When explicit models for macroscale quantities are unavailable or cease to be valid in some part of the computational domain, HMM provides a general, efficient and stable strategy for supplementing the incomplete macroscale model by an explicitly given microscale model. Since its inception, the method has already been applied with encouraging results to several classes of problems, including homogenization problems [3, 15, 18], time scale problems [21], coupling molecular dynamics with linear elasticity [16] as well as general nonlinear thermoelasticity [29], coupling molecular dynamics and hydrodynamics models for complex fluids [39], and stochastic differential equations with multiple scales [17]. Aside from these new developments, HMM also provides a general framework for unifying and extending several important existing multi-scale methods, such as the gas-kinetic scheme [47] and the quasi-continuum method [43].

The purpose of this paper is to discuss thoroughly the application of HMM to homogenization problems. There are two reasons for doing this. One is that a large variety of multi-scale problems are homogenization problems. The other is that we can use homogenization problems as examples for discussing general aspects of HMM, such as its efficiency and accuracy. There is already an extensive literature on the analytical studies of homogenization problems [7, 45]. Building upon this literature, we will carry out analysis of HMM when applied to these problems. In particular, we will discuss how naive application of multiscale methods can fail to approximate the right quantity in some cases.

General homogenization problems can be expressed as

$$L\left(x, \frac{x}{\varepsilon}\right) u^{\varepsilon}(x) = f(x), \qquad x \in \Omega \subset R^{d}, \tag{1}$$

together with some initial and/or boundary conditions, and where L is a linear or nonlinear differential operator. Problems of this type arise in the modeling of properties of a strongly heterogeneous medium such as composite materials, polycrystals and porous medium, where ε is the ratio between the scale of the medium and the scale of the microstructures in the medium. We commonly refer to $y = \frac{x}{\varepsilon}$ as the fast variable. Dependence on y is sometimes assumed to be periodic, but this is unnecessary for most of the numerical techniques we will discuss. What is important is that there is *scale separation* between the microstructures and the system size.

From a numerical point of view, resolving the microscopic details of (1) using typical numerical methods would require a cost of $O(\varepsilon^{-d})$ or more. This often becomes prohibitively expensive since $\varepsilon \ll 1$. One way of avoiding this is to solve instead the homogenized equation

$$\bar{L}(x)U(x) = f(x) \tag{2}$$

$u^{\varepsilon} \rightarrow U$ as $\varepsilon \rightarrow 0$. Indeed in many cases, this is a satisfactory approach. However, quite often it is difficult to obtain an explicit homogenized equation in the form of (2). In addition, the homogenized equation may neglect crucial microstructural information that are important for the applications. Therefore it is highly desirable to design numerical methods that are based on the original microscale model (1), but capture the large scale behavior and microscale statistics efficiently.

Numerical computation for homogenization problems was pioneered by Babuska [4] for elliptic problems and in [19] for hyperbolic problems. For the linear variational homogenization problem (see (3)) in one dimension, Babuska proposed to use a finite element method on a macroscale grid but with modified basis functions that are obtained from solving (3) with $f = 0$ and nodal boundary conditions [4, 6]. In this way the finite element trial functions are endowed with the correct microstructure. Babuska's idea was extended to higher dimensions in [25]. Other ideas based on modifying basis functions are found in [26, 8]. These methods require an overhead of $O(\varepsilon^{-d})$ operations for constructing the basis functions, a cost that is comparable to that of solving the original problem on a fine grid by an efficient standard method. Engquist and Runborg proposed to process the matrix obtained from fine-grid discretization of the microscale operator using wavelet basis and obtain effective operators for the macroscale properties of the solutions [20]. Schwab and co-workers make use of the analytical tool of multi-scale test functions developed by Nguetseng, E and Allaire [35, 14, 1], and analyzed finite element methods that use such test functions

[42]. The cost of this method is independent of ε but at the present time it is limited to problems for which the microstructure is periodic. For dynamic problems, Engquist proposed the idea of sampling in order to capture the effect of microscale statistics on the macroscale using a coarse grid [19]. See also [40] for related discussion on these issues.

Each of these techniques seems to be limited to a particular class of problems. In contrast, HMM is a general technique that can handle a large variety of homogenization problems, linear or nonlinear, periodic or non-periodic, stationary or dynamic. It also accomplishes this with minimum cost and complexity.

We end this introduction with a brief summary of HMM for homogenization problems. HMM takes a top-down approach for dealing with multiscale problems, namely it pretends that the macroscale model is known and uses a conventional macroscale numerical method as the starting point. In the process of implementing this macroscale numerical method, HMM replaces function evaluations that involve unknown quantities, due to the fact that the macroscale model is not really explicitly known, by measurements from numerical experiments using the microscale model. This is a key new idea in HMM. It is a general principle that applies to a large class of problems.

In more concrete terms, there are two main components in HMM: An overall macroscopic scheme for U and estimating the missing macroscopic data from the microscopic model. The right overall macroscopic scheme depends on the nature of the problem and typically there are more than one choice. For variational problems, we can use the standard finite element method. For dynamic problems that are conservative, we may use finite volume type of methods that take advantage of the conservative nature of the problem. Examples include the Godunov scheme, Lax-Friedrichs scheme, and the discontinuous Galerkin method [28]. For dynamic problems that are non-conservative, we can simply use a standard ODE solver, such as the forward Euler or the Runge-Kutta method, and estimate the forcing term at the right hand side of the equation using the microscale model.

The key to the efficiency of HMM is in the data estimation. For problems with scale separation, the needed macroscale data can be estimated to satisfactory accuracy by solving the microscale problem on domains of microscopic sizes. In the following we will discuss how this can be done and what the pitfalls are for doing this.

2 Variational Problems

Consider the variational problem

$$\min_{u\in H_0^1(D)} \left\{ \frac{1}{2}\int_D \sum_{i,j} a_{i,j}\left(x,\frac{x}{\varepsilon}\right)\frac{\partial u}{\partial x_i}\frac{\partial u}{\partial x_j}\mathrm{d}x - \int_D f(x)u(x)\mathrm{d}x \right\} \tag{3}$$

where $a(x,y)$ is smooth and periodic in y with period $I = [-\frac{1}{2},\frac{1}{2}]^d$, f is smooth. This is the standard problem considered in homogenization theory. The methods that we will describe, however, applies also to the case when the coefficient is of a more general form $a^\varepsilon(x)$.

To construct a HMM finite element method, we choose as the macroscale solver the conventional finite element method. As an example, we will use the standard C^0 piecewise linear element, on a triangulation T_H where H denotes the element size. We will denote by X_H the finite element space.

The data that need to be estimated from the microscale model is the effective stiffness matrix on T_H:

$$A = (A_{ij}) \tag{4}$$

where

$$A_{ij} = \int_D (\nabla \varPhi_i(x))^T \mathcal{A}_H(x) \nabla \varPhi_j(x)\, \mathrm{d}x \tag{5}$$

and $\mathcal{A}_H(x)$ is the effective coefficient (say conductivity) at the scale H and $\{\varPhi_i(x)\}$ are the basis functions for X_H. Had we known $\mathcal{A}_H(x)$, we could have evaluated (A_{ij}) by numerical quadrature. Let $f_{ij}(x) = (\nabla \varPhi_i(x))^T \mathcal{A}_H(x) \cdot \nabla \varPhi_j(x)$, then

$$A_{ij} = \int_D f_{ij}(x) \mathrm{d}x \simeq \sum_{K \in T_H} |K| \sum_{x_\ell \in K} w_\ell f_{ij}(x_\ell) \tag{6}$$

where $\{x_\ell\}$ and $\{w_\ell\}$ are the quadrature points and weights respectively, $|K|$ is the volume of the element K. Therefore our problem reduces to the approximation of $\{f_{ij}(x_\ell)\}$. This will be done by solving the original microscale model locally around each quadrature point $\{x_\ell\}$.

Let us first discuss the case when $i = j$. Our formulation of the local microscale problem is motivated by the following general principle. Given an arbitrary microscale variational problem $\min_u e(u)$,

$$\min_u e(u) = \min_U E(U) \tag{7}$$

where

$$E(U) = \min_{u: Qu = U} e(u) \tag{8}$$

Here Q is some compression operator. Using this, computing A_{ii} is equivalent to computing $E(\varPhi_i)$.

Let $I_\delta(x_\ell)$ be a cube of size δ. On each cell $I_\delta(x_\ell)$, we define Q as follows: $Qu = U$ if

$$\frac{1}{\delta^d} \int_{I_\delta(x_\ell)} u(x)\, \mathrm{d}x = U(x_\ell) \tag{9}$$

$$\frac{1}{\delta^d} \int_{I_\delta(x_\ell)} \nabla u(x)\, \mathrm{d}x = \nabla U(x_\ell) \tag{10}$$

With these, we can define the microscale problem to be solved in $I_\delta(x_\ell)$. Let φ_i^ε be the solution of the following problem

$$\min_{Qu=\Phi_i} \int_{I_\delta(x_\ell)} (\nabla u(x))^T a^\varepsilon(x) \nabla u(x) \mathrm{d}x, \tag{11}$$

we approximate $f_{ii}(x_\ell)$ by

$$f_{ii}(x_\ell) \simeq \frac{1}{\delta^d} \int_{I_\delta(x_\ell)} (\nabla \varphi_i^\varepsilon(x))^T a^\varepsilon(x) \nabla \varphi_i^\varepsilon(x) \, \mathrm{d}x \tag{12}$$

Similarly we approximate $f_{ij}(x_\ell)$ by

$$f_{ij}(x_\ell) \simeq \frac{1}{\delta^d} \int_{I_\delta(x_\ell)} (\nabla \varphi_i^\varepsilon(x))^T a^\varepsilon(x) \nabla \varphi_j^\varepsilon(x) \, \mathrm{d}x \tag{13}$$

Knowing $\{f_{ij}(x_\ell)\}$, we obtain the stiffness matrix A by (6).

(11) is equivalent to solving

$$-\mathrm{div}(a^\varepsilon(x) \nabla u(x)) = 0 \quad \text{on } I_\delta(x_\ell) \tag{14}$$

with boundary condition

$$a^\varepsilon(x) \frac{\partial u}{\partial n} = \lambda^T \hat{n} \quad \text{on } \partial I_\delta(x_\ell) \tag{15}$$

where λ is the Lagrange multiplier for the constraints that $\frac{1}{\delta^d} \int_{I_\delta(x_\ell)} \nabla u \ \mathrm{d}x = (\nabla \Phi_i)(x_\ell)$, $\hat{n}$ is the outward normal on $\partial I_\delta(x_\ell)$. Sometimes it is more convenient to consider the boundary condition that $u(x) - \Phi_i(x)$ is periodic with period $I_\delta(x_\ell)$. Other boundary conditions might also be contemplated. Overall, the effect of boundary conditions for the microscale solver is still an issue that needs systematic investigation.

The savings compared with solving the full fine scale problem comes from the fact that we can choose $I_\delta(x_\ell)$ to be smaller than K. The size of $I_\delta(x_\ell)$ is determined by many factors, including the accuracy and cost requirement, the degree of scale separation, and the microstructure in $a^\varepsilon(x)$. One purpose for the error estimates that we present in [18] is to give guidelines on how to select $I_\delta(x_\ell)$. If $a^\varepsilon(x) = a(x, x/\varepsilon)$ and $a(x, y)$ is periodic in y, we can simply choose $I_\delta(x_\ell)$ to be $x_\ell + \varepsilon I$, i.e., $\delta = \varepsilon$. If $a(x, y)$ is random, then δ should be a few times larger than the local correlation length of a^ε. In the former case, the total cost is independent of ε. In the latter case, the total cost depends only weakly on ε (see [33]).

The numerical performance of HMM including comparison with other methods is discussed in [33].

It is interesting to compare the philosophy of HMM with the methods based on modified basis functions [6, 25, 31]. HMM uses standard finite element spaces at the macroscale, but approximate *directly* the effective macroscale operator by solving the microscale problem on a small subset of each element. The computational complexity of HMM does not increase as ε is decreased. The flexibility is comparable to that of the macroscale finite element method. Indeed, as we show below, HMM extends easily to nonlinear and time-dependent problems. In contrast the method proposed

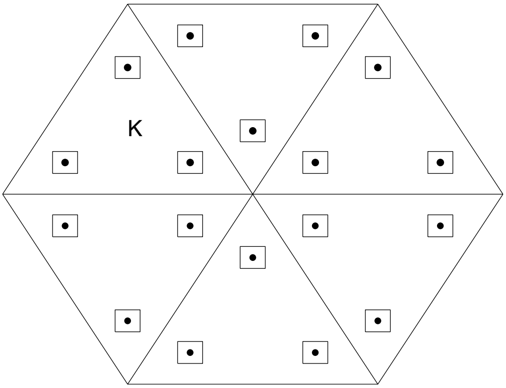


Fig. 1. Illustration of HMM. The dots are the quadrature points. The little squares are the microcell

in [25] was based on the idea of replacing the standard finite element basis functions with functions having the correct microstructures. This has the disadvantage that the basis functions are expensive to compute. In fact the overhead of solving for the basis functions is already comparable to the cost of solving the original microscale problem. It is also difficult to extend such methods to nonlinear problems, or problems for which the microstructure evolves in time.

It is also of interest to compare HMM with the approach of solving directly the homogenized equation. An interesting variant of this approach is to compute the homogenized coefficient matrix by solving the microscale problem on a block of suitable size and then use the result to compute the homogenized equation [12, 37]. For the model problem discussed here, this approach bears some similarity with that of HMM. In fact, the micro-cell problem solved in HMM is an analog of the cell problem for the homogenized equation. However, there is an important difference, namely that HMM does not assume any specific form of the homogenized equation. This flexibility makes it possible to extend to a wider class of problems.

Having the HMM solution U_{HMM}, one can obtain locally the microstructural information using an idea in [37]. Assume that we are interested in recovering u^ε and ∇u^ε only in the subdomain D. Consider the following auxiliary problem:

$$\begin{cases} -\mathrm{div}(a^\varepsilon(x)\nabla\tilde{u}(x)) = f(x) & x \in D_\eta, \\ \tilde{u}(x) = U_{\mathrm{HMM}}(x) & x \in \partial D_\eta, \end{cases} \tag{16}$$

where D_η satisfies $D \subset D_\eta \subset \Omega$ and $\mathrm{dist}(\partial D, \partial D_\eta) = \eta$. We then have: There exists a constant C such that

$$\left(\frac{1}{|D|}\int_D |\nabla(u^\varepsilon - \tilde{u})|^2 \mathrm{d}x\right)^{1/2} \leq \frac{C}{\eta}(\|u^\varepsilon - U_{\mathrm{HMM}}\|_{L^\infty(D_\eta)} + \|u^\varepsilon - u^\varepsilon\|_{L^\infty(D_\eta)}). \tag{17}$$

where $|D|$ is the volume of D.

We turn our attention now to nonlinear problems.

Consider a problem of the type

$$\min_{u \in H_0^1(D)} \int_D W\left(x, \frac{x}{\varepsilon}, u, \nabla u\right) \mathrm{d}x \tag{18}$$

W has to satisfy certain conditions in order to guarantee that this problem has a solution. The homogenized problem takes the form [34]

$$\min_{U \in H_0^1(D)} \int_D \bar{W}(x, U, \nabla U) \mathrm{d}x$$

In general it is quite difficult to find analytically or numerically $\bar{W}$, and we will assume that we do not have explicit knowledge of $\bar{W}$.

To apply HMM to this problem, we will select the standard piecewise linear finite element method as the macro scheme. The macroscale data that we need to estimate are $E(U) = \int_D \bar{W}(x, U, \nabla U)\mathrm{d}x$ for $U \in V_H$ and the variational derivative of $E(U)$.

To estimate $\int_D \bar{W}(x, U, \nabla U)$, we proceed as before, namely for each $U \in V_H$, $K \in T_H$, and each quadrature point $x_K \in K$, we let

$$\tilde{F}(U)(x_K) = \min \frac{1}{\varepsilon^d} \int_{x_K + \varepsilon I} W\left(x, \frac{x}{\varepsilon}, u, \nabla u\right) \mathrm{d}x \tag{19}$$

subject to the condition that $u(x) - U(x)$ is periodic with period εI. We will use this value in the numerical quadrature for approximating $E(U)$ in the same way as before, e.g.

$$\tilde{E}(U) = \sum_K |K| \tilde{F}(U)(x_K) \tag{20}$$

To compute the variational derivative of $\tilde{E}(U)$, we let $U, V \in V_H$, and denote by u_U the minimizer of (19). We then have

Lemma 1.

$$\left(\frac{\delta \tilde{F}(U)(x_K)}{\delta U}, V \right) = \int_{x_K + \varepsilon I} \frac{\delta W}{\delta u}(u_U) V \mathrm{d}x$$

This lemma says that we do not need to worry about the variation caused by the dependence of the boundary condition in (19) on U. It therefore gives us a very convenient way of computing the variational derivative – the effective Euler-Lagrange operator. As we will see in the proof, the lemma is applicable to general variational problems.

Proof. For simplicity of notation, for $\delta > 0$, denote by u_δ the minimizer of (19) corresponding to $U + \delta V$. Then $u_U = u_0$. For $\delta \ll 1$

$$\begin{aligned}
&\tilde{F}(U+\delta V)(x_K) - \tilde{F}(U)(x_K) = \\
&= \int_{x_K+\varepsilon I} \left(W\left(x, \frac{x}{\varepsilon}, u_\delta, \nabla u_\delta\right) - W\left(x, \frac{x}{\varepsilon}, u_0, \nabla u_0\right)\right) \mathrm{d}x \\
&= \int_{x_K+\varepsilon I} \frac{\delta W}{\delta u}(u_0)(u_\delta - u_0)\mathrm{d}x + O(\delta^2) \\
&= \int_{x_K+\varepsilon I} \frac{\delta W}{\delta u}(u_0)(u_\delta - u_0 - \delta V)\mathrm{d}x + \delta \int_{x_K+\varepsilon I} \frac{\delta W}{\delta u}(u_0) V \mathrm{d}x + O(\delta^2)\ .
\end{aligned}$$

Since $u_\delta - (u_0 + \delta V)$ is periodic on $x_K + \varepsilon I$, the first term on the right hand side vanishes as a consequence of the optimality condition for u_0. This proves the lemma.

With this, HMM proceeds in the same way as for solving standard single scale nonlinear variational problems using the finite element method.

It is important to note that HMM also allows us to recover microscale information, for example, point-wise approximation to ∇u^ε where u^ε is the exact solution of (18). This is similar to the linear problem.

For extension of this method to general Galerkin formulation, we refer to [15]. For extension to higher order finite element together with error analysis, we refer to [18].

3 Dynamic Problems

3.1 The Parabolic Homogenization Problem

The same basic principle applies to dynamic problems. Consider for example

$$\partial_t u^\varepsilon = \nabla \cdot \left(a\left(x, \frac{x}{\varepsilon}, t\right) \nabla u^\varepsilon \right) \tag{21}$$

Our purpose is to compute locally averaged quantities of u^ε. Since the problem is obviously conservative, we can choose standard finite volume method as the macroscale scheme. For this purpose we work with a macroscale grid with sizes $(\Delta x, \Delta t)$. The data that need to be estimated are the macroscale fluxes.

In order to estimate the macroscale fluxes at cell boundaries, we set up numerical experiments based on the microscale model (21) using a generalized Godunov procedure. Knowing $\{U_j^n\}$, the numerical approximation to the cell averages of u^ε at time $t = t^n$, we compute the approximation to the cell averages of u^ε at the next time step by the following steps.

Step 1. Reconstruction. From $\{U_j^n\}$, construct a piecewise polynomial $U^n(x)$. For example, we can use the piecewise linear reconstruction. In one-dimension, this is

$$RU(x) = U_j + \frac{U_{j+1} - U_j}{\Delta x}(x - x_j)$$

for $x \in [j\Delta x, (j+1)\Delta x]$.

Step 2. At each macroscopic cell boundary $x_{j+\frac{1}{2}}$, solve (21) on $x_{j+\frac{1}{2}} + \varepsilon I$ subject to the boundary condition that $u(x,t) - U^n(x)$ is periodic with period εI, for $n\Delta t < t < \tau + n\Delta t$ for some suitably chosen τ.

Step 3. Let $J^n_{j+\frac{1}{2}}$ be the average of the microscale flux $j^\varepsilon = a(x, \frac{x}{\varepsilon})\nabla u^\varepsilon(x, \tau + n\Delta t)$ over $x_{j+\frac{1}{2}} + \varepsilon I$.

Step 4. Compute $\{U_j^{n+1}\}$ using the finite volume method:

$$\frac{U_j^{n+1} - U_j^n}{\Delta t} + \frac{J^n_{j+\frac{1}{2}} - J^n_{j-\frac{1}{2}}}{\Delta x} = 0 \tag{22}$$

However, an additional issue is present for dynamic problems with regard to time scales. There are two obvious time scales in our problem. The first is the macro time scale of interest. The second is the microscopic relaxation time, which for the present problem is the time needed for the solution to develop the correct microstructure that matches the macroscale behavior. This time scale is estimated to be of $O(\varepsilon^2)$ [7]. This micro time scale is brought into the problem since we are using the microscale model to estimate the macroscale flux. It is obviously a source of inefficiency compared with the situation when we had an explicit macroscale model to use. However, the fact that the two time scales are separated can be exploited to effectively eliminate this source of inefficiency. As an illustration we plot in Fig. 2 a typical behavior of the microscopic flux $j^\varepsilon(x,t) = a\left(x, \frac{x}{\varepsilon}\right)\nabla u^\varepsilon(x,t)$ at a cell boundary over the time interval $[t^n, t^n + \Delta t]$ as a function of the micro time steps. It is quite clear that $j^\varepsilon(x,t)$ quickly settles down (after about 35 micro time steps) to a quasi-stationary value after a rapid transient. We obtain an efficient numerical scheme if we select this value as the macroscopic flux and use that to evolve U over a much larger time step Δt. This is the principle we use to choose the value of τ discussed earlier. For further results, see [3].

Similar ideas have been used in the literature, for example for stiff ODEs [2, 24, 27], for kinetic Monte Carlo simulations with disparate rates [36], for stochastic ODEs with small noise [46] and for multi-scale problems in [23].

Another interesting point for HMM in this application is the computation of J^n once the microscale data is obtained. In general this involves appropriate spatial and temporal averages of the microscale data. For the present problem, temporal averaging is unnecessary since the microscale flux does not fluctuate, as seen in Fig. 22. Next we discuss an example for which the microscale flux does fluctuate.

3.2 The Convection Problem

Consider the advection homogenization problem

$$u_t^\varepsilon + \nabla \cdot \left(a\left(x, \frac{x}{\varepsilon}, t\right) u^\varepsilon\right) = 0 \tag{23}$$

in one-dimension. Let us assume $a(x,y,t) > a_0 > 0$. We can proceed as before for the parabolic problem, except that we may take a piecewise constant reconstruction. In contrast to the previous example, the temporal oscillations in the solutions

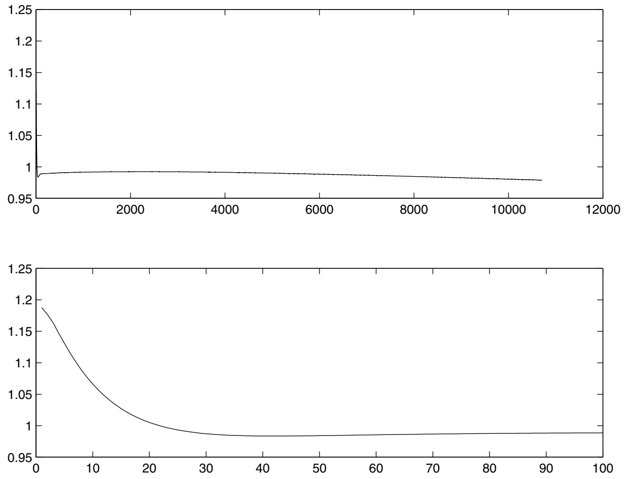


Fig. 2. Computed flux $\tau^{\varepsilon}(x,t) = a\left(x,\frac{x}{\varepsilon}\right)\nabla u^{\varepsilon}(x,t)$ as a function of the micro time step over one typical macro time step, for the parabolic homogenization $a\left(x,\frac{x}{\varepsilon}\right) = 2 + \sin 2\pi \frac{x}{\varepsilon}$. The bottom figure is a detailed view of the top figure for small time steps. Notice that $j^{\varepsilon}(x,t)$ quickly settles down (after about 35 micro time steps) to a quasi-stationary value after a rapid transient

of (23) do not die out. This is reflected in Fig. 3 where we plot the microscopic flux $j^{\varepsilon}(x,t) = a\left(x,\frac{x}{\varepsilon}\right)u^{\varepsilon}(x,t)$ over the time interval $[t^n, t^n + \varDelta t]$ as a function of the microscale time steps. j^{ε} remains oscillatory throughout the time interval. Nevertheless, if we plot the time average

$$\bar{j}(x,t) = \frac{1}{t}\int_{t^n}^{t^n+t} K\left(1-\frac{\tau}{t}\right) j^{\varepsilon}(x,\tau)\mathrm{d}\tau, \qquad K(\tau) = 1 - \cos 2\pi\tau \tag{24}$$

as shown in the bottom of Fig. 3, we see that it settles down to a quasi-stationary value on a time scale of $O(\varepsilon)$. This value is then used in the macroscale finite volume method.

These two dynamic examples illustrate how HMM takes advantage of the time scale separation in a very natural way. More sophisticated applications and convergence theorems are found in [21, 29, 39].

3.3 Hamilton–Jacobi Equation

In this subsection, we consider the homogenization problem for the Hamilton–Jacobi equation

$$u_t + H\left(\frac{x}{\varepsilon}, \nabla u\right) = 0 \tag{25}$$

More general forms of Hamiltonian can be considered. But it is convenient to restrict our attention to (25). Equations of this type can arise in front propagation and control

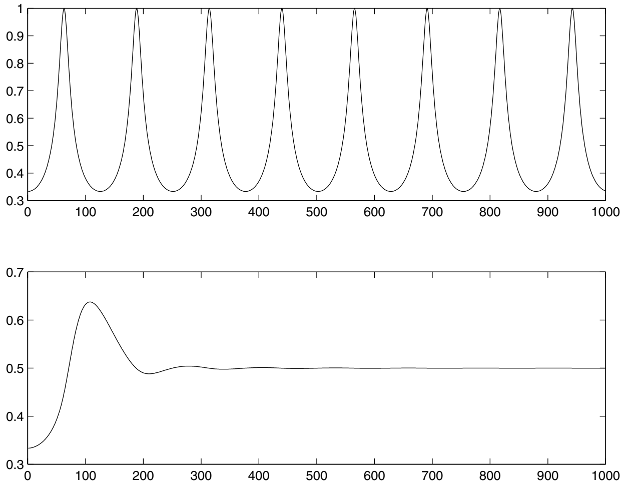


Fig. 3. Top figure: Computed flux $j^{\varepsilon}(x,t) = a\left(x, \frac{x}{\varepsilon}\right) u^{\varepsilon}$ as a function of the micro time step over one macro time step for the convection homogenization problem (23). Bottom figure: Time averaged flux $\bar{j}(x,t)$ as a function of the micro time step

problems in inhomogeneous medium. Details of the work reported here can be found in [9]. See also [10]

Homogenized equation for (25) was derived in the fundamental but unpublished paper of Lions, Papanicolaou and Varadhan [30]. See also related work in [13, 22]. The homogenized equation takes the form

$$U_t + \bar{H}(\nabla U) = 0 \tag{26}$$

The homogenized Hamiltonian $\bar{H}$ has very interesting properties, one of which is flattening in some regions [30]. Computing $\bar{H}(\cdot)$ is not an easy task. Therefore it is worthwhile to design efficient numerical methods that are based on (25) instead of (26).

To discuss the application of HMM to (25), we will first consider the one-dimensional case and explore the connection of (25) with nonlinear conservation laws. Higher dimensional case will be considered later.

In one-dimension, let $v = u_x$, then (25) is equivalent to considering entropic solutions of

$$\frac{\partial v}{\partial t} + \frac{\partial}{\partial x} H\left(\frac{x}{\varepsilon}, v\right) = 0 \tag{27}$$

Again we are interested in computing cell averages of v on a macro grid. For that purposes, we pick the finite volume scheme as our macroscopic scheme.

$$V_j^{n+1} = V_j^n - \frac{\Delta t}{\Delta x}\left(J_{j+\frac{1}{2}}^n - J_{j-\frac{1}{2}}^n\right) \tag{28}$$

where $\{V_j^n\}$ is the approximation to the cell averages of v at macro time t^n.

Next we need to estimate $\{J^n_{j+\frac{1}{2}}\}$. This is done again via a Godunov procedure.

Step 1. From $\{V^n_j\}$, reconstruct a function $v^n(x)$ such that the cell averages of $v^n(x)$ satisfies

$$\frac{1}{|I_j|}\int_{I_j} v^n(x)\mathrm{d}x = V^n_j \tag{29}$$

where $I_j = [x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}]$ is the j-th macro cell, $x_{j+\frac{1}{2}}$ is the location of the cell boundary between I_j and I_{j+1}.

Step 2. Solve for $t \in [n\triangle t, n\triangle t + \alpha\triangle t]$ equation (27) on the interval $I^\varepsilon_j = [x_{j+\frac{1}{2}} - \frac{1}{2}\varepsilon, x_{j+\frac{1}{2}} + \frac{1}{2}\varepsilon]$ subject to the boundary condition that $v(x,t) - v^n(x)$ is periodic on I^ε_j with period ε.

Step 3. Let

$$J^n_{j+\frac{1}{2}}(\alpha) = \frac{1}{\varepsilon}\int_{I^\varepsilon_j} H\left(\frac{x}{\varepsilon}, v(x, n\triangle t + \alpha\triangle t)\right)\mathrm{d}x \tag{30}$$

The efficiency of this algorithm comes from the fact that the microscale problems are solved on intervals of size ε and that $J^n_{j+\frac{1}{2}}(\alpha)$ quickly converges to a stationary value after a few microscale time steps. The rate of convergence is $O\left(\frac{\varepsilon}{\alpha\triangle t}\right)$ if $H(x,p)$ is strictly convex in p [13]. Therefore we can stop the microscale sub-step at $\alpha\triangle t$ for some α such that $\frac{\varepsilon}{\triangle t} \ll \alpha \ll 1$.

On the other hand, even though $J^n_{j+\frac{1}{2}}(\alpha)$ converges to a value that depends only on V^n_j, there is no simple way to recover the microstructural behavior. One reason is that stationary solutions to

$$v_t + H(y,v)_y = 0 \tag{31}$$

with periodic boundary conditions are not necessarily uniquely determined by their average $\bar{v} = \int_0^1 v(y,t)dy$. Given $\bar{v}$, there can be more than one steady state solutions to (31) with average $\bar{v}$, all of which give rise to the same average Hamiltonian $\bar{H} = \int_0^1 H(y, v(y))dy$ (see [13] for a detailed discussion). Consequently knowing the macroscale approximations $\{V^n_j\}$, there is no simple way to construct accurate approximations to the microscale solution, in contrast to the linear problems discussed earlier and in Sect. 4.

The same strategy can be applied to multi-dimensional problems. Even though there is no conservation form in high dimensions, the basic principle can still be used. The main difference is that the needed data, the values of the effective Hamiltonian, are estimated by looking at the linear growth rate of the microscopic solutions in micro time scale. For the macroscale scheme, we can use the ENO-type methods developed in [38]. The ENO-Lax-Friedrichs type of schemes seem to be the best candidate since the amount of macroscale data that need to be estimated seem to be minimized with the ENO-Lax-Friedrichs schemes. The data estimation proceeds in the same way, and we also expect to have the same rate of convergence, i.e. $O(\frac{\varepsilon}{\alpha\Delta t})$ toward the correct macroscale Hamiltonian if the microscale Hamiltonian $H(y,p)$ is strictly convex in p.

4 Stability and Accuracy

The basic principle, established in [15], is that if an associated macroscale scheme, called the Generalized Godunov Scheme (GGS) is stable, then HMM is stable and the error can be decomposed into two parts: the usual discretization error for the GGS and the error in the approximation of the macro data. For simplicity, we will restrict ourselves to the dynamic periodic homogenization problem and we will assume that the microscale problem is solved exactly. Results on more general problems can be found in [32].

Let us write HMM symbolically as

$$U_j^{n+1} = U_j^n + \Delta t F_j^n(U^n) \tag{32}$$

In order to study its stability and accuracy properties we can compare it to a macroscale scheme which is consistent with the effective macroscale equation

$$\bar{U}_j^{n+1} = \bar{U}_j^n + \Delta t \bar{F}_j^n(\bar{U}^n) \tag{33}$$

As long as (33) is stable, it can be shown that [15]

$$\max_j |U_j^n - \bar{U}_j^n| \le C \max_{k \le n, j, U} |F_j^k(U) - \bar{F}_j^k(U)| \tag{34}$$

Here U belongs to certain class of functions, discussed in [15]. In principle we can choose any macroscale scheme that maximizes the stability and accuracy property. In practice, it seems most convenient to choose (33) as the so-called generalized Godunov scheme (GGS), obtained by following the HMM procedure, except that in the data estimation step, we replace the microscale solver by the macroscale solver. We emphasize that the macroscale solver is used here only for the purpose of analysis, not in actual computation.

Let us consider the example of the parabolic homogenization problem (23), and consider the following HMM

$$U_j^{n+1} = U_j^n - \frac{\triangle t}{\triangle x}(J_{j+\frac{1}{2}}^n(U^n) - J_{j-\frac{1}{2}}^n(U^n))$$

where $\{J_{j+\frac{1}{2}}^n\}$ is given by

$$J_{j+\frac{1}{2}}^n(U^n) = j^\varepsilon(x_{j+\frac{1}{2}}, t^n + \alpha\triangle t)$$

where $j^\varepsilon(x,t) = a\left(x, \frac{x}{\varepsilon}\right)\nabla u^\varepsilon(x,t)$, and $u^\varepsilon(x,t)$ is the solution of (23) on $[x_{j+\frac{1}{2}} - \frac{1}{2}\varepsilon, x_{j+\frac{1}{2}} + \frac{1}{2}\varepsilon]$ with the boundary condition that $u(x,t) - (RU^n)(x)$ is periodic with period ε and initial condition

$$u^\varepsilon(x, t^n) = (RU^n)(x)$$

We take R to be the piecewise linear reconstruction

$$RU^n(x) = \frac{1}{2}(U_j^n + U_{j+1}^n) + \frac{U_{j+1}^n - U_j^n}{\triangle x}(x - x_{j+\frac{1}{2}})$$

for $x \in (x_j, x_{j+1}]$. The GGS is then given by

$$\bar{U}_j^{n+1} = \bar{U}_j^n - \frac{\triangle t}{\triangle x}(\bar{J}_{j+\frac{1}{2}}^n(\bar{U}^n) - \bar{J}_{j-\frac{1}{2}}^n(\bar{U}^n))$$

where

$$\bar{J}_{j+\frac{1}{2}}^n(\bar{U}^n) = A(x_{j+\frac{1}{2}})\nabla U_j(x_{j+\frac{1}{2}}, t^n + \alpha\triangle t)$$

Here $A(x)$ is the homogenized coefficient, and $U_j(x,t)$ is the solution of

$$\partial_t U_j = \nabla \cdot (A \nabla U_j)$$

with initial condition

$$U_j(x, t^n) = (R\bar{U}^n)(x),$$

for $x \in [x_{j+\frac{1}{2}} - \frac{1}{2}\varepsilon, x_{j+\frac{1}{2}} + \frac{1}{2}\varepsilon]$, and the boundary condition that $U_j(x,t) - R\bar{U}^n(x)$ is periodic with period ε. To study the stability of GGS, without loss of generality we may assume that $A(x)$ is a constant.

Lemma 2. *The GGS is stable if*

$$A\frac{\triangle t}{(\triangle x)^2} < \frac{1}{2}$$

Proof. It is easy to check that $U_j(x,t) = R\bar{U}^n(x)$. Hence the GGS is nothing but the classical finite difference scheme for the heat equation

$$\bar{U}_j^{n+1} = \bar{U}_j^n - A\frac{\triangle t}{(\triangle x)^2}(\bar{U}_{j+1}^n - 2\bar{U}_j^n + \bar{U}_{j-1}^n)$$

The lemma then follows.

To study the accuracy of HMM, let

$$F_j^n(U) = \frac{1}{\triangle x}(J_{j+\frac{1}{2}}^n(U) - J_{j-\frac{1}{2}}^n U))$$
$$\bar{F}_j^n(U) = \frac{1}{\triangle x}(\bar{J}_{j+\frac{1}{2}}^n(U) - \bar{J}_{j-\frac{1}{2}}^n(U))$$

We would like to estimate $J_{j+\frac{1}{2}}^n(U) - \bar{J}_{j+\frac{1}{2}}^n(U)$. For simplicity, we take $a\left(x, \frac{x}{\varepsilon}\right) = a(\frac{x}{\varepsilon})$. The microscale problem

$$\begin{cases} u_t^\varepsilon(x,t) = \nabla \cdot \left(a(\frac{x}{\varepsilon})\nabla u^\varepsilon(x,t)\right) \\ u^\varepsilon(x,0) = RU(x), x \in (x_{j+\frac{1}{2}} - \frac{1}{2}\varepsilon, x_{j+\frac{1}{2}} + \frac{1}{2}\varepsilon) \\ u^\varepsilon(x,t) - RU(x) \text{ is periodic with period } \varepsilon \end{cases}$$

relaxes to the “local equilibrium state”

$$\tilde{u}^\varepsilon(x) = RU(x) + \varepsilon u_1(x)$$
$$\varepsilon \partial_x u_1(x) = s_j \left(\frac{A}{a\left(\frac{x}{\varepsilon}\right)} - 1 \right)$$

where $s_j = \frac{U_{j+1} - U_j}{\triangle x}$, $A = (\int_0^1 \frac{1}{a(y)} dy)^{-1}$. The relaxation time is of order ε^2.

$$|u^\varepsilon(x,t) - \tilde{u}^\varepsilon(x)| \le \|RU\| \left(C_1 \mathrm{e}^{-\lambda \frac{t}{\varepsilon^2}} + C_2 \varepsilon \right)$$
$$|u_x^\varepsilon(x,t) - \tilde{u}_x^\varepsilon(x)| \le \|RU\| \left(C_1 \mathrm{e}^{-\lambda \frac{t}{\varepsilon^2}} + C_2 \varepsilon \right)$$

where $\|RU\| = |s_j| + |U_j| + |U_{j+1}|$, λ is some positive constant. Hence we have

$$\begin{aligned} |J_{j+\frac{1}{2}}^n(U) - \bar{J}_{j+\frac{1}{2}}^n(U)| &= \left| a\left(\frac{x_{j+\frac{1}{2}}}{\varepsilon}\right) u_x^\varepsilon(x_{j+\frac{1}{2}}, \alpha \triangle t) - A \frac{U_{j+1} - U_j}{\triangle x} \right| \\ &\le a(\frac{x_{j+\frac{1}{2}}}{\varepsilon}) |u_x^\varepsilon(x, \alpha\triangle t) - \tilde{u}_x^\varepsilon(x)| \\ &\le \|RU\| (C_1 \mathrm{e}^{-\lambda \frac{\alpha \triangle t}{\varepsilon^2}} + C_2 \varepsilon). \end{aligned}$$

Using this and the general stability theorem for HMM we obtain

$$|U_j^n - \bar{U}_j^n| \le \frac{C}{\triangle x} \left(C_1 \mathrm{e}^{-\lambda \frac{\alpha \triangle t}{\varepsilon^2}} + C_2 \varepsilon \right).$$

Here we have used the estimate that

$$\|RU\| \le \frac{C}{\Delta x}$$

if U is bounded.

Next we consider an example of the advection problem

$$u_t^\varepsilon + a\left(\frac{x}{\varepsilon}\right) u_x^\varepsilon = 0 \tag{35}$$

We will use the following version of HMM. For the macroscale scheme, we will simply use the forward Euler method, i.e. we will think of the macroscale model in the form

$$U_t = F(U)$$

and use

$$U_j^{n+1} = U_j^n + \triangle t F_j^n(U^n)$$

To estimate $F_j^n(U^n)$, we take a piecewise linear reconstruction

$$RU^n(x) = U_j^n + \frac{U_{j+1}^n - U_{j-1}^n}{2\triangle x}(x - x_j)$$

for $x \in [x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}]$. Let $u^\varepsilon(x,t)$ be the solution of (35) with initial condition $u^\varepsilon(x,t^n) = RU^n(x)$, for $x \in [x_j - \frac{1}{2}\varepsilon, x_j + \frac{1}{2}\varepsilon]$ and the boundary condition that $u^\varepsilon(x,t) - RU^n(x)$ is periodic with period ε. Let $f(t) = a\left(\frac{x_j}{\varepsilon}\right) u^\varepsilon_x(x_j, t^n + t)$. We will use

$$F^n_j(U^n) = \frac{1}{\tau}\int_0^\tau \varphi\left(\frac{\tau - s}{\tau}\right) f(s)\mathrm{d}s$$

where φ is a suitably chosen filter, $\tau \leq \triangle t$ is a suitably chosen time duration.

Even though this may seem like a perfectly reasonable numerical procedure. It suffers from stability problems. This can be seen from the stability of the corresponding GGS, which is given by

$$\bar{U}^{n+1}_j = \bar{U}^n_j + \triangle t \bar{F}^n_j(\bar{U}^n)$$

where

$$\bar{F}^n_j(\bar{U}^n) = \frac{1}{\tau}\int_0^\tau \varphi\left(\frac{\tau - s}{\tau}\right) \bar{F}(s)\mathrm{d}s$$
$$\bar{F}(s) = A\partial_x U_j(x_j, t^n + s)$$

Here U_j is the solution of

$$\partial_t U_j + A\partial_x U_j = 0$$

with initial condition

$$U_j(x,t^n) = R\bar{U}^n(x)$$

for $x \in [x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}]$, and boundary condition that $U_j(x,t) - R\bar{U}^n(x)$ in periodic with period ε.

We will assume that $a > 0$. Hence $A > 0$. It is easy to see that this is nothing but

$$\bar{U}^{n+1}_j = \bar{U}^n_j - A\frac{\triangle t}{\triangle x}(\bar{U}^n_{j+1} - \bar{U}^n_{j-1})$$

which is the notorious Richardson's scheme. This scheme is stable only if $\triangle t = O(\triangle x^2)$ which is too restrictive for an advection equation.

To fix this, we modify the reconstruction to

$$RU^n(x) = U^n_j + \frac{U^n_j - U^n_{j-1}}{\triangle x}(x - x_j)$$

The GGS then changes to the upwind scheme

$$\bar{U}^{n+1}_j = \bar{U}^n_j - A\frac{\triangle t}{\triangle x}(\bar{U}^n_j - \bar{U}^n_{j-1}) \tag{36}$$

which is stable if

$$A\frac{\triangle t}{\triangle x} \leq 1.$$

To analyze the accuracy of the HMM, we estimate $F^n_j(U) - \bar{F}^n_j(U)$. Observe that

Lemma 3. *Assume that φ satisfies $\varphi(0) = \varphi(1) = 0$, $\int_0^1 |\varphi_x| \mathrm{d}x < +\infty$, and let f be a continuous periodic function with period 1. Then*

$$\left| \frac{1}{\tau} \int_0^\tau \varphi\left(1 - \frac{s}{\tau}\right) f\left(\frac{s}{\varepsilon}\right) \mathrm{d}s - \bar{f} \right| \leq C \frac{\varepsilon}{\tau}$$

where $\bar{f} = \int_0^1 f(s) \mathrm{d}s$ is the average of f.

Proof. This is a classical result.

Lemma 4. *The solution to the microscale problem described above satisfies*

$$u^\varepsilon(x,t) = Bt + s_j x + v\left(\frac{x}{\varepsilon}, \frac{t}{\varepsilon}\right)$$

where v is a space-time periodic function, $s_j = \frac{U_j^n - U_{j-1}^n}{\triangle x}$ and B is some constant. Moreover

$$f^\varepsilon(t) = a\left(\frac{x_j}{\varepsilon}\right) \partial_x u^\varepsilon(x_j, t)$$

is periodic in t with period $T_\varepsilon = O(\varepsilon)$. $\bar{f}^\varepsilon = As_j$.

Proof. This is a trivial consequence of the observation that the microscale problem is equivalent to solving (35) over the whole space with initial data

$$u^\varepsilon(x, t^n) = RU^n(x) = U_j^n + s_j(x - x_j) \; .$$

Combining these two lemmas, we arrive at

Lemma 5. *Assume that φ satisfies the condition in Lemma 3. Then*

$$|F_j^n(U) - \bar{F}_j^n(U)| \leq C \max_j |s_j| \frac{\varepsilon}{\tau} \; .$$

From this, we obtain

Lemma 6.

$$|U_j^n - \bar{U}_j^n| \leq C \max_j |s_j| \frac{\varepsilon}{\tau} \; .$$

5 How Can HMM Fail?

Even though HMM is a very general tool, it also has some pitfalls, just as any other advanced numerical techniques. Understanding the limitations of HMM is important for further improving the methodology. Here we list some ways by which HMM fail to approximate the right quantities.

While detailed knowledge of the macro model is not necessary for the success of HMM, some information about the macroscale model is crucial. Without them, a blind application of HMM can give wrong results. We will illustrate this with some examples.

The *first* example concerns the averaging operator we use in estimating the forces/fluxes from the microscale data. Consider the HMM discussed in Sect. 3.1. If, for the flux estimator, we take

$$J^n_{j+\frac{1}{2}} = a^\varepsilon(x_{j+\frac{1}{2}})\nabla u^\varepsilon(x_{j+\frac{1}{2}}, n\triangle t + \alpha\triangle t), \tag{37}$$

where α is chosen so that the right hand side of (37) has reached a quasi-stationary value. At a first sight, this may seem like a perfectly reasonable method, and it can be shown that it does approximate the right quantity for one-dimensional problems. But a closer inspection reveals that this method fails in higher dimensions. The reason is because that in higher dimensions, $a^\varepsilon(x)\nabla u^\varepsilon(x,t) = j^\varepsilon(x,t)$ is in general an oscillatory quantity as a function of x. Therefore $\frac{J^n_{j+\frac{1}{2}} - J^n_{j-\frac{1}{2}}}{\triangle x}$ computed using (37) is a poor approximation to J_x. Explicit examples were discussed in [15].

The reason that HMM fails in this case is that we chose a poor averaging operator, namely the evaluation operator, which does not smooth out the small scale spatial oscillations in $j^\varepsilon(x,t) = a^\varepsilon(x)\nabla u^\varepsilon(x,t)$. This is easily fixed by spatially averaging the microscale fluxes over the cell I^ε_j and use the averaged values as the macroscale data. For more discussions on this issue, see [3].

Our *second* example concerns the a priori knowledge of the macroscale model. A crucial advantage of HMM, as we emphasized throughout this paper, is that it does not need explicit knowledge of the macroscale model. But some knowledge about the macroscale model is important, as we now illustrate. Consider the problem

$$u_t + \nabla \cdot (b\left(\frac{x}{\varepsilon}\right) u) = \nabla \cdot (a\left(\frac{x}{\varepsilon}\right)\nabla u)\ , \tag{38}$$

where b and a are smooth periodic functions with period 1. The homogenized equation for (38) is given by

$$U_t + \nabla \cdot (\bar{b}U) = \nabla \cdot (A\nabla U)\ , \tag{39}$$

where A is the homogenized coefficient matrix for the case when $b = 0$, $\bar{b}$ is the average of b over its period. Without knowing that the large scale model is a second order differential equation and the macroscale flux depends on the derivative of the macroscale variable U, we might use HMM as in Sect. 3 but with piecewise constant reconstruction. In this case, the estimated flux will only contain the part that approximates $\bar{b}U$, not the part that approximates $A\nabla U$. This results in an $O(1)$ error for the macroscale quantities.

The failure in this case is due to the reconstruction and the associated constraints we put on the microscale solver, which does not allow HMM to probe macroscale fluxes caused by ∇U. This can of course be fixed by changing the piecewise constant reconstruction to piecewise linear reconstruction.

In general, even though we do not need to know the detailed macroscale model, we do need to have an idea about the order of the effective macroscale equation so that we can design a reconstruction operator that probes all relevant forces and/or fluxes. It is often possible to determine either analytically or by numerical experiments on the microscale model the correct order of the homogenized operator.

Our *third* example is the advection of a passive scaler by an oscillatory shear flow

$$u_t + \partial_{x_1}\left(a\left(\frac{x_2}{\varepsilon}\right)u\right) = 0. \tag{40}$$

The homogenized equation for this problem is

$$U_t + \bar{a}\partial_{x_1}U = \partial_{x_1}^2 \int_{-\alpha}^{\alpha} \mathrm{d}\omega(\gamma) \int_0^t \mathrm{d}sU(x_1 + \gamma(t-s), x_2, s) \tag{41}$$

[44] and it is memory dependent. Here ω is some kernel that depends on a, $\bar{a}$ is the average of a over its period.

HMM can be straightforwardly applied to (40). For example we can take the macroscale scheme to be the finite volume method. For the micro sub-step, we can use piecewise constant reconstruction. The fluxes at the cell boundaries do not change in time if we assume that the microscale model is solved exactly. Therefore HMM would suggest stopping the microscale evolution at arbitrarily small times and use the fluxes to advance with macro time-steps. It is easy to check that in this way the computed macroscale behavior approximates

$$U_t + \bar{a}\partial_{x_1}U = 0$$

which is clearly incorrect.

The failure in this case comes from the fact that even though the fluxes across the cell boundaries are constant in time, the solution to the microscale problem does not reach local equilibrium.

In the framework of the general stability and accuracy theory discussed in the last subsection, in the first example the failure is due to the large difference between F_j^k and $\bar{F}_j^k$. In the second and last example, the failure is due to the inconsistency between GGS and the correct macroscale model.

6 Conclusion

In summary, we have seen that HMM does provide an ideal framework for the numerical computation of a large variety of homogenization problems. In most cases, HMM not only gives information on the macroscale properties, but can also be used to extract microstructural information. The general HMM convergence theory provides guidance for rational choice of macro- and micro- algorithms.

Further work in this direction includes exploring the application of HMM-based ideas for problems that are strongly inhomogeneous. These problems occur in a variety of applications such as transport through a porous medium that contains cracks of different scales.

Acknowledgement

We thank Pingbing Ming, Olof Runborg, Xingye Yue and Li-Tien Cheng for helpful discussions. The work of E was partially supported by ONR through grant N00014-01-1-0674. The work of Engquist was partially supported by NSF through grant DMS-9973341.

References

1. G. Allaire, "Homogenization and two-scale convergence". *SIAM J. Math. Anal.* 23 (1992), no. 6, 1482–1518.
2. A. Abdulle, "Fourth order Chebychev methods with recurrence relations", *SIAM J. Sci. Comput.*, vol 23, pp. 2041-2054, 2002.
3. A. Abdulle and W. E, "Finite difference HMM for homogenization problems", *J. Comput. Phys.*, 191, 18-39 (2003).
4. I. Babuska, "Homogenization and its applications", *SYNSPADE 1975*, B Hubbard ed. pp. 89-116.
5. I. Babuska, "Solution of interface problems by homogenization", I: *SIAM J. Math. Anal.*, 7 (1976), no. 5, pp. 603-634. II: *SIAM J. Math. Anal.*, 7 (1976), no. 5, pp. 635-645. III: *SIAM J. Math. Anal.*, 8 (1977), no. 6, pp. 923-937.
6. I. Babuska, G. Caloz anf J. Osborn, "Special Finite Element Methods for a Class of Second Order Elliptic Problems with Rough Coefficients", *SIAM J. Numer. Anal.*, vol. 31, pp. 945-981, 1994.
7. A. Benssousan, J.L. Lions and G. Papanicolaou, *"Asymptotic Analysis of Periodic Structures,"* North-Holland (1978).
8. F. Brezzi, D. Marini, and E. Suli, "Residual-free bubbles for advection-diffusion problems: the general error analysis." *Numerische Mathematik.* 85 (2000) 1, 31-47.
9. L. T. Cheng and W. E, "HMM for interface dynamics", in *Contemporary Mathematics: A special volume in honor of Stan Osher*, S. Y. Cheng, C. W. Shu and T. Tang eds.
10. L. T. Cheng and W. E, "HMM for Hamilton-Jacobi equations", in preparation.
11. P. G. Ciarlet, *"The Finite Element Methods for Elliptic Problems"*, Amsterdam ; New York : North-Holland Pub. Co., 1978.
12. L. Durlofsky, "Numerical calculation of equivalent grid block permeability tensors for heterogeneous porous media", *Water Resour. Res.*, 27, 699-708, 1991.
13. W. E, "Homogenization of scalar conservation laws with oscillatory forcing terms", *SIAM J. Appl. Math.*, vol. 52, pp. 959-972, 1992.
14. W. E, "Homogenization of linear and nonlinear transport equations", *Comm. Pure and Appl.*, vol. XLV, 301-326, 1992.
15. W. E and B. Engquist, "The heterogeneous multi-scale methods", *Comm. Math. Sci.* 1, 87-133 (2003).
16. W. E, B. Engquist and Z. Huang, *Phys. Rev. B*, 67 (9), 092101 (2003).
17. W. E, D. Liu and E. Vanden-Eijnden, "Analysis of Multiscale Methods for Stochastic Differential Equations," submitted to *Comm. Pure Appl. Math.*, 2003.
18. W. E, P. Ming and P. W. Zhang, "Analysis of the heterogeneous multi-scale method for elliptic homogenization problems", *J. Amer. Math. Soc.* vol 18, pp 121-156, 2005.
19. B. Engquist, "Computation of oscillatory solutions to hyperbolic differential equations", *Springer Lecture Notes in Mathematics*, 1270, 10-22 (1987).

20. B. Engquist and O. Runborg, “Wavelet-based numerical homogenization with applications”, *Lecture Notes in Computational Science and Engineering*, T.J. Barth et.al eds., Springer, 2002.
21. B. Engquist and R. Tsai, “HMM for stiff ODEs”, *Math. Comp.*, in press.
22. L. C. Evans, “The perturbed test function method for viscosity solutions of nonlinear PDE”, *Proc. Royal Soc. Edinburgh* Sect. A, 111 (1989), pp. 359-375.
23. C. W. Gear and I. G. Kevrekidis, “Projective methods for stiff differential equations: problems with gaps in their eigenvalue spectrum,” *SIAM J. Sci. Comp.*, vol 24, pp 1091-1106, 2003.
24. A. Gulliou and B. Lago, Domaine de stabilité associé aux formules d’intégration numérique d’équations différentielles, à pas séparés et à pas liés. *ler Congr. Assoc. Fran. Calcul, AFCAL, Grenoble*, pp. 43–56, Sept. 1960.
25. T. Hou and X. Wu, “A multiscale finite element method for elliptic problems in composite materials and porous media”, *J. Comput. Phys.*, 134(1), 169-189, 1997.
26. T. J. R. Hughes, “Multiscale phenomena: Green’s functions, the Dirichlet to Neumann formulation, subgrid, scale models, bubbles and the origin of stabilized methods”, *Comput. Methods Appl. Mech. Engrg.*, 127, 387-401 (1995).
27. V. I. Lebedev and S. I. Finogenov, “Explicit methods of second order for the solution of stiff systems of ordinary differential equations”, *Zh. Vychisl. Mat. Mat Fiziki*, vol. 16, No. 4, pp. 895-910, 1976.
28. R. LeVeque, *“Numerical Methods for Conservation Laws,”* Birkhäuser, 1990.
29. X. T. Li and W. E, “Multiscale modeling of the dynamics of solids at finite temperature”, submitted to *J. Mech. Phys. Solids*.
30. P. L. Lions, G. Papanicolaou and S. R. S. Varadhan, “Homogenization of Hamilton-Jacobi equations”, unpublished.
31. W. K. Liu, Y. F. Zhang and M. R. Ramirez, “Multiple Scale Finite Element Methods”, *International Journal for Numerical Methods in Engineering*, vol. 32, pp. 969-990, 1991.
32. P. B. Ming, “Analysis of multiscale methods”, in preparation.
33. P. B. Ming and X. Y. Yue, “Numerical Methods for Multiscale Elliptic Problems,” preprint, 2003.
34. S. Mueller, “Homogenization of nonconvex integral functionals and cellular materials”, *Arch. Rat. Anal. Mech.* vol. 99 (1987), 189-212.
35. G. Nguetseng, “A general convergence result for a functional related to the theory of homogenization”. *SIAM J. Math. Anal.* 20 (1989), no. 3, 608–623.
36. M. A. Novotny, “A tutorial on advanced dynamic Monte Carlo methods for systems with discrete state spaces”, *Ann. Rev. Comput. Phys.*, pp. 153-210 (2001).
37. J. T. Oden and K. S. Vemaganti, “Estimation of local modelling error and global-oriented adaptive modeling of heterogeneous materials: error estimates and adaptive algorithms”, *J. Comput. Phys.*, 164, 22-47 (2000).
38. S. Osher and C. W. Shu, “High order essentially non-oscillatory schemes for Hamilton-Jacobi equations”, *SIAM J. Numer. Anal.*, vol. 28, pp. 907-922 (1991).
39. W. Ren and W. E, “HMM for the modeling of complex fluids and microfluidics”, *J. Comput. Phys.*, to appear.
40. G. Samaey, D. Roose and I. G. Kevrekidis. “Combining the Gap-Tooth Scheme with Projective Integration: Patch Dynamics", this volume.
41. C. Schwab, “Two-scale FEM for homogenization problems”, *Proceedings of the Conference “Mathematical Modelling and Numerical Simulation in Continuum Mechanics”*, Yamaguchi, Japan, I. Babuska, P. G. Ciarlet and T. Myoshi (Eds.), *Lecture Notes in Computational Science and Engineering*, Springer Verlag 2002.

42. C. Schwab and A.-M. Matache, “Generalzied FEM for homogenization problems”, *Lecture Notes in Computational Science and Engineering*, T.J. Barth et.al eds., Springer, 2002.
43. E. B. Tadmor, M. Ortiz and R. Phillips, “Quasicontinuum analysis of defects in crystals,” *Phil. Mag.*, A73 , 1529–1563 (1996).
44. L. Tartar, “Solutions oscillantes des équations de Carleman”. *Goulaouic-Meyer-Schwartz Seminar*, 1980–1981, Exp. No. XII, 15 pp., École Polytech., Palaiseau, 1981.
45. S. Torquato, *“Random Heterogeneous Materials: Microstructure and Macroscopic Properties”*, Springer-Verlag, 2001.
46. E. Vanden-Eijnden, “Numerical techniques for multiscale dynamical systems with stochastic effects”, *Comm. Math. Sci.*, 1, 385-391 (2003).
47. K. Xu and K. H. Prendergast, “Numerical Navier-Stokes solutions from gas kinetic theory,” *J. Comput. Phys.*, 114, 9–17 (1994).

A Coarsening Multigrid Method for Flow in Heterogeneous Porous Media

Jens Eberhard and Gabriel Wittum

Simulation in Technology, University of Heidelberg, Im Neuenheimer Feld 368, D-69120
Heidelberg, Germany
eberhard@uni-hd.de, wittum@uni-hd.de

Summary. This paper focuses on multigrid methods for flow in heterogeneous porous media. We consider Darcy flow and the local permeability $K(x)$ being a stationary random field of lognormal distribution. We apply the recently developed coarse graining method for the numerical upscaling of permeability, and develop a new multigrid method which applies this technique to obtain the coarse grid operators. The coarse grid operators are adjusted to the scale-dependent behaviour of the system as it incorporates only fluctuations of K on larger scales. This kind of action is essential for an efficient interplay with simple smoothers. We investigate important properties of the new multigrid method such as dependence on the boundary conditions and on grid refinement for the coarse graining and dependence on the mesh size. We compare the resulting method with the algebraic method of Ruge and Stüben, a Schur-complement method, and matrix-dependent multigrid methods by solving the flow equation with K being random realizations as well as periodic media. The numerical convergence rates show that the new method is as efficient as the algebraic methods for variances $\sigma_f^2 \leq 3$ of K.

Key words: multigrid method, porous media, upscaling, heterogeneity

1 Introduction

Multigrid methods are among the fastest solvers for large systems of linear equations which come from discretization of partial differential equations. They belong to the class of iterative solvers. They are based on the idea to handle long wave and short wave error components of an approximation by two different ways, namely error smoothing for the high frequency components and coarse grid correction for the low frequency components. This procedure is recursively applied to the error on the coarser grids until the resulting equations can be solved directly. Hence, one obtains an iterative solver which usually reduces the total error spectrum equally. It is essential that in many cases the convergence of multigrid methods does not depend on the number of unknowns in contrast to classical iterative solvers such as Gauss-Seidel or SOR smoothers. A detailed introduction to multigrid methods can be found e.g. in the textbooks [8, 19].

In this paper we consider multigrid methods for solving the flow equation in heterogeneous porous media, as found e.g. in groundwater aquifers. Due to the heterogeneity the local permeability varies over many length scales and exhibits large jumps. For solving this type of problems the suitable choice of the multigrid components is important to get an efficient method. This was already pointed out by Brandt in [3]. A possible remedy is to determine coarse grid operators which represent sufficiently well the scale-dependent behaviour of the system on the coarser grids. If the operators are given by discretizations on these grids one has to use informations of the scale dependency of the problem to construct appropriate operators. Such informations can be incorporated by upscaling the permeability in the case of the flow equation.

Another possibility is to apply matrix-dependent transfers to obtain appropriate coarse grid operators, see e.g. [1, 5, 21]. The enhancement of this notion led to the algebraic multigrid methods (AMG), for instance the method of Ruge and Stüben [16], which show an improvement of convergence and robustness, see [17]. Algebraic methods try to ensure an efficient interplay between smoother and coarse grid correction by an appropriate construction of coarser grids. The approach normally avoids to rely on geometric information and extracts the information for grid constructions and matrix-dependent transfers out of the given matrix. This results in so-called "black-box" solvers [5]. There are so-called algebraically defined or element-based algebraic methods as well, see e.g. [4, 12, 10]. These methods involve some geometric information about the elements of the generated grids or apply a fixed hierarchy of grids.

We present a new multigrid method for solving flow in porous media which is based on upscaling of the permeability. We apply a numerical upscaling called coarse graining to determine the coarse grid operators [2, 6, 7]. The upscaling yields a scale-dependent permeability field which includes only fluctuations varying on length scales larger than a given scale. Taking the upscaled field as the basis for the computation of the coarse grid operator, the latter corrects large-scale error components which cannot be reduced by smoothing. We also develop new grid transfers which take advantage of the numerical coarse graining method. Compared to algebraic methods and methods which apply matrix-dependent transfers the new method works very efficiently for permeability fields where the variances are not greater than three.

The structure of the paper is the following. The next section introduces the flow equation which is the basis for our computations. Further, we describe the multigrid algorithm and the model problem. Section 3 is devoted to the coarse graining method and the numerical upscaling. In Sect. 4 we introduce the new method based on the numerical coarse graining, which is called Coarsening multigrid. In Sect. 5, we compare various multigrid methods for solving the model problem. We conclude with a summary.

2 Mathematical Statement and Definitions

2.1 Flow Equation

We consider the flow of an incompressible fluid in a porous medium that is governed by $q(x) = -K(x)\nabla u(x)$ (Darcy's law) and $\nabla \cdot q = \varrho$ (continuity) for $x \in \Omega$ where q is Darcy's velocity, ϱ is a source or sink term, u is the piezometric head, and Ω is the flow domain. The expressions lead to the flow equation

$$-\nabla \cdot K(x)\nabla u(x) = \varrho \quad \text{for } x \in \Omega \tag{1}$$

where the local permeability $K(x)$ is a symmetric, positive definite matrix. We consider K as a stationary random field of lognormal distribution. So, the medium is of random and stationary $f(x) := \ln K(x)$, of normal f with mean $\overline{f} = \overline{f(x)} = \ln K_g$, variance σ_f^2, and two-point covariance

$$\operatorname{cov}\big(f(x), f(x')\big) := \overline{\big(f(x) - \overline{f(x)}\big)\big(f(x') - \overline{f(x')}\big)},$$

which depends on the distance vector: $\operatorname{cov}\big(f(x), f(x')\big) =: w(x-x')$ for $x, x' \in \mathbb{R}^d$. The over-bar $\overline{(\cdot)}$ denotes the ensemble average, and K_g denotes the geometric mean.

Analogously, we split $K(x)$ into its mean and the fluctuations: $K(x) = \overline{K} + k(x)$, with $\overline{k(x)} \equiv 0$. For the correlation function we choose

$$w(x - x') := \sigma_f^2 \exp\left(-\sum_{i=1}^{d} \frac{(x_i - x_i')^2}{2l_i^2}\right),$$

with correlation lengths l_i, $i = 1, \ldots, d$, in the direction of x_i and variance σ_f^2 of $f(x)$. For an isotropic-correlated field the lengths l_i are equally denoted by l_0. In that case, the correlation function merely depends on the distance $|x - x'|$. Exploiting the statistical properties of $\ln K(x)$ one derives for the mean and the variance σ_K^2 of $K(x)$: $\overline{K(x)} = \overline{K} = K_g \exp\big(\sigma_f^2/2\big)$ and $\sigma_K^2 := \overline{k(x)\,k(x)} = K_g^2\big(\exp(2\sigma_f^2) - \exp(\sigma_f^2)\big)$.

For the generation of realizations of the random field $f(x)$ we use a numerical spectral method which is based on a superposition of many randomly chosen cosine modes, see [14, 6]. For the numerical experiments in Sect. 5 we always choose $\overline{f} = 0$.

2.2 Discretization

The flow equation describes a second-order elliptic boundary value problem. We discretize it by a bilinear finite element method, see e.g. [9]. We assume $\Omega \subset \mathbb{R}^2$ to be an open square and consider (1) with boundary conditions $u(x) = u_D(x)$ for $x \in \partial\Omega_D$ and $n(x) \cdot \big(K(x)\nabla u(x)\big) = 0$ for $x \in \partial\Omega_N$, where $\partial\Omega_D \cup \partial\Omega_N = \partial\Omega$. $n(x)$ denotes the outer normal unit vector on $\partial\Omega$. Further we assume $K(x) \in \big(W^{1,\infty}(\Omega)\big)^{2\times 2}$. The variational formulation of (1) reads:

Problem 1. For $\varrho \in L^2(\Omega)$ and $u_D \in H^1(\Omega)$ we seek $u \in H^1(\Omega)$ so that

$$a(u,w) = l(w) \quad \text{for all } w \in H^1_D(\Omega)\,, \quad u|_{\partial\Omega_D} = u_D|_{\partial\Omega_D}\,,$$

where $H^1_D(\Omega) := \{w \in H^1(\Omega)|\; w = 0 \text{ on } \partial\Omega_D\}$, $a(u,w) := \int_\Omega \nabla w \cdot (K\,\nabla u)\mathrm{d}^2x$, and $l(w) := \int_\Omega \varrho\, w\mathrm{d}^2x$.

Defining and substituting $v := u - u_D \in H^1_D(\Omega)$ and $l(w) \leftarrow l(w) + a(u_D, w)$ the variational formulation is given by:

Problem 2. Seek $v \in V := H^1_D(\Omega)$ such that $a(v,w) = l(w)$ for all $w \in H^1_D(\Omega)$.

Replacing the Sobolev function space V by a finite element subspace $V_h \subset V$ we end up with a system of linear equations.

For the discretization we consider structured grids of quadratic elements. We define the partition $\tau_i := \tau_{h_i} = \big\{T^{(i)}_j\big\}$ of Ω which consists of the smallest quadratic elements $T^{(i)}_j$ of length $h_i > 0$ given by the grid

$$\Omega_{h_i} := \big\{(x_1,x_2) \in \overline{\Omega}\,|\, x_1 = lh_i, x_2 = mh_i,\; l,m \in \mathbb{Z}\big\} \tag{2}$$

with mesh size h_i. The midpoint of an element T_j is denoted by $\tilde{x}(T_j)$. We assume that the boundary $\partial\Omega_D$ is resolved by elements $T^{(0)}_i$ of τ_0, i.e. $\partial\Omega_D = \bigcup_I \big(T^{(0)}_i \cap \partial\Omega\big)$ with an appropriate index set I. Further we define the subspace $V_i := V_{h_i} \subset V$ based on the partition τ_i by

$$V_{h_i} := \big\{v \in C\big(\overline{\Omega}\big)|\; v|_{T^{(i)}_j} \in Q^{(1)}\,\forall j,\; v|_{\partial\Omega_D} = 0\big\}$$

where $Q^{(t)} := \big\{u(x_1,x_2) = \sum_{0\le i,k\le t} c_{ik}\, x_1^i x_2^k\big\}$ is the set of polynomials. For the implementation we choose the standard basis $\big\{\psi^{(i)}_j\big\}_{1\le j\le N_i}$ of V_i which is given on the grid Ω_{h_i} by $\psi^{(i)}_j\big(x^{(k)}\big) = \delta_{jk}$.

The linear Problem 2 in V_0 is then equivalent to the system of linear equations $A\,v = b$ where $A := \big(a\big(\psi^{(0)}_j, \psi^{(0)}_i\big)\big)_{i,j=1}^{N_0}$ and $b := \big(l\big(\psi^{(0)}_i\big)\big)_{i=1}^{N_0}$. The solution is given by the vector $v = (v_i)_{i=1}^{N_0}$ due to $v_0(x) = \sum_{i=1}^{N_0} v_i\, \psi^{(0)}_i(x)$. The stiffness matrix A and the right-hand side b explicitly read

$$\begin{aligned} A_{ij} &= a\big(\psi^{(0)}_j, \psi^{(0)}_i\big) = \int_\Omega \nabla\psi^{(0)}_i \cdot \big(K\,\nabla\psi^{(0)}_j\big)\,\mathrm{d}^2x\,, \\ b_i &= l\big(\psi^{(0)}_i\big) = \int_\Omega \varrho\,\psi^{(0)}_i\,\mathrm{d}^2x + \int_\Omega \nabla\psi^{(0)}_i \cdot (K\,\nabla u_D)\,\mathrm{d}^2x\,. \end{aligned}$$

2.3 Multigrid Methods

We briefly state the algorithm of a multigrid method for the notations. One step to improve an approximation for v_0 for solving the system of equations $A_0\,v_0 = b_0$, where the components are given by $A_{ij} = \big(A_0\,\psi^{(0)}_j\big)\big(\psi^{(0)}_i\big)$ and $b_i = b_0\big(\psi^{(0)}_i\big)$, reads

Algorithm 1 (Multigrid cycle $MGC(k, v_k, b_k)$).

```
If $k = J$ return $v_J := A_J^{-1} b_J$.
Else Pre-Smoothing: $v_k := S_k^{\nu_1}(v_k, b_k)$
     Defect: $d_k := A_k\, v_k - b_k$
     Restriction: $d_{k+1} := r_k^{k+1} d_k$
     Coarse grid correction $e_{k+1} := 0$: Loop $1, \ldots, \gamma$:
          $e_{k+1} := MGC(k+1, e_{k+1}, d_{k+1})$
     Prolongation: $v_k := v_k - p_{k+1}^k e_{k+1}$
     Post-Smoothing: $v_k := S_k^{\nu_2}(v_k, b_k)$
     Return $v_k$.
```

We denote prolongations by $p_k^{k-1} : V_k \rightarrow V_{k-1}$, restrictions by $r_{k-1}^k : V_{k-1}' \rightarrow V_k'$, coarse grid operators by $A_k : V_k \rightarrow V_k'$ for $k = 1, \ldots, J$, and smoothers by $S_k : V_k \times V_k' \rightarrow V_k$ for $k = 0, \ldots, J$, where $S_k^\nu(v_k, b_k)$ denotes the result of ν smoothing steps. Very often the Galerkin product

$$A_{k+1} := r_k^{k+1} A_k p_{k+1}^k\,, \quad k \geq 0\,, \tag{3}$$

is applied to get the coarse grid operators. In the Galerkin case the prolongation and restriction is given by the injection $j : V_k \hookrightarrow V_{k-1}$ and the adjoint operator $j^* : V_{k-1}' \hookrightarrow V_k'$. The canonical prolongation $p_k^{k-1} : \mathbb{R}^{N_k} \rightarrow \mathbb{R}^{N_{k-1}}$ reads for bilinear finite elements using stencil notation

$$p_k^{k-1} = \begin{bmatrix} 1/4 & 1/2 & 1/4 \\ 1/2 & 1 & 1/2 \\ 1/4 & 1/2 & 1/4 \end{bmatrix},$$

which is identical to a bilinear interpolation. The canonical restriction is defined by the adjoint: $r_{k-1}^k = \left(p_k^{k-1}\right)^*$. We refer to them as standard transfers.

To apply a geometric multigrid method we define a hierarchy of grids by successive geometric coarsening, i.e. starting from the finest grid we merge four adjacent elements to an element of the next coarser grid, see Fig. 1. Thus, we can deduce partitions $\tau_{h_0}, \tau_{h_1}, \ldots, \tau_{h_J}$, $h_k = 2h_{k-1}$, which are assigned to the finite element spaces V_k with the property $V_0 \subset V_1 \subset \cdots \subset V_J$. The hierarchy of grids is defined by $\Omega_k := \Omega_{h_k}$ for $k = 0, \ldots, J$ with mesh sizes $h_k = h_{k+1}/2$. For the definition of multigrid transfers in Sect. 4, we split the grid Ω_k in four disjoint grids ($k \geq 0$):

$$\begin{aligned}
\Omega_k^{00} &:= \{(x_1, x_2) \in \Omega_k |\, x_1 = 2ih_k, x_2 = 2jh_k,\ i, j \in \mathbb{Z}\} = \Omega_{k+1} \\
\Omega_k^{10} &:= \{(x_1, x_2) \in \Omega_k |\, x_1 = (2i+1)h_k, x_2 = 2jh_k,\ i, j \in \mathbb{Z}\} \\
\Omega_k^{01} &:= \{(x_1, x_2) \in \Omega_k |\, x_1 = 2ih_k, x_2 = (2j+1)h_k,\ i, j \in \mathbb{Z}\} \\
\Omega_k^{11} &:= \{(x_1, x_2) \in \Omega_k |\, x_1 = (2i+1)h_k, x_2 = (2j+1)h_k,\ i, j \in \mathbb{Z}\}\,.
\end{aligned}$$

Unlike geometric multigrid methods algebraic methods try to match the multigrid components adapted to the problem to reduce the total error spectrum by smoothing

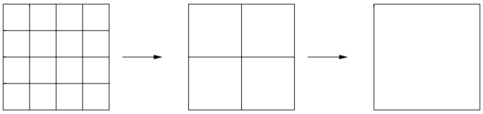


Fig. 1. Geometric coarsening

and coarse grid correction. This can be done using robust smoothers like ILU$_\beta$, see e.g. [20]. However, in the past 15 years algebraic methods tended to adjust the coarse grid correction and to apply a simple smoothing. As a consequence, the transfer operators must be adjusted when computing the coarse grid operators A_k by the Galerkin product. The construction of suitable transfers p_{k+1}^{k}, r_k^{k+1} can be made by preserving continuous quantities such as flux when transferring the defect d_k or correction e_k between the grids. In Sect. 5, we apply matrix-dependent grid transfers proposed by Alcouffe et al., see [1, 5]. They make use of the operators A_k to construct the prolongations and to guarantee the continuity of the flux. These grid transfers proved useful for solving diffusion problems by multigrid methods for media with jumping coefficients, periodic and random media, and media with inclusions, as shown in [1, 21, 13, 15]. For purely algebraic methods, such as the method of Ruge and Stüben (MG-RS) [17] and the Schur-complement method (MG-SC), see [18], robustness is also proved in practical cases, see [17, 18]. The method of Ruge and Stüben applies an appropriate coarsening algorithm to construct the grids adapted to the problem. MG-SC applies the geometric coarsening.

