# Algorithms for stochastic Galerkin projections: Solvers, basis adaptation and multiscale modeling and reduction

by

Ramakrishna Tipireddy

---

A Dissertation Presented to the
FACULTY OF THE USC GRADUATE SCHOOL
UNIVERSITY OF SOUTHERN CALIFORNIA
In Partial Fulfillment of the
Requirements for the Degree
DOCTOR OF PHILOSOPHY
(CIVIL ENGINEERING)

June 2013

# Dedication

To my mother, my first teacher.

# Acknowledgements

Foremost, I would like to express my sincere gratitude to my advisor Prof. Roger Ghanem for his continuous support, patience, insightful suggestions and critical comments, enthusiasm and continuous push for betterment of the work throughout the course of my PhD. I would like to thank him for introducing me to wide areas of research and providing many opportunities to collaborate with the experts in various research fields and present my work in numerous international conferences. His high academic standards will always be a source of inspiration for me.

I would like to thank Dr. Eric Phipps for mentoring me during my internship at Sandia National Labs, for serving on my qualifying committee and providing valuable support in technical work and technical writing during the course of this work. I would like to thank my collaborators, Prof. Somnath Ghosh and Dr. Daniel Paquet for providing experimental data and their support during this work.

I would like to thank my dissertation committee members, Prof. Sami Masri, Prof. Aiichiro Nakano and Prof. Robert Lucas for their valuable comments and suggestions for betterment of this work. I would like to thank Prof. Erik Johnson for serving on my qualifying committee and providing valuable suggestions regarding my presentation skills.

I gratefully acknowledge the financial support received from US Department of Energy, National Science foundation during the course of this work. I would like to thank Viterbi school of engineering for offering me Viterbi fellowship.

I would like to extend my sincere thanks to my former and current colleagues, friends who supported directly or indirectly during this work. I would like thank Dr. Maarten Arnst, Dr. Sonjoy Das and Dr. Johann Guilleminot for their support and guidance in the earlier phase of my PhD. My sincere thanks to my former colleagues Dr. Arash Noshadravan, Dr. Maud Comboul, Dr. Sara Abedi, Daniel Lakeland, Dr. Evangelia Kalligiannaki for their input during many discussions I had. I would also like to thank my other past and current fellow group members, Dr. Hadi Meidani, Dr. Yongbo Peng, Dr. Bedrick Sousdik, Iman Yadegaran, Vahid Keshavarzzadeh, Hamed Haddadzadegan, Charanraj Thimmisetty, Prasanth Koganti and Matthew Sexton for all the discussions and sharing thoughts on interesting problems. I would like to thank my friends, Dr. Chandrika Prakash Vyasarayani, Dr. Nagesh Janapala, Dr. Karthikeyan Chockalingam, Sukhpreet Sandhu, Dr. S. Kumar and Dr. Viswanath Chinthapenta for their constant support and encouragement.

I would like to thank Prof. C. S. Manohar, who taught me various subjects, guided my research work at the Indian Institute of Science. He has been a source of inspiration and he motivated and encouraged me to pursue higher studies. I am grateful to all my teachers who taught me and inspired me throughout my student life.

Last but not the least, I would like to thank my family: my father Chalama Reddy, my brother Harikrishna Reddy for their unconditional love and sacrifices without which this work would not have been completed. I would like to thank my uncle Kesavula Reddy, who has been influential in my early stages of education.

# Table of Contents

**Dedication** **1**

**Acknowledgements** **3**

**List of Tables** **7**

**List of Figures** **8**

**Abstract** **12**

**Chapter 1 Introduction** **1**
1.1 Motivation . . . . . 1
1.2 Problem statement . . . . . 2
1.3 Stochastic upscaling . . . . . 4
1.4 Challenges . . . . . 5

**Chapter 2 Stochastic Galerkin methods** **8**
2.1 Introduction . . . . . 8
2.2 Stochastic partial differential equations . . . . . 8
2.3 Input random field model . . . . . 9
2.3.1 Karhunen-Loève expansion . . . . . 10
2.3.2 Polynomial chaos expansion . . . . . 11
2.4 Stochastic Galerkin method . . . . . 12

**Chapter 3 Iterative solvers and preconditioners** **16**
3.1 Introduction . . . . . 16
3.2 Solution methods for stochastic Galerkin systems . . . . . 16
3.2.1 Jacobi mean algorithm . . . . . 17
3.2.2 Gauss-Seidel mean iterative method . . . . . 18
3.2.3 Krylov based iterative methods with matrix-free operations . . . . . 21
3.3 Problem Statement . . . . . 24
3.4 Sparse grid collocation method . . . . . 26
3.5 Numerical illustration . . . . . 27

**Chapter 4 Basis adaptation - dimension reduction** **39**
4.1 Introduction . . . . . 39

4.2 Polynomial Chaos Decomposition . . . 41
4.2.1 Change of Basis . . . 42
4.2.2 Reduction via Projection . . . 43
4.3 Special Cases . . . 44
4.3.1 Case where Gaussian components of $q$ are known . . . 45
4.3.2 Case where quadratic components of $q$ are known . . . 46
4.4 Application to Equations with Random Coefficients . . . 48
4.4.1 Non-Intrusive Implementation . . . 48
4.4.2 Intrusive Implementation . . . 50
4.4.3 Adapted Representation of Model Parameters . . . 52
4.5 Numerical Example . . . 53
4.5.1 Algebraic Equation . . . 53
4.5.2 Linear Elasticity with Nonlinear Quantities of Interest . . . 55
4.5.3 Random eigenvalue problem . . . 59

**Chapter 5 Stochastic upscaling 66**
5.1 Introduction . . . 66
5.2 Problem statement . . . 67
5.2.1 Navier-Stokes equations at fine-scale . . . 68
5.2.2 Darcy-Brinkman equations at coarse-scale . . . 69
5.3 Stochastic upscaling . . . 70
5.3.1 Permeability . . . 70
5.3.2 Thermal conductivity . . . 71
5.4 Basis adaptation to find QoI . . . 72
5.4.1 Linear basis adaptation . . . 73
5.4.2 Quadratic basis adaptation . . . 73
5.4.3 Basis adaptation with KL expansion . . . 74
5.5 Results and discussion . . . 75

**Chapter 6 High resolution micrograph synthesis 81**
6.1 Introduction . . . 81
6.2 Problem statement . . . 85
6.3 Parametric texture model . . . 89
6.3.1 Steerable pyramid decomposition . . . 90
6.3.2 Parameter estimation sketch . . . 92
6.3.3 Micrograph synthesis procedure . . . 95
6.4 Density-based Monte Carlo filter . . . 97
6.5 Numerical illustration . . . 101
6.6 Appendix . . . 107

**Bibliography 109**

# List of Tables


3.1 Scaled solution time (# of iterations) vs stochastic dimension for random field with uniform random variables, diffusion equation, PC order = 4 and $\sigma$=0.1 . . . 32

3.2 Scaled solution time (# of iterations) vs order of polynomial chaos for random field with uniform random variables, diffusion equation, Stoch. dim=4, $\sigma = 0.1$ . 33

3.3 Scaled solution time (# of iterations) vs standard deviation ($\sigma$) of input random field, for random field with uniform random variables, diffusion equation, Stoch dim = 4 and PC order = 4 . . . . . 33

3.4 Scaled solution time (# of iterations) vs stochastic dimension for random field with uniform random variables, advection-diffusion equation, PC order = 4, and $\sigma$=0.1 . . . . . 34

3.5 Scaled solution time (# of iterations) vs order of polynomial chaos for random field with uniform random variables, advection-diffusion equation, Stoch. dim=4, $\sigma = 0.1$ . . . . . 35

3.6 Scaled solution time (# of iterations) vs standard deviation ($\sigma$) for random field with uniform random variables, advection-diffusion equation, Stoch dim = 4, PC order = 4 . . . . . 35

3.7 Scaled solution time (# of iterations) vs stochastic dimension for log-normal random field, diffusion equation, PC order = 4, and $\sigma$=0.1 . . . . . 36

3.8 Scaled solution time (# of iterations) vs order of polynomial chaos for log-normal random field, diffusion equation, Stoch. dim=4, $\sigma = 0.1$ . . . . . 36

3.9 Scaled solution time (# of iterations) vs standard deviation ($\sigma$) for log-normal random field, diffusion equation, Stoch dim = 4 and PC order = 4 . . . . . 37

3.10 Sacled solution time (# of iterations) vs stochastic dimension for log-normal random field, advection-diffusion equation, PC order = 4 and $\sigma$=0.1 . . . . . 37

3.11 Scaled solution time (# of iterations) vs order of polynomial chaos for log-normal random field, advection-diffusion equation, Stoch. dim=4, $\sigma = 0.1$ . . . . . 38

3.12 Scaled solution time (# of iterations) vs standard deviation ($\sigma$) for log-normal random field, advection-diffusion equation, Stoch dim = 4 and PC order = 4 . . . 38

# List of Figures


1.1 Fine scale domain modeled with an array of solid discs and coarse scale domain modeled as porous media . . . . . 3

3.1 Scaled Galerkin solution time vs solution error for varying polynomial chaos order and spatial mesh for the diffusion problem with uniform random field . . 30

3.2 Relative residual norm vs iteration countto solve stochastic diffusion equation . 31

3.3 Relative residual norm vs iteration count to solve stochastic advection-diffusion equation . . . . . 31

4.1 Probability density function of $h(\boldsymbol{\xi})$. Results using a full 10-dimensional solution and the reduced approximations using prior knowledge on the Gaussian content and quadratic content of QoI. . . . . 54

4.2 Elastic domain with random Young's modulus and subjected to uniform internal pressure. Point $A$ (0.2977,0.1417) is highlighted. . . . . 56

4.3 Probability density function of $x$–displacement at point $A$. Results using a 1) full 10-dimensional solution and the approximations using the 2) dominant component in the Karhunen-Loeve expansion of the stochastic process associated with the solution, 3) the 1-d approximation associated with Gaussian prior knowledge, and 4) sequence of 10 1-d approximations associated with quadratic prior knowledge. . . . . 58

4.4 Coefficients of the PC expansion of the $x$-displacement at point $A$. Results using a full 10-dimensional solution and the reduced approximation using prior knowledge on the quadratic content of QoI. . . . . 58

4.5 Probability density function of the maximum vonMises stress over the spatial domain of the problem. Results using 1) full 10-dimensional solution and the approximations using the 2) 1-d approximation associated with Gaussian prior knowledge, and 3) sequence of 10 1-d approximations associated with quadratic prior knowledge. . . . . 59

4.6 The first six coefficients of the elasticity modulus with respect to a polynomial chaos expansion in the original gaussian variables $\boldsymbol{\xi}$. . . . . 60

4.7 The first six coefficients of the elasticity modulus with respect to a polynomial chaos expansion in the one-dimensional gaussian variable $\eta_1$. . . . . 60

4.8 2-d cantilever beam meshed with quad4 elements, is fixed on left end and eigenvalue analysis is performed . . . . . 62

4.9 Wind turbine blade meshed with quad4 elements, is fixed in all 6 dof on one end and eigenvalue analysis is performed . . . . . 63

4.10 pdf of eigenvalues of 2-d cantilever beam in 6d $\xi$ and in 1d $\eta$ . . . . . 64

4.11 pdf of eigenvalues of wind turbine blade in 6d $\xi$ and in 1d $\eta$ . . . . . 65

4.12 Polynomial chaos coefficients of of $\bar{\phi}_j^{(l)}$ in 6d $\xi$ and in 1d $\eta$ . . . . . 65

5.1 Fine scale domain modeled with an array of solid discs and coarse scale domain modeled as porous media . . . . . 68

5.2 Fluid and solid regions meshed with 2-d quad4 elements . . . . . 76

5.3 Realizations of the thermal conductivity in the circular discs . . . . . 76

5.4 Mean and std. dev of velocity in x-direction in 10d $\xi$ and order 3 . . . . . 77

5.5 Mean and std. dev of velocity in y-direction in 10d $\xi$ and order 3 . . . . . 77

5.6 Mean and std. dev of pressure in 10d $\xi$ and order 3 . . . . . 77

5.7 Mean and std. dev of temperature in 10d $\xi$ and order 3 . . . . . 77

5.8 Mean and std. dev. of permeability component $k_{11}$ using volume averaging . . 78

5.9 Eigenevalues obianed from KL expansion of thermeal conductivity at finescale and the eigen values obtained from KL analysis of Gaussian part of permeability at coarse scale . . . . . 78

5.10 Permeability computed at points 3 and 20, using linear and quadratic basis adaptation methods. Here, the basis adapted for each spatial point (QoI) separately . 79

5.11 Permeability computed at points 3 and 20 using basis adaptation method, where new random variables $\eta$ are obtained from Karhunen-Loeve analysis of the Gaussian part of the random filed. Here, basis is adapted for all spatial points (QoI) simultaneously. . . . . 79

5.12 Thermal conductivity computed at points 20 using basis adaptation method, where new random variables $\eta$ are obtained from Karhunen-Loeve analysis of the Gaussian part of the random filed. Here, basis is adapted for all spatial points (QoI) simultaneously. . . . . 80

6.1 **Low resolution micrograph** Low resolution, low magnification digital image of cast aluminum alloy W319, for which high resolution images are available at points **A** and **B** . . . . . 85

6.2 **High resolution digital image of cast aluminum alloy W319 at A and B** . . . 86

6.3 **Low resolution digital image of cast aluminum alloy W319 at A, B and C** . 87

6.4 Steerable pyramid diagram Block diagram for the steerable pyramid Portilla and Simoncelli (2000), Simoncelli and Freeman (1995). Input image is initially split into high-pass and low-pass bands. Low-pass band is further decomposed into a set of oriented subbands and low-pass residual band. The recursive decomposition is achieved by inserting the diagram in dotted lines into the solid circle. . . . . 93

6.5 Real parts of subbands Real parts of steerable pyramid orientation bands(4 orientations and 4 scales) and low-pass subband of image at location A. High-Pass subband is not shown here. . . . . 93

6.6 **Block diagram of the recursive micrograph synthesis procedure** . . . . . 97

6.7 Block diagram of synthesis procedure with particle filter. Block diagram of the recursive micrograph synthesis procedure along with the density-based Monte Carlo filter. . . . . 102

6.8 Micrograph at A synthesized without particle filter. Micrographs from left to right show respectively, the experimental high resolution micrograph, synthesized micrographs without particle filter which are statistically equivalent with high resolution micrograph at location **A**. . . . . 102

6.9 Micrograph at A synthesized with particle filter. From 6.9(**a**) to 6.9(**c**), experimental high resolution micrograph, experimental low resolution micrograph and high resolution micrograph synthesized with texture model and particle filter at location A. . . . . 103

6.10 Micrograph at B synthesized without particle filter. Micrographs from left to right show respectively, the experimental high resolution micrograph, synthesized micrographs without particle filter which are statistically equivalent with high resolution micrograph at location **B**. . . . . 103

6.11 Micrograph at B synthesized with particle filter. From6.11**(a)** to 6.11**(c)**, experimental high resolution micrograph, experimental low resolution micrograph and high resolution micrograph synthesized with texture model and particle filter at location B. . . . . 104

6.12 Micrograph at C synthesized without particle filter. Micrographs from left to right show respectively, the experimental high resolution micrograph, synthesized micrographs without particle filter which are statistically equivalent with high resolution micrograph at location **C**. High resolution micrograph at location **C** is used only for comparison and is not used in the synthesis process. . . 104

6.13 Micrograph at C synthesized with particle filter. From 6.13**(a)** to 6.13**(c)**, experimental high resolution micrograph, experimental low resolution micrograph and high resolution micrograph synthesized with texture model and particle filter at location **C**. High resolution micrograph at location **C** is used only for comparison and is not used in the synthesis process. . . . . 105

# Abstract

This dissertation focuses on facilitating the analysis of probabilistic models for physical systems. To that end, novel contributions are made to various aspects of the problem, namely, 1) development of efficient algorithms for solving stochastic system of equations governing the physical problem, 2) stochastic basis adaptation methods to compute the solution in reduced stochastic dimensional space and 3) stochastic upscaling methods to find coarse-scale models from the fine-scale stochastic solutions. In particular, algorithms are developed for stochastic systems that are governed by partial differential equations (PDEs) with random coefficients. Polynomial chaos-based stochastic Galerkin and stochastic collocation methods are employed for solving these equations. Solvers and preconditioners based on Gauss-Seidel and Jacobi algorithms are explored for solving system of linear equations arising from stochastic Galerkin discretization of PDEs with random input data. Gauss-Seidel and Jacobi algorithms are formulated such that the existing software is leveraged in the computational effort. These algorithms are also used to develop preconditioners to Krylov iterative methods. These solvers and preconditioners are tested by solving a steady state diffusion equation and a steady state advection-diffusion equation. Upon discretization, the former PDE results in a symmetric positive definite matrix on left-hand-side, whereas the latter results in a non-symmetric positive definite matrix. The stochastic systems face significant computational challenge due the curse of dimensionality as the solution often lives in very high dimensional space. This challenge is addressed in the present work by recognizing the low dimensional structure of many quantities of interest (QoI) even in problems that have been embedded, via parameterization, in very high-dimensional settings. A new method for the characterization of subspaces associated with low-dimensional QoI is presented here. The probability

density function of these QoI is found to be concentrated around one-dimensional subspaces for which projection operators are developed. This approach builds on the properties of Gaussian Hilbert spaces and associated tensor product spaces.

For many physical problems, the solution lives in multiple scales, and it is important to capture the physics at all scales. To address this issue, a stochastic upscaling methodology is developed in which the above developed algorithms and basis adaptation methods are used. In particular upscaling methodology is demonstrated by developing a coarse scale stochastic porous medium model that replaces a fine-scale which consists of flow past fixed solid inclusions. The inclusions have stochastic spatially varying thermal conductivities and generate heat that is transported by the fluid. The permeability and conductivity of the effective porous medium are constructed as statistically dependent stochastic processes that are both explicitly dependent on the fine scale random conductivity.

Another contribution of this thesis is development of a probabilistic framework for synthesizing high resolution micrographs from low resolution ones using a parametric texture model and a particle filter. Information contained in high resolution micrographs is relevant to the accurate prediction of microstructural behavior and the nucleation of instabilities. As these micrographs may be tedious and uneconomical to obtain over an extended spatial domain, A statistical approach is proposed for interpolating fine details over a whole computational domain starting with a low resolution prior and high resolution micrographs available only at a few spatial locations. As a first step, a small set of high resolution micrographs are decomposed into a set of multi-scale and multi-orientation subbands using a complex wavelet transform. Parameters of a texture model are computed as the joint statistics of the decomposed subbands. The synthesis algorithm then generates random micrographs satisfying the parameters of the texture model by recursively

updating the gray level values of the pixels in the input micrograph. A density-based Monte Carlo filter is used at each step of the recursion to update the generated micrograph, using a low resolution micrograph at that location as a measurement. The process is continued until the synthesized micrograph has the same statistics as those from the high resolution micrographs. The proposed method combines a texture synthesis procedure with a particle filter and produces good quality high resolution micrographs.

# Chapter 1

# Introduction

## 1.1 Motivation

Many physical problems can be described with coupled stochastic nonlinear partial differential equations. One such problem of interest is fluid flow through porous media coupled with heat equation. In this work a collection of two-dimensional solid discs is considered to be micro porous medium and the flow through pores at micro level can be modeled with Navier-Stokes equations coupled with heat equation. Due to inherent randomness in this physical system, it is modeled with uncertainty. Simulating such a system only at fine scale with Navier-Stokes equations can be computationally intractable. Hence, multi-scale modeling approach is employed where, coarse scale equations are modeled with Darcy-Brinkman equations. To avoid the complexity due to turbulence, the simulations are performed at a low Reynolds number, so that the flow is laminar. The coupling between the fluid flow and heat transfer is achieved through temperature dependent viscosity and fluid density. The multi-scale treatment of the problem makes the simulation numerically feasible but one has to be careful to maintain sufficient accuracy while making the multi-scale approximation. The main objective of this research is to come up with

stochastic upscaling approach so that the coarse scale porous media model can simulate the solution with required accuracy. Since the coarse scale models are much smaller compared to the fine-scale models, they are computationally less expensive to solve.

Another problem considered in this work is probabilistic treatment of high resolution micrograph synthesis from low resolution ones using a parametric texture model and a particle filter. High resolution microstructural information is required as a prerequisite for the multi-scale modeling and analysis of composite structures, and in particular to track the nucleation of instabilities in material behavior. As it remains expensive to experimentally obtain high resolution micrographs throughout a computational domain where coarse-scale behavior is being assessed, procedures for gleaning equivalent information from lower resolution images have demonstrated their value in computational materials science Ghosh et al. (2006), Valiveti and Ghosh (2007).

## 1.2 Problem statement

In this work, stochastic upscaling methodology is developed to model the flow through porous media coupled with heat equations. At fine scale, fluid flow through an array of solid circular discs is modeled with Navier-Stokes equations Betchen et al. (2006), Chandesris et al. (2006), Nakayama et al. (2002) coupled with heat equation to model heat transfer between solid and fluid domain.

Randomness in the the system is modeled by treating thermal conductivity of the solid region as a random filed. Spectral stochastic finite element methods are used to solve these stochastic partial differential equations. In these methods random coefficients and the solution are represented

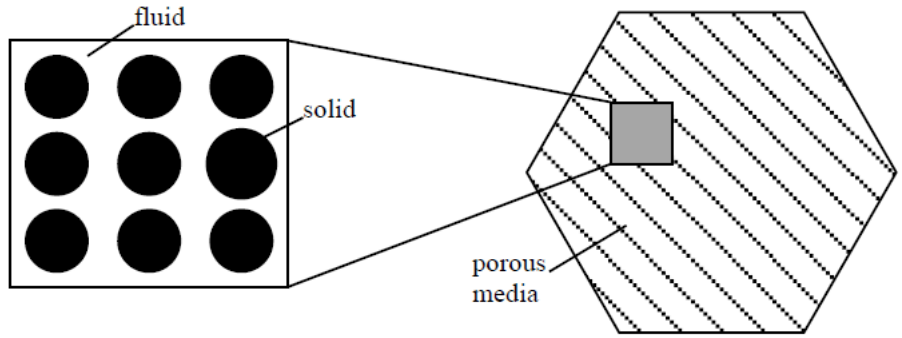



Figure 1.1: Fine scale domain modeled with an array of solid discs and coarse scale domain modeled as porous media

as series expansion of polynomial chaos basis, where, polynomial chaos basis are orthogonal polynomials in terms of input random variables.

It is computationally intractable to solve Navier-Stokes equations with random parameters at the fine scale. To overcome this difficulty, an approximate porous media model is proposed at the coarse scale that will compute the solution of interest accurately. The equations governing the flow at coarse scale are modeled with Darcy-Brinkman equations Schmidt et al. (2010). A stochastic upscaling procedure is followed to compute the parameters of the coarse scale model. The parameters of the coarse scale model that are to be computed from the stochastic upscaling method are permeability and thermal conductivity random fields at the coarse scale. Figure 1.1 shows fine scale and coarse scale domain modeled in 2-dimensions.

To synthesize high resolution micrographs, from low resolution ones, it is shown in chapter 6 that, a parametric texture model based in wavelet transformation and a particle filter can be used to accurately simulate the microstructural feature of the local domain. Starting with an image $x_0$ that represents the low-resolution micrograph at a location, our aim is to synthesize an image $x$ at the same location, representing the high-resolution micrograph, that is constrained, in some

sense, by our knowledge of experimental high resolution data available at few locations and low resolution data available at the point of interest. In order to complete the statement of the problem, we must specify the nature of the constraints (i.e. the specific functionals or features that describe our knowledge of the high resolution data). In what follows, a texture model will be used to provide a context in which features of a micrograph are quantified. This model will be constructed from available high resolution data and presumed to be valid everywhere else. Specifically, the texture model will be identified with the joint statistics of the subbands in a particular wavelet decomposition, namely, a steerable pyramid decomposition. Associated with this model is a mapping $f$ that maps $x_0$ into $x$ such that $x$ has the joint statistics corresponding to the model. In our work, we use a Bayesian framework where the experimental low resolution micrograph play the role of measurement used to update a prior model synthesized from a very small set of high resolution micrographs. We use a particle filter, namely a density-based Monte Carlo filter to implement the Bayesian formulation.

## 1.3 Stochastic upscaling

Objective of stochastic upscaling in this work is to identify the coarse scale models that capture the quantities of interest with good accuracy. The parameters of the coarse scale models are computed using volume average method. Further, stochastic basis adaptation methods are developed to represent the coarse scale parameters in the reduced stochastic dimension.

## 1.4 Challenges

There are many modeling, numerical and computational challenges involved in this problem. Significant modeling challenges in this problem are to choose the right governing equations and boundary conditions in both fine and coarse scale problems, to choose the spatial dependence of the coarse scale parameters and to come up with the appropriate distance measure between the quantities of interest at fine and coarse scales. Other modeling challenge is in representing the uncertainty at both the scales.

The numerical challenges involved are instabilities in the finite element formulations of incompressible flows. Main sources of these instabilities are spurious node to node oscillations in the velocity field in the advection dominated flow and spurious pressure modes in Navier-Stokes and Darcy-Brinkman equations due to equal order interpolation in pressure and velocity field in finite element discretizations. To address this issue, stabilization methods should be employed in the finite element discretization. In this work, stabilizations methods such as Streamline-upwind/Petrov-Galerkin (SUPG) and pressure-stabilizing/Petrov-Galerkin (PSPG) formulations Brooks and Hughes (1982), Shadid et al. (2006), Tezduyar (1992) are used.

There are many computational challenges in solving this problem. At fine scale and coarse scale, the governing equations are coupled stochastic nonlinear partial differential equations. In the optimizations process, these partial differential equations should be solved efficiently and accurately. The stochastic partial differential equations at the fine scale are Navier-Stokes equations, which are very complex to solve even in the deterministic case. Spectral stochastic Galerkin projections are used to discretize the stochastic partial differential equations. The linear algebraic equations arising from the discretization of these equations are very large in size and it is very

challenging to solve these equations efficiently. To address these issues, efficient solvers and preconditioners are proposed in chapter 3 based on Gauss-Seidel and Jacobi algorithms. To explain Gauss-Seidel and Jacobi algorithms let $Ax = b$ be a system of linear equations, where, $A = [a_{ij}], b = \{b_i\}$ and $x = \{x_j\}$ the Gauss-Seidel algorithms can be written as,

$$x_j^{(k+1)} = \frac{1}{a_{ij}} \left( b_i - \sum_{j>i} a_{ij} x_j^{(k)} - \sum_{j<i} a_{ij} x_j^{(k+1)} \right), \tag{1.1}$$

and the Jacobi algorithm can be written as

$$x_j^{(k+1)} = \frac{1}{a_{ij}} \left( b_i - \sum_{j \neq i} a_{ij} x_j^{(k)} \right), \tag{1.2}$$

These algorithms are adapted to solve the linear equations arising from spectral stochastic finite element discretization. The stochastic Galerkin system of equations for all the partial differential equations are formulated using Albany application software developed at Sandia National Labs. The solvers and preconditioners are incorporated as Stokhos package of Trilinos developed at Sandia National Labs.

Another computational challenge arises due to curse of dimensionality in stochastic Galerkin projections. As the number of input random variables (stochastic dimension) increases, the number of stochastic basis functions to represent the solution increases exponentially. Because of this curse of dimensionality, there is a limitation to go to larger stochastic dimensions. To address this issue by further reducing the size of the problem, dimension reduction methods based on basis adaptation discussed in chapter 4 will be used. In these methods, a new set of random variables in the reduced stochastic space is obtained b finding an isometry matrix $A$, the relates the new

set to the original set of random variables, $\boldsymbol{\eta} = A\boldsymbol{\xi}$. Here, $\boldsymbol{\xi} = \{\xi_1, \cdots, \xi_d\}^T$ are the original set of random variables and $\boldsymbol{\eta} = \{\eta_1, \cdots, \eta_l\}^T$ are the new set such that, $l << d$. The new set of random variables are adapted to specific quantities of interest (QoI) such that the solution of these QoI can be represented accurately with few random variables. Identifying the right set of random variables that are adapted to QoI is not unique. In chapter 4, various cases such as linear, quadratic basis adaptation is discussed. It is also shown that, if the QoI is scalar, it can be accurately represented in only on e random variable. Both, intrusive and non-intrusive methods can be employed for basis adaptation procedure.

This manuscript is organized as follows. Chapter 2 provides a brief introduction to stochastic Galerkin methods. Chapter 3 describes the solvers and preconditiners based on Gauss-Seidel and jacobi algorithms to solve stochastic Galerkin system of equations efficiently. In chapter 4, basis adaptation methods for dimension reduction is presented. Chapter 5 describes the stochastic upscaling methodology for a stochastic coupled nonlinear problem and shows the use of tools developed in chapters 3 and 4. Finally chapter 6 describes the probabilistic treatment of micrographs and their synthesis procedure from known statistical data.

# Chapter 2

# Stochastic Galerkin methods

## 2.1 Introduction

Real life physical problems are often modeled as partial differential equations (PDEs) where the coefficients of the partial differential equations are treated as random parameters to represent uncertainty in the problem. Monte Carlo techniques are popular methods to solve these problems as they only require solutions to the PDE for a given set of realizations of the input random coefficients. More recently however, the stochastic finite element methods Babuška et al. (2004), Ghanem and Spanos (1991) have become a popular choice for solving these problems because of their advantages over Monte Carlo methods. These methods compute statistical properties of the solution more efficiently than Monte Carlo methods when the number of sources of uncertainty is not too large and the solution exhibits some degree of regularity with respect to these sources.

## 2.2 Stochastic partial differential equations

Let $D$ be an open subset of $\mathbb{R}^n$ and $(\Omega, \Sigma, P)$ be a complete probability space with sample space $\Omega$, $\sigma$-algebra $\Sigma$ and probability measure $P$. We are interested in studying the following stochastic

partial differential equation: find a random field, $u(x, \omega) : D \times \Omega \to \mathbb{R}$ such that the following holds $P$-almost surely ($P$-a.s.):

$$\mathcal{L}(x, \omega; u) = f(x, \omega) \ \text{ in D} \times \Omega, \tag{2.1}$$

subject to the boundary condition

$$\mathcal{B}(x, \omega; u) = g(x, \omega) \ \text{ on } \partial\text{D} \times \Omega, \tag{2.2}$$

where $\mathcal{L}$ is a differential operator and $\mathcal{B}$ is a boundary operator.

## 2.3 Input random field model

Uncertainty in stochastic PDEs is often arises by treating coefficients in the differential operator, $\mathcal{L}(x, \omega; u)$, as a random fields. For computational purposes, each stochastic coefficient $a(x, \omega)$ must be discretized in both spatial and stochastic domains. To this end, it is often approximated with a truncated series expansion that separates the spatial variable $x$ from the stochastic variable $\omega$ resulting in a representation by a finite number of random variables. For this representation, second order information such as the covariance function of the random field is required. In the present problem, two cases of random field models are considered. In the first case, the random field is assumed to be uniformly distributed and is approximated through a truncated Karhunen-Loève expansion. In the second case, the random field is assumed to have a log-normal distribution, that is $a(x, \omega) = \exp(g(x, \omega))$ where $g(x, \omega)$ is a Gaussian random field, and is approximated by a truncated polynomial chaos expansion.

### 2.3.1 Karhunen-Loève expansion

Let $C(x_1, x_2) = E[a(x_1, \omega)a(x_2, \omega)]$ be the covariance function of the random field $a(x, \omega)$, where $E[\cdot]$ denotes mathematical expectation. Then $a(x, \omega)$ can be approximated through its truncated Karhunen-Loève (K-L) expansion Ghanem and Spanos (1991) given by

$$a(x, \omega) \approx \tilde{a}(x, \xi(\omega)) = a_0(x) + \sum_{i=1}^{M} \sqrt{\lambda_i} a_i(x) \xi_i(\omega), \tag{2.3}$$

where $a_0(x)$ is the mean of the random field $a(x, \omega)$ and $\{(\lambda_i, a_i(x))\}_{i \geq 1}$ are solutions of the integral eigenvalue problem

$$\int_D C(x_1, x_2) a_i(x_2) dx_2 = \lambda_i a_i(x_1). \tag{2.4}$$

The eigenvalues $\lambda_i$ are positive and non-increasing, and the eigenfunctions $a_i(x)$ are orthonormal, that is,

$$\int_D a_i(x) a_j(x) = \delta_{ij}, \tag{2.5}$$

where $\delta_{ij}$ is the Kronecker delta. In Eq. 2.3, $\{\xi_i\}_{i=1}^{M}$ are uncorrelated random variables with zero mean. As a first test-case, the diffusion coefficient $a(x, \omega)$ is modeled as a random field with following exponential covariance function

$$C(x_1, x_2) = \sigma^2 \exp(-\|x_1 - x_2\|_1 / L) \tag{2.6}$$

such that the random variables $\xi_i(\omega)$ in the K-L expansion are uniformly distributed. We further assume that the random variables are independent.

### 2.3.2 Polynomial chaos expansion

The K-L expansion approximates a random field by a linear combination of a finite set of random variables. To maintain positivity of the random field, such a representation is only appropriate if the random variables are bounded Ullmann (2010). For unbounded random variables (e.g., log-normal) a nonlinear polynomial chaos representation is more appropriate. The polynomial chaos expansion Ghanem and Spanos (1991), Weiner (1963) is used to approximate a random field in terms of multi-variate orthogonal polynomials. Let $\boldsymbol{\xi} = (\xi_1, \cdots, \xi_M)^T$ be the random variables from a truncated K-L expansion of a given random field $g(x, \omega)$, that is

$$g(x, \omega) \approx \tilde{g}(x, \boldsymbol{\xi}(\omega)) = g_0(x) + \sum_{i=1}^{M} \sqrt{\lambda_i} g_i(x) \xi_i(\omega). \tag{2.7}$$

Assume $a(x, \omega)$ is then given by a nonlinear transformation of $g(x, \omega)$. Then $a(x, \omega)$ can be represented through nonlinear functionals of the random variables $\xi_i(\omega)$. It has been shown in Ghanem and Spanos (1991), Weiner (1963) that this functional dependence can be expanded in terms of multi-dimensional orthogonal polynomials, called polynomial chaos, as

$$a(x, \omega) = \hat{a}_0(x) + \sum_{i_1=1}^{\infty} \hat{a}_{i_1}(x) \Gamma_1(\xi_{i_1}(\omega)) + \sum_{i_1=1}^{\infty} \sum_{i_2=1}^{i_1} \hat{a}_{i_1 i_2} \Gamma_2(\xi_{i_1}(\omega), \xi_{i_2}(\omega)) + \cdots \tag{2.8}$$

where $\Gamma_n(\xi_{i_1}, \cdots, \xi_{i_n})$ is the multi-dimensional polynomial chaos of order $n$ in random variables $(\xi_{i_1}, \cdots, \xi_{i_n})$. A one-to-one mapping of polynomials $\{\Gamma_i\}$ to a set of polynomials with ordered indices $\{\psi_i(\boldsymbol{\xi})\}$ can be introduced Ghanem and Spanos (1991). After substituting $\{\psi_i\}$ in Eq. 2.8

and truncating the series to finite number of terms $N_\xi$, the random field $a(x,\omega)$ can thus be approximated as

$$a(x,\omega) \approx \tilde{a}(x,\boldsymbol{\xi}(\omega)) = a_0(x) + \sum_{i=1}^{N_\xi} a_i(x)\psi_i(\boldsymbol{\xi}). \tag{2.9}$$

The polynomials $\{\psi_i(\boldsymbol{\xi})\}$ are orthogonal with respect to the inner product defined by expectation in the stochastic space,

$$\langle \psi_i(\boldsymbol{\xi}), \psi_j(\boldsymbol{\xi})\rangle \equiv \int_\Omega \psi_i(\boldsymbol{\xi}(\omega))\psi_j(\boldsymbol{\xi}(\omega))dP(\omega) = \delta_{ij}. \tag{2.10}$$

As a second test-case, the diffusion coefficient $a(x,\omega)$, is modeled as a log-normal random field Ghanem (1999) where $a(x,\omega) = \exp[g(x,\omega)]$ and $g(x,\omega)$ is a Gaussian random field with exponential covariance (2.6). Here, $g(x,\omega)$ is approximated with a truncated K-L expansion (2.7). In this case the random variables $\xi_i$ are standard normal random variables and thus are independent and $a(x,\omega)$ can be approximated with truncated polynomial chaos expansion (2.9). It also can be shown that the polynomials $\{\psi_i\}$ are tensor products of one-dimensional Hermite polynomials. For a given total polynomial order $p$, the total number of polynomials $\{\psi_i(\boldsymbol{\xi})\}$ is $N_\xi + 1 = \frac{(M+p)!}{M!p!}$.

## 2.4 Stochastic Galerkin method

Let $H_0^1(D)$ be the subspace of the Sobolev space $H^1(D)$ that vanishes on the boundary $\partial D$ and is equipped with the norm $\|u\|_{H_0^1(D)} = [\int_D |\nabla u|^2 dx]^{\frac{1}{2}}$. Problem (2.1) can then be written in the

following equivalent variational form Ghanem and Doostan (2006): find $u \in H_0^1(D) \otimes L_2(\Omega)$ such that

$$b(u, v) = l(v), \quad \forall v \in H_0^1(D) \otimes L_2(\Omega), \tag{2.11}$$

where $b(u, v)$ is a continuous and coercive bilinear form and $l(v)$ is a continuous bounded linear functional. In the stochastic Galerkin method, we seek the solution of the variational problem (2.11) in a tensor product space $X_h \otimes Y_p$, where, $X_h \subset H_0^1(D)$ is finite dimensional space of continuous polynomials corresponding to the spatial discretization of $D$ and $Y_p \subset L_2(\Omega)$ is the space of random variables spanned by polynomial chaos Ghanem and Spanos (1991) of order up to $p$. Then the finite dimensional approximation $u_{X_h Y_p}(x, \omega)$ of the exact solution $u(x, \omega)$ on the tensor product space $X_h \otimes Y_p$ is given as the solution to

$$b(u_{X_h Y_p}, v) = l(v) \quad \forall v \in X_h \otimes Y_p. \tag{2.12}$$

In equation (2.12) the input random field $a(x, \omega)$ in the bilinear form $b(u_{X_h Y_p}, v)$ can be approximated using either a K-L expansion or a polynomial chaos expansion depending on a choice of model for the random field. The finite dimensional approximation $u_{X_h Y_p}(x, \omega)$ is represented with truncated polynomial chaos expansion, where the multidimensional polynomial chaos are orthogonal with respect to the probability measure of the underlying random variables. The resulting set of coupled PDEs are then discretized using standard techniques such as the finite element or finite difference methods. In the present problem, the set of coupled PDEs are discretized using the finite element method and the resulting system of linear equations can be written as

$$\sum_{j=0}^{N_\xi}\sum_{i=0}^{\hat{P}} c_{ijk}K_i u_j = f_k, \quad k = 0, \cdots, N_\xi, \tag{2.13}$$

where $f_k = E\{f(x, \boldsymbol{\xi})\psi_k\}$, $c_{ijk} = E\{\xi_i\psi_j\psi_k\}$ and $\hat{P} = M$ when $a(x, \omega)$ is approximated by a truncated K-L expansion, or $c_{ijk} = E\{\psi_i\psi_j\psi_k\}$ and $\hat{P} = \hat{N}_\xi$ when $a(x, \omega)$ is approximated by a polynomial chaos expansion. Here $\{K_i \in \mathbb{R}^{N_x \times N_x}\}_{i=0}^{\hat{P}}$ are the polynomial chaos coefficients of the stiffness matrix (section (3.4) of Powell and Elman (2009)) and $\{u_j \in \mathbb{R}^{N_x}\}_{j=0}^{N_\xi}$ are the polynomial chaos coefficients of the discrete solution vector

$$u_j = [u_{0j}, \ldots, u_{N_x j}]^T, \quad j = 0, \ldots, N_\xi. \tag{2.14}$$

Equation (2.13) can be written in the form of a global stochastic stiffness matrix of size $((N_\xi + 1) \times N_x)$ by $((N_\xi + 1) \times N_x)$ as

$$\begin{bmatrix} K^{0,0} & K^{0,1} & \cdots & K^{0,N_\xi} \\ K^{1,0} & K^{1,1} & \cdots & K^{1,N_\xi} \\ \vdots & \vdots & \vdots & \vdots \\ K^{N_\xi,0} & K^{N_\xi,1} & \cdots & K^{N_\xi,N_\xi} \end{bmatrix} \times \begin{Bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{N_\xi} \end{Bmatrix} = \begin{Bmatrix} f_1 \\ f_2 \\ \vdots \\ f_{N_\xi} \end{Bmatrix} \tag{2.15}$$

where $K^{j,k} = \sum_{i=0}^{\hat{P}} c_{ijk}K_i$. We will denote this system as $\bar{K}\bar{u} = \bar{f}$. In practice it is prohibitive to assemble and store the global stochastic stiffness matrix in this form, rather each block of the stochastic stiffness matrix can be computed from the $\{K_i\}$ when needed. The stochastic stiffness matrix is block sparse, which means that some of the off-diagonal blocks are zero matrices, because of the fact that the $c_{ijk}$ defined above vanishes for certain combination of $i, j$ and $k$.

It is also interesting to note that the diagonal blocks are dominant over the off-diagonal blocks because, the diagonal blocks have the contribution from mean stiffness matrix $K_0$, whereas the off-diagonal blocks do not have the contribution from mean stiffness matrix. These properties of block sparsity and diagonal dominance are exploited to develop solvers and preconditioners based on Jacobi and Gauss-Seidel algorithms.

# Chapter 3

# Iterative solvers and preconditioners

## 3.1 Introduction

The linear algebraic equations arising from the discretization of these equations are very large in size and it is very challenging to solve these equations efficiently. In this work solvers and preconditioners based on Jacobi and Gauss-Seidel algorithms are developed to solve the system of equations arising from the stochastic finite element methods.

## 3.2 Solution methods for stochastic Galerkin systems

In this section, various solver techniques and preconditioning methods for solving the linear algebraic equations arising from stochastic Galerkin discretizations (2.13) are described. The solver methods discussed are: relaxation methods, namely, a Jacobi mean method and a Gauss-Seidel mean method, and Krylov-based iterative methods Saad (1996). Also various stochastic preconditioners used to accelerate convergence of the Krylov methods are discussed, including mean-based Powell and Elman (2009), Gauss-Seidel mean, approximate Gauss-Seidel mean, approximate Jacobi mean and Kronecker product Ullmann (2010) preconditioners. The relaxation

schemes can be viewed as fixed point iterations on a preconditioned system Saad (1996). In Jacobi and Gauss-Seidel methods, mean splitting is used rather than traditional diagonal block splitting as it allows use of the same mean matrix $K_0$ for all inner deterministic solves (and thus reuse of the preconditioner $P_0 \approx K_0$).

### 3.2.1 Jacobi mean algorithm

In this method, systems of equations of size equal to that of the deterministic system are solved iteratively by updating the right-hand-side to obtain the solution to the stochastic Galerkin system of equations (2.13):

$$c_{kk0} K_0 u_k^{new} = f_k - \sum_{j=0}^{N_\xi} \sum_{i=1}^{\hat{P}} c_{ijk} K_i u_j^{old}, \quad k = 0, \cdots, N_\xi. \tag{3.1}$$

The above system of equations are solved for $k = 0, \cdots, N_\xi$ using any solution technique appropriate for the mean matrix $K_0$. Thus existing legacy software can be used with minimal modification to solve the stochastic Galerkin system. In this work, Krylov-based iterative methods with appropriate preconditioners will be used. One cycle of solves from $k = 0, \cdots, N_\xi$ is considered one Jacobi outer iteration, and after each outer iteration, the right-hand-side in equation (3.1) is updated replacing $\{u_j^{old}\}$ with the new solution $\{u_j^{new}\}$. These outer iterations are continued until the required convergence tolerance is achieved. The Jacobi mean algorithm is shown in Algorithm 1.

**Algorithm 1** Jacobi mean algorithm

1. Choose initial guess $\bar{u}^0$ and compute residual $\bar{r} = \bar{K}\bar{u}^0 - \bar{f}$

2. Iteration count, $itr = 0$

3. **while** $\frac{\|\bar{r}\|_2}{\|\bar{f}\|_2} > \texttt{tol}$ **do**

4. **for** $k = 0 \ldots N_\xi$ **do**

5. Solve $c_{kk0}K_0 u_k^{(itr+1)} = f_k - \sum_{j=0}^{N_\xi}\sum_{i=1}^{\hat{P}} c_{ijk}K_i u_j^{(itr)}$

6. **end for**

7. $itr = itr + 1$

8. $\bar{r} = \bar{K}\bar{u}^{itr} - \bar{f}$

9. **end while**.

Note that for a given outer iteration, all of the right-hand-sides for $k = 0, \cdots, N_\xi$ are available simultaneously, and thus their solution can be efficiently parallelized. Moreover block algorithms optimized for multiple right-hand-sides may be used to further increase performance. Finally this approach does not require a large amount of memory to compute the solution. The disadvantage of the method is that it may not converge or may converge very slowly when the diagonal blocks of the stochastic stiffness matrix are less dominant over off-diagonal blocks.

### 3.2.2 Gauss-Seidel mean iterative method

The Gauss-Seidel method considered is similar to the Jacobi method, except the right-hand-side in equation (3.1) is updated after each deterministic solve with the newly computed $u_k^{new}$. Symbolically this is written

$$c_{kk0}K_0 u_k^{new} = f_k - \sum_{j=0}^{k-1}\sum_{i=1}^{\hat{P}} c_{ijk}K_i u_j^{new} - \sum_{j=k}^{N_\xi}\sum_{i=1}^{\hat{P}} c_{ijk}K_i u_j^{old}, \quad k = 0, \cdots, N_\xi. \tag{3.2}$$

As before, one cycle of solves from $k = 0, \cdots, N_\xi$ is considered one outer iteration of the Gauss-Seidel method, and these outer iterations are repeated until the required convergence tolerance is achieved. Note however that computing the updates as shown here would result in a large number of duplicated matrix-vector products $K_i u_j$ for each outer iteration. Instead, after each $u_k^{new}$ is computed by solving the mean linear system, we first compute $y = K_i u_k^{new}$ for all $i$ in which $c_{ijk}$ is nonzero for any $j$. Then for each corresponding $j$ we update $f_j \leftarrow f_j - c_{ijk} y$. This allows all of the right-hand-sides to be updated as required using the fewest number of matrix-vector products and without resorting to storing intermediate products. The complete Gauss-Seidel algorithm is shown in Algorithm 2.

**Algorithm 2** Gauss-Seidel mean algorithm

1. Choose initial guess $\bar{u}^0$ and compute residual $\bar{r} = \bar{f} - \bar{K}\bar{u}^0$

2. Iteration count, $itr = 0$

3. Initialize $\bar{z} = \bar{r}$

4. **while** $\frac{\|\bar{r}\|_2}{\|\bar{f}\|_2} > \texttt{tol}$ **do**

5. $\quad \bar{r} = \bar{f}$

6. $\quad$ **for** $k = 0 \ldots N_\xi$ **do**

7. $\quad\quad$ Solve $c_{kk0} K_0 u_k = z_k$

8. $\quad\quad z_k = f_k$

9. $\quad\quad$ **for** $i = 1, \ldots, \hat{P}$

10. $\quad\quad\quad y = K_i u_k$

11. $\quad\quad\quad$ **for** $j = 1, \ldots, N_\xi$

12. $\quad\quad\quad\quad$ **if** $c_{ijk} \neq 0$ **then**

13. $\quad\quad\quad\quad\quad z_j = z_j - c_{ijk} y$

14. $\quad\quad\quad\quad\quad r_j = r_j - c_{ijk} y$

15. $\quad\quad\quad\quad$ **endif**

16. $\quad\quad\quad$ **end for**

17. $\quad\quad$ **end for**

18. $\quad\quad r_k = r_k - c_{kk0} K_0 u_k$

19. $\quad$ **end for**

20. $\quad itr = itr + 1$

21. **end while**.

Often this method converges in fewer iterations than the Jacobi method, at the expense of no longer having all of the right-hand-sides available simultaneously. Unlike diagonal block splitting methods defined in Rosseel and Vandewalle (2010), mean block splitting is used in both Jacobi and Gauss-Seidel algorithms and hence the left-hand-side matrix is the mean matrix for all inner deterministic problems and only the right-hand-side changes. In such cases recycled Krylov basis methods could be explored to increase performance.

### 3.2.3 Krylov based iterative methods with matrix-free operations

Krylov based iterative methods Saad (1996) such as the conjugate gradient (CG) method and the generalized minimal residual (GMRES) method can be used to solve the stochastic Galerkin system (2.13) in which matrix vector products $\bar{v} = \bar{K}\bar{u}$ are computed using "matrix-free" operations:

$$v_k = \sum_{j=0}^{N_\xi} \sum_{i=0}^{\hat{P}} c_{ijk} K_i u_j, \quad k = 0, \cdots, N_\xi. \tag{3.3}$$

If the matrix vector products are computed from Eq. 3.3, it is not required to assemble the full stochastic Galerkin stiffness matrix, drastically decreasing memory requirements. However if a large number of iterations of a Krylov method such as GMRES are required, allocation of the Krylov basis may still require a very large amount of memory. Thus good preconditioning strategies for the stochastic Galerkin system are required, several of which will be discussed below.

### Mean-based preconditioner

The mean-based preconditioner Powell and Elman (2009) is given by $P = \mathrm{diag}\{\mathrm{P}_0, \cdots, \mathrm{P}_0\}$ where $P_0 \approx K_0$ is a preconditioner for the mean. The mean-based preconditioner is very efficient to compute and apply, since it only must be generated once from a matrix that is of the size of the deterministic system. However it doesn't incorporate any higher-order stochastic information, thus its performance degrades as the stochastic dimension, polynomial order, or random field variance increases Ullmann (2010).

### Gauss-Seidel preconditioner

One or more outer iterations of the Gauss-Seidel mean algorithm can be used as a preconditioner to the Krylov based iterative methods. An advantage of this method is that the cost of applying the preconditioner can be controlled by adjusting the tolerance of the inner deterministic solves and number of outer iterations. Decreasing this tolerance and increasing the number of outer iterations will reduce the number of iterations in the Krylov method, but make the preconditioner more expensive to apply, and thus these must be balanced to minimize overall computational cost. Generally we have found the cost of the preconditioner to be dominated by solving the mean systems, and thus the performance was improved by loosening the outer solver tolerance or limiting the number of outer iterations. In the results presented below we limited the preconditioner is limited to only one Gauss-Seidel iteration.

### Approximate Gauss-Seidel preconditioner

The process of increasing the inner solver tolerance can be taken to its extreme of replacing the inner mean solves by application of the mean preconditioner. As with the Gauss-Seidel preconditioner above, we found experimentally that this approach worked best with only one Gauss-Seidel iteration, and adding additional iterations did not improve the quality of the preconditioner. We also found that the cost of the preconditioner was reduced dramatically if only the first-order terms in the expansion for the stiffness matrix were used in the preconditioner and using higher-order terms did not improve performance. It is referred as the approximate Gauss-Seidel preconditioner here.

### Approximate Jacobi preconditioner

Similar to the approximate Gauss-Seidel preconditioner, Jacobi iterations can be used using a preconditioner in place of the mean stiffness matrix. In this case we used two outer Jacobi iterations, since the first iteration is equivalent to mean-based preconditioning (i.e., the additional terms on the right-hand-side of equation (3.1) are zero). Increasing the number of outer iterations did not improve the efficiency of the overall solver. It is referred as the approximate Jacobi preconditioner. Both approximate Gauss-Seidel and approximate Jacobi preconditioners are found to be very effective in reducing the number of Krylov iterations and are also not very expensive to apply.

**Kronecker product preconditioner**

The Kronecker product preconditioner Ullmann (2010) is defined as $P_1 = G \otimes K_0$, where $K_0$ is mean stiffness matrix and $G$ is

$$G = \sum_{i=0}^{\hat{P}} \frac{\text{tr}(K_i^T K_0)}{\text{tr}(K_0^T K_0)} G_i \tag{3.4}$$

where, $G_i(j,k) = c_{ijk}$. Unlike, the mean-based preconditioner, the Kronecker product preconditioner incorporates higher order stochastic information and hence the Krylov based algorithm converges in fewer iterations. However the disadvantage is that the cost of constructing the Kronecker product preconditioner is larger than that of mean-based preconditioner, and it is also more expensive to apply. Over all, solution time is found to be less than that with mean-based preconditioner as reported in Ullmann (2010). However, it was found in the numerical experiments that the approximate Gauss-Seidel and approximate Jacobi preconditioner performed better than the Kronecker product preconditioner.