2.4 Model Problem

For the numerical experiments we solve the flow equation (1) in the domain $\Omega = [0,1]^2$ with vanishing source term $\varrho(x) \equiv 0$. For the discretization of (1) the fine-scale permeability $K(x)$ within an element $T_j^{(0)}$ is set to the value at the midpoint $\tilde{x}\big(T_j^{(0)}\big)$ of the element, that is, $K(x)$ is defined by

$$K(x) := K\big(\tilde{x}(T_j^{(0)})\big) \quad \text{for all } x \in T_j^{(0)} .$$

By the geometric coarsening we obtain the hierarchy τ_i, $i > 0$, associated with the grids Ω_{h_i} on Ω, see (2). The boundary conditions are:

$$\begin{aligned} u(x) &= 1 \quad \text{for } x_1 = 0\,, x_2 \in [0,1]\,, \\ u(x) &= 0 \quad \text{for } x_1 = 1\,, x_2 \in [0,1]\,, \end{aligned}$$

and $n(x) \cdot \big(K(x)\nabla u(x)\big) = 0$ otherwise. The parameters for the multigrid cycle are always chosen to be $\nu_1 = \nu_2 = 1$ and $\gamma = 1$ (V-cycle). The smoother S is a symmetric Gauss-Seidel smoother (SGS), and the initial vector is the zero vector. On the coarsest grid level J, we also apply SGS as solver.

The convergence of the multigrid methods is determined by the averaged convergence rate $\overline{\rho} = \big(||d^{(m)}||/||d^{(0)}||\big)^{1/m}$, if m iteration steps are necessary to reduce the euclidean norm of the initial defect $||d^{(0)}||$ by 10 orders of magnitude for instance. We fix the stopping criterion of the methods to a relative reduction of 10^{-10} or 80 as a maximum of iterations. In our terminology an iterative solver qualifies as fast or optimal if its speed of convergence is best.

3 Numerical Coarse Graining

We briefly describe the coarse graining method for the upscaling of the flow equation. The concept of coarse graining for flow in porous media was developed in [6, 7]. It models the flow equation on larger scales starting from the equation with the fine-scale permeability $K(x)$. The result is an upscaled flow equation, which does not model the fine-scale heterogeneity up to an arbitrary length scale λ explicitly. The influences of subscale fluctuations are modeled by a scale-dependent effective permeability K^{eff} which incorporates the impact of the unresolved fine-scale fluctuations.

In the following, we denote the coarser length scale by λ and apply Einstein's sum convention. The Fourier transform is defined by

$$\hat{f}(q) := \int f(x) \exp(-\mathrm{i}x \cdot q)\mathrm{d}^d x \; .$$

3.1 Coarse Graining of the Flow Equation

The coarse graining is guided by the idea to smooth a function for local volumes of magnitude λ^d to get a function on a coarser resolution scale. The smoothing of the function is done in Fourier space, i.e. high oscillatory modes are eliminated by cutting off the function values of the Fourier transform $\hat{u}$ for large wave vectors. The starting point is the flow equation (1) with a permeability $K_{ij}(x) = \overline{K}_{ij} + k_{ij}(x)$, $i, j = 1, \dots, d$, where $K_{ij}(x) \in W^{1,\infty}(\mathbb{R}^d)$ is a realization of a scalar, lognormally distributed random field with mean $\overline{K}_{ij}$ and fluctuations $k_{ij}(x)$. As boundary conditions we choose zero boundary conditions at infinity and assume $u(x) \in H^2(\mathbb{R}^d)$. The flow equation in Fourier space then reads

$$-iq_i \, \overline{K}_{ij} iq_j \hat{u}(q) - iq_i \int \hat{k}_{ij}(q - q') \, iq'_j \, \hat{u}(q') \, \mathrm{d}^d q' = \hat{\varrho}(q) \, , \tag{4}$$

and we define projections for cutting off high and low frequency modes in Fourier space:

$$P^{+}_{\lambda,q}\big(\hat{u}(q)\big) := \begin{cases} \hat{u}(q) & \text{if } |q_i| > a_s/\lambda \text{ for an } i \in \{1, \dots, d\} \\ 0 & \text{otherwise,} \end{cases}$$

$$P^{-}_{\lambda,q}\big(\hat{u}(q)\big) := \begin{cases} \hat{u}(q) & \text{if } |q_i| \le a_s/\lambda \text{ for all } i \in \{1, \dots, d\} \\ 0 & \text{otherwise,} \end{cases}$$

where $P^+_{\lambda,q}\big(P^+_{\lambda,q}(\hat{u}(q))\big) = P^+_{\lambda,q}\big(\hat{u}(q)\big)$ and $a_s \geq 1$ is a constant. The idea is to split $\hat{u}$ by the projections, that is, $\hat{u}(q) = P^-_\lambda\left(\hat{u}(q)\right) + P^+_\lambda\left(\hat{u}(q)\right)$, and to project equation (4) onto high and low frequencies. Substituting the result for $P^+_\lambda\left(\hat{u}(q)\right)$ into the equation for $P^-_\lambda\left(\hat{u}(q)\right)$, one deduces a closed expression for $P^-_\lambda\left(\hat{u}(q)\right)$. As shown in [6], this procedure leads to an upscaled flow equation for an arbitrary scale λ based on the equation for $P^-_\lambda\left(\hat{u}(q)\right)$:

$$P^-_{\lambda,q}\Big(q_i K^{\text{eff}}_{ij}(\lambda) q_j\, \hat{u}(q) - i q_i \int \hat{k}_{ij}(q-q') i q'_j P^-_{\lambda,q'}\left(\hat{u}(q')\right) \mathrm{d}^d q'\Big) = P^-_{\lambda,q}\big(\hat{\varrho}(q)\big)\,. \tag{5}$$

Equation (5) includes the effective permeability tensor

$$K^{\text{eff}}_{ij}(\lambda) := \overline{K}_{ij} + \int \overline{\hat{k}_{il}(-q) i q_l\, P^+_{\lambda,q}\big(P^+_{\lambda,q'}\left(\hat{G}(q,-q')\right) i q'_m \hat{k}_{mj}(q')\big)}\, \frac{\mathrm{d}^d q' \mathrm{d}^d q}{(2\pi)^{2d}} \tag{6}$$

where $\hat{G}(q,q')$ is Green's function of the flow equation (4) in Fourier space. Applying a perturbation theory and a Renormalization group analysis $K^{\text{eff}}(\lambda)$ can be analyzed, and explicit results can be calculated, see [6]. According to (5) the upscaled flow equation reads in real space:

$$-div\left(K^{\text{eff}}(\lambda) + k(x)|_\lambda\right)\nabla u_\lambda(x) = \varrho(x)|_\lambda\,,$$

where $k(x)|_\lambda$ and $\varrho(x)|_\lambda$ are the upscaled quantities, and $u_\lambda(x)$ is the solution on the scale λ.

3.2 Numerically Upscaled Effective Permeability

The coarse graining method can be extended to a numerical upscaling technique. Local effective permeability coefficients are constructed by deriving an upscaled coefficient for a given realization. This leads to a numerically upscaled coefficient which depends on x. As a result, a varying effective permeability field is obtained, and by iteration, a hierarchy of fields can be established which successively corresponds to larger scales.

The effective permeability (6) leads to the approximate effective tensor

$$K^{\text{eff}}_{ij}(\lambda) \approx \overline{K}_{ij} + \int \overline{k_{il}(x)\, \partial_{x_l}\Big(\int_{E^{(x)}_\lambda} G(x,x')\, \partial_{x'_m} k_{mj}(x')\, \mathrm{d}^d x'\Big)}\mathrm{d}^d x$$

in real space, as shown in [6]. $G(x,x')$ denotes Green's function of the flow equation, and $E^{(x)}_\lambda$ defines the d-dimensional cube $E^{(x)}_\lambda := \prod_{i=1}^d [x_i - \lambda/a_s, x_i + \lambda/a_s]$ around x. The upscaled permeability tensor for a single realization is then defined by

$$K^{\text{real}}_{ij}(\lambda) := \overline{K}_{ij} + \int k_{il}(x)\, \partial_{x_l} \int_{E^{(x)}_\lambda} G(x,x')\, \partial_{x'_m} k_{mj}(x')\, \mathrm{d}^d x' \mathrm{d}^d x\,.$$

Further, Green's function $G(x,x')$ in K^{real} is replaced by a Green's function $G^{(x)}(x',x'')$ for $E_\lambda^{(x)}$ which fulfills[1]

$$-\text{div}\left(\overline{K}+k(x')\right)\nabla G^{(x)}(x',x'')=\delta(x'-x'')$$

in $E_\lambda^{(x)}$, $x\in\mathbb{R}^d$ fixed. Using

$$\delta K_{ij}^{\text{num}}(x',\lambda):=k_{il}(x')\,\partial_{x'_l}\int_{E_\lambda^{(x)}}G^{(x)}(x',x'')\,\partial_{x''_m}k_{mj}(x'')\,\mathrm{d}^dx''\quad\text{for } x'\in E_\lambda^{(x)}\,,$$

an upscaled permeability coefficient for x can be determined by

$$\overline{K}_{ij}+\int_{E_\lambda^{(x)}}\delta K_{ij}^{\text{num}}(x',\lambda)\,\mathrm{d}^dx'\,.$$

We introduce the function

$$\chi_j^{(x)}(x'):=\int_{E_\lambda^{(x)}}G^{(x)}(x',x'')\,\partial_{x''_m}k_{mj}(x'')\,\mathrm{d}^dx''\ ,\ x'\in E_\lambda^{(x)}\,,$$

which fulfills for fixed x the differential equation

$$\text{div}\,K(x')\nabla\left(\chi_j^{(x)}(x')+x'_j\right)=0\quad\text{for } x'\in E_\lambda^{(x)}\,,\tag{7}$$

and appropriate boundary conditions. Now, the definition for the numerically upscaled permeability coefficient, which depends on x for $E_\lambda^{(x)}$, is given by:

Definition 1 (Numerically upscaled permeability).

$$K_{ij}^{num}(x,\lambda):=\overline{K}_{ij}+\int_{E_\lambda^{(x)}}\delta K_{ij}^{num}(x',\lambda)\,\mathrm{d}^dx'=\overline{K}_{ij}+\int_{E_\lambda^{(x)}}k_{il}(x')\,\partial_{x'_l}\chi_j^{(x)}(x')\,\mathrm{d}^dx'\,.$$

Theorem 1. *If $K_{ij}(x)$ is diagonal and positive definite, then K_{ij}^{num} is symmetric and positive definite in the limit $\lambda\longrightarrow\infty$. Moreover, by the substitution $\overline{K}_{ij}\ \longrightarrow \int_{E_\lambda^{(x)}}K_{ij}(x')\,\mathrm{d}^dx'$ we get for finite length scales λ that*

$$\int_{E_\lambda^{(x)}}K_{ij}(x')\,\mathrm{d}^dx'+\int_{E_\lambda^{(x)}}k_{il}(x')\,\partial_{x'_l}\chi_j^{(x)}(x')\,\mathrm{d}^dx'$$

is symmetric, positive definite, if the function $\chi_j^{(x)}$ fulfills one of the following boundary condition on $\partial E_\lambda^{(x)}$: i) Dirichlet-zero boundary conditions, ii) periodic boundary conditions if K is periodic with period $2\lambda/a_s$, or iii) below defined NZF-boundary conditions, and $\int_{E_\lambda^{(x)}}\overline{K}_{il}\partial_{x'_l}\chi_j^{(x)}(x')\mathrm{d}^dx'=0$.

Proof. See [6].

[1] The superscript x in $G^{(x)}$ is regarded as an index.

Definition 2 (NZF-boundary conditions). *The NZF-boundary conditions*[2] *consist of mixed boundary conditions for* $\chi_j^{(x)}$ *on* $\partial E_\lambda^{(x)}$. *They are for* x *and* $j = 1, \ldots, d$ *fixed:*

$$\begin{aligned} \chi_j^{(x)}(x') = 0 \quad & \textit{if } x'_j = x_j - \lambda/a_s \textit{ or } x'_j = x_j + \lambda/a_s\,, \\ n(x') \cdot \big(K(x')\nabla\chi_j^{(x)}(x')\big) = 0 \quad & \textit{otherwise,} \end{aligned}$$

with $x' \in \partial E_\lambda^{(x)}$. $n(x')$ *denotes the outer unit normal vector of* $E_\lambda^{(x)}$ *in* x'.

As proved in [6], the auxiliary equation (7) corresponds to the cell problem in the method of homogenization in the case of periodic permeability fields.

3.3 Numerical Computation of Upscaled Fields

We consider the unit square Ω and a finest partition $\tau_0 = \big\{T_j^{(0)}\big\}_{1\le j\le N}$ given by the grid Ω_0 with mesh size h_0, as described in Sect. 2. The permeability field $K(x)$ is generated as a realization of the lognormally distributed random field and is given by $K(x) := K\big(\tilde{x}(T_j^{(0)})\big)$ for all $x \in T_j^{(0)}$ on the uniform grid of rectangles, analogously to Sect. 2.4. By geometric coarsening we obtain the hierarchy $\tau_i = \big\{T_j^{(i)}\big\}$, $i > 0$, associated with the grids Ω_i, see (2), with sizes $h_i = 2^i h_0$. For the computation of the upscaled permeability for a given element $T_j^{(i)} \in \tau_i$, $i > 0$, we set $E_\lambda^{(x)} = T_j^{(i)}$ with $x = \tilde{x}\big(T_j^{(i)}\big)$. Definition 1 leads to

$$K_{lm}^{\text{num}}(x,\lambda) = \overline{K}_{lm} + \int_{T_j^{(i)}} k_{ln}(x')\,\partial_{x'_n}\chi_m^{(x)}(x')\,\mathrm{d}^2x' \tag{8}$$

for $x = \tilde{x}(T_j^{(i)})$, $\lambda = h_i$, and $\overline{K}_{lm} = \int_\Omega K_{lm}(x)\mathrm{d}^2x$. We define this coefficient as the upscaled permeability for all $x \in T_j^{(i)}$:

$$K^{\text{num}}(x,\lambda) := K^{\text{num}}\big(\tilde{x}(T_j^{(i)}),\lambda\big) \qquad \text{for all } x \in T_j^{(i)}, \lambda = h_i\,.$$

So, the numerically upscaled permeability field on the scale $\lambda = h_i$ is given by $K(x)|_\lambda := K^{\text{num}}\big(\tilde{x}(T_j^{(i)}),\lambda\big) + k(x)\big|_\lambda$ for all $x \in T_j^{(i)}$, where the upscaled fluctuations $k(x)|_\lambda$ are computed by

$$k_{lm}(x)|_\lambda := \frac{1}{\int_{T_j^{(i)}}\mathrm{d}^2x'}\int_{T_j^{(i)}} k_{lm}(x')\,\mathrm{d}^2x' \quad \text{for } x \in T_j^{(i)}\,. \tag{9}$$

The solution for $\chi_m^{(x)}(x')$ of the differential equation (7) in $E_\lambda^{(x)} = T_j^{(i)}$ is computed for $K^{\text{num}}(x,\lambda)$ by the bilinear finite element method using SGS as a solver. We select the partition τ_T for discretizing (7) on $T_j^{(i)}$ for all j and $i \ge 0$ so fine, so

[2] NZF stands for Dirichlet zero (null) and zero flux.

that $K(x)$ is constant on each element of τ_T, and that we can apply the midpoint rule for the computations. As a stop criterion for the solver we choose the relative error being reduced to 10^{-10} or the absolute error being reduced to 10^{-15}.

As a result of the discretization K^{num} depends on the mesh size h_T of the chosen grid τ_T for T and on the boundary conditions for $\chi_m^{(x)}(x')$. As shown in [6, 7], the numerically upscaled permeability converges very fast to the asymptotic value for grids with $h_T/h_0 < 1/8$. Consequently, we choose the mesh size for the computation of K^{num} to $h_T/h_0 \leq \frac{1}{32} l_0/\lambda$.

Iterative Upscaling of the Fields

Since we consider a hierarchy of partitions τ_i we are able to perform an iterative upscaling by successive coarse graining from the field of the previous coarse graining step. The advantage is that the mesh size h_T for the computation of K^{num} for $\lambda/l_0 \geq 1/8$ must not be chosen as fine as in the case of upscaling from the fine-scale field.

We define $K(x)|_{\lambda_0} := K(x)$ and take $\lambda_0 < \lambda_1 < \ldots < \lambda_J$ with $\lambda_i = h_i$ as given scales. Under the assumption that the field $K(x)|_{\lambda_i}$ on λ_i, $i \geq 0$, exists, the field $K(x)|_{\lambda_{i+1}}$ can be computed by

$$K(x)|_{\lambda_{i+1}} := K^{\text{num}}\big(\tilde{x}(T_j^{(i+1)}), \lambda_{i+1}\big) + k(x, \lambda_i)|_{\lambda_{i+1}}\,,$$

for $x \in T_j^{(i+1)}$, $\lambda_{i+1} = h_{i+1} = 2\lambda_i$, and

$$K_{lm}^{\text{num}}\big(\tilde{x}(T_j^{(i+1)}), \lambda_{i+1}\big) = \overline{K_{lm}(x)|_{\lambda_i}} + \int_{T_j^{(i+1)}} k_{ln}(x')|_{\lambda_i}\,\partial_{x'_n}\chi_m^{(x)}(x')\,\mathrm{d}^2x'\,,$$

and $k(x, \lambda_i) := K(x)|_{\lambda_i} - \overline{K(x)|_{\lambda_i}}$.

4 Coarsening Multigrid Method

The Coarsening multigrid method (CN-MG) is an algebraically defined multigrid method which benefits from the numerical upscaling by the coarse graining. It uses the iterative upscaling of the permeability field to determine the coarse grid operators $A_l := A^{(l)}$ by discretizing the flow equation on these fields. As the upscaled fields $K(x)|_{\lambda=h_l}$ include only long wave fluctuations varying on scales larger than the scale λ, the so-defined coarse grid operators reduce the low frequency error spectrum and can efficiently interplay with the smoother.

The coarse grid operator on grid level l of CN-MG is defined by the entries of the stiffness matrix

$$A_{ij}^{(l)} := \sum_{m,n=1}^{2} \int_{\Omega} K_{mn}^{\text{CN}}(x)|_{\lambda_l}\,\partial_{x_n}\psi_j^{(l)}(x)\,\partial_{x_m}\psi_i^{(l)}(x)\,\mathrm{d}^2x\,, \tag{10}$$

where $\lambda_l = h_l$ for the partition τ_l. To ensure that the iteratively upscaled field $K(x)|_{\lambda_l}$ is symmetric and positive definite for $x \in \Omega$, which can be violated in cases of high variances σ_f^2, we apply the following modifications: We choose the upscaled permeability field in CN-MG as

$$K^{\mathrm{CN}}(x)|_{\lambda_{l+1}} = \begin{cases} K(x)|_{\lambda_{l+1}} + \left(\overline{K(x)|_{\lambda_{l+1}}} - K^{\mathrm{num}}(x,\lambda_{l+1})\right)\big|_{\lambda_{l+1}} & \text{for } \lambda/l_0 < \theta \\ K(x)|_{\lambda_{l+1}} & \text{otherwise,} \end{cases}$$

where the corrector field $\left(\overline{K(x)|_{\lambda_l}} - K^{\mathrm{num}}(x,\lambda_l)\right)\big|_{\lambda_{l+1}}$ is given by a simple arithmetic smoothing, analogous to the smoothed fluctuations $k(x,\lambda_l)|_{\lambda_{l+1}}$ in (9). To avoid choosing θ depending on x, we substitute the field $K^{\mathrm{CN}}(x)|_{\lambda_{l+1}}$ where it still violates the positiveness by

$$\frac{1}{h_{l+1}^2} \int_{T_j^{(l+1)}} K^{\mathrm{CN}}(x)|_{\lambda_l} \, \mathrm{d}^2 x \quad \text{for } x \in T_j^{(l+1)} . \tag{11}$$

The parameter θ is chosen as $\theta(\sigma_f^2) = 1/8$ for $\sigma_f^2 \geq 1$, and $\theta(\sigma_f^2) = 0$ otherwise. In all cases where $K(x)$ does not rely on a realization of the random field we set $\theta = 0$.

4.1 Setup Phase of CN–MG

The setup phase for the Coarsening multigrid method is as follows.

1. Iterative upscaling of the permeability fields, starting with $K(x)|_{\lambda_0} = K(x)$, using geometric coarsening to get the hierarchy $\tau_l := \tau_{h_l}$, $l > 0$.
   - The cell problem $\mathit{div}\, K(x')|_{\lambda_l} \nabla\left(\chi_i^{(x)}(x') + x_i'\right) = 0$, $\lambda_l = h_l$, has to be solved on each $T_j^{(l)}$ with given boundary conditions. Due to (8), $K^{\mathrm{num}}(x,\lambda_l)$ for $T_j^{(l)}$ is computed via $\chi_i^{(x)}$.
   - Symmetrizing of K^{num}: $K_{mn}^{\mathrm{num}}(x,\lambda_l) := \frac{1}{2}\left(K_{12}^{\mathrm{num}}(x,\lambda_l) + K_{21}^{\mathrm{num}}(x,\lambda_l)\right)$ for $m \neq n$.
   - Computation of $K^{\mathrm{CN}}(x)|_{\lambda_l}$ including the modifications due to (11). Therefore, the coefficients $K^{\mathrm{num}}(x,\lambda_l)$ and $K^{\mathrm{num}}(x,\lambda_{l-1})$ for $T_j^{(l)}$, and the smoothed fluctuations $k(x,\lambda_{l-1})|_{\lambda_l}$ are used.
2. Discretizing the differential operator $\mathit{div}\, K^{\mathrm{CN}}(x)|_{\lambda_l} \nabla u(x)$. According to (10), the discretization yields the operator A_l on the grid level l.

The cell problems are solved by SGS, and the stopping criterion is given by the relative or absolute reduction of the initial defect to 10^{-10} or 10^{-15}, or by 60 iteration steps.

4.2 Transfer Operators for CN–MG

The idea for the construction of appropriate grid transfers for the Coarsening multigrid method is taken from the method of homogenization. In the homogenization theory, the fine-scale solution is well approximated by the homogenized solution plus a correction including the solution to the cell problem, see e.g. [11]. Since problem (7) is similar to the cell problem on each element $T_j^{(l)} \in \tau_l$, we exploit the solutions $\chi_i^{(x)}$ of the cell problems in a similar fashion for the definition of the prolongation p of a correction $e_l(x)$, see [6]:

$$e_{l-1}(x) = \big(p_l^{l-1} e_l\big)(x) := e_l(x) + \frac{h_{l-1}}{h_l} \sum_{i=1}^{2} \chi_i^{(\tilde{x}(T_j^{(l)}))}(x)\, \partial_{x_i} e_l(x)$$

for $x \in T_j^{(l)} \cap \big(\Omega_{l-1} \setminus \Omega_l\big)$. Due to the fact that for adjacent elements $T_{j_1}^{(l)}$ and $T_{j_2}^{(l)}$ the solutions $\chi^{(\tilde{x}(T_{j_1}^{(l)}))}$ and $\chi^{(\tilde{x}(T_{j_2}^{(l)}))}$ are usually not identical, the prolongation is not unique in grid points $\Omega_{l-1}^{10} \cup \Omega_{l-1}^{01}$ which do not lie on the boundary $\partial\Omega$. Thus, we define the prolongation by:

Definition 3 (CN-Prolongation). *In fine grid points $\Omega_{l-1}^{00} = \Omega_l$, the prolongation is defined by the identical mapping:*

$$e_{l-1}(x) = \big(p_l^{l-1} e_l\big)(x) := e_l(x) \quad \textit{for } x \in \Omega_{l-1}^{00}\,.$$

In grid points $\Omega_{l-1} \setminus \Omega_{l-1}^{00}$, the prolongation is defined by:

$$\begin{aligned} e_{l-1}(x) = \big(p_l^{l-1} e_l\big)(x) &:= e_l(x) \\ &+ \frac{h_{l-1}}{h_l \sum_{T^{(l)} \textit{ with } x \in T^{(l)}} 1} \sum_{i=1}^{2} \left(\sum_{T^{(l)} \textit{ with } x \in T^{(l)}} \chi_i^{(\tilde{x}(T^{(l)}))}(x)\, \partial_{x_i} e_l(x) \right). \end{aligned}$$

The CN-restriction is defined by the adjoint operator of the CN-prolongation.

5 Numerical Results

5.1 Numerical Test of Convergence of CN–MG

We investigate the numerical convergence of different variants of the Coarsening multigrid method. These variants arise from employing different grid transfers and different boundary conditions for solving the cell problems in CN-MG. We fix the following notations:

- CN-ST: Coarsening multigrid with standard transfers.
- CN-M: Coarsening multigrid with matrix-dependent transfers as given by Alcouffe et al. [1].
- CN-CP: Coarsening multigrid with CN-prolongation and CN-restriction.

For the comparisons we compute the arithmetic mean of the convergence rates $\overline{\rho}$ for 10 realizations of $K(x)$ with correlation length $l_0 = 1/16$, and variances $\sigma_f^2 = 1$ and $\sigma_f^2 = 2$. The mesh size of the finest grid is $h_0 = 2^{-8}$. In addition, we vary the mesh size h_T/h_{l+1} of the partition τ_T for the elements of τ_{l+1} for solving the cell

Table 1. For variance $\sigma_f^2 = 1$: Averaged convergence rates for 10 isotropic realizations for CN-MG. h_T/h_0 is the mesh size of the grids for the cell problems, and ZBC and NZF denote Dirichlet zero boundary conditions and NZF-boundary conditions for the cell problems, respectively

h_T/h_0	CN-ST ZBC	CN-M ZBC	CN-CP ZBC	CN-ST NZF	CN-M NZF	CN-CP NZF
$1/2$	0.182	0.134	0.180	0.090	0.068	0.063
$1/4$	0.164	0.118	0.163	0.088	0.074	0.066
$1/8$	0.159	0.113	0.159	0.089	0.083	0.071
$1/16$	0.157	0.112	0.156	0.094	0.119	0.080

Table 2. For variance $\sigma_f^2 = 2$: Averaged convergence rates for 10 isotropic realizations for CN-MG

h_T/h_0	CN-ST ZBC	CN-M ZBC	CN-CP ZBC	CN-ST NZF	CN-M NZF	CN-CP NZF
$1/2$	0.415	0.304	0.413	0.207	0.262	0.183
$1/4$	0.388	0.264	0.389	0.177	0.322	0.143
$1/8$	0.377	0.251	0.378	0.161	0.318	0.123
$1/16$	0.374	0.247	0.375	0.170	0.271	0.130

problems in CN-MG. We always choose h_T/h_{l+1} independent of the grid level l, i.e. h_T is proportional to h_{l+1}. For simplicity, we denote h_T/h_{l+1} by h_T/h_0 in the following.

As can be seen from Table 1 and 2, the method CN-CP (NZF) performs best for nearly all test cases. CN-M is best among the variants using Dirichlet zero boundary conditions for the cell problems. The rates for CN-ST and CN-CP hardly differ from each other in the case of Dirichlet zero boundary conditions (ZBC). This is due to the fact that the standard prolongation and CN-prolongation merely differ in grid points $\Omega_l^{11} \subset \Omega_l$ in the ZBC case. For the case of NZF-boundary conditions the method CN-CP is optimal. Compared with CN-ST (ZBC) and CN-M (ZBC) the CN-CP (NZF) variant is more than 48.7% faster.

The numerical results also demonstrate clearly that the convergence of CN-MG depends on h_T/h_0. However, for CN-CP (NZF) and the methods with zero boundary conditions the rates improve for decreasing h_T in the range of $h_T/h_0 \geq 1/8$.

In the following, we consider only the Coarsening multigrid variants CN-CP (NZF) and CN-M (ZBC) which we simply denote by CN-CP and CN-M without indicating the boundary conditions for the cell problems. Further, we choose the mesh size for the cell problems in both methods to be $h_T/h_0 = 1/8$.

h-dependence of CN–MG

The dependence of the convergence on the finest mesh size h_0 is shown in Table 3 for the Coarsening methods. The table displays the rates $\overline{\rho}$ for three isotropic realizations with growing variance σ_f^2. Since the heterogeneity of the fine-scale permeability $K(x)$ are resolved better for smaller h_0, the rates improve for decreasing mesh size. So, the convergence rates clearly show that CN-MG gets more efficient for increasing the resolution of the fine-scale fluctuations of the permeability.

Table 3. Convergence rates for CN-M and CN-CP, and their dependence on the mesh size h_0 of the finest partition for three different realizations

Number of	h_0	Variance $\sigma_f^2 = 1$		Variance $\sigma_f^2 = 2$		Variance $\sigma_f^2 = 3$	
grid points		CN-M	CN-CP	CN-M	CN-CP	CN-M	CN-CP
1089	2^{-5}	0.169	0.107	0.243	0.130	0.533	0.272
4225	2^{-6}	0.158	0.093	0.229	0.119	0.512	0.457
16641	2^{-7}	0.141	0.084	0.212	0.077	0.490	0.229
66049	2^{-8}	0.130	0.071	0.192	0.116	0.468	0.231
263169	2^{-9}	0.112	0.064	0.179	0.099	0.444	0.203
1050625	2^{-10}	0.094	0.052	0.158	0.089	0.418	0.184

5.2 Numerical Comparison of Multigrid Methods

In this section we test various multigrid methods for solving the model problem as given in Sect. 2.4. We compare the multigrid solvers on the basis of their numerical convergence rates. In addition to the method of Ruge and Stüben (MG-RS), the Schur-complement method (MG-SC), and the Coarsening methods, we check multigrid methods applying simple upscaling methods to determine the coarse grid operators in combination with matrix-dependent transfers.

For simple upscaling schemes such as arithmetic or geometric averaging, we define the upscaled permeability fields and the coarse grid operator A_l similar to the Coarsening method in the previous section. On grid level l we define analogous to (10)

$$A_{ij}^{(l)} := \sum_{m,n=1}^{2} \int_{\Omega} K_{mn}^{\mathrm{ups}}(x)|_l \, \partial_{x_n} \psi_j^{(l)}(x) \, \partial_{x_m} \psi_i^{(l)}(x) \, \mathrm{d}^2 x \, . \tag{12}$$

The upscaled field $K^{\mathrm{ups}}(x)|_l$ stems from iterative upscaling of $K^{\mathrm{ups}}(x)|_{l-1}$ on the partition $\mathcal{T}_{l-1}$ applying simple upscaling of the permeability over elements $T_j^{(l-1)}$ which belong to one element $T^{(l)}$ of $\mathcal{T}_l$. For $l = 0$ we define $K^{\mathrm{ups}}(x)|_0 \equiv K(x)$. We indicate the different computation of the coarse grid operators by:

- MGA: Computation due to (12) where $K^{\mathrm{ups}}(x)|_l$ is given by arithmetic averaging.
- MGG: Computation due to (12) where $K^{\mathrm{ups}}(x)|_l$ is given by geometric averaging.
- GAP: Computation by the Galerkin product, see (3).

For the grid transfers we define the following notations: ST: standard transfers, M: matrix-dependent transfers due to Alcouffe et al. [1], and CP: CN-prolongation and CN-restriction. For CN-MG, we fix the parameter $\theta(\sigma_f^2)$ to

$$\theta(\sigma_f^2) = \begin{cases} 1/2 & \text{for } \sigma_f^2 \geq 2 \\ 1/8 & \text{for } 1 \leq \sigma_f^2 < 2 \\ 0 & \text{for } \sigma_f^2 < 1 \end{cases}$$

which proved suitable for all configurations of the medium.

Comparison for an Ensemble of Realizations

We consider the model problem for fields $K(x)$ being realizations of the random field. Table 4 shows the arithmetic averaged convergence rates for 10 isotropic realizations and varying variance σ_f^2. The Tables 5 and 6 consist of the rates for 10 realizations which show an anisotropic correlation, where the correlation lengths are: $(l_1, l_2) = (2l_0, l_0/2)$ and $(l_1, l_2) = (8l_0, l_0/2)$, $l_0 = 1/16$. The mesh size for the finest partition $\mathcal{T}_0$ is set to $h_0 = 2^{-8}$.

The numbers in brackets in the tables count the realizations where the method does not converge. In that case, the rate is the arithmetic average of the rates of the convergent realizations.

Table 4. Convergence rates for different multigrid methods for 10 realization of isotropic correlation for increasing variance σ_f^2 of the fields

σ_f^2	MGA-ST	MGA-M	MGG-ST	GAP-ST	GAP-M	CN-M	CN-CP	MG-SC	MG-RS
0.1	0.057	0.051	0.051	0.056	0.050	0.050	0.050	0.075	0.119
1	0.261	0.202	0.108	0.259	0.064	0.114	0.070	0.130	0.137
2	0.503	0.433	0.201	0.508	0.124	0.260	0.122	0.228	0.177
3	0.658	0.576	0.300	0.657	0.121	0.380	0.188	0.352	0.213
4	0.759	0.710	0.379 [1]	0.765	0.198	0.527	0.352	0.514	0.174
5	0.781	0.737	0.439	0.781	0.193	0.553	0.361	0.471	0.200

The numerical results for the convergence in Table 4 in the isotropic case show that the Galerkin product method GAP-M is the fastest in most cases. The Coarsening method CN-CP also shows very fast convergence but shows slower convergence than GAP-M for variances $\sigma_f^2 \geq 3$. However, CN-CP is faster than MG-SC and MG-RS for $\sigma_f^2 \leq 3$, whereas the method of Ruge and Stüben is very fast for high variances. The rates of CN-M are similar to the results for MG-SC. The multigrid methods that take simple upscaling for computation of the coarse grid operator and the standard method GAP-ST perform worse but for very small variances $\sigma_f^2 \leq 1$.

The numerical convergence results for anisotropic media are similar to the isotropic case. Here GAP-M is even faster compared with the other methods, see

Table 5. Convergence rates for 10 realizations with anisotropic correlation, $(l_1, l_2) = (2l_0, l_0/2)$, for increasing σ_f^2

σ_f^2	MGA-ST	MGA-M	MGG-ST	GAP-ST	GAP-M	CN-M	CN-CP	MG-SC	MG-RS
0.1	0.055	0.052	0.051	0.053	0.049	0.051	0.051	0.069	0.123
1	0.270	0.189	0.303	0.267	0.053	0.107	0.099	0.153	0.160
2	0.495	0.419	0.594 [1]	0.488	0.076	0.268	0.135	0.251	0.170
3	0.606	0.519	– [10]	0.608	0.127	0.350	0.237	0.320	0.178
4	0.670	0.615	–	0.659	0.149	0.444	0.331	0.411	0.205
5	0.781	0.736	–	0.788	0.211	0.586	0.403	0.536	0.190

Table 6. Convergence rates for 10 realizations with anisotropic correlation, $(l_1, l_2) = (8l_0, l_0/2)$, for increasing σ_f^2

σ_f^2	MGA-ST	MGA-M	MGG-ST	GAP-ST	GAP-M	CN-M	CN-CP	MG-SC	MG-RS
0.1	0.053	0.050	0.050	0.052	0.049	0.050	0.050	0.068	0.116
1	0.258	0.221	0.272	0.256	0.050	0.135	0.073	0.081	0.156
2	0.383	0.348	0.755 [3]	0.395	0.050	0.242	0.151	0.122	0.181
3	0.531	0.477	– [10]	0.539	0.058	0.341	0.403	0.163	0.178
4	0.633	0.592	–	0.635	0.097	0.427	0.475	0.258	0.208
5	0.668	0.622	–	0.670	0.110	0.491	0.437	0.269	0.190

Table 5 and 6. The growth of the rates for increasing variance σ_f^2 is smallest for MG-RS and the Galerkin product methods. It is also remarkable that the rates of GAP-M only vary between 0.049 and 0.110 for the anisotropic case in Table 6.

Comparison for a Single Realization

Table 7 reports the numerical convergence results for a single isotropic realization which is different for each value of the variance. The mesh size of the finest partition is set to $h_0 = 2^{-8}$. For small variances $\sigma_f^2 \leq 1$ all methods yield good results. For $\sigma_f^2 \geq 2$, the variants GAP-M, CN-CP, and MG-RS are the best methods. For increasing σ_f^2 the rates of GAP-M and CN-CP deteriorates slightly but they are still satisfactory for $\sigma_f^2 = 5$, whereas MG-RS does not show any dependence on σ_f^2.

Comparisons for Periodic Media

In the following we investigate the numerical convergence of the multigrid methods solving the model problem for periodic permeability fields $K(x)$. We denote the mesh size of the finest partition τ_0 by h_0, which is 2^{-7} for all computations of the remainder of the section.

Table 7. Convergence rates for a single realization with isotropic correlation for different values of the variance σ_f^2

σ_f^2	MGA-ST	MGG-ST	GAP-M	CN-CP	MG-SC	MG-RS
0.1	0.051	0.049	0.050	0.049	0.089	0.137
1	0.224	0.091	0.052	0.051	0.093	0.121
2	0.525	0.192	0.092	0.117	0.230	0.154
3	0.624	0.376	0.142	0.121	0.309	0.188
4	0.667	0.228	0.155	0.210	0.341	0.169
5	0.788	0.930	0.321	0.344	0.574	0.158

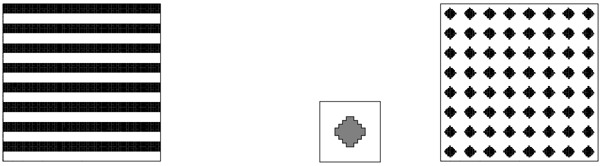


Fig. 2. (a) Periodic layered medium and (b) medium with periodic symmetric inclusions and its periodicity cell

Periodic Layered Medium

First, we consider a periodic medium consisting of two layers as shown in Fig. 2 (a). The width of the layers is $1/16$ which corresponds to $8h_0$, and the permeability $K(x)$ is given by:

$$K(x) = \begin{cases} v_1\, I & \text{for } x \in \Omega_1 \\ v_2\, I & \text{for } x \in \Omega_2\,, \end{cases}$$

where Ω_1 denotes the white layers, Ω_2 the black ones in Fig. 2 (a). I is the identity matrix, and we fix $v_1 = 1$. Table 8 lists the convergence rates for varying rate of v_2/v_1.

The rates in Table 8 show that the Galerkin product method GAP-M and the algebraic method MG-RS are best, where the method of Ruge and Stüben is optimal for $v_2 \geq 10^2$. The Schur-complement method exhibits convergence rates always smaller than 0.5. But all the other methods including the Coarsening multigrid yield acceptable results only in the range of $10^{-1} \leq v_2 \leq 10$. For $v_2 \geq 10^2$ the convergence of CN-CP is at least two times slower compared to the Galerkin product method.

Medium with Periodic Inclusions

Second, we consider a medium with periodic symmetric inclusions as shown in Fig. 2 (b). The length of the periodicity cell is $1/16$, that is $8h_0$. The permeability field is set to $K = v_2 I$ in the black area and $K = v_1 I$ in the white area for the periodicity cell. We obtain the convergence rates given by Table 9. For $v_2 \leq 1$,

Table 8. Convergence rates for a periodic layered medium for $K(x)$ as shown in Fig. 2 (a)

v_2/v_1	MGA-M	GAP-ST	GAP-M	CN-M	CN-CP	MG-SC	MG-RS
10^{-3}	0.719	0.749	0.146	0.665	0.460	0.346	0.214
10^{-2}	0.609	0.645	0.118	0.533	0.334	0.211	0.190
10^{-1}	0.270	0.290	0.050	0.192	0.119	0.074	0.214
1	0.051	0.051	0.051	0.051	0.051	0.065	0.107
10^{1}	0.315	0.310	0.066	0.229	0.175	0.073	0.179
10^{2}	0.692	0.696	0.215	0.623	0.728	0.243	0.190
10^{3}	0.799	0.798	0.375	0.767	0.905	0.407	0.192

all methods work well. Especially the method GAP-M, the Coarsening and Schur-complement method are very good. Whereas for stiff inclusions, that is $v_2 > 10$, the Galerkin product method is optimal with $\overline{\rho} < 0.16$. The Coarsening methods do not show satisfactory convergence for these values of v_2.

Table 9. Convergence rates for the medium with inclusions as permeability field $K(x)$ as shown in Fig. 2 (b)

v_2/v_1	MGA-M	GAP-ST	GAP-M	CN-M	CN-CP	MG-SC	MG-RS
10^{-3}	0.190	0.209	0.072	0.099	0.092	0.092	0.193
10^{-2}	0.185	0.204	0.071	0.098	0.090	0.091	0.202
10^{-1}	0.136	0.155	0.064	0.083	0.048	0.076	0.210
1	0.051	0.051	0.051	0.051	0.051	0.065	0.107
10^{1}	0.458	0.478	0.069	0.289	0.067	0.108	0.231
10^{2}	0.885	0.896	0.108	0.811	0.762	0.142	0.246
10^{3}	0.960	0.964	0.161	0.941	0.948	0.236	0.287

Chess Board Medium and Medium with Cross Layers

For a chess board medium as in Fig. 3 (a) with block width $1/16$, where $K(x)$ has value $v_2 I$ in the black zones and $v_1 I$ in the white zones, we obtain the rates given in Table 10. Compared with the other periodic test cases, they are worse for almost all multigrid methods. All methods except of MG-RS show good convergence only for $v_2/v_1 = 10^{-1}$ and $v_2/v_1 = 10$. Whereas the method of Ruge and Stüben is robust for all ratios v_2/v_1. The convergence rate of MG-RS is $0.21 < \overline{\rho} < 0.24$. The optimal convergence of MG-RS for the chess board medium can be deduced by its adjusted grid coarsening strategy, as shown for instance in [6].

The permeability field $K(x)$ for the cross layered medium, Fig. 3 (b), is again given by $v_2 I$ in the black area and $v_1 I$ in the white area. For moderate numbers of v_2/v_1 between 10^{-1} and 10, all multigrid methods converge fast, see Table 11. The Schur-complement method and GAP-M are optimal for small $v_2/v_1 \leq 10^{-1}$. However, for the total range of v_2/v_1 the only method which is robust is MG-RS.

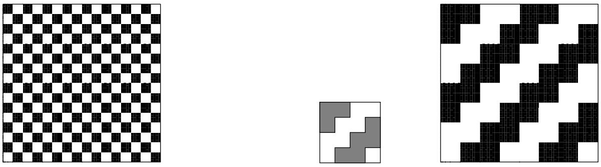


Fig. 3. (**a**) Chess board medium, (**b**) medium with periodic cross layers and its periodicity cell.

Table 10. Convergence rates for different multigrid method for a chess board medium

v_2/v_1	MGA-M	GAP-ST	GAP-M	CN-M	CN-CP	MG-SC	MG-RS
10^{-3}	0.532	0.531	0.507	0.448	0.421	0.543	0.213
10^{-2}	0.503	0.502	0.480	0.418	0.392	0.516	0.221
10^{-1}	0.303	0.303	0.284	0.229	0.221	0.328	0.235
10^{1}	0.346	0.347	0.326	0.273	0.262	0.362	0.227
10^{2}	0.583	0.583	0.561	0.499	0.478	0.591	0.211
10^{3}	0.640	0.640	0.620	0.562	0.540	0.647	0.233

Table 11. Convergence rates for the medium with cross layers as plotted in Fig. 3 (b)

v_2/v_1	MGA-M	GAP-M	CN-M	CN-CP	MG-SC	MG-RS
10^{-3}	0.527	0.182	0.445	0.543	0.136	0.191
10^{-2}	0.457	0.159	0.367	0.472	0.146	0.202
10^{-1}	0.162	0.098	0.109	0.095	0.113	0.183
10^{1}	0.278	0.116	0.202	0.140	0.176	0.175
10^{2}	0.676	0.322	0.601	0.684	0.397	0.274
10^{3}	0.786	0.480	0.729	0.791	0.464	0.228

6 Summary

We focus on multigrid methods for flow in heterogeneous porous media where the local permeability is given by a stationary random field of lognormal distribution or by a periodic medium. We consider the coarse graining method for the numerical upscaling of the permeability, see [2, 7], and we develop a new multigrid method which applies the upscaling concept to obtain the coarse grid operator. Thus the coarse grid operators of the Coarsening multigrid method (CN-MG) are adjusted to the scale-dependent fluctuations of the permeability which is important for an efficient interplay with simple smoothing. As a result the Coarsening multigrid method is adapted to the particular flow problem.

The investigation of important properties of the new method proves numerically the success of the combination of smoothing and coarse grid correction owing to the coarse graining. By a qualified choice of the boundary conditions for the cell problem and of the grid transfers we attain an improvement of the convergence of 48% for one of the new multigrid variants.

We compare the Coarsening multigrid method to algebraic methods. The convergence rates show that the variant CN-CP of the Coarsening method is as efficient as the Galerkin product and Ruge and Stüben methods for variances $\sigma_f^2 \leq 3$ which are by far sufficient for practical applications. For anisotropic and periodic media the rates of CN-CP are worse compared to GAP-M for large σ_f^2 in almost all cases. For large fluctuations in the field, MG-RS always yields good convergence or is optimal. For the medium with cross layers and the chess board it is the fastest method.

The comparison to the algebraic multigrid methods indicates that the concept of matrix-dependent transfers and an adaptive grid coarsening algorithm is indispensable for solving flow in highly heterogeneous media. In future work, we will combine the Coarsening multigrid method with adaptive coarsening strategies to improve the robustness.

References

1. R. E. Alcouffe, A. Brandt, J. E. Dendy, and J. W. Painter. The multi-grid method for the diffusion equation with strongly discontinuous coefficients. *SIAM J. Sci. Stat. Comput.*, 2(4):430–454, 1981.
2. S. Attinger, J. Eberhard, and N. Neuss. Filtering procedures for flow in heterogeneous porous media: Numerical results. *Comput. Visual. Sci.*, 5:67–72, 2002.
3. A. Brandt. Multi-level adaptive solutions to boundary-value problems. *Math. Comput.*, 31(138):333–390, 1977.
4. M. Brezina, A. J. Cleary, R. D. Falgout, V. E. Henson, J. E. Jones, T. A. Manteuffel, S. F. McCormick, and J. W. Ruge. Algebraic multigrid based on element interpolation (AMGe). *SIAM J. Sci. Comput.*, 22(5):1570–1592, 2000.
5. J. E. Dendy. Black box multigrid. *Journal of Computational Physics*, 48:366–386, 1982.
6. J. Eberhard. *Upscaling und Mehrgitterverfahren für Strömungen in heterogenen porösen Medien*. Dissertation, University of Heidelberg, Heidelberg, Germany, 2003.
7. J. Eberhard, S. Attinger, and G. Wittum. Coarse graining for upscaling of flow in heterogeneous porous media. *Multiscale Model. Simul.*, 2(2):269–301, 2004.
8. W. Hackbusch. *Multi-Grid Methods and Applications*. Springer, Berlin, 1985.
9. W. Hackbusch. *Theorie und Numerik elliptischer Differentialgleichungen*. Teubner, Stuttgart, 1986.
10. V. E. Henson and P. S. Vassilevski. Element-free AMGe: general algorithms for computing interpolation weights in AMG. *SIAM J. Sci. Comput.*, 23(2):629–650, 2001.
11. V. V. Jikov, S. M. Kozlov, and O. A. Oleinik. *Homogenization of Differential Operators and Integral Functionals*. Springer-Verlag, Berlin, Heidelberg, New York, 1994.
12. J. E. Jones and P. S. Vassilevski. AMGe based on element agglomeration. *SIAM J. Sci. Comput.*, 23(1):109–133, 2001.
13. S. Knapek. Matrix-dependent multigrid-homogenization for diffusion problems. *SIAM J. Sci. Comput.*, 20:515–533, 1998.
14. R. H. Kraichnan. Diffusion by a random velocity field. *Phys. Fluids*, 13(1):22–31, 1970.
15. J. D. Moulton, J. E. Dendy, and J. M. Hyman. The black box multigrid numerical homogenization algorithm. *Journal of Computational Physics*, 141:1–29, 1998.
16. J. W. Ruge and K. Stüben. Efficient solution of finite difference and finite element equations by algebraic multigrid (AMG). In D. J. Paddon and H. Holstein, editors, *Multigrid Methods for Integral and Differential Equations*, The Institute of Mathematics and its Applications Conference Series, pages 169–212. Clarendon Press, Oxford, 1985.

17. J. W. Ruge and K. Stüben. Algebraic multigrid. In S. F. McCormick, editor, *Multigrid methods*, pages 73–130. SIAM, 1987.
18. C. Wagner, W. Kinzelbach, and G. Wittum. Schur-complement multigrid. A robust method for groundwater flow and transport problems. *Numer. Math.*, 75:523–545, 1997.
19. P. Wesseling. *An introduction to multigrid methods*. Wiley, Chichester, England, 1991.
20. G. Wittum. On the robustness of ILU smoothing. *SIAM J. Sci. Statist. Comput.*, 10:699–717, 1989.
21. P. M. de Zeeuw. Matrix-dependent prolongations and restrictions in a blackbox multigrid solver. *Journal of Computational and Applied Mathematics*, 33:1–27, 1990.

On the Modeling of Small Geometric Features in Computational Electromagnetics

Fredrik Edelvik

Division of Scientific Computing, Department of Information Technology, Uppsala University, P.O. Box 337, SE-75105 Uppsala, Sweden
fredrik.edelvik@it.uu.se

Summary. The ability to model features that are small relative to the cell size is often important in electromagnetic simulations. In this paper we develop subcell models for thin wires and thin slots in the finite-element time-domain (FETD) method. The current along the wires is described by a modified telegraphers equation and for the slots a dual equation for the magnetic current is used. Therefore, a unified approach for modeling thin wires and thin slots is possible. Stability proofs show that the full time-continuous field-wire-slot system is stable and that the fully discrete system is unconditionally stable. The proposed method is demonstrated for a dipole and a circular loop antenna, and scattering from a circular slot in an infinite, perfectly conducting wall.

Key words: Maxwell's equations, finite element methods, subcell models, thin wires, thin slots

1 Introduction

Transient finite-element methods based on Whitney elements represent powerful techniques for solution of the Maxwell equations, see e.g. [10] and references therein. The ability to model features that are small relative to the cell size is often important in electromagnetic simulations. In principle, an unstructured grid could be used to resolve these small features. However, in practice, the number of unknowns can be prohibitive. Thus, the development of accurate models that characterize the physics of the feature without the need for a highly resolved grid is essential. In this paper models for thin wires and thin slots and their incorporation in the finite-element time-domain (FETD) method are addressed.

Thin wires are often important parts of electromagnetic compatibility and antenna problems. A subcell model for thin wires in the finite-difference time-domain (FDTD) method using modified telegraphers equations has been developed by Holland et al. [8]. These equations are driven by the electric field along the wire, whereas the current on the wire is a source term in the Maxwell equations. In [14] this model is used for grid aligned wires in the FETD method. However, stability problems might occur due to the non-symmetric coupling between field and wires.

Practical systems possess narrow cracks and gaps that can be challenging to include in an analysis. Therefore, subcell modeling techniques have been proposed for thin slots. Riley has proposed a differential equation based method for modeling grid aligned thin slots in FETD [14], where both the slot width and the slot depth are parameters. The slot model is based on a dual formulation to thin wires. Since the coupling between field and slot in [14] is non-symmetric the resulting field-slot system might exhibit instabilities.

We show that a consistent discretization of the wire and slot equations using nodal basis functions and the use of a radial weighting function result in a symmetric spatial coupling between field and wire, and field and slot. We prove using the energy method that this yields a stable time-continuous field-wire-slot system and that the fully discrete field-wire-slot system is unconditionally stable if the second-order accurate Newmark–Beta scheme is used for time-discretization. Furthermore, neither the wires nor the slots have to follow edges in the volume grid. This gives considerable modeling flexibility when including these subcellular features in the simulations.

The outline of the rest of the paper is as follows: In the next section we introduce the equations. In the following section we show how to incorporate wires and slots in the FETD method, in particular how the coupling between field, and wires and slots is performed. In Sect. 4 we prove that the time-continuous field-wire-slot system is stable and that the fully discrete system is unconditionally stable. In the results section the proposed method is applied to a dipole and a circular loop antenna as well as scattering from a circular slot in an infinite, perfectly conducting wall. The summary and conclusions are given in Sect. 6.

2 Governing Equations

The Maxwell equations for linear, isotropic and non-dispersive media are given by

$$\frac{\partial \mathbf{B}}{\partial t} + \nabla \times \mathbf{E} = -\mathbf{J}_m \,, \tag{1}$$

$$\epsilon \frac{\partial \mathbf{E}}{\partial t} - \nabla \times \frac{1}{\mu}\mathbf{B} = -\sigma \mathbf{E} - \mathbf{J} \,, \tag{2}$$

where $\mathbf{E}$ is the electric field, $\mathbf{B}$ is the magnetic flux density, $\mathbf{J}$ is the electric current density, $\mathbf{J}_m$ is the magnetic current density, ϵ is the electric permittivity, μ is the magnetic permeability and σ is the electric conductivity.

To derive the wire equation we follow Holland et al. [8] and study an infinitely long cylinder of radius a running in the z-direction, see Fig. 1. To simplify the derivation we assume that $\sigma = 0$ in the neighborhood of the wire. In cylindrical coordinates with the assumption that the electromagnetic fields, E_r and H_θ, are proportional to $1/r$ close to a thin wire we obtain [8]

$$L \frac{\partial^2 I}{\partial t^2} + R \frac{\partial I}{\partial t} - \frac{L}{\mu\epsilon} \frac{\partial^2 I}{\partial z^2} = \frac{\partial E_z}{\partial t} + \frac{\partial \tilde{V}^{\mathrm{inc}}}{\partial t} \,, \tag{3}$$

where I is the wire current, $\tilde{V}^{\text{inc}}$ is a voltage source per unit length, R is the wire resistance per unit length and L is the wire inductance per unit length given by

$$L = \frac{\mu}{2\pi} \log \frac{r_0 + a}{2a}\,, \tag{4}$$

where r_0 is a grid dependent radial distance from the wire. It is defined as $r_0 = 1.7\Delta_{av}$, where Δ_{av} is an average edge length in the unstructured grid local to the wire, and $(r_0 + a)/2$ is an average distance from the wire to the surrounding electric fields used to drive the wire. Note that L is positive as long as $r_0 > a$, which is necessary in order to have a well posed problem. The current vanishes at an open termination, whereas the spatial derivative of the current vanishes when the wire terminates on a large perfect electric conductor.

The subcell slot model is based on a dual formulation to thin wires. A thin slot with length L, width w and depth d, in a wall is shown in Fig. 2. It is assumed that $L \gg w$, and w and d are electrically small. All fields in the wall are set to zero and the slot is modeled through the following equation [14] (see [16] for corresponding frequency-domain equations):

$$C_s \frac{\partial^2 V_s}{\partial t^2} - \frac{C_s}{\mu\epsilon} \frac{\partial^2 V_s}{\partial \xi^2} = \frac{\partial H_\xi^{\text{diff}}}{\partial t}\,. \tag{5}$$

where V_s is the magnetic current (voltage across slot width), H_ξ^{diff} is the difference in the ξ-component of the magnetic field on opposite sides of the slot wall ($\hat{\xi} = \hat{y}$ in Fig. 2), and C_s is the slot capacitance per unit length. It is given by

$$C_s = \frac{2}{\pi} \epsilon \log \frac{r_0 + a_s}{2a_s}\,, \tag{6}$$

where $a_s = \frac{w}{4} \exp\left(-\pi d/(2w)\right)$ is an equivalent antenna radius [15]. The magnetic current vanishes at the end-points of the slot.

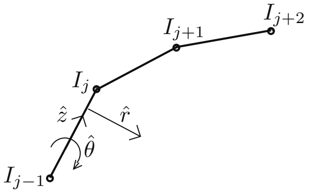


Fig. 1. Three segments of a discretized wire in cylindrical coordinates

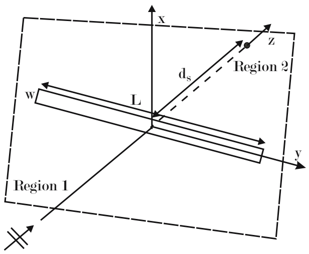


Fig. 2. Thin slot in a wall

3 Modeling Wires and Slots in FETD

Due to the fact that the source term in the slot equation is the difference in the magnetic field on opposite sides of the slot wall we have chosen to discretize the Maxwell equations on a tetrahedral grid using Edge/Facet Whitney elements [10]. It was pointed out by Bossavit et al. [1] that in solving Maxwell's equations, Whitney 1-forms (edge elements) should be used to approximate fields, whereas Whitney 2-forms (face elements) should be used to approximate fluxes. Therefore, the electric field, $\mathbf{E}$, is expanded in edge elements, $\boldsymbol{\phi}_j$, as

$$\mathbf{E} = \sum_j E_j \boldsymbol{\phi}_j \,, \tag{7}$$

where the unknowns are the circulation of the electric field along the edges. The magnetic flux density, $\mathbf{B}$, and the magnetic current density, $\mathbf{J}_m$, on the other hand are expanded in face elements, $\boldsymbol{\psi}_k$, as

$$\mathbf{B} = \sum_k B_k \, \boldsymbol{\psi}_k \,, \qquad \mathbf{J}_m = \sum_k J_{mk} \, \boldsymbol{\psi}_k \,. \tag{8}$$

where the unknowns are the fluxes across the facets.

The Whitney 1-forms and Whitney 2-forms are related such that

$$\nabla \times W^1 \subset W^2 \,, \tag{9}$$

where W^1 and W^2 are the vector spaces generated by the respective forms [1]. From the properties of Whitney 1-forms and 2-forms it follows that the curl of an edge element is a linear combination of the face elements whose faces contain that edge. Faraday's law can therefore be trivially discretized as

$$\frac{\mathrm{d}\boldsymbol{B}}{\mathrm{d}t} = -C\boldsymbol{E} - \boldsymbol{J}_m \,, \tag{10}$$

where C is the circulation matrix whose entries are zero or ± 1 and the vectors $\boldsymbol{B}$, $\boldsymbol{E}$ and $\boldsymbol{J}_m$ contain the expansion coefficients of $\mathbf{B}$, $\mathbf{E}$ and $\mathbf{J}_m$, respectively. Application of Galerkin's method for Ampère's law, where we multiply by $\boldsymbol{\phi}_k$ and integrate over the domain of interest, yields the weak formulation: Find $\mathbf{E} \in W^1$ and $\mathbf{B} \in W^2$ such that

$$\begin{aligned} \int_V \left(\epsilon \frac{\mathrm{d}\mathbf{E}}{\mathrm{d}t} + \sigma \mathbf{E} \right) \cdot \boldsymbol{\phi}_k \, \mathrm{d}V = & \int_V \frac{1}{\mu} \mathbf{B} \cdot (\nabla \times \boldsymbol{\phi}_k) \, \mathrm{d}V \\ & + \oint_\Gamma \left[\mathbf{n} \times \frac{1}{\mu} \mathbf{B} \right] \cdot \boldsymbol{\phi}_k \, \mathrm{d}\Gamma - \int_V \mathbf{J} \cdot \boldsymbol{\phi}_k \, \mathrm{d}V \,, \end{aligned} \tag{11}$$

$\forall \boldsymbol{\phi}_k \in W^1$, where Γ is the boundary of V. In matrix form (11) can be written as [18]

$$M_\epsilon \frac{\mathrm{d}\boldsymbol{E}}{\mathrm{d}t} + M_\sigma \boldsymbol{E} = C^T M_{\mu^{-1}} \boldsymbol{B} + \boldsymbol{f} \,, \tag{12}$$

where

$$\begin{aligned} M_\epsilon|_{jk} &= \int_V \epsilon \boldsymbol{\phi}_j \cdot \boldsymbol{\phi}_k \, \mathrm{d}V \,, \; M_\sigma|_{jk} = \int_V \sigma \boldsymbol{\phi}_j \cdot \boldsymbol{\phi}_k \, \mathrm{d}V \,, \\ M_{\mu^{-1}}|_{jk} &= \int_V \frac{1}{\mu} \boldsymbol{\psi}_j \cdot \boldsymbol{\psi}_k \, \mathrm{d}V \,, \; f_k = -\int_V \mathbf{J} \cdot \boldsymbol{\phi}_k \, \mathrm{d}V \,. \end{aligned} \tag{13}$$

It is interesting to note the close correspondence with the Finite Integration Technique (FIT) [17] in (10) and (12). These systems of equations are also obtained for FIT, where the mass matrices are diagonal operators on orthogonal hexahedral grids. If we now eliminate $\boldsymbol{B}$ from (12) using (10) we get

$$M_\epsilon \frac{\mathrm{d}^2 \boldsymbol{E}}{\mathrm{d}t^2} + M_\sigma \frac{\mathrm{d}\boldsymbol{E}}{\mathrm{d}t} + C^T M_{\mu^{-1}} C \boldsymbol{E} = -C^T M_{\mu^{-1}} \boldsymbol{J}_m + \frac{\partial \boldsymbol{f}}{\partial t} \,. \tag{14}$$

For constant or piecewise constant μ the $C^T M_{\mu^{-1}} C$ matrix is identically equal to the stiffness matrix

$$S_{jk} = \int_V \frac{1}{\mu} \left(\nabla \times \boldsymbol{\phi}_j \right) \cdot \left(\nabla \times \boldsymbol{\phi}_k \right) \mathrm{d}V \,, \tag{15}$$

which is obtained when the $\nabla \times \nabla \times$-operator is discretized using edge elements. This follows directly from property (9), which implies that the curl of the edge elements are related to the face elements through the matrix C^T. Substitution of this relationship into (15) yields the desired result. Note that the elimination of $\boldsymbol{B}$ is done with the aim of developing an unconditionally stable solver. An alternative would be to keep the coupled curl formulation and develop a conditionally stable solver, see e.g. [12] and references therein.

The current I along the wire can be expanded in basis functions as

$$I(z) = \sum_j I_j \varPhi_j(z) \,, \tag{16}$$

where $\varPhi_j$ is the standard linear nodal basis function in 1D, and I_j is the unknown current at wire node j. The current density $\mathbf{J}$ is now expressed as

$$\mathbf{J}(r, z) = I(z) \, g(r) \hat{z} = \sum_j I_j \varPhi_j(z) \, g(r) \hat{z} \,, \tag{17}$$

where r is the radial distance from the wire and $g(r)$ is a weighting function satisfying

$$\int_{r \geq a} g(r) \, 2\pi r \, dr = 1 \,, \tag{18}$$

and thus has dimension $\left[m^{-2} \right]$. Furthermore, it is important that this function decreases with r and equals zero for $r \geq r_0$, which gives a compact support. The function used in this paper is defined as

$$g(r) = \begin{cases} 0\,, & r < a\,, \\ \frac{1+\cos\left(\frac{\pi\, r}{r_0}\right)}{\pi\left(r_0^2 - a^2\right) + \frac{2r_0^2}{\pi}\left(-1-\cos\frac{\pi\, a}{r_0} - \frac{\pi\, a}{r_0}\sin\frac{\pi\, a}{r_0}\right)}\,, & a \le r \le r_0\,, \\ 0\,, & r > r_0\,. \end{cases} \tag{19}$$

If we multiply both sides of (3) by $g(r)\,\Phi_j(z)$ and integrate over all space we obtain [3]

$$M_w \frac{\mathrm{d}^2\boldsymbol{I}}{\mathrm{d}t^2} + M_R \frac{\mathrm{d}\boldsymbol{I}}{\mathrm{d}t} + S_w \boldsymbol{I} = P\,\frac{\mathrm{d}\boldsymbol{E}}{\mathrm{d}t} + \frac{\mathrm{d}\boldsymbol{V}^{\mathrm{inc}}}{\mathrm{d}t}\,, \tag{20}$$

where $\boldsymbol{I}$ is the vector of nodal current unknowns. The mass and stiffness matrices for the wire equation, the interpolation operator, and the voltage source are given by

$$M_w|_{jk} = \int_z L\,\Phi_j \Phi_k \,\mathrm{d}z\,, \tag{21}$$

$$M_R|_{jk} = \int_z R\,\Phi_j \Phi_k \,\mathrm{d}z\,, \tag{22}$$

$$S_w|_{jk} = \int_z \frac{L}{\epsilon\mu}\,\frac{\mathrm{d}\Phi_j}{\mathrm{d}z}\,\frac{\mathrm{d}\Phi_k}{\mathrm{d}z}\mathrm{d}z\,, \tag{23}$$

$$P_{jk} = \int_V \hat{z}\cdot\boldsymbol{\phi}_k\, g(r)\,\Phi_j(z)\,\mathrm{d}V\,, \tag{24}$$

$$V_j^{\mathrm{inc}} = \int_z \tilde{V}^{\mathrm{inc}}(z)\,\Phi_j(z)\,\mathrm{d}z\,. \tag{25}$$

By inserting (17) into the current density term (11) we obtain (cf. (24))

$$-\int_V \frac{\partial \mathbf{J}}{\partial t}\cdot\boldsymbol{\phi}_k\,\mathrm{d}V = -\sum_j \frac{\mathrm{d}I_j}{\mathrm{d}t}\int_V \hat{z}\cdot\boldsymbol{\phi}_k \Phi_j(z)\,g(r)\,\mathrm{d}V = -\sum_j P_{jk}\frac{\mathrm{d}I_j}{\mathrm{d}t}\,. \tag{26}$$

For the slots we proceed similarly and the slot voltage V is expanded in basis functions as

$$V_s(\xi) = \sum_j V_{s,j}\Phi_j(\xi)\,, \tag{27}$$

where Φ_j as before is the standard linear nodal basis function in 1D, and $V_{s,j}$ is the unknown voltage at slot node j. In addition we define another weighting function as $\tilde{g}(r) = \pm 2g(r)$ with a plus sign for region 2 and a minus sign for region 1, where the regions denote different sides of the slot wall (see Fig. 2).

Multiplying both sides of (5) by $g(r)\,\Phi_j(\xi)$ and integrating over the domain of interest yield [4]

$$M_s \frac{\mathrm{d}^2\boldsymbol{V}_s}{\mathrm{d}t^2} + S_s \boldsymbol{V}_s = P_s\,\frac{\mathrm{d}\boldsymbol{B}}{\mathrm{d}t} = P_s\,\left(-C\boldsymbol{E} - \boldsymbol{J}_m\right)\,, \tag{28}$$

where $\boldsymbol{V}_s$ is the vector of nodal voltage unknowns and (10) is inserted to get the last equality. The mass and stiffness matrices for the slot equation, and the interpolation operator are given by

$$M_s|_{jk} = \int_\xi C_s\, \varPhi_j \varPhi_k \,\mathrm{d}\xi\,, \tag{29}$$

$$S_s|_{jk} = \int_\xi \frac{C_s}{\mu\epsilon}\frac{\mathrm{d}\varPhi_j}{\mathrm{d}\xi}\frac{\mathrm{d}\varPhi_k}{\mathrm{d}\xi}\mathrm{d}\xi\,, \tag{30}$$

$$P_s|_{jk} = \int_V \frac{1}{\mu}\,\tilde{g}(r)\,\varPhi_j(\xi)\boldsymbol{\psi}_k\cdot\hat{\xi}\,\mathrm{d}V\,. \tag{31}$$

The magnetic current density is calculated from the slot voltage and is expressed as

$$\mathbf{J}_m(r,\xi) = V_s(\xi)\,\tilde{g}(r)\hat{\xi} = \sum_j V_{s,j}\varPhi_j(\xi)\,\tilde{g}(r)\hat{\xi}\,, \tag{32}$$

where r is the radial distance from the slot. The change of sign in $\mathbf{J}_m$ through $\tilde{g}(r)$ reflects the different normal directions on opposite sides of the slot wall. The two expressions (32) and (8) for $\mathbf{J}_m$ are put equal in a weak sense by multiplying both with $\boldsymbol{\psi}_i/\mu$ and integrating over the domain of interest. This results in

$$\int_V \frac{1}{\mu}\sum_j V_{s,j}\varPhi_j(\xi)\tilde{g}(r)\hat{\xi}\cdot\boldsymbol{\psi}_i\,\mathrm{d}V = \int_V \frac{1}{\mu}\sum_k J_{mk}\,\boldsymbol{\psi}_k\cdot\boldsymbol{\psi}_i\,\mathrm{d}V\,. \tag{33}$$

According to (13) and (31) this is on matrix form given by

$$P_s^T\boldsymbol{V}_s = M_{\mu^{-1}}\boldsymbol{J}_m\,, \tag{34}$$

which implies

$$\boldsymbol{J}_m = M_{\mu^{-1}}^{-1}P_s^T\boldsymbol{V}_s\,. \tag{35}$$

Inserting (35) and (26) in (14), together with (20) and (28) yield the following time-continuous field-wire-slot system to solve:

$$\begin{pmatrix} M_\epsilon & 0 & 0 \\ 0 & M_w & 0 \\ 0 & 0 & M_s \end{pmatrix}\begin{pmatrix} \ddot{\boldsymbol{E}} \\ \ddot{\boldsymbol{I}} \\ \ddot{\boldsymbol{V}}_s \end{pmatrix} + \begin{pmatrix} M_\sigma & P^T & 0 \\ -P & M_R & 0 \\ 0 & 0 & 0 \end{pmatrix}\begin{pmatrix} \dot{\boldsymbol{E}} \\ \dot{\boldsymbol{I}} \\ \dot{\boldsymbol{V}}_s \end{pmatrix}$$
$$+ \begin{pmatrix} S & 0 & C^TP_s^T \\ 0 & S_w & 0 \\ P_sC & 0 & S_s + P_sM_{\mu^{-1}}^{-1}P_s^T \end{pmatrix}\begin{pmatrix} \boldsymbol{E} \\ \boldsymbol{I} \\ \boldsymbol{V}_s \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ \dot{\boldsymbol{V}}^{\mathrm{inc}} \\ \mathbf{0} \end{pmatrix}\,. \tag{36}$$

For a more compact notation the time derivative of a vector $\boldsymbol{X}$ has been denoted $\dot{\boldsymbol{X}}$. The field-wire-slot system (36) is discretized in time by the second-order accurate, unconditionally stable Newmark–Beta scheme [9, 6]

$$\begin{pmatrix} A_e & 2\Delta tP^T & \Delta t^2C^TP_s^T \\ -2\Delta tP & A_w & 0 \\ \varDelta t^2P_sC & 0 & A_s \end{pmatrix}\begin{pmatrix} \boldsymbol{E}^{n+1} \\ \boldsymbol{I}^{n+1} \\ \boldsymbol{V}_s^{n+1} \end{pmatrix} = \begin{pmatrix} \boldsymbol{b} \\ \boldsymbol{b}_w \\ \boldsymbol{b}_s \end{pmatrix}\,, \tag{37}$$

where

$$A_e = 4M_\epsilon + 2\Delta t M_\sigma + \Delta t^2 S \,, \tag{38}$$

$$A_w = 4M_w + 2\Delta t M_R + \Delta t^2 S_w \,, \tag{39}$$

$$A_s = 4M_s + \Delta t^2 \left(S_s + P_s M_{\mu^{-1}}^{-1} P_s^T \right) \,, \tag{40}$$

$$\begin{aligned} \boldsymbol{b} &= \left(8M_\epsilon - 2\Delta t^2 S\right) \boldsymbol{E}^n - 2\Delta t^2 C^T P_s^T \boldsymbol{V}_s^n \\ &\quad - \left(4M_\epsilon - 2\Delta t M_\sigma + \Delta t^2 S\right) \boldsymbol{E}^{n-1} + 2\Delta t P^T \boldsymbol{I}^{n-1} - \Delta t^2 C^T P_s^T \boldsymbol{V}_s^{n-1} \,, \end{aligned} \tag{41}$$

$$\begin{aligned} \boldsymbol{b}_w &= 4\Delta t^2 \dot{\boldsymbol{V}}^{\mathrm{inc}}|_{t=n\Delta t} + \left(8M_w - 2\Delta t^2 S_w\right) \boldsymbol{I}^n - 2\Delta t P \boldsymbol{E}^{n-1} \\ &\quad - \left(4M_w - 2\Delta t M_R + \Delta t^2 S_w\right) \boldsymbol{I}^{n-1} \,, \end{aligned} \tag{42}$$

$$\begin{aligned} \boldsymbol{b}_s &= -2\Delta t^2 P_s C \boldsymbol{E}^n + \left(8M_s - 2\Delta t^2 \left(S_s + P_s M_{\mu^{-1}}^{-1} P_s^T\right)\right) \boldsymbol{V}_s^n \\ &\quad - \Delta t^2 P_s C \boldsymbol{E}^{n-1} - \left(4M_s + \Delta t^2 \left(S_s + P_s M_{\mu^{-1}}^{-1} P_s^T\right)\right) \boldsymbol{V}_s^{n-1} \,. \end{aligned} \tag{43}$$

To solve (37) we use the fact that A_w is a tridiagonal matrix (or tridiagonal circulant matrix for a wire loop). Hence, the wire part of the system can be solved as

$$\boldsymbol{I}^{n+1} = A_w^{-1} \left(\boldsymbol{b}_w + 2\Delta t P \boldsymbol{E}^{n+1}\right) \,. \tag{44}$$

By substituting this into (37) we obtain

$$\begin{pmatrix} A_e + 4\Delta t^2 P^T A_w^{-1} P & \Delta t^2 C^T P_s^T \\ \Delta t^2 P_s C & A_s \end{pmatrix} \begin{pmatrix} \boldsymbol{E}^{n+1} \\ \boldsymbol{V}_s^{n+1} \end{pmatrix} = \begin{pmatrix} \boldsymbol{b} - 2\Delta t P^T A_w^{-1} \boldsymbol{b}_w \\ \boldsymbol{b}_s \end{pmatrix} \,. \tag{45}$$

The symmetry and positive definiteness of the matrix in (45) can now be easily verified through the symmetry and definiteness properties of the respective mass and stiffness matrices. Hence, we can apply a preconditioned conjugate gradient (PCG) method to solve (45) at each time step. An incomplete Cholesky factorization is used for preconditioning. The wire currents are finally given by (44).