## 3.3 Problem Statement

In this work a stochastic steady-state elliptic diffusion equation with Dirichlet boundary conditions Elman et al. (2011) is used as test problem for various stochastic PDE solution methods. Assume $a(x,\omega) : D \times \Omega \to \mathbb{R}$ is a random field that is bounded and strictly positive, that is,

$$0 < a_l \leq a(x,\omega) \leq a_u < \infty \quad \text{a.e.} \quad \text{in} \quad \text{D} \times \Omega. \tag{3.5}$$

For the steady-state diffusion equation, we wish to compute a random field $u(x,\omega) : D \times \Omega \to \mathbb{R}$, such that the following holds $P$-almost surely ($P$-a.s.):

$$-\nabla.(a(x,\omega)\nabla u(x,\omega)) = f(x,\omega) \ \text{ in D} \times \Omega, \tag{3.6}$$

$$u(x,\omega) = u_0 \ \text{ on } \partial\text{D} \times \Omega. \tag{3.7}$$

Problem (3.6) can then be written in the following equivalent variational form Ghanem and Doostan (2006): find $u \in H_0^1(D) \otimes L_2(\Omega)$ such that

$$b(u,v) = l(v), \quad \forall v \in H_0^1(D) \otimes L_2(\Omega), \tag{3.8}$$

where $b(u,v)$ is the continuous and coercive (from assumption (3.5)) bilinear form given by

$$b(u,v) = E\left[\int_D a\nabla u \cdot \nabla v dx\right], \quad \forall u,v \in H_0^1(D) \otimes L_2(\Omega). \tag{3.9}$$

and $l(v)$ is the continuous bounded linear functional given by

$$l(v) = E\left[\int_D f v dx\right], \quad \forall v \in H_0^1(D) \otimes L_2(\Omega). \tag{3.10}$$

From the Lax-Milgram lemma, equation (3.8) has unique a solution in $H_0^1(D) \otimes L_2(\Omega)$.

## 3.4 Sparse grid collocation method

In the collocation method, the solution to the PDE is sampled at a pre-selected set of points called collocation points, $\Theta = (\boldsymbol{\xi}^{(1)}, \cdots, \boldsymbol{\xi}^{(N)})$. The stochastic solution is constructed by interpolating at these collocation points,

$$u(x, \boldsymbol{\xi}) \approx \sum_{k=0}^{N} u_k(x) L_k(\boldsymbol{\xi}), \tag{3.11}$$

where $\{L_k(\boldsymbol{\xi})\}$ are Lagrange interpolatory polynomials defined by $(L_k(\boldsymbol{\xi}_l) = \delta_{kl})$ and $u_k$ is the solution of following deterministic diffusion equation,

$$-\nabla.(a(x, \boldsymbol{\xi}^{(k)}) \nabla u_k(x) = f(x, \boldsymbol{\xi}^{(k)}) \ \text{ in D}, \tag{3.12}$$

$$u_k(x) = u_{k0} \ \text{ on } \partial\text{D}. \tag{3.13}$$

The collocation points can be chosen as tensor products of 1-D Gaussian quadrature points and the interpolating polynomials as tensor products of 1-D Lagrange interpolating functions. However the number of collocation points then grows exponentially with the number of random variables. An alternative method is to use Smolyak sparse grid quadrature (Nobile et al. (2008), Smolyak (1963)) where collocation points that do not increase asymptotic accuracy are removed from the tensor product grid. This results in many fewer collocation points but still more than the number of stochastic degrees of freedom in the stochastic Galerkin method. This method is fully non-intrusive and is easy to implement with existing legacy software (once the sparse grid is generated) Adams et al. (2010).

## 3.5 Numerical illustration

To compare the performance of different solvers and preconditioners discussed above, a 2-D stochastic diffusion equation and a 2-D stochastic advection-diffusion equation presented in section 3.3 are solved using the stochastic Galerkin method described in section 2.4. In the above two problems, the diffusion coefficient is modeled as both a random field discretized using a truncated K-L expansion with uniform random variables (section 2.3.1) and a log-normal random field discretized using a truncated polynomial chaos expansion (section 2.3.2). In the first case, the orthogonal polynomials used in the stochastic Galerkin method are tensor products of 1-D Legendre polynomials and in the second case tensor products of Hermite polynomials are used. For simplicity a constant unit force $f(x, \omega) = 1$ is used as the right-hand-side in equation (3.6). The spatial dimensions are discretized using standard finite element mesh with linear quadrilateral elements. In advection-diffusion equation, the parameter, $\vec{w} = [1, 1]^T$. The corresponding stochastic Galerkin linear system is constructed using the Stokhos and Epetra packages in Trilinos. For the Jacobi solver, Gauss-Seidel solver, Gauss-Seidel preconditioner, and stochastic collocation method, the linear systems are solved via multi-grid preconditioned GMRES provided by the AztecOO and ML packages in Trilinos. To use CG, a symmetric version of the Gauss-Seidel preconditioner is required which comprises of one forward and one backward Gauss-Seidel iteration, and hence it is twice as expensive as that of one forward Gauss-Seidel iteration. It was observed from numerical tests that the symmetric Gauss-Seidel improved the number of Krylov iterations slightly, though its cost is twice that of the non-symmetric Gauss-Seidel preconditioner. Hence, GMRES algorithm is employed for both diffusion and advection-diffusion problems instead of CG.

For the numerical comparisons, the domain $D = [-0.5, 0.5] \times [-0.5, 0.5]$ is discretized into a $64 \times 64$ grid resulting in a total number of nodes, $N_x = 4096$. In figure (3.1) the solution time for the stochastic Galerkin method, scaled by the deterministic solution time at the mean of the random field, is compared for different mesh sizes ($32 \times 32$, $64 \times 64$, $96 \times 96$ and $128 \times 128$) for the diffusion problem demonstrating that the solution time does not depend strongly on the mesh size (as is to be expected for the multigrid preconditioner). The scaled solution time for these solvers and preconditioning techniques as a function of the standard deviation of the input random field, stochastic dimension, and polynomial order are then tabulated in Tables 3.1-3.12. In the tables, the number of Krylov iterations for the aforementioned preconditioners and iterations for Gauss-Seidel and Jacobi solvers are provided in parentheses. In the tables, MB, AGS, AJ, GS and KP are the mean-based, approximate Gauss-Seidel, approximate Jacobi, Gauss-Seidel and Kronecker-product preconditioners respectively. GS in the solution methods refers to the Gauss-Seidel mean algorithm (Algorithm-2). "Jacobi" refers to the Jacobi mean algorithm (Algorithm-1). The solution tolerance for all of the stochastic Galerkin solvers is $1e^{-12}$. For the Gauss-Seidel and Jacobi solvers, the inner solver tolerance is $3e^{-13}$. All the computations are performed using a single core of an 8 core, Intel Xeon machine with 2.66 GHz and 16GB Memory.

Figures 3.2 and 3.3 show the plots of relative residual error vs iteration count for the stochastic Galerkin system with stochastic dimension 4 and polynomial order 4 and standard deviation 0.1. It can be observed that the matrix-free Krylov solver with the Gauss-Seidel preconditioner takes the least number of iterations in case of uniform random field and Gauss-Seidel solver in case of log-normal random field, whereas the Jacobi solver takes highest number of iterations in both cases for a given tolerance. However in terms of solution time, the matrix-free Krylov solver with the approximate Gauss-Seidel preconditioner is the most efficient compared to all other stochastic

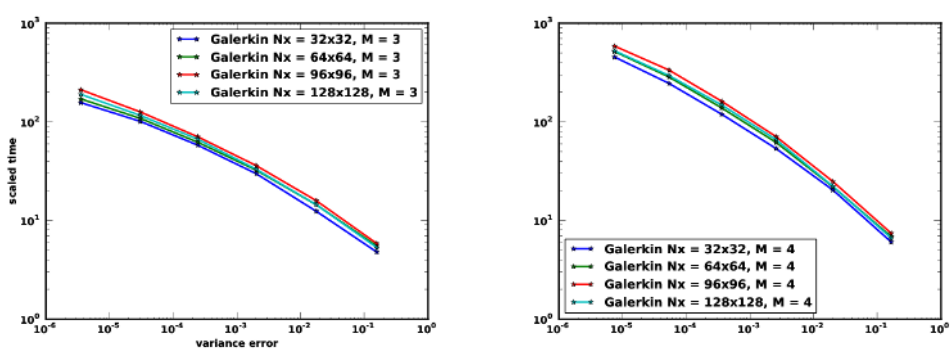
Galerkin solvers. Comparison in terms of iteration count alone is misleading to evaluate the computational cost because, in each iteration the cost of preconditioner as observed in the case of Gauss-Seidel preconditioner could be very high resulting in higher computational cost even with small number of iterations. Hence we also compare the solution time for all preconditioners and solvers.

In the tables, "Div" means diverged. In the case of the uniform random field with small variance ($\sigma = 0.1$), it can be observed from Tables 3.1 and 3.2 that more intrusive Krylov-based stochastic Galerkin solvers are more efficient than less intrusive Gauss-Seidel and Jacobi solvers. Moreover the approximate Gauss-Seidel and Jacobi preconditioners are a significant improvement over the traditional mean-based approach. However as the variance of the random field increases, we can see from Table 3.3 that the Gauss-Seidel and Jacobi solvers suffer considerably, whereas the the Krylov-based approaches (excluding the Gauss-Seidel preconditioner) still perform quite well. This is not unexpected, as the operator becomes more indefinite as the variance increases. In the case of the advection-diffusion equation, the stiffness matrix is non-symmetric.

Tables 3.4-3.6 show the scaled solution time for the advection-diffusion equation with the diffusion coefficient modeled using a Karhunen-Loève with uniform random variables. We can observe that the Krylov solver with the approximate Gauss-Seidel preconditioner performs better than the rest of the Galerkin solvers. In the case of the random field with uniform random variables, the solution time for the diffusion equation is smaller than that for the advection-diffusion equation. This observation could be due to the fact that the multi-grid preconditioner for solving the deterministic diffusion equation is more efficient than that for solving the advection-diffusion equation. Since the efficiency of the preconditioners and solvers based on Gauss-Seidel and Jacobi algorithms depends heavily on the efficiency of the deterministic solver, a good preconditioner for

the mean stiffness matrix will significantly improve the above mentioned stochastic Galerkin preconditioners and solvers.

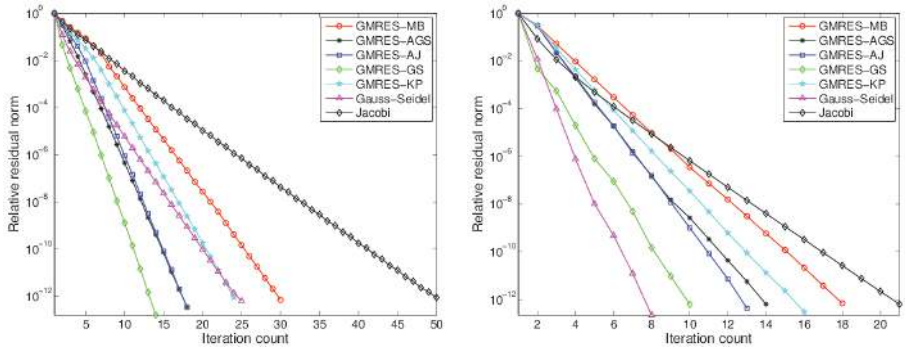
In the case of the log-normal random field, we can see from Tables 3.7 and 3.11 that Gauss-Seidel and Jacobi solvers have not performed well in terms of solution time. The Gauss-Seidel solver is comparable to the Krylov solver with the AGS preconditioner only in case of log-normal random field and at higher polynomial chaos order. For higher variance of the random field, we see from Table 3.9 that the Jacobi solver diverges. This problem might be addressed to some extent by using the true diagonal matrix $K^{k,k} = \sum_{i=0}^{M} c_{ikk} K_i$ from global stochastic stiffness matrix as the left-hand-side in the Jacobi solver and preconditioner instead of the mean matrix $K_0$. In the case of the log-normal random field, both the diffusion and advection-diffusion equations have similar trends and the solution times are comparable. From the Krylov iteration count provided in the parentheses of the tables, we can observe that the preconditiners and solvers based on Gauss-Seidel and Jacobi are robust with respect to stochastic dimension and polynomial order, however they are not robust with respect to the variance of the input random field.




(a) Stochastic dimension = 3

(b) Stochastic dimension = 4

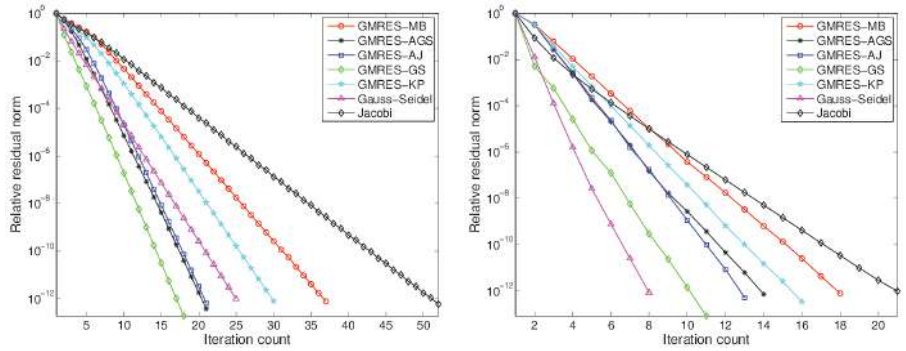
Figure 3.1: Scaled Galerkin solution time vs solution error for varying polynomial chaos order and spatial mesh for the diffusion problem with uniform random field




(a) difusion coefficient modeled as uniform random field

(b) difusion coefficient modeled as lognormal random field

Figure 3.2: Relative residual norm vs iteration countto solve stochastic diffusion equation




(a) difusion coefficient modeled with uniform random variables

(b) difusion coefficient modeled as lognormal random field

Figure 3.3: Relative residual norm vs iteration count to solve stochastic advection-diffusion equation

Table 3.1: Scaled solution time (# of iterations) vs stochastic dimension for random field with uniform random variables, diffusion equation, PC order = 4 and $\sigma$=0.1

| Stoch. | Preconditioners for GMRES | | | | | GS, Jacobi Solvers | |
|---|---|---|---|---|---|---|---|
| dim | MB | AGS | AJ | GS | KP | GS | Jacobi |
| 1 | 10 (21) | 6.5 (13) | 12 (13) | 46 (9) | 7 (14) | 67 (15) | 121 (29) |
| 2 | 39 (25) | 24 (15) | 43 (15) | 168 (11) | 28 (18) | 256 (19) | 490 (39) |
| 3 | 108 (28) | 62 (16) | 109 (16) | 429 (12) | 86 (22) | 723 (23) | 1355 (46) |
| 4 | 235 (29) | 138 (17) | 238 (17) | 937 (13) | 188 (23) | 1587 (25) | 2973 (50) |
| 5 | 482 (30) | 267 (17) | 454 (17) | 1706 (13) | 408 (25) | 2980 (26) | 5725 (53) |
| 6 | 890 (32) | 490 (18) | 822 (18) | 3093 (14) | 776 (27) | 5639 (29) | 10988 (59) |
| 7 | 1456 (33) | 787 (18) | 1308 (18) | 4902 (14) | 1241 (27) | 9361 (30) | 19666 (61) |

Table 3.2: Scaled solution time (# of iterations) vs order of polynomial chaos for random field with uniform random variables, diffusion equation, Stoch. dim=4, $\sigma = 0.1$

| PC | Preconditioners for GMRES | | | | | GS, Jacobi Solvers | |
|---|---|---|---|---|---|---|---|
| order | MB | AGS | AJ | GS | KP | GS | Jacobi |
| 2 | 34 (22) | 22 (14) | 40 (14) | 140 (9) | 28 (18) | 215 (16) | 408 (32) |
| 3 | 99 (26) | 62 (16) | 103 (15) | 397 (11) | 81 (21) | 660 (21) | 1216 (41) |
| 4 | 235 (29) | 138 (17) | 238 (17) | 937 (13) | 188 (23) | 1587 (25) | 2973 (50) |
| 5 | 523 (32) | 285 (18) | 480 (18) | 1850 (14) | 411 (25) | 3344 (29) | 6366 (59) |
| 6 | 948 (35) | 516 (19) | 863 (19) | 3265 (15) | 774 (27) | 6464 (32) | 12327 (67) |
| 7 | 1582 (37) | 821 (19) | 1439 (20) | 5536 (16) | 1338 (29) | 11354 (35) | 24511 (73) |

Table 3.3: Scaled solution time (# of iterations) vs standard deviation ($\sigma$) of input random field, for random field with uniform random variables, diffusion equation, Stoch dim = 4 and PC order = 4

| | Preconditioners for GMRES | | | | | GS, Jacobi Solvers | |
|---|---|---|---|---|---|---|---|
| $\sigma$ | MB | AGS | AJ | GS | KP | GS | Jacobi |
| 0.10 | 235 (29) | 138 (17) | 238 (17) | 937 (13) | 188 (23) | 1587 (25) | 2973 (50) |
| 0.11 | 272 (33) | 147 (18) | 255 (18) | 1005 (14) | 217 (26) | 1898 (30) | 3666 (61) |
| 0.12 | 298 (37) | 164 (20) | 283 (20) | 1139 (16) | 245 (29) | 2374 (37) | 4585 (75) |
| 0.13 | 330 (41) | 182 (22) | 316 (22) | 1340 (19) | 283 (33) | 3057 (48) | 5998 (97) |
| 0.14 | 384 (48) | 211 (25) | 361 (25) | 1544 (22) | 318 (38) | 4152 (65) | 8435 (133) |
| 0.15 | 487 (60) | 272 (31) | 458 (31) | 1956 (28) | 382 (46) | 6567 (101) | 13525 (206) |

Table 3.4: Scaled solution time (# of iterations) vs stochastic dimension for random field with uniform random variables, advection-diffusion equation, PC order = 4, and $\sigma$=0.1

| Stoch. | Preconditioners for GMRES | | | | | GS, Jacobi Solvers | |
|---|---|---|---|---|---|---|---|
| dim | MB | AGS | AJ | GS | KP | GS | Jacobi |
| 1 | 12 (24) | 7 (15) | 14 (15) | 65 (12) | 9 (18) | 64 (14) | 114 (27) |
| 2 | 47 (30) | 27 (17) | 48 (17) | 240 (15) | 37 (23) | 256 (19) | 489 (39) |
| 3 | 133 (34) | 74 (19) | 129 (19) | 631 (17) | 106 (27) | 727 (23) | 1353 (46) |
| 4 | 291 (36) | 165 (20) | 285 (20) | 1343 (18) | 246 (29) | 1579 (25) | 3088 (52) |
| 5 | 598 (38) | 339 (21) | 562 (21) | 2571 (19) | 532 (31) | 3098 (27) | 5941 (55) |
| 6 | 1068 (40) | 611 (22) | 1012 (22) | 4994 (21) | 989 (34) | 5808 (30) | 11567 (62) |
| 7 | 1740 (41) | 976 (22) | 1690 (22) | 7589 (21) | 1619 (35) | 9951 (32) | 21199 (64) |

Table 3.5: Scaled solution time (# of iterations) vs order of polynomial chaos for random field with uniform random variables, advection-diffusion equation, Stoch. dim=4, $\sigma = 0.1$

| PC | Preconditioners for GMRES | | | | | GS, Jacobi Solvers | |
|---|---|---|---|---|---|---|---|
| order | MB | AGS | AJ | GS | KP | GS | Jacobi |
| 2 | 41 (26) | 26 (16) | 45 (16) | 213 (13) | 35 (22) | 233 (17) | 429 (33) |
| 3 | 125 (32) | 70 (18) | 124 (18) | 604 (16) | 102 (26) | 663 (21) | 1278 (43) |
| 4 | 291 (36) | 165 (20) | 285 (20) | 1343 (18) | 246 (29) | 1579 (25) | 3088 (52) |
| 5 | 626 (40) | 336 (21) | 591 (22) | 2654 (20) | 543 (32) | 3302 (29) | 6436 (60) |
| 6 | 1163 (44) | 637 (23) | 1104 (24) | 5082 (22) | 1008 (35) | 6311 (33) | 12558 (68) |
| 7 | 1981 (47) | 1062 (24) | 1825 (25) | 8443 (23) | 1713 (38) | 11143 (36) | 25117 (74) |

Table 3.6: Scaled solution time (# of iterations) vs standard deviation ($\sigma$) for random field with uniform random variables, advection-diffusion equation, Stoch dim = 4, PC order = 4

| | Preconditioners for GMRES | | | | | GS, Jacobi Solvers | |
|---|---|---|---|---|---|---|---|
| $\sigma$ | MB | AGS | AJ | GS | KP | GS | Jacobi |
| 0.10 | 291 (36) | 165 (20) | 285 (20) | 1343 (18) | 246 (29) | 1579 (25) | 3088 (52) |
| 0.11 | 328 (41) | 182 (22) | 326 (23) | 1556 (21) | 284 (33) | 1960 (31) | 3711 (62) |
| 0.12 | 389 (49) | 218 (26) | 371 (26) | 1921 (25) | 325 (39) | 2404 (38) | 4624 (76) |
| 0.13 | 469 (58) | 257 (30) | 461 (31) | 2430 (30) | 380 (46) | 3039 (48) | 5975 (97) |
| 0.14 | 589 (73) | 324 (38) | 565 (39) | 2968 (38) | 478 (57) | 4123 (65) | 8153 (131) |
| 0.15 | 810 (101) | 437 (52) | 742 (52) | 3949 (52) | 650 (78) | 6110 (96) | 12514 (196) |

Table 3.7: Scaled solution time (# of iterations) vs stochastic dimension for log-normal random field, diffusion equation, PC order = 4, and $\sigma$=0.1

| Stoch. | Preconditioners for GMRES | | | | | GS, Jacobi Solvers | |
|---|---|---|---|---|---|---|---|
| dim | MB | AGS | AJ | GS | KP | GS | Jacobi |
| 1 | 9 (16) | 8 (13) | 12 (12) | 48 (9) | 7 (12) | 35.87 (8) | 86 (20) |
| 2 | 33 (17) | 26 (13) | 40 (12) | 149 (9) | 28 (14) | 108 (8) | 255 (20) |
| 3 | 90 (17) | 72 (13) | 102 (12) | 363 (9) | 81 (15) | 254 (8) | 629 (21) |
| 4 | 231 (17) | 183 (13) | 244 (12) | 782 (9) | 217 (15) | 534 (8) | 1327 (21) |
| 5 | 594 (17) | 473 (13) | 577 (12) | 1603 (9) | 580 (15) | 922 (8) | 2292 (21) |
| 6 | 1348 (17) | 1060 (13) | 1220 (12) | 3060 (9) | 1387 (15) | 1613 (8) | 4236 (21) |

Table 3.8: Scaled solution time (# of iterations) vs order of polynomial chaos for log-normal random field, diffusion equation, Stoch. dim=4, $\sigma = 0.1$

| PC | Preconditioners for GMRES | | | | | GS, Jacobi Solvers | |
|---|---|---|---|---|---|---|---|
| order | MB | AGS | AJ | GS | KP | GS | Jacobi |
| 2 | 25 (15) | 23 (13) | 36 (12) | 116 (7) | 24 (14) | 94 (7) | 202 (16) |
| 3 | 77 (16) | 65 (13) | 97 (12) | 355 (9) | 70 (14) | 253 (8) | 566 (19) |
| 4 | 231 (17) | 183 (13) | 244 (12) | 782 (9) | 217 (15) | 534 (8) | 1327 (21) |
| 5 | 738 (18) | 593 (14) | 644 (12) | 1888 (10) | 664 (15) | 918 (8) | 2642 (24) |
| 6 | 2181 (19) | 1645 (14) | 1660 (12) | 4111 (10) | 1963 (15) | 1606 (8) | 5046 (26) |

Table 3.9: Scaled solution time (# of iterations) vs standard deviation ($\sigma$) for log-normal random field, diffusion equation, Stoch dim = 4 and PC order = 4

| | Preconditioners for GMRES | | | | | GS, Jacobi Solvers | |
|---|---|---|---|---|---|---|---|
| $\sigma$ | MB | AGS | AJ | GS | KP | GS | Jacobi |
| 0.10 | 231 (17) | 183 (13) | 244 (12) | 782 (9) | 217 (15) | 534 (8) | 1327 (21) |
| 0.15 | 275 (20) | 217 (15) | 267 (13) | 942 (11) | 243 (17) | 638 (10) | 2030 (33) |
| 0.20 | 332 (24) | 247 (17) | 285 (14) | 1184 (14) | 273 (19) | 764 (12) | 3897 (62) |
| 0.25 | 377 (27) | 286 (20) | 288 (14) | 1340 (16) | 302 (21) | 957 (15) | 18672 (268) |
| 0.30 | 456 (32) | 328 (23) | 346 (17) | 1583 (19) | 333 (23) | 1150 (18) | Div |
| 0.35 | 518 (37) | 372 (26) | 591 (29) | 1827 (22) | 359 (25) | 1406 (22) | Div |

Table 3.10: Sacled solution time (# of iterations) vs stochastic dimension for log-normal random field, advection-diffusion equation, PC order = 4 and $\sigma$=0.1

| Stoch. | Preconditioners for GMRES | | | | | GS, Jacobi Solvers | |
|---|---|---|---|---|---|---|---|
| dim | MB | AGS | AJ | GS | KP | GS | Jacobi |
| 1 | 9 (16) | 8 (13) | 12 (12) | 48 (9) | 12 (12) | 36 (8) | 81 (19) |
| 2 | 33 (17) | 26 (13) | 40 (12) | 148 (9) | 40 (12) | 108 (8) | 255 (20) |
| 3 | 89 (17) | 72 (13) | 102 (12) | 396 (10) | 102 (12) | 252 (8) | 627 (21) |
| 4 | 229 (17) | 184 (13) | 241 (12) | 858 (10) | 215 (12) | 507 (8) | 1263 (21) |
| 5 | 594 (17) | 473 (13) | 576 (12) | 1765 (10) | 572 (12) | 915 (8) | 2399 (22) |
| 6 | 1426 (18) | 1061 (13) | 1221 (12) | 3345 (10) | 1370 (15) | 1737 (8) | 4044 (22) |

Table 3.11: Scaled solution time (# of iterations) vs order of polynomial chaos for log-normal random field, advection-diffusion equation, Stoch. dim=4, $\sigma = 0.1$

| PC | Preconditioners for GMRES | | | | | GS, Jacobi Solvers | |
|---|---|---|---|---|---|---|---|
| order | MB | AGS | AJ | GS | KP | GS | Jacobi |
| 2 | 25 (15) | 23 (13) | 36 (12) | 129 (8) | 36 (12) | 94 (7) | 202 (16) |
| 3 | 77 (16) | 65 (13) | 97 (12) | 353 (9) | 96 (12) | 252 (8) | 564 (19) |
| 4 | 229 (17) | 184 (13) | 241 (12) | 858 (10) | 215 (12) | 507 (8) | 1263 (21) |
| 5 | 733 (18) | 585 (14) | 645 (12) | 1882 (10) | 640 (12) | 1030 (9) | 2631 (24) |
| 6 | 2157 (19) | 1664 (14) | 1784 (13) | 4085 (10) | 1938 (15) | 1731 (9) | 4821 (26) |

Table 3.12: Scaled solution time (# of iterations) vs standard deviation ($\sigma$) for log-normal random field, advection-diffusion equation, Stoch dim = 4 and PC order = 4

| | Preconditioners for GMRES | | | | | GS, Jacobi Solvers | |
|---|---|---|---|---|---|---|---|
| $\sigma$ | MB | AGS | AJ | GS | KP | GS | Jacobi |
| 0.10 | 229 (17) | 184 (13) | 241(12) | 858 (10) | 215 (15) | 507 (8) | 1263 (21) |
| 0.15 | 275 (20) | 214 (15) | 267 (13) | 1020 (12) | 266 (13) | 634 (10) | 2083 (34) |
| 0.20 | 333 (24) | 255 (18) | 285 (14) | 1177 (14) | 285 (14) | 825 (13) | 3889 (62) |
| 0.25 | 391 (28) | 285 (20) | 305 (15) | 1415 (17) | 304 (15) | 1018 (16) | 17511 (254) |
| 0.30 | 464 (33) | 328 (23) | 361 (18) | 1656 (20) | 362 (18) | 1210 (19) | Div |
| 0.35 | 529 (38) | 387 (27) | 660 (32) | 1979 (24) | 664 (32) | 1464 (23) | Div |

# Chapter 4

# Basis adaptation - dimension reduction

## 4.1 Introduction

The curse of dimensionality is a significant challenge that remains at the forefront of scientific computing. We address it in the present paper by recognizing the low dimensional structure of many quantities of interest (QoI) even in problems that have been embedded, via parameterization, in very high-dimensional settings. We take advantage of the geometric structure associated with the parameterization to discover very low-dimensional structure around which the probabilistic content of the QoI is concentrated.

Recent advances in sensing technologies have enabled querying the natural world with resolutions that continue to shed new light on many physical processes that underlie phase transformation and other instabilities. The mathematical formulation of these problems typically involves systems of coupled equations parameterized with stochastic processes that attempt to capture the effect of unmodeled features on the state variables. This stochastic parameterization naturally embeds the problem in a high dimensional parameter space where methods of white-noise calculus have recently been adapted to characterizing and estimating the solution of these equations

Ghanem and Spanos (1991), Holden et al. (1996), Xiu and Karniadakis (2002). While this setting served to develop general insight into the significance of the stochastic component on the predictability of associated physical processes, it continues to present challenges when dealing with realistic problems that involve very high-dimensional parameterizations. The difficulty is associated with the observation that, in addition to the fine spatio-temporal discretization required to resolve underlying physical phenomena, resolving the dependence in parameter space increases the number of degrees of freedom exponentially with the number of parameters involved. Some attempts have been made in recent years to address this problem by pursuing sparse Doostan and Iaccarino (2009), Doostan and Owhadi (2011), Schwab and Gittelson (2011) or multilevel Doostan et al. (2007) representations that have succeeded in reducing the computational effort by an order of magnitude. Given the exponential growth with dimension, however, the residual effort remains often prohibitive.

This complexity is inherited from the mathematical construction used to describe the physical processes involved, and often belies the simplicity of many quantities of interest (QoI) upon which decisions are eventually based. These QoI's have typically slow dynamics and pertain to a small number of functionals of the solution. Thus while describing the full solution field as a high-dimensional mathematical object may seem conceptually reasonable, a similar characterization of inherently low-dimensional quantities seems like an overkill and begs for a rational reduction. Equation-free and projective integration approaches C. Gear et al. (2003) present a useful methodology for tackling these problems when the functional dependence of the QoI on the parameterization of the original problem is not significant to the QoI. In this paper we proceed along different lines, capitalizing on the mathematical structure of Gaussian Hilbert spaces and the associated Homogeneous Chaos space obtained from them through a tensor product construction,

to obtain a reduced model that, while capturing the probabilistic content of the QoI, characterizes its functional dependence on the original parameterization. This can be important for carrying out deterministic and stochastic sensitivity analyses and worth of information studies. The approach hinges on Gaussian parameterization of the problem. This does not signify that the parameters in the problem are assumed to be Gaussian, but rather that they are functions of some underlying Gaussian variables or processes. In a first step, an isometry is applied to an underlying Gaussian variables to induce a particular desired structure in the representation of the solution. In a second step, a reduction through projection of the resulting representation is carried out that has a probabilistic interpretation, thus providing further justification for the algebraic manipulations. The methodology is demonstrated on an algebraic problem and a problem associated with an elliptic equation from elasticity with stochastic coefficients.

## 4.2 Polynomial Chaos Decomposition

Consider an experiment described by the probability triple $(\Omega, \mathcal{F}, P)$ and $\boldsymbol{\xi} = (\xi_1, \cdots, \xi_d)$ a set of independent Gausssian random variables on $\Omega$. Let $H$ denote the Gaussian space spanned by $\boldsymbol{\xi}$ and $\mathcal{F}(H)$ the $\sigma$-algebra generated by $H$. For notational simplicity, let $L_2(\Omega)$ be a shorthand for $L_2(\Omega, \mathcal{F}(H), P)$. Denote by $H^{:n:}$ the space of d-dimensional polynomials in $\boldsymbol{\xi}$ of exact degree $n$. When generated by Gaussian variables, $H^{:n:}$ is also referred to as the Homogeneous Chaos of order $n$. Then it can be shown that $L_2(\Omega) = \bigoplus_n H^{:n:}$ Janson (1999). Given this construction, any basis in $H$ induces a corresponding basis in $H^{:n:}$ and $L_2(\Omega)$. The main contribution of this paper is to adapt the hilbertian basis in $L_2(\Omega)$ by a judicious basis transformation in the underlying Gaussian space $H$. Introducing the multi-index $\boldsymbol{\alpha} = (\alpha_1 \cdots, \alpha_d)$, then each $H^{:n:}$ is spanned

by $\{h_{\boldsymbol{\alpha}},\ |\boldsymbol{\alpha}| = n\}$, the set of multivariate Hermite polynomials of order $n$. We introduce a normalization of these polynomials

$$\psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi}) = H_{\boldsymbol{\alpha}}(\boldsymbol{\xi})/\sqrt{\boldsymbol{\alpha}!}\,, \qquad \|H_{\boldsymbol{\alpha}}\|^2_{L_2(\Omega)} = \boldsymbol{\alpha}!\,, \tag{4.1}$$

and consider $q : \mathbb{R}^d \mapsto \mathbb{R}$ such that $\int_{\mathbb{R}^d}(q(\boldsymbol{x}))^2 e^{-\frac{1}{2}\boldsymbol{x}^T\boldsymbol{x}} d\boldsymbol{x} < \infty$. Then $q \in L_2(\Omega)$ and can be represented in a Polynomial Chaos decomposition of the form Ghanem and Spanos (1991),

$$q^p(\boldsymbol{\xi}) = \sum_{|\boldsymbol{\alpha}|\leq p} q_{\boldsymbol{\alpha}}\psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi}), \tag{4.2}$$

where $\lim_{p\to\infty} q^p(\boldsymbol{\xi}) = q(\boldsymbol{\xi})$ in the $L_2(\Omega)$ norm. In the above, $|\boldsymbol{\alpha}| = \sum_{i=1}^d \alpha_i$ and $\boldsymbol{\alpha}! = \prod_{i=1}^d \alpha_i!$.

In order to guide our basis adaptation process, we next introduce a few sets of multi-indices. Denote by $\mathcal{I}_p$ the set of all $d$-dimensional multi-indices of degree less than or equal to $p$, by $\mathcal{E}_i$ the subset of $\mathcal{I}_p$ corresponding to non-zero indices only at the $i^{th}$ location, and let $\mathcal{E} = \bigcup_{i=1}^d \mathcal{E}_i$. Also let $\mathcal{E}_{ij}$ $(i \neq j)$ denote the subset of $\mathcal{I}_p$ with zeros except at locations $i$ and $j$. Moreover, let $\boldsymbol{e}_i$ and $\boldsymbol{e}_{ij}$ denote the unit vectors in $\mathcal{E}_i$ and $\mathcal{E}_{ij}$, respectively, where the non-zero entries are equal to 1.

### 4.2.1 Change of Basis

Let $\boldsymbol{A}$ be an isometry on $\mathbb{R}^d$ and let $\boldsymbol{\eta}$ be defined as

$$\boldsymbol{\eta} = \boldsymbol{A}\boldsymbol{\xi} \qquad \boldsymbol{A}\boldsymbol{A}^T = \boldsymbol{I}\,. \tag{4.3}$$

Also, let $q^{\boldsymbol{A}}$ be the image of $q$ under this mapping, $q^{\boldsymbol{A}}(\boldsymbol{\eta}) \stackrel{\Delta}{=} q(\boldsymbol{\xi})$. Since $\boldsymbol{\eta}$ is also a basis in $H$, the $n^{th}$ order Hermite polynomials in $\boldsymbol{\eta}$ span $H^{:n}$. Letting

$$\psi_{\boldsymbol{\alpha}}^{\boldsymbol{A}}(\boldsymbol{\xi}) = \psi_{\boldsymbol{\alpha}}(\boldsymbol{\eta}) \ , \tag{4.4}$$

and given the respective expansions of $q$ and $q^{\boldsymbol{A}}$ in the forms,

$$q(\boldsymbol{\xi}) = \sum_{\boldsymbol{\alpha}\in\mathcal{I}_p} q_{\boldsymbol{\alpha}}\psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi}), \quad q^{\boldsymbol{A}}(\boldsymbol{\eta}) = \sum_{\boldsymbol{\beta}\in\mathcal{I}_p} q_{\boldsymbol{\beta}}^{\boldsymbol{A}}\psi_{\boldsymbol{\beta}}(\boldsymbol{\eta}) \ , \tag{4.5}$$

results in

$$q_{\boldsymbol{\beta}}^{\boldsymbol{A}} = \sum_{\boldsymbol{\alpha}\in\mathcal{I}_p} q_{\boldsymbol{\alpha}}(\psi_{\boldsymbol{\alpha}}, \psi_{\boldsymbol{\beta}}^{\boldsymbol{A}}) \ , \tag{4.6}$$

where $(\psi_{\boldsymbol{\alpha}}, \psi_{\boldsymbol{\beta}})$ is the inner product defined as expectation of the product relative to the Gaussian measure.

### 4.2.2 Reduction via Projection

Consider a subspace $V_{\mathcal{I}}$ of $L^2(\Omega)$ spanned by $\{\psi_{\boldsymbol{\beta}}^{\boldsymbol{A}}; \boldsymbol{\beta} \in \mathcal{I}\}$, where $\mathcal{I} \subset \mathcal{I}_p$. The projection of $q^{\boldsymbol{A}}$ on $V_{\mathcal{I}}$, denoted by $q^{\boldsymbol{A},\mathcal{I}}$, can be expressed in the form

$$q^{\boldsymbol{A},\mathcal{I}}(\boldsymbol{\eta}) = \sum_{\boldsymbol{\beta}\in\mathcal{I}} \sum_{\boldsymbol{\alpha}\in\mathcal{I}_p} q_{\boldsymbol{\alpha}}(\psi_{\boldsymbol{\alpha}}, \psi_{\boldsymbol{\beta}}^{\boldsymbol{A}})\psi_{\boldsymbol{\beta}}(\boldsymbol{\eta}) \ . \tag{4.7}$$

Alternatively, the projection of $q^{\boldsymbol{A}}$ on $V_{\mathcal{I}}$ can be expressed with respect to $\{\psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi})\}$, resulting in

$$q^{\boldsymbol{A},\mathcal{I}}(\boldsymbol{\eta}) = \sum_{\boldsymbol{\gamma}\in\mathcal{I}_p} q_{\boldsymbol{\gamma}}^{\mathcal{I}}\psi_{\boldsymbol{\gamma}}(\boldsymbol{\xi}) \ . \tag{4.8}$$

This yields,

$$q_{\boldsymbol{\gamma}}^{\mathcal{I}} = \sum_{\boldsymbol{\beta}\in\mathcal{I}} \sum_{\boldsymbol{\alpha}\in\mathcal{I}_p} q_{\boldsymbol{\alpha}}(\psi_{\boldsymbol{\alpha}}, \psi_{\boldsymbol{\beta}}^{\boldsymbol{A}})(\psi_{\boldsymbol{\beta}}^{\boldsymbol{A}}, \psi_{\boldsymbol{\gamma}}) \tag{4.9}$$

Denoting $\boldsymbol{q}^{\mathcal{I}} = \{q_{\boldsymbol{\gamma}}^{\mathcal{I}}; \boldsymbol{\gamma} \in \mathcal{I}\}$ and $\boldsymbol{q} = \{q_{\boldsymbol{\alpha}}; \boldsymbol{\alpha} \in \mathcal{I}_p\}$, the projection $\boldsymbol{q}^{\mathcal{I}}$ is clearly the restriction of $q$ to $V_{\mathcal{I}}$. The accuracy of $\boldsymbol{q}^{\mathcal{I}}$ as an approximant to $\boldsymbol{q}$ is thus dependent both on the choice of $\boldsymbol{A}$ and on $\mathcal{I}$. Clearly, as $\boldsymbol{A}$ approaches the identity map on $\mathbb{R}^d$ and $\mathcal{I}$ approaches $\mathcal{I}_p$, $\boldsymbol{q}^{\mathcal{I}}$ approaches $\boldsymbol{q}$. Also, if $\boldsymbol{A}$ is fixed as the identity map, then this approximation merely consists of selecting particular terms in the standard polynomial expansion of $q$. The introduction of $\boldsymbol{A}$ provides a parameterization of the projections and hence additional freedom for adapting the approximation. Introducing the Gramian $\boldsymbol{C}$ defined by its components $\boldsymbol{C}_{\boldsymbol{\alpha\beta}} = (\psi_{\boldsymbol{\alpha}}, \psi_{\boldsymbol{\beta}}^{\boldsymbol{A}})$, the error associated with a projection on $V_{\mathcal{I}}$ can be expressed as,

$$\boldsymbol{q} - \boldsymbol{q}^{\mathcal{I}} = \left[\boldsymbol{I} - \boldsymbol{C}\boldsymbol{C}^T\right] \boldsymbol{q} \,. \tag{4.10}$$

It is clear that the null space of $\boldsymbol{I} - \boldsymbol{C}\boldsymbol{C}^T$ and the spectrum of $\boldsymbol{C}$ are shaped by both $\boldsymbol{A}$ and $\mathcal{I}$. Our construction does not attempt to minimize any measure of this error, but rather seeks probabilistic arguments for the construction of $\boldsymbol{A}$ and the selection of the projection set $\mathcal{I}$.

## 4.3 Special Cases

Two special cases are now considered where $\boldsymbol{A}$ is used to introduce knowledge about $q$ into the process of basis adaptation.

### 4.3.1 Case where Gaussian components of $q$ are known

If the projection of $q$ on the Gaussian subspace, $H$, is known, possibly through some extraneous numerical effort, a very interesting reduction ensues. Specifically, let $\boldsymbol{A}$ be constructed such that

$$\eta_1 = \sum_{\boldsymbol{\alpha}\in\mathcal{I}_1} q_{\boldsymbol{\alpha}}\psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi}) = \sum_{i=1}^{d} q_{\boldsymbol{e}_i}\xi_i \tag{4.11}$$

so that the first component of $\boldsymbol{\eta}$ captures the complete Gaussian component of $q$. Let the remaining components of the vector $\boldsymbol{\eta}$ be constructed through a Gram-Schmidt procedure, yielding the matrix $\boldsymbol{A}$. Furthermore, let the set $\mathcal{I}$ introduced previously consist of all multi-indices that reference only $\eta_1$ up to degree $p$, and is thus specified as $\mathcal{I} = \mathcal{E}_1$. In this case, $q^{\boldsymbol{A}}(\boldsymbol{\eta})$ and $q^{\boldsymbol{A},\mathcal{I}}(\boldsymbol{\eta})$ are given respectively as,

$$\begin{aligned} q^{\boldsymbol{A}}(\boldsymbol{\eta}) &= q_0^{\boldsymbol{A}} + q_{\boldsymbol{e}_1}^{\boldsymbol{A}}\eta_1 + \sum_{1<|\boldsymbol{\beta}|\leq p} q_{\boldsymbol{\beta}}^{\boldsymbol{A}}\psi_{\boldsymbol{\beta}}(\boldsymbol{\eta}) \\ &= q_0^{\boldsymbol{A}} + q_{\boldsymbol{e}_1}^{\boldsymbol{A}}\eta_1 + \sum_{\substack{1<|\boldsymbol{\beta}|\leq p \\ \boldsymbol{\beta}\in\mathcal{E}_1}} q_{\boldsymbol{\beta}}^{\boldsymbol{A}}\psi_{\boldsymbol{\beta}}(\boldsymbol{\eta}) + \sum_{\substack{1<|\boldsymbol{\beta}|\leq p \\ \boldsymbol{\beta}\notin\mathcal{E}_1}} q_{\boldsymbol{\beta}}^{\boldsymbol{A}}\psi_{\boldsymbol{\beta}}(\boldsymbol{\eta}) \,. \end{aligned} \tag{4.12}$$

and

$$q^{\boldsymbol{A},\mathcal{I}}(\boldsymbol{\eta}) = q_0^{\boldsymbol{A}} + \sum_{\boldsymbol{\beta}\in\mathcal{E}_1} q_{\boldsymbol{\beta}}^{\boldsymbol{A}}\psi_{\boldsymbol{\beta}}(\boldsymbol{\eta}) \tag{4.13}$$

and the error in the representation is given as,

$$q^{\boldsymbol{A}}(\boldsymbol{\eta}) - q^{\boldsymbol{A},\mathcal{I}}(\boldsymbol{\eta}) = \sum_{\substack{1<|\boldsymbol{\beta}|\leq p \\ \boldsymbol{\beta}\notin\mathcal{E}_1}} q_{\boldsymbol{\beta}}^{\boldsymbol{A}}\psi_{\boldsymbol{\beta}}(\boldsymbol{\eta}) \tag{4.14}$$

which has mean zero and is orthogonal to all the Gaussian terms. Thus, starting with a knowledge of the Gaussian component of $q$ (i.e. its projection on the Gaussian chaos), an isomery $\boldsymbol{A}$ is constructed such that this whole projection is carried by a single dimension. The choice of $\mathcal{I}$ is then merely adapted to the situation where the probability measure of $q$ is concentrated around this leading dimension. Clearly, the set $\mathcal{I}$ can be enlarged by activating successive $\eta$'s in the list of multi-indices. In the limit when all $\eta$ are active, $\mathcal{I} = \mathcal{I}_p$ and an exact representation is obtained.

### 4.3.2 Case where quadratic components of $q$ are known

It is assumed in this subsection that the quadratic components of $q$ are known. An expression of $q$ that highlights the second order terms takes the form,

$$q(\boldsymbol{\xi}) = q_0 + \sum_{i=1}^{d} \hat{q}_i \xi_i + \sum_{i=1}^{d} \sum_{j=1}^{d} \hat{q}_{ij} (\xi_i \xi_j - \delta_{ij}) + \sum_{3 \leq \boldsymbol{\alpha} \leq p} q_{\boldsymbol{\alpha}} \psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi}) \tag{4.15}$$

where $\hat{q}_i = q_{\boldsymbol{e}_i}$, $\hat{q}_{ij} = q_{\boldsymbol{e}_{ij}}/\sqrt{2}$, and $\hat{q}_{ii} = q_{2\boldsymbol{e}_i}/\sqrt{2}$. The second order truncation of equation (4.15) can be re-written as

$$q_{II}(\boldsymbol{\xi}) = (q_0 - \sum_{i=1}^{d} \hat{q}_{ii}) + \sum_{i=1}^{d} \hat{q}_i \xi_i + \sum_{i=1}^{d} \sum_{j=1}^{d} \hat{q}_{ij} \xi_i \xi_j \ . \tag{4.16}$$

Let matrix $\boldsymbol{S}$ have entries $\hat{q}_{ij}$, and let $\boldsymbol{\eta} = \boldsymbol{A}\boldsymbol{\xi}$ where

$$\boldsymbol{A}\boldsymbol{S}\boldsymbol{A} = \boldsymbol{D} \tag{4.17}$$

where $\boldsymbol{D}$ is a diagonal matrix whose elements are denoted as $d_i$. Thus $\boldsymbol{A}$ and $\boldsymbol{D}$ are, respectively, the matrix of eigenvectors and the eigenvalues of the symmetric matrix $\boldsymbol{S}$, and thus $\boldsymbol{A}$ is clearly a unitary operator. Noting that the trace of $\boldsymbol{A}$ is equal to the trace of $\boldsymbol{D}$, $q^{\boldsymbol{A}}(\boldsymbol{\eta})$ takes the form,

$$q^{\boldsymbol{A}}(\boldsymbol{\eta}) = q_0 + \sum_{i=1}^{d} b_i \eta_i + \sum_{i=1}^{d} d_i(\eta_i^2 - 1) + \sum_{|\boldsymbol{\beta}|>2} q_{\boldsymbol{\beta}} \psi_{\boldsymbol{\beta}}(\boldsymbol{\eta}) \ , \tag{4.18}$$

where $b_i = \sum_{j=1}^{d} a_{ij} \hat{q}_j$ and $a_{ij}$ denotes elements of $\boldsymbol{A}$. Letting $\mathcal{I} = \mathcal{E}$, the projection on $V_{\mathcal{I}}$ takes the form,

$$q^{\boldsymbol{A},\mathcal{I}} = q_0 + \sum_{i=1}^{d} \sum_{\boldsymbol{\beta} \in \mathcal{E}_i} q_{\boldsymbol{\beta}}^{\boldsymbol{A}} \psi_{\boldsymbol{\beta}}(\boldsymbol{\eta}) \ , \tag{4.19}$$

which captures exactly the second order effects while including the higher order effects only along one-dimensional subspaces. This approximation can be achieved using $d$ one-dimensional polynomial approximations, each of order $p$. Clearly, constructing the operator $\boldsymbol{A}$ requires additional work to evaluate the second order approximation of function $q$. The error associated with this approximation can be expressed as

$$q^{\boldsymbol{A}}(\boldsymbol{\eta}) - q^{\boldsymbol{A},\mathcal{I}}(\boldsymbol{\eta}) = \sum_{\substack{2<|\boldsymbol{\beta}|\leq p \\ \boldsymbol{\beta} \notin \mathcal{E}}} q_{\boldsymbol{\beta}}^{\boldsymbol{A}} \psi_{\boldsymbol{\beta}}(\boldsymbol{\eta}) \ . \tag{4.20}$$

The error reflects the interaction between the $\eta_i$ components with polynomial order greater than 2. It can be reduced by augmenting $V_{\mathcal{I}}$ with these higher-order interactions.

## 4.4 Application to Equations with Random Coefficients

Consider the situation where $u \in V$ is the solution of an equation with random coefficients, $V$ is some appropriate function space that has at least the structure of a topological vector space. Let $\mathcal{L}_{\boldsymbol{\xi}}$ denote the operator representing this equation and which is parameterized by $\boldsymbol{\xi} \in \mathbb{R}^d$, a vector of independent Gaussian variables. We note that this does not mean that the parameters of the underlying model are themselves Gaussian variables, but merely that they have been described as functions of such variables. We are interested in characterizing the probabilistic structure, through its chaos decomposition, of a functional, $q$, on $V$. The problem could then be posed as characterizing the solution of the following equation subject to sufficient additional constraints (such as boundary or initial conditions),

$$\begin{aligned} \mathcal{L}_{\boldsymbol{\xi}}[u] &= 0 \quad \forall u \in V \otimes L_2(\Omega) \\ q(\boldsymbol{\xi}) &= f(u(\boldsymbol{\xi})) \,, \end{aligned} \tag{4.21}$$

where the dependence of $u$ and $q$ on $\boldsymbol{\xi}$ is emphasized. We assume throughout that $\mathcal{L}_{\boldsymbol{\xi}}$ and $f$ are such that $u$ and $q$ are square-integrable with respect to the density of $\boldsymbol{\xi}$, and they therefore admit polynomial chaos decompositions.

### 4.4.1 Non-Intrusive Implementation

A good transformation $\boldsymbol{A}$ is such that the function $q$ does not vary significantly except along a small subset of $\boldsymbol{\eta}$ in which case integrals of $q$ over the whole $\boldsymbol{\xi}$ space can be expressed as much lower-dimensional integrals with respect to that subset.