The operator in (24) is used to calculate the electric field along the wire at each wire node. Due to the compact support of the nodal basis function $\varPhi_j$ and the weighting function g only edge projected fields in a neighborhood of the wire node contributes. To be more specific, each wire node is surrounded by an interpolation cylinder of radius r_0 and length equal to the sum of the two wire beams sharing the node. The integral in (24) is calculated using a sixth-degree Gaussian quadrature for tetrahedral elements [11]. In order to have a smooth interpolation cylinder following the wire some small modifications are necessary for bent wires as explained in detail in [3]. Furthermore, in certain cases the interpolation radius might have to be reduced to avoid that other parts of the geometry fall within the interpolation cylinder.

The interpolation between field and slots is almost identical to the interpolation between field and wires. The only differences are that the interpolation involves face elements for slots but edge elements for wires and the change of sign in the radial weighting function, $\tilde{g}$, on opposite sides of the slot wall.

The extra memory requirements for wires and slots include the storage of the wire current and slot voltage unknowns, their mass and stiffness matrices, and the interpolation operators P and P_s. Since the wires and slots are 1D features and the interpolation operators are compact and only nonzero close to wires and slots it follows that the extra memory requirements are negligible compared to the field matrices. The application of these operators requires for the same reason a small amount of extra arithmetic operations. What may look like a significant extra cost is the solution of (35). However, since $P^T\boldsymbol{V}$ is only nonzero close to the slot it is therefore not necessary to solve this equation using the full $M_{\mu^{-1}}$ matrix. Instead this equation is solved iteratively using a PCG method for a shrunk $M_{\mu^{-1}}$ matrix including only the rows and columns affected by the slot. Hence, the extra work for including wires and slots is in general small.

4 Stability Analysis

In this section we analyze the stability of the time-continuous field-wire-slot system (36) and the fully discrete field-wire-slot system (37). The matrices M_ϵ, M_w, M_s, $S(=C^T M_{\mu^{-1}} C)$, S_w, and S_s are all symmetric, M_ϵ, M_w and M_s are positive definite, whereas S, S_w and S_s are positive semi-definite. For simplicity we put $\sigma = 0$ and $R = 0$, which implies that $M_\sigma = 0$ and $M_R = 0$. A nonzero σ and/or R would mean that we have a loss of electromagnetic energy and this is straightforward to include in the proofs. Source terms do not effect stability [7] and are therefore not included in the analysis.

Let $S_w = GG^T, S_s = FF^T$, $\dot{\boldsymbol{B}} = -C\boldsymbol{E} - M_{\mu^{-1}}^{-1} P_s^T \boldsymbol{V}_s$, $\dot{\boldsymbol{V}} = G^T\boldsymbol{I}$ and $\dot{\boldsymbol{I}}_s = F^T\boldsymbol{V}_s$, and define the block-matrices

$$M = \begin{pmatrix} M_\epsilon & 0 & 0 & 0 & 0 & 0 \\ 0 & M_{\mu^{-1}} & 0 & 0 & 0 & 0 \\ 0 & 0 & M_w & 0 & 0 & 0 \\ 0 & 0 & 0 & I & 0 & 0 \\ 0 & 0 & 0 & 0 & M_s & 0 \\ 0 & 0 & 0 & 0 & 0 & I \end{pmatrix} \tag{46}$$

$$Q = \begin{pmatrix} 0 & -C^T M_{\mu^{-1}} & P^T & 0 & 0 & 0 \\ M_{\mu^{-1}} C & 0 & 0 & 0 & P_s^T & 0 \\ -P & 0 & 0 & G & 0 & 0 \\ 0 & 0 & -G^T & 0 & 0 & 0 \\ 0 & -P_s & 0 & 0 & 0 & F \\ 0 & 0 & 0 & 0 & -F^T & 0 \end{pmatrix} \tag{47}$$

and the row vector $\boldsymbol{W}^T = (\boldsymbol{E}^T\ \boldsymbol{B}^T\ \boldsymbol{I}^T\ \boldsymbol{V}^T\ \boldsymbol{V}_s^T\ \boldsymbol{I}_s^T)$. The total electromagnetic energy of the field-wire-slot system is then defined by

$$\mathcal{E}(t) = \frac{1}{2}\dot{\boldsymbol{W}}^T M \dot{\boldsymbol{W}}\,. \tag{48}$$

Then we have

Theorem 1. *The time-continuous field-wire-slot system (36) is stable in the following sense: The energy $\mathcal{E}(t)$ in (48) is preserved.*

Proof. By using the newly defined vectors and matrices we can write the system (36) as

$$M\ddot{\boldsymbol{W}} + Q\dot{\boldsymbol{W}} = 0\,. \tag{49}$$

Since the Q-matrix is skew-symmetric multiplying from the left by $\dot{\boldsymbol{W}}^T$ yields

$$\dot{\boldsymbol{W}}^T M\ddot{\boldsymbol{W}} = 0\,. \tag{50}$$

Hence, from (50) it follows that

$$\frac{\mathrm{d}\mathcal{E}}{\mathrm{d}t} = 0\,, \tag{51}$$

which implies that the total electromagnetic energy is preserved and the time-continuous problem is stable. □

For the fully discrete case we let $S_w = GG^T$, $S_s = FF^T$, $\boldsymbol{B}^n = -C\boldsymbol{E}^n - M_{\mu^{-1}}^{-1}P_s^T\boldsymbol{V}_s^n$, $\boldsymbol{V}^n = G^T\boldsymbol{I}^n$ and $\boldsymbol{I}_s^n = F^T\boldsymbol{V}_s^n$ $\forall n$. The following operators applied to a vector $\boldsymbol{X}$ are also used for a more compact notation:

$$\begin{aligned} \delta_+\boldsymbol{X}^n &= \frac{\boldsymbol{X}^{n+1} - \boldsymbol{X}^n}{\Delta t}\,,\ \delta_-\boldsymbol{X}^n = \frac{\boldsymbol{X}^n - \boldsymbol{X}^{n-1}}{\Delta t}\,,\\ \mu_+\boldsymbol{X}^n &= \frac{\boldsymbol{X}^{n+1} + \boldsymbol{X}^n}{2}\,,\ \mu_-\boldsymbol{X}^n = \frac{\boldsymbol{X}^n + \boldsymbol{X}^{n-1}}{2}\,. \end{aligned} \tag{52}$$

The total electromagnetic energy at time step $n+1$ is given by

$$\begin{aligned} \mathcal{E}^{n+1} = \frac{1}{2}\Big(&(\delta_+\boldsymbol{E}^n)^T\ M_\epsilon\ (\delta_+\boldsymbol{E}^n) + (\mu_+\boldsymbol{B}^n)^T\ M_{\mu^{-1}}\ (\mu_+\boldsymbol{B}^n)\\ &+ (\delta_+\boldsymbol{I}^n)^T\ M_w\ (\delta_+\boldsymbol{I}^n) + (\mu_+\boldsymbol{V}^n)^T\ (\mu_+\boldsymbol{V}^n)\\ &+ (\delta_+\boldsymbol{V}_s^n)^T\ M_s\ (\delta_+\boldsymbol{V}_s^n) + (\mu_+\boldsymbol{I}_s^n)^T\ (\mu_+\boldsymbol{I}_s^n)\Big)\ . \end{aligned} \tag{53}$$

For the discrete energy we have

Theorem 2. *The fully discrete field-wire-slot system (37) is unconditionally stable in the following sense:*

$$\mathcal{E}^{n+1} = \mathcal{E}^n,\ n = 0, 1, \ldots.$$

Proof. We will use a similar strategy as in the time-continuous case to prove that the fully discrete system is stable. Using the newly defined variables and operators we can rewrite the system (37) as

$$M \begin{pmatrix} \frac{\delta_+ - \delta_-}{\Delta t} \boldsymbol{E}^n \\ \frac{\mu_+ - \mu_-}{\Delta t} \boldsymbol{B}^n \\ \frac{\delta_+ - \delta_-}{\Delta t} \boldsymbol{I}^n \\ \frac{\mu_+ - \mu_-}{\Delta t} \boldsymbol{V}^n \\ \frac{\delta_+ - \delta_-}{\Delta t} \boldsymbol{V}_s^n \\ \frac{\mu_+ - \mu_-}{\Delta t} \boldsymbol{I}_s^n \end{pmatrix} + Q \begin{pmatrix} \frac{\delta_+ + \delta_-}{2} \boldsymbol{E}^n \\ \frac{\mu_+ + \mu_-}{2} \boldsymbol{B}^n \\ \frac{\delta_+ + \delta_-}{2} \boldsymbol{I}^n \\ \frac{\mu_+ + \mu_-}{2} \boldsymbol{V}^n \\ \frac{\delta_+ + \delta_-}{2} \boldsymbol{V}_s^n \\ \frac{\mu_+ + \mu_-}{2} \boldsymbol{I}_s^n \end{pmatrix} = \boldsymbol{0} \,. \tag{54}$$

We proceed in exactly the same manner as in the time-continuous case and multiply from the left by the vector in the second part, which yields

$$\begin{pmatrix} \frac{\delta_+ + \delta_-}{2} \boldsymbol{E}^n \\ \frac{\mu_+ + \mu_-}{2} \boldsymbol{B}^n \\ \frac{\delta_+ + \delta_-}{2} \boldsymbol{I}^n \\ \frac{\mu_+ + \mu_-}{2} \boldsymbol{V}^n \\ \frac{\delta_+ + \delta_-}{2} \boldsymbol{V}_s^n \\ \frac{\mu_+ + \mu_-}{2} \boldsymbol{I}_s^n \end{pmatrix}^T M \begin{pmatrix} \frac{\delta_+ - \delta_-}{\Delta t} \boldsymbol{E}^n \\ \frac{\mu_+ - \mu_-}{\Delta t} \boldsymbol{B}^n \\ \frac{\delta_+ - \delta_-}{\Delta t} \boldsymbol{I}^n \\ \frac{\mu_+ - \mu_-}{\Delta t} \boldsymbol{V}^n \\ \frac{\delta_+ - \delta_-}{\Delta t} \boldsymbol{V}_s^n \\ \frac{\mu_+ - \mu_-}{\Delta t} \boldsymbol{I}_s^n \end{pmatrix} = \frac{\mathcal{E}^{n+1} - \mathcal{E}^n}{\Delta t} = 0 \,, \tag{55}$$

due to the skew-symmetry of the Q-matrix and the definition of the fully discrete energy in (53). Hence, $\mathcal{E}^{n+1} = \mathcal{E}^n = \ldots = \mathcal{E}^0$, and our fully discrete system is unconditionally stable. □

Note that the key property for the stability of the field-wire-slot system is that we have a symmetric coupling between field and wire, and field and slot.

5 Numerical Results

In this section we apply the proposed methods to a few cases where measurements, analytical results or numerical results obtained by other methods are available for comparison. All simulations are performed using a hybrid FDTD-FETD solver[2]. Generation of the plane waves using Huygens' surfaces as well as the truncation of the grids using the U-PML [5] absorbing boundary condition are therefore performed in the FDTD region. The wires and slots are entirely located in the unstructured FETD region.

5.1 Transmitting Dipole Antenna

In this section we simulate a dipole antenna in transmitting mode. The dipole is 20 cm long and its radius is $0.05\,\mu$m. It is discretized using 40 wire segments and located arbitrarily in an unstructured grid with average edge length $\Delta = 5.8\,$mm. A few layers of Cartesian cells with edge lengths $\Delta = 5\,\text{mm}$ surround the unstructured grid. The antenna is excited at the midpoint using a voltage source. We register the current at the midpoint and calculate the input impedance and input admittance as

$$Z_{21}(f) = \frac{\hat{V}_{21}^{\text{inc}}(f)}{\hat{I}_{21}(f)}\,, \qquad Y_{21}(f) = \frac{1}{Z_{21}(f)}\,. \tag{56}$$

The real and imaginary parts of the impedance are resistance and reactance, respectively. For the admittance they are conductance and susceptance, respectively. In Figs. 3 and 4 we compare the input resistance and input conductance with results obtained by the method of moments (MoM) solver NEC. NEC is considered to be state-of-the art for thin wires and is therefore suitable for validation of our thin wire method. As seen in the figures a very good agreement is obtained. The frequency and resistance at half wavelength resonance are given in Table 1.

Table 1. The frequency and resistance at half wavelength resonance for a 20 cm long dipole antenna

	f [MHz]	$\Re e(Z)$ [Ω]
FETD	736.9	71.9
NEC	735.0	72.0
Theory	750.0	73.0

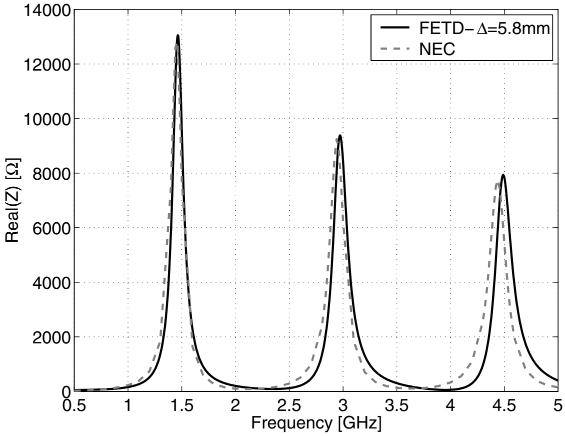


Fig. 3. The input resistance for a 20 cm dipole antenna

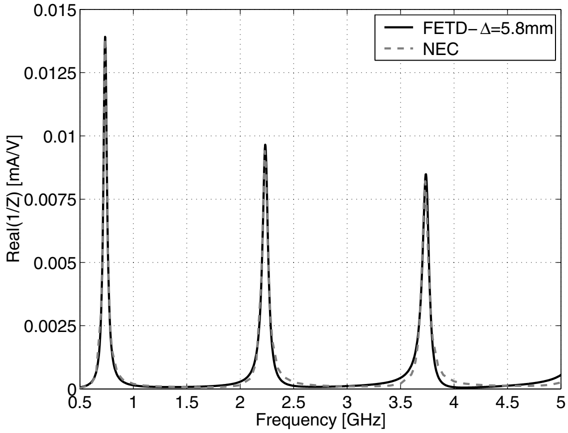


Fig. 4. The input conductance for a 20 cm dipole antenna

5.2 Transmitting Circular Loop Antenna

The proposed wire algorithm is not limited to straight wires and therefore this section is devoted to simulation of a circular loop antenna in transmitting mode. The loop diameter is 1 m and the wire radius is 1 mm. It is discretized using 50 wire segments and located arbitrary in an unstructured grid with average edge length $\Delta = 7.7\,\text{cm}$. The surrounding Cartesian cells have edge lengths $\Delta = 6.25\,\text{cm}$. One of the wire nodes is excited with a voltage source with the shape of a differentiated Gaussian pulse. We register the current at this particular wire node and calculate the input resistance and input conductance as in (56). In Figs. 5 and 6 we compare with results obtained by NEC. The frequency and resistance at half wavelength resonance are 98.8 MHz and 133.4 Ω for FETD, and 98.85 MHz and 137.2 Ω for NEC. The overall correspondence is good also in this case.

5.3 Scattering from a Circular Slot in an Infinite PEC Wall

Scattering from a circular slot in an infinite, perfectly electric conducting (PEC) wall has been studied in [13]. The slot has radius 5 cm, width 0.5 mm and no depth. It is discretized with 60 straight segments of length $\Delta\xi \approx 5.15\,\text{mm}$ that are not aligned with the unstructured grid. The unstructured grid has average edge length $\Delta = 6.4\,\text{mm}$ and the surrounding Cartesian cells have edge lengths $\Delta = 5\,\text{mm}$. A finer grid where the volume grid as well as the slot are refined with a factor two is also used. A normally incident plane wave illuminates the slot wall and the frequency response in the shadow region ($d_s = 5\,\text{cm}$ in Fig. 2) is shown in Fig. 7. The resonance at about 1 GHz corresponds to the first half wavelength resonance of the circular slot. The agreement with the experimental data is remarkably good for both grids and clearly shows that our proposed algorithm is able to model curved slots accurately. The small differences between the measured and numerical results are very similar to the ones observed for the integral equations solutions in [13].

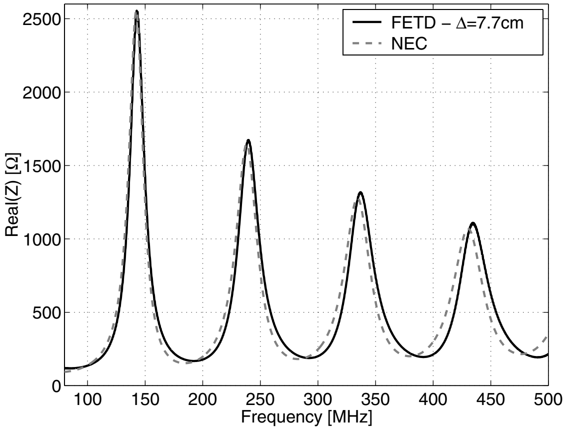


Fig. 5. The input resistance for a circular loop antenna

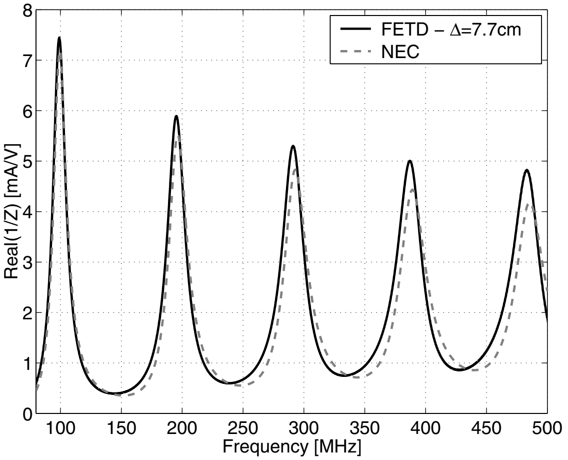


Fig. 6. The input conductance for a circular loop antenna

6 Conclusions

We have presented stable subcell models for accurate modeling of thin wires and slots in FETD. The slot model being a dual formulation to thin wires makes a unified treatment of slots and wires possible. Allowing the wires and slots to run arbitrarily and not aligned with the grid edges give considerable modeling flexibility when including these subcellular structures in the simulations. Traditionally subcell slot models have suffered from instability problem. A symmetric coupling between field and wires, and field and slots, makes it possible to prove that our resulting time-continuous field-wire-slot system is stable. Using a Newmark–Beta scheme for time discretization a similar proof shows that the fully discrete system is unconditionally

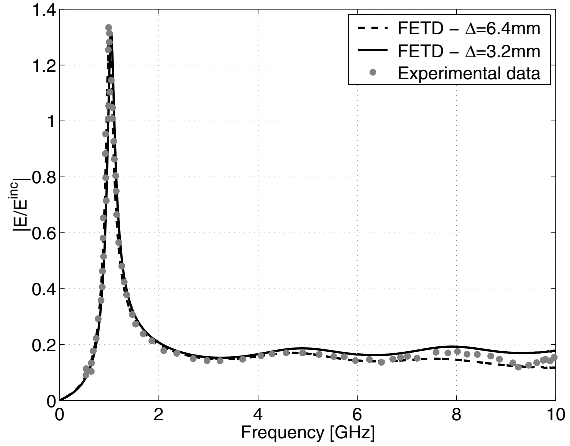


Fig. 7. Frequency response of the electric field on the shadow side of a PEC wall with a circular slot for a normally incident plane wave

stable. Results of good accuracy have been presented for a dipole and a circular loop antenna as well for scattering from a circular slot in a conducting wall.

7 Acknowledgments

The author gratefully acknowledge helpful assistance from Dr. D. J. Riley, Northrop Grumman Corp., Albuquerque, USA. Financial support for this work has been obtained from the Swedish Foundation for Strategic Research.

References

1. A. Bossavit and I. Mayergoyz. Edge elements for scattering problems. *IEEE Trans. Magn.*, 25(4):2816–2821, July 1989.
2. F. Edelvik. *Hybrid Solvers for the Maxwell Equations in Time-Domain.* Ph.D. thesis, Department of Information Technology, Uppsala University, September 2002.
3. F. Edelvik, G. Ledfelt, P. Lötstedt, and D. J. Riley. An unconditionally stable subcell model for arbitrarily oriented thin wires in the FETD method. *IEEE Trans. Antennas Propagat.*, 51(8):1797–1805, August 2003.
4. F. Edelvik and T. Weiland. Stable modeling of arbitrary thin slots in the finite-element time-domain method. *Int. Journal of Numerical Modelling*, 17(4):363–383, July/August 2004.
5. S. D. Gedney. An anisotropic PML absorbing media for the FDTD simulation of fields in lossy and dispersive media. *Electromagnetics*, 16(4):399–415, 1996.
6. S. D. Gedney and U. Navsariwala. An unconditionally stable implicit finite-element time-domain solution of the vector wave equation. *IEEE Microwave and Guided Wave Letters*, 5:332–334, 1995.

7. B. Gustafsson, H.-O. Kreiss, and J. Oliger. *Time Dependent Problems and Difference Methods*. Wiley-Interscience, New York, 1995.
8. R. Holland and L. Simpson. Finite-difference analysis of EMP coupling to thin struts and wires. *IEEE Trans. Electromagn. Compat.*, EMC-23(2):88–97, May 1981.
9. T. J. R. Hughes. *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*. Prentice-Hall, Englewood Cliffs, NJ, 1987.
10. J. Jin. *The Finite Element Method in Electromagnetics*. Wiley, New York, 2nd edition, 2002.
11. P. Keast. Moderate-degree tetrahedral quadrature formulas. *Computer Methods in Applied Mechanics and Engineering*, 55:339–348, 1986.
12. J.-F. Lee, R. Lee, and A. Cangellaris. Time-domain finite-element methods. *IEEE Trans. Antennas Propagat.*, 45(3):430–442, March 1997.
13. E. K. Reed and C. M. Butler. Time-domain electromagnetic penetration through arbitrarily shaped narrow slots in conducting screens. *IEEE Trans. Electromagn. Compat.*, 34(3):161–172, August 1992.
14. D. J. Riley. Transient finite-elements for computational electromagnetics: Hybridization with finite differences, modeling thin wires and thin slots, and parallel processing. In *17th Annual Review of Progress in Applied Computational Electromagnetics*, pages 128–138, Monterey, CA, March 2001.
15. L. K. Warne and K. C. Chen. Equivalent antenna radius for narrow slot apertures having depth. *IEEE Trans. Antennas Propagat.*, 37(7):824–834, July 1989.
16. L. K. Warne and K. C. Chen. A simple transmission line model for narrow slot apertures having depth and losses. *IEEE Trans. Electromagn. Compat.*, 34(3):173–182, August 1992.
17. T. Weiland. Time domain electromagnetic field computations with finite difference methods. *Int. Journal of Numerical Modelling*, 9(4):295–319, July 1996.
18. M.-F. Wong, O. Picon, and V. F. Hanna. A finite element method based on Whitney forms to solve Maxwell equations in time domain. *IEEE Trans. Magn.*, 31(3):1618–1621, May 1995.

Coupling PDEs and SDEs: The Illustrative Example of the Multiscale Simulation of Viscoelastic Flows

Benjamin Jourdain, Claude Le Bris, and Tony Lelièvre

CERMICS, Ecole Nationale des Ponts et Chaussées, 6 & 8 Av. Pascal, 77455
Champs-sur-Marne, France
{jourdain,lebris,lelievre}@cermics.enpc.fr

Summary. We present an overview of models which couple a partial differential equation with a stochastic differential equation posed at each point of the physical space. Such systems in particular arise in multiscale models of complex fluids, but also in the modeling of emission and transport of photons for example. For each case, we mention the mathematical and numerical issues and indicate the main results obtained so far.

Key words: multiscale models, coupled systems, stochastic differential equation, partial differential equation, Fokker–Planck equation, particle systems, Monte Carlo method

1 A Prototypical System

We would like to address here various mathematical and foremost numerical issues raised by the simulation of systems featuring a Partial Differential Equation (PDE) together with a Stochastic Differential Equation (SDE). For such a class of systems, that we henceforth called *hybrid* systems, we choose as a prototypical system the following one:

$$\begin{cases} \dfrac{\partial u}{\partial t}(t,y) - \dfrac{\partial^2 u}{\partial y^2}(t,y) = \dfrac{\partial f}{\partial y}(t,y) \\ \forall\, y, \quad \begin{cases} f(t,y) = \mathbb{E}\big(\varphi(X_t(y))\big) \\ \mathrm{d}X_t(y) = g(u(t,y), X_t(y), t)\,\mathrm{d}t \,+\, \sigma(t, X_t(y))\,\mathrm{d}W_t \end{cases} \end{cases} \tag{1}$$

Here, the PDE of the first line is supposed to hold, say, for the space-variable y varying in a one-dimensional interval $[0,L]$, while time t varies from 0 to T. With respect to the unknown scalar field u, it is of the form of the heat equation, with a right-hand side somehow unusual, though. For any $y \in [0,L]$, we then have the last two lines of (1). The second line rules the coupling between the PDE and the SDE: the solution $X_t(y)$ (varying in $\mathbb{R}$) of the SDE is used to evaluate an expectation value which provides the PDE with a right-hand side (that is a force term). The last line consists in an SDE, that is parameterized in y, and by the solution $u(t, y)$ of the PDE. The data are the functions φ, g, σ. System (1) is at this stage formulated somewhat vaguely, but the mathematical sense of the PDE and the SDE can be made precise, as well as the regularity of the data involved, and the initial conditions (1) is supplied with. The reason why such a system is not only a toy-system convenient for an expository survey, but meaningful and relevant from the application viewpoint will be made clear below.

The main feature we wish to already emphasize and discuss is the nature of system (1). For this purpose, let us at once mention that such a system stands at the intersection of various families

- that of systems coupling a *continuous description* with a *discrete description*, as is the case for instance when coupling a PDE and an Ordinary Differential Equation (ODE): a case of interest is e.g. that where the method of characteristics is used in addition to, or in replacement of, the solution of an advection equation; the same could apply to the use of particle methods; from the physical standpoint, the same could also apply to systems coupling different physical modelings, as is the case when an atomistic description of matter is coupled to a continuum description in material sciences;
- that of systems coupling *deterministic techniques* with *Monte Carlo type techniques* for solving one, or many, PDE(s); here we could have chosen to replace the third line of (1) by the associated Fokker–Planck equation, since what is only needed is the law of X_t to compute $f(t, y)$, and nothing else, but for computational purposes in the high dimensional case, we have preferred the simulation of the SDE;
- that of systems *coupling different scales*, where the effective coefficients involved in one equation are computed from another one, like is the case when homogenization techniques, or more generally averaging techniques, are resorted to: here the right-hand side f can be thought of as the averaged response of a finer scale (described by the internal variable X_t) subject to the solicitation $u(t, y)$.

The above problem is in some sense a superposition of all the previous contexts: it is a hybrid system in the continuous/discrete sense, in the deterministic/stochastic sense, in the multiscale sense. In a somewhat provocative way, we could tentatively say that system (1) is a *multiphysics, multimathematics, multiscale* system !

As we have just underlined the similarity with various classes of systems, let us now mention what system (1) is *not*:

- there are situations when a PDE and a SDE are simulated separately on different domains, and the coupling only holds in terms of boundary or compatibility conditions at the common interface (see [5] e.g.); such a coupling often holds for computational purposes (solving a PDE rather than a SDE can be cheaper on one zone, while the converse might be true on another zone); this is not the case here as one SDE holds at each point where the PDE is set;
- the deterministic equation and the stochastic equation can be coupled through the time variable, as is the case for mixed ODE/SDE systems in chemical kinetics,

e.g.; here we assume that the time variable is alike in the two equations, and that the difference of scales lies in the space variable;

- in some context, the stochastic nature comes as a perturbation of a deterministic equation, as is the case for a PDE with stochastic coefficients (see e.g. the Stochastic Navier–Stokes equation [36]); here *two* equations are at play.

Each of the above family of systems could equally justify a work in the spirit of the present one, but this is not our aim here.

In the following, we shall as announced concentrate on system (1). Our interest in such a system originates from a particular context, that of the simulation of polymeric fluid flows, which we shall introduce in the next section. We will review there the main results on the mathematical analysis and numerical analysis that are available in the literature to date, and mention some implementation issues. In doing this, as this is the main purpose of the present article, we will as much as possible try to emphasize the general facts and trends that seem to us to be valid outside the necessarily limited scope of the context under examination. Next, in Sect. 3, we shall see other situations, still in the general context of fluid flow simulations, where systems in the spirit of (1) are relevant. Sect. 4 will aim at showing one example, in a context far from fluid mechanics, also involving systems of the same type as (1).

Let us conclude this introductory section by emphasizing that simulating a hybrid system such as (1) of course requires up-to-date techniques for either of the two equations, for the PDE on the one hand, and for the SDE on the other hand. Our goal is not to present a state-of-the-art survey of either class of techniques separately, but rather to see how some representative techniques of either category interact with the other camp. Nevertheless, while our main focus is the *back and forth* interaction between the two equations, we shall allow us to review also some works when the SDE can be considered as parameterized by the solution of the PDE, the latter being considered as known (see Sects. 2.2 and 3.3).

2 Modeling Dilute Solutions of Flexible Polymers

The numerical simulation of incompressible viscous non-Newtonian fluids typically requires the simulation of systems of the type

$$\left\{\begin{aligned} \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\,\mathbf{u} - \Delta\mathbf{u} + \nabla p - \operatorname{div}\boldsymbol{\tau}_p &= \boldsymbol{f}_{\text{ext}}, \\ \operatorname{div}\mathbf{u} &= 0 \\ \frac{D\boldsymbol{\tau}_p}{Dt} &= \mathcal{G}(\boldsymbol{\tau}_p, \nabla\mathbf{u}), \end{aligned}\right. \qquad (2)$$

where the first line is the equation of conservation of momentum, the second one translates the incompressibility constraint and the third one is a differential equation ruling the evolution of the non-Newtonian part $\boldsymbol{\tau}_p$ of the stress tensor. In the above equations, $\mathbf{u}$ of course stands for the velocity of the fluid, p for its pressure, while $\boldsymbol{f}_{\text{ext}}$ is some external force. On purpose, we have omitted in the above system (and we will continue to do so throughout this article) all the physical parameters and constants, setting them to unity. The third line is often called a *constitutive law* or a *closure equation* and aims at providing a closed relation between the stress $\boldsymbol{\tau}_p$ and the velocity field $\mathbf{u}$: there, $\frac{D}{Dt}$ stands for a convective derivative, while the right-hand side $\mathcal{G}$ symbolically stands for an intricate function of the fields involved. One of the most famous instance of such a system is that for Oldroyd-B fluids, where the third equation precisely reads

$$\boldsymbol{\tau}_p + \frac{\partial \boldsymbol{\tau}_p}{\partial t} + \mathbf{u} \cdot \nabla \boldsymbol{\tau}_p - \boldsymbol{\tau}_p (\nabla \mathbf{u})^T - \nabla \mathbf{u}\, \boldsymbol{\tau}_p = \nabla \mathbf{u} + \nabla \mathbf{u}^T. \tag{3}$$

Alternately, one can replace the differential form of the third line of (2) by an equation in the integral form. We refer to [27, 43] for a general introduction to the mechanical context and the standard numerical tools to simulate systems of the form (2).

The well established commonly used strategy in fluid mechanics consists in derivating on the basis of mechanical arguments adequate differential (or integral) equations, i.e. forms for $\mathcal{G}$, and next solving system (2). Apart from this mainstream, an emerging field in non-Newtonian fluid mechanics, still mostly unexplored from the standpoint of mathematical analysis, was born in the early 1990s. It relies upon the introduction of a kinetic description of the fluid, at a finer scale, with a view to modeling the very phenomena from which the non-Newtonian feature of the fluid stems. A successful instance of this alternative track concerns the modeling of polymeric fluids. For such fluids, the key issue is to adequately simulate the evolution of the microstructures present at each macroscopic point of the fluid flow, that is the evolution of the polymeric chains wiggling in the fluid. A complete theory, initiated by the works of Doi and Edwards has given rise to a numerical approach, the so-called *micromacro* approach: see the reference treatises [14], [13], [3, 4] for the physical background, [42], [43] for the simulation techniques, and the recent review article [28]. The idea is to keep the first two equations of (2), but replace the third line of (2), i.e. the effective description of the evolution of $\boldsymbol{\tau}_p$, by the following two-step procedure: the expression of $\boldsymbol{\tau}_p$ reads

$$\boldsymbol{\tau}_p(t, \mathbf{x}) = \int (\mathbf{r} \otimes \boldsymbol{F}(\mathbf{r}))\, \psi(t, \mathbf{x}, \mathbf{r})\, \mathrm{d}\mathbf{r} \tag{4}$$

and is an averaged response of all the possible configurations of a representative polymer chain subject to the constraints in the flow, the latter being described by the Fokker–Planck equation

$$\begin{aligned} &\frac{\partial \psi(t, \mathbf{x}, \mathbf{r})}{\partial t} + \mathbf{u} \cdot \nabla_{\mathbf{x}} \psi(t, \mathbf{x}, \mathbf{r}) \\ &= -\mathrm{div}_{\mathbf{r}} \left((\nabla_{\mathbf{x}}\, \mathbf{u}\, \mathbf{r} - \boldsymbol{F}(\mathbf{r})) \psi(t, \mathbf{x}, \mathbf{r}) \right) + \frac{1}{2} \Delta_{\mathbf{r}} \psi(t, \mathbf{x}, \mathbf{r}). \end{aligned} \tag{5}$$

The distribution function $\psi(t, \mathbf{x}, \mathbf{r})$ describes the probability to find at time t (in $[0, T]$), and at the macro point $\mathbf{x}$ (in the computational domain $\mathcal{D}$), the polymer chain in the configuration $\mathbf{r}$, the latter variable typically varying in $\mathbb{R}^N$. Equation (5) will be considered henceforth as a prototypical Fokker–Planck type equation. Indeed analogous equations, more involved technically but of the same type formally, would hold when the configuration of the polymer chain is more in details described in a configuration space $\mathbb{R}^N$ of large dimension. Equation (5) is here written in the dumbbell case (see Sect. 2.1) for which N is equal to the dimension of the ambient space $\mathcal{D}$, i.e. 2 or 3. As N might be very large, depending on the degree of accuracy employed to describe the configuration of the chain, the simulation of the partial differential equation (5) in ψ might not be tractable numerically. Let us nevertheless mention that some groups are making huge efforts and progress in this direction, see [31, 46], and that this validates the need for mathematical studies of the coupled system with the Fokker–Planck equation: see [44] or [29] for a local-in-time existence result of regular solutions. An alternative possibility is to simulate the SDE underlying this PDE, and the approach summarizes as the simulation of the system

$$\left\{\begin{array}{l} \left\{\begin{array}{r} \dfrac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\,\mathbf{u} - \Delta\mathbf{u} + \nabla p - \operatorname{div}\boldsymbol{\tau}_p = \boldsymbol{f}_{\mathrm{ext}}, \\ \operatorname{div}\mathbf{u} = 0, \end{array}\right. \\ \\ \left\{\begin{array}{r} \boldsymbol{\tau}_p(t,\mathbf{x}) = \mathbb{E}\big(\mathbf{R}_t(\mathbf{x})\otimes\boldsymbol{F}(\mathbf{R}_t(\mathbf{x}))\big), \\ \\ \mathrm{d}\mathbf{R}_t(\mathbf{x}) + \mathbf{u}\cdot\nabla\mathbf{R}_t(\mathbf{x}) = (\nabla\mathbf{u}\,\mathbf{R}_t(\mathbf{x}) - \boldsymbol{F}(\mathbf{R}_t(\mathbf{x})))\,\mathrm{d}t + \mathrm{d}\mathbf{W}_t. \end{array}\right. \end{array}\right. \tag{6}$$

At this stage, the reader may understand much of the relevance of our toy system (1).

To give a synthetic view of the micromacro approach and comparing it to the more conventional purely macroscopic approach, a concise statement is to say that system (2), of the form

$$\left\{\begin{array}{ll} \dfrac{D\mathbf{u}}{Dt} & = \mathcal{F}(\boldsymbol{\tau}_p,\mathbf{u}), \\ \\ \dfrac{D\boldsymbol{\tau}_p}{Dt} & = \mathcal{G}(\boldsymbol{\tau}_p,\mathbf{u}), \end{array}\right. \tag{7}$$

is replaced by (6) of the form

$$\left\{\begin{array}{ll} \dfrac{D\mathbf{u}}{Dt} & = \mathcal{F}(\boldsymbol{\tau}_p,\mathbf{u}), \\ \\ \boldsymbol{\tau}_p & = \boldsymbol{\tau}_p(\Sigma) \\ \\ \dfrac{D\Sigma}{Dt} & = \mathcal{G}_\mu(\Sigma,\mathbf{u}), \end{array}\right. \tag{8}$$

where Σ stands for an internal variable describing the state of the microstructure. The micromacro approach (8) essentially consists in increasing the number of scalar unknowns, and therefore is intrinsically more costly (in term of CPU time and memory requirements) than the macroscopic approach (7). In the present state-of-the-art, the micromacro approach (8) is still in its infancy and cannot compete in terms of computational efficiency with the standard and much more mature purely macroscopic approach (7). Nevertheless, it provides with a systematic track for improving closure relations, or at least fitting parameters of those, and already reveals as an efficient backroom strategy for such a purpose. This, and the hope it generates, suffices to justify a mathematical investment in such systems. In addition to this, it must be emphasized that, when simulating system (8), the numerical treatment of the Fokker–Planck equation by deterministic techniques is definitely more efficient than that of the associated SDE. Nevertheless, due to the dimension of the space where Σ varies, the latter techniques are not always tractable. *Unless a deterministic technique can be applied*, the stochastic simulation at the SDE level remains the method of choice, and this calls for a numerical analysis of the approach. In the present state-of-the-art, the latter will be performed in a low dimensional space, but with a view to applying to the large dimension case.

2.1 The Simplest Possible Model

The simplest occurrence of a system such as (6) is obtained a) when coarse-graining the description of the polymer chain into a single *dumbbell*, that is *one* spring between two beads, b) when the (purely entropic) force between the two beads simply reads as the *Hookean* force $\boldsymbol{F}(\mathbf{r}) = \frac{1}{2}\mathbf{r}$, and c) when the ambient flow considered is a Couette flow. Then system (6) simplifies into

$$\begin{cases} \dfrac{\partial u}{\partial t}(t,y) - \dfrac{\partial^2 u}{\partial y^2}(t,y) = \dfrac{\partial \tau}{\partial y}(t,y) + f_{\text{ext}}(t,y), \\ \forall y, \quad \begin{cases} \tau = \mathbb{E}\left(X_t^y Y_t\right), \\ \begin{cases} \mathrm{d}X_t^y = (-\dfrac{1}{2}X_t^y + \dfrac{\partial u}{\partial y}(t,y)\, Y_t)\,\mathrm{d}t + \mathrm{d}V_t, \\ \mathrm{d}Y_t = -\dfrac{1}{2}Y_t\,\mathrm{d}t + \mathrm{d}W_t. \end{cases} \end{cases} \end{cases} \tag{9}$$

where u denotes the component along the x axis of the velocity $\mathbf{u}$ depending only on y, while τ denotes the off-diagonal term of the extra stress tensor $\boldsymbol{\tau}_p$, the only relevant component in view of the simple geometry. On the other hand, (X_t, Y_t) denotes the two components of $\mathbf{R}_t$, and (V_t, W_t) is a two dimensional Brownian motion. In the dumbbell case, the vector $\mathbf{R}_t$ represents the length and the orientation of the polymer chains, at point y.

The model is typically relevant for a polymeric flow in a rheometer. The radius of the inner cylinder is almost the same as that of the outer cylinder, both are large, and the streamlines are expected to be cylinders as well: this justifies geometrically the approximation by a one dimensional flow. Therefore the model is not only mathematically convenient, but also rather close to an experimental device indeed utilized in practice in Mechanics.

In comparison to the "general" system (6), system (9) is simplified in two respects. First, because we consider a *shear flow*, the SDE is not a stochastic *partial* differential equation: the transport term $\mathbf{u} \cdot \nabla \mathbf{R}_t$ vanishes for geometrical reasons.

Therefore the coupling between two processes $\mathbf{R}_t(\mathbf{x}) = (X_t^y, Y_t^y)$ at different $\mathbf{x}$ boils down into the simple coupling term $\dfrac{\partial u}{\partial y} Y_t^y$, i.e. *via* the macroscopic flow. This significantly simplifies both the analysis (see [34] for a more general mathematical work) and the implementation. Second, because we consider *Hookean dumbbells* in a shear flow, the nonlinear term $\nabla\,\mathbf{u}\,\mathbf{R}_t(\mathbf{x})$ reduces here to the term $\dfrac{\partial u}{\partial y} Y_t$, which is linear since Y_t can be computed independently from $\mathbf{u}$ and X_t^y (and therefore does not depend on y, thus the notation Y_t). It is thus rather easy to prove the existence and uniqueness of a global-in-time weak solution (see [22]). In fact, it is to be noted that the Hookean dumbbell model as written in (9) in the Couette case, is indeed equivalent to the Oldroyd-B model, for the stress tensor calculated from (9) indeed satisfies the simplest one-dimensional form of the Oldroyd-B equation (3). In this respect, the Hookean dumbbell appears as a test situation for mathematical analysis, numerical analysis, and also algorithmic techniques, and no more than that.

System (9) is typically discretized as follows: the equation of conservation of momentum is discretized by finite difference in the time variable, and by finite elements for the space variable. $P1$ finite elements for the velocity, and $P0$ finite elements for the stress tensor are both easy to manipulate and convenient. The Galerkin formulation of the macroscopic equation therefore reads:

$$\frac{1}{\Delta t}\int_{\mathcal{D}}(\overline{u}_h^{n+1} - \overline{u}_h^n)v + \int_{\mathcal{D}} \frac{\partial \overline{u}_h^{n+1}}{\partial y}\frac{\partial v}{\partial y} = -\int_{\mathcal{D}} \overline{S}_{h,n}\frac{\partial v}{\partial y} + \int f_{\text{ext}}(t_n, y)v, \quad (10)$$

for $P1$ test functions v, where the superscript n stands for the time discretization, while the subscript h stands for the space discretization. Regarding the SDEs, they are discretized by an Euler explicit scheme in time, and of course by a Monte Carlo sampling (M realizations of each random process)[1]:

$$\begin{cases} \overline{X}_{h,n+1}^j - \overline{X}_{h,n}^j = \left(-\dfrac{1}{2}\overline{X}_{h,n}^j + \dfrac{\partial \overline{u}_h^{n+1}}{\partial y}\overline{Y}_n^j\right)\Delta t + \left(V_{t_{n+1}}^j - V_{t_n}^j\right), \\ \overline{Y}_{n+1}^j - \overline{Y}_n^j \qquad = -\frac{1}{2}\,\overline{Y}_n^j\,\Delta t + (W_{t_{n+1}}^j - W_{t_n}^j). \end{cases} \quad (11)$$

This then provides

$$\overline{S}_{h,n+1} = \frac{1}{M}\sum_{j=1}^{M} \overline{X}_{h,n+1}^j \overline{Y}_{n+1}^j, \quad (12)$$

which is to be inserted in the right-hand side of (10) at the next timestep. The crucial point to make, and that applies to all the models we refer to in this section, is that unlike the continuous level where the velocity u is a deterministic quantity, the fully discretized equations involve a velocity $\overline{u}_h^n$ that is indeed a random variable, since the empirical mean (12) is inserted in (10), *in lieu of* the expectation $\tau = \mathbb{E}\,(X_t^y Y_t)$. The three-fold discretization (discretization in time, discretization in space, discretization

[1] We omit here a cut-off procedure on the process Y_t that is unbounded, and refer to [22] for this technical detail.

in Monte Carlo) is a highly unusual feature, that in some sense characterizes the family of problems we are dealing with here. Correspondingly, this translates in the error estimate that has been first established in [22] (see [15] for an independent work), that is

$$\left|\left|u(t_n) - \overline{u}_h^n\right|\right|_{L^2_y(L^2_\omega)} + \left|\left|\mathbb{E}(X_{t_n} Y_{t_n}) - \frac{1}{M}\sum_{j=1}^{M} \overline{X}_{h,n}^j \overline{Y}_n^j\right|\right|_{L^1_y(L^1_\omega)} \leq C\left(h + \Delta t + \frac{1}{\sqrt{M}}\right),$$

for a constant C independent of h and $\Delta t \leq \frac{1}{2}$, but dependent on the data.

Note the occurrence of L^p_ω norms in the left-hand side, in order to account for the random nature of the objects manipulated. The orders of convergence in the right-hand side are as expected: Δt in time because of the Euler scheme (used twice), $\dfrac{1}{\sqrt{M}}$ for the Monte Carlo sampling, while the rate h stands here because of the $P0$ finite element approximation for the stress (while it can be shown, see [33], that the error in L^2 norm for the velocity itself scales as h^2, again as expected for $P1$ finite element).

2.2 Non Hookean Models

Less simple models than the Hookean model have been introduced, with a view to accounting for various physical phenomena of importance. In this respect, one important step is to account for the finite extensibility of the polymer chains, a fact that was ignored in the simple Hookean dumbbell model where X_t and Y_t were unbounded processes.

The FENE Model

Still in the context of a Couette flow, the FENE model (this acronym standing for Finite Extensible Nonlinear Elastic) reads

$$\left\{\begin{array}{l}
\dfrac{\partial u}{\partial t}(t,y) - \dfrac{\partial^2 u}{\partial y^2}(t,y) = \dfrac{\partial \tau}{\partial y}(t,y) + f_{\text{ext}}(t,y),\\
\tau = \mathbb{E}\left(\dfrac{X_t^y Y_t^y}{1-\frac{(X_t^y)^2+(Y_t^y)^2}{b}}\right),\\
\left\{\begin{array}{l}
\mathrm{d}X_t^y = \left(-\dfrac{1}{2}\dfrac{X_t^y}{1-\frac{(X_t^y)^2+(Y_t^y)^2}{b}} + \dfrac{\partial u}{\partial y}(t,y)\,Y_t^y\right)\mathrm{d}t + \mathrm{d}V_t,\\
\mathrm{d}Y_t^y = \left(-\dfrac{1}{2}\dfrac{Y_t^y}{1-\frac{(X_t^y)^2+(Y_t^y)^2}{b}}\right)\mathrm{d}t + \mathrm{d}W_t.
\end{array}\right.
\end{array}\right. \tag{13}$$

where the parameter b stands for the maximum (squared) length of the polymer chain.

Contrary to (9), due to the fact that Y_t^y here depends on X_t^y, the system (13) is fully nonlinear through the term $\dfrac{\partial u}{\partial y}(t,y)\,Y_t^y$, and its mathematical analysis is one order of magnitude more difficult than that of (9).

Mathematically, only a small-in-time existence and uniqueness result for system (13) has been established to date. It can be established either in Sobolev spaces (see [23]) or in Hölder spaces (see [16]), the former aiming at giving a sound ground to the numerical simulations. Regarding the SDE itself, the proof of the existence of a strong global-in-time solution falls in four steps, by a standard sequence of arguments in stochastic analysis: first, proof of existence of strong solution to the SDE without the shear term $\frac{\partial u}{\partial y}\,Y_t^y$, second, proof of existence of a weak solution by the use of a Girsanov transform to account for the shear term, third, proof of trajectorial uniqueness, and fourth, application of the Yamada-Watanabe Theorem. An alternative direct proof of existence of strong solutions is also possible by using the notion of multivalued SDEs (see [9, 25]). Nevertheless, the introduction of the notion of weak solutions is useful for establishing the regularity of the stress τ with a view to proving the existence for the coupled system. For the latter, only a local-in-time result has been proved. All efforts to improve this local-in-time result into a global one have failed to date[2]. In particular, the mathematical study of the Cauchy problem for such a nonlinear system cannot be expected to be, by any means, simpler than purely nonlinear macroscopic system of the type (2), which requires huge efforts mathematically, see [35, 17].

This difficulty encountered at the very mathematical level gives us the opportunity to make a few remarks, that we believe to be valid generally. In the absence of a transport term in the SDE (a fact due, we recall it, to the simplicity of the geometry of the Couette flow), and in the absence of any dependence of the diffusion coefficient upon the stochastic processes (X_t^y, Y_t^y) (we will see an opposite situation for rod-like models in Sect. 3.1), the only difficulty in analyzing the SDE lies in the possible singular character of the drift coefficient. Of course, the lack of regularity of the drift term might be circumvented by dealing with weak solutions of the SDE rather than strong solutions[3].

Concerning the existence of solution for the SDE, if $\frac{\partial u}{\partial y}$ has a meaning pointwise in y, the natural idea is to also give a sense to the SDE pointwise in y. (If u is not regular enough to define its derivative $\frac{\partial u}{\partial y}$ pointwise in y, our approach collapses, and one would need to build a "variational" definition of the solution of the SDEs.) Then the stress τ has also a meaning pointwise in y, which is, one should notice it, precisely what is done for models with a macroscopic constitutive law (2), see the review article [17]. We then concentrate on the regularity *in time* of the drift coefficient. This coefficient is a combination of two terms which are very different in nature: the entropic force term $\frac{(X_t^y, Y_t^y)}{1-\frac{(X_t^y)^2+(Y_t^y)^2}{b}}$ and the shear term $\frac{\partial u}{\partial y}\,Y_t^y$. Due to

[2] J.W. Barrett, C. Schwab and E. Süli have recently published in [2] a proof of existence of a global weak solution for the FENE model with a regularized velocity.

[3] Recall that for a weak solution, the driving Brownian motion, the probability space and the filtration are altogether part of the solution, while they are considered given for a strong solution.

physical reasons, the force term often derives from a convex potential and is more an advantage than a difficulty for the analysis. On the other hand, the regularity in time of the shear term is typically bootstrapped from the macroscopic equation itself.

The solution of the SDE is used for the computation of the stress τ. Since all what is needed is the expectation value τ that only depends on the law, it seems it is enough to concentrate on the existence of weak solutions. However, this does not seem to be enough to provide the regularity needed for defining the stress. Fortunately, as the singularity of the function in the expression of the stress tensor is *the same as* that in the drift term of the SDE, the analysis turns out to be possible.

As far as the numerical analysis of system (13) is concerned, a bottleneck that has not been circumvented nor overcome so far, is the lack of a numerical analysis concerning the convergence of a singular function of the Euler discretized process associated with an SDE involving a drift coefficient with a related singularity. It is indeed possible to prove the weak convergence of the Euler scheme, even in the presence of a singular drift coefficient of the explosive form of (13) (see [20]), but the convergence of the stress τ remains an open problem. In the absence of such an analysis, it has not been possible to date to address that of the coupled system (13).

The FENE–P Model

A slight modification of the above FENE model proceeds from the wish to obtain an equivalent purely macroscopic model (a property that the FENE model does not enjoy, to the best of common knowledge) while keeping track in the modeling of the finite extensibility of the polymer chain. This modification consists in replacing the squared length of the chain in the denominators of (13) by its expectation value. The model obtained this way is called FENE-P (the P standing for Peterlin):

$$\left\{\begin{array}{l} \dfrac{\partial u}{\partial t} - \dfrac{\partial^2 u}{\partial y^2} = \dfrac{\partial \tau}{\partial y} + f_{\mathrm{ext}}, \\ \tau = \mathbb{E}\left(\dfrac{X_t^y Y_t^y}{1-\frac{\mathbb{E}\left((X_t^y)^2+(Y_t^y)^2\right)}{b}}\right), \\ \left\{\begin{array}{l} \mathrm{d}X_t^y = \left(-\dfrac{1}{2}\dfrac{X_t^y}{1-\frac{\mathbb{E}\left((X_t^y)^2+(Y_t^y)^2\right)}{b}} + \dfrac{\partial u}{\partial y} Y_t^y\right) \mathrm{d}t + \mathrm{d}V_t, \\ \mathrm{d}Y_t^y = \left(-\dfrac{1}{2}\dfrac{Y_t^y}{1-\frac{\mathbb{E}\left((X_t^y)^2+(Y_t^y)^2\right)}{b}}\right) \mathrm{d}t + \mathrm{d}W_t. \end{array}\right. \end{array}\right. \tag{14}$$

It should be remarked that the SDE is now nonlinear in the sense of MacKean, precisely because of the presence of the expectation value in the drift coefficient.

In [26], the well-posedness of the SDE, together with the convergence of the stress tensor, when the expectation values in the SDEs and in the expression of the stress are replaced by empirical means, toward the exact stress tensor are proved. Both proofs are performed for a more general geometry than that of a Couette flow, however in the context where the flow u is supposed to be known and regular enough.

2.3 Variance Reduction Issues

Needless to say, noise reduction issues are crucial in the numerical simulation of systems such as (1). Again, we do concentrate on the peculiarity of this question *in the presence of a coupling* and not on the general question of noise reduction. As a pedagogic case study, let us come back to the simulation of the simplest system (9) and mention the following practical observation. Two numerical experiments can be performed: the simulation of (9) as such and the simulation of (9) when the Brownian motion V_t is assumed to also depend on the space variable y (or more precisely on its discrete counterpart):

$$\mathrm{d}X_t^y = \left(-\frac{1}{2}X_t^y + \frac{\partial u}{\partial y}(t,y)\,Y_t\right)\,\mathrm{d}t + \mathrm{d}V_t^y.$$

It is intuitive that in the latter case, as the noise inserted in the system is more important, the variance on the result is higher. It is indeed the case, and this observation is valid beyond the simple one-dimensional simulation considered here, that the variance on the velocity u increases. However, and this is a highly counterintuitive fact, the variance on the stress tensor τ_p diminishes. In [24], the phenomenon was analyzed in details, which is possible precisely because of the simplicity of the situation at hand. It was demonstrated how a coupling between the SDEs and the PDE makes possible such an observation. Notably, it was shown that in the absence of the coupling, that is when the term $\dfrac{\partial u}{\partial y}$ is given extrinsically, the counterintuitive diminution of the variance of the stress is replaced by a growth in the variance, as is the case for the velocity !

Related to this observation on the crucial impact that the coupling may have on the variance of the results is the following one. As we pointed out, the velocity field u and the stress τ are both, once fully discretized, random variables. Therefore, the relevant output of a simulation is the averaged result over many simulations, carried out independently. Apart from the discretization parameters h, Δt and M mentioned above, a fourth relevant parameter is thus N_e, the number of numerical experiments carried out. In the absence of a coupling, the result is insensitive to each of M and N_e, only the product of the two being meaningful. But, because of the coupling between various realizations of the SDE *via* the macroscopic term $\dfrac{\partial u}{\partial y}$, there is an intricate non trivial interplay between M and N_e. On system (9), it can be again explained which is the most efficient choice for M and N_e (see [24]).

3 Modeling Various Fluids

3.1 Liquid Crystals

In Sect. 2, we have considered dilute solutions of flexible polymers. Some other polymers behave more like rigid rods, and this introduces some anisotropy in the system.

Solutions of such rigid polymers are called polymeric liquid crystals. One of the main aspect to take into account in the modeling of solutions of rodlike polymers is that the interaction of the polymers becomes important at much a lower concentration than with flexible polymers.

One model is the Doi model (see [14, 42]), which describes the evolution for a configuration vector $\mathbf{R}_t$ by a stochastic differential equation:

$$\begin{aligned}
&\mathrm{d}\mathbf{R}_t + \mathbf{u}\cdot\nabla\mathbf{R}_t\,\mathrm{d}t \\
&= \left(\mathrm{Id} - \frac{\mathbf{R}_t\otimes\mathbf{R}_t}{||\mathbf{R}_t||^2}\right)\left(\left(\nabla\mathbf{u}\,\mathbf{R}_t - \frac{1}{2}B^2\nabla V(\mathbf{R}_t)\right)\mathrm{d}t + B\mathrm{d}\mathbf{W}_t\right) \\
&\quad -\frac{d-1}{2}B^2\frac{\mathbf{R}_t}{||\mathbf{R}_t||^2}\,\mathrm{d}t,
\end{aligned} \tag{15}$$

where B is a positive constant and $d = 2$ or 3 is the dimension of the ambient space. Notice that B may also be a function $B(\mathbf{R}_t)$ in some models (with then an additional term involving $\nabla(B^2)$ in the drift term). Notice also that we assume that all the initial conditions $\mathbf{R}_0(\boldsymbol{x})$ have a fixed length L so that $\forall(t,\boldsymbol{x})$, $||\mathbf{R}_t(\boldsymbol{x})|| = ||\mathbf{R}_0(\boldsymbol{x})|| = L$. The potentiel V accounts for the mean-field interaction between the polymers. For example, the Maier–Saupe potential is:

$$V(\mathbf{R}) = -\frac{1}{L^4}\mathbb{E}(\mathbf{R}_t\otimes\mathbf{R}_t) : \mathbf{R}\otimes\mathbf{R}. \tag{16}$$

The stress tensor is then given by:

$$\boldsymbol{\tau}_p(t) = \mathbb{E}(\mathbf{U}_t\otimes\mathbf{U}_t) + \mathbb{E}\Big(\mathbf{U}_t\otimes\big((\mathrm{Id} - \mathbf{U}_t\otimes\mathbf{U}_t)\,\nabla V(\mathbf{U}_t)\big)\Big) - \mathrm{Id} \tag{17}$$

where $\mathbf{U}_t = \dfrac{\mathbf{R}_t}{L}$ is the rod orientation. We have neglected the viscous contribution in (17). The fully coupled system then consists in the first two equations of (6) with (15)–(17), thus again giving a system of type (1). Notice that the main differences with system (6) are the nonlinearity in the sense of MacKean due to the presence of the expectation value in the potential V and the fact that the diffusion term depends on the process $\mathbf{R}_t$.

For an analysis of the fully coupled system in the special case of a shear flow, we refer to [47] which deals with the Fokker–Planck version of (15)–(17) and [30] which also gives an error estimate for a finite difference-Monte Carlo hybrid numerical scheme. The longtime behaviour of the Fokker–Planck equation has been studied in [10] (see also [11]). Some numerical methods to solve the stochastic differential equation (15) are proposed in [42]. On the other hand, we are not aware of any rigorous numerical analysis of numerical methods to solve this system without closure approximation.

3.2 Concentrated Suspensions

Let us now slightly change the context. For concentrated suspensions (such as muds or clays), one model available in the literature is the Hebraud-Lequeux model [21].

This model describes the rheology of the fluid in terms of a Fokker–Planck equation ruling the evolution in time of the probability of finding, at each point, the fluid in a given state of stress. In a one-dimensional setting such as, again, the Couette flow, the stress at the point y and at time t is determined by one scalar variable σ:

$$\left\{\begin{array}{l} \dfrac{\partial p}{\partial t}(t,y,\sigma) = -\dfrac{\partial u}{\partial y}(t,y)\dfrac{\partial p}{\partial \sigma}(t,y,\sigma) + D(p)\,\dfrac{\partial^2 p}{\partial \sigma^2}(t,y,\sigma) \\ \qquad\qquad\qquad\qquad -H(|\sigma|-1)p(t,y,\sigma) + D(p)\delta_0, \\ \qquad D(p) = \displaystyle\int_{|\sigma|\geq 1} p(t,y,\sigma)\,\mathrm{d}\sigma. \end{array}\right. \tag{18}$$

In the above system, where we have again on purpose omitted all physical constants, the function H denotes the Heaviside function. It aims at modeling the presence of a threshold constraint (here set to one): when the constraint is above the threshold, the stress relaxes to zero, which translates into the two last terms of the Fokker–Planck equation. The diffusion in the stress space is also influenced nonlinearly by the complete state of stress, as indicated by the definition of $D(p)$. On the other hand, the function $\dfrac{\partial u}{\partial y}(t,y)$ accounts for a shear rate term, here provided by the macroscopic flow. The contribution to the stress at the point y under consideration is then given by the average

$$\tau(t,y) = \int_{\mathbf{R}} \sigma\, p(t,y,\sigma)\,\mathrm{d}\sigma. \tag{19}$$

The fully coupled system consisting of the Fokker–Planck equation (18), the expression (19) of the stress tensor, and the macroscopic equation for the Couette flow (first line of (9)) has been studied mathematically in a series of work [6, 7, 8].

Alternately to a direct attack of the Fokker–Planck equation (18), one might wish to simulate the associated SDE *with jumps* that reads

$$d\sigma_t = \frac{\partial u}{\partial y}\,\mathrm{d}t \;+\; \sqrt{2\mathbb{P}(|\sigma_t| \geq 1)}\,\mathrm{d}W_t \;-\; 1_{\{|\sigma_{t^-}|\geq 1\}}\sigma_{t^-}\,\mathrm{d}N_t, \tag{20}$$

where W_t is a Brownian motion and N_t is an independent Poisson process with unit intensity. Note that, in addition to the jumps, equation (20) is nonlinear in the sense of MacKean, as the diffusion coefficient depends on the marginal law of the solution at time t.

The coupled system to simulate then reads in the form of a system of type (1) (note the SDE has jumps, though)

$$\left\{\begin{array}{l} \dfrac{\partial u}{\partial t}(t,u) - \dfrac{\partial^2 u}{\partial y^2}(t,y) = \dfrac{\partial \tau}{\partial y}(t,y) \\ \forall y, \left\{\begin{array}{l} \tau(t,y) = \mathbb{E}(\sigma_t(y)) \\ \mathrm{d}\sigma_t(y) = \dfrac{\partial u}{\partial y}\,\mathrm{d}t \;+\; \sqrt{2\mathbb{P}(|\sigma_t(y)| \geq 1)}\,\mathrm{d}W_t \;-\; 1_{\{|\sigma_{t^-}(y)|\geq 1\}}\sigma_{t^-}(y)\,\mathrm{d}N_t. \end{array}\right. \end{array}\right. \tag{21}$$

Numerical simulations of this system have been carried out successfully (see [19]), but in the absence of any numerical analysis to date.

3.3 Coupling PDEs and SDEs for the Simulation of Dispersed Two-Phase Flows

Dispersed two-phase flows are characterized by the presence of one phase (either solid, liquid or vapour) as separate inclusions called particles in the other phase called fluid. In both the following examples, the evolution of particles is modeled by a SDE (or the associated Fokker–Planck equation) while fluid equations are written for the other phase. In the example of dispersed turbulent two-phase flows, there is only a one-way coupling : the particles motion is influenced by the drag force of the fluid. In the example of sprays, the reverse coupling also holds : the drag force appears as a driving force in the equation for the conservation of the momentum of the fluid.

Dispersed Turbulent Two-Phase Flows

In the approach proposed by [37], the fluid phase is described by a classical turbulence model such as the $k-\epsilon$ model. It gives at each time t and each point x the mean velocity of the fluid $\langle U_f \rangle$, the covariance matrix of the velocity, the mean pressure $\langle P \rangle$ and the mean dissipation rate of energy $\langle \epsilon \rangle$. On the other hand, the particles are described by a Lagrangian approach. An extension of Kolmogorov theory suggests that the acceleration of the fluid velocity U_s seen by particles is a fast variable which can be modeled by a SDE driven by a d-dimensional Brownian motion W. This leads to the following evolution for the position X, the velocity U_p and the fluid velocity seen by particles U_s

$$\mathrm{d}X(t) = U_p(t)\mathrm{d}t \tag{22}$$

$$\mathrm{d}U_p(t) = \frac{1}{\tau_p}(U_s(t) - U_p(t))\mathrm{d}t + g\mathrm{d}t \tag{23}$$

$$\begin{aligned}\mathrm{d}U_s^i(t) = &\left(\sum_{j=1}^{d}(\langle U_p^j \rangle - \langle U_s^j \rangle)\frac{\partial \langle U_f^i \rangle}{\partial x_j} - \frac{1}{\rho_f}\frac{\partial \langle P \rangle}{\partial x_i} - \frac{U_s^i(t) - \langle U_s^i \rangle}{T_{L,i}^*}\right)(t, X(t))\mathrm{d}t \\ &+ \sqrt{C_0^* \langle \epsilon \rangle (t, X(t))}\,\mathrm{d}W_t^i,\ i \leq d,\end{aligned} \tag{24}$$

where g denotes the gravity and ρ_f the fluid density. On the other hand, the quantities τ_p, $T_{L,i}^*$ and C_0^* depend on U_p, U_s and on the mean fields representing the fluid in a very intricate manner that is made precise in [37]. Function $\langle U_p \rangle(t, x)$ (resp. $\langle U_s \rangle(t, x)$) stands for the conditional expectation $\mathbb{E}(U_p(t)|X(t) = x)$ (resp. $\mathbb{E}(U_s(t)|X(t) = x)$). The full coupled system then reads:

$$\begin{cases} k - \epsilon \text{ model giving } \langle U_f \rangle, \langle P \rangle, \langle \epsilon \rangle \\ \qquad \text{and (together with } U_p \text{ and } U_s) \ \tau_p, T_{L,i}^* \text{ and } C_0^*, \\ \text{(22)–(23)–(24).} \end{cases}$$

When $(X(t), U_p(t), U_s(t))$ admits a density $p(t, x, u, v)$ with respect to the Lebesgue measure, one has

$$\langle U_p\rangle(t,x) = \frac{\displaystyle\int_{\mathbf{R}^{2d}} up(t,x,u,v)\mathrm{d}u\mathrm{d}v}{\displaystyle\int_{\mathbf{R}^{2d}} p(t,x,u,v)\mathrm{d}u\mathrm{d}v} \text{ and } \langle U_s\rangle(t,x) = \frac{\displaystyle\int_{\mathbf{R}^{2d}} vp(t,x,u,v)\mathrm{d}u\mathrm{d}v}{\displaystyle\int_{\mathbf{R}^{2d}} p(t,x,u,v)\mathrm{d}u\mathrm{d}v}.$$

Because of the presence of these functions in the right-hand side of (24), the SDE (22)–(24) is nonlinear in the sense of MacKean with an ill-behaved nonlinearity. As a consequence, a rigorous study of existence and uniqueness is an open issue probably of outstanding difficulty.
The numerical approach proposed in [39] is a particle-mesh method. The mean quantities are evaluated at the grid points either from the $k-\epsilon$ fluid model ($\langle U_f\rangle$, $\langle P\rangle$ and $\langle\epsilon\rangle$) or from the particle data ($\langle U_p\rangle$ and $\langle U_s\rangle$). These values are projected on the particles positions to integrate forward in time (22)–(24). Last, $\langle U_p\rangle$ and $\langle U_s\rangle$ are averaged at the grid points from the new positions and velocities of the particles.

Modeling of Sprays in a Fluid Phase

According to [12], a spray can be modeled by a kinetic equation, usually a variant of the Boltzmann equation in which a force acting on the particles is due to the surrounding fluid. It describes the evolution of the particle density function $f(t,x,v,r)$ which gives the density of particles in the spray with position x, velocity v and radius r at time t. In a simple form, it writes

$$\partial_t f + v.\nabla_x f + \nabla_v.(Ff) = Q(f), \tag{25}$$

where Q is a kernel modeling the effects of collisions, coalescences and breakups of the particles and the drag force F of the fluid on the particles is given by

$$F(t,x,v,r) = -\frac{4}{3}\pi r^3 \nabla_x p(t,x) - D(v-u(t,x)), \tag{26}$$

p and u being the pressure and the velocity fields of the fluid and D a drag coefficient. This equation can be considered as the Fokker–Planck equation associated with a stochastic process with jumps, at least in the absence of the coalescence and breakup phenomena which may modify the total amount of particles $\displaystyle\int f\mathrm{d}x\mathrm{d}v\mathrm{d}r$.
The ambient fluid is described by a set of Euler equations, relative to the density ρ and the velocity u multiplied by the volume fraction $\alpha = 1 - \int \frac{4}{3}\pi r^3 f\mathrm{d}v\mathrm{d}r$:

$$\begin{cases} \dfrac{\partial(\alpha\rho)}{\partial t} + \nabla_x.(\alpha\rho u) = 0, \\ \dfrac{\partial(\alpha\rho u)}{\partial t} + \nabla_x.(\alpha\rho u\otimes u) + \nabla_x p = \displaystyle\int -Ff\,\mathrm{d}v\mathrm{d}r, \\ p = p(\rho). \end{cases} \tag{27}$$

The full coupled system of type (1) is then (27)–(25).

From a numerical point of view, the fluid equations (27) are usually solved by standard deterministic methods (finite volume techniques for instance). As the phase space dimension is 7, equation (25) is discretized by a particle method involving a stochastic treatment of the right-hand side like for the standard Boltzmann equation [1]. Only few mathematical studies concerning the derivation of the above equations or the existence and behaviour of solutions seem to exist (see the references in [12]).

4 An Example Outside Fluid Mechanics: Photon Transport

Hot enough matter (such as plasma) spontaneously emits photons. The photons travel in the spatial domain $\mathcal{D}$ and can be emitted, scattered by the electrons or absorbed by the matter. A simple model reads as follows:

$$\lambda(\theta)\frac{\partial\theta}{\partial t}(t,x) + q(x)\theta(t,x) = \frac{q(x)}{4\pi}\int_{\mathcal{S}_2} f(t,x,w)\mathrm{d}w, \tag{28}$$

$$\frac{\partial f}{\partial t}(t,x,v) + v.\nabla_x f(t,x,v) + \sigma(x)\left(f(t,x,v) - \frac{1}{4\pi}\int_{\mathcal{S}_2} f(t,x,w)\mathrm{d}w\right) \\ + q(x)f(t,x,v) = q(x)\theta(t,x), \tag{29}$$

where the unknowns are the photon density (or radiative intensity) $f(t,x,v)$ (here supposed not to depend on the frequency of the photons) and the fourth power $\theta(t,x)$ of the temperature. The space variable is $x \in \mathcal{D}$ and v denotes the velocity which belongs here to the unit sphere $\mathcal{S}_2$. Equation (28) is the energy balance equation, while (29) rules the motion of the photons. In equations (28)–(29), $\lambda(\theta)$ denotes the heat capacity of the matter multiplied by $\theta^{3/4}$, and q and σ are respectively the opacity of the matter and the Thomson scattering coefficient. We assume that the nonnegative function σ is bounded from above by the constant $\bar{\sigma}$.

As in previous cases (see e.g. Sect. 3.2), one can use a stochastic process to represent solutions to (29). More precisely, when $q \equiv 0$, (29) is the Fokker–Planck equation associated with the following SDE with jumps (see [32]):

$$\begin{cases} \mathrm{d}X_r^{x,v} = V_r^{x,v}\,\mathrm{d}r, \\ \mathrm{d}V_r^{x,v} = 1_{\{\bar{\sigma}\mathcal{U}_{N_r} \leq \sigma(X_{r-}^{x,v})\}}\left(\mathcal{V}_{N_r} - V_{r-}^{x,v}\right)\mathrm{d}N_r, \\ (X_0^{x,v}, V_0^{x,v}) = (x,v), \end{cases} \tag{30}$$

where $(N_r)_{r\geq 0}$ is a Poisson process with intensity $\bar{\sigma}$ independent from the sequence $(\mathcal{U}_k, \mathcal{V}_k)_{k\geq 1}$ of independent vectors uniformly distributed on $[0,1] \times \mathcal{S}_2$. Using the process $(X_r^{x,v}, V_r^{x,v})$, the solution of (29) can be expressed for a general opacity q as the solution of the following variational formulation: for any test function φ on $\mathcal{D} \times \mathcal{S}_2$, $\forall t \geq s$,

$$\int_{\mathcal{D}\times\mathcal{S}_2} \varphi(x,v)f(t,x,v)\mathrm{d}x\mathrm{d}v \\ = \int_{\mathcal{D}\times\mathcal{S}_2} \mathbb{E}\left(\varphi(X_{t-s}^{x,v}, V_{t-s}^{x,v})\mathrm{e}^{-\int_0^{t-s} q(X_\tau^{x,v})d\tau}\right) f(s,x,v)\,\mathrm{d}x\mathrm{d}v \\ + \int_s^t \int_{\mathcal{D}\times\mathcal{S}_2} \mathbb{E}\left(\varphi(X_{t-r}^{x,v}, V_{t-r}^{x,v})\mathrm{e}^{-\int_0^{t-r} q(X_\tau^{x,v})d\tau}\right) q(x)\theta(r,x)\,\mathrm{d}x\mathrm{d}v\mathrm{d}r. \tag{31}$$

From a numerical point of view, the difficulty in the discretization of (28)–(29) comes from the right-hand side of (28). It is needed to build a discretization scheme which allows for an implicit treatment of the dependence of $\int_{\mathcal{S}_2} f(t,x,w)\mathrm{d}w$ upon θ, in order to get the most stable scheme. The so-called Symbolic Monte Carlo method (see [45]) consists in computing f as a function of θ from (29) in order to get a closed implicit equation (see (32) below) for θ (see also [18] for another approach).

Let us introduce a spatial mesh $\{M_i,\ i \in I\}$ and a time-step $\Delta t > 0$. The numerical procedure consists in approximating θ by space-time functions, piecewise constant on the cells $[n\Delta t, (n+1)\Delta t] \times M_i$:

$$\theta^n = \sum_{i \in I} \theta_i^n 1_{M_i}(x)$$

and f by a sum of (weighted) Dirac masses:

$$f^n = \sum_{k=1}^{\nu_n} w_k^n \delta_{(X_k^n, V_k^n)},$$

where ν_n denotes the number of Dirac masses, w_k^n the weights and (X_k^n, V_k^n) some random variables associated with a discretization of (30). Equation (28) is then discretized by a classical implicit Euler and finite element scheme, while one uses (31) with $\varphi(x,v) = 1_{M_i}(x)$ to implicitly compute the right-hand side of (28).

We thus obtain the following algorithm: knowing (θ^n, f^n), θ^{n+1} is obtained as the solution (obtained by a Newton method) of:

$$\lambda(\theta_i^{n+1}) \frac{\theta_i^{n+1} - \theta_i^n}{\Delta t} + q_i \theta_i^{n+1} = \frac{q_i}{4\pi |M_i| \Delta t} \left(A_i^n + \sum_{j \in J} W_{i,j} \theta_j^{n+1} \right),\ i \in I \quad (32)$$

where A_i^n and $W_{i,j}$ are defined by:

$$A_i^n = \int_{\mathcal{D} \times \mathcal{S}_2} \mathbb{E} \left(\int_0^{\Delta t} 1_{M_i}(X_s^{x,v}) \mathrm{e}^{-\int_0^s q(X_\tau^{x,v}) d\tau} \mathrm{d}s \right) f^n(\mathrm{d}x, \mathrm{d}v),$$

$$W_{i,j} = \int_{M_j \times \mathcal{S}_2} \mathbb{E} \left(\int_0^{\Delta t} (\Delta t - s) 1_{M_i}(X_s^{x,v}) \mathrm{e}^{-\int_0^s q(X_\tau^{x,v}) d\tau} \, \mathrm{d}s \right) q(x) \, \mathrm{d}x \mathrm{d}v.$$

Notice that the matrix W does not depend on time and can be precomputed offline by a Monte Carlo method. The vector A^n is also computed by a Monte Carlo method, using an ensemble of processes obtained by a time-discretization of (30). These processes are then used to compute f^{n+1}, together with an appropriate updating of the weights and of the number of particles, to account for the opacity q in (29).

References

1. Baranger, C., Desvillettes, L.: Study at the numerical level of the kernels of collision, coalescence and fragmentation for sprays. Preprint 2003-29 of the CMLA, (2003).
2. Barrett, J.W., Schwab, C., Suli, E.: Existence of Global Weak Solutions for Polymeric Flow Model, Mathematical models and methods in applied sciences, 15(3) 2005.
3. Bird, R., Armstrong, C., Hassager, O.: Dynamics of polymeric liquids, volume 1, Wiley (1987).
4. Bird, R., Curtiss, Ch., Armstrong, C., Hassager, O.: Dynamics of polymeric liquids, volume 2, Wiley (1987).
5. Bourgat, J.F., Le Tallec, P., Tidriri, D., Qiu, Y.: Numerical coupling of nonconservative or kinetic models with the conservative compressible Navier–Stokes equations, In: Domain decomposition methods for partial differential equations, Proc. 5th Int. SIAM Symp., Norfolk/VA (USA) 1991, 420–440 (1992).
6. Cancès, E., Catto, I., Gati, Y.: Mathematical analysis of a nonlinear parabolic equation arising in the modeling of non-Newtonian flows, to appear in SIAM Journal on Mathematical Analysis.
7. Cancès, E., Catto, I., Gati, Y., Le Bris, C., Well-posedness of a multiscale model for concentrated suspensions, submitted to SIAM Multiscale Modeling and Simulation.
8. Cancès, E., Le Bris, C.: Convergence to equilibrium for a multiscale model for suspensions, submitted.
9. Cépa, E.: Équations différentielles stochastiques multivoques, PhD thesis, Université d'Orléans (1994).
10. Constantin, P., Kevrekidis, I., Titi, E.S.: Remarks on a Smoluchowski equation, Discrete and Continuous Dynamical Systems, **11**(1), 101–112 (2004).
11. Constantin, P., Kevrekidis, I., Titi, E.S.: Asymptotic states of a Smoluchowski equation, Archive for Rational Mechanics and Analysis, vol 174, 3, pp 365–384, (2004).
12. Desvillettes, L.: About the modeling of complex flows by gas-particle methods. Electronic proceedings of the congress "Trends in Numerical and Physical Modeling for Industrial Multiphase Flows" Cargèse (2000).
13. Doi, M.: Introduction to polymer physics, Oxford Science publications (1992).
14. Doi, M., Edwards, SF.: The theory of polymer dynamics, Oxford science publications (1986).
15. E, W., Li, T., Zhang, P.: Convergence of a stochastic model for the modeling of polymeric fluids, Acta Mathematicae Applicatae Sinicae, **18**(4), 529–536 (2002).
16. E, W., Li, T., Zhang, P.: Well-posedness for the dumbbell model of polymeric fluids, Comm. Math. Phys., **248**(2), 409–427 (2004).
17. Fernández-Cara, E., Guillén, F., Ortega., R.R.: Handbook of numerical analysis, Vol. 8, chapter Mathematical modeling and analysis of viscoelastic fluids of the Oldroyd kind, 543–661, Amsterdam: North Holland/ Elsevier (2002).
18. Fleck, J.A., Cummings, J.D.: An implicit scheme for calculating nonlinear radiative transport, J. Comp. Physics **8**, 315–342 (1971)
19. Gati, Y., Analyse mathématique et simulations numériques d'un modèle de fluides complexes, Thèse de l'Ecole Nationale des Ponts et Chaussées, 2004. See also: Gati, Y. Numerical simulation of micro-macro model of concentrated suspensions, Int. J. for Numerical Methods in Fluids, Special Issue: ICFD Conference on Numerical Methods for Fluid Dynamics, to appear.
20. Gyöngy, I., Krylov, N.: Existence of strong solutions for Itô's stochastic equations via approximations, Probab. Theory Relat. Fields, **105**, 143–158 (1996).

21. Hébraud, P., Lequeux, F.: Mode coupling theory for the pasty rheology of soft glassy materials, Phys. Rev. Lett., **81**(14), 2934–2937 (1998).
22. Jourdain, B., Lelièvre, T., Le Bris, C.: Numerical analysis of micro-macro simulations of polymeric fluid flows: a simple case, Mathematical Models and Methods in Applied Sciences, **12**(9), 1205–1243 (2002).
23. Jourdain, B., Lelièvre, T., Le Bris, C.: Existence of solution for a micro-macro model of polymeric fluid: the FENE model, Journal of Functional Analysis, **209**, 162–193 (2004).
24. Jourdain, B., Lelièvre, T., Le Bris, C.: On a variance reduction technique for the micromacro simulations of polymeric fluids, Journal of non-Newtonian fluid mechanics, volume 122, pp 91-106, (2004).
25. Jourdain, B., Lelièvre, T.: Mathematical analysis of a stochastic differential equation arising in the micro-macro modeling of polymeric fluids, Probabilistic Methods in Fluids Proceedings of the Swansea 2002 Workshop, Eds: Davies, I.M., Jacob,N., Truman, A., Hassan, O., Morgan, Y., Weatherill, N.P., 205–223 (2003).
26. Jourdain, B., Lelièvre, T.: Convergence of a stochastic particle approximation of the stress tensor for the FENE-P model, preprint (2004).
27. Keunings, R.: Simulation of viscoelastic fluid flow, In: Computer modeling for polymer processing, Ch. Tucker III, Hanser (1989).
28. Keunings, R.: Micro-macro methods for the multiscale simulation of viscoelastic flows using molecular models of kinetic theory, in Rheology Reviews 2004, D.M Binding and K. Walters, Eds., British Society of Rheology, pp 67-98.
29. Li, T., Zhang, H., Zhang, P.: Local existence for the dumbbell model of polymeric fluids, Comm. PDE, **29**(5-6), 903–923 (2004).
30. Li, T., Zhang, P., Zhou, X.: Analysis of 1+1 dimensional stochastic models of liquid crystal polymer flows, Comm. Math. Sci. 2(2), 295-316, (2004).
31. Lozinski, A.: Spectral methods for kinetic theory models of viscoelastic fluids, PhD thesis, Ecole Polytechnique Fédérale de Lausanne (2003).
32. Lapeyre, B., Pardoux, E., Sentis, R.: Méthodes de Monte Carlo pour les équations de transport et de diffusion, Mathématiques et Applications, Springer-Verlag (1998).
33. Lelièvre, T.: Optimal error estimate for the CONNFFESSIT approach in a simple case, Computers and Fluids, **33**, 815–820 (2004).
34. Le Bris, C., Lions, P.L.: Renormalized solutions of some transport equations with partially $W^{1,1}$ velocities and applications, Annali di Matematica pura ed applicata, **183**, 97–130 (2004).
35. Lions, P.L., Masmoudi, N.: Global solutions for some Oldroyd models of non-Newtonian flows, Chin. Ann. Math., Ser. B, **21**(2), 131–146 (2000).
36. Mattingly, J.C.: The stochastically forced Navier–Stokes equations: energy estimates and phase space contraction, PhD Thesis, Princeton University (1998).
37. Minier, J.P.: Probabilistic approach to turbulent two-phase flows modeling and simulation : theoretical and numerical issues. Monte Carlo Methods and Appl., **7**(3-4), 295–310 (2001).
38. Minier, J.P., Peirano, E.: The pdf approcah to polydispersed turbulent two-phase flows. Physics Reports., **352**(1-3), 1–214 (2001).
39. Minier, J.P., Peirano, E., Chibbaro, S.: Weak first- and second-order numerical schemes for stochastic differential equations appearing in Lagrangian two-phase flow modeling. Monte Carlo Methods and Appl., **9**(2), 93–133 (2003).
40. N'Kaoua, T.: Solution of the nonlinear radiative transfer equations. SIAM J. Sci. Stat. Comput., **12**, 505–520 (1991).
41. N'Kaoua, T., Sentis, R.: A New Time-Discretization for the radiative transfer equations. SIAM J. Numer. Analysis, **30**, 733–748 (1993).

42. Ottinger, H.C.: Stochastic processes in polymeric fluids, Springer (1996).
43. Owens, R.G., Phillips, T.N.: Computational rheology, Imperial College Press, (2002).
44. Renardy, M.: An existence theorem for model equations resulting from kinetic theories of polymer solutions, SIAM J. Math. Anal., **22**, 313–327 (1991).
45. Sentis, R., : Monte Carlo methods in neutron and photon transport problems. Monte Carlo Methods and Appl., **7**(3-4), 383–396 (2001).
46. Suen, J.K.C., Joo, Y.L., Armstrong, R.C.: Molecular orientation effects in viscoelasticity, Annu. Rev. Fluid Mech., **34**, 417–444 (2002).
47. Zhang, H., Zhang, P.: A theoretical and numerical study for the rod-like model of a polymeric fluid, J. Comp. Math., **22**(2), 319–330 (2004).

Adaptive Submodeling for Linear Elasticity Problems with Multiscale Geometric Features

Mats G. Larson, Fredrik Bengzon, and August Johansson

Department of Mathematics, Umeå University, 90187 Umeå, Sweden
mats.larson@math.umu.se, fredrik.bengzon@math.umu.se,
august.johansson@math.umu.se

Summary. Submodeling is a procedure for local enhancement of the resolution of a coarse global finite element solution by solving a local problem on a subdomain containing an area of particular interest. We focus on linear elasticity and computation of local stress levels determined by the local geometry of the domain. We derive a posteriori error estimates for the submodeling procedure using duality techniques. Based on these estimates we propose an adaptive procedure for automatic choice of the resolution and size of the submodel. The procedure is illustrated for problems of industrial interest.

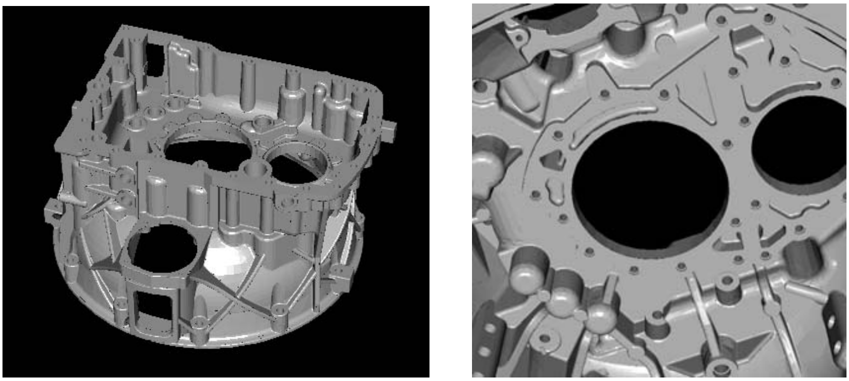
Key words: adaptive multiscale method, a posteriori error estimate, finite element, meshrefinement

1 Introduction

1.1 Submodeling

In many industrial engineering applications simulation technology must deal with very complex geometries with details on many different scales. See Fig. 1 for a typical example of a gearbox casing with detailed geometry. Often one first removes very small geometric details and then create a mesh based on the simplified model. Even this simplified mesh is often large due to the complexity of the geometry of the problem. Such large initial grids make it difficult to apply standard automatic adaptive procedures based on a posteriori error estimates since in only a few refinement steps may result in a very large mesh. Nevertheless local mesh refinement may often be necessary to compute accurate local values of the stress field. In such situations submodeling is an attractive alternative.

Submodeling is based on solving a local problem on a subdomain of the global domain containing an area of particular interest, for instance an area with high stress levels in the coarse grid solution, with improved resolution. Boundary conditions are obtained from the coarse grid solution. Submodeling may be viewed as a basic multiscale algorithm in the sense that small scale features of the solution are resolved using a localized problem with higher resolution in contrast to standard adaptive finite elements where one solves a global problem on a locally refined grid. The submodeling procedure may also be iterated resulting in a multiscale zooming algorithm. The size of the submodel problem may be much smaller than the global coarse grid and may thus be solved in a short amount of time, sometimes close to real time. In complex engineering applications such a technique opens up the possibility of interactive simulation and optimization of local design changes. Another important application is simulation of the effect of small features, for instance holes, which are not present in the coarse global mesh. Removing small scale features from the CAD data in order to simplify meshing is common practice in industrial computations.



(a) Gearbox casing

(b) Zoom of gearbox casing

Fig. 1. Example of the complex geometry of a gearbox casing illustrating the presence of large and small geometric details

1.2 Contributions

In this paper we consider a submodeling procedure for linear elasticity based on displacement boundary conditions. We develop a posteriori error estimates for the submodeling procedure. The error in the submodel solution depends on the size of the submodel, the resolution of the fine submodel grid, and the resolution of the coarse grid. Our a posteriori error estimates captures these dependencies and are based on error representation formulas derived using duality techniques. We also propose adaptive algorithms for automatic tuning of the submodel resolution and size. We focus on applications of industrial interest where the local geometry of the domain in the problem determines local stress levels. We illustrate the submodeling procedure on test problems of industrial interest. For an extensive treatment we refer to [2].

1.3 Related Work

The idea of submodeling is old and is a common technique in industrial finite element computations. Recently submodeling has been exploited by Oden and Venmanganti in [10] and [12] for simulation of problems with local microstructure. Here the material model in the submodel is different from the global coarse problem where homogenized material parameters are used. We also mention the work by Xu and Zhou [13] where submodeling is used as a parallel algorithm and a priori and a posteriori error estimates are presented for the Poisson equation. Submodeling is also closely related to local error estimates for the finite element method, see Wahlbin and Schatz [11]. Estep, Holst, and Larson [4] presents a posteriori error estimates based on duality related to our estimates with application to a kind of domain decomposition algorithms. In the recent work of Larson and Målqvist [8, 9] an adaptive variational multiscale algorithm, see Hughes [6] and [7], based on parallel submodel approximation of the fine scale solution and a posteriori error estimates is proposed.

1.4 Outline

In Sect. 2 we formulate the linear elasticity equations and the finite element method. In Sect. 3 we define the submodeling procedure, derive a posteriori error estimates, and formulate an adaptive algorithm. We also present several numerical examples. Finally, in Sect. 4 we draw some conclusions of our work.

2 Linear Elasticity and Finite Element Method

2.1 Linear Elasticity

We consider the equations of linear elasticity in $d = 3$ dimensions: find the displacement $\boldsymbol{u} = [u_i]_{i=1}^d$ such that

$$-\nabla \cdot \boldsymbol{\sigma} = \boldsymbol{f} \quad \text{in } \Omega, \tag{1}$$

$$\boldsymbol{\sigma} = 2\mu\boldsymbol{\varepsilon}(\boldsymbol{u}) + \lambda\, \nabla \cdot \boldsymbol{u}\boldsymbol{I} \quad \text{in } \Omega, \tag{2}$$

$$\boldsymbol{u} = \boldsymbol{g}_D \quad \text{on } \partial\Omega_{\mathrm{D}}, \tag{3}$$

$$\boldsymbol{\sigma} \cdot \boldsymbol{n} = \boldsymbol{g}_N \quad \text{on } \partial\Omega_{\mathrm{N}}, \tag{4}$$

where $\boldsymbol{\sigma} = [\sigma_{ij}]_{i,j=1}^d$ is the symmetric stress tensor; $\boldsymbol{\varepsilon}\,(\boldsymbol{u}) = [\varepsilon_{ij}(\boldsymbol{u})]_{i,j=1}^d$ is the strain tensor with components

$$\varepsilon_{ij}(\boldsymbol{u}) = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right); \tag{5}$$

$\nabla \cdot \boldsymbol{\sigma} = \left[\sum_{j=1}^d \partial\sigma_{ij}/\partial x_j\right]_{i=1}^d$; $\boldsymbol{I} = [\delta_{ij}]_{i,j=1}^d$ with $\delta_{ij} = 1$ if $i = j$ and $\delta_{ij} = 0$ if $i \neq j$; Ω is a closed subset of $\mathbf{R}^d$ with boundary $\partial\Omega = \partial\Omega_D \cup \partial\Omega_N$, where $\partial\Omega_D$ is closed and nonempty; $\boldsymbol{f}$ and $\boldsymbol{g}_N$ are given loads; $\boldsymbol{g}_D$ is a given boundary displacement; and $\boldsymbol{n}$ is the exterior unit normal to $\partial\Omega$. Parameters λ and μ are the Lamé parameters and satisfy $0 < \mu_1 < \mu < \mu_2$ and $0 < \lambda < \infty$.

2.2 Finite Element Method

To define the finite element method we introduce a partition $\mathcal{K}_H = \{K\}$ of Ω into shape regular tetrahedra K of size H_K. We let $\mathcal{V}_H$ be the space of continuous piecewise vector polynomials of local degree p_K, in other words

$$\mathcal{V}_H = \{v \in [H^1(\Omega)]^d : v|_K = [\mathcal{P}_{p_K}(K)]^d\}, \tag{6}$$

where $\mathcal{P}_q(K)$ denotes the space of piecewise polynomials of degree q defined on element K.

The finite element method reads: find $\boldsymbol{u}_H \in \mathcal{V}_H$ such that

$$a(\boldsymbol{u}_H, \boldsymbol{v}) = l(\boldsymbol{v}) \quad \text{for all } \boldsymbol{v} \in \mathcal{V}_H, \tag{7}$$

where

$$a(\boldsymbol{u}, \boldsymbol{v}) = \int_\Omega \boldsymbol{\sigma}(\boldsymbol{u}) : \boldsymbol{\varepsilon}(\boldsymbol{v}) \, \mathrm{d}x, \tag{8}$$

$$l(\boldsymbol{v}) = \int_\Omega \boldsymbol{f} \cdot \boldsymbol{v} \, \mathrm{d}x + \int_{\partial \Gamma_N} \boldsymbol{g}_N \cdot \boldsymbol{v} \, \mathrm{d}s. \tag{9}$$

3 Adaptive Submodeling

3.1 The Submodeling Procedure

Submodeling is a technique for local enhancement of the accuracy of the finite element solution in a subdomain ω_0 of Ω based on solution of a finite element problem with a refined mesh on a slightly larger subdomain ω of Ω, see Fig. 2. More precisely we let ω be the union of a subset $\mathcal{K}_{H,\omega}$ of elements containing the domain ω_0 of interest, i.e.,

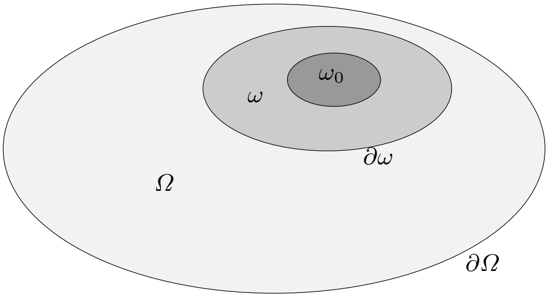


Fig. 2. Basic submodel geometry: ω_0 the domain of interest, ω the submodel with boundary $\partial\omega$, and the global domain Ω with boundary $\partial\Omega$. Note that the submodel may also intersect the global boundary $\partial\Omega$

$$\omega_0 \subset \omega = \bigcup_{K \in \mathcal{K}_{H,\omega}} K. \tag{10}$$

The precise size and shape of ω is variable and can be determined in an adaptive fashion.

We let $\mathcal{K}_\omega^h$ be a partition of the submodel ω into elements K of size h_K. Next we introduce the submodel finite element space

$$\mathcal{V}_g^h(\omega) = \{\boldsymbol{v} \in [H^1(\omega)]^d : \boldsymbol{v}|_K = [\mathcal{P}_{p_K}(K)]^d, \boldsymbol{v}|_{\partial\omega\setminus\partial\Omega_N} = \boldsymbol{g}|_{\partial\omega\setminus\partial\Omega_N}\}, \tag{11}$$

and the submodel problem: find $\boldsymbol{u}^h \in \mathcal{V}_{u_H}^h(\omega)$ such that

$$a(\boldsymbol{u}^h, \boldsymbol{v}) = l(\boldsymbol{v}) \quad \text{for all } v \in \mathcal{V}_0^h(\omega). \tag{12}$$

We define the enhanced global solution $\boldsymbol{u}_H^h$ as the combination

$$\boldsymbol{u}_H^h = \begin{cases} \boldsymbol{u}_H & \text{in } \Omega \setminus \omega, \\ \boldsymbol{u}^h & \text{in } \omega. \end{cases} \tag{13}$$

In Fig. 3 we illustrate the submodeling procedure on the mechanical part (a). A point load acts on the inside of the geometry and creates an area of high stress levels. We define a submodel containing the area of interest (b). Using mesh refinement a refined model is created (c), and von Mises effective stresses are computed in (d).

3.2 An Error Representation Formula

The error in the submodeling process has three main sources:

- Error caused by the coarse scale solution.
- Error caused by the numerical solution of the local subdomain problem.
- Error caused by restriction to a submodel problem.

We now wish to develop an a posteriori estimate of the error in a linear functional

$$m(\boldsymbol{u}) - m(\boldsymbol{u}_H^h) = m(\boldsymbol{e}) = (\boldsymbol{e}, \boldsymbol{\psi}), \tag{14}$$

where $\boldsymbol{e} = \boldsymbol{u} - \boldsymbol{u}_H^h$ is the error and $\boldsymbol{\psi}$ is a given distribution supported in the region of interest ω_0. To represent the error in the linear functional $m(\boldsymbol{e})$ we introduce the global dual problem: find $\boldsymbol{\phi} = [\phi_i]_{i=1}^d$ such that

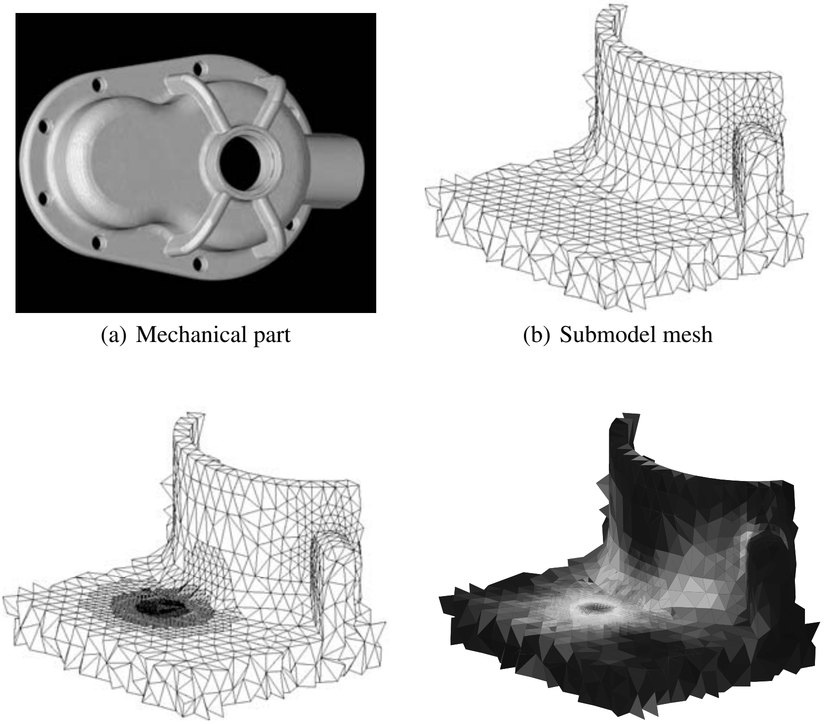
$$-\nabla \cdot \boldsymbol{\sigma} = \boldsymbol{\psi} \quad \text{in } \Omega, \tag{15}$$

$$\boldsymbol{\sigma} = 2\mu\varepsilon(\boldsymbol{\phi}) + \lambda\, \nabla \cdot \boldsymbol{\phi} \boldsymbol{I} \quad \text{in } \Omega, \tag{16}$$

$$\boldsymbol{\phi} = \boldsymbol{0} \quad \text{on } \partial\Omega_D, \tag{17}$$

$$\boldsymbol{\sigma}_{\boldsymbol{n}} = \boldsymbol{0} \quad \text{on } \partial\Omega_{\mathrm{N}}. \tag{18}$$

Multiplying (15) with the error $\boldsymbol{e}$ and using Green's formula we get



(a) Mechanical part

(b) Submodel mesh

(c) Refined submodel mesh

(d) Von Mises effective stresses on submodel

Fig. 3. Example of the submodeling procedure. A point load acts on the inside of the geometry and creates an area of high stress levels. We identify the area of interest in the mechanical part (a) and define a submodel (b) the mesh in the submodel is refined (c) and the von Mises effective stresses are computed in (d)

$$
\begin{aligned}
(\boldsymbol{e}, \boldsymbol{\psi}) &= a(\boldsymbol{e}, \boldsymbol{\phi}) && (19)\\
&= a(\boldsymbol{e}, \boldsymbol{\phi} - \boldsymbol{\pi}_H^h \boldsymbol{\phi}) + a(\boldsymbol{e}, \boldsymbol{\pi}_H^h \boldsymbol{\phi}) && (20)\\
&= \underbrace{l(\boldsymbol{\phi} - \boldsymbol{\pi}_H^h \boldsymbol{\phi}) - a(\boldsymbol{u}_H^h, \boldsymbol{\phi} - \boldsymbol{\pi}_H^h \boldsymbol{\phi})}_{I} + \underbrace{l(\boldsymbol{\pi}_H^h \boldsymbol{\phi}) - a(\boldsymbol{u}_H^h, \boldsymbol{\pi}_H^h \boldsymbol{\phi})}_{II}. && (21)
\end{aligned}
$$

Here we subtracted and added the interpolant $\boldsymbol{\pi}_H^h \boldsymbol{\phi}$ defined by

$$\boldsymbol{\pi}_H^h \boldsymbol{\phi} = \begin{cases} \boldsymbol{\pi}_H \boldsymbol{\phi} & \text{in } \Omega \setminus \omega, \\ \boldsymbol{\pi}^h \boldsymbol{\phi} & \text{in } \omega, \end{cases} \tag{22}$$

where $\boldsymbol{\pi}_H$ is the Scott–Zhang interpolation operator and $\boldsymbol{\pi}^h$ is the Scott–Zhang interpolation operator satisfying Dirichlet boundary conditions $\boldsymbol{\pi}^h \boldsymbol{\phi} = \boldsymbol{\pi}_H \boldsymbol{\phi}$ on $\partial\omega$ see [3]; and finally we used the fact that $\boldsymbol{u}$ is the exact solution to (1).

The first term I can be written

$$I = \sum_{K \in \mathcal{K}_\omega^h} (R_K(\boldsymbol{u}^h), \boldsymbol{\phi} - \boldsymbol{\pi}^h \boldsymbol{\phi}) + \sum_{K \in \mathcal{K}_H \setminus \mathcal{K}_{H,\omega}} (R_K(\boldsymbol{u}_H), \boldsymbol{\phi} - \boldsymbol{\pi}_H \boldsymbol{\phi}), \tag{23}$$

where the element residual $R_K(\boldsymbol{v}) \in H^{-1}(K)$ is defined by

$$\begin{aligned}(R_K(\boldsymbol{v}), \boldsymbol{w}) = {} & (f + \nabla \cdot \boldsymbol{\sigma}(\boldsymbol{v}), \boldsymbol{w})_K \\ & + ([\boldsymbol{\sigma}(\boldsymbol{v}) \cdot \boldsymbol{n}]/2, \boldsymbol{w})_{\partial K \setminus \partial \Omega} + (\boldsymbol{g}_N - \boldsymbol{\sigma}(\boldsymbol{v}) \cdot \boldsymbol{n}, \boldsymbol{w})_{\partial K \cap \partial \Omega_N},\end{aligned} \tag{24}$$

for all $\boldsymbol{w} \in H^1(K)$. Here $[\boldsymbol{v}(\boldsymbol{x})] = \lim_{\epsilon \to 0^+} \boldsymbol{v}(\boldsymbol{x} + \epsilon \boldsymbol{n}) - \boldsymbol{v}(\boldsymbol{x} - \epsilon \boldsymbol{n})$ denotes the jump over element interfaces in function $\boldsymbol{v}$.

The second term II can be interpreted as the jump in the variationally consistent flux $\boldsymbol{\Sigma}_n$, see [5], on the submodel boundary

$$II = l(\boldsymbol{\pi}_H^h \boldsymbol{\phi}) - a(\boldsymbol{u}_H^h, \boldsymbol{\pi}_H^h \boldsymbol{\phi}) \tag{25}$$

$$= (\boldsymbol{\Sigma}_n(\boldsymbol{u}^h) - \boldsymbol{\Sigma}_n(\boldsymbol{u}_H), \boldsymbol{\pi}_H^h \boldsymbol{\phi})_{\partial\omega}. \tag{26}$$

Collecting these expressions we obtain the error representation formula

$$\begin{aligned}(\boldsymbol{e}, \boldsymbol{\psi})_\omega = {} & \sum_{K \in \mathcal{K}_\omega^h} (R_K(\boldsymbol{u}^h), \boldsymbol{\phi} - \boldsymbol{\pi}^h \boldsymbol{\phi}) \\ & + (\boldsymbol{\Sigma}_n(\boldsymbol{u}^h) - \boldsymbol{\Sigma}_n(\boldsymbol{u}_H), \boldsymbol{\pi}_H^h \boldsymbol{\phi})_{\partial\omega} \\ & + \sum_{K \in \mathcal{K}_H \setminus \mathcal{K}_{H,\omega}} (R_K(\boldsymbol{u}_H), \boldsymbol{\phi} - \boldsymbol{\pi}_H \boldsymbol{\phi}).\end{aligned} \tag{27}$$

Here the first term accounts for the approximation of the submodel problem, the second the effect of the size of the submodel, and the third the error from the coarse grid solution.

3.3 Dual Weighted Residual A Posteriori Error Estimates

Starting from (27) and estimating the right hand side using the triangle inequality followed by the Cauchy–Schwarz inequality on an element level we obtain the following dual weighted residual estimate of the error

$$\begin{aligned}|(\boldsymbol{e},\boldsymbol{\psi})_\omega| \leq& \sum_{K\in\mathcal{K}^h} \mathcal{R}_K(\boldsymbol{u}^h)\mathcal{W}_K(\boldsymbol{\phi}) \qquad (28)\\ &+ \mathcal{R}_{\partial\omega}(\boldsymbol{u}_H^h)\mathcal{W}_{\partial\omega}(\boldsymbol{\phi})\\ &+ \sum_{K\in\mathcal{K}_H\setminus\mathcal{K}_{H,\omega}} \mathcal{R}_K(\boldsymbol{u}_H)\mathcal{W}_K(\boldsymbol{\phi})\\ =& \rho_1+\rho_2+\rho_3. \qquad (29)\end{aligned}$$

Here

$$\mathcal{R}_{\partial\omega}(\boldsymbol{u}_H^h) = \|\boldsymbol{\Sigma}_n(\boldsymbol{u}^h) - \boldsymbol{\Sigma}_n(\boldsymbol{u}_H)\|_{\partial\omega\setminus\partial\Omega_N}, \qquad (30)$$

$$\mathcal{W}_{\partial\omega}(\boldsymbol{\phi}) = \|\boldsymbol{\pi}_H^h\boldsymbol{\phi}\|_{\partial\omega\setminus\partial\Omega_N}, \qquad (31)$$

and the element residual $\mathcal{R}_K(\cdot)$ and the weight $\mathcal{W}_K(\cdot)$ are estimates of the residual and the local interpolation error in the solution $\boldsymbol{\phi}$ to the dual problem defined by

$$\begin{aligned}\mathcal{R}_K^2(\boldsymbol{v}) &= \|f + \nabla\cdot\boldsymbol{\sigma}(\boldsymbol{v})\|_K^2 + h_K^{-1}\|[\boldsymbol{\sigma}(\boldsymbol{v})\cdot\boldsymbol{n}]\|_{\partial K\setminus\partial\Omega}^2 \qquad (32)\\ &\quad + h_K^{-1}\|\boldsymbol{g}_N - \boldsymbol{\sigma}(\boldsymbol{v})\cdot\boldsymbol{n}\|_{\partial K\cap\partial\Omega_N}^2,\\ \mathcal{W}_K^2(\boldsymbol{\phi}) &= \|\boldsymbol{\phi} - \boldsymbol{\pi}_H^h\boldsymbol{\phi}\|_K^2 + h_K\|\boldsymbol{\phi} - \boldsymbol{\pi}_H^h\boldsymbol{\phi}\|_{\partial K\setminus\partial\Omega_D}^2, \qquad (33)\end{aligned}$$

where we scaled the edge residuals and weights by suitable powers of the local mesh size h_K in order for all contributions to the residual and weight to have the same dependence of h_K.

3.4 Simplified A Posteriori Error Estimates

Using interpolation theory, see [3], the weights can be estimated in terms of the local sizes of derivatives of the dual problem

$$\mathcal{W}_K(\boldsymbol{\phi}) \leq Ch_K^\alpha|\boldsymbol{\phi}|_{\mathcal{N}(K),\alpha}, \qquad (34)$$

where $\mathcal{N}(K)$ denotes the set of elements neighboring K. If $\boldsymbol{\psi}\in H^{-1}(\omega_0)$ we expect $\boldsymbol{\phi}|_\omega \in H^1(\omega)$ and if the boundary of Ω is smooth we expect $\boldsymbol{\phi}|_{\Omega\setminus\omega}\in H^2(\Omega)$. With these assumptions we obtain the simplified estimate

$$|(\boldsymbol{e},\boldsymbol{\psi})_{\omega_0}| \leq C\left(\sum_{K\in\mathcal{K}_\omega^h} h_K\mathcal{R}_K(\boldsymbol{u}^h) + \mathcal{R}_{\partial\omega}(\boldsymbol{u}_H^h) + \sum_{K\in\mathcal{K}_H\setminus\mathcal{K}_{H,\omega}} H_K^2\mathcal{R}_K(\boldsymbol{u}_H)\right), \qquad (35)$$

which can be used as a basis for an adaptive algorithm if an approximation of the dual problem is not available. Note that there are no powers of the meshsize multiplying $\mathcal{R}_{\partial\omega}(\boldsymbol{u}_H^h)$ since this residual measures the deviation from global Galerkin orthogonality.

There are of course many variants of these estimates and we have chosen to present two of the most basic approaches. We refer to [1] for an introduction to the dual weighted residuals method.

3.5 Representation of Errors in Stresses and Strains

The data $\boldsymbol{\psi}$ to the dual problem (15) defines the measure of the error. In solid mechanics one is often interested in different localized measures of the stresses or the strains. We let

$$\delta_{x,\gamma} \tag{36}$$

denote a continuous approximation of the Dirac delta function at x with regularization parameter γ. We then have $(\partial_j \boldsymbol{e}, \delta_{x,\gamma}) = (\boldsymbol{e}, -\partial_j \delta_{x,\gamma})$. Letting $\{\boldsymbol{\xi}_i\}_{i=1}^3$ be the standard orthogonal coordinate system in $\mathbf{R}^3$ we see that with

$$\boldsymbol{\psi}_{\epsilon_{ij}} = -(\boldsymbol{\xi}_i \partial_j \delta_{x,\gamma} + \boldsymbol{\xi}_j \partial_j \delta_{x,\gamma})/2, \tag{37}$$

we obtain the representation formula

$$(\epsilon_{ij}(\boldsymbol{u}) - \epsilon_{ij}(\boldsymbol{u}_H^h), \delta_{x,\gamma}) = (\boldsymbol{e}, \boldsymbol{\psi}_{\epsilon_{ij}}), \tag{38}$$

for the error in the strains. To achieve error estimates for the elements in the stress tensor we recall the constitutive relation $\sigma_{ij} = 2\mu\epsilon_{ij} + \lambda tr(\varepsilon)\delta_{ij}$. Setting

$$\boldsymbol{\psi}_{\sigma_{ij}} = 2\mu\boldsymbol{\psi}_{\epsilon_{ij}} + \lambda\Big(\sum_{k=1}^{3} \boldsymbol{\psi}_{\epsilon_{kk}}\Big)\delta_{ij}, \tag{39}$$

we obtain the representation formula

$$(\sigma_{ij}(\boldsymbol{u}) - \sigma_{ij}(\boldsymbol{u}_H^h), \delta_{x,\gamma}) = (\boldsymbol{e}, \boldsymbol{\psi}_{\sigma_{ij}}), \tag{40}$$

for the error in stresses. Furthermore, the principal stresses σ_i satisfies the identity $\boldsymbol{\eta}_i \cdot \boldsymbol{\sigma}\boldsymbol{\eta}_i = \sigma_i$ with $\boldsymbol{\eta}_i$ the principal stress unit vectors. Setting

$$\boldsymbol{\psi}_{\sigma_i} = \sum_{j,k=1}^{3} \boldsymbol{\psi}_{\sigma_{jk}} \eta_{i,j}\eta_{i,k}, \tag{41}$$

we obtain the linearized error representation formula

$$(\sigma_i(\boldsymbol{u}) - \sigma_i(\boldsymbol{u}_H^h), \delta_{x,\gamma}) \approx (\boldsymbol{e}, \boldsymbol{\psi}_{\sigma_{ij}}). \tag{42}$$

In practice we use the numerical approximations of the unit principal stress vectors. Using the representation formulas for the principal stresses together with linearization we can derive similar representation formulas for the von Mises effective stresses.

In Figs. 4 and 5 we show the element weights for the dual solution corresponding to control of a localized mean value of the principal stress σ_1.

We have chosen to scale the weights with h_K^{-2} to get a quantity which is more independent of the meshsize h_K, see the discussion on the simplified a posteriori error estimate (35) above. Note, in particular, the small area where the weight is large indicating that submodeling can be successful.

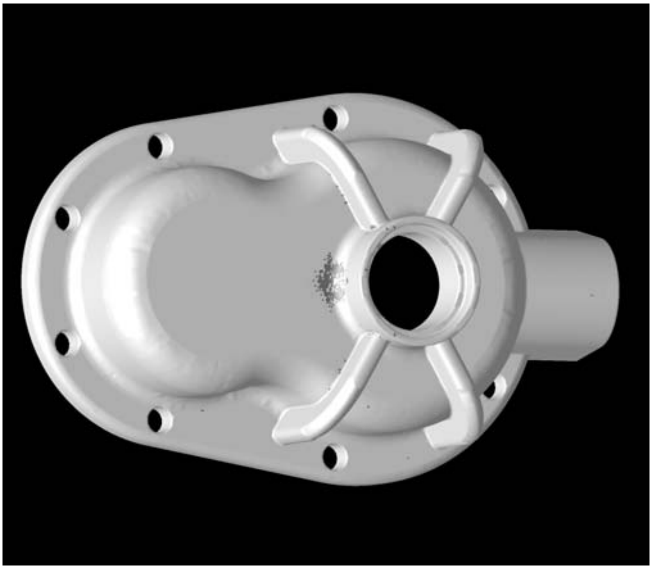


Fig. 4. The scaled dual element weights $\mathcal{W}_K/h_K^2$

3.6 An Adaptive Submodeling Algorithm

Based on the a posteriori error estimate (29), we propose the following algorithm for adaptive choice of the meshsize in the subdomain and the size of the subdomain ω.

Algorithm:

Given a coarse grid global approximation $\boldsymbol{u}_H$ of the displacement field $\boldsymbol{u}$ and a subdomain $\omega_0 \subset \Omega$ of interest:

- Compute an initial approximation of the subdomain ω.
- Solve the subdomain dual problem.
- If $\rho_1 > \rho_3/2$ refine the submodel mesh using a standard adaptive mesh refinement algorithm.
- If $\rho_2 > \rho_3/2$ increase the size of the subdomain ω.
- Repeat until $\rho_1 + \rho_2 \leq \rho_3$ or the submodel problem is too large.

In practice we can of course not solve the global dual problem, instead a localized approximation is computed on the submodel or a slightly larger patch using homogeneous Dirichlet conditions on the boundary, see [2] for further details.

4 Conclusions

We have described the submodeling procedure for local enhancement of the accuracy in finite element stress computations. The error in the submodel stresses can be

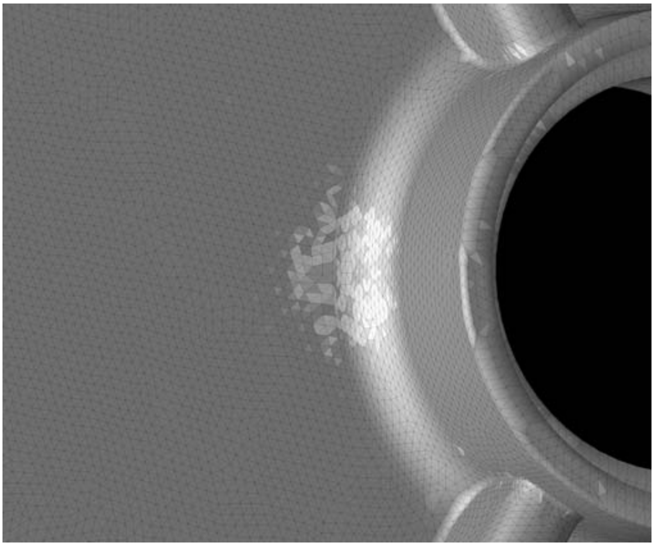


Fig. 5. Zoom of the scaled dual element weights $\mathcal{W}_K/h_K^2$

estimated using a posteriori error estimates which are used as a basis for design of adaptive algorithms for tuning of critical parameters such as the size of the subdomain and the resolution of the submodel mesh size. Submodeling is a fundamental building block in the construction of multiscale algorithms where information from the submodel solution is fed back to the coarse grid to account for fine scale effects. Such algorithms are currently under investigation.

Acknowledgment

This work has been supported by EC-project ViSiCADE.

References

1. Wolfgang Bangerth and Rolf Rannacher. *Adaptive finite element methods for differential equations*. Lectures in Mathematics ETH Zürich. Birkhäuser Verlag, Basel, 2003.
2. Mats G. Larson, Fredrik Bengzon and August Johansson. Adaptive submodeling for elasticity. Preprint to appear 2005, Chalmers Finite Element Center, Göteborg, Sweden, `www.phi.chalmers.se`
3. Susanne C. Brenner and L. Ridgway Scott. *The mathematical theory of finite element methods*, volume 15 of *Texts in Applied Mathematics*. Springer-Verlag, New York, second edition, 2002.
4. Donald Estep, Michael Holst, and Mats G. Larson. Generalized Greens functions and the effective domain of influence. Preprint 10, Chalmers Finite Element Center, Göteborg, Sweden, 2003. to appear in SIAM J. Sci. Comp.

5. Thomas J. R. Hughes, Gerald Engel, Luca Mazzei, and Mats G. Larson. The continuous Galerkin method is locally conservative. *J. Comput. Phys.*, 163(2):467–488, 2000.
6. Thomas J. R. Hughes, Gonzalo R. Feijóo, Luca Mazzei, and Jean-Baptiste Quincy. The variational multiscale method—a paradigm for computational mechanics. *Comput. Methods Appl. Mech. Engrg.*, 166(1-2):3–24, 1998.
7. Thomas J. R. Hughes and Assad A. Oberai. The variational multiscale formulation of LES with application to turbulent channel flows. In *Geometry, mechanics, and dynamics*, pages 223–239. Springer, New York, 2002.
8. Mats G. Larson and Axel Målqvist. Adaptive variational multiscale methods based on a posteriori error estimation: energy norm estimates for elliptic problems. Preprint 2004–18, Chalmers Finite Element Center, Göteborg, Sweden, `www.phi.chalmers.se`
9. Mats G. Larson and Axel Målqvist. Adaptive variational multiscale methods based on a posteriori error estimation. Proceedings of ECCOMAS 2004 conference, Jyväskylä, Finland.
10. J. Tinsley Oden and Kumar S. Vemaganti. Estimation of local modeling error and goal-oriented adaptive modeling of heterogeneous materials. I. Error estimates and adaptive algorithms. *J. Comput. Phys.*, 164(1):22–47, 2000.
11. A. H. Schatz and L. B. Wahlbin. Interior maximum-norm estimates for finite element methods. II. *Math. Comp.*, 64(211):907–928, 1995.
12. Kumar S. Vemaganti and J. Tinsley Oden. Estimation of local modeling error and goal-oriented adaptive modeling of heterogeneous materials. II. A computational environment for adaptive modeling of heterogeneous elastic solids. *Comput. Methods Appl. Mech. Engrg.*, 190(46-47):6089–6124, 2001.
13. Jinchao Xu and Aihui Zhou. Local and parallel finite element algorithms based on two-grid discretizations. *Math. Comp.*, 69(231):881–909, 2000.

Adaptive Variational Multiscale Methods Based on A Posteriori Error Estimation: Duality Techniques for Elliptic Problems

Mats G. Larson[1] and Axel Målqvist[2]

[1] Department of Mathematics, Umeå University, 90187 Umeå, Sweden
mats.larson@math.umu.se

[2] Department of Mathematics, Chalmers University of Technology, S-412 96, Göteborg, Sweden
axel@math.chalmers.se

Summary. The variational multiscale method (VMM) provides a general framework for construction of multiscale finite element methods. In this paper we propose a method for parallel solution of the fine scale problem based on localized Dirichlet problems which are solved numerically. Next we present a posteriori error representation formulas for VMM which relates the error in linear functionals to the discretization errors, resolution and size of patches in the localized problems, in the fine scale approximation. These formulas are derived by using duality techniques. Based on the a posteriori error representation formula we propose an adaptive VMM with automatic tuning of the critical parameters. We primarily study elliptic second order partial differential equations with highly oscillating coefficients or localized singularities.

Key words: variational multiscale method, a posteriori error estimation, duality techniques, finite element method, elliptic problems

1 Introduction

Many problems in science and engineering involve models of physical systems on many scales. For instance, models of materials with microstructure such as composites and flow in porous media. In such problems it is in general not feasible to seek for a numerical solution which resolves all scales. Instead we may seek to develop algorithms based on a suitable combination with a global problem capturing the main features of the solution and localized problems which resolves the fine scales. Since the fine scale problems are localized the computation on the fine scales is parallel in nature.

Previous Work

The Variational Multiscale Method (VMM) is a general framework for derivation of basic multiscale method in a variational context, see Hughes [7] and [9]. The basic idea is to decompose the solution into fine and coarse scale contributions, solve the fine scale equation in terms of the residual of the coarse scale solution, and finally eliminate the fine scale solution from the coarse scale equation. This procedure leads to a modified coarse scale equation where the modification accounts for the effect of fine scale behavior on the coarse scales. In practice it is necessary to approximate the fine scale equation to make the method realistic. In several works various ways of analytical modeling are investigated often based on bubbles or element Green's functions, see Oberai and Pinsky, [11] and Arbogast [1]. In [6] Hou and Wu present a different approach. Here the fine scale equations are solved numerically on a finer mesh. The fine scale equations are solved inside coarse elements and are thus totally decoupled.

New Contributions

In this work we present a simple technique for numerical approximation of the fine scale equation in the variational multiscale method. The basic idea is to split the fine scale residual into localized contributions using a partition of unity and solving corresponding decoupled localized problems on patches with homogeneous Dirichlet boundary conditions. The fine scale solution is approximated by the sum $U_f = \sum_i U_{f,i}$ of the solutions $U_{f,i}$ to the localized problems. The accuracy of U_f depends on the fine scale mesh size h and the size of the patches. We note that the fine scale computation is naturally parallel.

To optimize performance we seek to construct an adaptive algorithm for automatic control of the coarse mesh size H, the fine mesh size h, and the size of patches. Our algorithm is based on the following a posteriori estimate of the error $e = u - U_c - U_f$ for the Poisson equation with variable coefficients a:

$$(e, \psi) = \sum_{i \in \mathcal{C}} (\varphi_i R(U_c), \phi_f) + \sum_{i \in \mathcal{F}} \left((\varphi_i R(U_c), \phi_f)_{\omega_i} - a(U_{f,i}, \phi_f) \right), \qquad (1)$$

where $\psi \in H^{-1}(\Omega)$, $\mathcal{C}$ refers to nodes where no local problems have been solved, $\mathcal{F}$ to nodes where local problems are solved, U_c is the coarse scale solution, $U = U_c + U_f$, $R(U) = f + \nabla \cdot a \nabla U$ is the residual, $\{\varphi_i\}_{i \in \mathcal{C} \cup \mathcal{F}}$ are coarse base functions, and ϕ_f is the fine scale part of a dual solution driven by ψ.

If no fine scale equations are solved we only obtain the first term in the estimate. The second term relates the fine scale mesh parameter h to the patch size ω_i on which the local problems are solved. We have derived a similar estimate for the error in energy norm, see [10].

The framework is fairly general and may be extended to other types of multiscale methods, for instance, based on localized Neumann problems.

Outline.

First we introduce the model problem and the variational multiscale formulation of this problem, we also discuss the split of the coarse and fine scale spaces. In the following section we present a posteriori estimates of the error. These results leads to an adaptive algorithm. We present numerical results and finally we present concluding remarks and suggestions on future work.

2 The Variational Multiscale Method

2.1 Model Problem

We study the Poisson equation with a highly oscillating coefficient a and homogeneous Dirichlet boundary conditions: find $u \in H_0^1(\Omega)$ such that

$$-\nabla \cdot a\nabla u = f \quad \text{in } \Omega, \tag{2}$$

where Ω is a polygonal domain in $\mathbf{R}^d$, $d = 1, 2$, or 3 with boundary Γ, $f \in H^{-1}(\Omega)$, and $a \in L^\infty(\Omega)$ such that $a(x) \geq \alpha_0 > 0$ for all $x \in \Omega$. The variational form of (2) reads: find $u \in \mathcal{V} = H_0^1(\Omega)$ such that

$$a(u, v) = (f, v) \quad \text{for all } v \in \mathcal{V}, \tag{3}$$

with the bilinear form

$$a(u, v) = (a\nabla u, \nabla v) \tag{4}$$

for all $u, v \in \mathcal{V}$.

2.2 The Variational Multiscale Method

We employ the variational multiscale scale formulation, proposed by Hughes see [7, 9] for an overview, and introduce a coarse and a fine scale in the problem. We choose two spaces $\mathcal{V}_c \subset \mathcal{V}$ and $\mathcal{V}_f \subset \mathcal{V}$ such that

$$\mathcal{V} = \mathcal{V}_c \oplus \mathcal{V}_f. \tag{5}$$

Then we may pose (3) in the following way: find $u_c \in \mathcal{V}_c$ and $u_f \in \mathcal{V}_f$ such that

$$\begin{aligned} a(u_c, v_c) + a(u_f, v_c) &= (f, v_c) \quad \text{for all } v_c \in \mathcal{V}_c, \\ a(u_c, v_f) + a(u_f, v_f) &= (f, v_f) \quad \text{for all } v_f \in \mathcal{V}_f. \end{aligned} \tag{6}$$

Introducing the residual $R : \mathcal{V} \to \mathcal{V}'$ defined by

$$(R(v), w) = (f, w) - a(v, w) \quad \text{for all } w \in \mathcal{V}, \tag{7}$$

the fine scale equation takes the form: find $u_f \in \mathcal{V}_f$ such that

$$a(u_f, v_f) = (R(u_c), v_f) \quad \text{for all } v_f \in \mathcal{V}_f. \tag{8}$$

Thus the fine scale solution is driven by the residual of the coarse scale solution. Denoting the solution u_f to (8) by $u_f = TR(u_c)$ we get the modified coarse scale problem

$$a(u_c, v_c) + a(TR(u_c), v_c) = (f, v_c) \quad \text{for all } v_c \in \mathcal{V}_c. \tag{9}$$

Here the second term on the left hand side accounts for the effects of fine scales on the coarse scales.

2.3 A VMM Based on Localized Dirichlet Problems

We introduce a partition $\mathcal{K} = \{K\}$ of the domain Ω into shape regular elements K of diameter H_K and we let $\mathcal{N}$ be the set of nodes. Further we let $\mathcal{V}_c$ be the space of continuous piecewise polynomials of degree p defined on $\mathcal{K}$.

We shall now construct an algorithm which approximates the fine scale equation by a set of decoupled localized problems. We begin by writing $u_f = \sum_{i \in \mathcal{N}} u_{f,i}$ where

$$a(u_{f,i}, v_f) = (\varphi_i R(u_c), v_f) \quad \text{for all } v_f \in \mathcal{V}_f, \tag{10}$$

and $\{\varphi_i\}_{i \in \mathcal{N}}$ is the set of Lagrange basis functions in $\mathcal{V}_c$. Note that $\{\varphi_i\}_{i \in \mathcal{N}}$ is a partition of unity with support on the elements sharing the node i. We call the set of elements with one corner in node i a mesh star in node i and denote it S_1^i. Thus functions $u_{f,i}$ correspond to the fine scale effects created by the localized residuals $\varphi_i R(u_c)$. Introduce this expansion of u_f in the right hand side of the fine scale equation (6) and get: find $u_c \in \mathcal{V}_c$ and $u_f = \sum_{i \in \mathcal{N}} u_{f,i} \in \mathcal{V}_f$ such that

$$\begin{aligned} a(u_c, v_c) + a(u_f, v_c) &= (f, v_c) \quad \text{for all } v_c \in \mathcal{V}_c, \\ a(u_{f,i}, v_f) &= (\varphi_i R(u_c), v_f) \quad \text{for all } v_f \in \mathcal{V}_f \text{ and } i \in \mathcal{N}. \end{aligned} \tag{11}$$

We use this fact to construct a finite element method for solving (11) approximately in two steps.

- For each coarse node we define a patch ω_i such that $\text{supp}(\varphi_i) \subset \omega_i \subset \Omega$. We denote the boundary of ω_i by $\partial\omega_i$.
- On these patches we define piecewise polynomial spaces $\mathcal{V}_f^h(\omega_i)$ with respect to a fine mesh with mesh function $h = h(x)$ defined as a piecewise constant function on the fine mesh. Functions in $\mathcal{V}_f^h(\omega_i)$ are equal to zero on $\partial\omega_i$.

The resulting method reads: find $U_c \in \mathcal{V}_c$ and $U_f = \sum_i^n U_{f,i}$ where $U_{f,i} \in \mathcal{V}_f^h(\omega_i)$ such that

$$\begin{aligned} a(U_c, v_c) + a(U_f, v_c) &= (f, v_c) \quad \text{for all } v_c \in \mathcal{V}_c, \\ a(U_{f,i}, v_f) &= (\varphi_i R(U_c), v_f) \quad \text{for all } v_f \in \mathcal{V}_f^h(\omega_i) \text{ and } i \in \mathcal{N}. \end{aligned} \tag{12}$$

Since the functions in the local finite element spaces $\mathcal{V}_f^h(\omega_i)$ are equal to zero on $\partial\omega_i$, U_f and therefore $U = U_c + U_f$ will be continuous.

Remark 1. For problems with multiscale phenomena on a part of the domain it is not necessary to solve local problems for all coarse nodes. We let $\mathcal{C} \subset \mathcal{N}$ refer to nodes where no local problems are solved and $\mathcal{F} \subset \mathcal{N}$ refer to nodes where local problems are solved. Obviously $\mathcal{C} \cup \mathcal{F} = \mathcal{N}$. We let $U_{f,i} = 0$ for $i \in \mathcal{C}$.

Remark 2. The choice of the subdomains ω_i is crucial for the method. We introduce a notation for extended stars of many layers of coarse elements recursively in the following way. The extended mesh star $S_L^i = \cup_{j \in S_{L-1}^i} S_1^j$ for $L > 1$. We refer to L as layers, see Fig. 1.

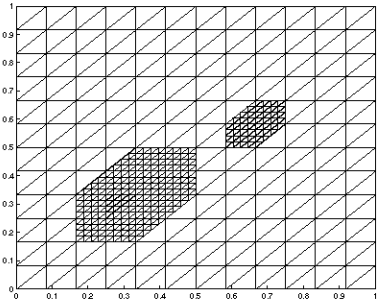


Fig. 1. Two (left) and one (right) layer stars

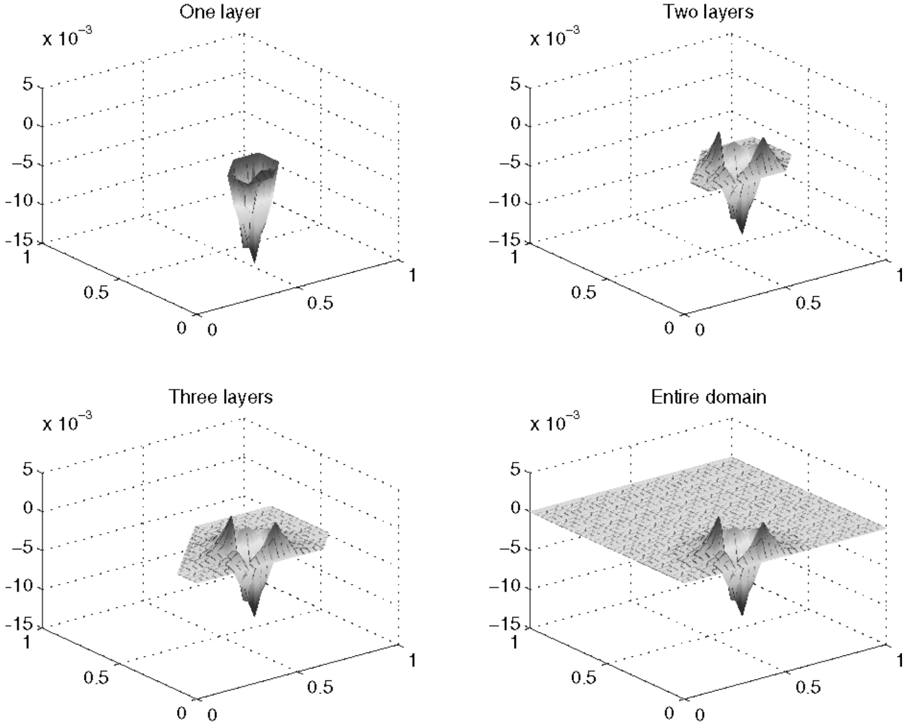


Fig. 2. The fine scale solution $U_{f,i}$ for different patches

In Fig. 2 we plot solutions to localized fine scale problems $U_{f,i}$ on different patches. We note how $U_{f,i}$ decays rapidly outside the support of φ_i. It appears to be enough to use two layers in this example to capture the true behaviour of the fine scale solution.

2.4 Subspaces

The choice of the fine scale space $\mathcal{V}_f$ can be done in different ways. In a paper by Aksoylu and Holst [4] three suggestions are made.

Hierarchical Basis Method

The first and perhaps easiest approach is to let $\mathcal{V}_f = \{v \in \mathcal{V} : v(x_j) = 0, j = \mathcal{N}\}$, where $\{x_i\}_{i\in\mathcal{N}}$ are the coarse mesh nodes. When $\mathcal{V}_f$ is discretized by the standard piecewise polynomials on the fine mesh this means that the fine scale base functions will have support on fine scale stars.

BPX Preconditioner

The second approach is to let $\mathcal{V}_f$ be $L^2(\Omega)$ orthogonal to $\mathcal{V}_c$. In this case we will have global support for the fine scale base functions but for the discretized space we have rapid decay outside fine mesh stars.

Wavelet Modified Hierarchical Basis Method

The third choice is a mix of the other two. The fine scale space $\mathcal{V}_f$ is defined as an approximate $L^2(\Omega)$ orthogonal version of the Hierarchical basis method. We let $Q_c^a v \in \mathcal{V}_c$ be an approximate solution (a small number of Jacobi iterations) to

$$(Q_c^a v, w) = (v, w), \quad \text{for all } w \in \mathcal{V}_c. \tag{13}$$

and define the Wavelet modified hierarchical basis function associated with the hierarchical basis function φ_{HB} to be,

$$\varphi_{WHB} = (I - Q_c^a)\varphi_{HB}, \tag{14}$$

see Fig. 3.

For an extended description of these methods see [3, 4, 2]. In this paper we focus on the WHB method.

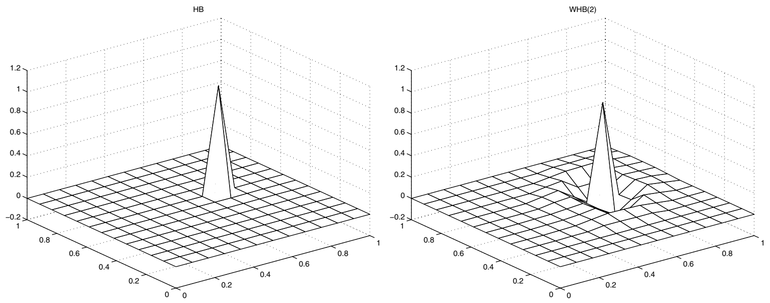


Fig. 3. HB-function and WHB-function with two Jacobi iterations

3 A Posteriori Error Estimates

3.1 The Dual Problem

To derive a posteriori error estimates of the error in a given linear functional (e,ψ) with $e = u - U$ and $\psi \in H^{-1}(\Omega)$ a given weight. We introduce the following dual problem: find $\phi \in \mathcal{V}$ such that

$$a(v,\phi) = (v,\psi) \quad \text{for all } v \in \mathcal{V}. \tag{15}$$

In the VMM setting this yields: find $\phi_c \in \mathcal{V}_c$ and $\phi_f \in \mathcal{V}_f$ such that

$$\begin{aligned} a(v_c,\phi_c) + a(v_c,\phi_f) &= (v_c,\psi), \quad \text{for all } v_c \in \mathcal{V}_c, \\ a(v_f,\phi_f) + a(v_f,\phi_c) &= (v_f,\psi), \quad \text{for all } v_f \in \mathcal{V}_f. \end{aligned} \tag{16}$$

3.2 Error Representation Formula

We now derive an error representation formula involving both the coarse scale error $e_c = u_c - U_c$ and the fine scale error $e_f = \sum_{i\in\mathcal{N}} e_{f,i} := \sum_{i\in\mathcal{N}}(u_{f,i} - U_{f,i})$ that arises from using our finite element method (12).

We use the dual problem (16) to derive an a posteriori error estimate for a linear functional of the error $e = e_c + e_f$. If we subtract the coarse part of equation (12) from the coarse part of equation (11) we get the Galerkin orthogonality,

$$a(e_c,v_c) + a(e_f,v_c) = 0 \quad \text{for all } v_c \in \mathcal{V}_c. \tag{17}$$

The same argument on the fine scale equation gives for $i \in \mathcal{F}$,

$$a(e_{f,i},v_f) = -a(e_c,\varphi_i v_f), \quad \text{for all } v_f \in \mathcal{V}_f^h(\omega_i). \tag{18}$$

We are now ready to state an error representation formula.

Theorem 1. *If $\psi \in H^{-1}(\Omega)$ then,*

$$(e,\psi) = \sum_{i\in\mathcal{C}} (\varphi_i R(U_c),\phi_f) + \sum_{i\in\mathcal{F}} \left((\varphi_i R(U_c),\phi_f)_{\omega_i} - a(U_{f,i},\phi_f)\right). \tag{19}$$

Proof. Starting from the definition of the dual problem and letting $v = e = u - U_c - U_f$ we get

$$\begin{aligned} (e,\psi) &= a(e,\phi) && (20)\\ &= a(e,\phi_f) && (21)\\ &= a(u - U_c,\phi_f) - a(U_f,\phi_f) && (22)\\ &= (R(U_c),\phi_f) - a(U_f,\phi_f) && (23)\\ &= (R(U_c),\phi_f) - \sum_{i\in\mathcal{F}} a(U_{f,i},\phi_f) && (24)\\ &= \sum_{i\in\mathcal{C}} (\varphi_i R(U_c),\phi_f) + \sum_{i\in\mathcal{F}} (\varphi_i R(U_c),\phi_f) - a(U_{f,i},\phi_f). && (25) \end{aligned}$$

which proves the theorem.

Since equation (12) holds we can subtract functions $v^h_{f,i} \in \mathcal{V}^h_f(\omega_i)$ where $i \in \mathcal{F}$ from equation (25). For example we choose $v^h_{f,i} = \pi_{h,i}\phi_f$, where $\pi_{h,i}\phi_f$ is the Scott–Zhang interpolant, see [5], of ϕ_f onto $\mathcal{V}^h_f(\omega_i)$ to get

$$\begin{aligned}(e,\psi) = &\sum_{i\in\mathcal{C}}(\varphi_i R(U_c),\phi_f) \\ &+ \sum_{i\in\mathcal{F}}\left((\varphi_i R(U_c),\phi_f - \pi_{h,i}\phi_f)_{\omega_i} - a(U_{f,i},\phi_f - \pi_{h,i}\phi_f)\right).\end{aligned} \tag{26}$$

Remark 3. Since the dual problem defined in equation (16) is equally hard to solve as the primal problem we need to solve it numerically as well. Normally it would not be sufficient to solve the dual problem with the same accuracy as the primal due to the Galerkin Orthogonality. However in this setting things are a bit different. Calculating ϕ_f with minimum refinement (one time) on the local problems for $i \in \mathcal{N}$ will not result in an error (e,ψ) equal to zero. The important thing is to only store the part of ϕ_f with support on ω_i when calculating term i in the sum of equation (19). The entire function ϕ_f might be hard to store in the memory of the computer.

4 Adaptive Algorithm

We use the error representation formula in Theorem 1 to construct an adaptive algorithm. We remember the result,

$$(e,\psi) = \sum_{i\in\mathcal{C}}(\varphi_i R(U_c),\phi_f) + \sum_{i\in\mathcal{F}}\left((\varphi_i R(U_c),\phi_f)_{\omega_i} - a(U_{f,i},\phi_f)\right). \tag{27}$$

The first sum of the error representation formula is very similar to what we would get from using standard Galerkin on the coarse mesh. The function $\phi_f = \phi - \phi_c \sim H\nabla\phi$ which is exactly what we would expect. For the second sum we have an extra orthogonality namely that from equation (26). We have $\phi_f - \pi_{h,i}\phi_f \sim h\nabla\phi$ if the patches $\omega_i = \Omega$ i.e. we get the fine scale convergence. But in practice the patches are much smaller so we end up somewhere in between h and H convergence. To sum up this discussion there are three parameters of interest that need to be considered in an adaptive algorithm, H, h, and the size of the patches.

Adaptive Algorithm.

1. Start with no nodes in $\mathcal{F}$.
2. Calculate the primal U_c.
3. Calculate the dual solution locally ϕ_f with low accuracy for all coarse nodes. (ϕ_f does not need to be solved very accurately to point out the correct nodes for local calculations.)
4. Calculate the contributions to the error for each coarse node, $C_i = (\varphi_i R(U_c),\phi_f)$.
5. Solve local problems where C_i is large to get a new U_c.

6. Calculate C_i and $F_i = ((\varphi_i R(U_c), \phi_f)_{\omega_i} - a(U_{f,i}, \phi_f))$, for large values in C_i solve more local problems and for large values in F_i either increase the number of layers or decrease the fine scale mesh size h for local problem i. Stop if the desired tolerance is reached or go to 2.

5 Numerical Examples

We solve two dimensional model problems with linear base functions defined on a uniform triangular mesh.

Example 1.

In this example we demonstrate how we can get highly improved accuracy in one part of the domain by choosing the load in the dual problem ψ equal to the indicator function for this domain. We consider the unit square with a crack in the form of a plus sign on which the solution is forced to be zero, see Fig. 4 (left). We let ψ be equal to one in the lower left quadrant (marked with a thin lattice in the figure) and zero in the rest of the domain. To the right in Fig. 4 we see a reference solution to the Poisson equation with $a = f = 1$ and homogeneous Dirichlet boundary conditions on this geometry. The idea is to use the adaptive algorithm to choose which areas that needs to be solved with higher accuracy. In Fig. 5 we plot the dual solution, with ψ chosen as described above, to the left and the fine scale part of the dual solution to the right. After two iterations in the adaptive algorithm we see clearly that local problems have only nodes in the lower left corner. In Fig. 6 the small circles refer to fine scale problems solved with two layer stars and the bigger circles refer to fine scale problems solved with three layer stars. The improvement in the solution after two iterations in the adaptive algorithm is very clear. In Fig. 7 we compare the

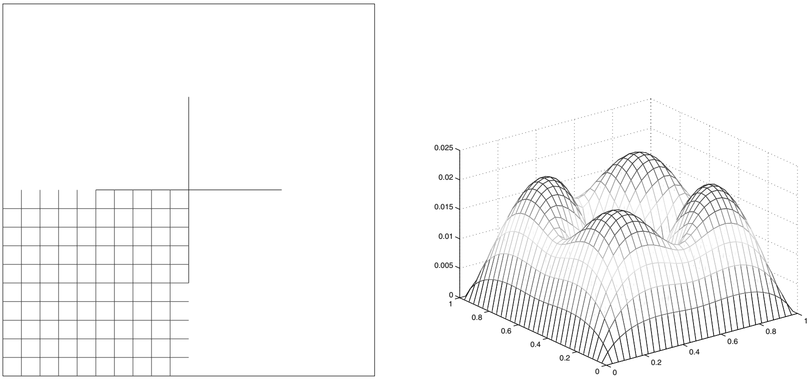


Fig. 4. Geometry (left) and Reference solution (right)

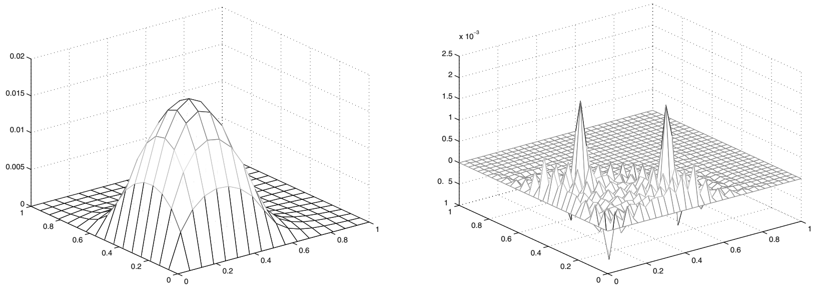


Fig. 5. Dual solution (left) and fine scale part of the dual solution (right) calculated with $\psi = I_{\{0 \leq x, y \leq 0.5\}}$

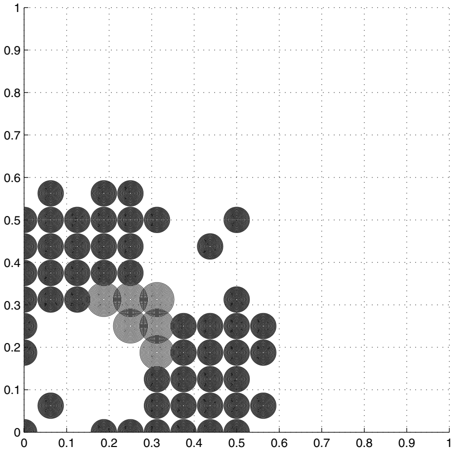


Fig. 6. Local problems solved with two and three layer stars

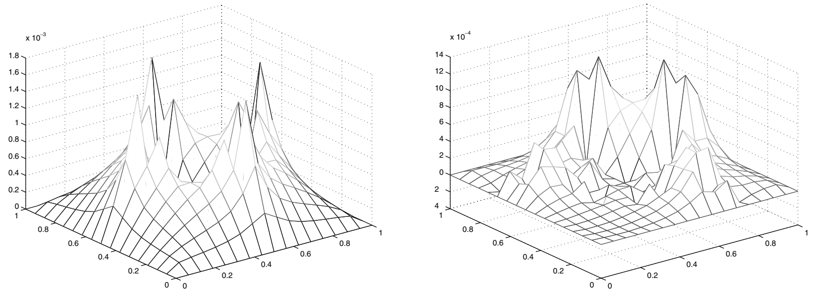


Fig. 7. Galerkin error (left) and adaptive variational multiscale method error (right)

standard Galerkin solution and the adaptive solution to a reference solution. We see how the error in the lower left quadrant is much smaller but the error in the rest of the domain is very similar to the standard Galerkin error.

Example 2.

Next we turn our attention to a model problem with oscillating coefficient a in a part of the domain, see Fig. 8. In this example we choose $f = \psi = 1$ which makes the primal and the dual equivalent. In Fig. 9 we note that the adaptive algorithm automatically picks nodes in the left part of the domain for local problems to increase accuracy. In the first example we want to refine a certain part of the domain and therefore we choose ψ in order to do so, here we want good accuracy on the whole domain and the adaptive algorithm chooses where to refine automatically. Again we compare standard Galerkin and our solution to a reference solution calculated on a finer mesh. The result can be seen in Fig. 10. Again we see a nice improvement compared to the standard Galerkin error. The choice $\psi = 1$ indicates control of the mean of the error over the domain.

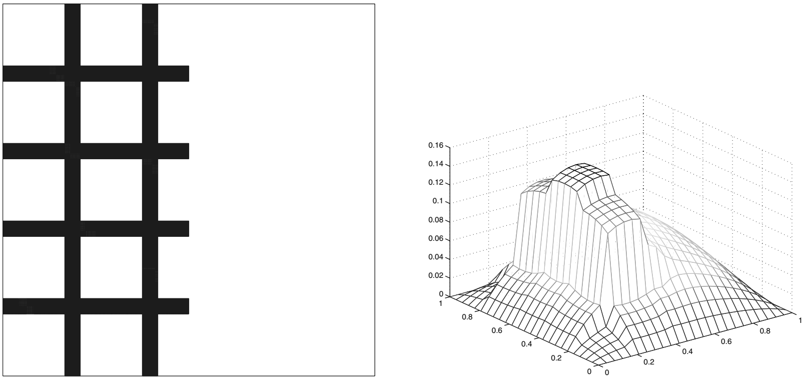


Fig. 8. The coefficient $a = 1$ on the white parts and $a = 0.05$ on the lattice (left) and reference solution on this geometry (right)

6 Conclusions and Future Work

We have presented a method for parallel solution of the fine scale equations in the variational multiscale method based on solution of localized Dirichlet problems on patches and developed an a posteriori error analysis for the method. Based on the estimates we design a basic adaptive algorithm for automatic tuning of the critical parameters: resolution and size of patches in the fine scale problems. It is also possible to decide whether a fine scale computation is necessary or not and thus the

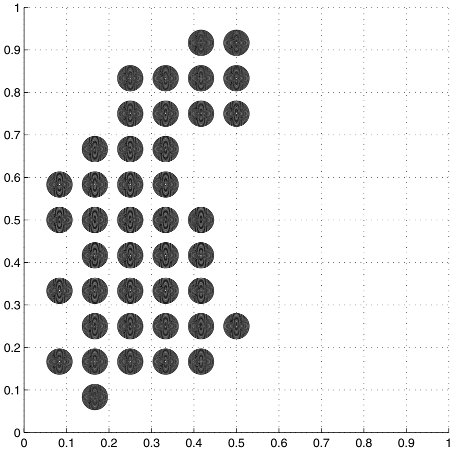


Fig. 9. The dots marks coarse nodes where local problems have been solved

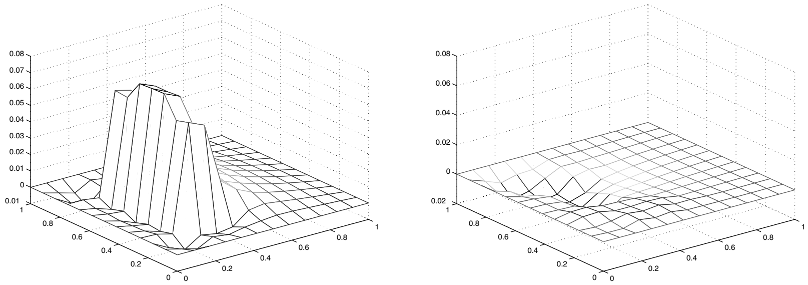


Fig. 10. Standard Galerkin error (left) and the error using adaptivity (right)

proposed scheme may be combined with a standard adaptive algorithm on the coarse scales. The method is thus very general in nature and may be applied to any problem where adaptivity is needed.