The coefficients in the chaos decomposition of $q$ can be expressed as an integral which is then approximated through quadrature as,

$$q_{\boldsymbol{\beta}} = \int_{\mathbb{R}^d} q(\boldsymbol{\xi})\, \psi_{\boldsymbol{\beta}}(\boldsymbol{\xi})\, e^{\frac{1}{2}\boldsymbol{\xi}^T\boldsymbol{\xi}}\, d\boldsymbol{\xi} \;\approx\; \sum_{r=1}^{s} q(\boldsymbol{\xi}^{(r)}) w_r \;, \quad \boldsymbol{\beta} \in \mathcal{I}_p \tag{4.22}$$

where $\{\boldsymbol{\xi}^{(r)}\}$ and $\{w_r\}$ are the coordinates and associated weights for a $d$-dimensional quadrature rule with $s$ points, and the integrand is the solution of the following deterministic problem

$$\begin{aligned} \mathcal{L}_{\boldsymbol{\xi}^{(r)}}[u] &= 0 \quad u \in V, \quad r = 1, \cdots, s \\ q(\boldsymbol{\xi}^{(r)}) &= f(u(\boldsymbol{\xi}^{(r)})), \quad r = 1, \cdots, s\,. \end{aligned} \tag{4.23}$$

The computational burden of carrying out the above high-dimensional quadrature increases with $|\boldsymbol{\beta}|$ and quickly becomes prohibitive. This procedure can be used, however, to construct a first order ($|\boldsymbol{\beta}| = 1$) or second order ($|\boldsymbol{\beta}| = 2$) representation, from which an isometry $\boldsymbol{A}$ is constructed as described above. Assuming, for instance, that the projection of the solution on the Gaussian subspace has been computed using a $d$-dimensional quadrature as described in this section, equation (4.11) is used to construct $\boldsymbol{A}$, and then a one dimensional representation of $q$ is sought in the form,

$$q = q_0 + \sum_{\boldsymbol{\beta} \in \mathcal{E}_1} q_{\boldsymbol{\beta}} \psi_{\boldsymbol{\beta}}(\boldsymbol{\eta})\;. \tag{4.24}$$

The coefficients $q_{\boldsymbol{\beta}}$ in this expansion can now be obtained through one-dimensional $s_o$-point quadrature with respect to $\eta_1$ in the form,

$$q_{\boldsymbol{\beta}} = \sum_{r=1}^{s_o} q(\boldsymbol{\eta}^{(r)})\psi_{\boldsymbol{\beta}}(\boldsymbol{\eta}^{(r)})w_r \ , \quad \boldsymbol{\beta} \in \mathcal{E}_1 \ , \tag{4.25}$$

where $\boldsymbol{\eta}^{(r)} = (\eta_1^{(r)}, 0, \cdots, 0)$, and $q(\boldsymbol{\eta}^{(r)})$ is obtained by solving the following equations,

$$\begin{aligned} \boldsymbol{\xi}^{(r)} &= \boldsymbol{A}^{-1}\boldsymbol{\eta}^{(r)}, \quad r = 1, \cdots, s_o \\ \mathcal{L}_{\boldsymbol{\xi}^{(r)}}[u] &= 0 \quad u \in V, \quad r = 1, \cdots, s_o \\ q(\boldsymbol{\eta}^{(r)}) &= f(u(\boldsymbol{\xi}^{(r)})), \quad r = 1, \cdots, s_o \ . \end{aligned} \tag{4.26}$$

In this case, and as previously explained, $\boldsymbol{A}$ is merely constrained by requiring $\boldsymbol{\eta} = \boldsymbol{A}\boldsymbol{\xi}$, and its construction completed through an arbitrary Gram-Schmidt procedure. Clearly, the non-uniqueness of $\boldsymbol{A}$ will be manifested in the first of equations (4.26). For quantities of interest that are strongly dominated by $\eta_1$, however, it is expected that this arbitrariness in specifying $\boldsymbol{\xi}$ is inconsequential in evaluating the integral.

### 4.4.2 Intrusive Implementation

In some instances, reduction of the multi-dimensional integral as indicated above may not be justified. In that case, a procedure as explained in this subsection can be implemented. It will again be assumed that some effort has already been expanded in computing the isometry $\boldsymbol{A}$ that is adapted to $q$, together with an approximating subspace $V_{\mathcal{I}}$. Without loss of generality, and to keep the focus of the presentation on the basis adaptation procedure, it is assumed throughout this section that $\mathcal{L}_{\boldsymbol{\xi}}$ is a linear operator. Describing $u$ in its polynomial chaos decomposition, and

requiring the error to be orthogonal to the approximating space results in the following system of deterministic equations to be solved for the coefficients $u_{\boldsymbol{\beta}}$,

$$\sum_{\boldsymbol{\beta}\in\mathcal{I}_p} \left(\mathcal{L}_{\boldsymbol{\xi}}[u_{\boldsymbol{\beta}}]\ \psi_{\boldsymbol{\beta}}, \psi_{\boldsymbol{\gamma}}\right) = 0 \quad \forall \boldsymbol{\gamma} \in \mathcal{I}_p\ . \tag{4.27}$$

where $(.,.)$ is the inner product introduced in equation (4.6). In some instances, the operator $\mathcal{L}_{\boldsymbol{\xi}}$ lends itself to a chaos decomposition, whereby we can introduce the deterministic operators $\mathcal{L}_{\boldsymbol{\alpha}}$ in the form,

$$\mathcal{L}_{\boldsymbol{\alpha}} = \left(\mathcal{L}_{\boldsymbol{\xi}}, \psi_{\boldsymbol{\alpha}}\right), \quad \boldsymbol{\alpha} \in \mathcal{I}_p\ , \tag{4.28}$$

resulting in

$$\sum_{\boldsymbol{\beta}\in\mathcal{I}_p} \sum_{\boldsymbol{\alpha}\in\mathcal{I}_p} \mathcal{L}_{\boldsymbol{\alpha}}[u_{\boldsymbol{\beta}}]\ \left(\psi_{\boldsymbol{\alpha}}\psi_{\boldsymbol{\beta}}, \psi_{\boldsymbol{\gamma}}\right) = 0 \quad \forall \boldsymbol{\gamma} \in \mathcal{I}_p\ . \tag{4.29}$$

If the solution is instead represented relative to the adapted basis $\boldsymbol{\eta}$ the following set of equations ensues,

$$\sum_{\boldsymbol{\beta}\in\mathcal{I}_p} \sum_{\boldsymbol{\alpha}\in\mathcal{I}_p} \mathcal{L}_{\boldsymbol{\alpha}}[u_{\boldsymbol{\beta}}]\ \left(\psi_{\boldsymbol{\alpha}}\psi^{\boldsymbol{A}}_{\boldsymbol{\beta}}, \psi^{\boldsymbol{A}}_{\boldsymbol{\gamma}}\right) = 0 \quad \forall \boldsymbol{\gamma} \in \mathcal{I}_p\ , \tag{4.30}$$

where $\left(\psi_{\boldsymbol{\alpha}}\psi^{\boldsymbol{A}}_{\boldsymbol{\beta}}, \psi^{\boldsymbol{A}}_{\boldsymbol{\gamma}}\right)$ is an integral with respect to the Gaussian measure, and represents an intrinsic property of Hermite polynomials that could be evaluated ahead of the numerical solution of the equations. While analytical expressions for this integral can be obtained, it may be most effective to tabulate it, for different values of $\boldsymbol{A}$, using numerical quadrature in $\mathbb{R}^d$. The integrand in this quadrature is an inexpensive function whose value is independent of the physics of the problem or the accuracy of the discretization scheme of the space-time continuum (once $\boldsymbol{A}$ is known).

### 4.4.3 Adapted Representation of Model Parameters

It is reasonable to expect that different quantities of interest should poll the parameters of the model differently for relevant information. Given the mathematical formalism adopted in this paper for model reduction, namely basis adaptation and orthogonal projection, this issue can be stated clearly and concisely. For that, we revisit the intrusive implementation in section (4.4.4.4.2) and we express the random operator $\mathcal{L}_{\boldsymbol{\xi}}$ in terms of its polynomial chaos decomposition relative to $\boldsymbol{\eta}$, with equations (4.28) and (4.30) replaced by the following equations, respectively,

$$\mathcal{L}_{\boldsymbol{\alpha}} = (\mathcal{L}_{\boldsymbol{\xi}}, \psi_{\boldsymbol{\alpha}}^{\boldsymbol{A}}) \qquad \boldsymbol{\alpha} \in \mathcal{I} \,, \tag{4.31}$$

and

$$\sum_{\boldsymbol{\beta}\in\mathcal{I}} \sum_{\boldsymbol{\alpha}\in\mathcal{I}_p} \mathcal{L}_{\boldsymbol{\alpha}}[u_{\boldsymbol{\beta}}] \; \left(\psi_{\boldsymbol{\alpha}}^{\boldsymbol{A}} \psi_{\boldsymbol{\beta}}^{\boldsymbol{A}}, \psi_{\boldsymbol{\gamma}}^{\boldsymbol{A}}\right) = 0 \quad \forall \boldsymbol{\gamma} \in \mathcal{I} \,, \tag{4.32}$$

where we have, in addition, accounted for the projection of the solution on $V_{\mathcal{I}}$. Since both $\boldsymbol{\eta}$ and $\boldsymbol{\xi}$ are standard orthogonal gaussian vectors,

$$\left(\psi_{\boldsymbol{\alpha}}^{\boldsymbol{A}} \psi_{\boldsymbol{\beta}}^{\boldsymbol{A}}, \psi_{\boldsymbol{\gamma}}^{\boldsymbol{A}}\right) = \left(\psi_{\boldsymbol{\alpha}} \psi_{\boldsymbol{\beta}}, \psi_{\boldsymbol{\gamma}}\right) \,, \tag{4.33}$$

and equation (4.32) can be rewritten as,

$$\sum_{\boldsymbol{\beta}\in\mathcal{I}} \sum_{\boldsymbol{\alpha}\in\mathcal{I}} \mathcal{L}_{\boldsymbol{\alpha}}[u_{\boldsymbol{\beta}}] \; (\psi_{\boldsymbol{\alpha}} \psi_{\boldsymbol{\beta}}, \psi_{\boldsymbol{\gamma}}) + \sum_{\boldsymbol{\beta}\in\mathcal{I}} \sum_{\boldsymbol{\alpha}\notin\mathcal{I}} \mathcal{L}_{\boldsymbol{\alpha}}[u_{\boldsymbol{\beta}}] \; (\psi_{\boldsymbol{\alpha}} \psi_{\boldsymbol{\beta}}, \psi_{\boldsymbol{\gamma}}) = 0 \quad \forall \boldsymbol{\gamma} \in \mathcal{I} \,. \tag{4.34}$$

If the multi-index set $\mathcal{I}$ is chosen to be orthgonal to its complement in $\mathcal{I}_p$ (which is indeed the case for the two proposed choices of the cases consi dered in sections 4.3.4.3.1 and 4.3.4.3.2, then the second double summation in the last equation vanishes, and one is left with a problem identical in structure to the original problem stated in equation (4.29). Thus, if we had the foresight of knowing the subset $\mathcal{I}$ before solving the problem, and expanding the model parameters along that multi-index set, the solution of that problem for the QoI would coincide with the solution for the QoI obtained from a full-dimensional solution of the problem. In general, however, it is not possible to expand the model parameters, ahead of time, with respect to the multi-index set $\mathcal{I}$ since this set characterizes the yet to be determined solution. The value of the foregoing analysis, however, is to recognize that to any given QoI adapted to the multi-index set $\mathcal{I}$, there exists a commensurate characterization of the model parameters, obtained via projection, that quantifies how much of these properties is needed for the accurate estimation of the QoI.

## 4.5 Numerical Example

We now consider two examples, demonstrating the applicability of the foregoing analysis to problems specified, respectively, by algebraic and partial differential equations.

### 4.5.1 Algebraic Equation

Consider the function $\hat{h} : \mathbb{R}^d \mapsto \mathbb{R}$ given by the expression,

$$\hat{h}(\hat{\xi}_1, \cdots, \hat{\xi}_d) = \frac{d}{1+b} \frac{\prod_{i=1}^{d}(1 + a_i\hat{\xi}_i + b_i\hat{\xi}_i^2)}{\sum_{j=1}^{d}\prod_{\substack{i=1\\ i\neq j}}^{d}(1 + a_i\hat{\xi}_i + b_i\hat{\xi}_i^2)} \tag{4.35}$$

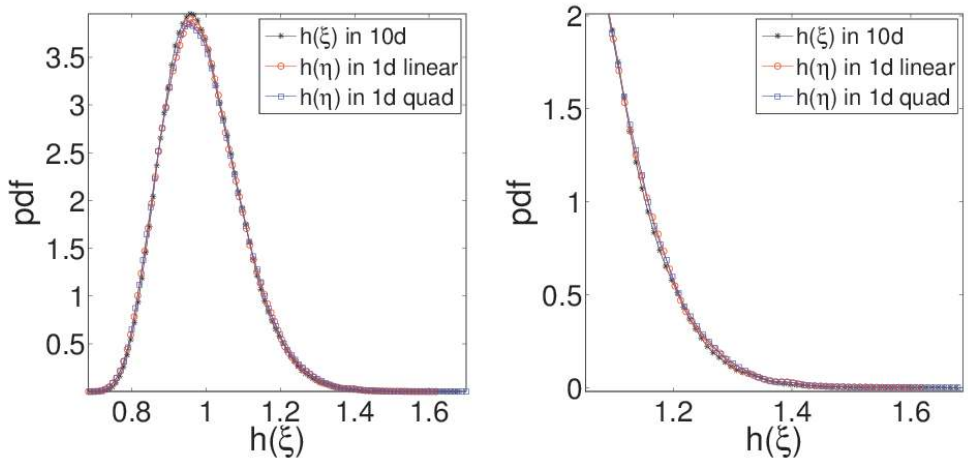



Figure 4.1: Probability density function of $h(\boldsymbol{\xi})$. Results using a full 10-dimensional solution and the reduced approximations using prior knowledge on the Gaussian content and quadratic content of QoI.

which represents the effective modulus for $d$ springs in series, each of modulus $(1 + a_i\hat{\xi}_i + b_i\hat{\xi}_i^2)$. Correlation is introduced between the $\mathcal{N}(0,1)$ Gaussian random variables $\hat{\xi}_i$ in the form of $E\{\hat{\xi}_i\hat{\xi}_j\} = \sigma^2 e^{-|i-j|/l}$. A Karhunen-Loeve expansion is first affected to transform the set $\{\hat{\xi}_i\}$ into a set of independent random variables $\{\xi_i\}$. All $d$ terms are retained in the resulting expansion, and the induced mapping, $h$, has an $L_2$ expansion with respect to the Hermite polynomials with coefficients $\{h_{\boldsymbol{\alpha}}\}$. Evaluating the coefficients in the full third order 10-dimensional expansion required a level 5 Smolyak quadrature (8761 function evaluations). First and second order expansions are obtained by specifying, respectively, $\mathcal{I}_p = \mathcal{I}_1$ and $\mathcal{I}_p = \mathcal{I}_2$. Coefficients in these expansions can be obtained via level 2 (21 function evaluations) and level 3 (221 function evaluations) Smolyak quadratures, respectively. Matrix $\boldsymbol{A}$ is then obtained as indicated by equations (4.11) and (4.17), and the set $\mathcal{I}$ is identified, respectively, with $\mathcal{E}_1$ and $\mathcal{E}$. The numerical results below are associated with statistically identical springs with $b_i = b = 1$, $a_i = 0.5$, $\sigma = 0.05$ and

$p = 3$. The correlation length of the springs was taken as $l = 4$ We note that the effective modulus becomes increasingly Gaussian as the correlation length approaches zero. Figure (4.1) shows the results for the probability density function of $h$, for both linear and quadratic prior knowledge. The quality of the approximation in the tail region is also shown.

### 4.5.2 Linear Elasticity with Nonlinear Quantities of Interest

As a second example, consider an elastic material occupying a square domain and subjected to a uniform pressure along its boundary with an empty circular inclusion. Due to symmetry, only a quarter of the domain is analysed with suitable boundary conditions, as depicted in Figure (2). The material behavior is assumed to be characterized in the realm of linear elasticity as a plane stress problem. It is further assumed that the modulus of elasticity of the medium is a spatially-varying stochastic process while Poisson ratio is a deterministic constant over space. This assumption is clearly inconsistent with linear elasticity since the modulus of elasticity and Poisson ratio are both dependent on the same microstructure. Our working assumption, however, is consistent with observations of much smaller statistical scatter in Poisson ratio than the modulus of elasticity, and can be viewed as resulting in a model that is a perturbation to mathematical elasticity. The governing equations are discretized using the finite element method according to the mesh shown in Figure (4.2). The modulus of elasticity is assumed to be the exponential of a Gaussian field and is defined by its normalized spatially uniform mean value, $E_0 = 1N/m^2$., an isotropic exponentially decaying covariance function with correlation coefficient equal to $L/8$ where $L = 1m$ is the length of one side of the square domain, and a coefficient of variation equal to $0.5$. Poisson ratio is taken to be $0.25$, and the pressure along the boundary with the inclusion is taken as $0.1\ N/m^2$. A Karhunen-Loeve expansion is used to discretize the Gaussian

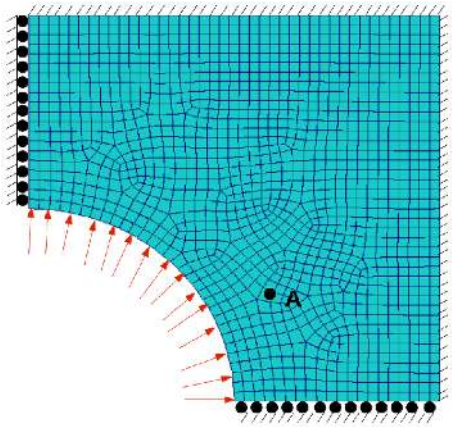



Figure 4.2: Elastic domain with random Young's modulus and subjected to uniform internal pressure. Point $A$ (0.2977,0.1417) is highlighted.

process associated with the modulus of elasticity, with 10 terms retained in its expansion ($d = 10$). The exponential of this representation is then developed in a $3^{rd}$ order 10-dimensional Hermite polynomial expansion. The solution to the elasticity equations with this representation of material properties consists of two displacement fields (one along each spatial dimension) that form a 2-variate spatial stochastic process. A polynomial chaos expansion of this process up to order 3 in all 10 dimensions is carried out using level-5 Smolyak quadrature, as a benchmark against which to verify the model reduction results. This provides our estimate for the process $u$. We then focus attention on two separate scalar quantities of interest (QoI). The first QoI is a linear functional of $u$ and consists of the displacement in the horizontal direction at one point in the domain. The second QoI is a nonlinear functional and consists of the maximum vonMises stress over the two dimensional spatial extent of the problem. VonMises stress, $\sigma$, at any given point in the material is given as $\sigma = \sqrt{\sigma_1^2 - \sigma_1\sigma_2 + \sigma_2^2}$ where $\sigma_1$ and $\sigma_2$ are the two principal stresses at that location.

As a motivation to the reduction methodology developed in this paper, we computed the Karhunen-Loeve expansion of the Gaussian component of the $x$-displacement process, and used the Gaussian random variable associated with its dominant mode to construct a 1-dimensional

polynomial chaos approximation of the first QoI described above. Figure (4.3) compares the results from this 1-dimensional approximation against the full 10-dimensional evaluation of the QoI and the approximations according to the two methods described in this paper. It is clear that the reduction using the dominant KL variable fails to capture the probabilistic measure of the QoI, emphasizing the need for more rational adaptation strategies. The probability density functions of the first QoI obtained using the two procedures described earlier in the paper are shown in the same figure. The linear (Gaussian) approximation is obtained using a level 2 Smolyak quadrature whereas the quadratic approximation required a level 3 rule. The match is amazing even in the tail of the distributions, using just the one-dimensional approximation. Figure (4.4) shows a comparison of the polynomial chaos coefficients associated with a 10-dimensional approximation and the approximation obtained using a sequence of 10 one-dimensional solutions adapted to the quadratic components of QoI. It is clear that the terms being neglected in the proposed $1d$ reduction have minor contribution to the $L_2$ error.

Figure (4.5) shows analogous results for the second, nonlinear, QoI. In this case, an approximation built around the Gaussian content of the QoI is insufficient to completely characterize its probability density function. The reduced model obtained by including information from the second order terms, however, provides an excellent approximation.

The analysis presented in section 4.4.3 is applied to the present problem. Figure (4.6) shows the the first six polynomial chaos coefficients of Young's modulus associated with an expansion of the lognormal process. Note the global features in these coefficients. Figure (4.7) shows the same quantities for the problem adapted to the QoI specified by the vonMises stresses at point $A$. Note the localization around point $A$ in the polynomial chaos coefficients, induced by the quantity of interest.

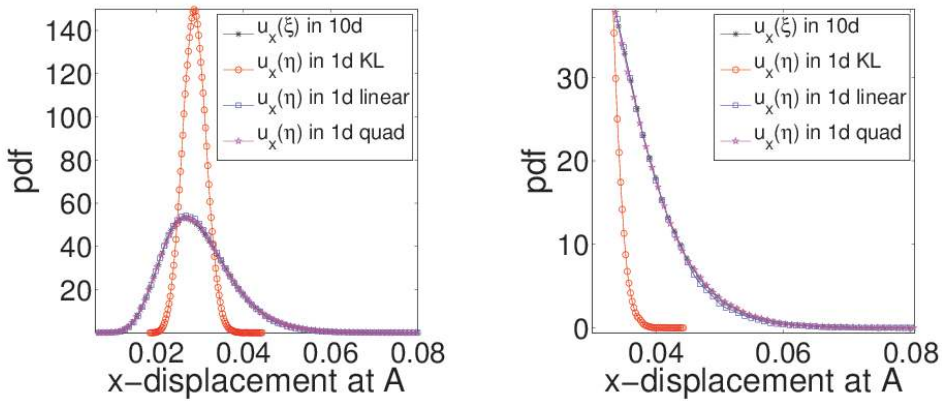



Figure 4.3: Probability density function of $x$–displacement at point $A$. Results using a 1) full 10-dimensional solution and the approximations using the 2) dominant component in the Karhunen-Loeve expansion of the stochastic process associated with the solution, 3) the 1-d approximation associated with Gaussian prior knowledge, and 4) sequence of 10 1-d approximations associated with quadratic prior knowledge.

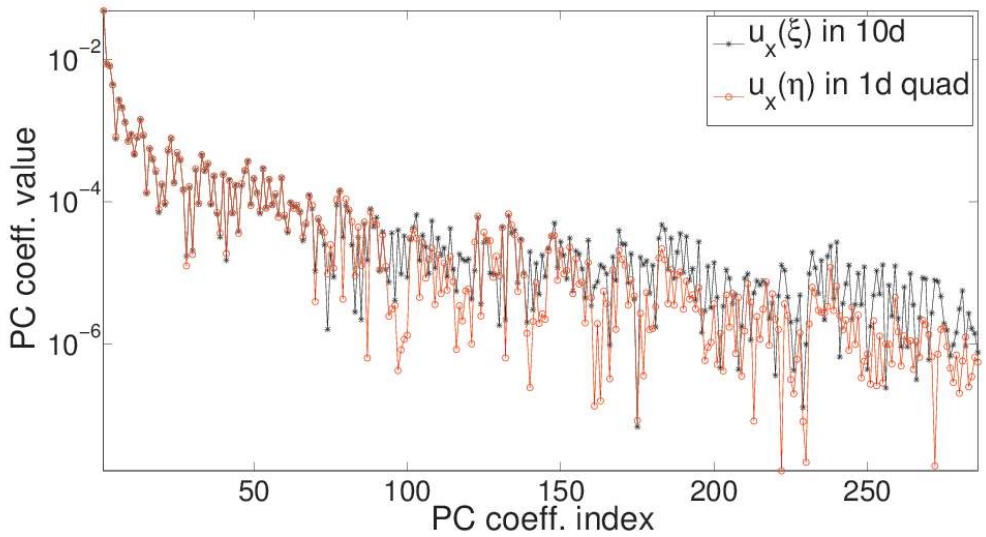



Figure 4.4: Coefficients of the PC expansion of the $x$-displacement at point $A$. Results using a full 10-dimensional solution and the reduced approximation using prior knowledge on the quadratic content of QoI.

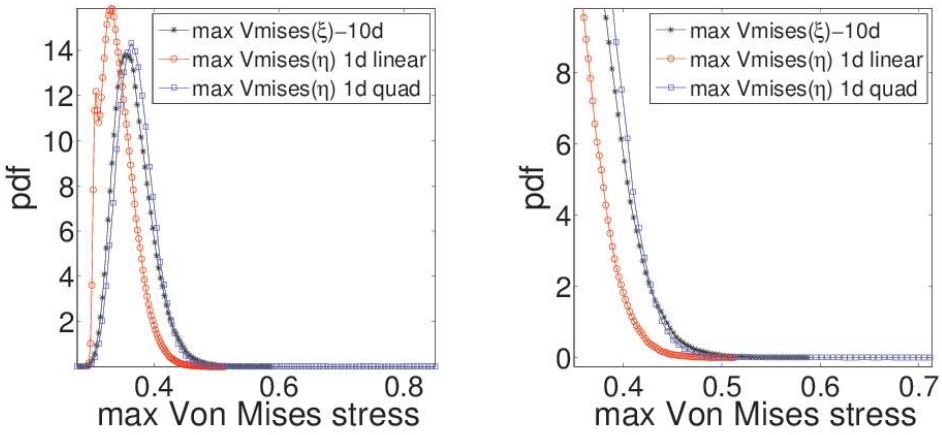



Figure 4.5: Probability density function of the maximum vonMises stress over the spatial domain of the problem. Results using 1) full 10-dimensional solution and the approximations using the 2) 1-d approximation associated with Gaussian prior knowledge, and 3) sequence of 10 1-d approximations associated with quadratic prior knowledge.

This correspondance between a QoI and certain features of model parameters should have implications on the characterization of material properties from experimental data and could serve as a guide for the development of adapted regularization strategies for the associated inverse problem. That invesigation, however, is beyond the scope of the present work where we limit ourselves at noting the presence of such an interplay.

### 4.5.3 Random eigenvalue problem

Eigenvalue analysis has very important application in structural dynamics to find fundamental frequency of the dynamical system and and corresponding eigenmodes. When parameters of the system of the system are random, the eigenvalue problem becomes, random eigenvalue analysis.

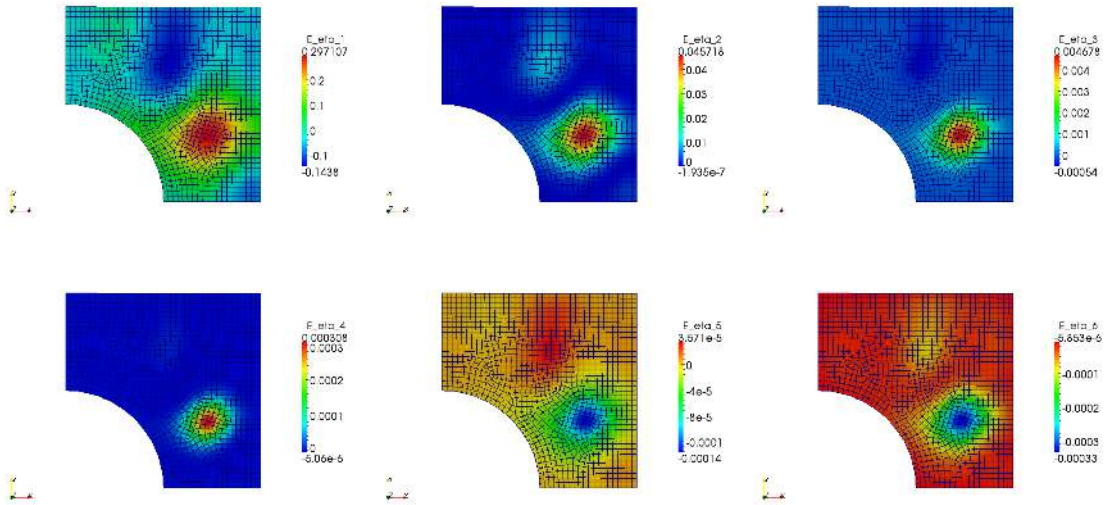


Figure 4.6: The first six coefficients of the elasticity modulus with respect to a polynomial chaos expansion in the original gaussian variables $\boldsymbol{\xi}$.

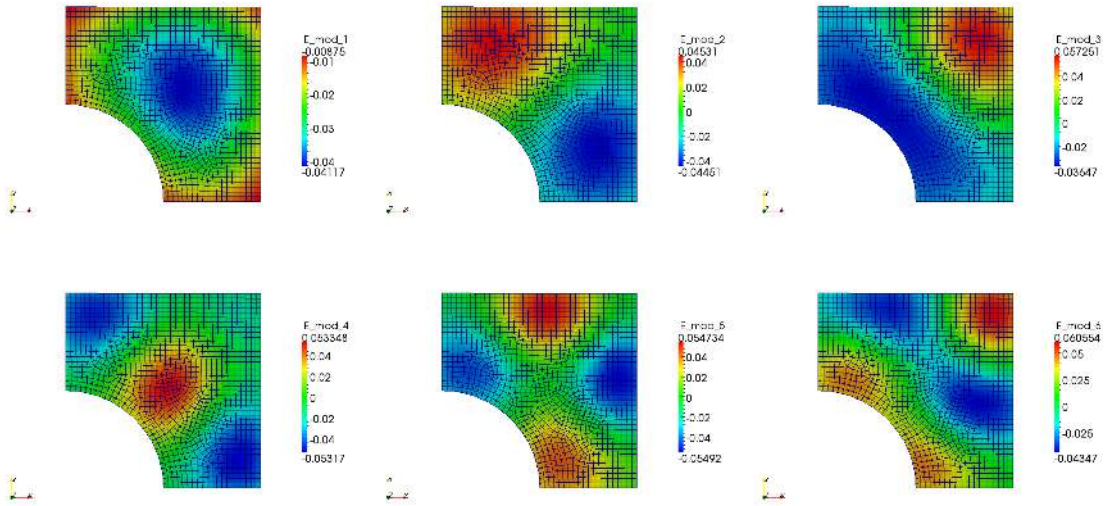


Figure 4.7: The first six coefficients of the elasticity modulus with respect to a polynomial chaos expansion in the one-dimensional gaussian variable $\eta_1$.

**Deterministic eigenvalue problem**

Let $\bar{K} \in \mathbb{R}^{n\times n}$ and $\bar{M} \in \mathbb{R}^{n\times n}$ be stiffness and mass matrices obtained from finite element discretization. To find fundamental frequency and corresponding mode shapes of the dynamical system, following eigenvalue problem has to be solved,

$$\bar{K}\bar{\phi} = \bar{\lambda}\bar{M}\bar{\phi}, \quad \bar{\lambda} \in \mathbb{R}, \quad \bar{\phi} \in \mathbb{R}^n, \tag{4.36}$$

where, $\lambda$ and $\phi$ are eigenvalue and eigenvector respectively.

**Random eigenvalue problem**

When there is uncertainty in the system, then the eigenvalue problem becomes following random eigenvalue problem,

$$K(\theta)\phi(\theta) = \lambda(\theta)M(\theta)\phi(\theta), \quad \theta \in \Omega$$

where, $K(\theta) \in \mathbb{R}^{n\times n}$, $M(\theta) \in \mathbb{R}^{n\times n}$ are random stiffness and mass matrices, $\lambda(\theta) \in \mathbb{R}, \phi(\theta) \in \mathbb{R}^n$ are random eigenvalue and corresponding random eigenvector and $(\Omega, \mathcal{F}, \mu)$ is underlying probability space with sample space $\Omega$, $\sigma$-algebra, $\mathcal{F}$ and probability measure $\mu$ such that, each sample point $\theta \in \Omega$.

**Polynomial chaos approach**

Random eigenvalue problem discussed in section 4.5.3 can be solved using polynomial chaos expansion method discussed in section 2. Stiffness matrix $K$, mass matrix $M$, eigenvalue $\lambda$ and eigenvector $\phi$ can be expanded in terms of polynomial chaos in $L_2(\Omega, \mathcal{F}, \mu)$ as $K = \sum_{i=0}^{P-1} K_i\psi_i(\boldsymbol{\xi})$, $M = \sum_{i=0}^{P-1} M_i\psi_i(\boldsymbol{\xi})$, $\lambda = \sum_{i=0}^{P-1} \lambda_i\psi_i(\boldsymbol{\xi})$ and $\phi = \sum_{i=0}^{P-1} \phi_i\psi_i(\boldsymbol{\xi})$, where $\boldsymbol{\xi} = \{\xi_i, \cdots, \xi_d\}^T$.

Random eigenvalue problem can be solved using stochastic Galerkin or stochastic collocation methods. Stochastic Galerkin projection of random eigenvalue problem results in following coupled system of equations,

$$\sum_{i=0}^{P-1}\sum_{j=0}^{P-1} E\{\psi_i\psi_j\psi_k\}K_i\phi_j = \sum_{i=0}^{P-1}\sum_{j=0}^{P-1} E\{\psi_i\psi_j\psi_l\psi_k\}\lambda_i M_j\phi_l, \quad k = 0, \cdots, P-1. \tag{4.37}$$

Stochastic collocation method involves solving deterministic eiegenvalue problems at a given set of quadrature points.

$$K(\boldsymbol{\xi}^{(q)})\phi(\boldsymbol{\xi}^{(q)}) = \lambda(\boldsymbol{\xi}^{(q)})M(\boldsymbol{\xi}^{(q)})\phi(\boldsymbol{\xi}^{(q)}), \quad q = 1, \cdots, q_n$$

In this work stochastic collocation methods is used for testing basis adaptation methods for random eigenvalue problem. The basis adaptation methodology can be readily applied to random

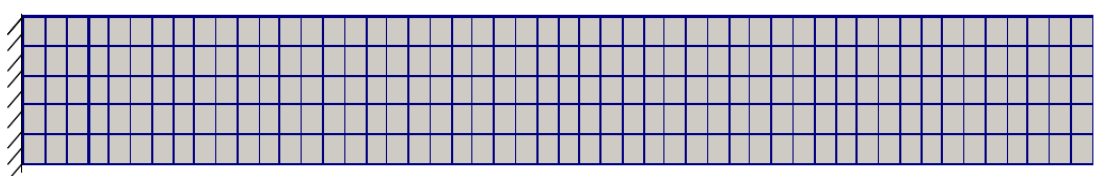


Figure 4.8: 2-d cantilever beam meshed with quad4 elements, is fixed on left end and eigenvalue analysis is performed

eigenvalue problem to represent the eigenvalue in one random variable and solve a one dimensional random eigenvalue problem to compute the eigenvalue accurately. These approaches are tested in two structural systems, a two dimensional cantilever beam shown in figure 4.8 and a wind turbine blade sgown in figure 4.9. Figure 4.10 shows the probability density functions of first six

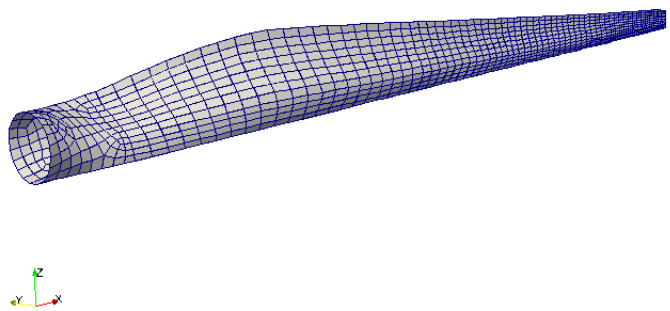
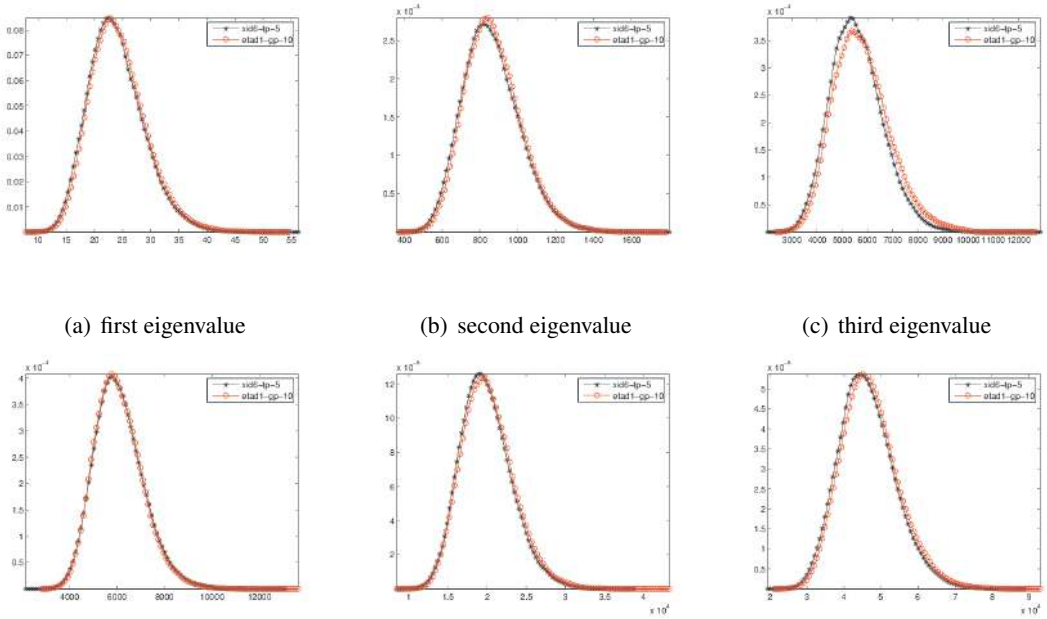



Figure 4.9: Wind turbine blade meshed with quad4 elements, is fixed in all 6 dof on one end and eigenvalue analysis is performed

eigenvalues of a two dimensional cantilever beam and figure 4.12 shows the pdfs of first six eigenvalues of the wind turbine. It can be observed from both cantilever beam and the wind turbine blade problems that the eigenvalues can be captured very accurately using one dimensional basis adaptation. Although the eigenvector corresponding to each eigenvalue can be computed after the eigenvalue is computed, gain in the computational effort has to be investigated. To adapt the basis adaptation to eigenvectos, let $\{\bar{\phi}_i\}_{i=1}^{r}$ be a set of deterministic eigenvectors, then a random eigenvector $\phi^{(l)}(\boldsymbol{\xi})$ cane be can be projected on to the deterministic set of nominal eigenvectore $\{\bar{\phi}_i\}_{i=1}^{r}$ to get coefficients,

$$\phi^{(l)}(\boldsymbol{\xi}) = \sum_{i=0}^{P-1} \phi_i^{(l)} \psi(\boldsymbol{\xi})$$

$$\phi^{(l)}(\boldsymbol{\xi}) = \sum_{j=1}^{r} \bar{\phi}_j^{(l)} \sum_{i=0}^{P-1} a_{ij}^{(l)} \psi_i(\boldsymbol{\xi}), \tag{4.38}$$

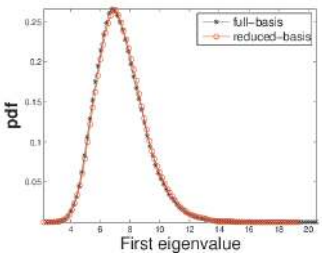



(a) first eigenvalue (b) second eigenvalue (c) third eigenvalue

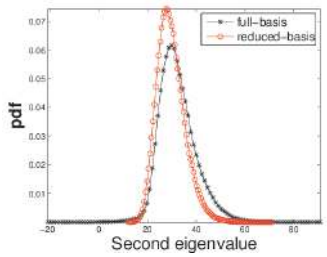
(d) fourth eigenvalue (e) fifth eigenvalue (f) sixth eigenvalue

Figure 4.10: pdf of eigenvalues of 2-d cantilever beam in 6d $\xi$ and in 1d $\eta$

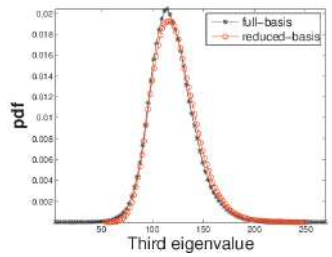
then, the coefficients, $\bar{\phi}_j^{(l)}$ can be computed as quantities of interest using basis adaptation methods. The first coefficient $\bar{\phi}_1^{(l)}$, polynomial chaos coefficients computed in reduced dimension form basis adaptation is in very good agreement with those computed in high dimension. However, there is a significant error in other coefficients. It could be due to the lower magnitude of the second and third coefficients.



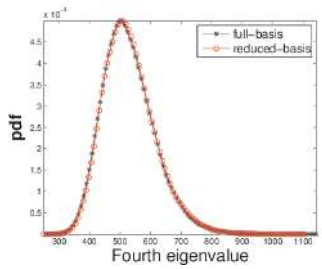

(a) first eigenvalue



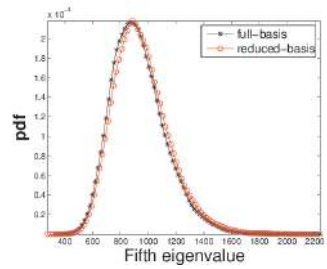

(b) second eigenvalue



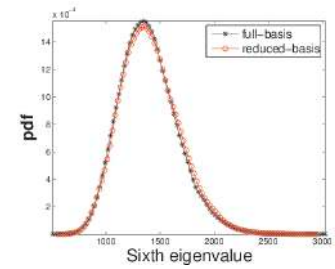

(c) third eigenvalue




(d) fourth eigenvalue

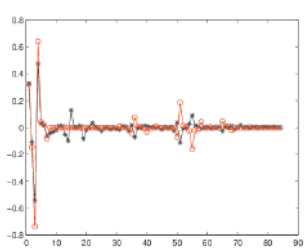



(e) fifth eigenvalue

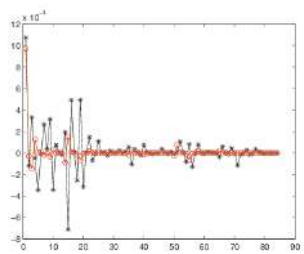



(f) sixth eigenvalue

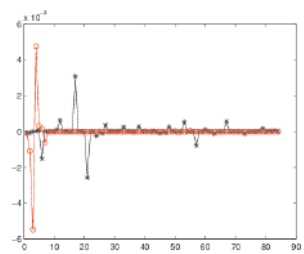
Figure 4.11: pdf of eigenvalues of wind turbine blade in 6d $\xi$ and in 1d $\eta$



(a) $\bar{\phi}_1^{(l)}$



(b) $\bar{\phi}_2^{(l)}$



(c) $\bar{\phi}_3^{(l)}$

Figure 4.12: Polynomial chaos coefficients of of $\bar{\phi}_j^{(l)}$ in 6d $\xi$ and in 1d $\eta$

# Chapter 5

# Stochastic upscaling

## 5.1 Introduction

In this chapter, stochastic upscaling methodology for a steady state laminar flow through an array circular solid discs and heat transfer explored. The fine scale problem is modeled using Navier-Stokes equations and solid discs act as heat source (sink). The uncertainty in the system is modeled by treating thermal conductivity of the solid discs as a random filed. At coarse scale the medium is modeled as a porous medium and Darcy-Brinkman equations are used. Due to temperature dependent fluid density and fluid viscosity, the uncertainty in the fine scale is propagated to fluid flow and parameters such as permeability and thermal conductivity at the coarse scale. Volume average method is adopted to compute the permeability and thermal conductivity at the coarse scale from fine scale solution. Polynomial chaos expansion methodology is used to solve stochastic partial differential equations at fine and coarse scales. Preconditioners developed in chapter 3 are used to solved stochastic Galerkin system of equations efficiently. Basis adaption methods developed in chapter 4 used to further reduce the stochastic dimension while

computing the coarse scale parameters, namely, for stochastic dimension reduction for computing permeability random field and thermal conductivity random field.

## 5.2 Problem statement

Many physical problems can be described with coupled stochastic nonlinear partial differential equations. One such problem of interest is fluid flow through porous media coupled with heat equation. At fine scale the porous media problem can be solved using Navier-Stokes equation. However, it is computationally very expensive to solve using Navier-Stokes equations with random parameters. For this reason, the system upscaled to porous media equations and parameters of the porous media are computed from fine scale solutions. Uncertainty at the fines scale is modeled by treating the thermal conductivity of the solid discs as a log-normal random field. The uncertainty at the fine sale is propagated to the coarse scale which can be modeled by treating the permeability and thermal conductivity as random parameters. To solve these stochastic partial differential equations at fine and coarse scales, use spectral stochastic finite element method Ghanem and Spanos (1991) is used. Stabilized finite element methods Tezduyar (1992) are used to circumvent the problem of instability due to spurious pressure modes that arise due to equal order interpolation for pressure and velocity fields in finite element discretization of incompressible fluid flow equations.

Figure 5.1 shows fine scale and coarse scale domain modeled in 2-dimensions.

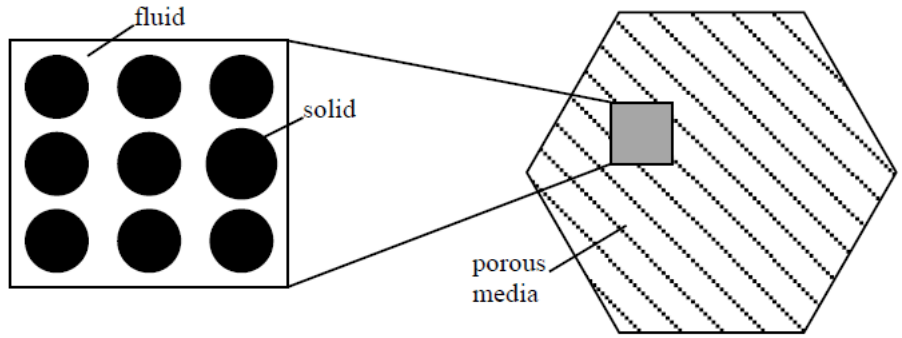



Figure 5.1: Fine scale domain modeled with an array of solid discs and coarse scale domain modeled as porous media

### 5.2.1 Navier-Stokes equations at fine-scale

At fine scale, two dimensional system is modeled as an incompressible flow though an array of solid circular discs. Heat transfer between solid circular discs and the fluid is modeled through a coupled heat equation. As the fluid flows through the space between the discs, heat is transferred from discs to the fluid. Uncertainty due to thermal conductivity of the solid discs is propagated to the fluid velocity, pressure and temperature through, temperature dependent fluid viscosity and fluid density. This physics at the fine scale can be modeled with Navier-Stokes equations Nakayama et al. (2002) and heat equation as

$$\rho_f(\nabla \cdot \boldsymbol{u}) = 0 \tag{5.1}$$

$$\rho_f \left(\nabla \cdot \boldsymbol{u}\right) \boldsymbol{u} = -\nabla P + \mu_f \nabla^2 \boldsymbol{u} + \rho_f \boldsymbol{f} \tag{5.2}$$

$$\rho_f c_{p,f} \left[\nabla \cdot (\boldsymbol{u} T)\right] = k_f \nabla^2 T \tag{5.3}$$

where, equations 5.1, 5.2 and 5.3 are the continuity, conservation of momentum (Navier-Stokes) and energy equations respectively. In these equations, $\boldsymbol{u}$ is the velocity vector of the fluid, $P$ is pressure in the fluid, $T$ is the temperature and $\boldsymbol{f}$ is the forcing term.

### 5.2.2 Darcy-Brinkman equations at coarse-scale

At coarse scale the domain is treated as a porous medium with heat source in the solid region and the physics is modeled with the Darcy-Forchheimer equation Schmidt et al. (2010). The equations governing flow through porous media are as follows,

$$\rho_f(\nabla \cdot \tilde{\boldsymbol{u}}) = 0 \tag{5.4}$$

$$\rho_f \frac{\rho_f}{\phi}(\nabla \cdot \tilde{\boldsymbol{u}})\tilde{\boldsymbol{u}} = -\phi \nabla \tilde{P} + \mu_f \nabla^2 \tilde{\boldsymbol{u}} - \frac{\phi \mu_f}{K}\tilde{\boldsymbol{u}} + \phi \rho_f \tilde{\boldsymbol{f}} \tag{5.5}$$

$$\nabla \cdot \left(\bar{\rho}\bar{C}\tilde{\boldsymbol{u}}\bar{T}\right) = \bar{k}_{eff}\nabla^2\bar{T} + \dot{\bar{Q}} \tag{5.6}$$

where, equations 5.4 and 5.5 are the continuity and momentum equations and 5.6 is the heat equation under thermal equilibrium between fluid and solid phases.

where, $\tilde{\boldsymbol{u}}$ is the velocity of the fluid, $\tilde{P}$ is the pressure in the fluid, $\phi$ is the porosity of the medium, $\bar{C}$ is the specific heat $K$ is the permeability, $\bar{k}_{eff}$ is effective heat conductivity of porous media and $\dot{Q}_s$ is heat source in the solid.

At the coarse scale, uncertainty in the system can be represented by treating the permeability and thermal conductivity as random parameters. To solve these stochastic partial differential equations at fine and coarse scales, spectral stochastic Galerkin projection method Ghanem and Spanos (1991) will be used. Stabilized finite element methods Shadid et al. (2006), Tezduyar (1992) are used to circumvent the problem of spurious pressure modes that arises due to equal order interpolation in solving incompressible Navier-Stokes equations.

## 5.3 Stochastic upscaling

Stochastic upscaling is achieved by computing the coarse scale random parameters using volume average method from the fine scale solution. Further, stochastic basis adaptation methods are used to represent the coarse scale parameters in reduced stochastic dimension.

### 5.3.1 Permeability

The coarse scale permeability is computed by volume average method Nakayama et al. (2002). At low velocity the Darcy law can be used to compute the permeability

$$-\frac{\partial \langle P \rangle}{\partial x_i} = \mu k_{ij}^{-1} \langle u_j \rangle \tag{5.7}$$

Where, $\langle P \rangle$ and $\langle u \rangle$ are volume averaged pressure and velocity obtained from fine scale solution. Volume average is computed as

$$\langle a \rangle = \frac{1}{V} \int_V a dV \tag{5.8}$$

where $V$ is the representative volume considered for averaging. The solution of the fine scale stochastic partial differential equations are obtained using non-intrusive Smolyak quadrature method. At each quadrature point, a deterministic fine scale problems is solved and fine scale, velocity, pressure and temperature distribution are obtained. From the velocity and pressure solution, a permeability filed is constructed for each solution. Due to very small variability of pressure gradient is y-direction, the permeability is considered only along x-direction. From the quadrature points, polynomial chaos expansion of the permeability filed and higher order statistics can be readily computed. Figure 5.8, shows the mean and standard deviation of the permeability field. Thus the uncertainty due to thermal conductivity in the fine scale problem is propagated to the coarse scale problem through the coarse scale permeability parameter. Further, the covariance function of the permeability field is constructed and Karhunen-Loève analysis is performed to find the dominant eigenvalues and eigenmodes.

### 5.3.2 Thermal conductivity

Thermal conductivity of the porous media is upscaled from the finescale heat equations. At coarse sclae thermal conductivity relates the Darcy velocity to the temperature distribution at the coarse scale. The coarse scale heat equation is written as

$$\nabla \cdot \left(\bar{\rho}\bar{C}\tilde{\boldsymbol{u}}\bar{T}\right) = \bar{k}_{eff}\nabla^2\bar{T} \tag{5.9}$$

Above relation can be rewritten as

$$\bar{\rho}\bar{C}\nabla \cdot \left(\tilde{\boldsymbol{u}}\bar{T}\right) = \bar{k}_{eff}\nabla \cdot (\nabla\bar{T}) \tag{5.10}$$

Where, $\bar{\rho}$ and $\bar{C}$ density of the fluid and specific heat at the coarse scale. $\bar{k}_{eff}$ is effective thermal condictivity of the porous media $\tilde{u}$ is the coarse scale velocity and $\bar{T}$ is the coarse scale temperature. Writing above equation interms of fine scale solution,$u$ and $T$,

$$\bar{\rho}\bar{C}\int_D \nabla\cdot(\boldsymbol{u}T)\,dA = \bar{k}_{eff}\int_D \nabla\cdot(\nabla T)dA. \tag{5.11}$$

Using Gauss-divergence theorem, area integrals in the above equations can be converted intoline integrals as follows,

$$\bar{\rho}\bar{C}\int_C \vec{n}\cdot(\boldsymbol{u}T)\,ds = \bar{k}_{eff}\int_C \vec{n}\cdot(\nabla T)ds, \tag{5.12}$$

where, $\boldsymbol{u}$ and $T$ are velocity and temperature from the fine scale solution.

## 5.4 Basis adaptation to find QoI

In this section, we discus the basis adaptation methods to compute the permeability random filed in reduced dimension. From the volume average method discussed in section 5.3, permeability random field $k(x,\xi)$ can be computed from the fine scale solution as following polynomial chaos expansion,

$$k(x,\xi) = \sum_{i=0}^{p} k_i(x)\psi(\boldsymbol{\xi}), \tag{5.13}$$

where, $\boldsymbol{\xi} = \{\xi_1, \cdots, \xi_d\}^T$ is the input random vector obtained from truncated Karhunen-Loeve expansion of the thermal conductivity random field at the fine-scale. Thermal conductivity random field is modeled with log-normal distribution and hence, it can be expanded in terms of Hermite polynomials of standard Gaussian random variables $\boldsymbol{\xi}$.

### 5.4.1 Linear basis adaptation

Linear basis adaptation for permeability random field is pursued by computing a new set of random variables $\boldsymbol{\eta} = A\boldsymbol{\xi}$ using an isometry $A$ such that $AA^T = I$. $A$ is not unique and one way to construct $A$ is using the Gaussian components in the PC expansion of the quantity of interest (QoI). In the current problem, the permeability $k(x, \boldsymbol{\xi}$ at a spatial location $x_j$ can be computed as a QoI in the reduced dimension. That is ,

$$\eta_1 = \sum_{i=1}^{d} k_i(x_j)\xi_i, \tag{5.14}$$

here, $k_i(x_j)$, after normalization forms the first row of matrix $A$. Rest of the rows in the matrix $A$ can be constructed using Gram-Schmidt process. Now the permeability at location, $x_j$ can be constructed in reduced stochastic dimension $\boldsymbol{\eta}$, with very few randm variables. However, using linear basis adaptation, new set of $\boldsymbol{\eta}$ has to be computed for each spatial location.