In this paper we have focused on two scales in two spatial dimensions. A natural extension would be to solve three dimensional problems with multiple scales. It is also natural to extend this theory to other equations modeling for instance flow and materials. We also intend to study non-linear and time dependent equations using this approach.

References

1. T. Arbogast and S. L. Bryant, A two-scale numerical subgrid technique for waterflood simulations, SPE J., Dec. 2002, pp 446-457.

2. B. Aksoylu, S. Bond, and M. Holst *An odyssey into local refinement and multilevel preconditioning III: Implementation of numerical experiments,* SIAM J. Sci. Comput., 25(2003), 478-498.
3. B. Aksoylu and M. Holst *An odyssey into local refinement and multilevel preconditioning I: Optimality of the BPX preconditioner,* SIAM J. Numer. Anal. (2002) in review
4. B. Aksoylu and M. Holst *An odyssey into local refinement and multilevel preconditioning II: stabilizing hierarchical basis methods,* SIAM J. Numer. Anal. in review
5. S. C. Brenner and L. R. Scott, *The mathematical theory of finite element methods,* Springer Verlag, 1994.
6. T. Y. Hou and X.-H. Wu, *A multiscale finite element method for elliptic problems in composite materials and porous media,* J. Comput. Phys. 134 (1997) 169-189.
7. T. J. R. Hughes, *Multiscale phenomena: Green's functions, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods,* Comput. Methods Appl. Mech. Engrg. 127 (1995) 387-401.
8. T. J. R. Hughes, L Mazzei, A. A. Oberai, and M. G. Larson, *The continuous Galerkin method is locally conservative,* J. Comput. Phys., Vol. 163(2) (2000) 467-488.
9. T. J. R. Hughes, G. R. Feijóo, L. Mazzei and Jean-Baptiste Quincy, *The variational multiscale method - a paradigm for computational mechanics,* Comput. Methods Appl. Mech. Engrg. 166 (1998) 3-24.
10. M. G. Larson and A. Målqvist, *Adaptive Variational Multiscale Methods Based on A Posteriori Error Estimates: Energy Norm Estimates for Elliptic Problems,* preprint
11. A. A. Oberai and P. M. Pinsky, *A multiscale finite element method for the Helmholtz equation,* Comput. Methods Appl. Mech. Engrg. 154 (1998) 281-297.

Multipole Solution of Electromagnetic Scattering Problems with Many, Parameter Dependent Incident Waves

Martin Nilsson and Per Lötstedt

Division of Scientific Computing, Department of Information Technology, Uppsala University, SE-75105 Uppsala, Sweden,
martin.nilsson@it.uu.se, per.lotstedt@it.uu.se

Summary. The electromagnetic scattering problem is solved with incoming plane waves from many directions with different frequencies. The solution of the Maxwell equations in integral form in the frequency domain is computed by a Galerkin discretization and multipole expansion. After discretization, systems of linear equations with many right hand sides and variable system matrix have to be solved. A minimal residual interpolation method reduces the computational work with at least an order of magnitude compared to a straightforward method. A numerical example with about 65000 unknowns and 15300 right hand sides illustrates the method.

Key words: multipole method, Maxwell's equations, scattering problem, right hand sides, interpolation

1 Introduction

We are interested in computing the time-harmonic electromagnetic scattering from a perfect electric conductor (PEC). An example of such a problem is the calculation of the radar cross section (RCS) of an airplane. The preferred method for these applications in the frequency domain is the method of moments where an integral equation is solved for the currents on the surface of the object. After discretization of the integral equation, a system of linear equations with a dense system matrix has to be solved. The solution of this system is often computed by a Krylov method. A Krylov method requires at least one matrix–vector multiplication in each iterative step [5]. A very efficient method in terms of computational complexity for this multiplication is the fast multipole method (FMM) [3, 4]. Suppose that the system has N unknowns. The work for one matrix–vector multiplication with the multilevel multipole algorithm is of $\mathcal{O}(N \log N)$ compared to $\mathcal{O}(N^2)$ for the standard method. This difference leads to significant savings in computational work for large N and also in memory requirements, since there is no need to store the whole matrix.

In RCS computations, a planar incident wave from a given direction with a given frequency is reflected by the object and the scattered signal is measured far away from the object. The waves usually have many different distinct frequencies f_i and come from many different angles ϕ_j. The angles generate many right hand sides in the system of linear equations and the system matrix depends on the frequency. Let L be the number of frequencies f_i and M be the number of angles ϕ_j. With Gaussian elimination, the computational work is of $\mathcal{O}(LN^3)$, since the matrix has to be factorized for each frequency, and the work grows like $\mathcal{O}(LMN^2)$ to back-substitute for each angle and frequency. Iterative solution with K iterations in a Krylov method for each solution and FMM for matrix–vector multiplication requires $\mathcal{O}(KLMN \log N)$ operations in a straightforward application to solve for the LM different cases. For small N, Gaussian elimination with LU-factorization may be the best alternative, especially if M is large, since the constant in front of the scaling of the work for FMM is quite large, but for large N the LU-factorization is impossible because of the cubic growth in the number of operations and the quadratic growth in storage. We will show how the number LM of iterative solutions of systems of equations using FMM can be reduced substantially by utilizing the smooth dependence of the system matrix on f and the right hand sides on f and ϕ.

The Combined Field Integral Equation (CFIE) in variational form to be solved for the surface current $\mathbf{J}$ on the surface Γ of the scatterer is

$$\begin{aligned}
&\alpha \int_\Gamma \int_\Gamma G(\mathbf{x},\mathbf{x}') \left(\mathbf{J}\cdot\mathbf{J}' - \frac{1}{\kappa^2} \boldsymbol{\nabla}_\Gamma \cdot \mathbf{J} \boldsymbol{\nabla}_\Gamma \cdot \mathbf{J}' \right) \mathrm{d}\Gamma \mathrm{d}\Gamma \\
&+ (1-\alpha) \frac{\mathrm{i}}{\kappa} \int_\Gamma \hat{\mathbf{n}} \times \int_\Gamma \boldsymbol{\nabla}_{\mathbf{x}'} G(\mathbf{x},\mathbf{x}') \times \mathbf{J} \cdot \mathbf{J}' \mathrm{d}\Gamma \mathrm{d}\Gamma \\
&= -\alpha \frac{1}{\mathrm{i}\kappa Z} \int_\Gamma \mathbf{E}_a \cdot \mathbf{J}' \mathrm{d}\Gamma + (1-\alpha) \frac{\mathrm{i}}{\kappa} \int_\Gamma \hat{\mathbf{n}} \times \mathbf{H}_a \cdot \mathbf{J}' \mathrm{d}\Gamma.
\end{aligned} \tag{1}$$

In (1), $\mathbf{J}'$ is the test current, $\kappa = 2\pi f/c$ is the wavenumber, c is the speed of light, Z is the impedance in free space, $\hat{\mathbf{n}}$ is the unit normal pointing outward from Γ, and $\mathrm{i} = \sqrt{-1}$. The function $G(\mathbf{x},\mathbf{x}')$ is the free-space Green's function for Helmholtz' equation. The parameter α can vary between 0, when we have the Magnetic Field Integral Equation (MFIE), and 1, when we have the Electric Field Integral Equation (EFIE). The right hand side depends on κ (and f) and the applied electric and magnetic fields, $\mathbf{E}_a(\phi)$ and $\mathbf{H}_a(\phi)$, and the left hand side depends on κ.

The equations are discretized with the Galerkin method and the rooftop or Rao–Wilton–Glisson basis functions [3]. The resulting system of equations has a dense, complex system matrix and the solution is the coefficients of the basis functions for the current. The monostatic RCS σ is defined by the quotient between the scattered far field $\mathbf{E}_s(\phi)$ in the same direction ϕ as the direction of propagation of the incident field $\mathbf{E}_a(\phi)$

$$\sigma(\phi) = \lim_{r\to\infty} 4\pi r^2 \frac{|\mathbf{E}_s(\phi)|^2}{|\mathbf{E}_a(\phi)|^2}, \tag{2}$$

where r is the distance to the object.

An efficient parallel implementation of FMM is found in [7]. The interpolation between different levels of the multilevel algorithm is improved, memory is saved when the translation operator is evaluated, and the size of the boxes in FMM is chosen to reduce the memory requirements. The parallelization with OpenMP for a shared memory computer is performed over the boxes or over the quadrature points depending on the level in the multilevel FMM.

The method to reduce the work for many f_i and ϕ_j is described in the next section. The algorithm is not restricted to electromagnetic problems in the frequency domain but can be applied successfully to all systems of linear equations with a smooth dependence on parameters. The method is applied in the last section to the computation of the RCS in the wing plane of an aircraft model for $L = 17$ at about $f = 6$ GHz and for all angles with $M = 900$. Another example with $L = 1$ and $f = 1.5$ GHz is found in [6].

2 Minimal Residual Interpolation (MRI)

The systems of linear equations to be solved are

$$\mathbf{A}_i \tilde{\mathbf{x}}_{ij} = \mathbf{b}_{ij},\ i = 1 \ldots L,\ j = 1 \ldots M,\ \mathbf{A}_i \in \mathbb{C}^{N\times N},\ \tilde{\mathbf{x}}_{ij}, \mathbf{b}_{ij} \in \mathbb{C}^N. \quad (3)$$

The system matrix depends on a parameter f so that $\mathbf{A}_i = \mathbf{A}(f_i)$ and the right hand side depends on f and ϕ so that $\mathbf{b}_{ij} = \mathbf{b}(f_i, \phi_j)$. The residual for the approximate solution $\mathbf{x}_{ij}$ is

$$\mathbf{r}_{ij} = \mathbf{b}_{ij} - \mathbf{A}_i \mathbf{x}_{ij}.$$

The equations are solved with an iterative method such that $\mathbf{r}_{ij}$ satisfies a convergence criterion in the Euclidean vector norm $\|\mathbf{r}_{ij}\| \leq \varepsilon$ for some given tolerance ε.

2.1 Solution for a Fixed *f*

Assume that the solutions to m right hand sides are known for fixed f_i and $\mathbf{A}_i$ to some precision given by the residual and that the solutions are linearly independent. The initial guess $\mathbf{x}^{(0)}_{i,m+1}$ for an iterative method applied to the solution of

$$\mathbf{A}_i \tilde{\mathbf{x}}_{i,m+1} = \mathbf{b}_{i,m+1} \quad (4)$$

is generated by a linear combination of old solutions

$$\mathbf{x}^{(0)}_{i,m+1} = \sum_{j=1}^{m} y_j \mathbf{x}_{ij},$$

where y_j is chosen so that

$$\mathbf{b}_{i,m+1} \approx \sum_{j=1}^{m} y_j \mathbf{b}_{ij}, \quad (5)$$

implying that $\mathbf{A}_i\mathbf{x}^{(0)}_{i,m+1} \approx \sum_{j=1}^{m} y_j\mathbf{b}_{ij}$ if we assume that $\|\mathbf{r}_{ij}\| \ll \|\mathbf{b}_{ij}\|$. The coefficients y_j are taken to be the linear least squares approximation of (5).

Introduce the following definitions:

$$\begin{aligned} &\mathbf{s}_j \equiv \mathbf{A}_i\mathbf{x}_{ij} = \mathbf{b}_{ij} - \mathbf{r}_{ij},\ j = 1\dots m, \\ &\mathbf{X}_{im} = \left[\mathbf{x}_{i1}\,\mathbf{x}_{i2}\,\dots\,\mathbf{x}_{im}\right],\ \mathbf{S}_m = \left[\mathbf{s}_1\,\mathbf{s}_2\,\dots\,\mathbf{s}_m\right]. \end{aligned} \tag{6}$$

The $\mathbf{QR}$-decomposition [1] of $\mathbf{S}_m$ is

$$\mathbf{A}_i\mathbf{X}_{im} = \mathbf{S}_m = \mathbf{Q}_m\mathbf{R}_m,\ \mathbf{Q}_m \in \mathbb{C}^{N\times m},\ \mathbf{R}_m \in \mathbb{C}^{m\times m}. \tag{7}$$

The linear least squares solution of

$$\begin{aligned} \|\mathbf{r}^{(0)}_{i,m+1}\| &= \|\mathbf{b}_{i,m+1} - \mathbf{A}_i\mathbf{x}^{(0)}_{i,m+1}\| = \|\mathbf{b}_{i,m+1} - \mathbf{A}_i\mathbf{X}_{im}\mathbf{y}_m\| \\ &= \|\mathbf{b}_{i,m+1} - \mathbf{S}_m\mathbf{y}_m\| = \|\mathbf{b}_{i,m+1} - \mathbf{Q}_m\mathbf{R}_m\mathbf{y}_m\| \end{aligned} \tag{8}$$

yields $\mathbf{y}_m = \mathbf{R}_m^{-1}\mathbf{Q}_m^H\mathbf{b}_{i,m+1}$ and the initial guess is

$$\mathbf{x}^{(0)}_{i,m+1} = \mathbf{X}_{im}\mathbf{R}_m^{-1}\mathbf{Q}_m^H\mathbf{b}_{i,m+1}. \tag{9}$$

If $\|\mathbf{r}^{(0)}_{i,m+1}\| \le \varepsilon$ in (8), then a satisfactory solution $\mathbf{x}^{(0)}_{i,m+1}$ is obtained without any iterations. This is the usual case if the right hand side has a slow variation with ϕ and the difference $\Delta\phi = \phi_{j+1} - \phi_j$ is small.

The residual for the initial guess $\mathbf{x}^{(0)}_{i,m+1}$ in (9) is

$$\begin{aligned} \mathbf{r}^{(0)}_{i,m+1} &= \mathbf{b}_{i,m+1} - \mathbf{A}_i\mathbf{X}_{im}\mathbf{y}_m = \mathbf{b}_{i,m+1} - \mathbf{S}_m\mathbf{R}_m^{-1}\mathbf{Q}_m^H\mathbf{b}_{i,m+1} \\ &= (\mathbf{I} - \mathbf{Q}_m\mathbf{Q}_m^H)\mathbf{b}_{i,m+1}. \end{aligned} \tag{10}$$

This is an expression for $\mathbf{r}^{(0)}_{i,m+1}$ which is cheap to evaluate since $m \ll N$ and $\mathbf{Q}_m^H\mathbf{b}_{i,m+1}$ is already computed in (9). The residual is small if $\mathbf{b}_{i,m+1}$ is almost spanned by the previous $\mathbf{s}_j$.

If $\|\mathbf{r}^{(0)}_{i,m+1}\| > \varepsilon$ then $\mathbf{x}_{i,m+1}$ has to be improved by the iterative method. The method in our numerical experiments in Sect. 3 is the GMRES algorithm [5]. Let the k:th iteration of $\mathbf{x}_{i,m+1}$ be $\mathbf{x}^{(k)}_{i,m+1}$ with its residual $\mathbf{r}^{(k)}_{i,m+1}$. Then

$$\mathbf{s}^{(k)}_{m+1} = \mathbf{b}_{i,m+1} - \mathbf{r}^{(k)}_{i,m+1} = \mathbf{S}_m\mathbf{y}_m + \mathbf{r}^{(0)}_{i,m+1} - \mathbf{r}^{(k)}_{i,m+1}.$$

If $\|\mathbf{r}^{(k)}_{i,m+1}\| \le \varepsilon_I$ for an $\varepsilon_I \le \varepsilon$ then the iterations are interrupted and $\mathbf{s}^{(k)}_{m+1}$ is included in the basis $\mathbf{S}_m$ if

$$\|(\mathbf{I} - \mathbf{Q}_m\mathbf{Q}_m^H)\mathbf{s}^{(k)}_{m+1}\| = \|(\mathbf{I} - \mathbf{Q}_m\mathbf{Q}_m^H)(\mathbf{r}^{(0)}_{i,m+1} - \mathbf{r}^{(k)}_{i,m+1})\| > \varepsilon_s, \tag{11}$$

where $\varepsilon_s > \varepsilon + \varepsilon_I$. The tolerances ε_I and ε are chosen equal in the next section. Otherwise, $\mathbf{s}^{(k)}_{m+1}$ is almost linearly dependent of the columns of $\mathbf{S}_m$ and $\mathbf{R}_{m+1}$ would be ill-conditioned. This is particularly the case when $\mathbf{x}_{i,m+1} = \mathbf{x}^{(0)}_{i,m+1}$ and no iterations are necessary.

Once the solution is found and (11) is satisfied we can construct $\mathbf{X}_{i,m+1} = [\mathbf{X}_{im}\,\mathbf{x}_{i,m+1}]$ and $\mathbf{S}_{m+1} = [\mathbf{S}_m\,\mathbf{s}_{m+1}]$. In our case, one of the columns in $\mathbf{S}_m$ and $\mathbf{R}_m$ is dropped when a new one is introduced. The strategy for choosing which column to drop can be first in, first out. An alternative is to pick the column corresponding to the smallest diagonal element in $\mathbf{R}_m$.

The $\mathbf{QR}$-decomposition of $\mathbf{S}_m$ is updated after solution of (4) when a column is appended to and deleted from $\mathbf{S}_m$, see [1].

2.2 Solution for a Fixed ϕ

Assume that ϕ is constant in (3) and that the system matrix $\mathbf{A}_i$ and $\mathbf{b}_{ij}$ depend smoothly on f. Then the equations to solve are

$$\mathbf{A}_i\mathbf{x}_{ij} = \mathbf{b}_{ij},\ i = 1\ldots L. \tag{12}$$

The solutions are known for l matrices and we wish to solve for the $(l+1)$:th matrix $\mathbf{A}_{l+1}$. The initial guess $\mathbf{x}^{(0)}_{l+1,j}$ for the $(l+1)$:th linear system can be interpolated in the same manner as in Sect. 2.1. Let

$$\mathbf{s}_i = \mathbf{A}_{l+1}\mathbf{x}_{ij} = \mathbf{b}_{ij} - \mathbf{r}_{ij} + (\mathbf{A}_{l+1} - \mathbf{A}_i)\,\mathbf{x}_{ij},\ i = 1\ldots l,$$

and define $\mathbf{S}_l$ as in (6) and $\mathbf{X}_{lj}$ by

$$\mathbf{X}_{lj} = [\mathbf{x}_{1j}\,\mathbf{x}_{2j}\ \ldots\ \mathbf{x}_{lj}]\,. \tag{13}$$

The linear combination of $\mathbf{x}_{ij}$ is chosen to minimize the initial residual

$$\mathbf{r}^{(0)}_{l+1,j} = \mathbf{b}_{l+1,j} - \mathbf{A}_{l+1}\mathbf{X}_{lj}\mathbf{y}_l = \mathbf{b}_{l+1,j} - \mathbf{S}_l\mathbf{y}_l = \mathbf{b}_{l+1,j} - \mathbf{Q}_l\mathbf{R}_l\mathbf{y}_l, \tag{14}$$

in the linear least squares sense as in (8). The calculation of $\mathbf{S}_l$ is more expensive here than in (6) since we have to multiply $\mathbf{X}_{lj}$ by the new matrix $\mathbf{A}_{l+1}$. Note that in certain cases $\mathbf{A}_{l+1} - \mathbf{A}_i$ is easily obtained, for instance if $\mathbf{A}_i = \mathbf{A} + f_i\mathbf{I}$.

2.3 Multilevel Interpolation

The right hand sides and ϕ_j in (3) are partitioned into different levels with $2^\ell + 1$ angles at level ℓ as in Fig. 1. Then the distance between the angles at two levels is $\Delta\phi_\ell = \Delta\phi_{\ell-1}/2$. The solution is computed first at level 1 for the 3 angles there. Then from level $\ell - 1$ to level ℓ the $2^{\ell-1}$ new intermediate values are interpolated using adjacent known data. If the residual $\|\mathbf{r}^{(0)}_{ij}\|$ there is sufficiently small, then no GMRES iterations are necessary at level l. Otherwise, (3) is solved with GMRES. It is possible to show [6] that the number of angles for which GMRES iteration is necessary is bounded independent of the number of right hand sides M under general conditions. The interpolation in the frequency in Sect. 2.2 is carried out in the same fashion.

The following theorem is a combination of two theorems proved in [6] and [7]. The solutions for many parameters f_i and ϕ_j are computed as above with interpolation in the ϕ-direction with f constant or in the f-direction with ϕ constant.

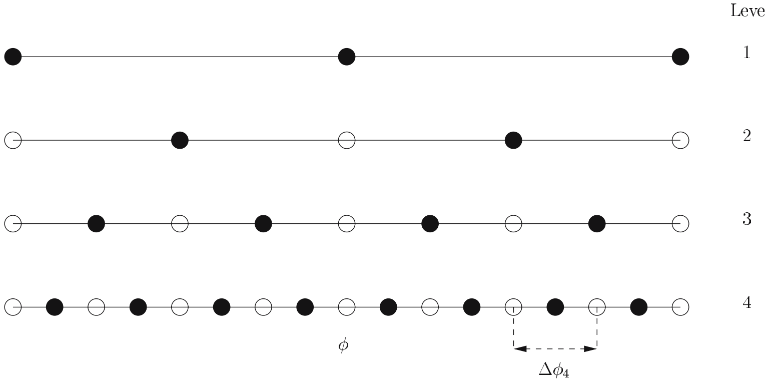


Fig. 1. The angle distribution at the different levels ℓ. The solutions at angles • are either obtained by interpolation or interpolation and iteration. The solutions at angles ∘ are known from a lower level

Theorem 1. *Assume that the components in the matrices* $\mathbf{A}_i = \mathbf{A}(f_i)$ *and the right hand side vectors* $\mathbf{b}_{ij} = \mathbf{b}(f_i, \phi_j)$ *have* p *continuous derivatives in* $f \in [f_{\min}, f_{\max}]$ *and* $\phi \in [\phi_{\min}, \phi_{\max}]$*, respectively. Furthermore, assume that the residuals satisfy* $\|\mathbf{r}_{ij}\| \le \varepsilon_I$ *and that* $\det(\mathbf{A}(f)) \ge C > 0$ *for all* $f \in [f_{\min}, f_{\max}]$.

An approximation to $\mathbf{b}_{i,m+1}$ *at* ϕ_{m+1} *at level* ℓ *is computed by the minimization (8)*

$$\min_{\mathbf{y}} \left\| \mathbf{b}_{i,m+1} - \sum_{i=1}^{p} \mathbf{s}_i y_i \right\| .$$

Then

$$\left\| \mathbf{b}_{i,m+1} - \sum_{i=1}^{p} \mathbf{s}_i y_i \right\| \le \min(\sqrt{N} b_{\max}^{(p)} \Delta\phi_{l-1}^{p} + \sqrt{p} \max_k \|\mathbf{l}_k\| \varepsilon_I, \|\mathbf{b}_{i,m+1}\|), \quad (15)$$

where $b_{\max}^{(p)} = \max_j \max_\phi |b_j^{(p)}(f_i, \phi)|$, $b_j^{(p)}$ *is the p:th derivative of the j:th component of* $\mathbf{b}_{i,m+1}$ *with respect to* ϕ*, and* $\mathbf{l}_k$ *consists of the coefficients of the Lagrange interpolation polynomial of* $(\mathbf{b}(f_i, \phi))_k$ *at the point* ϕ_{m+1}.

An approximation to $\mathbf{b}_{l+1,j}$ *at* f_{l+1} *at level* ℓ *is computed by minimization of (14)*

$$\min_{\mathbf{y}} \left\| \mathbf{b}_{l+1,j} - \sum_{i=1}^{p} \mathbf{s}_i y_i \right\| .$$

Then

$$\begin{aligned}\left\| \mathbf{b}_{l+1,j} - \sum_{i=1}^{p} \mathbf{s}_i y_i \right\| \le \min(&\sqrt{N} a_{\max}^{(p)} \Delta f_{l-1}^{p} \\ &+ \sqrt{p} \max_k \|\mathbf{l}_k\| \|\mathbf{A}_{l+1}\| \max_i \|\mathbf{A}_i^{-1}\| \varepsilon_I, \|\mathbf{b}_{l+1,j}\|),\end{aligned} \quad (16)$$

where

$$a^{(p)}_{\max} = \max_i \max_f |\sum_{k=1}^{N} (\mathbf{A}_{l+1})_{ik} \tilde{x}^{(p)}_k (f, \phi_j)| < \infty,$$
$$\tilde{\mathbf{x}}(f, \phi_j) = \mathbf{A}^{-1}(f)\mathbf{b}(f, \phi_j),\ \tilde{x}^{(p)}_k = \partial^p \tilde{\mathbf{x}}_k(f, \phi_j)/\partial f^p,$$

and $\mathbf{l}_k$ *consists of the coefficients of the Lagrange interpolation polynomial of* $(\mathbf{A}_{l+1}\tilde{\mathbf{x}}(f, \phi_j))_k$ *at the point* f_{l+1}.

Proof. The linear least squares fit in $\| \cdot \|$ has a smaller error than Lagrange interpolation. The error in componentwise Lagrange interpolation is bounded by the derivatives of the approximated function and $\Delta\phi^p$ and Δf^p as in [2]. The inverse of $\mathbf{A}_i$ is computed by Cramer's rule and with $\det(\mathbf{A}_i) > 0$ this inverse and its derivatives exist and $a^{(p)}_{\max}$ is bounded. Then the claims in the theorem follow in the same manner as in [6] and [7]. □

The performance of the method depends on the smoothness of $\mathbf{A}$ and the right hand sides and that $\mathbf{A}$ is not close to singular in the parameter interval. We find that if ε_I is negligible then the higher the level ℓ is, the smaller Δf or $\Delta\phi$ are and the smaller the minimal residual interpolation error is depending on the number of points p in the interpolation. The chances increase that no iteration is necessary the larger ℓ and p are. The bounds (15) and (16) are confirmed in numerical experiments in [6] and [7]. Sharper bounds depending on $(\kappa R \Delta\phi)^p$, where R is the radius of the smallest sphere surrounding the object, are derived in [6] for the case with variable ϕ.

3 Numerical Results

The monostatic radar cross section (2) for many incident waves from different angles ϕ_j and with different frequencies f_i is computed using the minimal residual interpolation from the previous section. The RCS is computed in the wing plane of an aircraft model *RUND*. The model is represented with triangles with 64959 edges. A coarser triangulation of the surface of *RUND* with about 4000 edges is found in Fig. 2. The incident angle ϕ is in the interval $\mathcal{I}_\phi = [0°, 360°]$ with $\phi = 90°$ at the nose of the model and the frequency f is in $\mathcal{I}_f = [5.9, 6.1]$ GHz. The RCS is measured in dB and computed at (f_i, ϕ_j) in Fig. 3 and the difference between the angles is $\Delta\phi = 0.4°$ and between the frequencies $\Delta f = 0.0125$. The dependence on f is relatively smooth while there are many troughs and peaks in the angular direction.

In Fig. 4, the relative residual $\|\mathbf{r}^{(0)}\|/\|\mathbf{b}\|$ of the initial guess after interpolation of $\mathbf{x}^{(0)}$ is depicted for all the different combinations of ϕ and f. About 150 solutions are first computed with GMRES-iterations at each of the frequencies $f = 5.9, 6.0, 6.1$. These are the points with peaks in Fig. 4. For the solutions at the intermediate angular and frequency values, sufficient accuracy is achieved by only interpolating between the previous solutions without invoking GMRES.

The expensive part in the GMRES-iterations is the matrix–vector multiplication by FMM. The number of matrix–vector multiplications to solve for all 15300 angle–frequency combinations is 13341 with full interpolation in both angle and frequency.

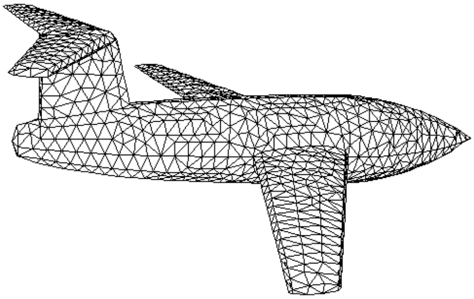


Fig. 2. The aircraft model *RUND* and its triangulation

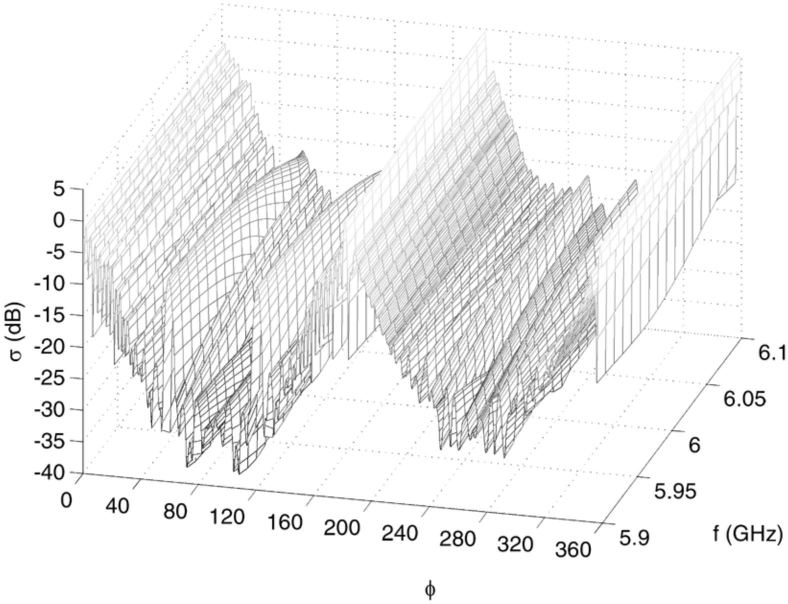


Fig. 3. The radar cross section depending on the incident angle ϕ and the frequency f

If interpolation is restricted to the angular direction as in Sect. 2.1 then 41310 matrix–vector multiplications are needed. Without MRI the estimated number of matrix–vector multiplications is 275000. The dominant part of the computational work is reduced by more than a factor 6 by introducing interpolation in the angle. By interpolating in both the angle and the frequency directions the work is reduced by more than a factor of 20.

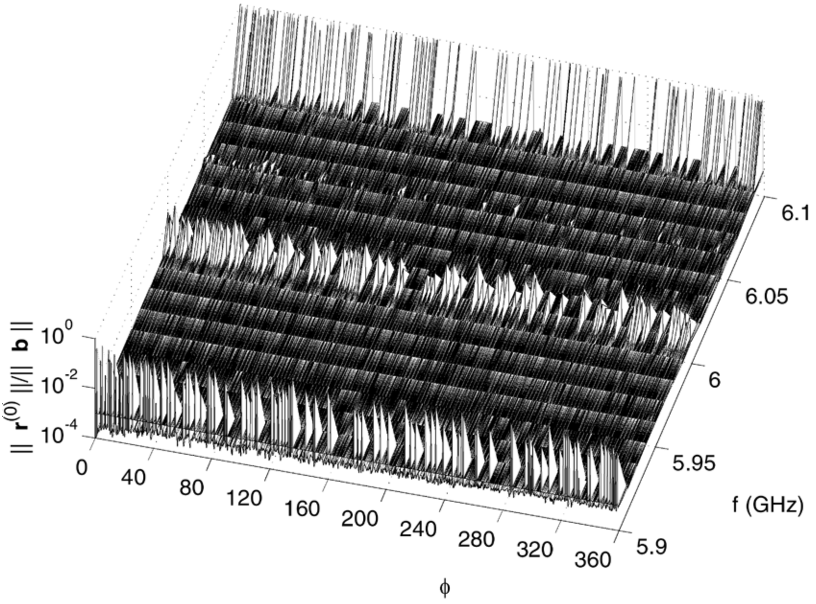


Fig. 4. The residuals of the initial guess after one interpolation in the angle–frequency plane

Acknowledgment

This work has been supported by the Parallel and Scientific Computing Institute (PSCI) and the Swedish National Aeronautical Research Program (NFFP). The industrial partner has been Aerotech Telub, Stockholm, Sweden.

References

1. Björck, Å: Numerical Methods for Least Squares Problems. SIAM, Society for Industrial and Applied Mathematics, Philadelphia, (1996)
2. Cheney, E. W.: Introduction to Approximation Theory. McGraw-Hill, New York, (1966)
3. Chew, W. C., Jin, J.-M., Michielssen, E., Song, J.: Fast and Efficient Algorithms in Computational Electromagnetics. Artech House, Inc., Norwood, (2001)
4. Coifman, R., Rokhlin, V., Wandzura, S.: The fast multipole method for the wave equation: A pedestrian prescription. IEEE Trans. Antennas Prop., **35**, 7–12, (1993)
5. Greenbaum, A.: Iterative Methods for Solving Linear Systems. SIAM, Society for Industrial and Applied Mathematics, Philadelphia, (1997)
6. Lötstedt, P., Nilsson, M.: A Minimal Residual Interpolation method for linear equations with multiple right hand sides. SIAM J. Sci. Comput., **25**, 2126–2144, (2004)
7. Nilsson, M.: Fast numerical techniques for electromagnetic problems in frequency domain. PhD Thesis, Department of Information Technology, Uppsala University, Uppsala, Sweden (2004)

Introduction to Normal Multiresolution Approximation

Olof Runborg

Department of Numerical Analysis and Computer Science, KTH, S–100 44 Stockholm, Sweden
olofr@nada.kth.se

Summary. A multiresolution analysis of a curve is normal if each wavelet detail vector with respect to a certain subdivision scheme lies in the local normal direction. In this paper we give an introduction to the analysis of normal approximations in [3]. We define the normal approximation in its basic form and show simplified proofs of the method's convergence, approximation quality and stability. We also explain how higher order approximations can be constructed using subdivision operators and give a brief summary of the corresponding results for these more general schemes.

Key words: subdivision, wavelet, normal mesh, normal multiresolution

1 Introduction

Finding representations of three-dimensional geometric data that allow for efficient computational processing have become an increasingly important problem, spurred by recent advances in shape acquisition technology such as laser scanners. It is now possible to sample real world three-dimensional objects with a very high level of detail; scanners can generate huge amounts of data typically in the form of triangular meshes with complex topology, [13]. The irregular format makes processing like compression, denoising, filtering and texturing, difficult. New ways of describing three-dimensional objects can lead to significantly improved compression algorithms and facilitate other processing. In addition, it is often desirable to support *progressive reconstruction*: a coarse version of the object is first quickly reconstructed and additional levels of detail are added as the reconstruction continues. This is useful for instance in streaming applications in networked environments.

Of particular interest for progressive reconstruction are multiresolution meshes, where the object is described through an hierarchy of increasingly detailed meshes. Each new mesh level is computed from the previous one by first *predicting* a new point, for instance by subdivision schemes like Butterfly or Loop [4, 14], and then *correcting* the predicted point by a *wavelet* (or detail) vector. Only the wavelet vectors need to be stored and because of the surface smoothness most wavelet vectors will be small, lending the representation well to compression.

The mathematical properties of wavelets are well understood in the so-called "functional setting", i.e., for the approximation of *functions* of one or more variables, which is the setting for image and sound processing. However, for the case of 1-D curves in the plane, or 2-D surfaces in 3-space, much less is known. Typically one takes a parameterization of the original curve or surface and ends up using wavelet analysis in each of the two or three components. This means the wavelet coefficients now become 2- or 3-vectors. It is important to choose an appropriate coordinate frame to describe these wavelet vectors. It is known that using an absolute coordinate frame for the wavelet or detail vectors leads to undesirable effects when editing curves; using a local coordinate frame defined by the normal works much better, as shown in [5, 6, 7, 15, 17].

In [8] the notion of normal approximation for curves or surfaces was introduced. A multiresolution approximation of a curve or surface is *normal* if all the wavelet vectors perfectly align with a locally defined normal direction which only depends on the coarser levels. Note that by the normal direction we mean a normal onto an approximation of the curve or surface. Given that this normal direction only depends on coarser levels, only a *single* scalar coefficient needs to be stored instead of the standard 2- or 3-vector. This is clearly extremely useful for compression applications, see [10, 11, 12]. In addition, [8] gives an algorithm to build normal mesh approximations of large complex scanned geometry.

Because they depend on the computation of a normal, these approximations lead to non-linear representations for which there is no general theory. However, a detailed study of the mathematical properties, such as convergence, regularity, and stability of normal multiresolution approximation for curves were made in [3]. In particular it was shown that these properties critically depend on the underlying subdivision scheme and that in general the convergence of the normal approximation of smooth curves equals the convergence of the subdivision scheme. See also [9] for the case of nonsmooth surfaces.

This paper is based on [3]. Here we give an introduction to the analysis of normal approximations in that article. In Sect. 2 we define the normal approximation in its basic form and show simplified proofs of the method's convergence, approximation quality and stability. In Sect. 3 we explain how higher order approximations can be constructed using subdivision operators and give a brief summary of the corresponding results for these more general schemes.

2 Basic Normal Multiresolution Analysis

2.1 Definitions and Notation

Fig. 1 illustrates the main idea from [8] in the case of a normal approximation based on midpoint subdivision. The original curve Γ is described by successively finer approximations, which are organized in different *multiresolution layers* indexed by j. We assume that Γ is a continuous, non intersecting curve in the plane, whose endpoints we shall take to be the zeroth level multiresolution points $v_{0,0}$ and $v_{0,1}$.

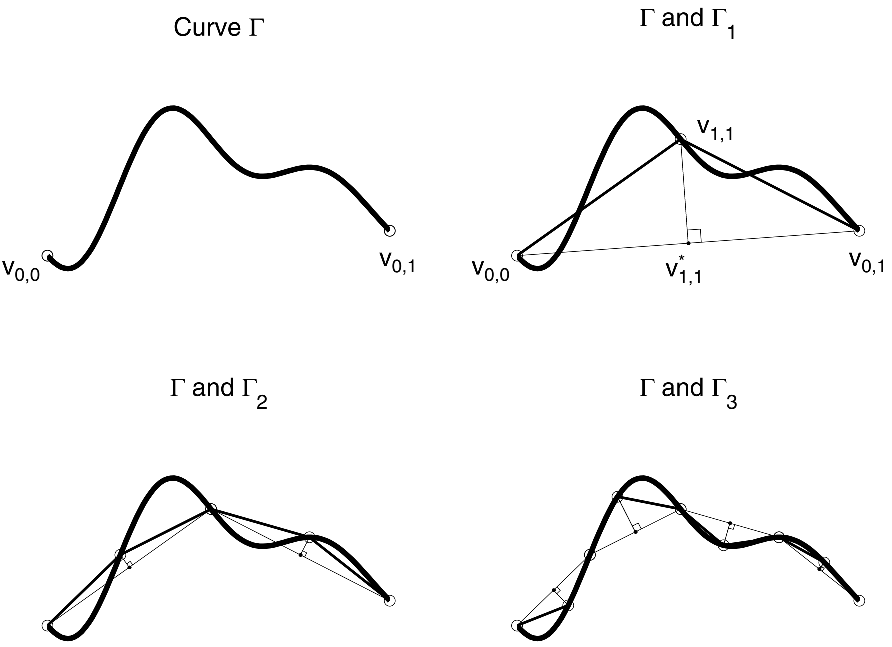


Fig. 1. Example of the normal mesh algorithm using the mean value of adjacent points as predictor

To construct the vertices at level $(j+1)$, we first set $v_{j+1,2k} = v_{j,k}$; this is what makes the construction interpolating. We also compute new points $v_{j+1,2k+1}$; each $v_{j+1,2k+1}$ lies in between the two old points $v_{j,k}$ and $v_{j,k+1}$. This is done by first computing a predicted or base point as the mean value of the old points, $v^*_{j+1,2k+1} = (v_{j,k} + v_{j,k+1})/2$. We next draw a line from $v^*_{j+1,2k+1}$ in the direction orthogonal to the line segment $(v_{j,k}, v_{j,k+1})$. This line is guaranteed to cross the curve segment between $v_{j,k}$ and $v_{j,k+1}$ at least once and we call one of those points $v_{j+1,2k+1}$. As this procedure continues, the polyline Γ_j, i.e. the piecewise linear curve connecting the points $v_{j,k}$ comes closer and closer to Γ. We can now think of this as a wavelet transformation similar to the notion of lifting [16]. Think of $v^*_{j+1,2k+1}$ as a prediction of the real point $v_{j+1,2k+1}$ computed based only on coarser information. Then the difference $v^*_{j+1,2k+1} - v_{j+1,2k+1}$ is a wavelet vector. Given that this vector points in a direction normal to a segment that again only depends on coarser data, we only need to store the length and one sign bit for this normal component to characterize it completely.

Before we go on to analyze this scheme, let us introduce some more notation. The norm $||\cdot||$ will always represent the euclidean norm in $\mathbb{R}^2$. We define the differences between breakpoints on Γ_j as $\Delta\Gamma_{j,k} = v_{j,k+1} - v_{j,k}$. We will always assume $\Gamma(s)$ is the arc length parameterization of Γ. This defines the corresponding breakpoints $s_{j,k}$ in parameter space: $\Gamma(s_{j,k}) = v_{j,k}$ for all j, k. The length along the curve between breakpoints is denoted $\Delta s_{j,k} = s_{j,k+1} - s_{j,k} > 0$. We also let $I_{j,k}$ be the interval $[s_{j,k}, s_{j,k+1}]$. Finally, we introduce the arc length ratio at which the scheme cuts the curve at each interval, $\beta_{j,k} = \Delta s_{j+1,2k}/\Delta s_{j,k}$.

Throughout the analysis we assume that $\Gamma(s)$ is twice continuously differentiable with bounded curvature. It can then be Taylor expanded

$$\Gamma(t) - \Gamma(s) = (t-s)\Gamma'(s) + \frac{1}{2}(t-s)^2 R(t,s) \ , \tag{1}$$

where the restterm is uniformly bounded in its arguments,

$$\sup_{s,t} ||R(t,s)|| \leq c_\gamma \ . \tag{2}$$

Most of the results are true with some modification also for less regular curves and we make a comment on this at the end of the section. For that case we introduce the Hölder spaces C^r; when $r = p + \kappa$, $p \in \mathbb{Z}$ and $0 < \kappa < 1$ we define C^r as the set of functions $f \in C^p$ for which the p-th derivative is Hölder continuous with exponent κ, i.e. $|f^{(p)}(t) - f^{(p)}(s)| \times |t-s|^{-\kappa}$ is bounded for all t, s.

2.2 Convergence

The construction of the normal scheme immediately begs the following question: how good an approximation to Γ is the polyline Γ_j? In particular, does the distance between Γ and the j-th level polyline decay to zero as j tends to ∞? In order to measure the distance between Γ and Γ_j we define the following parameterization for Γ_j,

$$\Gamma_j(s) = \frac{(s - s_{j,k})}{\Delta s_{j,k}} v_{j,k+1} + \frac{(s_{j,k+1} - s)}{\Delta s_{j,k}} v_{j,k} \ , \qquad s \in I_{j,k} \ . \tag{3}$$

With this parameterization we can prove uniform convergence $\Gamma_j \rightarrow \Gamma$. Before doing this, however, we need a lemma showing that the distance along the curve between the breakpoints $v_{j,k}$ on Γ_j decays exponentially to zero with j and that, in the limit, points at the next level will be added precisely in between points on the previous level.

Lemma 1. *Let $\Gamma(s) \in C^2([0,1]; \mathbb{R}^2)$ and $||\Gamma'(s)|| = 1$. Also assume that $\Gamma(s)$ does not cross itself. Then there are constants c_1 and c_2, independent of j, such that*

$$\sup_{k} \Delta s_{j,k} \leq c_1 2^{-j} \ . \tag{4}$$

and

$$\left| \beta_{j,k} - \frac{1}{2} \right| \leq c_2 \Delta s_{j,k} \ . \tag{5}$$

Proof. The proof is made in four steps.

1. Estimate difference between $||\varGamma(s+\varDelta s)-\varGamma(s)||$ and $\varDelta s$.
   Let $0 \le s < s+\varDelta s \le 1$. Using (1, 2) we get

$$\begin{aligned}
&0 \le \varDelta s - ||\varGamma(s)-\varGamma(s+\varDelta s)|| = \varDelta s||\varGamma'(s)|| - ||\varGamma(s)-\varGamma(s+\varDelta s)|| \\
&\quad \le ||\varGamma(s+\varDelta s)-\varGamma(s)-\varDelta s\varGamma'(s)|| = \frac{1}{2}(\varDelta s)^2||R(s+\varDelta s,s)|| \le \frac{1}{2}c_\gamma(\varDelta s)^2 .
\end{aligned}$$

2. Show that $\varGamma^{-1}$ is Lipschitz.
   First, suppose $\varDelta s \le 1/c_\gamma$. Then by the previous result,

$$||\varGamma(s+\varDelta s)-\varGamma(s)|| \ge \varDelta s - \frac{1}{2}c_\gamma(\varDelta s)^2 \ge \frac{\varDelta s}{2} \ .$$

   Second, for $1 \ge \varDelta s \ge 1/c_\gamma$,

$$||\varGamma(s+\varDelta s)-\varGamma(s)|| \ge \inf_{\substack{r\ge 1/c_\gamma \\ 0\le s\le s+r\le 1}} ||\varGamma(s+r)-\varGamma(s)|| \equiv c \ge c\varDelta s \ .$$

   Since $\varGamma$ is continuous and does not cross itself, $c > 0$. In conclusion,

$$||\varGamma(s+\varDelta s)-\varGamma(s)|| \ge q\varDelta s \ , \tag{6}$$

   where $q = \min(c, 1/2) > 0$.
3. Estimate deviation of $\beta_{j,k}$ from $1/2$.
   Since by the scheme's definition $||\varDelta\varGamma_{j+1,2k}|| = ||\varDelta\varGamma_{j+1,2k+1}||$, we have

$$\begin{aligned}
\left|\beta_{j,k}-\frac{1}{2}\right| &= \left|\frac{\varDelta s_{j+1,2k}}{\varDelta s_{j+1,2k}+\varDelta s_{j+1,2k+1}}-\frac{1}{2}\right| \\
&= \frac{1}{2}\left|\frac{\varDelta s_{j+1,2k}-||\varDelta\varGamma_{j+1,2k}||-\varDelta s_{j+1,2k+1}+||\varDelta\varGamma_{j+1,2k+1}||}{\varDelta s_{j,k}}\right| \\
&\le \frac{1}{4}c_\gamma\frac{(\varDelta s_{j+1,2k})^2+(\varDelta s_{j+1,2k+1})^2}{\varDelta s_{j,k}} \le \frac{1}{2}c_\gamma\varDelta s_{j,k} \ .
\end{aligned}$$

   This shows (5). We also derive another estimate of this difference. Through (6) we get a lower bound on $\beta_{j,k}$,

$$\beta_{j,k} = \frac{\varDelta s_{j+1,2k}}{\varDelta s_{j,k}} \ge \frac{||\varDelta\varGamma_{j,k}||}{2\varDelta s_{j,k}} \ge \frac{q}{2} \ .$$

   In the same way we get $(1-\beta_{j,k}) \ge q/2$. Together, this yields

$$\left|\beta_{j,k}-\frac{1}{2}\right| \le \frac{1-q}{2} \ . \tag{7}$$

4. Show the convergence rate of $\varDelta s_{j,k}$.
   We start by using (7),

$$\begin{aligned}\sup_k \Delta s_{j+1,k} &= \sup_k \max(\beta_{j,k}\Delta s_{j,k}, (1-\beta_{j,k})\Delta s_{j,k}) \\ &\leq \left(\frac{1}{2} + \left|\beta_{j,k} - \frac{1}{2}\right|\right) \sup_k \Delta s_{j,k} \leq \left(1 - \frac{q}{2}\right) \sup_k \Delta s_{j,k} \ .\end{aligned}$$

This shows that $\sup_k \Delta s_{j,k} \leq \delta^j$ where $\delta = 1 - q/2$. Next, we use (5),

$$\begin{aligned}\sup_k \Delta s_{j+1,k} &= \sup_k \max(\beta_{j,k}\Delta s_{j,k}, (1-\beta_{j,k})\Delta s_{j,k}) \\ &\leq \sup_k \left(\frac{1}{2}\Delta s_{j,k} + \frac{1}{2}c_\gamma(\Delta s_{j,k})^2\right) \ .\end{aligned}$$

Together with the exponential convergence, we get

$$\begin{aligned}2^j \sup_k \Delta s_{j,k} &\leq 2^j \sup_k \Delta s_{j-1,k}\left(\frac{1}{2} + \frac{1}{2}c_\gamma\delta^{j-1}\right) \\ &\leq 2^j \Delta s_0 \prod_{i=0}^{j-1}\left(\frac{1}{2} + \frac{1}{2}c_\gamma\delta^i\right) = \prod_{i=0}^{j-1}\left(1 + c_\gamma\delta^i\right) \\ &\leq \prod_{i=0}^{j-1}\exp\left(c_\gamma\delta^i\right) = \exp\left(c_\gamma\sum_{i=0}^{j-1}\delta^i\right) \leq \exp\left(\frac{c_\gamma}{1-\delta}\right) \ .\end{aligned}$$

The estimate (4) follows. □

We can now easily prove convergence.

Theorem 1. *Under the assumptions of Lemma* 1 *the normal approximation converges,*

$$\lim_{j\to\infty} \sup_{0\leq s\leq 1} ||\Gamma_j(s) - \Gamma(s)|| = 0 \ . \tag{8}$$

Proof. Consider first the interval $I_{j,k}$. Taylor expand and use (1, 2),

$$\begin{aligned}&\sup_{s\in I_{j,k}} ||\Gamma(s) - \Gamma_j(s)|| \\ &\quad= \sup_{s\in I_{j,k}} \left\| \frac{s - s_{j,k}}{\Delta s_{j,k}}[\Gamma(s_{j,k+1}) - \Gamma(s)] + \frac{s_{j,k+1} - s}{\Delta s_{j,k}}[\Gamma(s_{j,k}) - \Gamma(s)] \right\| \\ &\quad= \sup_{s\in I_{j,k}} \left\| \frac{(s - s_{j,k})(s - s_{j,k+1})^2}{2\Delta s_{j,k}} R(s, s_{j,k+1}) \right. \\ &\qquad\qquad \left. + \frac{(s_{j,k+1} - s)(s - s_{j,k})^2}{2\Delta s_{j,k}} R(s, s_{j,k}) \right\| \\ &\quad\leq c_\gamma(\Delta s_{j,k})^2 \ .\end{aligned}$$

Consequently, by Lemma 1,

$$\sup_{0\leq s\leq 1} ||\Gamma(s) - \Gamma_j(s)|| \leq c_\gamma \sup_k (\Delta s_{j,k})^2 \leq c\, 2^{-2j} \ ,$$

and (8) follows. □

2.3 Decay of Wavelet Coefficients

One of the important features of a normal multiresolution is the decay of the offsets in each of the normal directions, defined as

$$w_{j,k} = ||v_{j+1,2k+1} - v^*_{j+1,2k+1}|| \ .$$

We will refer to these as wavelet coefficients. Fast decay of those coefficients together with stability means that efficient compression schemes can be devised. We show here that they decay as 2^{-2j}.

Theorem 2. *Under the assumptions of Lemma* 1 *then*

$$w_{j,k} \leq c_\gamma (\Delta s_{j,k})^2 \tag{9}$$

with c_γ as in (2). *Moreover, there is a constant c independent of j such that*

$$\sup_k w_{j,k} \leq c\, 2^{-2j} \ . \tag{10}$$

Proof. As above, consider the interval $I_{j,k}$ and note that $s_{j+1,2k+1} \in I_{j,k}$ is the point where the constructed normal pierces the curve, $\Gamma(s_{j+1,2k+1}) = v_{j+1,2k+1}$. Denote the offset vector by u,

$$u = \Gamma(s_{j+1,2k+1}) - \frac{1}{2}(\Gamma(s_{j,k}) + \Gamma(s_{j,k+1})) \ ,$$

There exists $\eta \in I_{j,k}$ such that $\Gamma'(\eta)$ is parallel to $\Delta\Gamma_{j,k}$. We Taylor expand around η and using (1, 2) we get

$$\begin{aligned} u = {} & \frac{1}{2}(2s_{j+1,2k+1} - s_{j,k} - s_{j,k+1})\Gamma'(\eta) + \frac{(s_{j+1,2k+1} - \eta)^2}{2} R(s_{j+1,2k+1}, \eta) \\ & - \frac{(s_{j,k} - \eta)^2}{4} R(s_{j,k}, \eta) - \frac{(s_{j,k+1} - \eta)^2}{4} R(s_{j,k+1}, \eta) \ . \end{aligned}$$

Let $\hat{n}$ be the vector normal to $\Gamma'(\eta)$ and to $\Delta\Gamma_{j,k}$. Then,

$$\begin{aligned} w_{j,k} = ||u|| = |\hat{n} \cdot u| & \leq \frac{2(s_{j+1,2k+1} - \eta)^2 + (s_{j,k} - \eta)^2 + (s_{j,k+1} - \eta)^2}{4} c_\gamma \\ & \leq c_\gamma (\Delta s_{j,k})^2 \ , \end{aligned}$$

which is (9). The decay rate for $\Delta s_{j,k}$ established in Lemma 1 finally gives (10). □

2.4 Normal Parameterization

Normal multiresolution induces a parameterization of the curve Γ, as exemplified in Fig. 2. Analytically, this parameterization is described as follows: We define, at every level j, $\mathsf{s}_j : [0,1] \mapsto \mathbb{R}$ to be the piecewise affine map with breakpoints at the $t_{j,k} = 2^{-j}k$, $k = 0, \ldots, 2^j$, and for which $\Gamma(\mathsf{s}_j(t_{j,k})) = v_{j,k}$, i.e. $\mathsf{s}_j(t_{j,k}) = s_{j,k}$,

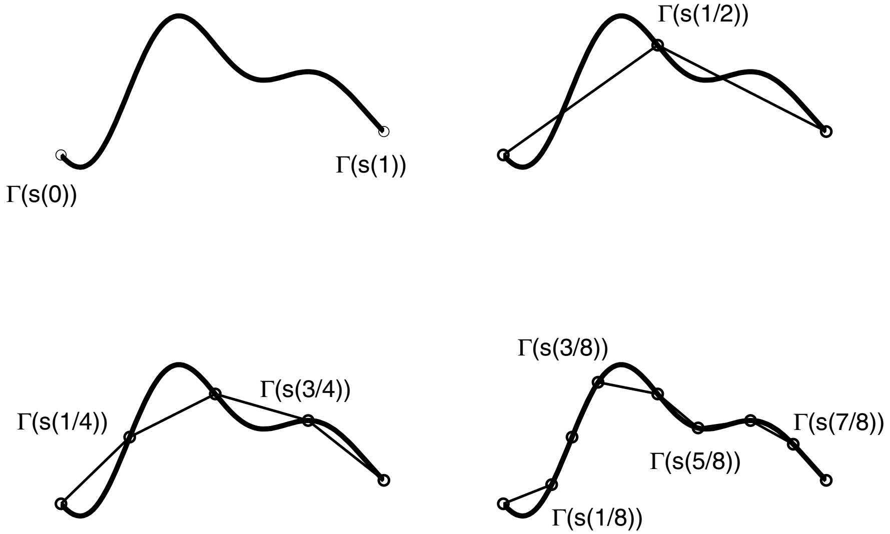


Fig. 2. Example of how the normal multiresolution induces a parameterization

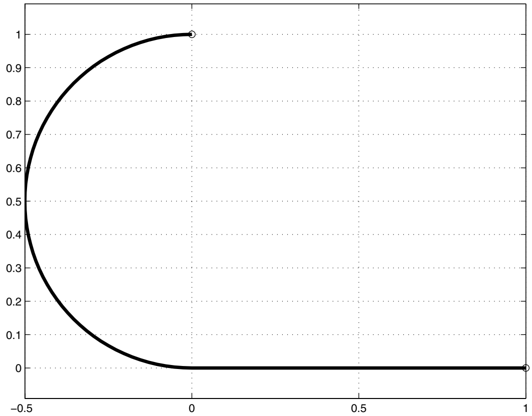


Fig. 3. Example of a curve with non smooth normal parameterization

see Fig. 2. If the node points on Γ_j approach each other when $j \to \infty$, i.e. if $\sup_k \Delta s_{j,k} \to 0$ then $\mathtt{s}_j(t)$ converges uniformly to a function $\mathtt{s}(t)$. The parameterization of the curve Γ induced by the normal multiresolution then maps $t \in [0,1]$ to $\Gamma(\mathtt{s}(t))$; we shall call this the *normal parameterization* of the curve Γ.

The regularity of the normal parameterization is related to the decay of the wavelet coefficients which directly determines the approximation quality. It can, however, also be significant independently, for instance when the normal scheme is used to generate meshes for computational purposes; the accuracy and stability of a numerical scheme is affected by the smoothness of the map from the computational to the physical domain. It should therefore be noted that the normal parameterization need *not* be smooth, even if Γ is. Consider the normal multiresolution as applied to the curve in Fig. 3, which consists of a $180°$ circle arc and a straight, tangent line segment with length equal to the diameter of the circle. At level zero, we have $v_{0,0} = (0,1) = \Gamma(\mathtt{s}(0))$ and $v_{0,1} = (1,0) = \Gamma(\mathtt{s}(1))$. Because of the special construction of Γ, the first inserted point $v_{1,1}$ coincides with the origin, $(0,0) = \Gamma(\mathtt{s}(1/2))$. After that, the normal multiresolution will induce a parameterization that corresponds to arc length for both the right and the left piece of the curve:

$$\Gamma(\mathtt{s}(t)) = \begin{cases} \frac{1}{2}(-\sin(2t\pi), 1+\cos(2t\pi)), & 0 \le t < \frac{1}{2} \ , \\ (2t-1, 0), & \frac{1}{2} \le t \le 1 \ . \end{cases}$$

However, the two pieces have different lengths, so the parameterization must have a discontinuity in its gradient, indeed

$$\left\| \frac{\mathrm{d}\Gamma(\mathtt{s}(t))}{\mathrm{d}t} \right\| = \begin{cases} \pi, & 0 \le t < \frac{1}{2} \ , \\ 2, & \frac{1}{2} < t \le 1 \ . \end{cases}$$

In this case the curve $\Gamma(s)$ is almost C^2 (its gradient $\Gamma'(s)$ is Lipschitz), yet its normal parameterization is only Lipschitz. This is because the regularity of the parameterization turns out to be limited not only by the smoothness of the curve, but also by the way the predicted point $v^*_{j,k}$ is computed as shown by the following argument. By definition,

$$\Big(\Gamma(\mathtt{s}(t+h)) - \Gamma(\mathtt{s}(t-h))\Big) \cdot \Big(\Gamma(\mathtt{s}(t+h)) - 2\Gamma(\mathtt{s}(t)) + \Gamma(\mathtt{s}(t-h))\Big) = 0 \ ,$$

at odd dyadic points ($t = (2k+1)2^{-j}$, $h = 2^{-j}$), where ' $\cdot$ ' stands for the $\mathbb{R}^2$ inner product, $(u,v) \cdot (u',v') = uu' + vv'$. Now *if* the parameterization were $C^{4+\varepsilon}$, with $\varepsilon > 0$, then we could Taylor expand $\Gamma(\mathtt{s}(t \pm h))$ around t and obtain

$$\frac{\mathrm{d}}{\mathrm{d}t} \left\| \frac{\mathrm{d}\Gamma(\mathtt{s}(t))}{\mathrm{d}t} \right\|^2 + \frac{h^2}{12} \left(\frac{\mathrm{d}^3}{\mathrm{d}t^3} \left\| \frac{\mathrm{d}\Gamma(\mathtt{s}(t))}{\mathrm{d}t} \right\|^2 - \frac{\mathrm{d}}{\mathrm{d}t} \left\| \frac{\mathrm{d}^2\Gamma(\mathtt{s}(t))}{\mathrm{d}t^2} \right\|^2 \right) = \mathrm{O}(h^4) \ ,$$

at odd dyadic points; if $\Gamma(\mathtt{s}(t)) \in C^{4+\varepsilon}$, then this equation extends to all t, h. By letting $h \to 0$ we see that we must have both $|\Gamma'|$ and $|\Gamma''|$ constant for this to hold, i.e. the curve Γ must be either a straight line or a circle segment, which is obviously not the case for general smooth curves Γ. In fact, typically the parameterization is not even piecewise smooth; the derivative is discontinuous at every dyadic point. We can, however, prove that it is always Lipschitz:

Theorem 3. *Under the assumptions of Lemma* 1 *there exists a Lipschitz continuous limit* $\mathtt{s}(t)$

$$\lim_{j\to\infty} \sup_{0 \le t \le 1} |\mathtt{s}_j(t) - \mathtt{s}(t)| = 0 \ . \tag{11}$$

Proof. Using Lemma 1 we get

$$|\mathtt{s}_{j+1}(t) - \mathtt{s}_j(t)| \leq \sup_k \left|\beta_{j,k} - \frac{1}{2}\right| \Delta s_{j,k} \leq c \sup_k (\Delta s_{j,k})^2 \leq c' 4^{-j} \ .$$

Hence, if $m < n$

$$|\mathtt{s}_n(t) - \mathtt{s}_m(t)| \leq \sum_{j=m}^{n-1} |\mathtt{s}_{j+1}(t) - \mathtt{s}_j(t)| \leq c' \sum_{j=m}^{n-1} 4^{-j} \leq \frac{c' 4^{-m+1}}{3} \ ,$$

which tends to zero with m. Therefore $\{\mathtt{s}_j(t)\}$ is Cauchy in sup norm and there exists a continuous limit $\mathtt{s}(t)$ satisfying (11). Finally, by Lemma 1,

$$|\mathtt{s}(t_1) - \mathtt{s}(t_2)| = \lim_{j\to\infty} |\mathtt{s}_j(t_1) - \mathtt{s}_j(t_2)| \leq \lim_{j\to\infty} \sup_k 2^j |\Delta s_{j,k}||t_1 - t_2| \leq c|t_1 - t_2| \ ,$$

which shows that $\mathtt{s}(t)$ is Lipschitz. □

2.5 Stability

Finally, we look at the stability of normal multiresolution. We assume that the initial points on the curve as well as the wavelet coefficients have some error or round-offs. The curve is then reconstructed as if they were exact. The result is a perturbed sequence of points $\tilde{v}_{j,k}$ and polylines $\tilde{\Gamma}_j$. We show here that this perturbation remains close to the exact curve Γ if the initial perturbations are small.

Let the errors in initial data be bounded as

$$\sup_k ||v_{0,k} - \tilde{v}_{0,k}|| = E_\gamma \ .$$

Since the wavelet coefficients rapidly decay with j we cannot expect stability unless the absolute error in the coefficients also decays. We therefore assume

$$\sup_k |w_{j,k} - \tilde{w}_{j,k}| = E_w 2^{-j\nu} \ , \qquad \nu > 0 \ .$$

We can then show

Theorem 4. *Under the assumptions of Lemma* 1 *and with the definition of the perturbations above there is a constant c depending on ν, but independent of j, E_γ and E_w such that*

$$\sup_k ||v_{j,k} - \tilde{v}_{j,k}|| \leq c(E_\gamma + E_w) \ . \tag{12}$$

Proof. This proof consists of two points.

1. Estimate of error in one reconstruction step.
   We start from a fixed level j and consider how the error

$$\varepsilon_j := \sup_k ||v_{j,k} - \tilde{v}_{j,k}||$$

is amplified in level $j+1$. Since the even points in level $j+1$ are the same as the points in level j the error in those points does not change. We therefore only need to investigate the odd points. For simplicity we consider fixed indices j, k and drop all indices in the notation, so that $v = v_{j,k}$, $s = s_{j,k}$, $\varDelta s = \varDelta s_{j,k}$, etc. We also call the new odd points $v^* = v_{j+1,2k+1}$ and $\tilde{v}^* = \tilde{v}_{j+1,2k+1}$. Finally, we let $\hat{n}$ and $\tilde{\hat{n}}$ be the normal and perturbed normal respectively at the indices j, k. The error in the new value v^* can then be estimated,

$$\begin{aligned} ||v^* - \tilde{v}^*|| &= \left\| \frac{\varGamma(s) + \varGamma(s+\varDelta s)}{2} + w\hat{n} - \frac{\tilde{\varGamma}(s) + \tilde{\varGamma}(s+\varDelta s)}{2} - \tilde{w}\tilde{\hat{n}} \right\| \\ &\leq \varepsilon_j + ||w(\tilde{\hat{n}} - \hat{n})|| + |w - \tilde{w}| \leq \varepsilon_j + E_w 2^{-j\nu} + w||\tilde{\hat{n}} - \hat{n}|| \ . \end{aligned}$$

Moreover, if $||\varDelta\varGamma - \varDelta\tilde{\varGamma}|| \geq ||\varDelta\varGamma||$,

$$||\tilde{\hat{n}} - \hat{n}|| \leq 2 \leq 2\frac{||\varDelta\varGamma - \varDelta\tilde{\varGamma}||}{||\varDelta\varGamma||} \leq \frac{4\varepsilon_j}{||\varDelta\varGamma||} \ .$$

On the other hand, if $||\varDelta\varGamma - \varDelta\tilde{\varGamma}|| < ||\varDelta\varGamma||$,

$$\begin{aligned} ||\tilde{\hat{n}} - \hat{n}|| &= \frac{\left\| \varDelta\varGamma||\varDelta\tilde{\varGamma}|| - ||\varDelta\varGamma||\varDelta\tilde{\varGamma} \right\|}{|\varDelta\tilde{\varGamma}|||\varDelta\varGamma||} \leq \frac{\left| ||\varDelta\tilde{\varGamma}|| - ||\varDelta\varGamma|| \right| + ||\varDelta\varGamma - \varDelta\tilde{\varGamma}||}{||\varDelta\varGamma||} \\ &\leq \frac{2||\varDelta\varGamma - \varDelta\tilde{\varGamma}||}{||\varDelta\varGamma||} \leq \frac{4\varepsilon_j}{||\varDelta\varGamma||} \end{aligned}$$

These estimates together with (6, 9) then gives

$$\begin{aligned} ||\tilde{v}^* - v^*|| &\leq \varepsilon_j + E_w 2^{-j\nu} + 4\frac{w\varepsilon_j}{||\varDelta\varGamma||} \leq \varepsilon_j + E_w 2^{-j\nu} + 4\varepsilon_j c_\gamma \frac{(\varDelta s)^2}{||\varDelta\varGamma||} \\ &\leq \varepsilon_j \left(1 + \frac{4c_\gamma}{q}\varDelta s \right) + E_w 2^{-j\nu} \ , \end{aligned}$$

which implies that

$$\varepsilon_{j+1} \leq \varepsilon_j \left(1 + \frac{4c_\gamma}{q} \sup_k \varDelta s_{j,k} \right) + E_w 2^{-j\nu} \leq \varepsilon_j (1 + c2^{-j}) + E_w 2^{-j\nu}. \quad (13)$$

2. Stability.
   Let us define the sequence $\{a_j\}_{j=1}^{\infty}$ as,

$$a_1 = E_\gamma(1+c) + E_w, \qquad a_{j+1} = a_j(1 + c2^{-j}) + E_w 2^{-j\nu} \ , \quad (14)$$

and set $b = \min(2, 2^\nu) > 1$. Clearly a_j is increasing. We then get

$$\begin{aligned}
a_{j+1} &\leq a_j \left(1 + b^{-j}\left(c + \frac{E_w}{a_1}\right)\right) \leq a_j(1 + b^{-j}(c+1)) \\
&\leq a_1 \prod_{i=1}^{j}(1 + b^{-i}(c+1)) \leq a_1 \prod_{i=0}^{j} \exp(b^{-i}(c+1)) \\
&= a_1 \exp\left((c+1)\sum_{i=0}^{j} b^{-i}\right) \leq a_1 \exp\left(\frac{b(c+1)}{b-1}\right) .
\end{aligned}$$

Moreover, since $\varepsilon_0 = E_\gamma$, induction on (13) and (14) shows that $\varepsilon_j \leq a_j$ for all j. Consequently,

$$\varepsilon_j \leq a_j \leq ca_1 \leq c'(E_\gamma + E_w) .$$

This shows (12). □

Remark 1. The result in Theorem 4 demonstrates that a compression scheme based on thresholding wavelet coefficients is stable. Suppose that we set

$$\tilde{w}_{j,k} = \begin{cases} w_{j,k}, & |w_{j,k}| \geq \varepsilon , \\ 0, & |w_{j,k}| < \varepsilon . \end{cases}$$

We then have $|w_{j,k} - \tilde{w}_{j,k}| \leq \varepsilon$ for all j, k, as well as $|w_{j,k} - \tilde{w}_{j,k}| \leq |w_{j,k}| \leq c2^{-2j}$. It follows that, for $0 \leq \kappa \leq 1$,

$$|w_{j,k} - \tilde{w}_{j,k}| \leq \varepsilon^{1-\kappa} c^\kappa 2^{-2j\kappa} .$$

If $\tilde{v}_{0,k} = v_{0,k}$ and $\kappa > 0$, then we obtain from Theorem 4 that

$$\sup_k ||\tilde{v}_{j,k} - v_{j,k}|| \leq c' \, c^\kappa \varepsilon^{1-\kappa} , \qquad \kappa > 0 ,$$

where c' depends on κ. To compare the perturbed polyline $\tilde{\Gamma}_j$ with Γ_j we need to consider their normal parameterizations,

$$\Gamma_j(t) = 2^j(t - t_{j,k})v_{j,k+1} + 2^j(t_{j,k+1} - t)v_{j,k} , \qquad t_{j,k} \leq t \leq t_{j,k+1} ,$$

and similarly for $\tilde{\Gamma}_j$. Then, clearly,

$$\sup_{0 \leq t \leq 1} ||\Gamma_j(t) - \tilde{\Gamma}_j(t)|| \leq c' \, c^\kappa \varepsilon^{1-\kappa} , \tag{15}$$

We note that after a finite number of refinement levels the perturbed wavelet coefficients will all be zero by the decay estimate (10). This implies that there is a well defined limit $\tilde{\Gamma}_j \to \tilde{\Gamma}$ as $j \to \infty$. It then follows from (15) that

$$\sup_{0 \leq t \leq 1} ||\Gamma(\mathtt{s}(t)) - \tilde{\Gamma}(t)|| \leq c' \, c^\kappa \varepsilon^{1-\kappa} .$$

A similiar, simpler, argument can be made when wavelet coefficients $\tilde{w}_{j,k}$ are set to zero beyond a certain level J. It leads to the estimate

$$\sup_{0 \leq t \leq 1} ||\Gamma(\mathtt{s}(t)) - \tilde{\Gamma}(t)|| \leq c2^{-2J} .$$

Remark 2. When Γ is not in C^2 the proofs of the theorems above become more complicated, but the main results are still true. As long as $\Gamma \in C^r$ with $1 < r < 2$ Lemma 1 and Theorem 1 to Theorem 4 all hold, although the decay rates in (5), (9) and (10) go down to $(\Delta s_{j,k})^{r-1}$, $(\Delta s_{j,k})^r$ and 2^{-jr} respectively. In fact, Theorem 1 and (4) hold also for Lipschitz continuous Γ and Theorem 1 even for $\Gamma \in C^r$ with $0 < r < 1$. In the latter case the decay in (4) is just algebraic however, $\sup_k \Delta s_{j,k} \leq c/(1 + j^{r/(1-r)})$. There is no improvement in decay rates when Γ is smoother than C^2. See [3] for these proofs.

3 Higher Order Generalizations

3.1 Definitions and Notation

We shall be interested in using more general methods, which will lead to higher quality approximation for smooth curves. As illustrated in Fig. 4, the same basic plan is followed: we still set $v_{j+1,2k} = v_{j,k}$. However, the predicted point $v^*_{j+1,2k+1}$ is now defined via a *subdivision* scheme S (see below), and $v_{j+1,2k+1}$ is an intersection point between $v_{j,k}$ and $v_{j,k+1}$ of Γ with the normal through $v^*_{j+1,2k+1}$ to the segment $(v_{j,k}, v_{j,k+1})$. We still define the wavelet coefficient as $w_{j,k} = ||v_{j+1,2k+1} - v^*_{j+1,2k+1}||$. The results now also depend on S. In short, convergence requires additional assumptions but regularity and wavelet decay can be much improved.

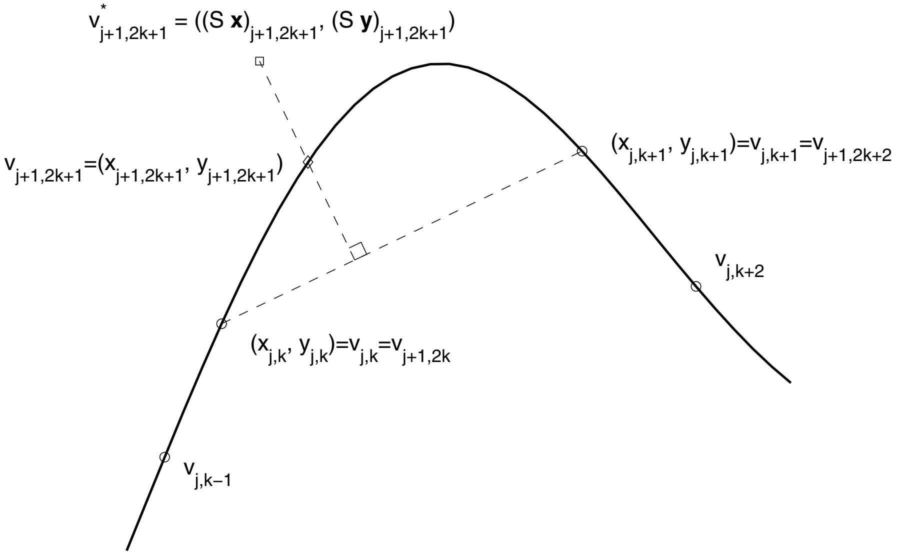

$v^*_{j+1,2k+1} = ((S\,\mathbf{x})_{j+1,2k+1}, (S\,\mathbf{y})_{j+1,2k+1})$
$v_{j+1,2k+1}=(x_{j+1,2k+1}, y_{j+1,2k+1})$
$(x_{j,k+1}, y_{j,k+1})=v_{j,k+1}=v_{j+1,2k+2}$
$v_{j,k+2}$
$(x_{j,k}, y_{j,k})=v_{j,k}=v_{j+1,2k}$
$v_{j,k-1}$

Fig. 4. Notation for the higher order normal scheme

To simplify notation we introduce infinite sequences. Sequences will be written in bold face, and elements of sequences in normal font, $\boldsymbol{x} := (x_k)$. We define the difference operator Δ as

$$(\Delta \boldsymbol{x})_k = x_{k+1} - x_k \ .$$

Often a sequence itself is indexed by the refinement level j; then we use the convention that $\boldsymbol{x}_j := (x_{j,k})$. We use the usual sup-norm for sequences, $|\boldsymbol{x}|_\infty = \sup_k |x_k|$.

A local, stationary subdivision scheme is characterized by a bounded linear operator S, defined by a sequence $\boldsymbol{a}$ as follows

$$(S\boldsymbol{x})_k = \sum_\ell a_{k-2\ell} x_\ell \ ,$$

where the number of non-zero elements in $\boldsymbol{a}$, and consequently also the sum, is finite. Given a starting sequence $\boldsymbol{x}_0$ we can apply S iteratively and define, for all $j \geq 0$,

$$\boldsymbol{x}_{j+1} = S\boldsymbol{x}_j \ . \tag{16}$$

The sequence $\boldsymbol{x}_0$ can be viewed as a coarse approximation of a function, on the integer grid; the sequences $\boldsymbol{x}_j$ then give successively finer approximations of the function on grids with spacing 2^{-j}. We are interested in the case when this process converges to a smooth limit function as j increases. A subdivision scheme is *interpolating* if $s_{2l} = \delta_{l,0}$, implying $x_{j+1,2k} = x_{j,k}$ for all j, k; in this case the $x_{j,k}$ can be interpreted as function values of the limit function $f(x)$ with $x_{j,k} = f(2^{-j}k)$.

Let $\boldsymbol{v}_j$ be a two-dimensional sequence, $\boldsymbol{v}_j = (\boldsymbol{x}_j, \boldsymbol{y}_j)$ and define $S\boldsymbol{v}_j = (S\boldsymbol{x}_j, S\boldsymbol{y}_j)$. The iteration corresponding to (16) for the normal scheme can then be written

$$\boldsymbol{v}_{j+1} = S\boldsymbol{v}_j + \boldsymbol{w}_j \cdot \hat{\boldsymbol{n}}_j \ ,$$

where $\hat{\boldsymbol{n}}_j$ is the two-dimensional sequence of normals and the last product is taken elementwise. Since the $\boldsymbol{w}_j$ sequences decay rapidly to zero we can therefore view the normal scheme as a non-linear perturbation of the underlying linear subdivision scheme S.

Examples of interpolating subdivision schemes are given by the so-called Lagrange interpolation subdivision schemes. For those the new odd-indexed points are defined as the values taken by a polynomial determined by several neighboring old points. We shall denote the 2ℓ-point scheme by $S_{2\ell}$. For instance, in the two-point scheme, $x_{j+1,2k+1}$ is given the value at $t = 1/2$ of the *linear* polynomial that takes the values $x_{j,k}$ at $t = 0$ and $x_{j,k+1}$ at $t = 1$; in other words,

$$(S_2\boldsymbol{x})_{2k+1} := x_{j+1,2k+1} = \frac{1}{2}(x_{j,k} + x_{j,k+1}) \ .$$

This is hence the scheme that was used in the basic normal approximation in Sect. 2. The construction strategy generalizes to higher order and in the four-point scheme, $x_{j+1,2k+1}$ is given the value at $t = 1/2$ of the *cubic* that takes the values $x_{j,k-1}$, $x_{j,k}$, $x_{j,k+1}$ and $x_{j,k+2}$ at $t = -1, 0, 1, 2$ respectively, leading to

$$(S_4\boldsymbol{x})_{2k+1} := x_{j+1,2k+1} = \frac{9}{16}(x_{j,k} + x_{j,k+1}) - \frac{1}{16}(x_{j,k-1} + x_{j,k+2}) \ .$$

In general, the 2ℓ-point scheme gives $x_{j+1,2k+1}$ the value at $t = 1/2$ of the $(2\ell - 1)$-degree polynomial that takes the values $x_{j,k+m}$ at $t = m$, where $m = -\ell + 1, \dots, \ell$. Since these are all interpolating schemes, we have of course $(S_2\boldsymbol{x}_j)_{2k} = (S_4\boldsymbol{x}_j)_{2k} = (S_{2\ell}\boldsymbol{x}_j)_{2k} = x_{j,k}$.

We assume that the curve Γ is at least $C^2([0,1],\mathbb{R}^2)$. This means that it can always be broken up into adjacent finite length pieces, possibly overlapping, that can be well parameterized by the x-coordinate (with, say, $|dy/dx| \leq 2$) or by the y-coordinate (with, say, $|dx/dy| \leq 2$). For technical reasons we shall also assume that it can be broken up in this way in a *finite* number of pieces. Then the theorems below follow directly from the theorems in [3], since in each piece we would have either

$$|\Delta x_{j,k}| \leq |\Delta s_{j,k}| \leq \sqrt{5}|\Delta x_{j,k}| \quad \text{or} \quad |\Delta y_{j,k}| \leq |\Delta s_{j,k}| \leq \sqrt{5}|\Delta y_{j,k}| \ .$$

3.2 Convergence

When a general subdivision scheme is used to determine the predicted point there is no longer any guarantee that the normal cuts the curve in between the previous two points. This makes the convergence analysis more complicated. To have a proper parameterization, we need to ensure that all $\boldsymbol{s}_j$ sequences are increasing, i.e., $\Delta\boldsymbol{s}_j > 0$, given that the initial sequence $\boldsymbol{s}_0$ is increasing. In general there are very few subdivision schemes that preserve increasing sequences. In our case, the $\boldsymbol{s}_j$ sequences are obtained by a nonlinear perturbation of subdivision so the situation is even more complex. Fortunately, there are conditions on both the subdivision scheme and the initial sequence that guarantee that the $\boldsymbol{s}_j$ will be increasing. The following theorem introduces a non-uniformity measure $\mathcal{N}$ of a sequence which is the maximal ratio of the length of two neighboring intervals; it states that if the non-uniformity of the initial sequence is bounded and the subdivision scheme preserves this bound, the sequences $\boldsymbol{s}_j$ generated by the normal scheme will be increasing and converge exponentially, provided the initial sequence also resolves the curve well enough.

Theorem 5. *Let S be an interpolating subdivision scheme. Let the non-uniformity $\mathcal{N}(\boldsymbol{x})$ be defined by*

$$\mathcal{N}(\boldsymbol{x}) := \sup_k \max\left(\frac{|(\Delta\boldsymbol{x})_k|}{|(\Delta\boldsymbol{x})_{k+1}|}, \frac{|(\Delta\boldsymbol{x})_{k+1}|}{|(\Delta\boldsymbol{x})_k|}\right) . \tag{17}$$

Suppose there is an R such that for every strictly increasing $\boldsymbol{x}$ with $\mathcal{N}(\boldsymbol{x}) \leq R$, $S\boldsymbol{x}$ is strictly increasing as well, and satisfies $\mathcal{N}(S\boldsymbol{x}) \leq \mathcal{N}(\boldsymbol{x})$. Suppose $\boldsymbol{s}_0$ is strictly increasing, $\mathcal{N}(\boldsymbol{s}_0) < R$ and that $|\Delta\boldsymbol{s}_0|_\infty$ is sufficiently small. If $\Gamma \in C^2([0,1];\mathbb{R}^2)$, then the normal approximation converges,

$$\lim_{j\to\infty} \sup_{0\leq s\leq 1} ||\Gamma_j(s) - \Gamma(s)|| = 0 \ ,$$

where $\varGamma_j$ is defined in (3). *Moreover, $\boldsymbol{s}_j$ is strictly increasing for all j, with $\mathcal{N}(\boldsymbol{s}_j) \leq R$ for all j, and the $\boldsymbol{s}_j$ converge exponentially, i.e., there is a $\delta < 1$ so that*

$$|\varDelta \boldsymbol{s}_j|_\infty \leq c\, \delta^j\, |\varDelta \boldsymbol{s}_0|_\infty , \qquad \forall j .$$

Examples of subdivision schemes that meet the requirements in the theorem are, for instance, the first Lagrange interpolation schemes introduced above.

Remark 3. As was seen in Sect. 2, when $S = S_2$ the smallness assumptions on $|\varDelta \boldsymbol{s}_0|_\infty$ and $\mathcal{N}(\boldsymbol{s}_0)$ are not necessary and exponential convergence follows when $\varGamma$ is merely Lipschitz continuous.

3.3 Wavelet Decay, Regularity and Stability

To characterize the approximation quality and stability of the higher order normal scheme we need to introduce two additional subdivision concepts: the *order* of the operator and the *derived* operators.

The *order* of a subdivision scheme S is the largest degree for which it leaves the corresponding space of monic polynomials invariant. More precisely, let $\boldsymbol{k}$ be the sequence for which the k-th entry is k itself and denote the order of S by $\mathcal{P}$. Then $\mathcal{P}$ is the largest integer such that for all p-degree monic polynomials P with $0 \leq p < \mathcal{P}$, a p-degree monic polynomial Q exists so that $SP(\boldsymbol{k}) = Q(\boldsymbol{k}/2)$, where the polynomials are applied elementwise to generate new sequences. If S is interpolating, then $SP(\boldsymbol{k}) = P(\boldsymbol{k}/2)$. For example, S_2 is of order two and $S_2\boldsymbol{k} = \boldsymbol{k}/2$. In general the 2ℓ-point Lagrange interpolation scheme is of order 2ℓ. We always assume that $\mathcal{P}$ is at least one so that for a constant sequence $\mathbf{1}$ we have $S\mathbf{1} = \mathbf{1}$.

The *derived* subdivision schemes are defined as

$$S^{[0]} = S, \qquad S^{[p]} = 2\varDelta S^{[p-1]}\varDelta^{-1}, \qquad p > 0 .$$

The significance of those schemes is that if the sequences $\{\boldsymbol{x}_j\}$ are generated by (16) then the divided differences of those sequences are generated by the corresponding derived schemes,

$$\boldsymbol{x}_{j+1}^{[p]} = S^{[p]}\boldsymbol{x}_j^{[p]},$$

where

$$\boldsymbol{x}_j^{[p]} = D_j^p \boldsymbol{x}_j , \qquad (D_j\boldsymbol{x})_k = 2^j(x_{k+1} - x_k) .$$

Note that $S^{[p]}$ is well-defined as long as $S^{[p-1]}$ has at least order one, and that the order of $S^{[p]}$ is one less than the order of $S^{[p-1]}$. Thus $S^{[p]}$ is defined for $p \leq \mathcal{P}$. The derived schemes can easily be written down explicitly for a given subdivision scheme. For example, the first derived scheme for the four-point scheme is

$$(S_4^{[1]}\boldsymbol{x})_{2k} = \frac{1}{8}(x_{k-1}+8x_k-x_{k+1}) , \qquad (S_4^{[1]}\boldsymbol{x})_{2k+1} = \frac{1}{8}(-x_{k-1}+8x_k+x_{k+1}) .$$

Note that derived schemes are typically not interpolating even if the base scheme is.

We are now ready to state a theorem that corresponds to Theorem 3, Theorem 2 and Theorem 4 in Sect. 2. We use the same definitions and notation. Furthermore, we introduce in (18) a growth rate μ that characterizes a derived operator $S^{[p]}$. The value of μ can in particular be chosen as $\log_2 |S^{[p]}|_\infty$. If the ℓ_∞-spectral radius σ_p of $S^{[p]}$ is strictly smaller, a more precise choice can be made: $\mu = \log_2 \sigma_p + \varepsilon$ for any $\varepsilon > 0$. (This follows from the well-known identity $\sigma_p = \lim_{j\to\infty} |S^{[p]^j}|_\infty^{1/j}$.)

The results in this higher order case depend on the smoothness of S, determined via $\mathcal{P}$, p and μ, and on the regularity r of the curve Γ. For sufficiently regular Γ and high order S, i.e. for r and $\mathcal{P}$ large enough, the regularity of $\mathrm{s}(t)$ is almost $C^{p-\mu}$ while the wavelet decay is one order higher, $2^{-j(p-\mu+1)}$. The best bound one can get from this theorem is thus obtained for that combination of p and μ where $p-\mu$ is maximal. This maximum need not be reached at $p = \mathcal{P}$.

Note that the exponential decay rate of $|\Delta \boldsymbol{s}_j|_\infty$ in (19) is, for example, established by Theorem 5.

Theorem 6. *Let S be the $\mathcal{P}$-th order interpolating subdivision scheme used in the normal scheme. Assume that there are positive real numbers C, μ and integer $p \le \mathcal{P}$ such that*

$$\left|S^{[p]^j}\right|_\infty \le C 2^{\mu j} , \qquad \forall j \ge 0 , \qquad \mu \le p-1 , \tag{18}$$

that there is a $\delta < 1$ so that

$$|\Delta \boldsymbol{s}_j|_\infty \le C\delta^j \tag{19}$$

and that $\Gamma \in C^r([0,1];\mathbb{R}^2)$ with $r \ge 2$ satisfies the hypothesis given in Sect. 3.1. Then

1. *Decay of wavelet coefficients.*
   For all $\varepsilon > 0$ there is a constant C_ε such that

$$|\boldsymbol{w}_j|_\infty \le C_\varepsilon 2^{-j(Q-\varepsilon)} , \qquad Q := \min(p-\mu+1, r, \mathcal{P}) .$$

2. *Regularity of normal parameterization.*
   There is a limit function $\mathrm{s}(t)$ such that

$$\lim_{j\to\infty} \sup_{0\le t\le 1} |\mathrm{s}_j(t) - \mathrm{s}(t)| = 0 .$$

   The limit $\mathrm{s}(t)$ belongs to $C^{Q'-\varepsilon}([0,1])$, for all $\varepsilon > 0$, and $Q' := \min(p-\mu, r)$.
3. *Stability.*
   If $p-\mu > 1$ when $p > 1$ there is a constant c depending on ν, but independent of j, E_γ and E_w such that

$$\sup_k ||v_{j,k} - \tilde{v}_{j,k}|| \le c(E_\gamma + E_w) .$$

Remark 4. The regularity of the normal parameterization is the same regularity that we get for the limit function of the pure predictor subdivision scheme S when we use the same method of proof. If we take the very special case $\Gamma(s) = (s, 0)$ for all s, then the normal multiresolution scheme gives $\boldsymbol{s}_{j+1} = S\boldsymbol{s}_j$. In this case $\boldsymbol{w}_j = 0$, and we no longer have a curve approximation problem. However, we can define $\mathtt{s}_j(t)$ as before, and the convergence of $\mathtt{s}_j(t)$ still holds, as a special case of the theorem. Theorem 6 can thus be viewed as an extension, without loss in the strength of the estimates, of standard convergence results for linear subdivision, see e.g. [1, 2].

We conclude with some examples where we apply Theorem 6 to the first Lagrangian interpolating subdivision predictors S_{2l}. A numerical illustration is presented in Fig. 5. In the two-point case we simply have

$$\left|S_2^{[1]}\boldsymbol{x}_j^{[1]}\right|_\infty = \left|\boldsymbol{x}_j^{[1]}\right|_\infty .$$

Hence we can take $\mu = 0$ and $p = 1$. Since $\mathcal{P} = 2$ we get $Q = 2$ and $Q' = 1$ if $\Gamma \in C^2$. Then $\mathtt{s} \in C^{1-\varepsilon}$ and $|\boldsymbol{w}_j| \leq c2^{-(2-\varepsilon)j}$, agreeing with the results in Sect. 2. For the four-point operator we start from the estimate

$$\left|S_4^{[3]}\boldsymbol{x}_j^{[3]}\right|_\infty \leq 2\left|\boldsymbol{x}_j^{[3]}\right|_\infty .$$

Here we thus take $\mu = 1$, $p = 3$ and with $\mathcal{P} = 4$ we get $Q = 3$, $Q' = 2$ when $\Gamma \in C^3$. Thus, $\mathtt{s}(t)$ is almost C^2 and $\boldsymbol{w}_j$ decay, almost, as 2^{-3j}. For S_6 and S_8 it is more difficult to get estimates of the optimal pair p, μ. Empirically the wavelet decay is consistent with what is obtained for pure subdivision, where the limit functions of the S_6-scheme belong to $C^{2.83}$ and S_8 generates $C^{3.55}$ functions; the corresponding wavelet decay in the normal scheme is 3.83 and 4.55 respectively.

References

1. A. S. Cavaretta, W. Dahmen, and C. A. Micchelli. Stationary subdivision. *Memoirs Amer. Math. Soc.*, 93(453), 1991.
2. I. Daubechies and J. C. Lagarias. Two-scale difference equations I. Existence and global regularity of solutions. *SIAM J. Math. Anal.*, 22(5):1388–1410, 1991.
3. I. Daubechies, O. Runborg, and W. Sweldens. Normal multiresolution approximation of curves. *Constr. Approx.*, 20:399–463, 2004.
4. N. Dyn, D. Levin, and J. Gregory. A butterfly subdivision scheme for surface interpolation with tension control. *ACM Trans. on Graphics*, 9(2):160–169, 1990.
5. A. Finkelstein and D. H. Salesin. Multiresolution curves. In *Computer Graphics (SIGGRAPH '94 Proceedings)*, pages 261–268, 1994.
6. D. R. Forsey and R. H. Bartels. Hierarchical B-spline refinement. In *Computer Graphics (SIGGRAPH '88 Proceedings)*, pages 205–212, August 1988.
7. S. J. Gortler and M. F. Cohen. Hierarchical and variational geometric modeling with wavelets. In *Proceedings Symposium on Interactive 3D Graphics*. Siggraph, May 1995.
8. I. Guskov, K. Vidimče, W. Sweldens, and P. Schröder. Normal meshes. In *Computer Graphics (SIGGRAPH '00 Proceedings)*, pages 259–268, 2000.
9. M. Jansen, H. Choi, S. Lavu, and R. Baraniuk. Multiscale Image Processing Using Normal Triangulated Meshes. In *IEEE International Conference on Image Processing*, volume 2, pages 229–232, Thessaloniki, Greece, October 2001.

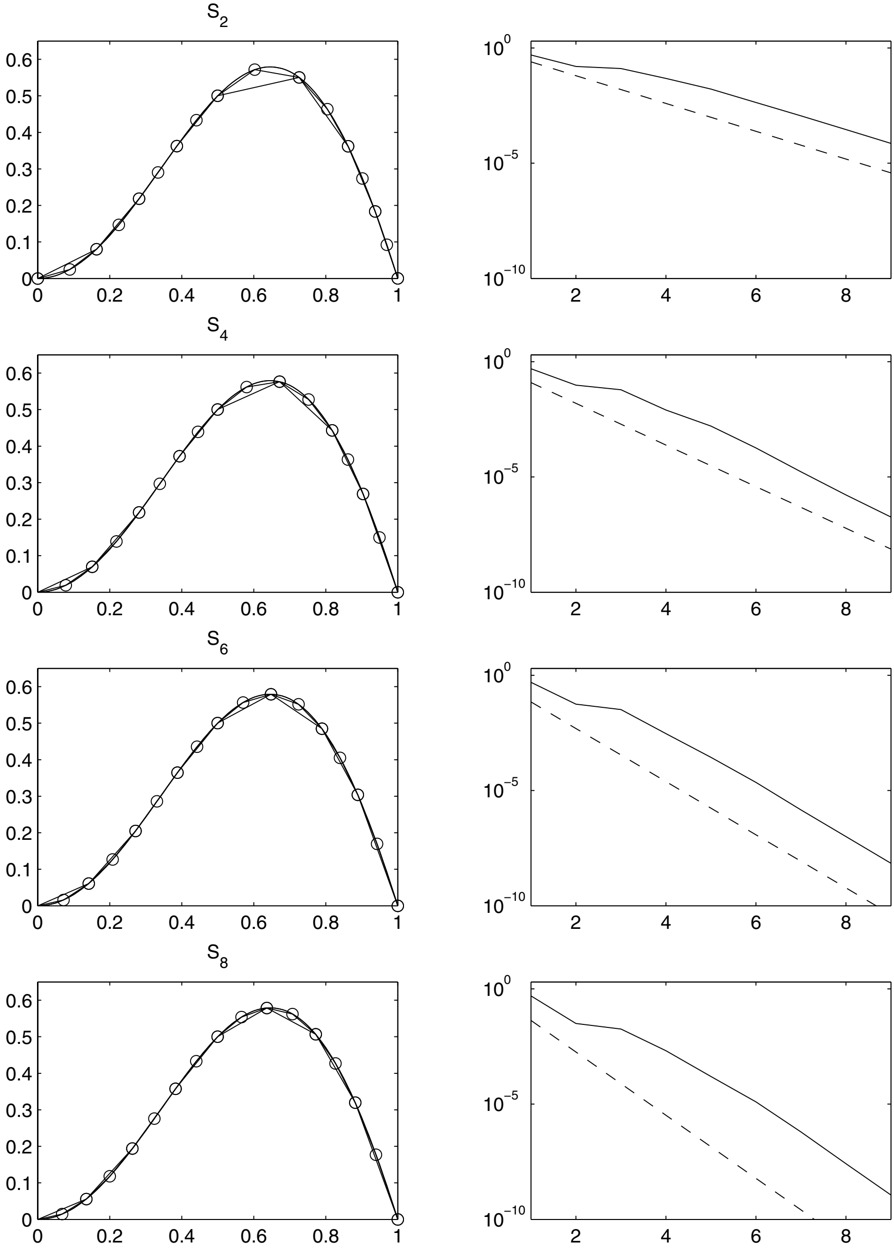


Fig. 5. Numerical study of the first Lagrange interpolation subdivision schemes, S_2 to S_8. Left column shows the normal multiresolution approximation at levels $j = 3, 4$. Right column shows the decay of wavelet coefficients as a function of level j (*solid line*) compared with the function 2^{-jQ} with $Q = 2, 3, 3.83, 4.55$ (*dashed line*)

10. A. Khodakovsky and I. Guskov. Compression of normal meshes. In *Geometric Modeling for Scientific Visualization*. Springer Verlag, 2004.
11. A. Khodakovsky, P. Schröder, and W. Sweldens. Progressive geometry compression. In *Computer Graphics (SIGGRAPH '00 Proceedings)*, pages 271–278, 2000.
12. S. Lavu, H. Choi, and R. Baraniuk. Geometry Compression of Normal Meshes using Rate-Distortion Algorithms. In *Proceedings of the Eurographics/ACM SIGGRAPH Symposium on Geometry Processing*, pages 52–61, Aachen, Germany, 2003.
13. M. Levoy, K. Pulli, B. Curless, S. Rusinkiewicz, D. Koller, L. Pereira, M. Ginzton, S. Anderson, J. Davis, J. Ginsberg, J. Shade, and D. Fulk. The digital Michelangelo project: 3D scanning of large statues. In *Computer Graphics (SIGGRAPH '00 Proceedings)*, pages 131–144, 2000.
14. C. Loop. Smooth subdivision surfaces based on triangles. Master's thesis, University of Utah, Department of Mathematics, 1987.
15. M. Lounsbery, T. D. DeRose, and J. Warren. Multiresolution surfaces of arbitrary topological type. *ACM Trans. on Graphics*, 16(1):34–73, 1997.
16. W. Sweldens. The lifting scheme: A construction of second generation wavelets. *SIAM J. Math. Anal.*, 29(2):511–546, 1997.
17. D. Zorin, P. Schröder, and W. Sweldens. Interactive multiresolution mesh editing. In *Computer Graphics (SIGGRAPH '97 Proceedings)*, pages 259–268, 1997.

Combining the Gap-Tooth Scheme with Projective Integration: Patch Dynamics

Giovanni Samaey[1], Dirk Roose[1], and Ioannis G. Kevrekidis[2]

[1] Department of Computer Science, K.U. Leuven, Celestijnenlaan 200A, 3000 Leuven, Belgium.
{giovanni.samaey,dirk.roose}@cs.kuleuven.ac.be

[2] Department of Chemical Engineering, PACM and Department of Mathematics, Princeton University, Princeton, NJ, USA.
yannis@princeton.edu

Summary. An important class of problems exhibits macroscopically smooth behaviour in space and time, while only a microscopic evolution law is known, which describes effects on fine space and time scales. A simulation of the full microscopic problem in the whole space-time domain can therefore be prohibitively expensive. In the absence of a simplified model, we can approximate the macroscopic behaviour by performing appropriately initialized simulations of the available microscopic model in a number of small spatial domains ("boxes") over a relatively short time interval. Here, we show how to obtain such a scheme, called "patch dynamics," by combining the gap-tooth scheme with projective integration. The gap-tooth scheme approximates the evolution of an unavailable (in closed form) macroscopic equation in a macroscopic domain using simulations of the available microscopic model in a number of small boxes. The projective integration scheme accelerates the simulation of a problem with multiple time scales by taking a number of small steps, followed by a large extrapolation step. We illustrate this approach for a reaction-diffusion homogenization problem, and comment on the accuracy and efficiency of the method.

Key words: equation-free multiscale computation, gap-tooth scheme, patch dynamics, homogenization

1 Introduction

For an important class of multiscale problems, a separation of scales exists between the (microscopic, detailed) level of description of the available model, and the (macroscopic, continuum) level at which one would like to observe the system. Consider, for example, a kinetic Monte Carlo model of bacterial chemotaxis [24]. A stochastic biased random walk model describes the probability of an individual bacterium to run or "tumble," based on the rotation of its flagella. Technically, it would be possible to run the detailed model in the whole space-time domain, and observe the macroscopic variables of interest, but this could be prohibitively expensive. It is known, however, that, under certain conditions, the evolution of *concentration* of the bacteria as a function of space and time obeys a deterministic evolution law on macroscopic scales sufficiently well; only, in general this evolution law cannot be written down explicitly in closed form.

The recently proposed *equation-free framework* [14, 25] can then be used to confine the use of stochastic time integration to a small fraction of the space-time domain. This framework is built around the central idea of a *coarse time-stepper*, which consists of the following steps: (1) *lifting*, i.e. the creation of *appropriate* initial conditions for the microscopic model, conditioned upon the prescribed initial conditions for the macroscopic (coarse) variables; (2) *evolution*, using the microscopic model and (possibly) some constraints for a time δt; and (3) *restriction*, i.e. the projection of the detailed solution to the macroscopic "observation" variables. This procedure amounts to a time-δt map from coarse variables to coarse variables. This coarse time-stepper can subsequently be used as "input" for time-stepper based algorithms performing macroscopic numerical analysis tasks. These include, for example, time-stepper based bifurcation codes to perform bifurcation analysis for the unavailable macroscopic equation [26, 25, 18, 19]. A coarse time-stepper can also be used in conjunction with a *projective integration method* to increase efficiency of time integration in the presence of time scale separation [6, 7].

When dealing with systems that would be described by (in our case, unavailable) *partial* differential equations, one may also be able to reduce the *spatial* complexity. For systems with one space dimension, the *gap-tooth scheme* [14, 22, 21] was proposed; it can be generalized in several space dimensions. A number of small intervals, separated by large gaps, are introduced; they qualitatively correspond to mesh points for a traditional, continuum solution of the unavailable equation. In higher space dimensions, these intervals would become *boxes* around the coarse mesh points, a term that we will also use throughout this paper. We construct a coarse time-δt map as follows. We first choose a number of macroscopic grid points. Then, we choose a small interval around each grid point; initialize the fine scale, microscopic solver within each interval consistently with the macroscopic initial condition profiles (*lift*); and provide each box with appropriate boundary conditions. Subsequently, we use the microscopic model in each interval to simulate until time δt (*run*), and obtain macroscopic information (*restrict*, e.g. by computing the average density in each box) at time δt. This amounts to a coarse time-δt map.

The gap-tooth scheme was analyzed for pure diffusion [14] and reaction-diffusion homogenization problems [22], where it was shown to be close to a finite difference space discretization for the effective equation, combined with an explicit Euler step in time. For this problem, the "microscopic" model is a partial differential equation with rapidly oscillating coefficients. The macroscopic model is the *effective* equation that describes the evolution of the average behaviour. In the limit, when the period of the oscillations becomes zero, this effective equation is the classical homogenized equation. The goal of the gap-tooth scheme is to approximate the effective equation by using only the microscopic problem inside the small boxes. A related numerical approach to obtain the effective equation was presented in [20]. There, however, the simulations were performed over the full spatial domain, instead of a number of small boxes.

Generally, a given microscopic code allows us to run with a set of pre-defined boundary conditions. It is highly non-trivial to impose macroscopically inspired boundary conditions on such microscopic codes, see e.g. [17] for a control-based strategy. This can be circumvented by introducing buffer regions at the boundary of each small box, which shield the *short-time* dynamics within the computational domain of interest from boundary effects. One then uses the microscopic code with its *built-in* boundary conditions. The gap-tooth scheme with buffers was introduced in [21, 22]. In this paper, we illustrate the relation between buffer size, time-step and accuracy numerically. More detail will be given in [23].

Since the gap-tooth time-stepper is a time-δt map from coarse variables to coarse variables, we can combine it with projective integration. The resulting scheme is called *patch dynamics* [14]. It advances the macroscopic variables on macroscopic space and time scales, using only simulations in small portions of the space-time domain. To this end, the projective integration scheme takes a few gap-tooth steps of size δt, and extrapolates the obtained macroscopic states using a (large) step size Δt. Here, we will show how to implement these ideas in the context of numerical homogenization. We will use the gap-tooth scheme with buffers as a coarse time-stepper, and combine this with a projective forward Euler step.

In their recent work, inspired by our equation-free approach, E and Engquist and collaborators address the same problem of simulating only the macroscopic behaviour of a multiscale model, see e.g. [3]. In what they call the heterogeneous multiscale method, a macroscale solver is combined with an estimator for quantities that are unknown because the macroscopic equation is not available. This estimator consequently uses appropriately constrained runs of the microscopic model [3]. It should be clear that patch dynamics does exactly this: by taking one gap-tooth step, we estimate the time derivative of the unknown effective equation, and give this as input to an ODE solver, such as projective integration. The elements of the gap-tooth step itself are based on an explicit in time macroscopic finite difference solver (the initial conditions within a box and the boundary conditions for each box are designed based on what a macroscopic finite difference scheme effectively approximates). The difference in their work is that, for conservation laws, the macro-field time derivative is estimated from the *flux* of the conserved quantity; the generalized Godunov scheme is based on this principle. Perhaps the most important difference in implementation, which also affects the numerical analysis, is the fact that we try to minimize changes to a *given microscopic simulator*. In this context, imposing the constraints required by the specific implementation proposed in [3] may be impractical (e.g. if there are constraints on macroscopic quantities that have to be estimated), undesirable (e.g. if the development of the code is expensive and time-consuming) or even impossible (e.g. if the microscopic code is a *legacy code*). Due to the use of buffers such problem are to some extent mitigated in our implementation (at the cost of simulating in larger patches).

Here, we investigate the behaviour of the patch dynamics scheme (with buffers) for a homogenization reaction-diffusion problem. The paper is organized as follows.

In Sect. 2, we formally state the problem that we want to solve. Subsequently, we summarize earlier results on the gap-tooth scheme in Sect. 3, and we describe the full patch dynamics scheme in Sect. 4. We discuss convergence in Sect. 5 and we conclude in Sect. 6.

2 Problem Statement

As a model problem, we consider the following parabolic partial differential equation,

$$\frac{\partial}{\partial t}u_\epsilon(x,t) = \frac{\partial}{\partial x}\left(a\left(\frac{x}{\epsilon}\right)\frac{\partial}{\partial x}u_\epsilon(x,t)\right) + g(u_\epsilon(x,t)), \tag{1}$$

with initial condition $u_\epsilon(x,0) = u^0(x)$ and suitable boundary conditions. In this equation, $a(y) = a\left(\frac{x}{\epsilon}\right)$ is periodic in y and ϵ is a small parameter.

Consider equation (1) with Dirichlet boundary conditions $u_\epsilon(0,t) = v_0$ and $u_\epsilon(1,t) = v_1$. According to classical homogenization theory [1], the solution to (1) can be written as an asymptotic expansion in ϵ,

$$u_\epsilon(x,t) = u_0(x,t) + \sum_{i=1}^{\infty}\epsilon^i u_i(x,\frac{x}{\epsilon},t), \tag{2}$$

where the functions $u_i(x,y,t) \equiv u_i(x,\frac{x}{\epsilon},t)$, $i = 1,2,\ldots$ are periodic in y. Here, $u_0(x,t)$ is the solution of the *homogenized equation*

$$\frac{\partial}{\partial t}u_0(x,t) = \frac{\partial}{\partial x}\left(a^*\frac{\partial}{\partial x}u_0(x,t)\right) + g(u_0(x,t)) \tag{3}$$

with initial condition $u_0(x,0) = u^0(x)$ and Dirichlet boundary conditions $u_0(0,t) = v_0$ and $u_0(1,t) = v_1$; a^* is the constant effective diffusion coefficient, given by

$$a^* = \int_0^1 a(y)\left(1 - \frac{\mathrm{d}}{\mathrm{d}y}\chi(y)\right)\mathrm{d}y, \tag{4}$$

and $\chi(y)$ is the periodic solution of

$$\frac{\mathrm{d}}{\mathrm{d}y}\left(a(y)\frac{\mathrm{d}}{\mathrm{d}y}\chi(y)\right) = \frac{\mathrm{d}}{\mathrm{d}y}a(y), \tag{5}$$

the so-called *cell problem*. The solution of (5) is only defined up to an additive constant, so we impose the extra condition

$$\int_0^1 \chi(y)\mathrm{d}y = 0. \tag{6}$$

From this cell problem, we can derive $u_1(x,y,t) = \frac{\partial u_0}{\partial x}\chi(y)$.

These asymptotic expansions have been rigorously justified in the classical book [1]. Under appropriate smoothness assumptions, one can obtain pointwise convergence of u_ϵ to u_0 as $\epsilon \to 0$. Therefore, we can write

$$\|u_\epsilon(x,t) - u_0(x,t)\| \leq C_0\epsilon, \tag{7}$$

where $\|f(x)\| \equiv \|f(x)\|_\infty = \max_x |f(x)|$ denotes the ∞-norm of f. Throughout this text, whenever we use $\|\cdot\|$, we mean the ∞-norm.

For $u(x,t)$ sufficiently smooth, the averaged function

$$U(x,t) = \mathcal{S}_h(u)(x,t) := \frac{1}{h}\int_{x-\frac{h}{2}}^{x+\frac{h}{2}} u(\xi,t)\mathrm{d}\xi$$

can be asymptotically expanded in h as follows,

$$U(x,t) = u(x,t) + \sum_{l=1}^{\infty}\left(\frac{h}{2}\right)^{2l}\frac{1}{(2l+1)!}\,\frac{\partial^{2l}}{\partial^{2l}\xi}u(\xi,t)\bigg|_{\xi=x}.$$

The difference between the homogenized solution $u_0(x,t)$ and the averaged solution $U(x,t) = h^{-1}\int_{x-\frac{h}{2}}^{x+\frac{h}{2}} u_\epsilon(\xi,t)\mathrm{d}\xi$ is bounded by

$$\|U(x,t) - u_0(x,t)\| \leq C_1h^2 + C_2\epsilon.$$

Therefore, the averaged solution is a good approximation of the homogenized solution for sufficiently small box width h.

The goal of the gap-tooth scheme is to approximate the solution $U(x,t)$, while only making use of the detailed model (1). Moreover, we assume that a time integration code for (1) has already been written and is available with a number of *standard* boundary conditions, such as no-flux or Dirichlet.

3 The Gap-Tooth Scheme

We briefly revise the gap-tooth algorithm as it was introduced in [21, 22]. Suppose we want to obtain the solution of the *unknown* equation (3) on the interval $[0,1]$, using an equidistant, macroscopic mesh $\varPi(\varDelta x) := \{0 = x_0 < x_1 = x_0 + \varDelta x < \ldots < x_N = 1\}$. To this end, consider a small interval (*tooth*, "inner" box) of length $h \ll \varDelta x$ centered around each mesh point, as well as an interval of size $H > h$ (the *buffer* box). (See Fig. 1.) To perform time integration using the microscopic model (1) in each box, we provide each box with initial and boundary conditions as follows.

Initial Condition

We define the initial condition by constructing a polynomial, based on the (given) box averages U_i^n, $i = 1, \ldots, N$,

$$\tilde{u}^i(x,t_n) \approx p_i^d(x;t_n), \qquad x \in [x_i - \frac{H}{2}, x_i + \frac{H}{2}], \tag{8}$$

where $p_i^d(x;t_n)$ denotes a polynomial of (even) degree d, and H denotes the size of the buffer. We require that the approximating polynomial has the same box averages as the initial condition in box i and in $\frac{d}{2}$ boxes to the left and to the right. This gives us

$$\frac{1}{h}\int_{x_{i+j}-\frac{h}{2}}^{x_{i+j}+\frac{h}{2}} p_i^d(\xi;t_n)\mathrm{d}\xi = U_{i+j}^n, \qquad j = -\frac{d}{2},\dots,\frac{d}{2}. \tag{9}$$

The box averages are computed over the inner box of width h. One can easily check that

$$\mathcal{S}_h(p_i^d)(x,t_n) = \sum_{j=-\frac{d}{2}}^{\frac{d}{2}} U_{i+j}^n L_{i,j}^d(x), \qquad L_{i,j}^d(x) = \prod_{\substack{l=-\frac{d}{2}\\ l\neq j}}^{\frac{d}{2}} \frac{(x - x_{i+l})}{(x_{i+j} - x_{i+l})} \tag{10}$$

where $L_{i,j}^d(x)$ denotes a Lagrange polynomial of degree d.

Boundary Conditions

The time integration of the microscopic model in each box should provide information on the evolution of the global problem at that location in space. It is therefore crucial that the boundary conditions are chosen such that the solution inside each box evolves *as if it were embedded in a larger domain*. We already mentioned that, in many cases, it is not possible or convenient to impose macroscopically-inspired constraints on the microscopic model (e.g. as boundary conditions). However, we can introduce a larger box of size $H \gg h$ around each macroscopic mesh point, but still only use (for macro-purposes) the evolution over the smaller, inner box. The simulation can subsequently be performed using any of the *built-in* boundary conditions of the microscopic code. Lifting and evolution (using *arbitrary* available boundary conditions) are performed in the larger box; yet the restriction is done by processing the solution (here taking its average) over the inner, small box only. The goal of the additional computational domains, the *buffers*, is to buffer the solution inside the small box from the artificial disturbance caused by the boundary conditions. This can be accomplished over *short enough* times, provided the buffers are *large enough*; analyzing the method is tantamount to making these statements quantitative.

The idea of using a buffer region was also used in the multiscale finite element method (oversampling) of Hou [12] to eliminate the boundary layer effect; also Hadjiconstantinou makes use of overlap regions to couple a particle simulator with a continuum code [11]. If the microscopic code allows a choice of different types of microscopic boundary conditions, selecting the size of the buffer may also depend on this choice.

The Algorithm

The complete *gap-tooth* algorithm to proceed from t_n to $t_{n+1} = t_n + \delta t$ is given below:

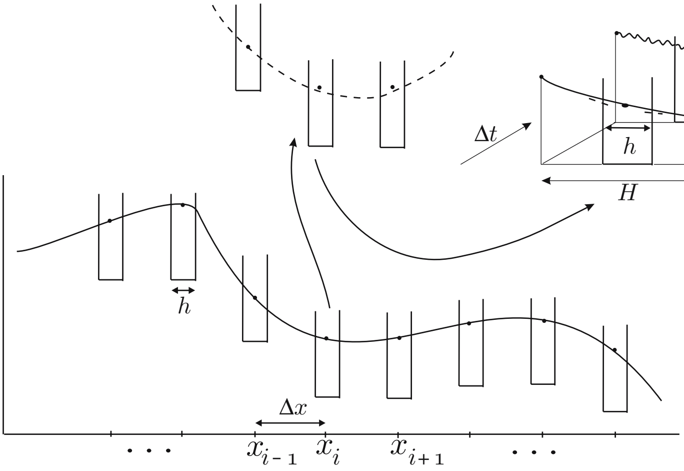


Fig. 1. A schematic representation of a gap-tooth time step with buffer boxes. We choose a number of boxes of size h around each macroscopic mesh point x_i and interpolate the initial averages (dots) in a number of boxes around x_i. This polynomial is taken as the initial condition around x_i, and simulation is performed in boxes of size H

1. At time t_n, construct the initial condition $\tilde{u}^i(x, t_n)$, $i = 0, \ldots, N$, using the box averages U_j^n ($j = 0, \ldots, N$), as defined in (8-9).
2. Compute $\tilde{u}^i(x, t)$ by solving the equation (1) in the interval $[x_i - \frac{H}{2}, x_i + \frac{H}{2}]$ until time $t_{n+1} = t + \delta t$ with *some* boundary conditions. The boundary conditions can be anything that the time integration routine allows.
3. Compute the box average U_i^{n+1} at time t_{n+1}.

It is clear that this amounts to a map of the macroscopic variables $U^n \approx U(n\delta t)$ at time t_n, to the macroscopic variables U^{n+1} at time $t_{n+1} = t_n + \delta t$, i. e. a "coarse to coarse" time-δt map. We write this map as follows,

$$U^{n+1} = S_d(U^n; t_n + \delta t), \qquad (11)$$

where S represents the numerical time-stepping scheme for the macroscopic (coarse) variables and d denotes the degree of the interpolating polynomial. Here, $U(t)$ is the exact solution of a system of ordinary differential equations that represent a method of lines semi-discretization of the effective equation, while $U^n \approx U(n\delta t)$ represents a numerical approximation to this solution.

Microscopic Simulators

Above, we have assumed that the microscopic model is a partial differential equation. However, some microscopic simulators are of a different nature, e.g. kinetic Monte Carlo or molecular dynamics codes. In fact, this is the case where we expect our method to be most useful. In this case, several complications arise. First of all, the choice of the box width h becomes important, since there will generally exist a trade-off between statistical accuracy (e.g. enough sampled particles) and spatial resolution.

Second, the *lifting* step, i.e. the construction of box initial conditions, also becomes more involved. In general, the microscopic model will have many more degrees of freedom, the *higher order moments* of the evolving distribution. These will quickly become slaved to the governing moments (the ones where the lifting is conditioned upon), see e.g. [14, 18], but it might be better to do a "constrained" run before initialization to create "mature" initial conditions [13, 8, 15, 4, 2].

Finally, determining which and how many neighbouring boxes are needed for the interpolation polynomial is a delicate issue. The degree of the interpolation polynomial determines how many spatial derivatives are initialized consistently in each box. This is related with the order of the partial differential equation, i.e. the order of the highest spatial derivative. A systematic way to estimate this order, without having the macroscopic equation, is given in [16].

Numerical experiments with the gap-tooth scheme using a kinetic Monte Carlo microscopic model are presented in [10, 5].

4 Patch Dynamics

Once we have constructed a coarse time-stepper that exploits the *spatial* scale separation, we can combine it with the projective integration scheme [6] to exploit *time* scale separation. The crucial idea is that one can estimate the time derivative of the macroscopic system using the gap-tooth scheme, and perform a large extrapolation step.

We will briefly summarize the projective integration scheme as it was presented in [14, 6], and subsequently show how to combine it with the gap-tooth scheme.

The Projective Integration Scheme

Let $\Delta t \gg \delta t$ be a large time step (commensurate with the slow dynamics), and denote the numerical approximations of the coarse solution $U(t)$ as $U^n \approx U(n\Delta t)$. Suppose we are given a (coarse) time-stepper that permits to compute

$$U^{\alpha,n} = S(U^{0,n}, t_n + \alpha\delta t), \tag{12}$$

for $\alpha \in [0,1]$. Here, $U^{\alpha,n} \approx U(n\Delta t + \alpha\delta t)$, and therefore we have $U^{0,n} = U^n$ by consistency.

We cannot afford to compute $U^{n+1} \approx U((n+1)\Delta t)$ this way, because $\Delta t \gg \delta t$. Instead, we compute U^{n+1} by extrapolation, using a coarse *projective* scheme of the type

$$U^{n+1} = U^{\alpha,n} + (\Delta t - \alpha\delta t)\tilde{F}(U^{k,n}), \tag{13}$$

where we approximate the time derivative by

$$\tilde{F}(U^{\alpha,n}) = \frac{U^{1,n} - U^{\alpha,n}}{(1-\alpha)\delta t} \tag{14}$$

for some α in $[0,1)$, which has to be chosen large enough to ensure that lifting errors have been removed by the microscopic simulation. (The higher order moments are then slaved to the lower order ones.) Here, assuming that lifting errors can be neglected, we choose $\alpha = 0$. As discussed in Sect. 3, techniques to minimize lifting errors can be devised, and therefore the choice $\alpha = 0$ is not that artificial. The time step Δt has to be chosen such that the resulting macroscopic time-stepper is stable. In order to increase the stability region, one could take $k > 1$ gap-tooth steps before performing the extrapolation, see [14, 9] for more details.

Patch Dynamics for the Homogenization Problem

It is clear that the gap-tooth time-stepper, constructed in Sect. 3, can be considered as a coarse time-stepper for the projective integration scheme. Here, for simplicity, we take one gap-tooth step with $\alpha = 0$ for the projective integration. (Taking $k > 1$ gap-tooth steps might lead to an increased stability region [6].) This gives the following algorithm (Fig. 2).

1. At time t_n, take *one* gap-tooth step,
$$U^{1,n} = S_d(U^n, t_n + \delta t)$$
2. Compute the approximate effective time derivative as
$$\tilde{F}(U^n) = \frac{U^{1,n} - U^{0,n}}{\delta t}$$
3. Perform a projective forward Euler step, using this approximate time derivative.
$$U^{n+1} = U^n + \Delta t \tilde{F}(U^n)$$

5 Convergence Results

Theoretical Results

We can easily obtain a convergence result for patch dynamics taking advantage of a theorem from [3].

If we would have the macroscopic equation (3), we could obtain a method of lines semi-discretization by replacing the spatial derivatives by finite differences of order d, yielding a system of ordinary differential equations of the form

$$\dot{U} = F(U). \tag{15}$$

Consider as a macroscopic solver a standard forward Euler scheme

$$U^{n+1} = U^n + \Delta t F(U^n). \tag{16}$$

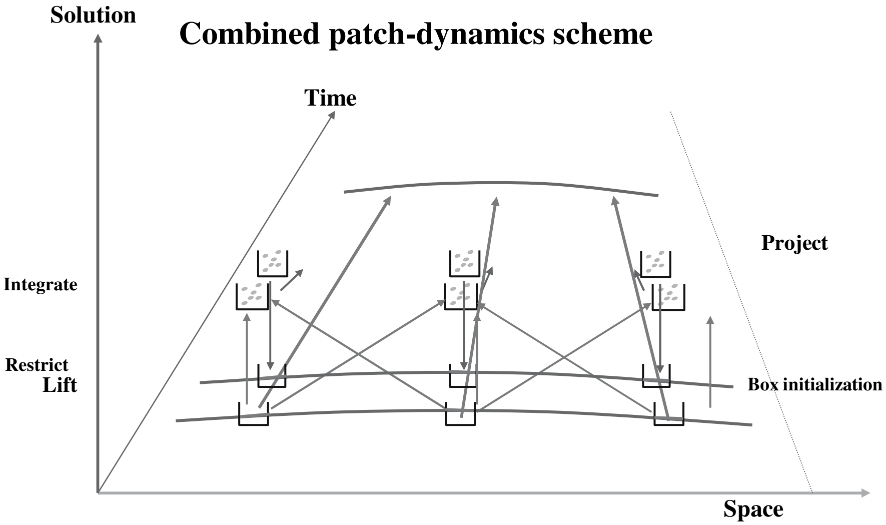


Fig. 2. A schematic representation of a patch dynamics scheme. After a gap-tooth step, we extrapolate over a large time step

Because the macroscopic equation is not available, we need to estimate $F(U^n)$, by taking a gap-tooth step,

$$U^{n+1} = U^n + \Delta t \tilde{F}(U^n), \qquad \tilde{F}(U^n) = \frac{U^{1,n} - U^{0,n}}{\delta t}. \tag{17}$$

Then, under appropriate assumptions on F and U, we can state that [3]

Theorem 1. *Patch dynamics is stable if the forward Euler scheme is stable. Moreover, the total discretization error is bounded by*

$$\|U^n - U(t_n)\| \leq C(\Delta t + \max_{0 \leq k \leq \frac{t_n}{\Delta t}} \|F(U^k) - \tilde{F}(U^k)\|),$$

where $U(t_n)$ is the exact solution of the semi-discrete system (15).

Thus, what remains is to estimate $\|F(U^k) - \tilde{F}(U^k)\|$. We mention a theorem that shows the error that is made in the estimation of the time derivative.

Theorem 2. *Consider the model problem (1). When performing one gap-tooth step with Dirichlet boundary conditions,*

$$U^{1,n} = S_d(U^n, t_n + \delta t),$$

we can bound the error

$$\|\tilde{F}(U^n) - F(U^n)\| \leq C\left(h^2 + \frac{\epsilon}{\delta t} + \delta t^2 + E(\delta t, H)\right), \tag{18}$$

where $F(U^n)$ is the time derivative as defined by (15) for (3) and

$$\lim_{H\to\infty} E(\delta t, H) = \lim_{\delta t \to 0} E(\delta t, H) = 0$$

In this theorem, $E(\delta t, H)$ represents the error term that is due to the buffers. We see that this term can be made arbitrarily small by choosing H large enough and δt small enough. A heuristic to make this error term comparable in size with the others is given in [23], where this theorem is proved. Also, due to the term $\frac{\epsilon}{\delta t}$, it is impossible to obtain convergence when the small scale ϵ is fixed. In this context, the theorem has to be seen as a bound for optimal error.

Numerical Results

Consider the following model problem,

$$\frac{\partial}{\partial t} u_\epsilon(x,t) = \frac{\partial}{\partial x}\left(a(\frac{x}{\epsilon})\frac{\partial}{\partial x}u_\epsilon(x,t)\right), \qquad a(\frac{x}{\epsilon}) = 1.1 + \sin(2\pi\frac{x}{\epsilon}) \tag{19}$$

with $\epsilon = 1 \cdot 10^{-5}$, $x \in [0,1]$, initial conditions $u_\epsilon(x,0) = 1 - 4(x-0.5)^2$, and Dirichlet boundary conditions $u_\epsilon(0,t) = u_\epsilon(1,t) = 0$. We choose $\epsilon = 1 \cdot 10^{-5}$. To solve the microscopic problem, we use a standard finite difference discretization in space and a variable step/variable order time integration method (`ode23s` in Matlab), with mesh width $\delta x = 1 \cdot 10^{-7}$. The corresponding homogenized equation is given by

$$\frac{\partial}{\partial x}\left(a^*\frac{\partial}{\partial x}u_0(x,t)\right), \qquad a^* \approx 0.45825686. \tag{20}$$

We first investigate the error in the time derivative estimator (18) as a function of buffer size and time step. To this end, we take a gap-tooth step with $\Delta x = 0.1$ and $h = 2 \cdot 10^{-3}$. We let the buffer size H vary from $2 \cdot 10^{-3}$ to $3 \cdot 10^{-2}$, and the time step from $1 \cdot 10^{-7}$ to $5 \cdot 10^{-6}$. Inside each box, we use the microscopic solver with Dirichlet boundary conditions. For the lifting step, we use quadratic interpolation, which is equivalent to a standard second order finite difference approximation of the diffusion term [22]. Figure 3 shows the evolution of the error as a function of buffer width and time step. The error is measured as the difference of (18) with respect to the second order finite difference discretization of the homogenized equation (3),

$$F(U_j^n) = a^* \frac{U_{j+1}^n - 2U_j^n + U_{j-1}^n}{\Delta x^2}$$

The leftmost picture shows the error as a function of buffer size for a fixed time step. We see that the error decreases exponentially with the buffer size. We see that the smaller the time step, the faster the initial decay of the error, but the optimal error that can be obtained is larger. This is due to the term $\frac{\epsilon}{\delta t}$ in (18). This is made clear in the rightmost picture, which shows the error as a function of time step, for a fixed buffer size. We expect that for a fixed buffer size, the error decreases with decreasing time step, and the optimal error curve has a slope $\frac{1}{\delta t}$. This is visible only for very small δt, due to interference with the time integration error of the microscopic solver.

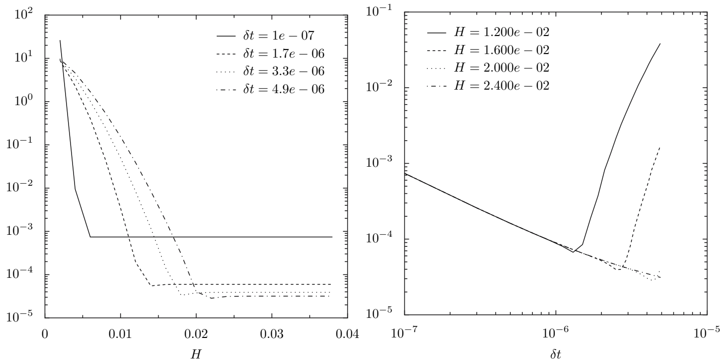


Fig. 3. Left: Error of gap-tooth time derivative estimate in function of buffer size for the given values of the time step δt. Right: Error of gap-tooth time derivative estimate in function of time step for the given values of the buffer size

In order to show the convergence in the absence of this term, we performed the same experiment, but we replaced the microscopic solver with a finite difference discretization of the homogenized equation. The result is shown in Fig. 4. We see that we have convergence up to 8 digits. The remaining digits are lost due to cancellation errors in estimating the derivative.

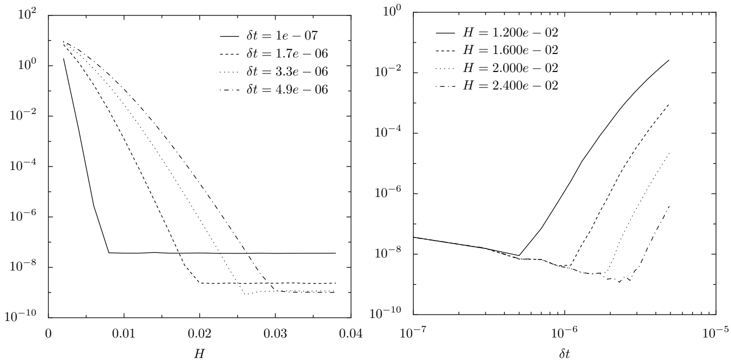
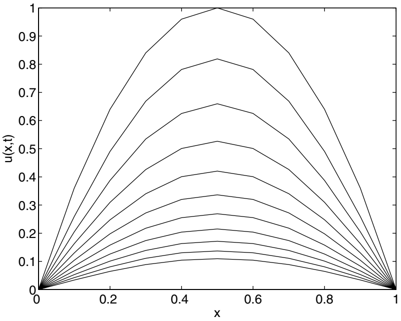


Fig. 4. Left: Error of gap-tooth time derivative estimate in function of buffer size for a the give values of the time step δt, using the homogenized equation as a microscopic solver. Right: Error of gap-tooth time derivative estimate in function of time step for the given value of the buffer size H

Next, we use patch dynamics to integrate this system until $t = 1$. We choose $\Delta t = 1 \cdot 10^{-3}$. In this case $\nu = \frac{\Delta t}{\Delta x^2} = 0.1$, so the macroscopic scheme is certainly stable. Based on the previous tests, we choose a buffer size of $H = 7 \cdot 10^{-3}$ and $\delta t = 1 \cdot 10^{-6}$. Figure 5 shows the results. We depict the error with respect to the finite difference scheme on the homogenized equation in the accompanying table.

We see that the scheme has the ability of computing the solution to the unavailable homogenized equation with 3 correct digits, using simulations on only 7% of the spatial domain and 0.1% of the time domain (a gap-tooth time step of 10^{-6} versus a macroscopic time step of 10^{-3}).



time	error
$5 \cdot 10^{-2}$	$6.4683 \cdot 10^{-4}$
$1 \cdot 10^{-1}$	$1.9202 \cdot 10^{-3}$
$1.5 \cdot 10^{-1}$	$8.3803 \cdot 10^{-4}$
$2 \cdot 10^{-1}$	$1.2133 \cdot 10^{-3}$
$2.5 \cdot 10^{-1}$	$9.8365 \cdot 10^{-4}$
$3 \cdot 10^{-1}$	$1.2383 \cdot 10^{-3}$
$3.5 \cdot 10^{-1}$	$1.0178 \cdot 10^{-3}$
$4 \cdot 10^{-1}$	$7.4551 \cdot 10^{-4}$
$4.5 \cdot 10^{-1}$	$7.5973 \cdot 10^{-4}$
$5 \cdot 10^{-1}$	$5.6377 \cdot 10^{-4}$

Fig. 5. Left: Solution of the unavailable homogenized equation using patch dynamics, at $t = 0, 5 \cdot 10^{-2}, 1 \cdot 10^{-1}, 1.5 \cdot 10^{-1} \ldots, 5 \cdot 10^{-1}$. Right: Error in maximum norm with respect to the finite difference comparison scheme for the homogenized equation

6 Conclusions

We described a patch dynamics algorithm for the numerical simulation of multiscale problems. This scheme simulates the macroscopic behaviour over a macroscopic domain when only a microscopic model is explicitly available; it only uses appropriately initialized short simulations over small sub-domains. We numerically illustrated convergence properties of this scheme for a parabolic homogenization problem, and related these properties to theoretical results that were obtained in [3].

We showed that our method approximates a finite difference scheme for the homogenized equation when the buffer regions are chosen large enough with respect to the gap-tooth time step. Our analysis revealed that the presence of microscopic scales introduces errors that can be made small, but not arbitrarily small. In this sense, there is an optimal accuracy that can be reached with these methods.

Numerical simulations on a model problem show that it is possible to obtain simulation results over large space-time domains, using only a fraction of the computational complexity that would be needed by a full microscopic simulation.

Acknowledgments

GS is a Research Assistant for the Fund of Scientific Research – Flanders. This work has been partially supported by grant IUAP/V/22 (Belgian Federal Science Policy

Office) and by the FWO grant G.0130.03 (GS, DR), and an NSF/ITR grant and DOE (IGK).

References

1. A. Bensoussan, J. L. Lions, and G. Papanicolaou. *Asymptotic analysis of periodic structures*, volume 5 of *Studies in Mathematics and its Applications*. North-Holland, Amsterdam, 1978.
2. W. E. Analysis of the heterogeneous multiscale method for ordinary differential equations. *Comm. Math. Sci.*, 1(3):423–436, 2003.
3. W. E and B. Engquist. The heterogeneous multi-scale methods. *Comm. Math. Sci.*, 1(1):87–132, 2003.
4. I. Fatkullin and E. Vanden Eijnden. A computational strategy for multiscale chaotic systems with applications to Lorenz 96 model. 2004. Preprint.
5. C. W. Gear and I. G. Kevrekidis. Boundary processing for Monte Carlo simulations in the gap-tooth scheme. Technical Report 2002-031N, NEC Research Institute, 2002.
6. C. W. Gear and I. G. Kevrekidis. Projective methods for stiff differential equations: problems with gaps in their eigenvalue spectrum. *SIAM Journal of Scientific Computation*, 24(4):1091–1106, 2003. Can be obtained as NEC Report 2001-029, `http://www.neci.nj.nec.com/homepages/cwg/projective.pdf`.
7. C. W. Gear and I. G. Kevrekidis. Telescopic projective methods for stiff differential equations. *Journal of Computational Physics*, 187(1):95–109, 2003.
8. C. W. Gear and I. G. Kevrekidis. Constraint-defined manifolds: a legacy code approach to low-dimensional computation. *J. Sci. Comp.*, 2004. in press.
9. C. W. Gear, I. G. Kevrekidis, and C. Theodoropoulos. "Coarse" integration/bifurcation analysis via microscopic simulators: micro-Galerkin methods. *Computers and Chemical Engineering*, 26(7-8):941–963, 2002.
10. C. W. Gear, J. Li, and I. G. Kevrekidis. The gap-tooth method in particle simulations. *Physics Letters A*, 316:190–195, 2003. Can be obtained as physics/0303010 at `arxiv.org`.
11. N. G. Hadjiconstantinou. Hybrid atomistic-continuum formulations and the moving contact-line problem. *Journal of Computational Physics*, 154:245–265, 1999.
12. T. Y. Hou and X. H. Wu. A multiscale finite element method for elliptic problems in composite materials and porous media. *Journal of Computational Physics*, 134:169–189, 1997.
13. G. Hummer and I. G. Kevrekidis. Coarse molecular dynamics of a peptide fragment: free energy, kinetics and long-time dynamics computations. *Journal of Chemical Physics*, 118(23):10762–10773, 2003. Can be obtained as physics/0212108 at `arxiv.org`.
14. I. G. Kevrekidis, C. W. Gear, J. M. Hyman, P. G. Kevrekidis, O. Runborg, and C. Theodoropoulos. Equation-free multiscale computation: enabling microscopic simulators to perform system-level tasks. *Comm. Math. Sciences*, 1(4):715–762, 2003.
15. I. G. Kevrekidis, C. W. Gear, T. Kaper and A. Zagaris. Projecting to a slow manifold: singularly perturbed systems and legacy codes. submitted, 2004.
16. J. Li, P. G. Kevrekidis, C. W. Gear, and I. G. Kevrekidis. Deciding the nature of the “coarse equation” through microscopic simulations: the baby-bathwater scheme. *SIAM Multiscale modeling and simulation*, 1(3):391–407, 2003.
17. J. Li, D. Liao, and S. Yip. Imposing field boundary conditions in MD simulation of fluids: optimal particle controller and buffer zone feedback. *Mat. Res. Soc. Symp. Proc*, 538:473–478, 1998.

18. A. G. Makeev, D. Maroudas, and I. G. Kevrekidis. Coarse stability and bifurcation analysis using stochastic simulators: kinetic Monte Carlo examples. *Journal of Chemical Physics*, 116:10083–10091, 2002.
19. A. G. Makeev, D. Maroudas, A. Z. Panagiotopoulos, and I. G. Kevrekidis. Coarse bifurcation analysis of kinetic Monte Carlo simulations: a lattice-gas model with lateral interactions. *Journal of Chemical Physics*, 117(18):8229–8240, 2002.
20. O. Runborg, C. Theodoropoulos, and I. G. Kevrekidis. Effective bifurcation analysis: a time-stepper based approach. *Nonlinearity*, 15:491–511, 2002.
21. G. Samaey, I. G. Kevrekidis, and D. Roose. Damping factors for the gap-tooth scheme. In S. Attinger and P. Koumoutsakos, editors, *Multiscale modelling and simulation*, volume 39 of *Lecture Notes in Computational Science and Engineering*, pages 94–103. Springer, 2004.
22. G. Samaey, D. Roose, and I. G. Kevrekidis. The gap-tooth scheme for homogenization problems. *SIAM Multiscale modelling and simulation*, 2005. In press. Can be obtained as physics/03120004 at `arxiv.org`.
23. G. Samaey, I.G. Kevrekidis and D. Roose. Patch dynamics with buffers for homogenization problems. Submitted to *Journal of Computational Physics*, 2004. Can be obtained as physics/0412005 at `arxiv.org`.
24. S. Setayeshar, C. W. Gear, H. G. Othmer, and I. G. Kevrekidis. Application of coarse integration to bacterial chemotaxis. *SIAM Multiscale modelling and simulation*, 2004. In press. Can be obtained as physics/0308040 at `arxiv.org`.
25. C. Theodoropoulos, Y. H. Qian, and I. G. Kevrekidis. Coarse stability and bifurcation analysis using time-steppers: a reaction-diffusion example. In *Proc. Natl. Acad. Sci.*, volume 97, pages 9840–9843, 2000.
26. C. Theodoropoulos, K. Sankaranarayanan, S. Sundaresan, and I. G. Kevrekidis. Coarse bifurcation studies of bubble flow Lattice–Boltzmann simulations. *Chem. Eng. Sci.*, 2004. in press. Can be obtained as nlin.PS/0111040 from `arxiv.org`.

Multiple Time Scale Numerical Methods for the Inverted Pendulum Problem

Richard Sharp[1], Yen-Hsi Tsai[2], and Björn Engquist[2]

[1] Program in Applied and Computational Mathematics, Princeton University, NJ 08544
rsharp@math.princeton.edu

[2] Department of Mathematics, University of Texas at Austin, 1 University Station C1200, Austin, Texas 78712, USA
ytsai@math.utexas.edu, engquist@math.utexas.edu

Summary. In this article, we study a class of numerical ODE schemes that use a time filtering strategy and operate in two time scales. The algorithms follow the framework of the heterogeneous multiscale methods (HMM) [1]. We apply the methods to compute the averaged path of the inverted pendulum under a highly oscillatory vertical forcing on the pivot. The averaged equation for related problems has been studied analytically in [9]. We prove and show numerically that the proposed methods approximate the averaged equation and thus compute the average path of the inverted pendulum.

Key words: multiscale, ordinary differential equation, averaging

1 Introduction

The focus of this paper is the application of numerical methods for dynamical systems whose solutions oscillate around a slow manifold. We assume that the oscillations take place on a much faster time scale than the rate of change of the slow manifold with respect to time. More precisely, we hypothesize that the wavelength of the fast oscillations is proportional to a positive constant ϵ, and that in an $\mathcal{O}(\epsilon)$ time interval the slow manifold changes by at most $\mathcal{O}(\epsilon)$. This is the case in the inverted pendulum problem with highly oscillatory forcing, and we will show that our methods yield consistent approximations to the averaged equations.

We consider the inverted pendulum example, in which the pivot of a rigid pendulum with length l is attached to a strong periodic forcing, vibrating above the horizontal axis with amplitude ϵ and frequency $C\epsilon^{-1}$. The system has one degree of freedom, and can be described by the angle, θ, between the pendulum arm and the upward vertical direction, as shown in Fig. 1. The motion is determined by

$$l\ddot{\theta} = \left(g + \frac{1}{\epsilon}\sin\left(2\pi\frac{t}{\epsilon}\right)\right)\sin(\theta), \tag{1}$$

with initial conditions $\theta(0) = \theta_0, \dot{\theta}(0) = \omega_0$. When ϵ^{-1} is sufficiently large, and θ_0 and ω_0 are sufficiently close to 0, the pendulum will oscillate slowly back and forth with displacement $\theta < \theta_{\text{max}}$. The period of the oscillation is "independent" of the forcing frequency $C\epsilon^{-1}$, and in addition to this slow motion, the trajectory of θ also exhibits fast oscillations with amplitude and period proportional to ϵ. The behavior of this system and other generalizations are analyzed analytically in [9]. In short, the governing second order equation for these stiff problems take the general form

$$\ddot{x} = \epsilon^{-1} a \left(\frac{t}{\epsilon}\right) f(x), \qquad x(0) = x_0, \quad \dot{x}(0) = y_0; \tag{2}$$

where a is a smooth, 1-periodic function, $0 < \epsilon \ll 1$, and f is a bounded smooth function. It will be convenient to consider (2) as a system of first-order equations where $dx/dt = y$ and $dy/dt = \epsilon^{-1} a(t/\epsilon) f(x)$.

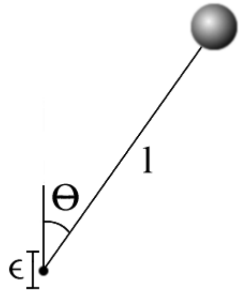


Fig. 1. The inverted pendulum. A mass is connected to an arm of length l which makes an angle θ with the vertical axes

Typically, the computational difficulties in solving the above system stem from the short wavelength in the periodic function $a(t/\epsilon)$. If explicit time stepping methods are employed to solve such a system, the corresponding stability condition requires the step size Δt to be proportional to ϵ, and if the solution is needed in an interval with length independent of ϵ, the computation would require $\mathcal{O}(\epsilon^{-1})$ operations, rendering the solution method unusable if ϵ is very small. On the other hand, implicit schemes with larger time steps typically damp out the oscillations or represent them inaccurately. There is also the problem of inverting the corresponding nonlinear system.

In many situations, one is interested in a set of quantities X that are derived from the solution of the given stiff system. Typically, these quantities change slowly in time. A pedagogical example, pointed out to the authors by G. Dahlquist, is the drift path of a mechanical alarm clock due to its shaking and rattling when it is set off on a hard surface. If the slowly changing quantities X depend only locally in time on the fast oscillations, it is then reasonable to devise a scheme that tracks the slow quantities by measuring the effects of the fast solutions only locally in time. Herein lies the possibility of reducing the computational complexity. Under this context, and recasting (2) as a first-order system, it is natural to look for an explicit numerical method that appears in the general form:

$$X_{n+1} = Q_H(\tilde{F}[x_n(t)], X_n, X_{n-1}, \cdots) \quad X(0) = X_0, \tag{3}$$

where $H = t^{n+1} - t^n$ denotes the slow time scale step size, h denotes the fast time scale step size, $x_n(t)$ is the microscopic data, and Q_H and $\tilde{F}$ denote some suitable operators; we will make precise all of these in the following. The functional $\tilde{F}$ relates x, the solution to the stiff problem, to the slowly changing quantities X.

In our specific problem with model (2), $x(t)$ is $\theta(t)$, and $X(t)$ is the average over the period $[t-\epsilon/2, t+\epsilon/2]$, $X(t) = \langle x \rangle = \frac{1}{\epsilon}\int_{t-\epsilon/2}^{t+\epsilon/2} x(s/\epsilon)\mathrm{d}s$, and as shown in [9], it satisfies the averaged effective equation

$$\ddot{X} = \langle a \rangle f(X) - \langle v^2 \rangle f(X) f'(X) + E, \quad X(0) = X_0. \tag{4}$$

The "velocity", v is a function of the "acceleration" a,

$$v(t) = \int_{s_0}^{t} \left(a(\tfrac{s}{\epsilon}) - \langle a \rangle \right) ds \tag{5}$$

and s_0 is selected so that $\langle v \rangle = 0$. The error in (4) is small, $E \sim \mathcal{O}(\sqrt{\epsilon})$, [9].

Many existing methods can be cast into the above form (3,4), with X directly related to either the strong or weak limit of the original variable x. For example, in the methods proposed in [5] and [10] for oscillatory ODEs, X is the envelope of x. The given stiff system is then integrated from current state X_n for some integer number of periods $\eta = C\epsilon$, fully resolving the oscillation with step size h. The method then estimates and returns the time derivative of the envelop, and finally Q_H corresponds to the discrete solution operator of the macroscopic scheme.

In [1], the general framework of Heterogeneous Multiscale Methods (HMM) was proposed. Under this framework approximation schemes can be conveniently constructed and analyzed for general problems involving multiple separated temporal and spatial scales. In [3], under the HMM framework, we proposed and analyzed a class of HMM ODE schemes that operate in two time scales. There, the operator $\tilde{F}$ plays the role of approximating the average force by time filtering the microscopic evolution in the time interval $[t_n - \eta/2, t_n + \eta/2]$, using convolution with a suitable kernel. If the forward Euler scheme is adopted in Q_H, then the schemes appear to be

$$X_{n+1} = X_n + H \cdot \tilde{F}[x_n(t)].$$

The equations considered in [3] are of the form

$$\begin{cases} \frac{\mathrm{d}}{\mathrm{d}t}x = A(\frac{t}{\epsilon})x + f(x,y) \\ \frac{\mathrm{d}}{\mathrm{d}t}y = g(x,y) \end{cases}.$$

It is proved there that if f does not depend on the phase of x, i.e. if $f(\mathrm{e}^{i\theta}x, y) = f(x,y)$ for any $\theta \in [0, 2\pi)$, the constructed approximations converges to the solution of the averaged equation, and the averaged equation is of the following form:

$$\begin{cases} \frac{\mathrm{d}}{\mathrm{d}t}\bar{x} = \frac{1}{2\pi}\int_0^{2\pi} \mathrm{e}^{-\mathrm{i}\phi} f(\mathrm{e}^{\mathrm{i}\phi}\bar{x}, \bar{y})\mathrm{d}\phi \\ \frac{\mathrm{d}}{\mathrm{d}t}\bar{y} = \frac{1}{2\pi}\int_0^{2\pi} g(\mathrm{e}^{\mathrm{i}\phi}\bar{x}, \bar{y})\mathrm{d}\phi \end{cases}$$

The class of second order equations under consideration in this paper, i.e. (2), written as first order systems, do not fall into the category considered in [3]. However, it turns out that with some modifications to the schemes developed in [3], we can show that the modified HMM schemes approximate the averaged equations analyzed in [9]. This is the main purpose of our paper.

There has been much development of methods for special Hamiltonian systems $H(p,q) = \frac{1}{2}p^T M^{-1} p + W(q)$. These methods typically either assume an explicit separate grouping of solution components that change rapidly (fast modes) from the slow modes, or they assume that the potential W is the sum of a strong one and a weak one. Correspondingly, in the first case, slow and fast modes are solved separately in the whole interval $[t_n, t_n + H]$, and in the second case, a splitting approach is adopted to solve alternately the whole system with only the strong potential or the weak potential. They are called multirate methods and impulse method respectively. Please refer to [4], [6], [8], and more generally [7] for details. Even though these types of methods also use time averaging and appear to share certain resemblance to the HMM methods, it is important to point out that there is a fundamental difference. In the multirate or the impulse methods, the stiff part of the system, being either the fast modes or the split equations with strong potentials, is really solved globally in time, thus the high computational cost that results from the stiffness still remains. Whereas in the HMM methods, as we alluded earlier, the stiff system is solved only rarely for very short period of time. The macroscopic step size H is independent of ϵ and the overall number of operations is lower than ϵ^{-1}.

For a given $\epsilon > 0$, all well known methods will converge as the step-size $H \longrightarrow 0$, and there is no difference between stiff and nonstiff problems. We define what we mean by convergence such that it makes sense for very stiff problems ($\epsilon \ll H$) by the following error:

$$E = \max_n (\lim_{H \to 0} (\sup_{0<\epsilon<\epsilon_0(H)} |X(t_n) - X_n|)). \tag{6}$$

Here, $\epsilon_0(H)$ is a positive function of H, serving as an upper bound for the range of ϵ that we consider. With this notion, it is clear that a sensible multiscale method has to utilize the slow varying property of X and generate accurate approximation with a complexity sublinear to ϵ^{-1}.