### 5.4.2 Quadratic basis adaptation

Quadratic basis adaptation procedure can be pursued when, the quadratic terms in the polynomial chaos expansion QoI are known. In this case the isometry matrix $A$ is obtained as follows,

$$ASA^T = D, \tag{5.15}$$

, where $D$ is the diagonal matrix, whose elements $d_i$ are eigenvalues and $A$ is the matrix of eigen vectors of symmetric matrix $S$. Here, matrix $S$ contains the quadratic terms in the PC expansion of the QoI. This transformation eliminates, the cross terms in the PC expansion and hence reduces the d-dimensional stochastic Galerkin system in $\boldsymbol{\xi}$ to d one dimensional systems in $\eta_i, i = 1, \cdots, d$. As seen in the case, linear basis adaptation, quadratic adaptation also, uses new set of $\boldsymbol{\eta}$ at each location and hence, it is difficult to construct entire permeability random field simultaneously. To address this issue, we pursue, basis adaptation using, KL expansion of the Gaussian part of the random filed which is discussed in the next section. Figures 5.10 show the permeability at points 3 and 20, computed using linear and quadratic basis adaptation separately.

### 5.4.3 Basis adaptation with KL expansion

When, the QoI is a scalar, it can be effectively represented in one random variable, using linear or quadratic basis adaptations discussed in the previous sections. However, if the QoI is a random field, above methods can't be used to simultaneously construct entire random filed. For this reason we use, when the Gaussian part of the permeability random field $k_g = k_0(x) + \sum_{i=1}^{d} k_i(x)\xi$ is known, we construct the isometry from KL expansion of the covariance $cov_{k_g}$ of the Gaussian part of the permeability field,

$$cov_{k_g}(x_1, x_2) = \sum_{i=1}^{d} k_i(x_1)k_i(x_2), \tag{5.16}$$

and

$$k_g(x, \boldsymbol{\xi}) = \tilde{k}_0 + \sum_{i=1}^{d} \sqrt{\gamma_i}\tilde{k}_i(x)\eta_i, \tag{5.17}$$

where, $\eta_i$ are the new set of standard normal random variables and can be computed as

$$\eta_i = \frac{1}{\sqrt{\gamma_i}} \int k_g(x, \boldsymbol{\xi}) \tilde{k}_i(x) dx, \tag{5.18}$$

Figure 5.9(b) shows the decay, of the eigenvalues ($\gamma_i$). It can be observed that the decay of random variable is very sharp and very few number (2-3) of random variables is sufficient to represent the entire permeability random field. Figure 5.12 shows the pdf of permeability computed at a point using, full ten dimensional $\boldsymbol{\xi}$, reduced one, two and three dimensional $\boldsymbol{\eta}$. It can be observed that, three random, variables $\eta_1$, $\eta_2$ and $\eta_3$ are sufficient to capture the pdf accurately.

## 5.5 Results and discussion

A two dimensional fluid flow problem is considered with the geometry shown in figure 5.2. Red circles are the solid discs with heat source and the rest of the domain is fluid flow domain. Thermal conductivity of the solid discs is modeled as a log normal random field with mean 150 and coefficient of variation 1.0. Temperature dependent fluid density and viscosity are used to maintain a coupling between fluid flow and heat equations. A Neumann boundary condition of velocity, $50cm/s$ is applied on the left side boundary and velocities at top and bottom boundaries are provided as zero Dirichlet boundary condition. Heat source in the solid discs is provided by specifying a Dirichlet boundary condition at a small region inside the solid disc. Figure 5.3 shows two realizations of the thermal conductivity. Figure 5.9(a) shows the decay of eigenvalues from Karhunen-Loève analysis of thermal conductivity random filed and figure 5.9(b) shows the decay of eigenvalues from Karhunen-Loève analysis of the Gaussian part of the permeability random filed at the coarse scale. Figures 5.4, 5.6 and 5.7 show the mean and standard deviation of the

velocity in x-direction, pressure and temperature respectively. Figures 5.8(a) and 5.8(b) show the mean and standard deviation of the coarse scale permeability.

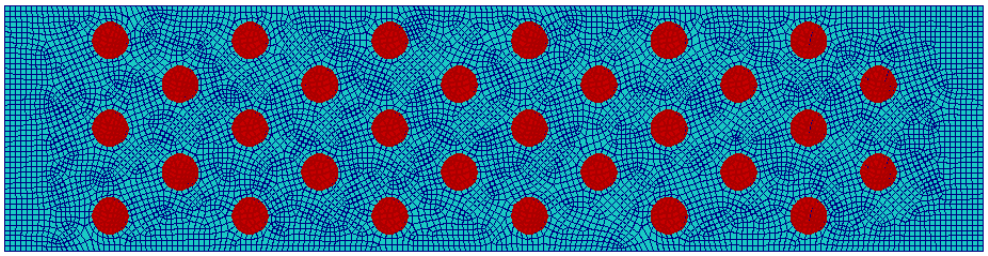


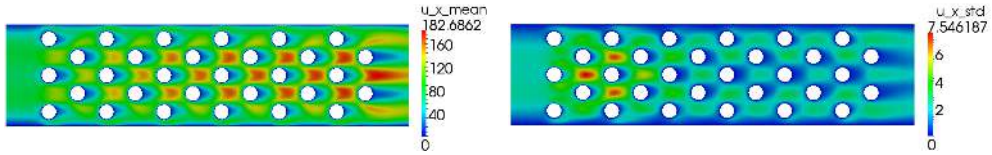
Figure 5.2: Fluid and solid regions meshed with 2-d quad4 elements


(a) Realization 1


(b) Realization 2

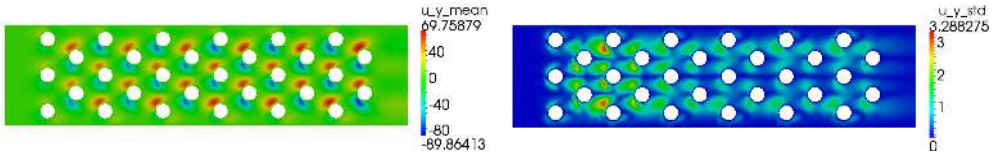
Figure 5.3: Realizations of the thermal conductivity in the circular discs

Figure 5.10 shows the pdfs of permeability of the coarse scale parameter at two spatial locations (at point number 3 and 20) computed using 10-dimensional $\xi$-space and reduced 1-dimensional and 2-dimensional $\eta$-space. It can be observed the the coarse scale permeability



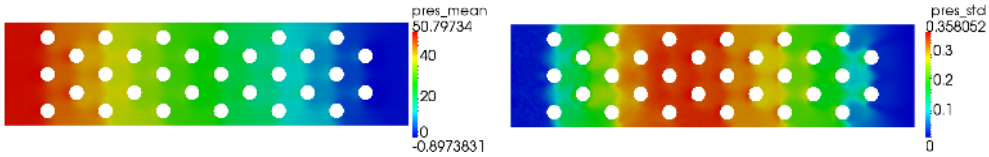

(a) Mean of $u_x$ (b) std. dev. of $u_x$

Figure 5.4: Mean and std. dev of velocity in x-direction in 10d $\xi$ and order 3



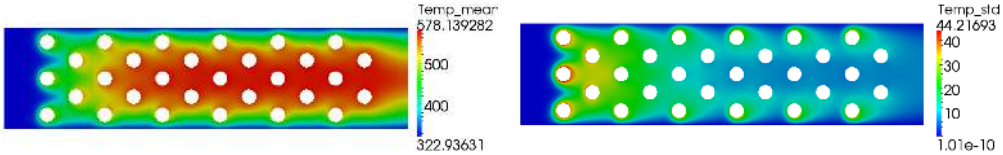

(a) Mean of $u_y$ (b) std. dev. of $u_y$

Figure 5.5: Mean and std. dev of velocity in y-direction in 10d $\xi$ and order 3




(a) Mean of pressure (b) std. dev. of presure

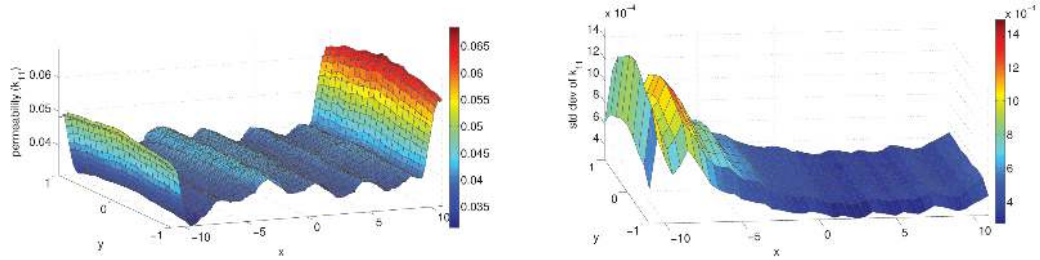
Figure 5.6: Mean and std. dev of pressure in 10d $\xi$ and order 3




(a) Mean of temperature (b) std. dev. of temperature

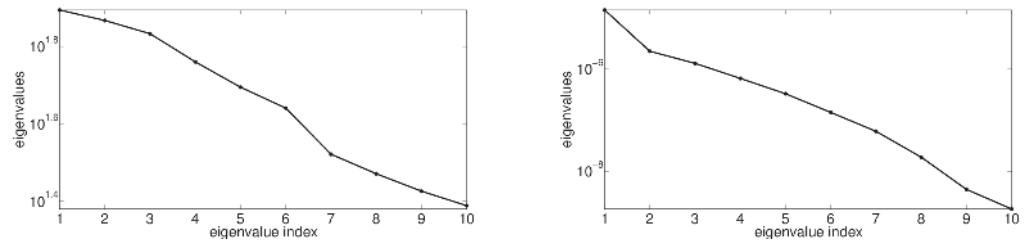
Figure 5.7: Mean and std. dev of temperature in 10d $\xi$ and order 3

parameter represented in reduced dimension is very accurate. However, with this method, a new set of $\eta$ has to be identified for each spatial locations. Alternately, figure 5.12 shows the pdfs of permeability computed in reduced dimension using basis adaptation based on Karhunen-Loève




(a) mean of $k_{11}$ (b) std. dev. of $k_{11}$

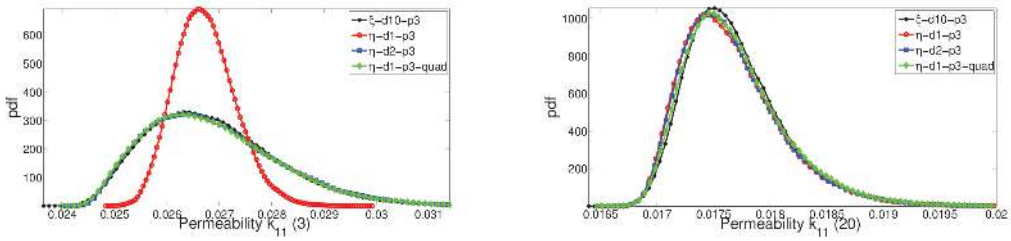
Figure 5.8: Mean and std. dev. of permeability component $k_{11}$ using volume averaging




(a) mean of $k_{11}$ (b) std. dev. of $k_{11}$

Figure 5.9: Eigenevalues obianed from KL expansion of thermeal conductivity at finescale and the eigen values obtained from KL analysis of Gaussian part of permeability at coarse scale

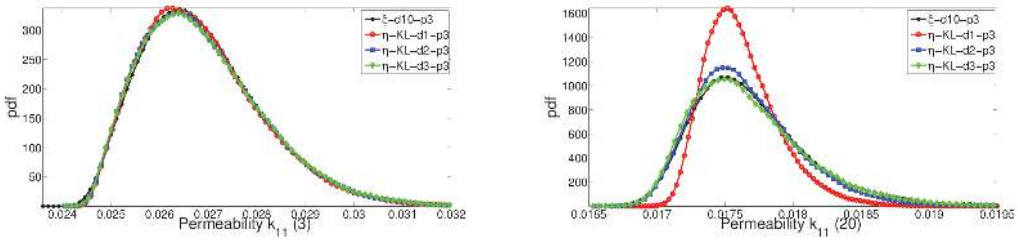
expansion as discussed in section 5.4.3. Using this method, entire permeability is computed simultaneously, but needs 3-dimensional space with $\eta_1, \eta_2$ and $\eta_3$. Figure 5.12 shows thermal conductivity computed at point 20 using the basis adaptation method based on Karhunen-Loève expansion.




(a) Permeability at point 3

(b) Permeability at point 20

Figure 5.10: Permeability computed at points 3 and 20, using linear and quadratic basis adaptation methods. Here, the basis adapted for each spatial point (QoI) separately




(a) Permeability at point 3

(b) Permeability at point 20

Figure 5.11: Permeability computed at points 3 and 20 using basis adaptation method, where new random variables $\eta$ are obtained from Karhunen-Loeve analysis of the Gaussian part of the random filed. Here, basis is adapted for all spatial points (QoI) simultaneously.

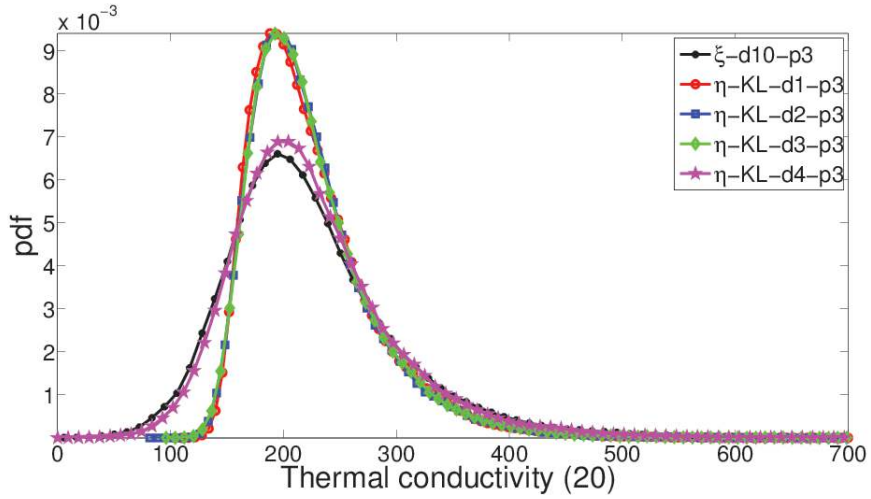



Figure 5.12: Thermal conductivity computed at points 20 using basis adaptation method, where new random variables $\eta$ are obtained from Karhunen-Loeve analysis of the Gaussian part of the random filed. Here, basis is adapted for all spatial points (QoI) simultaneously.

# Chapter 6

# High resolution micrograph synthesis

## 6.1 Introduction

Recent capabilities in the multi-scale modeling and simulation of material behavior have been matched by technologies for sensing and characterizing materials at distinct spatial scales. Several technical challenges ensue in the process of pairing multi-scale models to multi-scale data, one of which is addressed in this paper namely that of characterizing knowledge about multi-scale material structure in a manner that is conducive to probabilistic conditioning and updating.

In this work experimental data provided through micrographs are considered, which are digital images of the microstructure taken through a microscope, showing a magnified view of the material domain, and allowing digital representation of fine-scale features. In this paper, the terms "image", "texture" and "micrograph" are used interchangeably. High resolution microstructural information is required as a prerequisite for the multi-scale modeling and analysis of composite structures, and in particular to track the nucleation of instabilities in material behavior. As it remains expensive to experimentally obtain high resolution micrographs throughout a computational domain where coarse-scale behavior is being assessed, procedures for gleaning equivalent

information from lower resolution images have demonstrated their value in computational materials science Ghosh et al. (2006), Valiveti and Ghosh (2007). This paper contributes to that body of knowledge by developing a statistical procedure that updates low resolution priors with spatially localized high resolution data using a parametric texture model and a particle filter. Three approaches are commonly used, in light of their simplicity, for generating high resolution images from lower resolution ones. These are based on nearest neighbor, bilinear and cubic interpolation methods Lehmann et al. (1999). In the nearest neighbor interpolation method, a piecewise constant interpolation maps the low resolution to the high resolution image, typically producing blocky images. Bilinear and cubic interpolation methods smoothen these discontinuities, while still allowing for a blur in the high resolution image. Methods based on wavelet based interpolation with gradient based enhancement (WIGE) and higher-order polynomial interpolation with gradient based enhancement (PIGE) techniques have been proposed and successfully adapted to problems of computational multi-scale material modeling Ghosh et al. (2006), Valiveti and Ghosh (2007). In WIGE, high resolution images are obtained by interpolating low resolution images using wavelet basis with the interpolated images enhanced using gradient based methods calibrated from a few high resolution images. In PIGE, higher order polynomial bases are used for interpolation. In these methods, gradient based enhancement accounts for the position of calibration images relative to the spatial location of the simulated image. Both WIGE and PIGE methods are shown to be very effective in synthesizing high resolution images from low resolution ones with the aid of few calibrating high resolution images. In Singh et al. (2006), digital micrographs are simulated by representing the microstructures with two-point correlation function. Characterization and simulation of three-dimensional microstructures is also discussed in Groeber et al. (2008a,b), Rollett et al. (2007), Tewari et al. (2004). In these approaches, the

synthesized micrographs are deterministic and do not account for uncertainty in experimentally obtained micrographs. Other common and effective approaches for microstructure modeling include Markov random field (MRF) models whereby a Gibbs distribution is used to represent a microstructure and Gibbs sampling method is used for micrograph synthesis Geman and Geman (1984), Geman and Graffigne (1986). Parameters of the Markov random field model are typically estimated from a small number of calibrating images and the micrographs are synthesized using Markov chain Monte Carlo (MCMC) methods. A large number of parameters is required to produce high resolution features with MRF models and estimating them with credible confidence from few calibrating images remains a challenge. Numerous variations on and enhancements to standard MRF have adapted it to specific situations which have included deterministic relaxation for image classification Berthod et al. (1996), Fan and Xia (2003), texture segmentation based on wavelets and hidden Markov tree models Choi and Baraniuk (2001). A Markov random field model for image segmentation that combines both color and texture features has also been proposed Kato and Pong (2000) as well as image imputing based on hierarchical Bayesian models to reconstruct image from partially observed multivariate data Dass and Nair (2003). Quantification of microstructural variance in the formalism of stochastic processes is developed in Niezgoda et al. (2011). Algorithms based on sequential testing and feature learning have also been developed for face detection applications Fleuret and Geman (2001) while edge preserving image restoration with Huber-Markov random field models has been used to restore degraded images Pan and Reeves (2006), and procedures based on Gibbs reaction-diffusion have yielded a theory for learning generic prior models from a set of natural images Zhu and Mumford (1997). Maximum a posteriori (MAP) estimation methods have also been used for image restoration Schultz and Stevenson (1994) sometimes in conjunction with adaptive parameter estimation Zhang et al.

(2005). An approach based on filters, random fields and maximum entropy (FRAME) Zhu et al. (1998) has also been very effective in capturing high resolution features in the image, but remains computationally expensive. Its equivalence with Julesz ensembles has also been investigated Wu et al. (2000). Finally, a parametric texture models based on an over complete complex wavelet transform was also used Portilla and Simoncelli (2000) to model micrographs based on the steerable pyramid decomposition of texture (Simoncelli and Freeman (1995), Simoncelli et al. (2000)). In this paper, a similar texture model Portilla and Simoncelli (2000) is used with parameters obtained from a few high resolution calibrating images and low resolution image at the synthesis location. A particle filter is used to estimate the high resolution micrograph at a location given the low resolution micrograph at that location.

In this work, a Bayesian framework is used, where the experimental low resolution micrograph play the role of measurement used to update a prior model synthesized from a very small set of high resolution micrographs. A particle filter, namely a density-based Monte Carlo filter is used to implement the Bayesian formulation. In Tipireddy et al. (2009), Kalman filter is used for system identification based on multiple static and dynamic test data. Kalman filter methodology can not be used in our problem, because the state equation in the filtering problem is modeled as an implicit micrograph synthesis algorithm, where as the Kalman filter requires the state equation to be a linear equation. Monte Carlo based particle filters Doucet et al. (2000), Ghosh et al. (2008), Gordon et al. (1993), Manohar and Roy (2006), Tanizaki (1996) are used for system identification of dynamical system. In this work a density-based Monte Carlo filter proposed in Tanizaki (1996) is used. Synthesis of high resolution micrographs from low resolution images with parametric texture model proposed in Portilla and Simoncelli (2000) along with the density-based Monte Carlo filter Tanizaki (1996) is demonstrated in our paper. The main contribution of the present work

is to compute the parameters of the texture model from high resolution calibrating images and low resolution images and combine particle filter with a texture synthesis model to characterize the high resolution micrograph. The paper is organized as follows. Motivation of the work is described with the problem statement in section “Problem statement”. In section “Parametric texture model”, the parametric model along with the synthesis algorithm is described. Section “Density-based Monte Carlo filter” illustrates the density-based Monte Carlo filter. Numerical experiments are provided in section “Numerical illustration”.

## 6.2 Problem statement

Figure 6.1 shows a low resolution microscopic image of cast aluminum alloy W319. At locations A and B marked on the low resolution image, high resolution images are obtained from scanning electron microscopy.

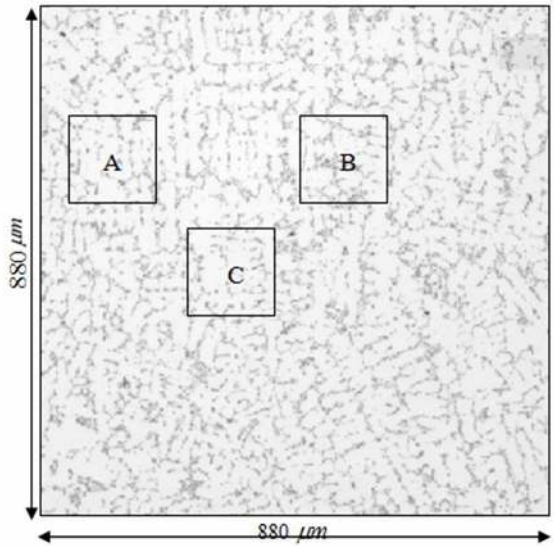
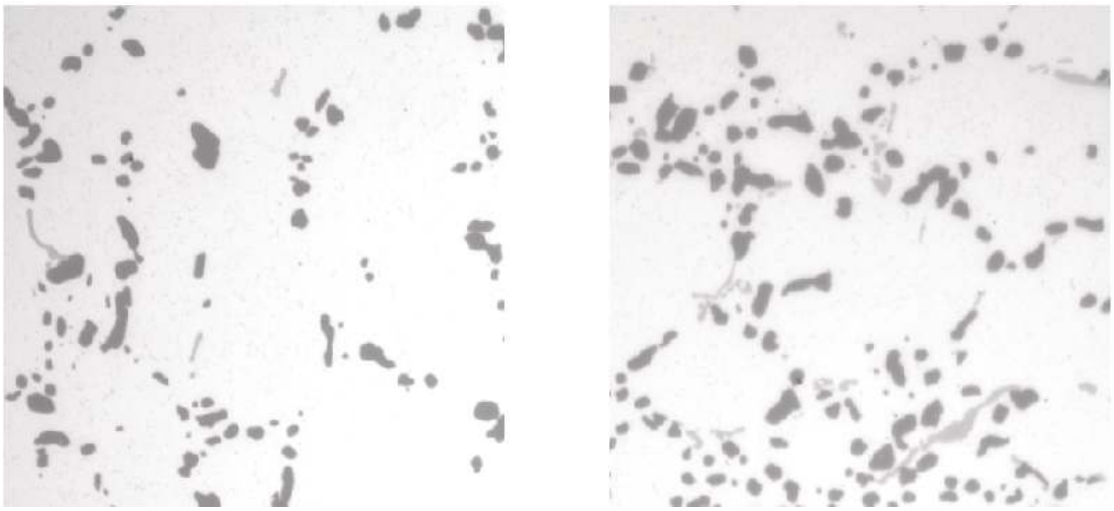



Figure 6.1: **Low resolution micrograph** Low resolution, low magnification digital image of cast aluminum alloy W319, for which high resolution images are available at points **A** and **B**

Figure 6.2 shows experimentally taken high resolution micrographs using scanning electron microscope at locations A and B. Figure 6.3 shows magnified low resolution micrographs available at locations A, B and C. Our objective is to simulate high resolution micrographs at all the locations where the high resolution microscopic images are not available. The proposed high resolution micrograph synthesis methodology is demonstrated by generating a high resolution micrograph at location C. The procedure can be repeated to simulate the high resolution micrographs at other locations.



(a) A (b) B

Figure 6.2: **High resolution digital image of cast aluminum alloy W319 at A and B**

Starting with an image $x_0$ that represents the low-resolution micrograph at location C, our aim is to synthesize an image $x$ at the same location, representing the high-resolution micrograph, that is constrained, in some sense, by our knowledge of high resolution data at locations A and B and low resolution data at C. In order to complete the statement of the problem, we must specify the nature of the constraints (i.e. the specific functionals or features that describe our knowledge of the high resolution data). In what follows, a texture model will be used to provide a context in

(a) A (b) B (c) C

Figure 6.3: **Low resolution digital image of cast aluminum alloy W319 at A, B and C**

which features of a micrograph are quantified. This model will be constructed from available high resolution data and presumed to be valid everywhere else. Specifically, the texture model will be identified with the joint statistics of the subbands in a particular wavelet decomposition, namely, a steerable pyramid decomposition. Associated with this model is a mapping $f$ that maps $x_0$ into $x$ such that $x$ has the joint statistics corresponding to the model.

The texture model maps an initial micrograph $x_0$ into a high resolution micrograph $x$. Let $x_0^i$ be a set of distinct initial micrographs, then the texture model maps each micrograph $x_0^i$ into a high resolution micrograph $x^i$ such that the parameters (joint statistics of the subbands) of each micrograph, $x^i$ are same as those of the texture model. However the visible microstructural features of each $x^i$ will be different from each other and also from those observed in the low resolution micrograph at that location. A Bayesian framework is employed using a particle filter, such that a high resolution micrograph $x$ is selected from all possible micrographs $x^i$ that is also consistent with the local microstructure. Based on maximum-entropy arguments, prior knowledge for the Bayesian framework takes the form of a Gaussian distribution with an interpolated low resolution micrograph as a mean and an assumed variance. The Likelihood is obtained from the

texture model $f$, that maps low resolution image $x_0$ to high resolution image $x$. To complete the specification of the Bayesian problem, observations consisting of a low resolution micrograph available at location C are used to update the prior.