The structure of the paper is as follows. In Sect. 2, we describe the HMM strategies of [1] in the context of building ODE schemes for problems with different time scales. In Sect. 3, we apply these types of schemes to compute average trajectories of an inverted pendulum. We then show in Sect. 4 the convergence of this scheme that is suggested by our numerical study. Finally, we summarize our results in Sect. 5.

2 HMM Strategy

Given a stiff system

$$\frac{\mathrm{d}}{\mathrm{d}t}u = f_\epsilon(u,t), \tag{7}$$

an HMM method integrates an effective system

$$\frac{\mathrm{d}}{\mathrm{d}t}U = \bar{f}(U),$$

whose force $\bar{f}$ is evaluated using many short time integrations of (7) with suitable initial data. So a generic HMM method is described by 1) the scheme used to integrate the (macro) system for U, 2) its accompanying scheme for the integration of (7), the microscopic system, and 3) the data transfer between the macro and micro systems. A microscopic evolution of the system is invoked *only when* the effective force at certain time, t_n, is needed by the macro-scheme. At that time, (7) is solved accurately on the corresponding micro-grid, with the initial condition determined from U, for a duration of time, η, to resolve the transient or the oscillations. The resulting microscale data, including the time history of microscale variables and the force, is then averaged by a suitable kernel K to evaluate the effective force and, in some cases, also a modified macroscopic variable U, at the appropriate time. We will use $\mathbb{K}^{p,q}$ to denote the kernel space discussed in this paper. $K \in \mathbb{K}^{p,q}(I)$ if $K \in C_c^q(\mathbb{R})$ with $\mathrm{supp}(K) = I$, and

$$\int_{\mathbb{R}} K(t)t^r \mathrm{d}t = \begin{cases} 1, \; r = 0; \\ 0, \; 1 \leq r \leq p. \end{cases}$$

Furthermore, we will use $K_\eta(t)$ to denote the scaling of K: $K_\eta(t) := \eta^{-1}K(t/\eta)$.

Hence we may present the above procedures algorithmically as follows:

1. Force estimation:
   a) Reconstruction: at $T = t_n$, $R(U_n) \mapsto u_n$.
   b) Solve for the micro variables: $u_n(t)$, for $t \in [t_n - \frac{\eta}{2}, t_n + \frac{\eta}{2}]$, with $u_n(t_n) = u_n$.
   c) Compression: $U_* = Q[u_n]$.
   d) Estimate force: $\bar{f}(t_*) = \tilde{F}[u_n] = K_\eta * u_n(t_*)$.
2. Evolve the macro variables: $\{U_n\} \bigcup \{U_*\} \longrightarrow U_{n+1}, T = t_{n+1}$.
3. Repeat

The reconstruction operator and the compression operator should satisfy a compatibility condition:

$$Q(R(U)) = U.$$

An essential feature in this paper is the introduction of a reconstruction operator R so that the average of the fast modes in u is preserved in each microscopic evolution, and correspondingly, the compression operator Q that prepares the macroscopic variable in the suitable form.

We also notice that the number of micro-time steps needed depends on the nature of the problem. For example, for stiff problems with fast transients, we only need to evolve the micro variables until the transients vanish; in molecular dynamics, e.g.

[2], the micro variables are evolved until equilibrium and then some further time to estimate the effective flux. We shall see that it also depends on the method used to estimate the effective force.

Figure 2 depicts schematics of the HMM ODE solvers. In these images, the top axis represents the macro grid used, and the bottom axis contains the microgrids established in a neighborhood of each macrogrid for microscale simulations. The arrow pointing from each macro grid point down to a micro grid denotes the action taken in step 1a, while the arrows pointing from each micro grid up toward the macro axis represent steps 1c and 1d. The effective force is estimated either at some new time t_* which is laid down to be a new macroscale grid point, or the original macroscale grid point t_n, depending on the macroscale scheme used (see Fig. 2). The advantage of schemes depicted in Fig. 2 is that $\bar{f}$ can be evaluated on a uniform grid, and thus facilitate the implementation of linear multistep methods on the macro-grid. In the upper image in Fig. 2, a non-symmetric kernel is needed to perform the force evaluation, and in the problems with transients, the macrogrid variables are projected to the invariant manifolds. The lower scheme in Fig. 2 can be applied to reversible systems that have no transients. The advantage is the possibility of using symmetric kernels in force estimation.

We will call a method HMMpq-X-y, if X-method is used in step 2, y-method is used in Step 1b, and a kernel $K \in \mathbb{K}^{p,q}$ is used in Step 1d. Most of the time, we will suppress the parameters pq. Therefore, HMM-FE-rk4 is a method that uses forward Euler for macroscale evolution, and a fourth order Runge-Kutta method for microscale integrator. In Sect. 3, we will present a few standard HMM schemes and their stability in more detail. The structure of the HMM-X-y schemes presented are illustrated by Fig. 2.

3 Main Example

In this section, we propose three HMM ODE schemes to solve for the slow periodic motion of the inverted pendulum. Recall the original equation of motion for the pendulum,

$$l\ddot{\theta} = \left(g + \frac{1}{\epsilon}\sin\left(2\pi\frac{t}{\epsilon}\right)\right)\sin\theta, \tag{8}$$

where θ is the angle between the pendulum and the upward vertical position, l is the length of the pendulum, and g is the gravitational constant. This takes the form of the model equation (2) where,

$$\frac{1}{\epsilon}a(x) = a_\epsilon(x) := \frac{1}{l}\left(g + \frac{1}{\epsilon}\sin(2\pi x)\right).$$

We will compare our approximations to the solution of the averaged equation,

$$l\ddot{\Theta} = g\sin\Theta - \frac{1}{8\pi^2 l}\sin\Theta\cos\Theta, \tag{9}$$

where $\Theta = \langle\theta\rangle$.

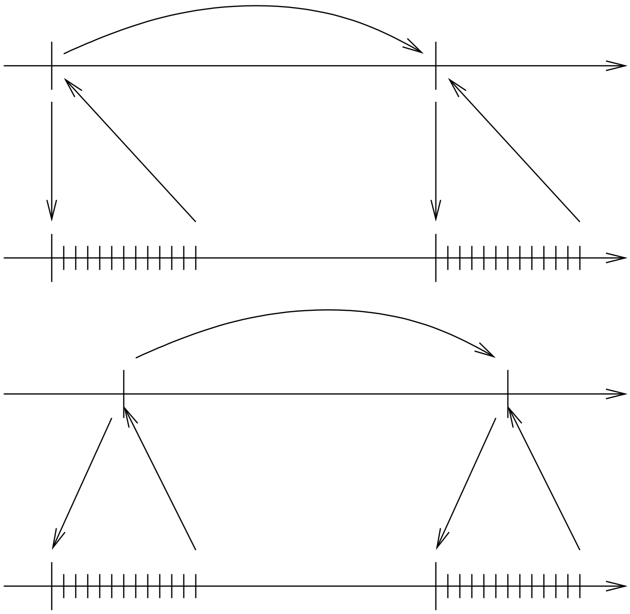


Fig. 2. Schematic pictures of the interaction between the macro (upper lines) and micro (lower lines) scale computational domains for use with non-symmetric (top) and symmetric (bottom) kernels

In this case, depending on the initial condition, the inverted pendulum can reach a maximum angle $\theta_{\max} = \cos^{-1}(gl/\langle v^2 \rangle)$, subject to the stability criterion. In the examples below $g = 0.1$ and $l = 0.05$ making $\theta_{\max} \approx 1.16$, although under the initial conditions $\theta_0 = 0, \quad \dot{\theta}_0 = -0.4$, the pendulum will sweep out a more conservative angle $\theta_\delta \approx 0.23$.

We use $u_n(t) = (\theta_n(t), \omega_n(t))$ to denote the solution of the first order system corresponding to equation (8), for $|t - t_n| \leq \eta/2$ with $u_n(t_n) = (\theta_n(t_n), \omega_n(t_n))$ given. The function $\theta_n(t)$ represents the angle, and $\omega_n(t) = \dot{\theta}_n(t)$ is the angular velocity. Similarly, $U(t) = (\Theta(t), \Omega(t))$ denotes the solution to the averaged equation (9) for $t \geq t_0$ with $U(t_0) = (\Theta(t_0), \Omega(t_0))$ given, and $\Omega(t) = \dot{\Theta}(t)$. Discretize the averaged equation with time-step size H, $t_n = t_0 + nH, \quad n = 1, 2, 3, \ldots$ and $U_n = U(t_n)$.

Assume that the HMM strategy described in the previous section does discretize the average equation (9) by solving (8) locally near every t_n. This assumption imposes a compatibility condition on the reconstruction step. We need to reconstruct $u_n^0 = (\theta_n^0, \omega_n^0) = R(\Theta_n, \Omega_n)$ such that $\langle \theta_n(t) \rangle \approx \Theta_n$ and $\langle \omega_n(t) \rangle \approx \Omega_n$ if $\theta_n(t_n) = \theta_n^0$ and $\omega_n(t_n) = \omega_n^0$. It is shown that $|\Theta(t) - \theta(t)| \sim \mathcal{O}(\epsilon)$ for $t \in (t_0, T]$ in [9], so taking $\theta_n^0 = \Theta_n$ insures that $\Theta_n \approx \langle \theta_n(t) \rangle$. Writing Ω_n as the force acting on Θ at t_n gives,

$$\Omega_n \approx \langle \omega_n(t) \rangle = \omega_n^0 + \left\langle \int_{t_n}^{t} a_\epsilon(\frac{s}{\epsilon}) \sin(\theta_n(s)) \mathrm{d}s \right\rangle .$$

Hence, we can set,

$$\begin{aligned} \omega_n^0 &= \Omega_n - \left\langle \int_{t_n}^{t} a_\epsilon(\frac{s}{\epsilon}) \sin(\theta_n(s)) \mathrm{d}s \right\rangle \\ &\approx \Omega_n - \sin(\Theta_n) \frac{\cos(2\pi \frac{t_n}{\epsilon})}{2\pi l}, \end{aligned}$$

(in the second step we note that $\sin(\theta(s))$ varies slowly where $s \in [t_n - \epsilon/2, t_n + \epsilon/2]$) to ensure $\Omega_n \approx \langle \omega_n(t) \rangle$.

The force estimator should yield an approximation to the force of the averaged equation,

$$\tilde{F}[u_n(\cdot)] = \begin{pmatrix} \tilde{F}_{(1)}[u_n(\cdot)] \\ \tilde{F}_{(2)}[u_n(\cdot)] \end{pmatrix} \approx \begin{pmatrix} \Omega_n \\ g \sin \Theta_n - \frac{1}{(8\pi^2)l} \sin \Theta_n \cos \Theta_n \end{pmatrix} . \tag{10}$$

Given a kernel $K \in \mathbb{K}^{p,q}$ as described in the previous section, let

$$\tilde{F}[u_n(\cdot)] = \begin{pmatrix} K * \omega_n(\cdot) \\ K * (g + \frac{1}{\epsilon} \sin(2\pi \frac{(\cdot)}{\epsilon})) \sin(\theta(\cdot)) \end{pmatrix} , \tag{11}$$

with initial conditions,

$$u_n^0 = \begin{pmatrix} \Theta_n \\ \Omega_n - \sin \Theta_n \frac{\cos(2\pi t_n / \epsilon)}{2\pi l} \end{pmatrix} .$$

The convolution $K * g(\cdot)$ is defined as

$$(K * f)(t) = \int_{t_n - \frac{\eta}{2}}^{t_n + \frac{\eta}{2}} K_\eta(t - s) g(s) \mathrm{d}s \ .$$

Using this estimated force, it will be possible to prove (in Sect. 4) that $\tilde{F}[u_n(\cdot)]$ approximates (10).

We present three HMM schemes and related numerical results for the inverted pendulum. The first order macroscopic Forward Euler schemes HMM-FE-* can be presented as follows:

Algorithm 1 *HMM-FE-**
Given $U_0 = (\Theta_0, \Omega_0)$, *for* $n = 0, 1, 2, ...$

$$\begin{aligned} \Theta_{n+1} &= \Theta_n + H \cdot \tilde{F}_{(1)}[\omega_n(\cdot)], \\ \Omega_{n+1} &= \Omega_n + H \cdot \tilde{F}_{(2)}[\theta_n(\cdot)], \end{aligned}$$

where $\tilde{F}[u_n(\cdot)]$ *is defined by (11).*

Provided that Ω_n is sufficiently accurate, one may directly replace $\tilde{F}_{(1)}[u_n(\cdot)]$ with Ω_n. This is done in practice to reduce computation, and in this case there is no explicit need to calculate ω_n.

Next, a semi-implicit first order HMM-IFE-* scheme is,

Algorithm 2 *HMM-IFE-**
Given $U_0 = (\Theta_0, \Omega_0)$, *for* $n = 0, 1, 2, \ldots$

$$\begin{aligned}\Omega_{n+1} &= \Omega_n + H \cdot \tilde{F}_{(2)}[\theta_n(\cdot)]\\ \Theta_{n+1} &= \Theta_n + H \cdot \Omega_{n+1}\end{aligned}$$

In this case Ω_{n+1} is found using the explicit forward Euler step and then used to find Θ_{n+1}.

The final algorithm is a second order HMM-Verlet-* scheme,

Algorithm 3 *HMM-Verlet-**
Given $U_n = (\Theta_n, \Omega_n)$, *for* $n = 0, 1, 2, \ldots$

$$\begin{aligned}\Omega_{n+\frac{1}{2}} &= \Omega_n + \frac{H}{2} \cdot \tilde{F}_{(2)}[\theta_n(\cdot)],\\ \Theta_{n+1} &= \Theta_n + H \cdot \Omega_{n+\frac{1}{2}},\\ \Omega_{n+1} &= \Omega_{n+\frac{1}{2}} + \frac{H}{2} \cdot \tilde{F}_{(2)}[\theta_{n+1}(\cdot)],\end{aligned}$$

where $\theta_{n+1} = \Theta_{n+1}$ *is used to initialize the final force estimation.*

This final method requires twice the computational effort per macroscale step as the first order schemes, but the total operation count is still much smaller than that of a direct calculation ($\eta/\epsilon \ll T/\epsilon$).

Several numerical simulations were completed using the parameters in Table 1. In each calculation, either the standard Verlet method (v) or fourth order Runge-Kutta (rk4) was used to solve the microscopic equations. In all cases, the exponential kernel,

Table 1. The parameters used for the numerical examples include ϵ, the initial condition (Θ_0, Ω_0), the time interval from t_0 to t_f, H (intervals indicate the range of values used for separate calculations to determine error behavior), h, and η (both fixed and scaled with respect to H; r, s, and q in row three are the orders of the macroscale and microscale schemes, and the smoothness of the kernel respectively, and α_h and α_η are constants), the exponential kernel K described in equation (12), the gravitational acceleration g, and the length of the pendulum arm l

$[t_0, T]$	H	η	h
$[0.0, 50.0]$	0.01	10ϵ	$\epsilon/10$
$[0.0, 12.0]$	$[0.001, 1.0]$	50ϵ	$\epsilon/50$
$[0.0, 12.0]$	$[0.001, 1.0]$	$\alpha_\eta H^{-s/q}\epsilon^{1-1/q}$	$\alpha_h H^{s/r}\epsilon^{1+2/r}\eta^{-1/r}$

$$K_\eta\left(\frac{t}{\eta}\right) = \frac{422.11}{\eta}\exp\left[5\left(\frac{4\left(t-t_n\right)^2}{\eta^2}-1\right)^{-1}\right], \tag{12}$$

is used ($K \in \mathbb{K}^{1,\infty}$) and $\epsilon = 10^{-6}$, $(\Theta_0, \Omega_0) = (0.0, -0.4)$, $g = 0.1$, $l = 0.05$.

Figure 3 shows the macroscale behavior of the system over the period from $t_0 = 0$ to $T = 50$. The parameters used in the calculations are in the first row of Table 1. The HMM schemes HMM-FE-v and HMM-IFE-v are compared to the solution of the averaged equation (4). At the microscale, the resolution gives about 10 grid points per oscillation, and the convolution with K is a domain containing about 10 cycles. This is relatively coarse compared to later calculations. The computational savings, measured by the total number of times the force is evaluated and compared to a traditional first order method using the same step-size h, is on the order of 10^8. More importantly, fully resolved traditional methods cannot maintain sufficient accuracy to carry the calculation to $T = 50$.

The force estimation error for HMM is $\mathcal{E}_{HMM} = \mathcal{E}_{\text{micro}} + \mathcal{E}_K + \mathcal{E}_{\text{quad}}$ [1, 3] and the local error of the HMM scheme is

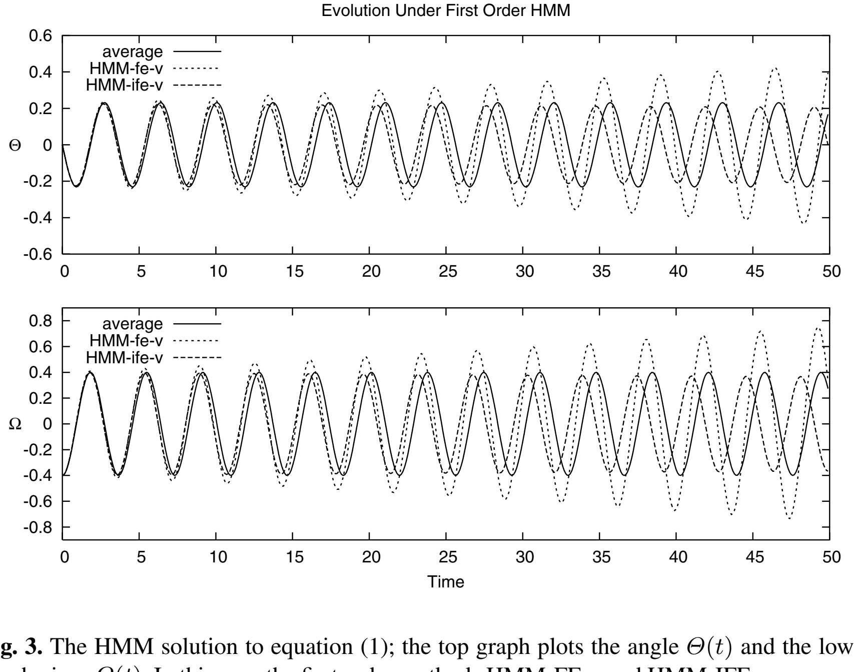


Fig. 3. The HMM solution to equation (1); the top graph plots the angle $\Theta(t)$ and the lower graph gives $\Omega(t)$. In this case the first order methods HMM-FE-v and HMM-IFE-v were used to approximate the average motion of the pendulum over a long time interval. The solution of the averaged equation (4) is also shown for comparison. The parameters used to produce the graph are those in the first row of Table 1. Six orders of magnitude ($\epsilon = 10^{-6}$) separate the period of the slow oscillation apparent in the graphs from the fast oscillation at the microscale

$$\mathcal{E}_n = \mathcal{E}_H + \mathcal{E}_{HMM}.$$

$\mathcal{E}_H$ denotes the local truncation error of the macroscopic scheme, and in many cases, $\mathcal{E}_H$ dominates and determines the order of $\mathcal{E}_n$. The convergence of various HMM schemes as $H \longrightarrow 0$ was confirmed using the error metric,

$$E = \max_n \sqrt{(\Theta_n - \Theta(t_n))^2 + (\Omega_n - \Omega(t_n))^2} \tag{13}$$

where $\Theta(t)$ and $\Omega(t)$ are values of the solution of the averaged equation. Error calculations were carried out over the time period $[0, 12]$, corresponding to roughly three oscillations on the macroscale. Figure 4 shows error as a function of $1/H$ for the first order methods HMM-FE-* and HMM-IFE-*. The calculations correspond to row two of Table 1. In all cases η and h are fixed with respect to H, and $O(H)$ convergence is achieved. In the HMM-IFE-v method however, approximation error associated with the microscale calculation and convolution dominates the contribution from $\mathcal{E}_H$, for small H.

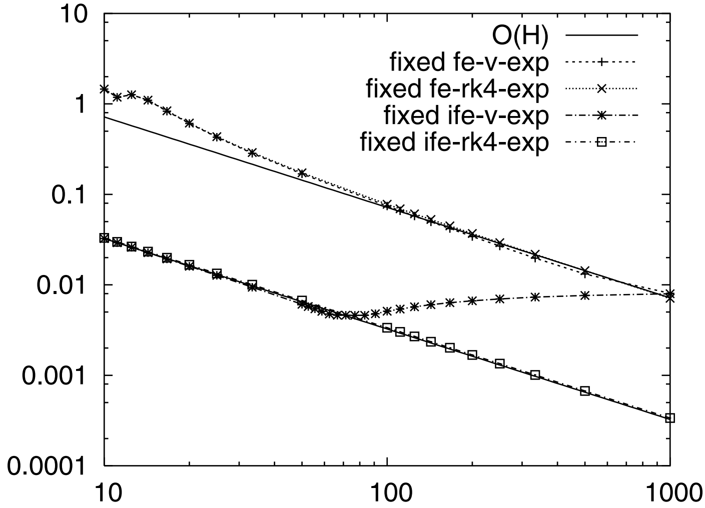


Fig. 4. The error as a function of $1/H$ for the first order schemes HMM-X-y, where X is FE or IFE and y is v or rk4. The width of the microscale domain and the step-size h are fixed with respect to H. The parameters used are listed in row two of Table 1. The slopes of the solid lines indicate decrease at first order in H

Figure 5 shows the analogous cases for the second order methods HMM-V-*. As previously, the HMM-X-v method levels off due to other contributions to the overall error, while HMM-X-rk4 is able to maintain its performance for small H.

Notice that some of the curves in Figs. 4 and 5 eventually flatten out as $1/H$ increases. These are the situations in which the error $\mathcal{E}_{HMM}$ finally dominates the

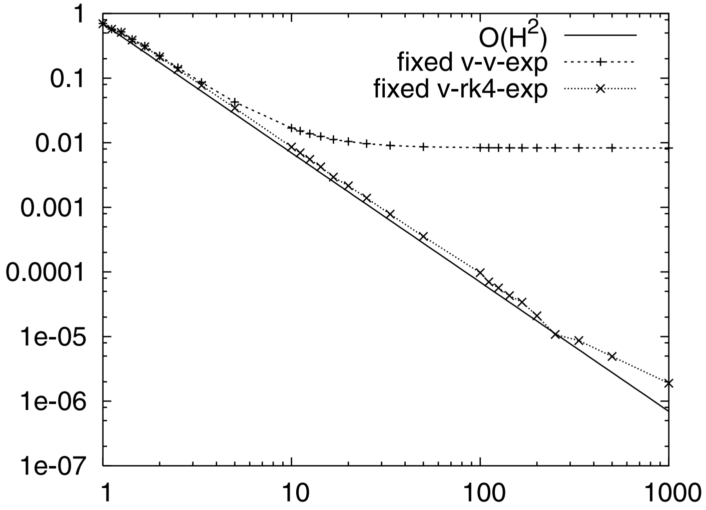


Fig. 5. The error as a function of $1/H$ for the second order schemes HMM-V-y, where y is v or rk4. The width of the microscale domain and the step-size h are fixed with respect to H. The parameters used are listed in row two of Table 1. The slope of the solid line indicates decrease at second order in H

global error of the computations. It is possible to overcome the flattening of the HMM-X-v cases by scaling the parameters η and h with H. See [3] for more detail. By setting $\eta = \alpha_\eta H^{-s/q}\epsilon^{1-1/q}$ and $h = \alpha_h H^{s/r}\epsilon^{1+2/r}\eta^{-1/r}$, where r, s, p, and q are the orders of the macroscale and microscale schemes, and the number of vanishing moments and smoothness of the kernel respectively, and α_h and α_η are constants, the HMM-V-v scheme is able to maintain second order behavior as shown in Fig. 6. The drawback to scaling is that the constants α_h and α_η need to be chosen carefully to place η and h in reasonable ranges.

HMM-V-rk4 maintains its second order performance as H decreases in the case of fixed η and h. This performance is matched when η and h are scaled as described above as shown in Fig. 7.

4 Generalizations

It is clear from the HMM structure that the stability of an HMM scheme requires that both the macroscopic and microscopic schemes be stable. However, one must also establish the consistency of the estimated force in (11). The notion of consistency can be defined as in (6). Consider the general case,

$$\ddot{x} = a_\epsilon\left(\frac{t}{\epsilon}\right) f(x), \qquad x(0) = x_0, \quad \dot{x}(0) = y_0 \tag{14}$$

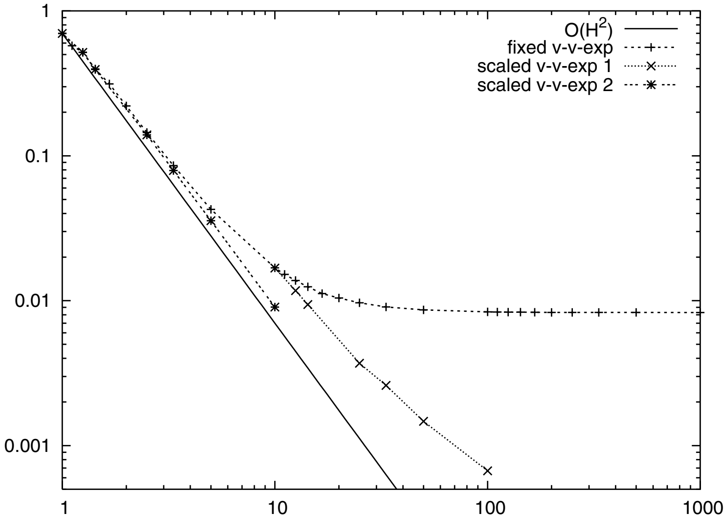


Fig. 6. A comparison of the error, as a function of $1/H$, for the second order schemes HMM-V-v for fixed and scaled microscale domains and step-sizes. The parameters and scaling used are listed in rows two and three of Table 1. The slope of the solid line indicates decrease at second order in H. In the cases where η and h are scaled with respect to H, the method is able to maintain second order performance, despite the leveling off seen in the fixed case. The constants α_h and α_η were chosen such that the scaled and fixed versions of the calculation would match at $H = 0.1$ and $H = 1$. The lines labeled "scaled v-v-exp 1" and "scaled v-v-exp 2" illustrate this recalibration. The constants were reset at these values of H so that the ratio of η to h would remain reasonable as H decreases

and the associated averaged equation,

$$\ddot{X} = \langle a_\epsilon \rangle f(X) - \langle v^2 \rangle f(X) f'(X) + C\sqrt{\epsilon}. \tag{15}$$

The basic assumption is that $a_\epsilon(t)$ is an ϵ-periodic smooth function satisfying $\langle a_\epsilon(t) \rangle \leq C$, and $f \in C^p$ with its derivatives uniformly bounded, i.e. $||f^{(k)}||_\infty < C_0$ for $k = 0, \cdots, p$. Note that $||a_\epsilon||_\infty$ can still be of $\mathcal{O}(\epsilon^{-1})$, even if $\langle a_\epsilon(t) \rangle \leq C$.

These assumptions hold for the inverted pendulum. In this case $a_\epsilon = a_\epsilon(t/\epsilon)$, $\langle a_\epsilon(t) \rangle = g/l$, $||x||_\infty \leq x_{\max} \sim \mathcal{O}(1)$, $f(x) \approx f(X)$, and $||\dot{x}||_\infty \leq E_y = \sqrt{2(E_0 - g)/l} \sim \mathcal{O}(1)$. The constants $x_{\max}$ and E_0 are determined by the given parameters and initial conditions. They may be calculated by considering the effective potential implied by (9), $V(x) = g\cos(x) + \sin^2 x/16\pi^2 l$, and the energy $E_0 = l y_0^2/2 + V(x_0)$.

Lemma 1. *The following results can be found in [3]*

1. *If $g \in C(\mathbb{R})$ is an α-periodic function with zero average, then for any $K \in \mathbb{K}^{p,q}$ and $\epsilon > 0$, there exists a positive constant $\hat{C}$ such that*

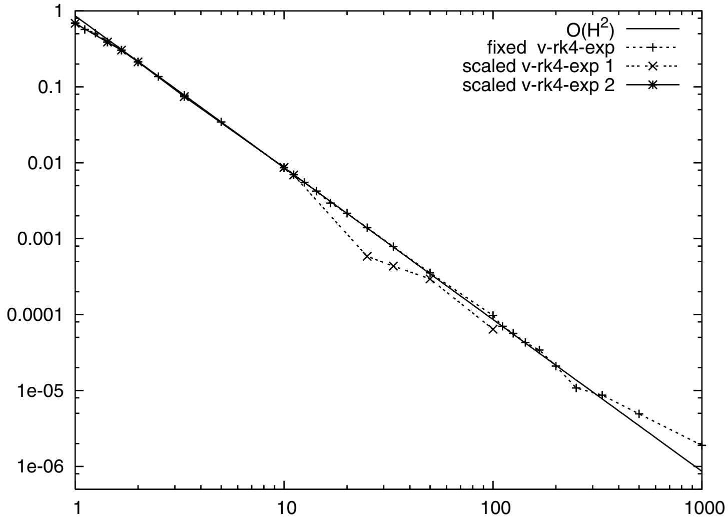


Fig. 7. A comparison of the error, as a function of $1/H$, for the second order schemes HMM-V-rk4 for fixed and scaled microscale domains and step-sizes. The parameters and scaling used are listed in rows two and three of Table 1. The slope of the solid line indicates decrease at second order in H. In the case where η and h are scaled with respect to H, the method is able to match the second order performance of the fixed method. The lines labeled "scaled v-rk4-exp 1" and "scaled v-rk4-exp 2" illustrate the recalibration of α_h and α_η as in Fig. 6

$$|K_\eta * g(\cdot/\epsilon)(t)| \leq \hat{C} \cdot \alpha^q \left(\frac{\epsilon}{\eta}\right)^q ||K||_{W^{1,q}}. \tag{16}$$

2. *Let $f_\epsilon(t) = f(t, t/\epsilon)$, where $f(t,s)$ is 1-periodic in the second variable and $\partial^r f(t,s)/\partial t^r$ is continuous for $r = 0, \ldots, p-1$, for any $K \in \mathbb{K}^{p,q}$. Then there exist constants C_1 and C_2, independent of ϵ and η, such that*

$$|K_\eta * f_\epsilon(t) - \bar{f}(t)| \leq C_1\eta^p + C_2 \left(\frac{\epsilon}{\eta}\right)^q. \tag{17}$$

In the discussion which follows we will make use of the velocity

$$v\left(\frac{s}{\epsilon}\right) = \int_{s_n}^{s} \left(a_\epsilon\left(\frac{\sigma}{\epsilon}\right) - \langle a_\epsilon\rangle\right) \mathrm{d}\sigma, \quad s \in \left[t_n - \frac{\eta}{2}, t_n + \frac{\eta}{2}\right]$$

where s_n is chosen so that $\langle v\rangle = 0$ and $|s_n - t_n| \leq \epsilon$. The velocity is related to $\dot{x}$ and likewise the scaling of a_ϵ implies that $||v||_\infty \sim \mathcal{O}(1)$. Another useful term will be $\Delta x = x(s) - X_n$, where $s \in [t_n - \eta/2, t_n + \eta/2]$. Provided that $||\dot{x}||_\infty$ is bounded as shown above, Δx is small.

Lemma 2. *(Δx is small) $\Delta x = x(s) - X_n$, for $s \in [t_n - \eta/2, t_n + \eta/2]$*

$$|\Delta x| \leq C_1\epsilon + C_2\eta$$

Proof. From [9] we have $|x(t_n) - X_n| \leq C\epsilon$.

$$\begin{aligned} |x(s) - X_n| &\leq |x(t_n) - X_n| + \left| \int_{t_n}^{s} \dot{x}(\tau)\mathrm{d}\tau \right| \\ &\leq C\epsilon + \frac{\eta}{2}||\dot{x}||_\infty . \end{aligned}$$

A more explicit description of $\dot{x}(s)$ will also be needed in addition to its boundedness.

Lemma 3. *(Expression for $\dot{x}(s)$)*

$$\dot{x}(s) = f(X_n)v\left(\frac{s}{\epsilon}\right) + \dot{x}(s_n) + f(X_n)\langle a\rangle(s - s_n) + \int_{s_n}^{s} a_\epsilon\left(\frac{\sigma}{\epsilon}\right) f'(z)\Delta x \mathrm{d}\sigma,$$

where $f'(z)$ is the remainder term of a Taylor expansion.

Proof. By definition

$$\begin{aligned} \dot{x}(s) &= \dot{x}(s_n) + \int_{s_n}^{s} a_\epsilon\left(\frac{\sigma}{\epsilon}\right) f(x(\sigma))\mathrm{d}\sigma \\ &= \dot{x}(s_n) + f(X_n)\langle a_\epsilon\rangle(s - s_n) + f(X_n)\int_{s_n}^{s}\left(a\left(\frac{\sigma}{\epsilon}\right) - \langle a\rangle\right)\mathrm{d}\sigma \\ &\quad + \int_{s_n}^{s} a\left(\frac{\sigma}{\epsilon}\right) f'(z)\Delta x\mathrm{d}\sigma \\ &= \dot{x}(s_n) + f(X_n)\langle a_\epsilon\rangle(s - s_n) + f(X_n)v\left(\frac{s}{\epsilon}\right) + \int_{s_n}^{s} a\left(\frac{\sigma}{\epsilon}\right) f'(z)\Delta x\mathrm{d}\sigma\ . \end{aligned}$$

In the previous section, the problem is recast as a first order system of equations, and the average force is estimated by using the microscale solution with suitable initial data. Setting $y = \dot{x}$ and $Y_n = \langle y\rangle$, the force

$$\begin{pmatrix} \tilde{F}_{(1)} \\ \tilde{F}_{(2)} \end{pmatrix} = \begin{pmatrix} K * y \\ K * \ddot{x} \end{pmatrix},$$

given the initial data

$$\begin{pmatrix} x_n \\ y_n \end{pmatrix} = \begin{pmatrix} X_n \\ Y_n - f(X_n)\langle\int_{t_n}^{s} a_\epsilon\left(\frac{\sigma}{\epsilon}\right)\mathrm{d}\sigma\rangle \end{pmatrix},$$

accurately estimates the average force of (15). The accuracy of the force estimator $\tilde{F}_{(1)}$ may be quickly shown given the initial value y_n above.

Theorem 1. *(Consistency of $\tilde{F}_{(1)}$) Given $\tilde{F}_{(1)} = K * y$, $K \in \mathbb{K}^{p,q}$,*

$$|\tilde{F}_{(1)} - Y(t_n)| \leq C_1\frac{\eta^2}{\epsilon} + C_2\left(\frac{\epsilon}{\eta}\right)^q$$

Proof. By definition,

$$\begin{aligned}
K * y &= \int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K_\eta(t_n-s)y(s)\mathrm{d}s \\
&= \int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K_\eta(t_n-s)\left[y_n + f(X_n)\int_{t_n}^{s} a_\epsilon\left(\frac{\sigma}{\epsilon}\right)\mathrm{d}\sigma\right. \\
&\quad \left. + \int_{t_n}^{s} a_\epsilon\left(\frac{\sigma}{\epsilon}\right) f'(z)\Delta x \mathrm{d}\sigma\right]\mathrm{d}s \\
&= Y_n \int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K_\eta(t_n-s)\mathrm{d}s + f(X_n)\int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K_\eta(t_n-s)\left[\int_{t_n}^{s} a_\epsilon\left(\frac{\sigma}{\epsilon}\right)\mathrm{d}\sigma\right. \\
&\quad \left. - \left\langle \int_{t_n}^{s} a_\epsilon\left(\frac{\sigma}{\epsilon}\right)\mathrm{d}\sigma\right\rangle + \int_{t_n}^{s} a_\epsilon\left(\frac{\sigma}{\epsilon}\right) f'(z)\Delta x \mathrm{d}\sigma\right]\mathrm{d}s \\
&= Y_n + \mathcal{O}\left(\left(\frac{\epsilon}{\eta}\right)^q, \frac{\eta}{\epsilon}||\Delta x||_\infty\right),
\end{aligned}$$

where $f'(z)$ is the remainder term from the expansion of f. Using the last part of Lemma 1, the second term in the third line above reduces to $\mathcal{O}((\frac{\epsilon}{\eta})^q)$ but is dominated by a simple error estimate of the last term of line three. The final result is reached by recalling Lemma 2.

We now prove our main result, that the force estimator $\tilde{F}_{(2)}$ provides a good approximation of the averaged force $\ddot{X}$.

Theorem 2. *(Consistency of $\tilde{F}_{(2)}$) Given $\tilde{F}_{(2)} = K * g$, $K \in \mathbb{K}^{p,q}$, and $g(t) = a_\epsilon(t/\epsilon)f(x(t))$, then*

$$\left|\tilde{F}_{(2)} - \ddot{X}(t_n)\right| \leq \mathcal{O}\left(\frac{\eta^2}{\epsilon}, \frac{\epsilon}{\eta}, \frac{\epsilon^{q-1}}{\eta^q}, \eta, \sqrt{\epsilon}\right)$$

Proof. Let

$$G(t) = \langle a_\epsilon\rangle f(X) - \langle v^2\rangle f(X)f'(X). \tag{18}$$

We show that $K * g \sim G$.

$$\begin{aligned}
K * g &= \int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K_\eta(t_n-s)a_\epsilon\left(\frac{s}{\epsilon}\right) f(x(s))\mathrm{d}s \\
&= f(X_n)\int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K_\eta(t_n-s)a_\epsilon\left(\frac{s}{\epsilon}\right)\mathrm{d}s \\
&\quad + f'(X_n)\int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K_\eta(t_n-s)a_\epsilon\left(\frac{s}{\epsilon}\right)\Delta x\mathrm{d}s \\
&\quad + \int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K_\eta(t_n-s)a_\epsilon\left(\frac{s}{\epsilon}\right) f''(z)\frac{\Delta x^2}{2}\mathrm{d}s \\
&= I_1 + I_2 + I_3,
\end{aligned}$$

where $f''(z)$ is the remainder term from the expansion of f.

Consider I_3,

$$\begin{aligned}|I_3| &\leq \frac{1}{2}||a_\epsilon||_\infty \cdot ||f''||_\infty \cdot ||\Delta x||_\infty^2 \int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} |K_\eta(t_n-s)|\mathrm{d}s \\ &\sim \mathcal{O}(||\Delta x||_\infty^2/\epsilon).\end{aligned}$$

Using Lemma 1, we may estimate I_1,

$$\begin{aligned}I_1 &= \langle a_\epsilon\rangle f(X_n) + f(X_n)\int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K(t_n-s)\left(a_\epsilon\left(\frac{s}{\epsilon}\right) - \langle a_\epsilon\rangle\right)\mathrm{d}s \\ &\leq \langle a_\epsilon\rangle f(X_n) + ||f||_\infty \cdot ||a_\epsilon||_\infty \left(\frac{\epsilon}{\eta}\right)^q \\ &\leq \langle a_\epsilon\rangle f(X_n) + C\left(\frac{\epsilon^{q-1}}{\eta^q}\right)\end{aligned}$$

The estimate of I_2 relies again on Lemma 1

$$\begin{aligned}I_2 &= \langle a_\epsilon\rangle f'(X_n)\int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K_\eta(t_n-s)\Delta x \mathrm{d}s \\ &\quad + f'(X_n)\int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K_\eta(t_n-s)\left(a_\epsilon\left(\frac{s}{\epsilon}\right) - \langle a_\epsilon\rangle\right)\Delta x\mathrm{d}s \\ &= \langle a_\epsilon\rangle f'(X_n)\int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K_\eta(t_n-s)\Delta x\mathrm{d}s \\ &\quad + f'(X_n)\int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K'_\eta(t_n-s)v\left(\frac{s}{\epsilon}\right)\Delta x\mathrm{d}s \\ &\quad - f'(X_n)\int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K_\eta(t_n-s)v\left(\frac{s}{\epsilon}\right)\dot{x}(s)\mathrm{d}s\end{aligned}$$

where we have integrated by parts and $v(s/\epsilon) = \int_{s_0}^{s}(a_\epsilon(\sigma/\epsilon) - \langle a_\epsilon\rangle)\mathrm{d}\sigma$, and $\langle v\rangle = 0$.

Using Lemma 3 to replace $\dot{x}(s)$, the last term in the expression above for I_2 yields the final part of the average force,

$$
\begin{aligned}
-f'(X_n)\int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K_\eta'(t_n-s)v\left(\frac{s}{\epsilon}\right)\dot{x}(s)\mathrm{d}s \quad &= \\
-f'(X_n)\dot{x}(s_n)\int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K_\eta(t_n-s)v\left(\frac{s}{\epsilon}\right)\mathrm{d}s \\
-f'(X_n)f(X_n)\langle a_\epsilon\rangle\int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K_\eta(t_n-s)v\left(\frac{s}{\epsilon}\right)(s-s_n)\mathrm{d}s \\
-f'(X_n)f(X_n)\int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K_\eta(t_n-s)v^2\left(\frac{s}{\epsilon}\right)\mathrm{d}s \\
-f'(X_n)\int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K_\eta(t_n-s)v\left(\frac{s}{\epsilon}\right)\int_{s_n}^{s} a_\epsilon\left(\frac{\sigma}{\epsilon}\right)f'(z)\varDelta x\mathrm{d}\sigma\mathrm{d}s) \\
= -f'(X_n)f(X_n)\langle v^2\rangle + \mathcal{O}\left(\frac{\epsilon^{q-1}}{\eta^q},\frac{\eta^2}{\epsilon}\right)
\end{aligned}
$$

All that remains is to show that the terms of I_2 that are not part of G are small.

$$
\begin{aligned}
I_2 = -f'(X_n)f(X_n)\langle v^2\rangle + f'(X_n)\langle a_\epsilon\rangle\int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K_\eta(t_n-s)\varDelta x\mathrm{d}s \\
+f'(X_n)\int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K_\eta'(t_n-s)v\left(\frac{s}{\epsilon}\right)\varDelta x\mathrm{d}s + \mathcal{O}\left(\frac{\eta^2}{\epsilon},\frac{\epsilon^{q-1}}{\eta^q}\right)
\end{aligned}
$$

In this expression, the first term is the term of interest and appropriate estimates will reduce the remaining terms to some small order. Most of these terms are reduced by a direct application of Hölder's inequality or the Lemma 1, but the estimate of the final term, involving $K_\eta'(t_n-s)$, is more involved. First define,

$$
\Im = \int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K_\eta'(t_n-s)v\left(\frac{s}{\epsilon}\right)\varDelta x(s)\mathrm{d}s
$$

By expanding $\varDelta x(s)$ and applying Lemma 1, $|\Im|$ becomes,

$$
\begin{aligned}
|\Im| \le &\left|\int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K_\eta'(t_n-s)v\left(\frac{s}{\epsilon}\right)x(s)\mathrm{d}s\right| \\
&+ |\langle X_n\rangle|\cdot\left|\int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K_\eta'(t_n-s)v\left(\frac{s}{\epsilon}\right)\mathrm{d}s\right| \\
&\le |\Im_1| + |\langle X_n\rangle|\cdot C_1\left(\frac{\epsilon^{q-1}}{\eta^q}\right)
\end{aligned}
$$

Integrate (2) twice and replace $x(s)$ in $\Im_1$ to give,

$$\begin{aligned}|\Im_1| \leq & \left|\int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K'_\eta(t_n-s)v\left(\frac{s}{\epsilon}\right)\int_{t_n}^{s}\int_{t_n}^{\sigma}\frac{1}{\epsilon}a\left(\frac{\gamma}{\epsilon}\right)\tilde{f}(\gamma)\mathrm{d}\gamma\mathrm{d}\sigma\mathrm{d}s\right| + \\ & \left|\int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K'_\eta(t_n-s)v\left(\frac{s}{\epsilon}\right)(C_2 s + C_3)\,\mathrm{d}s\right| \\ \leq & |\Im_2| + C_4\left(\frac{\epsilon^{q-1}}{\eta^q}\right),\end{aligned}$$

again calling on Lemma 1 in the final step. Observe that $\frac{1}{\epsilon}a\left(\frac{s}{\epsilon}\right) = \langle a_\epsilon\rangle + \frac{1}{\epsilon}b\left(\frac{s}{\epsilon}\right)$ where $\|b\|_\infty \sim \mathcal{O}(1)$, $b(s+1) = b(s)$, and $\langle b\rangle = 0$, and define,

$$a^{[1]}(s) \equiv \textstyle\int_{t_n}^{s} \frac{1}{\epsilon}a\left(\frac{\sigma}{\epsilon}\right)\mathrm{d}\sigma = \langle a_\epsilon\rangle(s-t_n) + \beta(s)$$

where $\beta(s+\epsilon) = \beta(s)$ and $\|\beta\|_\infty \leq \|b\|_\infty$. It will be important to note that $\|a^{[1]}\|_\infty \sim \mathcal{O}(1)$. With these definitions (and again Lemma 1) estimate $|\Im_2|$,

$$\begin{aligned}|\Im_2| \leq & \left|\int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K'_\eta(t_n-s)v\left(\frac{s}{\epsilon}\right)\int_{t_n}^{s}\int_{t_n}^{\sigma} a^{[1]}(\gamma)\dot{\tilde{f}}(\gamma)\mathrm{d}\gamma\mathrm{d}\sigma\mathrm{d}s\right| + \\ & \left|\int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K'_\eta(t_n-s)v\left(\frac{s}{\epsilon}\right)\int_{t_n}^{s} a^{[1]}(\sigma)\tilde{f}(\sigma)\mathrm{d}\sigma\mathrm{d}s\right| + \\ & \left|a^{[1]}(t_n)\tilde{f}(t_n)\right| \cdot \left|\int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K'_\eta(t_n-s)v\left(\frac{s}{\epsilon}\right)(s-t_n)\mathrm{d}s\right| \\ \leq & |\Im_3| + |\Im_4| + \left(\|a^{[1]}\|_\infty \cdot \|\tilde{f}\|_\infty\right) C_5\left(\frac{\epsilon^{q-1}}{\eta^q}\right)\end{aligned}$$

$\Im_3$ may be quickly reduced,

$$|\Im_3| \leq \left(\|\dot{K}\|_\infty \cdot \|v\|_\infty \cdot \|a^{[1]}\|_\infty \cdot \|\dot{f}\|_\infty \cdot \|\dot{x}\|_\infty\right)\eta$$

In similar fashion to the treatment of $a^{[1]}$, break $\beta(s)$ into a constant part equal to $\langle\beta\rangle$ and an oscillating part with mean 0. Then

$$a^{[2]}(s) = \textstyle\int_{t_n}^{s} a^{[1]}(\sigma)\mathrm{d}\sigma = \dfrac{\langle a_\epsilon\rangle(s-t_n)^2}{2} + \langle\beta\rangle(s-t_n) + \epsilon\Lambda(s)$$

where $\Lambda(s+\epsilon) = \Lambda(s)$ and $\|\Lambda\|_\infty \sim \mathcal{O}(1)$. After expanding $\tilde{f}(s)$ about t_n,

$$\begin{aligned}
|\Im_4| &= \left| \int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K'_\eta(t_n-s) v\left(\frac{s}{\epsilon}\right) \int_{t_n}^{s} a^{[1]}(\sigma)\left(\tilde{f}(t_n) + \dot{\tilde{f}}(z(\sigma))\Delta x(\sigma)\right) \mathrm{d}\sigma \mathrm{d}s \right| \\
&\leq \left|\tilde{f}(t_n)\right| \cdot \left| \int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K'_\eta(t_n-s) v\left(\frac{s}{\epsilon}\right) a^{[2]}(s) \mathrm{d}s \right| + |\Im_5| \\
&\leq \|\tilde{f}\|_\infty \cdot \left| \int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K'_\eta(t_n-s) v\left(\frac{s}{\epsilon}\right) \left(\frac{\langle a_\epsilon\rangle (s-t_n)^2}{2} + \langle \beta \rangle (s-t_n)\right) \mathrm{d}s \right| \\
&\quad + \|\tilde{f}\|_\infty \cdot \left| \int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K'_\eta(t_n-s) v\left(\frac{s}{\epsilon}\right) \Lambda(s) \mathrm{d}s \right| + |\Im_5| \\
&\leq \left(\|\tilde{f}\|_\infty\right) C_6 \left(\frac{\epsilon^{q-1}}{\eta^q}\right) + \left(\|\tilde{f}\|_\infty \cdot \|\dot{K}\|_\infty \cdot \|v\|_\infty \cdot \|\Lambda\|_\infty\right) \frac{\epsilon}{\eta} + |\Im_5|
\end{aligned}$$

where Lemma 1 is used in the last step. Finally, a straightforward estimate shows,

$$\begin{aligned}
|\Im_5| &= \left| \int_{t_n-\frac{\eta}{2}}^{t_n+\frac{\eta}{2}} K'_\eta(t_n-s) v\left(\frac{s}{\epsilon}\right) \int_{t_n}^{s} a^{[1]}(\sigma) \tilde{f}(z(\sigma)) \Delta x(\sigma) \mathrm{d}\sigma \mathrm{d}s \right| \\
&\leq \left(\|\dot{K}\|_\infty \cdot \|v\|_\infty \cdot \|a^{[1]}\|_\infty \cdot \|\tilde{f}\|_\infty\right) \|\Delta x\|_\infty
\end{aligned}$$

Combining all of the estimates and Lemma 2,

$$|\Im| \sim \mathcal{O}\left(\frac{\epsilon^{q-1}}{\eta^q}, \frac{\epsilon}{\eta}, \eta\right)$$

and

$$I_2 = -f'(X_n) f(X_n) \langle v^2 \rangle + \mathcal{O}\left(\frac{\epsilon^{q-1}}{\eta^q}, \eta\right).$$

Combining the results for I_1, I_2, and I_3 yields the complete estimate,

$$\begin{aligned}
|K * g - \ddot{X}(t_n)| &= |K * g - G + C\sqrt{\epsilon}| \\
&\sim \mathcal{O}\left(\frac{\eta^2}{\epsilon}, \frac{\epsilon}{\eta}, \frac{\epsilon^{q-1}}{\eta^q}, \eta, \sqrt{\epsilon}\right),
\end{aligned}$$

where G is defined in (18).

5 Conclusion

The inverted pendulum exhibits stable slow oscillation due to rapid microscale oscillatory forcing. This macroscale behavior is captured very well by a set of HMM algorithms for which the computational complexity is much lower than that of standard numerical methods. The HMM approach requires only $\mathcal{O}(T/H \cdot \eta/\epsilon)$ operations, which lead to a computational savings of $\mathcal{O}(H/\eta)$ or about 10^3 for the parameters in our numerical experiments compared to standard numerical methods. Notably, standard methods lack sufficient accuracy to solve the model problem here with $\epsilon = 10^{-6}$ for macroscopic time scales.

Acknowledgment

The second author is partially supported by NSF Grant No. DMS-0111298.

References

1. Weinan E and Bjorn Engquist. The heterogeneous multi-scale methods. *Comm. Math. Sci.*, 1(1):87–133, 2003.
2. Xiantao Li and Weinan E. Multiscale modeling of the dynamics of solids at finite temperature. Submitted to *J. Mech. Phys. of Solids*, 2004.
3. Bjorn Engquist and Yen-Hsi Tsai. Heterogeneous multiscale methods for stiff ordinary differential equations. 2003. To appear, *Math. Comp.*
4. B. García-Archilla, J. M. Sanz-Serna, and R. D. Skeel. Long-time-step methods for oscillatory differential equations. *SIAM J. Sci. Comput.*, 20(3):930–963 (electronic), 1999.
5. C. W. Gear and K. A. Gallivan. Automatic methods for highly oscillatory ordinary differential equations. In *Numerical analysis (Dundee, 1981)*, volume 912 of *Lecture Notes in Math.*, pages 115–124. Springer, Berlin, 1982.
6. C. W. Gear and D. R. Wells. Multirate linear multistep methods. *BIT*, 24(4):484–502, 1984.
7. Ernst Hairer, Christian Lubich, and Gerhard Wanner. *Geometric numerical integration*, volume 31 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 2002. Structure-preserving algorithms for ordinary differential equations.
8. Ben Leimkuhler and Sebastian Reich. A reversible averaging integrator for multiple timescale dynamics. *J. Comput. Phys.*, 171(1):95–114, 2001.
9. Mark Levi. Geometry and physics of averaging with applications. *Phys. D*, 132(1-2):150–164, 1999.
10. Linda R. Petzold. An efficient numerical method for highly oscillatory ordinary differential equations. *SIAM J. Numer. Anal.*, 18(3):455–479, 1981.

Multiscale Homogenization of the Navier–Stokes Equation

Nils Svanstedt[1] and Niklas Wellander[2]

[1] Department of Mathematics, Chalmers University of Technology and Göteborg University, SE-412 96 Göteborg, Sweden
nilss@math.chalmers.se

[2] Swedish Defence Research Agency, FOI, SE-581 11 Linköping, Sweden
niklas@foi.se

Summary. The incompressible Navier–Stokes equation is studied. By using multiscale expansion methods we obtain local and homogenized Navier–Stokes equations. We then derive a homogenization based eddy viscosity model.

Key words: multiscale expansion, Navier–Stokes equation, homogenization

1 Introduction

The starting point in our study is the incompressible Navier–Stokes equation. In a recent work [5] we prove the existence of a two-scale limit of the incompressible stationary Navier–Stokes equation.

In the present work we introduce a slow time scale (of order $\sqrt{\varepsilon t}$) where t is the normal time scale. The slow time scale allows us to derive the global behaviour, (i.e., the slow time–large scale asymptotics) of the flow. This will be referred to as the homogenized Navier–Stokes equation.

A motivation for the two-scale asymptotics is the separation of scales in cellular flow as Rayleigh–Bénard convection [3] and in flow in porous media [11]. The classical derivation of Darcy's law by Tartar in the appendix of Sanchez-Palencia [14] has been a starting point for various further developments, see e.g. the survey article [11] by Mikelic or our recent paper [5]. In order to adequately model more complex flows we first need to understand the simplest convective flow, Rayleigh–Bénard convection, which is driven by unstable buoyancy forces caused by a large enough temperature difference between two parallel plates. In fact fluid motion driven by thermal convection is a common and important phenomenon in nature. It is the major feature of the dynamics of the oceans, atmosphere and the interior of stars and planets. Convection is also an important phenomena in numerous industrial processes. In [3] and [4] the Rayleigh–Bénard convection problem is studied in the Boussinesq approximation and local equations are derived for the coupled Navier–Stokes and heat equations. We also prove existence and uniqueness in the large aspect ratio regime (small boyancy forcing). We also prove existence of strong attractors. It is well known that a turbulent flow exhibits an extensive range of temporal and spatial scales, see [6]. The transfer of energy between the motion on the different scales is complex. In recent years it has been modeled by multiple scales techniques. In the present study we present a simple multiple scales homogenization for the derivation of the local and homogenized effective equations for the incompressible Navier–Stokes equations in a periodic porous medium. The advantage of such a derivation is that it gives simple equations which describe the coupling between the dynamics at different scales. Up to the authors knowledge the slow-time large-scale Navier–Stokes equation (6) has not been in the literature before. For rigorous results concerning the convergence process in the derivation of the asymptotic homogenization limits we refer to [5] where a general compensated compactness theorem is proved in the context of two-scale convergence. We use the homogenized Navier–Stokes equation to define a subgrid tensor and construct a Large Eddy Simulation model based on this tensor. An advantage by such a framework is that it takes into account the contribution from finer scales to the subgrid model. For numerical simulations using this approach we refer to [7] and [8]. The paper is organized as follows: In Sect. 2 we expand the velocity field and the pressure of a scaled Navier–Stokes system in a power series with multiple scales in the terms. We use standard multiple scales homogenization techniques to derive the system describing the local behaviour and the large scale (homogenized) system, respectively. In Sect. 3 we propose a Large Eddy Simulation model based on the homogenized Navier–Stokes system derived in Sect. 2. In Sect. 4 we present a homogenization result for the 3D stationary Navier–Stokes system in an application to flow in periodic porous media. We here review the results from [5] and refer to [5] for a full exposition with proofs of the statements. The proof in Sect. 4 is based on a new compensated compactness result (Theorem 5) and a new div-curl lemma (Corollary 3) which are stated in the Appendix (Sect. 5) together with the basic results of Nguetseng's two-scale convergence. Also here we refer to [5] for a full exposition. Concerning notations and exact meaning of function spaces for smooth oscillating test functions we refer to [13].

2 Scaling and Expansions

We start with the incompressible Navier–Stokes equation

$$\begin{cases} \dfrac{\partial u}{\partial s} + (u \cdot \nabla)u - \nu \Delta u + \nabla p = f, \\ \operatorname{div} u = 0 \end{cases} \quad x \in \Omega,\ s \in (0, S). \tag{1}$$

A multiple scales analysis together with the a priori estimates gives us the small spatial scaling $y = x/\varepsilon$. Here ε is the typical pore size in porous media or the size of a cell in cellular flow. The multiple scales analysis also motivates a renormalization of time $t := \sqrt{\varepsilon} s$. We introduce the fast temporal scaling $\tau := t/\varepsilon$ associated to the renormalized time. This gives us a scaled Navier–Stokes equation:

$$\begin{cases} \varepsilon^{1/2}\dfrac{\partial u_\varepsilon}{\partial t} + (u_\varepsilon \cdot \nabla)u_\varepsilon - \varepsilon^{3/2}\nu \Delta u_\varepsilon + \nabla p_\varepsilon = f_\varepsilon, \\ \operatorname{div} u_\varepsilon = 0. \end{cases} \quad x \in \Omega,\ t \in (0,T). \tag{2}$$

We equip (2) with the initial data $u_\varepsilon(x,0) = u_\varepsilon^0(x)$ which is assumed to be bounded in $L^2(\Omega)$. We will throughout the paper assume that the forcing is of the form

$$f_\varepsilon(x,t) = f_0\left(x, t, x/\varepsilon, \frac{t}{\varepsilon}\right) + \varepsilon f_1\left(x, t, x/\varepsilon, \frac{t}{\varepsilon}\right) + \ldots,$$

where $f_\varepsilon \in L^2(0,T; L^2(\Omega; \mathbb{R}^n))$ and where $f_i,\ i = 0,\ 1, ..,$ are regular enough to permit the above scaling. A typical situation is the case of periodic cellular flow like in Rayleigh–Bénard convection see [3].

In this section we expand the functions $u_\varepsilon = u_\varepsilon(x,t)$ and $p_\varepsilon = p_\varepsilon(x,t)$ in multiple scales power series in order to find the leading order systems in terms of small scale and large scale spatial variables and fast time and slow time variables, respectively. Our goal is to find a scaling which preserves the structure of the original system, at least to the leading order approximation.

We assume in a standard fashion that the functions $u_\varepsilon = u_\varepsilon(x,t)$ and $p_\varepsilon = p_\varepsilon(x,t)$ admit multiple scales expansions on the forms

$$u_\varepsilon(x,t) = \varepsilon^{1/2}\sum_{i=0}^{\infty} \varepsilon^i u_i\left(x, t, \frac{x}{\varepsilon}, \frac{t}{\varepsilon}\right), \tag{3}$$

$$p_\varepsilon(x,t) = \sum_{i=0}^{\infty} \varepsilon^i p_i\left(x, t, \frac{x}{\varepsilon}, \frac{t}{\varepsilon}\right), \tag{4}$$

where the u_i's and the p_i's are all assumed to be Y-periodic with respect to $y \in \mathbb{R}^n$, $n = 2,\ 3$ and 1-periodic in the fast time variable τ. For simplicity we assume that Y is the unit cube in $\mathbb{R}^n$. If we formally put $y = x/\varepsilon$ and $\tau = t/\varepsilon$, the chain rule transforms the differential operators as

$$\frac{\partial}{\partial t} \mapsto \frac{\partial}{\partial t} + \frac{1}{\varepsilon}\frac{\partial}{\partial \tau}, \quad \frac{\partial}{\partial x} \mapsto \frac{\partial}{\partial x} + \frac{1}{\varepsilon}\frac{\partial}{\partial y}.$$

The divergence, gradient, curl and Laplace operators transform accordingly and we denote differentiation with repect to x and y by subscript x and y, respectively. In a standard way we now insert the series (3)-(4) into the system (2). By employing the chain rule we can list a hierarchy of equations in increasing orders of powers of ε. For the first equation in (2) we obtain:

$$\varepsilon^0 \ : \ \frac{\partial u_0}{\partial \tau} + (u_0 \cdot \nabla_y)u_0 - \nu \Delta_{yy} u_0 + \nabla_y p_1 = f_0 - \nabla_x p_0;$$

$$\varepsilon^1 \ : \ \frac{\partial u_0}{\partial t} + \frac{\partial u_1}{\partial \tau} + (u_0 \cdot \nabla_x)u_0 + (u_1 \cdot \nabla_y)u_0 + (u_0 \cdot \nabla_y)u_1$$

$$- \nu \Delta_{yy} u_1 - \nu \mathrm{div_x} \nabla_y u_0 - \nu \mathrm{div_y} \nabla_x u_0 + \nabla_y p_2 = f_1 - \nabla_x p_1.$$

The second equation in (2) yields:

$$\left(\mathrm{div}_x + \frac{1}{\varepsilon}\mathrm{div}_y\right)(\varepsilon^{1/2}u_0 + \varepsilon^{3/2}u_1 + ...) = 0,$$

i.e.,

$$\varepsilon^{-1/2} : \mathrm{div}_y\, u_0 = 0;$$

$$\varepsilon^{1/2} : \mathrm{div}_x\, u_0 + \mathrm{div}_y\, u_1 = 0.$$

The fast time - small scale system corresponding to (2) is given by:

$$\begin{cases} \dfrac{\partial u_0}{\partial \tau} + (u_0 \cdot \nabla_y)u_0 - \nu \varDelta_{yy} u_0 + \nabla_y p_1 = f_0 - \nabla_x p_0 \\ \mathrm{div_y} u_0 = 0. \end{cases} \tag{5}$$

The existence of the stationary version of the system (5) is rigorously derived by the help of a general two-scale compensated compactness theorem recently proved in [5]. In Sect. 4 we review the result for the readers convenience.

The corresponding slow time - large scale system derived above reads:

$$\begin{cases} \dfrac{\partial u_0}{\partial t} + \dfrac{\partial u_1}{\partial \tau} + (u_0 \cdot \nabla_x)u_0 + (u_1 \cdot \nabla_y)u_0 + (u_0 \cdot \nabla_y)u_1 \\ -\nu \varDelta_{yy} u_1 - \nu \mathrm{div_x} \nabla_y u_0 - \nu \mathrm{div_y} \nabla_x u_0 + \nabla_y p_2 = f_1 - \nabla_x p_1, \\ \mathrm{div}_x\, u_0 + \mathrm{div}_y\, u_1 = 0. \end{cases}$$

An averaging (denoted overbar) of the fast time and small scale, i.e., over $(0, 1)$ in τ and Y in y gives the (homogenized system), i.e., the slow time - large scale asymptotics:

$$\begin{cases} \dfrac{\partial \overline{u_0}}{\partial t} + \overline{(u_0 \cdot \nabla_x)u_0} = \overline{f_1} - \nabla_x \overline{p_1}, \\ \mathrm{div_x} \overline{u_0} = 0. \end{cases} \tag{6}$$

Here we have used the integration by parts formula. The local inertial terms (involving ∇_y) vanish by local periodicity and incompressibility, respectively, c.f. [3] pp. 164–165.

3 Reynolds Stress Tensor and Eddy Viscosity

Let us consider again the homogenized system (6):

$$\begin{cases} \dfrac{\partial \overline{u_0}}{\partial t} + \overline{(u_0 \cdot \nabla_x)u_0} = \overline{f_1} - \nabla_x \overline{p_1}, \\ \mathrm{div_x} \overline{u_0} = 0. \end{cases}$$

Let us also consider the system

$$\begin{cases} \dfrac{\partial \overline{u_0}}{\partial t} + (\overline{u_0} \cdot \nabla_x)\overline{u_0} + F(\overline{u_0}) = \overline{f_1} - \nabla_x \overline{p_1}, \\ \mathrm{div_x} \overline{u_0} = 0. \end{cases}$$

If we write the function $F(\overline{u_0}) = \mathrm{div}_x \sigma(\overline{u_0})$ we can define the Reynolds stress tensor

$$\sigma(\overline{u_0}) = \overline{u_0 \otimes u_0} - \overline{u_0} \otimes \overline{u_0}.$$

The procedure of averaging the Navier–Stokes equations over a certain scale is referred to as *Large Eddy Simulation* LES. See e.g. [7] or [8], where a homogenization based LES method is developed and compared numerically to traditional LES models. Other related recent subgrid models for incompressible multiscale flow can be found in [10] and [9], where a dynamic subgrid model based on self similarity and a multiresolution Haar-base wavelet analysis is developed. The main difficulty is how to model F in terms of $\overline{u_0}$. The simplest and most commonly used subgrid models are so called *eddy viscosity* models, where the effect of the Reynolds stress tensor is modeled as an extra viscosity. In conventional eddy viscosity based LES models the eddy viscosity has been considered too dissipative. For the recent homogenization based LES models the eddy viscosity comes from a rigorous derivation of the equations describing the microstructure of the flow and is therefore balanced to the global viscosity, see [7] and [8] or the book [6] by Frisch. In linear homogenization the separation of scales allows a decoupling of the form

$$U(x, y) = u_0(x) + \sum_i \chi_i(y) \frac{\partial u_0}{\partial x_i}(x)$$

where χ_i solves a classical cell problem, where $\overline{F} = 0$:

$$\begin{cases} -\Delta_{yy}\chi_i = F \ \text{ in } \ Y, \\ \chi_i \in H^1_{\mathrm{per}}(Y). \end{cases}$$

In the case of separated scales as in Rayleigh–Bénard convection or jet engine flow this motivates a simple homogenization based eddy viscosity model where one assumes that near the mean field the flow can be approximated as:

$$u_0(x, t, y, \tau) = \overline{u_0}(x, t) + \nabla\overline{u_0}(x, t) : \xi(y, \tau),$$

where $\xi = \xi_{ij}$ is a 3×3-matrix function in the local variables with $\overline{\xi_{ij}} = 0$. For rotating jet engine flow the eddy viscosity tensor is explicitly derived in [2] and for the Navier–Stokes equation the details can be found in [7] and [8] under this assumption. These calculations will not be repeated here but the consequence of this approximation is that we can write the global equation as:

$$\begin{cases} \dfrac{\partial \overline{u_0}}{\partial t} + (\overline{u_0} \cdot \nabla_x)\overline{u_0} - \mathrm{div}_x(\mathcal{A} : \nabla_x)\overline{u_0} = \overline{f_1} - \nabla_x \overline{p_1}, \\ \mathrm{div_x} \overline{u_0} = 0. \end{cases}$$

where $\mathcal{A}$ is the eddy viscosity tensor. In practice $\mathcal{A}$ is computed approximately for a given local random forcing. In [7] the forcing in the local Navier–Stokes equation is chosen to be a Wiener process where the Fourier components are chosen in such a way that the Kolmogorov inertial range $5/3$-power-law scaling is satisfied.

4 Stationary Flow in Porous Media, Homogenization of the Navier–Stokes Equations

In this section we consider a periodic porous medium in three dimensions, for simplicity, let us assume the domain being a periodic repetition of $]0,L[^3=\Omega$. We set periodic boundary conditions on the outer boundary $\partial\Omega$ which will somewhat simplify the a priori estimates. Following [11] we define the unit cell $Y=]0,1[^3$, and let Y_S, the solid part, be a closed subset of $\bar{Y}$ and $Y_F=Y\neg Y_S$ be the fluid part. Further we make a periodic repetition of Y_S all over $\mathbb{R}^3$ and set $Y_S^k=Y_S+k$, $k\in\mathbb{Z}^3$. The set $E_S=\bigcup_{k\in\mathbb{Z}^3}Y_S^k$ is a closed subset of $\mathbb{R}^3$ and $E_F=\mathbb{R}^3\neg E_S$ is an open set in $\mathbb{R}^3$. We will assume that Y_F is an open connected set of strictly positive measure, with a Lipschitz boundary, Y_S has a strictly positive measure in $\bar{Y}$, E_F and the interior of E_S are open sets with the boundary of class $C^{0,1}$, which are locally located on one side of their boundary and that E_F is connected. Let Ω be covered with a regular mesh of size ε, each cell being an εY-cube, Y_i^ε, $1\leq i\leq N(\varepsilon)$. Each Y_i^ε is homeomorphic to Y, by linear homeomorphism Π_i^ε.

Define $Y_{S_i}^\varepsilon=(\Pi_i^\varepsilon)^{-1}(Y_S)$ and $Y_{F_i}^\varepsilon=(\Pi_i^\varepsilon)^{-1}(Y_F)$. For sufficiently small $\varepsilon>0$ we consider the set $T_\varepsilon=\{k\in\mathbb{Z}^3|Y_{S_k}^\varepsilon\in\Omega\}$ and define $O_\varepsilon=\bigcup_{k\in T_\varepsilon}Y_{S_k}^\varepsilon$, $S^\varepsilon=\partial O_\varepsilon$ and $\Omega_\varepsilon=\Omega\neg O_\varepsilon=\Omega\bigcap\varepsilon E_F$. We have $\partial\Omega_\varepsilon=\partial\Omega\bigcup S^\varepsilon$. The domains O_ε and Ω_ε represents the solid and fluid parts of the porous medium Ω. Obviously there is a characteristic length L and a microscopic length l. The ratio between these length scales yields a small parameter $\varepsilon=l/L$, which is assumed to be an even integer.

The fluid flow in a porous medium with the appropriate scaling can be modelled by the following incompressible stationary Navier–Stokes equations:

$$\begin{cases}(u^\varepsilon\cdot\nabla)\,u^\varepsilon-\varepsilon^{3/2}\nu\triangle u^\varepsilon+\nabla p^\varepsilon=f^\varepsilon\\ \operatorname{div}u^\varepsilon=0\\ u^\varepsilon|_{\partial\Omega_\varepsilon}=0,\end{cases}\tag{7}$$

almost everywhere in Ω_ε. Here u^ε is the velocity vector field, p^ε is the pressure and ν is the kinematic viscosity of the fluid. We assume the forcing f^ε two-scale converges weakly to f_0 in $L^2(\Omega\times Y;\mathbb{R}^3)$, $\int_Y f_0(x,y)\,\mathrm{d}y=0$. The fluid-solid boundary is denoted by $\partial\Omega_\varepsilon$. The scaling exponent $3/2$ for the viscosity is carried out in [3] for cellular flows and it is the critical exponent which preserves the Navier–Stokes structure on a local scale also in the porous media case. This scaling is previously used in e.g. [14] and Mikelic [11].

The existence of weak (Leray) solutions of (7) can be found in e.g. Mikelic [11]. We note that the solutions, u^ε, only exist in-between the solid part of Ω, i.e., in $\Omega_\varepsilon\subset\Omega$. When ε is decreasing it corresponds to smaller and smaller cells ($\varepsilon-Y$-cells), with smaller and smaller channels.

There is a vast literature on fluid flow in porous media. Here we refer to the survey article [11] by Mikelic where both the existence of solution to (7) and homogenization problem for (7) are considered. Indeed [11] is a nice introduction to filtration and fluid flow in porous media.

4.1 A Priori Estimates

To prove a priori estimates we need the following improved Poincaré inequality, see e.g. Tartar's proof in the appendix of [14].

Lemma 1. *Let $u_i^\varepsilon \in H^1(\Omega_\varepsilon)$ and assume that we have a non-slip boundary condition somewhere in the unit cube Y, then*

$$\int_{\Omega_\varepsilon} |u^\varepsilon(x)|^2 \mathrm{d}x \leq \varepsilon^2 C \int_{\Omega_\varepsilon} |\nabla u^\varepsilon(x)|^2 \mathrm{d}x.$$

The basic estimate is the following.

Lemma 2. *Assume $f^\varepsilon \in L^2(\Omega_\varepsilon;\mathbb{R}^3)$, with a uniform bound and let u^ε and p^ε be solutions of (7). Then*

$$\varepsilon^{-1/2}\|u^\varepsilon\|_{L^2(\Omega_\varepsilon;\mathbb{R}^3)} \leq C,$$

and

$$\varepsilon^{1/2}\|\nabla u^\varepsilon\|_{L^2(\Omega_\varepsilon;\mathbb{R}^{3^2})} \leq C.$$

Proof. See [5].

We can extend smoothly the solutions, u^ε, by zero to the solid part, such that the a priori estimates become valid on the constant domain Ω. Hence

Lemma 3.

$$\varepsilon^{-1/2}\|u^\varepsilon\|_{L^2(\Omega;\mathbb{R}^3)} \leq C,$$
$$\varepsilon^{1/2}\|\nabla u^\varepsilon\|_{L^2(\Omega;\mathbb{R}^{3^2})} \leq C.$$

By Schwarz inequality we also have the following estimate for the convective term:

Corollary 1.

$$\|(u^\varepsilon \cdot \nabla)u^\varepsilon\|_{L^1(\Omega;\mathbb{R}^3)} \leq C,$$

Proof. See [5].

For the estimate of the remaining terms we need the following function spaces

$$H = \{u \in L^2(\Omega;\mathbb{R}^3), \nabla \cdot u = 0, u^\varepsilon|_{\partial\Omega} = 0\},$$

Let P be the projection onto the orthogonal complement of H, denoted by $H^\perp$.

Corollary 2.

$$\|\nabla p^\varepsilon - \varepsilon^{3/2} P \Delta u^\varepsilon\|_{L^1(\Omega;\mathbb{R}^3)} \leq C.$$

Proof. See [5].

4.2 Homogenization

We recognize two spatial scales, the global x and the local (pore level) of order $\varepsilon^{-1}x$. We introduce a local spatial scale $y = \varepsilon^{-1}x$. In order to capture oscillations in the flow by two-scale convergence this means that we use test functions $\phi^{\varepsilon}(x) = \phi\left(x, \frac{x}{\varepsilon}\right)$, where $\phi \in C_0^{\infty}(\Omega; C^{\infty}(Y))$. We do not have a priori estimates to be able to identify the global equations but we can prove convergence for the local equations.

Theorem 1. *Let u^{ε} and p^{ε} be solutions of (7). Then, $\varepsilon^{-1/2}u^{\varepsilon}$ two-scale converges weakly to u^0 and ∇p^{ε} two-scale converges in the distributional sense to $\nabla_y p^1$, the solutions of the local stationary Navier–Stokes equation,*

$$\begin{cases} \left(u^0 \cdot \nabla_y\right) u^0 - \nu \bigtriangleup_{yy} u^0 = -\nabla_y p^1 + f_0 \\ \operatorname{div}_y u^0 = 0 \\ u^0(x, \cdot)|_{\partial Y_s} = 0, \end{cases} \tag{8}$$

where ∂Y_s is the fluid-solid boundary of Y.

Proof. The proof uses Corollary 3 below. For a proof, see [5].

Remark 1. Of course, the solutions of (8) may be more regular than the spaces where the convergence takes place. With more regular initial data and forcing one can also prove compensated compactness for the time-dependent Navier–Stokes equation. The corresponding rigorous convergence analysis for the homogenized problem (6) is a more delicate problem which is open.

5 Appendix: Two-Scale Compensated Compactness

Below we review some classical and new results concering compacness of oscillation sequences. In 1989 G. Nguetseng came up with a new concept of weak convergence (two-scale convergence) using periodically oscillating test functions (period ε) with vanishing period as ε tends to zero.

Definition 1. *A sequence $\{u^{\varepsilon}\}$ in $L^2(\Omega)$ is said to two-scale converge weakly to a function $u_0 = u_0(x, y)$ in $L^2(\Omega \times Y)$ if*

$$\lim_{\varepsilon \to 0} \int_{\Omega} u^{\varepsilon}(x)\varphi\left(x, \frac{x}{\varepsilon}\right) \mathrm{d}x = \int_{\Omega} \int_{Y} u_0(x, y)\varphi(x, y) \, \mathrm{d}y\mathrm{d}x,$$

for all test functions $\varphi \in L^2(\Omega; C^{\infty}(Y))$.

The basic compactness result with respect to two-scale convergence is due to Nguetseng [13] and reads

Theorem 2. *For every bounded sequence $\{u^{\varepsilon}\}$ in $L^2(\Omega)$ there exist a subsequence and a function u_0 in $L^2(\Omega \times Y)$ such that u^{ε} two-scale converges weakly to u_0.*

A fundamental result for applications to homogenization problems is the characterization of two-scale limits of functions in $H^1(\Omega)$ and their gradients is, see [13]:

Theorem 3. *Assume that* $\{u^\varepsilon\}$ *is a bounded sequence in* $H^1(\Omega)$. *Then*

$$\lim_{\varepsilon\to 0}\int_\Omega u^\varepsilon(x)\varphi\left(x,\frac{x}{\varepsilon}\right)\mathrm{d}x=\int_\Omega\int_Y u(x)\varphi(x,y)\,\mathrm{d}y\mathrm{d}x,$$

for all $\varphi\in L^2(\Omega;C^\infty(Y))$ *and*

$$\lim_{\varepsilon\to 0}\int_\Omega \nabla u^\varepsilon(x)\cdot\psi\left(x,\frac{x}{\varepsilon}\right)\mathrm{d}x=\int_\Omega\int_Y (\nabla_x u(x)+\nabla_y u_1(x,y))\cdot\psi(x,y)\,\mathrm{d}y\mathrm{d}x,$$

for all $\psi\in H^1(\Omega;C^\infty(Y;\mathbb{R}^n))$, *where* u *is the weak* $L^2(\Omega)$*-limit and* $u_1=u_1(x,y)\in L^2(\Omega;H^1(Y))$.

By the Rellich theorem, u is of course the strong L^2-limit of $\{u^\varepsilon\}$.

Remark 2. In [1] it is proven that the splitting of the gradient as in Theorem 3 is true in the sense of Radon measures if u^ε belongs to $BV(\Omega)$, which is the space of $L^1(\Omega)$ functions whose distributional gradients are in an $\mathbb{R}^n$-valued Radon measure with bounded total variation in Ω. We refer to [1] for the details. An observation is that if $u^\varepsilon\in L^1(\Omega)$ and the gradients belongs to $L^1(\Omega;\mathbb{R}^n)$ then u^ε belongs to $BV(\Omega)$.

A decade earlier F. Murat and L. Tartar developed a theory (compensated compactness) which opened the way to study the asymptotic behaviour of products of weakly compact sequences. They proved the div-curl lemma:

Lemma 4. *Suppose that* $\{u^\varepsilon\}$ *and* $\{v^\varepsilon\}$ *are two uniformly bounded sequences in* $L^2(\Omega;\mathbb{R}^n)$, *with weak limits* u *and* v, *respectively. Suppose further that* $\{\operatorname{div}u^\varepsilon\}$ *is a compact set of* $H^{-1}(\Omega)$ *and* $\{\operatorname{curl}v^\varepsilon\}$ *is a compact set of* $H^{-1}(\Omega;\mathbb{R}^n)$, *respectively. Then, for every test function* $\varphi\in C_0^\infty(\Omega)$, *we have*

$$\int_\Omega u^\varepsilon(x)\cdot v^\varepsilon(x)\varphi(x)\,\mathrm{d}x\to\int_\Omega u(x)\cdot v(x)\varphi(x)\,\mathrm{d}x.$$

They also proved a more general Compensated-Compactness Theorem from which the div-curl lemma follows as a special case.

Theorem 4 (Compensated compactness). *Let* Q *be a quadratic form on* $\mathbb{R}^p$. *If*

(i) $\{u^\varepsilon\}$ *converges weakly to* $u\in L^2(\Omega;\mathbb{R}^p)$

(ii) $\{\sum_{j=1}^p\sum_{l=1}^n a_{ijl}\frac{\partial u_j^\varepsilon}{\partial x_l}\}$ *belongs to a compact set of* $H^{-1}_{\mathrm{loc}}(\Omega)$, *for* $i=1,\ldots,q$,

then

$$Q(u^\varepsilon)\rightharpoonup l_0\quad\in L^1(\Omega)\ \textit{(or in the sense of measures)},$$

and the following holds true:

(i) *If* $Q(\lambda)\geq 0$ *for all* $\lambda\in\Lambda$ *then,* $l_0(x)\geq Q\left(u(x)\right)$.

(ii) *If* $Q(\lambda) = 0$ *for all* $\lambda \in \Lambda$ *then* $l_0(x) = Q\left(u(x)\right).$

Here the characteristic set Λ is defined by

$$\Lambda = \left\{ \lambda \in \mathbb{R}^p : \sum_{j=1}^{p}\sum_{l=1}^{n} a_{ijl}\lambda_j\xi_l = 0 \text{ for some } \xi \in \mathbb{R}^n \neg \{0\} \right\}.$$

For a proof of Lemma 4, Theorem 4 and related results we refer to [12] and [15] and the references therein.

A result of compensated compactness type for the method of two-scale convergence has been missing in the theory which has put restrictions on the applicability of two-scale convergence. For instance the identification of the two-scale limit of the nonlinear inertial term in the Navier–Stokes equation has been hard, see [11].

This was a strong motivation for the study in [5]. In this paper the results of Murat and Tartar are extended to the context of two-scale convergence. We prove:

Theorem 5 (Two-scale compensated compactness). *Let* $\{\varepsilon\}$ *be a sequence of positive numbers which tends to zero such that* $1/\varepsilon$ *is an even integer, let the characteristic set* Λ *be defined by*

$$\Lambda = \left\{ \lambda \in \mathbb{R}^p : \sum_{j=1}^{p}\sum_{l=1}^{n} a_{ijl}\lambda_j\xi_l = 0 \textit{ for some } \xi \in \mathbb{R}^n \neg \{0\} \right\}$$

and let Q *be a quadratic form on* $\mathbb{R}^p$. *If*

(i) $\{u^\varepsilon\}$ *two-scale converges weakly to* $u_0 \in L^2(\Omega \times Y; \mathbb{R}^p)$

(ii) $\{\sum_{j=1}^{p}\sum_{l=1}^{n} a_{ijl}\frac{\partial u_j^\varepsilon}{\partial x_l}\}$ *is bounded in* $L^2(\Omega)$ *for* $i = 1, \ldots, q$,

then

$$Q(u^\varepsilon) \textit{ two-scale converges weakly to } l_0 \in L^1(\Omega \times Y)$$

(or in the sense of measures) and the following holds true:

(i) *If* $Q(\lambda) \geq 0$ *for all* $\lambda \in \Lambda$ *then,* $l_0 \geq Q\left(u_0\right).$

(ii) *If* $Q(\lambda) = 0$ *for all* $\lambda \in \Lambda$ *then* $l_0 = Q\left(u_0\right).$

As in the usual compensated compactness setting we get as a corollary a two-scale div-curl lemma. This result is used in the proof of the homogenization of the Navier–Stokes system (8) in Sect. 4.

Corollary 3 (Two-scale div-curl lemma). *Let* $\{\varepsilon\}$ *be a sequence of positive numbers which tends to zero such that* $1/\varepsilon$ *is an even integer and assume that*

(i) $\{u^\varepsilon, v^\varepsilon\}$ *two-scale converges weakly to* $\{u_0, v_0\} \in L^2(\Omega \times Y; \mathbb{R}^6)$

(ii) $\{\mathit{div}\, u^\varepsilon, \mathit{curl}\, v^\varepsilon\}$ *is bounded in* $L^2(\Omega; \mathbb{R}^4)$.

Then

$$\lim_{\varepsilon\to 0}\int_{\Omega} u^{\varepsilon}(x)\phi\left(x,\frac{x}{\varepsilon}\right)\cdot v^{\varepsilon}(x)\phi\left(x,\frac{x}{\varepsilon}\right)\mathrm{d}x =$$

$$\int_{\Omega}\int_{Y} u_0(x,y)\phi(x,y)\cdot v_0(x,y)\phi(x,y)\,\mathrm{d}y\mathrm{d}x,$$

for all $\phi \in C_0^{\infty}(\Omega; C^{\infty}(Y))$.

References

1. M. Amar, *Two-scale convergence and homogenization on* $BV(\Omega)$, Asymptotic Anal. Vol 16 (1998), 65-84.
2. B. Birnir, S. Hou and N. Wellander, *A homogenization of the Navier–Stokes equation obtaining the viscous Moore–Greitzer equation for aero-engine flow*, submitted.
3. B. Birnir and N. Svanstedt, *Existence and Homogenization of the Rayleigh–Bénard problem*, J. Nonlinear Mathemathical Physics, 7, No. 2, (2000), 1-34.
4. B. Birnir and N. Svanstedt, *Existence theory and strong attractors for the Rayleigh–Bénard problem with a large aspect ratio*, Discrete and Continuous Dynamical Systems, Vol. 10, No. 1-2, (2004), 53-74
5. B. Birnir, N. Svanstedt and N. Wellander, *Two-scale compensated compactness*, manuscript, 2004.
6. U. Frisch, *Turbulence - The Legacy of A. N. Kolmogorov*, Cambridge University Press, 1995.
7. C. Fureby, L. Persson and N. Svanstedt, *On homogenization based methods for large eddy simulation*, J. Fluids Engineering, 124, (2002), 892-904.
8. C. Fureby, N. Alin, N. Svanstedt, S. Menon and L. Persson. *On large eddy simulation of high Reynolds number wall bounded flows*, AIAA Journal, Vol. 42, No 3, (2004), 457-467.
9. J. Hoffman, *Adaptive finite element methods for turbulent flow*, Preprint 2002-08, Chalmers Finite Element Center, Chalmers, Göteborg, 2002.
10. J. Hoffman and C. Johnson, *Adaptive multiscale computational modeling of complex incompressible fluid flow*, Preprint 2002-09, Chalmers Finite Element Center, Chalmers, Göteborg, 2002
11. A. Mikelic, *Homogenization Theory and Applications to Filtration Through Porous Media*, in Filtration in Porous Media and Industrial Applications, Lecture Notes in Mathematics 1734, Springer-Verlag, (2000), 127-214.
12. F. Murat, *Compacite par compensation*, Ann. Scuola Norm. Sup. Pisa Cl. Sci., 5, (1978), 489-507.
13. G. Nguetseng, *A general Convergence Result For a Functional Related to the Theory of Homogenization,* SIAM J. Math. Anal., Vol. 20, No.3, (1989), 608-623.
14. E. Sanchez-Palencia, *Non-Homogeneous media and Vibration Theory*, Springer Lecture Notes in Physics 127, Springer-Verlag Berlin, 1980.
15. L. Tartar, *Compensated compactness and applications to partial differential equations*, In: Nonlinear Analysis and Mechanics, Heriott-Watt Symposium, Vol. 4, Research Notes in Mathematics 39, Pitman Publ., London, (1979), 136-212.

Numerical Simulations of the Dynamics of Fiber Suspensions

Anna-Karin Tornberg

Courant Institute of Mathematical Sciences, New York University, 251 Mercer Street, New York, NY 10012-1185, USA
tornberg@cims.nyu.edu

Summary. The dynamics of flexible fibers or filaments immersed in a fluid are important to understanding many interesting problems arising in biology, engineering, and physics. For most applications, the flows are at very low Reynolds numbers, and the fibers can have aspect ratios of length to radius from a few tens to several thousands.

This class of problems is difficult to solve accurately to a reasonable cost with grid based methods, partly due to the different scales in length and radius of the fibers and the fact that elastic equations must be solved within the fibers.

Making explicit use both of the fact that we are considering Stokes flow, as well as of the slenderness of the fibers, we have designed a cost-effective method to simulate multiple interacting elastic fibers in a three dimensional Stokes flow. The key points are that for Stokes flow, boundary integral methods can be employed to reduce the three-dimensional dynamics to the dynamics of the two-dimensional fiber surfaces, and that using slender body asymptotics, this can be further reduced to the dynamics of the one-dimensional fiber center-lines. The resulting integral equations include both the effect of the fibers on the flow field, as well as the interactions of fibers, as mediated by the flow.

We have developed a numerical method based on this theory that allows for simulating multiple interacting highly flexible fibers. Considering the efficiency of the method, another important fact is that the framework is suitable for introducing a semi-implicit time-stepping scheme, eliminating the severe constraint on the time-step size arising from the elasticity. Our numerical approach is based on second-order divided differences for spatial derivatives, combined with special product integration methods that reflect the nearly singular nature of the integral operators.

Key words: fluid-structure interaction, boundary integral method, slender body approximation, flexible fibers

1 Background and Introduction

Flows in nature and engineering often acquire their interesting aspects by the presence in and interaction of the fluid with immersed elastic objects. Fish, tree leaves, flagella, and rigid polymers all come to mind. A very important special case is when the elastic bodies are microscopic and filamentary. For example, flexible fibers make up the micro-structure of suspensions that show strongly non-Newtonian bulk behavior, such as elasticity, shear-thinning, and normal stresses in shear flow [4, 9]. Moreover, micro-organisms utilize for locomotion the anisotropic drag properties of their long flexible flagella [2]. The dynamics of flexible filaments are also relevant to understanding soft materials. Liquid crystal phase transitions for example lead to the study of "soft" growing filaments in a smectic-A phase, [12, 16]. In all these problems, the filaments have large aspect ratios (length over radius), ranging from order ten to a thousand for natural to synthetic fibers, and up to many thousands in biological settings.

In the listed examples, the flows are at very low Reynolds numbers, for which the fluid dynamics is described by the Stokes equations. The Stokes equations are linear, and time enters only as a parameter, thus leading to its celebrated reversibility. However, this reversibility is broken by surface forces such as those induced by bending rigidity, and simple forcing flows can lead to very nontrivial dynamics.

Consider a plane shear flow. A rigid straight fiber placed in this flow will rotate and translate with the fluid. However, the dynamics can be very different when the fiber is flexible. As the strength of the shear flow increases relative to the bending rigidity of the fiber, there is a sharp bifurcation beyond which the fiber is unstable to buckling [1, 19] and small shape perturbations can grow into substantial bending of the filament. This stores elastic energy in the fiber which can later be released back to the system as the fiber is extended. This is related to the anomalous stresses that elastic fluids can develop, such as normal stress differences that push apart bounding walls in linear shear experiments [9]. The first normal stress difference is zero in the absence of the fiber, and is zero in temporal mean for a rigid fiber. For a fiber that bends, the symmetry of the first normal stress difference that holds for a straight fiber is broken, and the integrated normal stress difference now yields a positive net contribution [1, 19].

Thus, the dynamics show a surprising richness even for a single fiber, and for suspensions there is much that is still not well understood. It is worth noting that while experiments capture such sharp changes in fluidic response, continuum theories generally do not.

There are multiple scales present in this problem. First, at the level of individual fibers, the radius is much smaller than the length. Further, considering suspensions with a large number of fibers, the macroscopic dimensions of the suspension are much larger than the microscopic dimensions of the fibers. Ultimately, one would like to have a macroscopic model for such suspensions, and eliminate the need of computer simulations to resolve the micro-structure. However, a greater understanding of these flows is needed in order to develop such models, and this requires both experiments and numerical simulations.The main challenge for a numerical method lies in its ability to include many fibers in the simulation, at a reasonable cost, while maintaining accuracy.

Given the scales of the problem – many fibers, slenderness, complicated individual dynamics – several approximate methods have been developed. One such class is the so-called *bead-models*, in which a flexible fiber is modeled as a chain of linked rigid bodies, such as spherical beads [7, 21], elliptical solids [15] or cylinders [11, 18]. The number of building blocks in each fiber is typically moderate, with the dynamics based upon moment and force balances between them. In general, the nonlocal effects, induced by fluid incompressibility, of a fiber upon itself, or upon other fibers in the flow, are neglected [7, 11, 15, 18, 21].

The *immersed boundary method* [13] has also been applied to this class of problems. In this method, an elastic boundary is discretized with connected Lagrangian markers, and its relative displacements by fluid motion are used to calculate the boundary's elastic responses. These elastic forces are then distributed onto a background grid covering the computational domain, and used as forces acting upon the fluid, thereby modifying the surrounding fluid flow. For example, Stockie [17] used an immersed boundary method (at moderate Reynolds number) to simulate a single "filament" (modeled as an infinitesimally thin elastic boundary) buckling in a two-dimensional linear shear-flow. To add a physical width to the fiber, a fiber structure must be constructed from a bundle of intertwined immersed elastic boundaries. Lim and Peskin [10] used such a construction to study the so-called whirling instability [20] of one fiber at low Reynolds number. While this method has the advantage that flows at finite Reynolds numbers can be simulated, being fundamentally grid based, it would be very difficult to use this method to simulate a large number of high aspect ratio fibers.

As a different starting point, we have developed a numerical approach based on a formulation of the problem where we make explicit use both of the Stokes equations, and of the slenderness of the fibers. The key points are that for Stokes flow, boundary integral methods can be employed to reduce the three-dimensional dynamics to the dynamics of the two-dimensional fiber surfaces, [14], and by using slender body asymptotics, this can be further reduced to the dynamics of the one-dimensional fiber center-lines. The resulting integral equations capture the nonlocal interaction of the fiber with itself, as well as with any other structures within the fluid, such as other fibers.

We present first the formulation of the problem and the nonlocal slender body theory in Sect. 2. In Sect. 3, we describe numerical methods that we have developed based on the slender body formulation. In Sect. 4, we show and discuss some results from simulations, including multiple interacting, highly flexible fibers.

2 Mathematical Formulation

Let Ω denote the fluid domain in $\mathbb{R}^3$, external to the fiber. Consider a Newtonian fluid of viscosity μ, with velocity field $\mathbf{u}(\mathbf{x})$, and pressure $p(\mathbf{x})$, where $\mathbf{x} = (x, y, z) \in \mathbb{R}^3$. Assuming that fluid inertia is negligible, $\mathbf{u}$ and p satisfy the Stokes equations:

$$\nabla p - \mu \Delta \mathbf{u} = 0 \;\;\&\;\; \nabla \cdot \mathbf{u} = 0 \quad \text{in}\, \Omega.$$

Let Γ denote the surface of the fiber and $\mathbf{u}_\Gamma$ its surface velocity. We impose the no-slip condition on Γ and require that far away $\mathbf{u}(\mathbf{x})$ is equal to a background velocity $\mathbf{U}_0(\mathbf{x})$, also a solution to the Stokes equations. Hence,

$$\mathbf{u} = \mathbf{u}_\Gamma \text{ on } \Gamma, \qquad \mathbf{u} \to \mathbf{U}_0 \text{ for } \|\mathbf{x}\| \to \infty.$$

In the case of several fibers this can be generalized by considering the union of all fiber surfaces, and imposing no-slip conditions thereon.

A full boundary integral formulation for this problem would yield integral equations on the surfaces of the fibers relating surface stress and surface velocity [14]. For long, slender fibers, such a formulation would be very expensive to solve numerically. Instead we use the fiber slenderness to reduce the integral equations to the fiber center-lines.

2.1 Non-Local Slender Body Approximation

Consider a *slender* fiber; that is $\varepsilon = a/L \ll 1$, where a is the fiber radius, and L is its length. A nonlocal slender body approximation can be derived by placing fundamental solutions to the Stokes equations (Stokeslets and doublets) on the fiber center-line, then applying the technique of matched asymptotics to derive the approximate equation. Such an approximation was derived by Keller and Rubinow in 1976 [8]. Their derivation yields an integral equation with a modified Stokeslet kernel on the fiber center-line and relates the fiber forces to the velocity of the center-line. Johnson [6] added a more detailed analysis and a modified formulation that included accurate treatment of the fiber's free ends, yielding an equation that is asymptotically accurate to $O(\varepsilon^2 \log \varepsilon)$ if the fiber ends are tapered.

This integral expression for a single fiber includes the nonlocal interaction of the fiber with itself, as mediated by the surrounding incompressible fluid. Götz [5] gives an integral expression for the fluid velocity $U(\mathbf{x})$ at any point $\mathbf{x}$ outside the fiber. If there are multiple fibers, their contributions simply add due to the superposition principle of Stokes flow. Hence, to the integral equation derived for one single fiber, one can add the contributions from multiple fibers.

In this manner, we obtain a coupled system of integral equations relating the velocities of fiber center-lines to the forces acting upon the fibers. Here we assume that the fiber forces can be described by Euler-Bernoulli elasticity. Assuming the background flow to be a shear flow of strength $\dot{\gamma}$, we make the problem non-dimensional using a typical fiber length $\tilde{L}$, flow time-scale $\dot{\gamma}^{-1}$, and the force $F = E/\tilde{L}^2$, where E is the rigidity of the fiber.

Denote the fibers by Γ_l, $l = 1, \ldots, M$. Let the center-line of each fiber be parameterized by arclength $s \in [0, L]$, where L is the non-dimensional length of the fiber and let $\mathbf{x}_l(s,t)$ be the coordinates of the fiber center-line. The fibers are assumed inextensible, and therefore the range of the arclength s will not change, nor will the arclength value at any point along the fiber. Hence, s is the material parameter for the fiber, and can be taken as an independent variable. We assume that each fiber exerts a force per unit length, $\mathbf{f}_l(s,t)$, upon the fluid. For fiber Γ_l, we have

$$\bar{\mu}\left(\frac{\partial \mathbf{x}_l(s,t)}{\partial t} - \mathbf{U}_0(\mathbf{x}_l,t)\right) = -\Lambda_l[\mathbf{f}_l](s) - \mathbf{K}_{l,\delta}\left[\mathbf{f}_l\right](s) - \sum_{k=1,k\neq l}^{M}\left[\mathbf{V}_k(\mathbf{x}_l(s)) + \frac{\varepsilon^2}{2}\mathbf{W}_k(\mathbf{x}_l(s))\right], \tag{1}$$

where the sum is over the contributions from all other fibers to the velocity of fiber l, and $\mathbf{U}_0(\mathbf{x},t)$ is the undisturbed background velocity. The non-dimensional parameters are the effective viscosity $\bar{\mu} = 8\pi\mu\dot{\gamma}L^2/(E/L^2)$, representing a ratio between characteristic fluid drag and the fiber elastic force, and the asymptotic parameter $c = \log(\varepsilon^2 e)$, where the radius of the fiber is $r(s) = 2\varepsilon\sqrt{(}s(L-s))$ [6] and e is the natural logarithmic base.

The local operator Λ_l is given by

$$\Lambda_l[\mathbf{f}](s) = \left[-c\left(\mathbf{I} + \hat{\mathbf{s}}(s)\hat{\mathbf{s}}(s)\right) + 2(\mathbf{I} - \hat{\mathbf{s}}(s)\hat{\mathbf{s}}(s))\right]\mathbf{f}(s), \tag{2}$$

and the integral operator $\mathbf{K}_{l,\delta}\left[\mathbf{f}\right](s)$ by

$$\mathbf{K}_{l,\delta}\left[\mathbf{f}\right](s) = \int_{\Gamma_l}\left(\frac{\mathbf{I} + \hat{\mathbf{R}}(s,s')\hat{\mathbf{R}}(s,s')}{\sqrt{|\mathbf{R}(s,s')|^2 + \delta(s)^2}}\,\mathbf{f}(s') - \frac{\mathbf{I} + \hat{\mathbf{s}}(s)\hat{\mathbf{s}}(s)}{\sqrt{|s-s'|^2 + \delta(s)^2}}\,\mathbf{f}(s)\right)\mathrm{d}s'. \tag{3}$$

Here, $\mathbf{R}(s,s') = \mathbf{x}_l(s) - \mathbf{x}_l(s')$, $\hat{\mathbf{R}} = \mathbf{R}/|\mathbf{R}|$ is the normalized $\mathbf{R}$-vector, and $\hat{\mathbf{s}}(s)$ is the unit tangent vector at $\mathbf{x}_l(s)$. $\hat{\mathbf{R}}\,\hat{\mathbf{R}}$ and $\hat{\mathbf{s}}\hat{\mathbf{s}}$ are dyadic products, i.e. $(\hat{\mathbf{R}}\hat{\mathbf{R}})_{ij} = \hat{\mathbf{R}}_i\hat{\mathbf{R}}_j$. Note that these two operators depend on the shape of the fiber (given by $\mathbf{x}_l(s,t)$).

In the original slender-body formulations [8, 6, 5], the regularization parameter δ in (3) is zero. An analysis of the straight fiber case shows that these original slender body formulations are not suitable for numerical computations, due to high wave number instabilities at length-scales not accurately described by slender-body theory [16, 19]. The regularization introduced can remove this instability while retaining the same asymptotic accuracy as the original formulation of Johnson. In particular, we use $\delta(s) = \delta_0\phi(s)$, where $\delta_0 = m\varepsilon$, $m > \sqrt{2}$, and $\phi(s) \in C^1(s)$ is given by

$$\phi(s) = \begin{cases} \nu(s/\gamma) & 0 \le s < \gamma, \\ 1 & \gamma \le s \le 1-\gamma, \\ \nu((1-s)/\gamma) & 1-\gamma < s \le 1, \end{cases} \tag{4}$$

where $\nu(\xi) = \xi^2(3-2\xi)$.

The Stokeslet and doublet contributions from the other fibers are given by

$$\mathbf{V}_k(\bar{\mathbf{x}}) = \int_{\Gamma_k}\left[\frac{\mathbf{I} + \hat{\mathbf{R}}_k(s')\hat{\mathbf{R}}_k(s')}{|\mathbf{R}_k(s')|}\right]\mathbf{f}_k(s')\,\mathrm{d}s', \tag{5}$$

$$\mathbf{W}_k(\bar{\mathbf{x}}) = \int_{\Gamma_k}\left[\frac{\mathbf{I} - 3\hat{\mathbf{R}}_k(s')\hat{\mathbf{R}}_k(s')}{|\mathbf{R}_k(s')|^3}\right]\mathbf{f}_k(s')\,\mathrm{d}s', \tag{6}$$

where $\mathbf{R}_k(s') = \bar{\mathbf{x}} - \mathbf{x}_k(s')$, and $\hat{\mathbf{R}}$ is the normalized $\mathbf{R}$-vector.

For periodic boundary conditions, the sum in (1) must be extended to include contributions from all the periodic images of all fibers. Assuming the domain is periodic in the $\hat{\mathrm{e}}_j$ direction with period length d_j, the sum in (1) becomes

$$\Pi_l^{\mathrm{per}}(s) = \sum_{\substack{k=1,\\ k\neq l}}^{M} \sum_{\substack{p,\\ p\neq 0}} \left[\mathbf{V}_k^p(\mathbf{x}_l(s)) + \frac{\varepsilon^2}{2}\mathbf{W}_k^p(\mathbf{x}_l(s))\right] \tag{7}$$

where $\mathbf{V}_k^p(\bar{\mathbf{x}})$ is defined as $\mathbf{V}_k(\bar{\mathbf{x}})$ in (5), with $\mathbf{R}_k$ replaced by $\mathbf{R}_k^p(s') = \bar{\mathbf{x}} - \mathbf{x}_k(s') + p\,d_j\hat{\mathbf{e}}_j$. Note that $\mathbf{V}_k^0(\bar{\mathbf{x}}) = \mathbf{V}_k(\bar{\mathbf{x}})$. Similarly, $\mathbf{W}_k^p(\bar{\mathbf{x}})$ is defined as $\mathbf{W}_k(\bar{\mathbf{x}})$ in (6) with $\mathbf{R}_k(s')$ replaced by $\mathbf{R}_k^p(s')$. The extension to more periodic directions is straightforward.

2.2 Force Definition

The integral equation (1) relates the velocity of fiber l to the forces acting upon the fiber, as well as to the forces acting on the other fibers. Here we assume that fiber forces are described by Euler-Bernoulli elasticity, and for a fiber given by $\mathbf{x}(s)$ the non-dimensional force (per unit length) is given by

$$\mathbf{f}(s) = -(\,T(s)\mathbf{x}_s\,)_s + \mathbf{x}_{ssss}, \tag{8}$$

where derivatives with respect to arclength are denoted by a subscript s. The first term in (8) is the fiber tensile force, with T the tension, that resists compression and extension. The second term represents bending forces. Twist elasticity is neglected [3]. The ends of the fiber are considered “free”, that is, no forces or moments are exerted upon them, so that $\mathbf{x}_{ss}|_{s=0,L} = \mathbf{x}_{sss}|_{s=0,L} = 0$ and $T|_{s=0,L} = 0$. Note that $\mathbf{f}(s) = \frac{d}{ds}\mathbf{F}(s)$, where $\mathbf{F}(s) = -T(s)\mathbf{x}_s + \mathbf{x}_{sss}$, and so $\mathbf{F}(0) = \mathbf{F}(L) = 0$.

Using these facts, and integrating by parts, $\mathbf{V}_k(\bar{\mathbf{x}})$ as defined in (5) can be rewritten as

$$\mathbf{V}_k(\bar{\mathbf{x}}) = -\int_{\Gamma_k} \frac{(\hat{\mathbf{R}}_k\cdot(\mathbf{x}_k)_s)(\mathbf{I}+3\hat{\mathbf{R}}_k\hat{\mathbf{R}}_k) - ((\mathbf{x}_k)_s\hat{\mathbf{R}}_k + \hat{\mathbf{R}}_k(\mathbf{x}_k)_s)}{|\mathbf{R}_k|^2}\mathbf{F}_k(s')\,\mathrm{d}s', \tag{9}$$

where again $\mathbf{R}_k(s') = \bar{\mathbf{x}} - \mathbf{x}_k(s')$, and $\hat{\mathbf{R}}$ is the normalized $\mathbf{R}$-vector. This formula for $\mathbf{V}_k(\bar{\mathbf{x}})$ shows explicitly the $1/|\mathbf{R}|^2$ decay of the interaction terms.

Similarly, integration by parts of $\mathbf{W}_k(\bar{\mathbf{x}})$ gives,

$$\mathbf{W}_k(\bar{\mathbf{x}}) = -\int_{\Gamma_k} \frac{3(\hat{\mathbf{R}}_k\cdot(\mathbf{x}_k)_s)(\mathbf{I}-5\hat{\mathbf{R}}_k\hat{\mathbf{R}}_k) + 3((\mathbf{x}_k)_s\hat{\mathbf{R}}_k + \hat{\mathbf{R}}_k(\mathbf{x}_k)_s)}{|\mathbf{R}_k|^4}\mathbf{F}_k(s')\,\mathrm{d}s', \tag{10}$$

which shows explicitly its $1/|\mathbf{R}|^4$ decay.

For the periodic sum in (7), using this formulation, one can show that for large p, $|\mathbf{V}_k^{-p}(\mathbf{x}_l(s)) + \mathbf{V}_k^p(\mathbf{x}_l(s))| \sim (p\,d_j)^{-3}$, and hence, by rearranging the sum, these terms can be shown to decay as $1/|\mathbf{R}|^3$, and similarly for $\mathbf{W}_k^p(\bar{\mathbf{x}})$, as $1/|\mathbf{R}|^5$.

2.3 Completing the Formulation

Now, consider the assumption of inextensibility. This condition will determine the line tensions in the fibers. We have that

$$\partial_t((\mathbf{x}_l)_s \cdot (\mathbf{x}_l)_s) = 0 \quad \Rightarrow \quad (\mathbf{x}_l)_s \cdot (\mathbf{x}_l)_{ts} = 0. \tag{11}$$

This condition can be combined with (1) to derive a system of integro-differential equations for the line tensions. The line tensions $T_l(s)$ will then act as Lagrangian multipliers, constraining the motion of the fibers to obey the inextensibility condition. This will work as long as the fibers are exactly the correct length, and hence $(\mathbf{x}_l)_s \cdot (\mathbf{x}_l)_s = 1$ for all s. However, if there is a small length error present, this error will not be corrected. On the contrary, the computed line tension can, depending on the configuration, even act so as to increase this error. Hence, we stabilize the constraint, by replacing the inextensibility condition in (11) with $\frac{1}{2}\,\partial_t((\mathbf{x}_l)_s \cdot (\mathbf{x}_l)_s) = (\mathbf{x}_l)_s \cdot (\mathbf{x}_l)_{ts} = \bar{\mu}\beta(1 - (\mathbf{x}_l)_s \cdot (\mathbf{x}_l)_s)$, which is equivalent to the original condition when $(\mathbf{x}_l)_s \cdot (\mathbf{x}_l)_s = 1$, and which acts to dynamically remove length errors if they are present (β is the penalization parameter, typically set to be of order $O(10)$).

With this, a system of equations for the line tensions $T_l(s)$ $l = 1, \ldots, M$, can be derived. First, use the definition of the force (8) and insert it into the time-dependent equation (1). Then differentiate this equation once with respect to s, so that an equation for $(\mathbf{x}_l)_{ts}$ is obtained. Then take a scalar product with $(\mathbf{x}_l)_s$ and apply the penalized inextensibility condition. Finally, the resulting equation can be simplified using a ladder of differential identities, derived by successive differentiations of $(\mathbf{x}_l)_s \cdot (\mathbf{x}_l)_s = 1$.

The integro-differential equation for the line tensions $T_l(s)$ for fiber l, $l = 1, \ldots, M$, is then given by

$$L_{l,s}[T_l, \mathbf{x}_l] = J_l[\mathbf{x}_l, \mathbf{U}_0] - \sum_{k=1, k\neq l}^{M} (\mathbf{x}_l)_s \cdot \frac{\partial}{\partial s}\left[\mathbf{V}_k(\mathbf{x}_l(s)) + \frac{\varepsilon^2}{2}\mathbf{W}_k(\mathbf{x}_l(s))\right], \tag{12}$$

with

$$\begin{aligned} L_{l,s}[T, \mathbf{x}] &= 2cT_{ss} + (2-c)\,T\,(\mathbf{x}_{ss} \cdot \mathbf{x}_{ss}) - \mathbf{x}_s \cdot \frac{\partial}{\partial s}\mathbf{K}_{l,\delta}\,[(T\mathbf{x}_s)_s] \\ J_l[\mathbf{x}, \mathbf{U}_0] &= \bar{\mu}\mathbf{x}_s \cdot \frac{\partial}{\partial s}\mathbf{U}_0 + (2-7c)(\mathbf{x}_{ss} \cdot \mathbf{x}_{sss}) - 6c(\mathbf{x}_{sss} \cdot \mathbf{x}_{sss}) \\ &\quad - \mathbf{x}_s \cdot \frac{\partial}{\partial s}\mathbf{K}_{l,\delta}\,[\mathbf{x}_{ssss}] - \bar{\mu}\beta(1 - \mathbf{x}_s \cdot \mathbf{x}_s), \end{aligned} \tag{13}$$

together with the boundary condition $T = 0$ at $s = 0, 1$.

In summary, for fiber Γ_l, $l = 1, \ldots, M$, the evolution equation is given by (1), where $\mathbf{f}_l = -(T_l(\mathbf{x}_l)_s)_s + (\mathbf{x}_l)_{ssss}$. The local operator $\Lambda_l[\mathbf{f}](s)$ is given in (2) and the integral operator $\mathbf{K}_{l,\delta}\,[\mathbf{f}]\,(s)$ in (3). The integrals for $\mathbf{V}_k(\bar{\mathbf{x}})$ and $\mathbf{W}_k(\bar{\mathbf{x}})$ in (9)-(10) contain the integrated force $\mathbf{F}_k = -T_k(\mathbf{x}_k)_s + (\mathbf{x}_k)_{sss}$. The auxiliary equation in (12) determines the line-tension for each fiber, completing the formulation of the problem.

3 The Numerical Method

In this section, we briefly describe the current numerical method used to evolve these equations. More details can be found in [19].

3.1 Temporal Discretization

An explicit treatment of all terms in the time-dependent equation (1) would yield a very strict fourth-order stability constraint upon the time-step Δt. This arises basically from the large number of derivatives in the bending term. To avoid this, we treat all occurrences of $\mathbf{x}_{ssss}$ implicitly, and combine this with a second-order backward differentiation formula. Schematically, we write

$$\mathbf{x}_t = \mathbf{F}(\mathbf{x}, \mathbf{x}_{ssss}) + \mathbf{G}(\mathbf{x}), \tag{14}$$

where $\mathbf{x}(s,t)$ are the coordinates of fiber number l, and where the dependence on $\mathbf{U}_0$ and $\mathbf{x}_k$, $k \neq l$ is not explicitly described. Neither is the dependence on the lower s derivatives of $\mathbf{x}(s,t)$, since they will be treated as $\mathbf{x}(s,t)$ itself. The fourth order s-derivative of $\mathbf{x}(s,t)$ ($\mathbf{x}_{ssss}$) is treated implicitly, and all other terms are treated explicitly.

We approximate this decomposition by

$$\frac{1}{2\Delta t}\left(3\mathbf{x}^{n+1} - 4\mathbf{x}^n + \mathbf{x}^{n-1}\right) = \mathbf{F}(2\mathbf{x}^n - \mathbf{x}^{n-1}, \mathbf{x}^{n+1}_{ssss}) + 2\mathbf{G}(\mathbf{x}^n) - \mathbf{G}(\mathbf{x}^{n-1}), \tag{15}$$

where $t^n = n\Delta t$. We find that this scheme yields only a first-order constraint on Δt (i.e. proportional to the spatial grid size).

The dynamics of multiple fibers are coupled to each other through the summation in (1). We treat this coupling term explicitly, that is, as part of $\mathbf{G}(\mathbf{x})$ in (15). In the resulting linear system for $\mathbf{x}_l^{n+1}(s)$, $l = 1, \ldots, M$, the contribution from the other fibers will therefore be in the right hand side, and so the big system decouples into separate linear systems for $\mathbf{x}_l^{n+1}(s)$, $l = 1, \ldots, M$.

The equation for the line tensions $T_l(s)$, $l = 1, \ldots, M$ is given in (12). This is a system of coupled integro-differential equations for the corresponding line tensions that must be solved at every time. To avoid solving one very large linear system for the line tensions on all the fibers, we introduce a fixed point iteration, in which we use the newest updates of the T_k's available ($k \neq l$), when computing $T_l(s)$.

3.2 Spatial Discretization

The fiber center-lines are discretized uniformly in arclength s, with N intervals of step size $h = 1/N$. The discrete points are denoted $s_j = j\,h$, $j = 0, \ldots, N$, and the values $f_j = f(s_j)$. Second-order divided differences are used to approximate spatial derivatives. Standard centered operators are used whenever possible, but at boundaries skew operators are applied.

For the integral operator $\mathbf{K}$ in (3), both terms in the integrand are singular at $s' = s$ for $\delta = 0$, and the integral is only well defined for the difference of these two terms. For the regularized operator, the terms are still nearly singular, and the numerical scheme must be designed with care to accurately treat the difference of these terms.

To do this, we subtract off a term from the first part of the integral, and add the same term to the second part, and write the integral operator (3) as

$$\mathbf{K}_\delta[\mathbf{g}](s) = \int_0^1 \frac{\mathbf{G}(s,s')\,\mathbf{g}(s')}{\sqrt{(s-s')^2+\delta(s)^2}}\,\mathrm{d}s' + (\mathbf{I}+\hat{\mathbf{s}}\,\hat{\mathbf{s}})\int_0^1 \frac{\mathbf{g}(s')-\mathbf{g}(s)}{\sqrt{(s-s')^2+\delta(s)^2}}\,\mathrm{d}s' \tag{16}$$

where $\mathbf{G}(s,s')$ is given by

$$\mathbf{G}(s,s') = \sqrt{\frac{(s-s')^2+\delta(s)^2}{|\mathbf{R}|^2+\delta(s)^2}}\,(\mathbf{I}+\hat{\mathbf{R}}\,\hat{\mathbf{R}}) - (\mathbf{I}+\hat{\mathbf{s}}\,\hat{\mathbf{s}}). \tag{17}$$

We then treat each part separately, by approximating the argument to the operator, as well as $\mathbf{G}(s,s')$ by piecewise polynomials. These are all smooth, well behaved functions.

In the end, we need to evaluate integrals of the form

$$\int_{s_j}^{s_{j+1}} \frac{(s'-s_j)^p}{\sqrt{|s-s'|^2+\delta(s)^2}}\,\mathrm{d}s' = \int_0^h \frac{\alpha^p}{\sqrt{\alpha^2+b\alpha+c+\delta(s)^2}}\,\mathrm{d}\alpha$$

where $b = 2(s_j - s)$ and $c = (s_j - s)^2$, and $p = 0,\dots,4$. These integrals have analytical formulas, becoming somewhat lengthy as p increases. By evaluating these integrals analytically, the rapidly changing part where s' is close to s can be treated exactly.

In the line tension equation (12), terms like $\mathbf{x}_s \cdot \frac{\partial}{\partial s}\mathbf{K}_\delta\,[\mathbf{g}]$ appear. These differentiated integral terms are approximated to second order by

$$\frac{\partial}{\partial s}\mathbf{K}_\delta\,[\mathbf{g}]\,(s)|_{s=s_i} \approx \frac{1}{h}\left[\mathbf{K}_\delta\,[\mathbf{g}]\,(s_{j+1/2}) - \mathbf{K}_\delta\,[\mathbf{g}]\,(s_{j-1/2})\right]. \tag{18}$$

This compact centered approximation of the derivative is important to achieve a stable numerical approximation of the line tension equation.

3.3 The Interaction Terms

In the case of periodicity, we need to compute a sum over p for the contribution from each fiber, as indicated in (7). This infinite sum over all the periodic images of a fiber is approximated by including $\mathbf{V}_k^p$ for the three closest images, and approximations to $\mathbf{V}_k^p$ for $2(Q-1)$ more images. We have typically used $Q = 20$. With the justification that as $|\mathbf{R}_k^p|$ gets large, $\mathbf{R}_k^p(s')$ will not vary much in the integrand for $\mathbf{V}_k^p(\bar{\mathbf{x}})$, an approximation is made by replacing $\mathbf{R}_k^p(s')$ with $\mathbf{R}_k^p(1/2)$. This is then a constant vector, which can be moved out of the integral. We will then only need to integrate one symmetric dyadic product,

$$\int_{\Gamma_k} (\mathbf{x}_k)_s \, \mathbf{F}_k(s')\mathrm{d}s'. \tag{19}$$

Once we have the 6 independent components of this integral, and $\mathbf{R}_k^p(1/2)$ for any p, we can for any $\bar{\mathbf{x}}$ compute the approximation $\widetilde{\mathbf{V}}_k^p(\bar{\mathbf{x}})$ to $\mathbf{V}_k^p(\bar{\mathbf{x}})$. Including these approximate terms in the periodic sum gives a substantial improvement when compared to simply truncating the sum, and it can be done at very small extra cost.

To evaluate the integrals $\mathbf{V}_k(\mathbf{x}_l(s))$ and $\mathbf{W}_k(\mathbf{x}_l(s))$ we simply use the trapezoidal rule. This is a second-order method and it is accurate so long as $|\mathbf{R}|$ is not too small. If two fibers come within very close proximity of each other, a refined calculation is made. At a first stage, it is done by simply subdividing intervals in the integral, to achieve a better approximation of the $1/|\mathbf{R}|^2$ and $1/|\mathbf{R}|^4$-terms. If the fibers are within an ε-scale away from each other, an interpolation is done between the velocity computed from this integral and the velocity on the fiber center-line as given by (1).

4 The Dynamics of Fiber Suspensions

As was noted in the introduction, a straight fiber placed in a plane shear flow will become unstable to buckling as the strength of the shear flow increases relative to the bending rigidity of the filament. This means that small shape perturbations can grow into substantial bending of the filament. Also, perturbations in the flow field, such as the disturbance from other fibers in the flow can trigger such a buckling.

In Fig. 1, the rotation of one single fiber placed in an oscillatory background shear flow is shown in a sequence of plots. The fiber is of unit length, and we set $\bar{\mu} = 5 \cdot 10^5$ and $\varepsilon = 10^{-3}$. We use $N = 100$ points to discretize the fiber, and time-step $\Delta t = 0.0128$. The background shear flow is given by $\mathbf{U}_0 = (\sin(2\pi\omega t)y, 0, 0)$, where $\omega = (2000\,\Delta t)^{-1}$, so that one period is 2000 time-steps, i.e 25.6 time units.

The results are plotted at different times within the second period of the shear flow, and show simply a straight fiber rotating and translating in the shear flow. At this high $\bar{\mu}$ for this value of ε, the buckling instability is however very strong, and it is only in a very clean case - no perturbations in the flow, no shape perturbation of the fiber - that it will stay straight. Once the fiber bends, it can bend in many different ways, depending on the perturbation.

In Fig. 2, results are shown for a simulation where we initialize this one fiber exactly as before (plotted in black), but this time, we also include three other fibers in the simulation (plotted in gray). We impose periodic boundary conditions in the streamwise (x) direction, with a period twice the filament length. All the fibers are initially straight.

The results are shown at the same times as in Fig. 1, i.e. at different times within the second period. To be able to compare Figs. 1 and 2, the plots in Fig. 1, are plotted

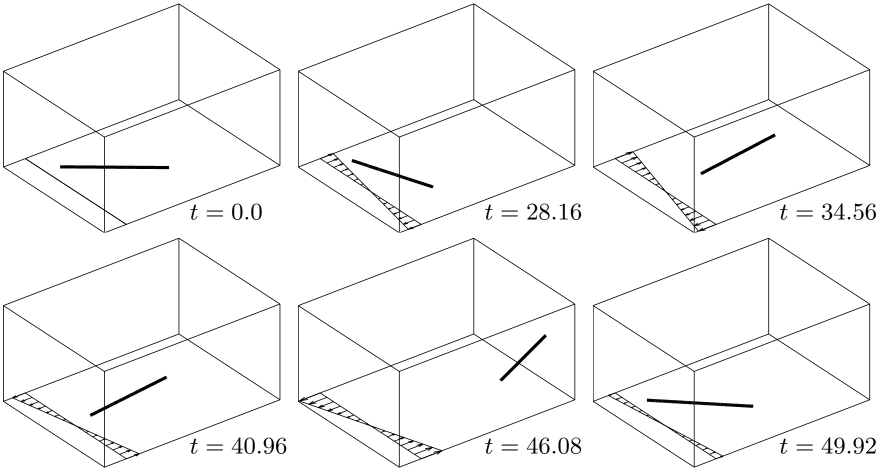


Fig. 1. One single fiber translating and rotating in an oscillatory background shear flow. The results are plotted at times within the second period of this flow (one period is 25.6 time units). The velocity profile of the background flow is indicated in each plot. The fiber is shifted periodically within the plotted box when needed for it to fall inside the box

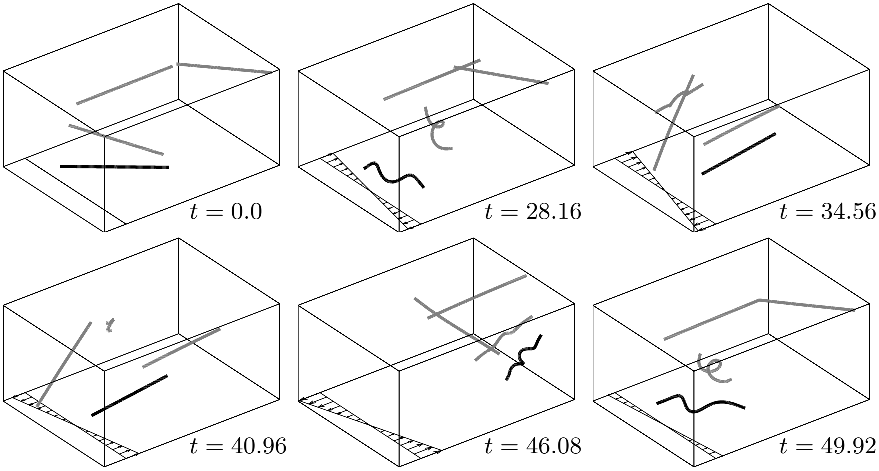


Fig. 2. Four fibers (initially straight) in an oscillatory background shear flow. Fiber initialized as in Fig. 1 is plotted in black. Times for plots etc. as in Fig. 1

as if the domain was periodic as described above, i.e. the fiber is periodically shifted into this domain.

The domain of the computations for the single fiber was however not periodic. The disturbances from periodic images of the fiber itself could also induce a buckling. Actually, with these parameters, and therefore such a strong instability, numerical errors could potentially also initiate a buckling if the simulation is continued over a longer time. However, perturbations arising from other fibers in the flow are naturally much larger than those from numerical errors, and will hence when such are present, trigger the instability at a much earlier time, as shown in the figures.

As the fibers bend, they store elastic energy, that will later be released back to the system. The elastic energy is defined as $\mathcal{E}_{el} = \sum_l \int \kappa_l^2(s)ds$, where κ_l is the pointwise curvature of fiber l, and will hence directly depend on how substantial bending that occurs.

A fiber is susceptible to bending when it is under compression. In the case of this oscillatory background shear flow, whether a fiber is under compression or extension at a certain instant depends on the angle of the fiber in the plane of the shear, relative to the flow direction. As the flow reverses direction, a fiber that was under compression will then be under extension.

In general, a fiber must be under compression for some time before buckling occurs. On the other hand, a fiber that is bent at one instant, will become more straight the longer it is under extension. Therefore, a suspension of fibers will behave somewhat differently depending on the period of the oscillating flow. For example, the maximum elastic energy stored by the fibers over one period will differ.

In Figs. 3-4, results from simulations of 30 interacting fibers are shown for $N = 50$, $\Delta t = 0.0256$, $\varepsilon = 10^{-3}$, $\bar{\mu} = 3 \cdot 10^5$. The domain is the periodic in the streamwise direction with a period twice the fiber length. The background

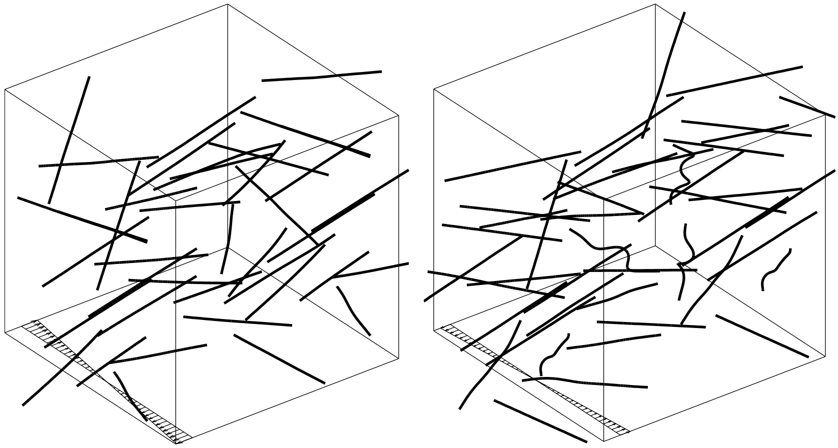


Fig. 3. The fiber configurations at times $t = 53.76$ and $t = 62.72$ (run A). These are the times of minimum and maximum elastic energy within the fifth period of the flow

shear flow is given by $\mathbf{U}_0 = (\sin(2\pi\omega t)y, 0, 0)$, where for the results in Fig. 3, $\omega = (500\,\Delta t)^{-1}$, so that one period is 500 time-steps, i.e 12.8 time units (run A). In Fig. 4, we have $\omega = (1000\,\Delta t)^{-1}$, so that one period is 25.6 time units (run B).

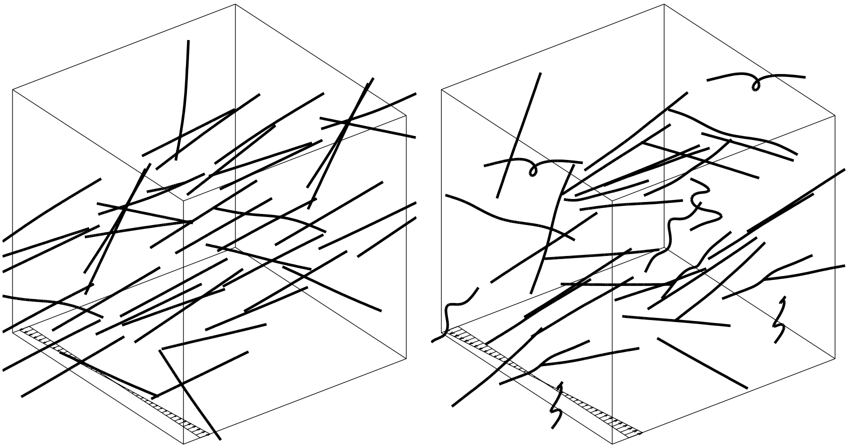


Fig. 4. The fiber configurations at times $t = 117.76$ and $t = 124.15$ (run B). These are the times of minimum and maximum elastic energy within the fifth period of the flow

In Figs. 5-6, the elastic energies for the two simulations are plotted as functions of time, for eight periods of the background flow. The maximum energy for run A, over all of the eight periods is 68.64. The elastic energy for the configurations in Fig. 3 are 0.35 and 65.01, respectively. For run B, the maximum energy over all of the eight periods is 155.03. The elastic energy for the configurations in Fig. 4 are 0.27 and 122.94, respectively. While the peak value of the elastic energy within a period increases over the eight periods simulated for run A, it initially decreases for run B, where after we see a slight increase.

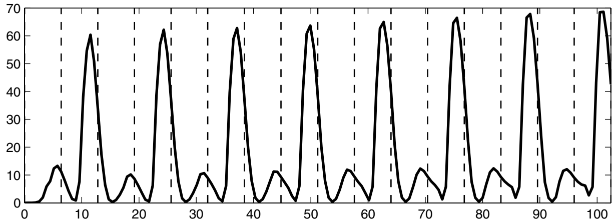


Fig. 5. Elastic energy plotted versus time t for run A. (Instant fiber configurations plotted in Fig. 3). Dashed lines indicate each half period, i.e. times when a change in flow direction occurs

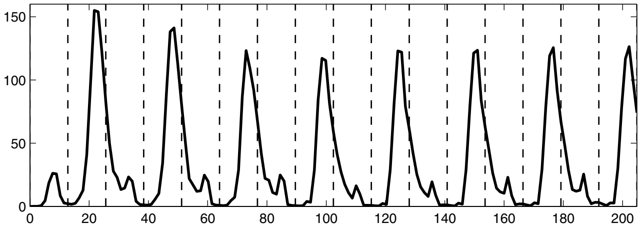


Fig. 6. Elastic energy plotted versus time t for run B. (Instant fiber configurations plotted in Fig. 4). Dashed lines indicate each half period, i.e. times when a change in flow direction occurs

These two runs have the same initial configuration of fibers, and all the physical parameters, except the frequency of the oscillating background shear flow, are the same. We however get a much larger maximum energy for the background flow with the longer period, and the two simulations show a different pattern in the development of the elastic energy.

Hence, the simulations we can perform offer a wealth of phenomena to study, and many interesting observations can be made. However, the results still depend on the specific initial configuration of the fibers, and simulations of a larger number of fibers is desirable to compute representative quantities. We are currently working to increase the number of fibers that can be simulated, by parallelizing the code and by employing a fast summation method to compute the fiber-fiber interactions.

5 Concluding Remarks

In this paper, we have considered the challenging problem of simulating multiple, highly flexible and slender fibers in a Stokesian fluid.

We have developed a formulation for this three dimensional problem that is based on slender body asymptotics, and an efficient numerical method based on this formulation. The mathematical description takes the form of a coupled system of integral equations along the center-lines of the fibers. This is a formulation that takes into account both fluid-fiber and fiber-fiber interactions, as mediated by the fluid. The numerical method is based on finite differences to compute derivatives in space and time, implicit time-stepping, and product integration to treat the integral terms. Special care has been taken in the quadrature algorithm to ensure proper cancellation of nearly singular terms.

The possibility to perform these simulations opens up a range of phenomena to study. Quantities like elastic energy and normal stress differences can easily be computed, and we can now start to address the onset of the non-Newtonian effects seen as flexible fibers are added to a Newtonian solvent.

References

1. L. Becker and M. Shelley. The instability of elastic filaments in shear flow yields first normal stress differences. *Phys. Rev. Lett.*, 87:198301, 2001.
2. S. Childress. *Mechanics of Swimming and Flying*. Cambridge University Press, Cambridge, 1981.
3. R. Goldstein, T. Powers, and C. Wiggins. Viscous nonlinear dynamics of twist and writhe. *Phys. Rev. Lett.*, 80:5232, 1998.
4. S. Goto, H. Nagazono, and H. Kato. Polymer solutions. 1: Mechanical properties. *Rheol. Acta*, 25:119–129, 1986.
5. T. Götz. *Interactions of fibers and flow: Asymptotics, theory and numerics*. PhD thesis, University of Kaiserslautern, Germany, 2000.
6. R.E. Johnson. An improved slender-body theory for Stokes flow. *J. Fluid Mech.*, 99:411–431, 1980.
7. C.G. Joung, N. Phan-Thien, and X. Fan. Direct simulation of flexible fibers. *J. Non-Newtonian Fluid Mech.*, 99:1–36, 2001.
8. J. Keller and S. Rubinow. Slender-body theory for slow viscous flow. *J. Fluid Mech.*, 75:705–714, 1976.
9. R.G. Larson. *The Structure and Rheology of Complex Fluids*. Oxford University Press, 1998.
10. S. Lim and C. S. Peskin. Simulations of the whirling instability by the immersed boundary method. *SIAM J. Sci. Comput.*, 25:2066–2083, 2004.
11. Z. Ning and J. R. Melrose. A numerical model for simulating mechanical behavior of flexible filaments. *J. Chem. Phys.*, 111:10717–10726, 1999.
12. P. Palffy-Muhoray, B. Bergersen, H. Lin, R. Meyer, and Z. Racz. Filaments in liquid crystals: Structure and dynamics. In S. Kai, editor, *Pattern Formation in Complex Dissipative Systems*, Singapore, 1991. World Scientific.
13. C. S. Peskin. The immersed boundary method. *Acta Numerica*, 11:479–517, 2002.
14. C. Pozrikidis. *Boundary integral and singularity methods for linearized viscous flow*. Cambridge University Press, 1992.
15. R.F. Ross and D.J. Klingenberg. Dynamic simulation of flexible fibers. *J. Chem. Phys.*, 106:2949–2960, 1997.
16. M.J. Shelley and T. Ueda. The Stokesian hydrodynamics of flexing, stretching filaments. *Physica D*, 146:221–245, 2000.
17. J.M. Stockie. Simulating the motion of flexible pulp fibers using the immersed boundary method. *J. Comput. Phys.*, 147:147–165, 1998.
18. L.H. Switzer and D.J. Klingenberg. Rheology of sheared flexible fiber suspensions via fiber-level simulations. *J. of Rheol.*, 47:759–778, 2003.
19. A.K. Tornberg and M.J. Shelley. Simulating the dynamics and interactions of flexible fibers in Stokes flow. *J. Comput. Phys.*, 196:8–40, 2004.
20. C. Wolgemuth, T. Powers, and R. Goldstein. Twirling and whirling: Viscous dynamics of rotating elastic filaments. *Phys. Rev. Lett.*, 84:1623, 2000.
21. S. Yamamoto and T. Matsuoka. Dynamic simulations of fiber suspensions in shear flow. *J. Chem. Phys.*, 102:2254–2260, 1995.

Editorial Policy

§1. Volumes in the following three categories will be published in LNCSE:

i) Research monographs
ii) Lecture and seminar notes
iii) Conference proceedings

Those considering a book which might be suitable for the series are strongly advised to contact the publisher or the series editors at an early stage.

§2. Categories i) and ii). These categories will be emphasized by Lecture Notes in Computational Science and Engineering. **Submissions by interdisciplinary teams of authors are encouraged.** The goal is to report new developments – quickly, informally, and in a way that will make them accessible to non-specialists. In the evaluation of submissions timeliness of the work is an important criterion. Texts should be well-rounded, well-written and reasonably self-contained. In most cases the work will contain results of others as well as those of the author(s). In each case the author(s) should provide sufficient motivation, examples, and applications. In this respect, Ph.D. theses will usually be deemed unsuitable for the Lecture Notes series. Proposals for volumes in these categories should be submitted either to one of the series editors or to Springer-Verlag, Heidelberg, and will be refereed. A provisional judgment on the acceptability of a project can be based on partial information about the work: a detailed outline describing the contents of each chapter, the estimated length, a bibliography, and one or two sample chapters – or a first draft. A final decision whether to accept will rest on an evaluation of the completed work which should include

- at least 100 pages of text;
- a table of contents;
- an informative introduction perhaps with some historical remarks which should be accessible to readers unfamiliar with the topic treated;
- a subject index.

§3. Category iii). Conference proceedings will be considered for publication provided that they are both of exceptional interest and devoted to a single topic. One (or more) expert participants will act as the scientific editor(s) of the volume. They select the papers which are suitable for inclusion and have them individually refereed as for a journal. Papers not closely related to the central topic are to be excluded. Organizers should contact Lecture Notes in Computational Science and Engineering at the planning stage.

In exceptional cases some other multi-author-volumes may be considered in this category.

§4. Format. Only works in English are considered. They should be submitted in camera-ready form according to Springer-Verlag's specifications.
Electronic material can be included if appropriate. Please contact the publisher.
Technical instructions and/or TEX macros are available via
http://www.springeronline.com/sgw/cda/frontpage/0,10735,5-111-2-71391-0,00.html
The macros can also be sent on request.

General Remarks

Lecture Notes are printed by photo-offset from the master-copy delivered in camera-ready form by the authors. For this purpose Springer-Verlag provides technical instructions for the preparation of manuscripts. See also *Editorial Policy*.

Careful preparation of manuscripts will help keep production time short and ensure a satisfactory appearance of the finished book.

The following terms and conditions hold:

Categories i), ii), and iii):
Authors receive 50 free copies of their book. No royalty is paid. Commitment to publish is made by letter of intent rather than by signing a formal contract. Springer-Verlag secures the copyright for each volume.

For conference proceedings, editors receive a total of 50 free copies of their volume for distribution to the contributing authors.

All categories:
Authors are entitled to purchase further copies of their book and other Springer mathematics books for their personal use, at a discount of 33,3 % directly from Springer-Verlag.

Springer-Verlag, Mathematics Editorial IV
Tiergartenstrasse 17
D-69121 Heidelberg, Germany
Tel.: *49 (6221) 487-8185
Fax: *49 (6221) 487-8355
e-mail: Martin.Peters@springer-sbm.com

Lecture Notes in Computational Science and Engineering

Vol. 1 D. Funaro, *Spectral Elements for Transport-Dominated Equations.* 1997. X, 211 pp. Softcover. ISBN 3-540-62649-2

Vol. 2 H. P. Langtangen, *Computational Partial Differential Equations.* Numerical Methods and Diffpack Programming. 1999. XXIII, 682 pp. Hardcover. ISBN 3-540-65274-4

Vol. 3 W. Hackbusch, G. Wittum (eds.), *Multigrid Methods V.* Proceedings of the Fifth European Multigrid Conference held in Stuttgart, Germany, October 1-4, 1996. 1998. VIII, 334 pp. Softcover. ISBN 3-540-63133-X

Vol. 4 P. Deuflhard, J. Hermans, B. Leimkuhler, A. E. Mark, S. Reich, R. D. Skeel (eds.), *Computational Molecular Dynamics: Challenges, Methods, Ideas.* Proceedings of the 2nd International Symposium on Algorithms for Macromolecular Modelling, Berlin, May 21-24, 1997. 1998. XI, 489 pp. Softcover. ISBN 3-540-63242-5

Vol. 5 D. Kröner, M. Ohlberger, C. Rohde (eds.), *An Introduction to Recent Developments in Theory and Numerics for Conservation Laws.* Proceedings of the International School on Theory and Numerics for Conservation Laws, Freiburg / Littenweiler, October 20-24, 1997. 1998. VII, 285 pp. Softcover. ISBN 3-540-65081-4

Vol. 6 S. Turek, *Efficient Solvers for Incompressible Flow Problems.* An Algorithmic and Computational Approach. 1999. XVII, 352 pp, with CD-ROM. Hardcover. ISBN 3-540-65433-X

Vol. 7 R. von Schwerin, *Multi **B**ody **S**ystem **SIM**ulation.* Numerical Methods, Algorithms, and Software. 1999. XX, 338 pp. Softcover. ISBN 3-540-65662-6

Vol. 8 H.-J. Bungartz, F. Durst, C. Zenger (eds.), *High Performance Scientific and Engineering Computing.* Proceedings of the International FORTWIHR Conference on HPSEC, Munich, March 16-18, 1998. 1999. X, 471 pp. Softcover. 3-540-65730-4

Vol. 9 T. J. Barth, H. Deconinck (eds.), *High-Order Methods for Computational Physics.* 1999. VII, 582 pp. Hardcover. 3-540-65893-9

Vol. 10 H. P. Langtangen, A. M. Bruaset, E. Quak (eds.), *Advances in Software Tools for Scientific Computing.* 2000. X, 357 pp. Softcover. 3-540-66557-9

Vol. 11 B. Cockburn, G. E. Karniadakis, C.-W. Shu (eds.), *Discontinuous Galerkin Methods.* Theory, Computation and Applications. 2000. XI, 470 pp. Hardcover. 3-540-66787-3

Vol. 12 U. van Rienen, *Numerical Methods in Computational Electrodynamics.* Linear Systems in Practical Applications. 2000. XIII, 375 pp. Softcover. 3-540-67629-5

Vol. 13 B. Engquist, L. Johnsson, M. Hammill, F. Short (eds.), *Simulation and Visualization on the Grid.* Parallelldatorcentrum Seventh Annual Conference, Stockholm, December 1999, Proceedings. 2000. XIII, 301 pp. Softcover. 3-540-67264-8

Vol. 14 E. Dick, K. Riemslagh, J. Vierendeels (eds.), *Multigrid Methods VI.* Proceedings of the Sixth European Multigrid Conference Held in Gent, Belgium, September 27-30, 1999. 2000. IX, 293 pp. Softcover. 3-540-67157-9

Vol. 15 A. Frommer, T. Lippert, B. Medeke, K. Schilling (eds.), *Numerical Challenges in Lattice Quantum Chromodynamics.* Joint Interdisciplinary Workshop of John von Neumann Institute for Computing, Jülich and Institute of Applied Computer Science, Wuppertal University, August 1999. 2000. VIII, 184 pp. Softcover. 3-540-67732-1

Vol. 16 J. Lang, *Adaptive Multilevel Solution of Nonlinear Parabolic PDE Systems.* Theory, Algorithm, and Applications. 2001. XII, 157 pp. Softcover. 3-540-67900-6

Vol. 17 B. I. Wohlmuth, *Discretization Methods and Iterative Solvers Based on Domain Decomposition.* 2001. X, 197 pp. Softcover. 3-540-41083-X

Vol. 18 U. van Rienen, M. Günther, D. Hecht (eds.), *Scientific Computing in Electrical Engineering.* Proceedings of the 3rd International Workshop, August 20-23, 2000, Warnemünde, Germany. 2001. XII, 428 pp. Softcover. 3-540-42173-4

Vol. 19 I. Babuška, P. G. Ciarlet, T. Miyoshi (eds.), *Mathematical Modeling and Numerical Simulation in Continuum Mechanics.* Proceedings of the International Symposium on Mathematical Modeling and Numerical Simulation in Continuum Mechanics, September 29 - October 3, 2000, Yamaguchi, Japan. 2002. VIII, 301 pp. Softcover. 3-540-42399-0

Vol. 20 T. J. Barth, T. Chan, R. Haimes (eds.), *Multiscale and Multiresolution Methods.* Theory and Applications. 2002. X, 389 pp. Softcover. 3-540-42420-2

Vol. 21 M. Breuer, F. Durst, C. Zenger (eds.), *High Performance Scientific and Engineering Computing.* Proceedings of the 3rd International FORTWIHR Conference on HPSEC, Erlangen, March 12-14, 2001. 2002. XIII, 408 pp. Softcover. 3-540-42946-8

Vol. 22 K. Urban, *Wavelets in Numerical Simulation.* Problem Adapted Construction and Applications. 2002. XV, 181 pp. Softcover. 3-540-43055-5

Vol. 23 L. F. Pavarino, A. Toselli (eds.), *Recent Developments in Domain Decomposition Methods.* 2002. XII, 243 pp. Softcover. 3-540-43413-5

Vol. 24 T. Schlick, H. H. Gan (eds.), *Computational Methods for Macromolecules: Challenges and Applications.* Proceedings of the 3rd International Workshop on Algorithms for Macromolecular Modeling, New York, October 12-14, 2000. 2002. IX, 504 pp. Softcover. 3-540-43756-8

Vol. 25 T. J. Barth, H. Deconinck (eds.), *Error Estimation and Adaptive Discretization Methods in Computational Fluid Dynamics.* 2003. VII, 344 pp. Hardcover. 3-540-43758-4

Vol. 26 M. Griebel, M. A. Schweitzer (eds.), *Meshfree Methods for Partial Differential Equations.* 2003. IX, 466 pp. Softcover. 3-540-43891-2

Vol. 27 S. Müller, *Adaptive Multiscale Schemes for Conservation Laws.* 2003. XIV, 181 pp. Softcover. 3-540-44325-8

Vol. 28 C. Carstensen, S. Funken, W. Hackbusch, R. H. W. Hoppe, P. Monk (eds.), *Computational Electromagnetics.* Proceedings of the GAMM Workshop on "Computational Electromagnetics", Kiel, Germany, January 26-28, 2001. 2003. X, 209 pp. Softcover. 3-540-44392-4

Vol. 29 M. A. Schweitzer, *A Parallel Multilevel Partition of Unity Method for Elliptic Partial Differential Equations.* 2003. V, 194 pp. Softcover. 3-540-00351-7

Vol. 30 T. Biegler, O. Ghattas, M. Heinkenschloss, B. van Bloemen Waanders (eds.), *Large-Scale PDE-Constrained Optimization.* 2003. VI, 349 pp. Softcover. 3-540-05045-0

Vol. 31 M. Ainsworth, P. Davies, D. Duncan, P. Martin, B. Rynne (eds.), *Topics in Computational Wave Propagation.* Direct and Inverse Problems. 2003. VIII, 399 pp. Softcover. 3-540-00744-X

Vol. 32 H. Emmerich, B. Nestler, M. Schreckenberg (eds.), *Interface and Transport Dynamics.* Computational Modelling. 2003. XV, 432 pp. Hardcover. 3-540-40367-1

Vol. 33 H. P. Langtangen, A. Tveito (eds.), *Advanced Topics in Computational Partial Differential Equations.* Numerical Methods and Diffpack Programming. 2003. XIX, 658 pp. Softcover. 3-540-01438-1

Vol. 34 V. John, *Large Eddy Simulation of Turbulent Incompressible Flows.* Analytical and Numerical Results for a Class of LES Models. 2004. XII, 261 pp. Softcover. 3-540-40643-3

Vol. 35 E. Bänsch (ed.), *Challenges in Scientific Computing - CISC 2002.* Proceedings of the Conference *Challenges in Scientific Computing*, Berlin, October 2-5, 2002. 2003. VIII, 287 pp. Hardcover. 3-540-40887-8

Vol. 36 B. N. Khoromskij, G. Wittum, *Numerical Solution of Elliptic Differential Equations by Reduction to the Interface.* 2004. XI, 293 pp. Softcover. 3-540-20406-7

Vol. 37 A. Iske, *Multiresolution Methods in Scattered Data Modelling.* 2004. XII, 182 pp. Softcover. 3-540-20479-2

Vol. 38 S.-I. Niculescu, K. Gu (eds.), *Advances in Time-Delay Systems.* 2004. XIV, 446 pp. Softcover. 3-540-20890-9

Vol. 39 S. Attinger, P. Koumoutsakos (eds.), *Multiscale Modelling and Simulation.* 2004. VIII, 277 pp. Softcover. 3-540-21180-2

Vol. 40 R. Kornhuber, R. Hoppe, J. Périaux, O. Pironneau, O. Wildlund, J. Xu (eds.), *Domain Decomposition Methods in Science and Engineering.* 2005. XVIII, 690 pp. Softcover. 3-540-22523-4

Vol. 41 T. Plewa, T. Linde, V.G. Weirs (eds.), *Adaptive Mesh Refinement – Theory and Applications.* 2005. XIV, 552 pp. Softcover. 3-540-21147-0

Vol. 42 A. Schmidt, K.G. Siebert, *Design of Adaptive Finite Element Software.* The Finite Element Toolbox ALBERTA. 2005. XII, 322 pp. Hardcover. 3-540-22842-X

Vol. 43 M. Griebel, M.A. Schweitzer (eds.), *Meshfree Methods for Partial Differential Equations II.* 2005. XIII, 303 pp. Softcover. 3-540-23026-2

Vol. 44 B. Engquist, P. Lötstedt, O. Runborg (eds.), *Multiscale Methods in Science and Engineering.* 2005. XII, 291 pp. Softcover. 3-540-25335-1

For further information on these books please have a look at our mathematics catalogue at the following URL: `www.springeronline.com/series/3527`

Texts in Computational Science and Engineering

Vol. 1 H. P. Langtangen, *Computational Partial Differential Equations.* Numerical Methods and Diffpack Programming. 2nd Edition 2003. XXVI, 855 pp. Hardcover. ISBN 3-540-43416-X

Vol. 2 A. Quarteroni, F. Saleri, *Scientific Computing with MATLAB.* 2003. IX, 257 pp. Hardcover. ISBN 3-540-44363-0

Vol. 3 H. P. Langtangen, *Python Scripting for Computational Science.* 2004. XXII, 724 pp. Hardcover. ISBN 3-540-43508-5

For further information on these books please have a look at our mathematics catalogue at the following URL: `www.springeronline.com/series/5151`