A parametric texture model based on complex wavelet transform is used to represent the micrograph. High resolution micrograph is selected randomly from all possible micrographs satisfying a set of constraint functions, which will be discussed in section "Julesz conjecture and constraint functions". The synthesis is performed by recursively projecting the initial guess of the micrograph onto the constraint surfaces using gradient projection methods Portilla and Simoncelli (2000). Density-based Monte Carlo filter Tanizaki (1996) along with the gradient projection algorithms are used to synthesize the high resolution micrograph at location C that is consistent with experimentally obtained low resolution micrograph at the same location. The problem is formulated as an estimation of a high resolution micrograph at location C using high resolution calibrating images available at locations A and B and low resolution image available at location C. The parametrization of the texture model is discussed in section "Parameter estimation sketch". Texture parameters at location C are estimated from the parameters of images at locations A and B and those of coarse scale image at location C. The high resolution image at location C is simulated by recursively adjusting the pixel values of the iterates to satisfy a set of constraint functions and is updated using density-based Monte Carlo filter with coarse scale image at C as an observation.

## 6.3 Parametric texture model

A texture is modeled as a real two dimensional random field, $X(i,j)$ defined on a finite lattice, $(i,j) \in L \subset \mathbb{Z}^2$. Here $X(i,j)$ takes gray level values in $[0,255] \subset \mathbb{N}$. In this section we describe the steerable pyramid decomposition, which is a particular decomposition based on the complex wavelet transform of an image. A steerable pyramid acts as a filter and decomposes the image into multi-level, multi-oriented subbands. Joint statistics of the subbands, described in section "Parameter estimation sketch" are used as the parameters of the texture model. Section "Micrograph synthesis procedure" describes the texture synthesis procedure.

### Julesz conjecture and constraint functions

The Julesz conjecture (Julesz (1962), Zhu et al. (1998), Portilla and Simoncelli (2000)) states that there exists a set of functions $\{\phi_k : \mathbb{R}^{|L|} \mapsto \mathbb{R}, k = 1, \cdots, N_c\}$ , called constraint functions Portilla and Simoncelli (2000) such that samples drawn from any two random fields are visually indistinguishable under some conditions, if expectations over this set of functions are equal. That is,

$$E[\phi_k(X)] = E[\phi_k(Y)] \Rightarrow X \text{and } Y \text{are perceptually indistinguishable,} \tag{6.1}$$

where, $\{\phi_k(\cdot)\}$ is the set of constraint functions acting on a random filed and $E[\cdot]$ is mathematical expectation. To compute the expectations, practical ergodicity defined in Portilla and Simoncelli (2000) is assumed. Specifically, a homogeneous random field $X$ has the property of practical ergodicity with respect to function $\phi : \mathbb{R}^{|L|} \to \mathbb{R}$, with tolerance $\varepsilon$, and probability $p$, if and

only if the spatial average of $\phi$ over a realization drawn from $X$ is a good approximation to the expectation of $\phi$ with sufficiently high probability. More concretely,

$$P_x\left(|\overline{\phi(x(n,m))} - E[\phi(X)]| < \varepsilon\right) \geq p\ . \tag{6.2}$$

In the present case, where realizations refer to sample images $x(i,j)$, the spatial average of the functional $\phi$ can be approximated as,

$$\overline{\phi(x(n,m))} = \frac{1}{L}\sum_{(i,j)\in L} \phi\left(x(\lfloor n+i\rfloor_N, \lfloor m+j\rfloor_M)\right), \tag{6.3}$$

where, $\lfloor\cdot\rfloor_N$ means that the result is taken modulo $N$. With these definitions, our objective then is to draw samples from random field $X$ satisfying statistical constraints of the form,

$$E[\phi_k(X)] = c_k, \forall k\ . \tag{6.4}$$

The values $c_k$ in these constraints are computed from calibrating image $x(i,j)$ and constitute the parameters of the texture model. The set of all samples satisfying equation 6.4 is referred to as the Julesz ensemble Zhu et al. (1998) given by

$$\mathfrak{T}_{\phi,C} = \{x : E[\phi_k(X)] = c_k, \forall k\}\ . \tag{6.5}$$

### 6.3.1 Steerable pyramid decomposition

A parametric texture model based on over-complete complex wavelet transform Portilla and Simoncelli (2000) is used in this work. The complex wavelet transform decomposes the calibrating

image into multi-scale, multi-oriented subbands. The joint statistics of the decomposed subbands are used as the parameters of the texture model. The decomposition is known as steerable pyramid when the basis functions are directional derivatives. The filters used in this decomposition are polar-separable in the Fourier domain. The traditional orthogonal separable wavelet decomposition suffers from aliasing whereas the steerable pyramid decomposition overcomes this limitation since the support of lowpass filter obeys the Nyquist sampling criterion Portilla and Simoncelli (2000). A disadvantage with the steerable pyramid decomposition is that the basis is over-complete. A system is said to be over-complete if the the number of basis functions used to represent a vector is larger than required. Translation and rotation invariance of the steerable pyramid is important to represent the oblique orientations properly. Filters used in the steerable pyramid belong to one of three categories, namely low-pass, high-pass or oriented band-pass filters which are characterized by their respective response functions,

$$L(r) = \begin{cases} 2\cos\left(\frac{\pi}{2}\log_2\left(\frac{4r}{\pi}\right)\right), & \frac{\pi}{4} < r < \frac{\pi}{2} \\ 2, & r \leq \frac{\pi}{4} \\ 0, & r \geq \frac{\pi}{2} \end{cases}, \tag{6.6}$$

$$H(r) = \begin{cases} \cos\left(\frac{\pi}{2}\log_2\left(\frac{2r}{\pi}\right)\right), & \frac{\pi}{4} < r < \frac{\pi}{2} \\ 1, & r \leq \frac{\pi}{4} \\ 0, & r \geq \frac{\pi}{2} \end{cases}, \tag{6.7}$$

and

$$B_k(r,\theta) = H(r)G_k(\theta), \quad k \in [0, K-1], \tag{6.8}$$

where the angular part of $B_k$ is given by,

$$G_k(\theta) = \begin{cases} \alpha_k \left[\cos\left(\theta - \frac{\pi k}{K}\right)\right]^{K-1}, & \left|\theta - \frac{\pi k}{K}\right| < \frac{\pi}{2} \\ 0, & \text{otherwise} \end{cases} . \tag{6.9}$$

Here, $r$ and $\theta$ are polar coordinates and $\alpha_k = 2^{k-1}\frac{(K-1)!}{\sqrt{K[2(K-1)]!}}$. $L(r)$ and $H(r)$ are low-pass and high-pass filters respectively and $\{B_k(r,\theta)\}$ are oriented band-pass filters. The steerable pyramid decomposition is initiated by splitting an image into high-pass and low-pass subbands. The low-pass band is further decomposed into a set of orientation bands and the low-pass residual band. The low-pass residual band is sub-sampled by two and the decomposition in the next level is carried out by further decomposing the sub-sampled low-pass residual band into orientation bands and low-pass residual band. The procedure is continued till the image is decomposed into required number of pyramid levels. A system diagram for steerable pyramid is shown in fig 6.4.

In our work four orientation bands and four pyramids with a total of 18 subbands are used (16 orientation bands, one high-pass and one low pass residual band). Figure 6.5 shows the real parts of four level and four orientation steerable pyramid subbands and low-pass residual band. High-pass subband is not shown in figure 6.5.

### 6.3.2 Parameter estimation sketch

Outline for estimating parameters of the texture model from a sample image is described in the following steps:

(i). Decompose the high resolution sample image $x$ into high-pass subband, $h_0$, orientation bands, $\{b_{s,k}, s = 1, \cdots, 4, k = 1, \cdots, 4\}$ and low-pass residual bands, $\{l_0, l_1, l_2, l_3, l_4\}$.

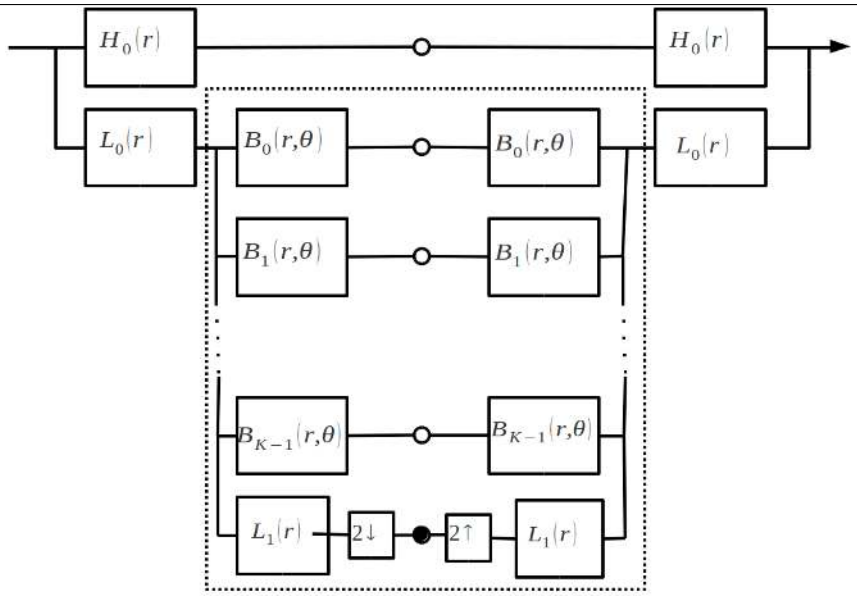



Figure 6.4: Steerable pyramid diagram Block diagram for the steerable pyramid Portilla and Simoncelli (2000), Simoncelli and Freeman (1995). Input image is initially split into high-pass and low-pass bands. Low-pass band is further decomposed into a set of oriented subbands and low-pass residual band. The recursive decomposition is achieved by inserting the diagram in dotted lines into the solid circle.

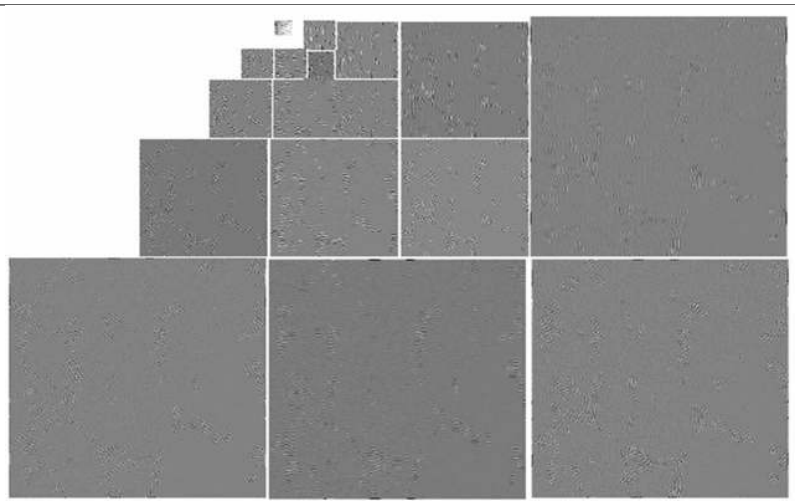


Figure 6.5: Real parts of subbands Real parts of steerable pyramid orientation bands(4 orientations and 4 scales) and low-pass subband of image at location A. High-Pass subband is not shown here.

(ii). Compute a set of joint statistics on the decomposed subbands. Following are the statistics Portilla and Simoncelli (2000) that are considered as parameters of the texture model:

- Marginal statistics are the statistics of gray-level values of the pixels in the image.
  - Mean, $\mu_1(x) = E\{x(i,j)\}$.
  - Variance, $\mu_2(x) = E\{(x(i,j) - \mu_1(x))^2\}$
  - Skewness, $\eta(x) = \frac{\mu_3(x)}{(\mu_2(x))^{1.5}}$
  - Kurtosis, $\kappa(x) = \frac{\mu_4(x)}{(\mu_2(x))^2}$
  - Range $= [\min(x)), \max(x)]$

  where, $\mu_n(x) = E\{(x(i,j) - \mu_1(x))^n\}, \forall n > 1$. In the parametric texture model, mean $\mu_1(x)$, variance $\mu_2(x)$, skewness $\eta(x)$, kurtosis $\kappa(x)$, minimum and maximum of the sample image, variance of the high-pass band $\mu_2(h_0)$, skewness $\{\eta(l_0), \eta(l_1), \eta(l_2), \eta(l_3), \eta(l_4)\}$ and kurtosis, $\{\kappa(l_0), \kappa(l_1), \kappa(l_2), \kappa(l_3), \kappa(l_4)\}$ of partially reconstructed low-pass images at each scale, $\{l_0, l_1, l_2, l_3, l_4\}$ are considered as a set of parameters from marginal statistics.

- Raw coefficient correlations, are defined as the central samples of auto-correlation of the partially reconstructed low-pass images, $A_k(n,m) = E\{l_k(i,j)l_k(\lfloor i+n \rfloor_N, \lfloor j+m \rfloor_M)\}$. In the texture model, auto correlations of the partially reconstructed low-pass images, $\{l_0, l_1, l_2, l_3, l_4\}$ at each scale, are used as a set of parameters. These statistics characterize the salient spatial frequencies and the regularity of the texture Portilla and Simoncelli (2000).

- Coefficient magnitude statistics are defined as the central samples of auto-correlation of magnitude of each subband, cross correlation of magnitude of each subband with

those of the other orientations at the same scale, cross correlation of subband magnitudes with those of all orientations at a coarser scale. Cross-correlation of any two subbands $b_s(i,j)$ and $b_t(i,j)$ is given as $C_{s,t}(n,m) = E\{b_s(i,j)b_t(\lfloor i+n \rfloor_N, \lfloor j+m \rfloor_M)\}$. These represent the structures such as edges, bars and corners in the image.

- Cross-scale phase statistics are cross-correlation of the real part of the coefficients with both the real and imaginary part of the phase doubled coefficients at all orientations at the next coarser scale. The rate at which the phase changes for fine scale coefficients is twice the rate change for coarse scale coefficients Portilla and Simoncelli (2000). To compensate for the difference in rate of phase change, the cross correlations are computed between the phase doubled coarse scale coefficients and fine scale coefficients. These statistics distinguish edges from lines.

All the joint statistics described in the above list are used as the parameters of the texture model. Practical ergodicity described in section "Julesz conjecture and constraint functions" is used to compute expectations $E[\cdot]$ in the above expressions .

(iii). Once a set of parameters is computed from high resolution micrograph, a micrograph with that set of parameters can be constructed using the synthesis procedure described in the next section.

### 6.3.3 Micrograph synthesis procedure

The synthesis procedure proposed in Portilla and Simoncelli (2000) to generate an image with desired parameters is described in the following steps. The parameters of the texture model are computed from calibrating images as joint statistics of the decomposed subbands.

(i). The synthesis procedure is initiated by choosing as initial guess an image with mean and variance equal to those of the target image.

(ii). The initial image is decomposed into multi-scale, multi-oriented subbands and joint statistics of the desired image are imposed recursively using gradient projection method Portilla and Simoncelli (2000).

(iii). Starting from coarse scale to fine scale, statistical constrains are imposed on low-pass and orientation bands while simultaneously reconstructing the low-pass image.

(iv). Auto-correlation, skewness and kurtosis of the reconstructed low-pass image at the fine scale are adjusted and resulting low-pass image is added to variance adjusted high-pass image to obtain the fine scale image.

(v). The marginal statistics are adjusted on the pixels of the reconstructed image in step-4.

(vi). The procedure is repeated from step-2 to step-5 until the image with desired statistics is generated.

(vii). This algorithm synthesizes an image that is statistically equivalent to the sample images from which the parameters are estimated.

Fig 6.6 shows one iteration step of synthesis algorithm. This algorithm can be used to simulate many statistically equivalent micrographs for a given set of parameters. But the simulated micrographs will not be visually similar to the calibrating micrograph from which the parameters are computed. When a low resolution microscopic image from an experiment is available, data assimilation methods can be used to synthesize a high resolution micrograph that is consistent with the low resolution image. We use the Matlab tools provided at Simoncelli (2009) for steerable

pyramid decomposition and micrograph synthesis. In the next section a particle filter known as the density-based Monte Carlo filter Tanizaki (1996) along with micrograph synthesis procedure is used to simulate high resolution micrographs that are consistent with the corresponding low resolution one.

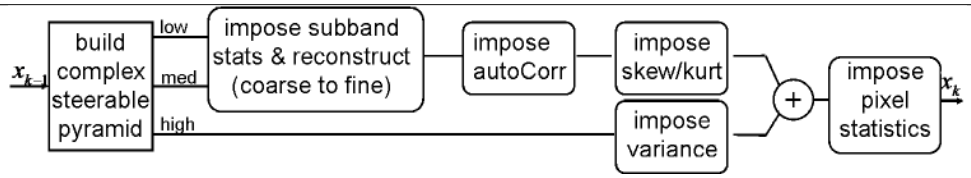



Figure 6.6: **Block diagram of the recursive micrograph synthesis procedure**

## 6.4 Density-based Monte Carlo filter

The density-based Monte Carlo filter is a nonlinear, non-Gaussian filter proposed in Tanizaki (1996) for state estimation of a dynamical system. In the current problem, an image is synthesized by an extension of the procedure described in section "Micrograph synthesis procedure". In the synthesis procedure, an initial guess of the image is assumed and the desired joint statistics are imposed on the initial guess to obtain a new image. Let $f$ be an implicit algorithm that implements steps 2-5 in the synthesis procedure shown in section "Micrograph synthesis procedure". The parameters of the texture model are repeatedly imposed on the initial image $x_0$ such that the final image has the desired parameters. That is the implicit function $f$ acts on $x_0$ to generate $x_1$, on $x_1$ to generate $x_2$ and in general acts on $x_{k-1}$ to generate $x_k$. Action of $f$ on $x_{k-1}$ to generate $x_k$, is shown in figure 6.6. We treat this as one function evaluation in the synthesis algorithm. Let $x_0, x_1, \cdots, x_k$ denote the images, modeled as random fields, generated at each function evaluation of the synthesis algorithm. These are construed as successive states of a

dynamical system where an implicit algorithm takes a dynamical state (in the present case, the state is a random field) $x_{k-1}$ to $x_k$ where $x_k$ depends only on $x_{k-1}$ and the parameters of the texture model. Here, function evaluation steps, $k = 0, \cdots, K$ are treated analogous to time steps in the dynamical system. Coarse scale images available at synthesis locations are treated as measurements and are used in the measurement equation of the particle filter algorithm to update the estimated image at each synthesis step. The state and measurement equations of the dynamical system can thus be written as

$$x_k = f(x_{k-1}) + \eta_k, \qquad \text{state equation} \tag{6.10}$$

$$y_k = g(x_k) + \varepsilon_k, \qquad \text{measurement equation} \tag{6.11}$$

where, $x_k$ is the synthesized image at $k^{th}$ step in the recursive synthesis process, $f$ is an implicit algorithm relating $x_k$ to $x_{k-1}$ as shown in figure 6.6 and $g$ is a coarsening operator relating state $x_k$ and observation $y_k$. Here, coarsening operator is an averaging operator that averages $8 \times 8$ pixels in fine scale image to produce one pixel in the coarse scale image. Errors in the synthesis process and experimental measurements are modeled as additive white Gaussian noise processes $\eta_k$ and $\varepsilon_k$ respectively. The covariance of the noises are assumed to be known and are indeed parameters of the filter. Let $X_k = \{x_0, \cdots, x_k\}$ be a vector of states and $Y_k = \{y_0, \cdots, y_k\}$ be a vector of measurements. In our work, measurements are assumed to consist of a coarse scale image available at location C and does not change with each synthesis step. Hence, we assume, $Y_k = \{y_0, y_1 = y_0, \cdots, y_k = y_0\}$. The joint probability density of all the states and the measurements is given by

$$P(X_k, Y_k) = P(Y_k|X_k)P(X_k), \tag{6.12}$$

where, $P(Y_k|X_k)$ and $P(X_k)$ can be written as,

$$P(Y_k|X_k) = \prod_{s=1}^{k} P(y_s|x_s) \tag{6.13}$$

and

$$P(X_k) = P(x_0) \prod_{s=1}^{k} P(x_s|x_{s-1}). \tag{6.14}$$

The conditional densities, $P(y_s|x_s)$ and $P(x_s|x_{s-1})$, are obtained from the measurement and state equations respectively. From equations 6.13 and 6.14, the filter density $P(x_k|Y_k)$ can be written as,

$$P(x_k|Y_k) = \frac{\int P(Y_k|X_k)P(X_k)\mathrm{d}X_{k-1}}{\int P(Y_k|X_k)P(X_k)\mathrm{d}X_k} \ . \tag{6.15}$$

The estimated micrograph, $\hat{x}_k$ at step $k$, can be obtained by computing expectation of $x_k$ with respect to the filtering density, $P(x_k|Y_k)$, i.e.,

$$\hat{x}_k = \frac{\int x_k P(Y_k|X_k)P(X_k)\mathrm{d}X_k}{\int P(Y_k|X_k)P(X_k)\mathrm{d}X_k} \ . \tag{6.16}$$

Evaluating the integrals in equation 6.16 using Monte Carlo sampling results in the filter estimate $\hat{x}_k$ at step $k$. Generating $n$ random samples of $x_0$ and $\eta_s$ for $s = \{1, \cdots, k\}$, a set of random draws $X_{i,s}$ can be obtained from the state equation as

$$x_{i,s} = f(x_{i,s-1}, \eta_{i,s}), \quad i = 1, \cdots, n \ \& \ s = 1, \cdots, k \ . \tag{6.17}$$

The filtering estimate $\hat{x}_k$ in equation (6.16) is approximated by drawing samples of $x_k$ using Monte Carlo sampling according to,

$$\hat{x}_k = \frac{\sum_{i=1}^{n} x_{i,k} P(Y_k|X_{i,k})}{\sum_{i=1}^{n} P(Y_k|X_{i,k})} \ . \tag{6.18}$$

Using equation 6.12,

$$\hat{x}_k = \frac{\sum_{i=1}^{n} x_{i,k} \prod_{s=1}^{k} P(y_s|x_{i,s})}{\sum_{i=1}^{n} \prod_{s=1}^{k} P(y_s|x_{i,s})} \ , \tag{6.19}$$

where, $(y_s|x_{i,s}) \sim N\left((g_s(x_{i,s})), \mathrm{cov}(\varepsilon_s)\right)$ denotes the likelihood and $\hat{x}_k$ is the filter estimate. Here, $X_{i,k}$ is a collection of random draws, defined as $X_{i,k} = \{x_{i,0}, \cdots, x_{i,k}\}$. Equation (6.18) can be written as

$$\hat{x}_k = \sum_{i=1}^{n} x_{i,k} w_{i,k}, \tag{6.20}$$

where the weight function $w_{i,k}$ is defined as

$$w_{i,k} = \frac{P(y_k|(x_{i,k}) w_{i,k-1}}{\sum_{j=1}^{n} P(y_k|(x_{j,k}) w_{j,k}}, \tag{6.21}$$

such that $\sum_{i=1}^{n} w_{i,k} = 1$. For the initial step, when $k = 0$, equal weights are assigned for all the samples $\{x_i, 0\}_{i=1}^{n}$, i.e., $w_{i,0} = \frac{1}{n}$ . Writing $e_k = (x_k - \hat{x}_k)$, the variance of $e_k$ can be written as

$$\Sigma_k^* = \frac{\int e_k e_k^T (P(Y_k|X_k))^2 P(X_k) \mathrm{d}X_k}{(\int (P(Y_k|X_k)) P(X_k) \mathrm{d}X_k)^2} \tag{6.22}$$

and the sample variance can be written as

$$\Sigma_k = \frac{\sum_{i=1}^{n} e_k e_k^T (P(Y_k|X_k))^2 P(X_k)}{(\sum_{i=1}^{n} (P(Y_k|X_k)) P(X_k))^2} \tag{6.23}$$

$$\Sigma_k = \sum_{i=1}^{n} (x_{i,k} - \hat{x}_k)(x_{i,k} - \hat{x}_k)^T w_{i,k}^2. \tag{6.24}$$

The following limiting conditions as $n$ goes to infinity can be verified Tanizaki (1996)

- $\hat{x}_k \longrightarrow E(x_k|Y_k)$ almost surely,
- $\sqrt{n}(x_k - E(x_k|Y_k)) \longrightarrow N(0, \Sigma_k^*)$ in distribution ,
- $n\Sigma_k \longrightarrow \Sigma_k^*$ almost surely.

Following the above constructions, the computational procedure of the density-based Monte Carlo filter is as follows:

(i). The random draws of the initial state $x_0$, represented as $x_{i,0}$ are taken from the initial density $P(x_0)$,

(ii). Given $x_{i,k-1}$ and $\eta_{i,k-1}$, estimates of $x_{i,k}$ are obtained from the state equation 6.10,

(iii). Given the initial weight $w_{i,0}$, weight functions $w_{i,k}$ for $i = 1, \cdots, n$ and $k = 1, \cdots, K$ are obtained from equation 6.21,

(iv). Finally, the filtered state $\hat{x}_k$ is evaluated from equation 6.20 while the variance of the error in the estimate is computed from equation 6.24.

Fig 6.7 shows the block diagram of the micrograph synthesis using parametric texture model and density-based Monte Carlo filter.

## 6.5 Numerical illustration

The high resolution micrograph synthesis procedure is demonstrated by synthesizing a high resolution micrograph at location C starting with low resolution micrograph at that location and

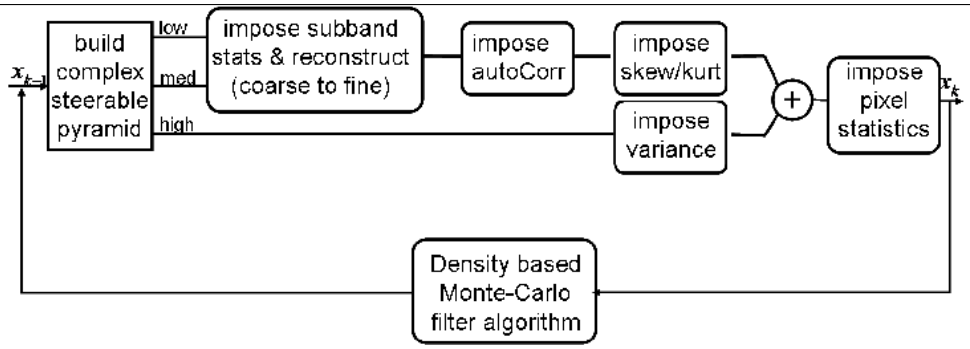
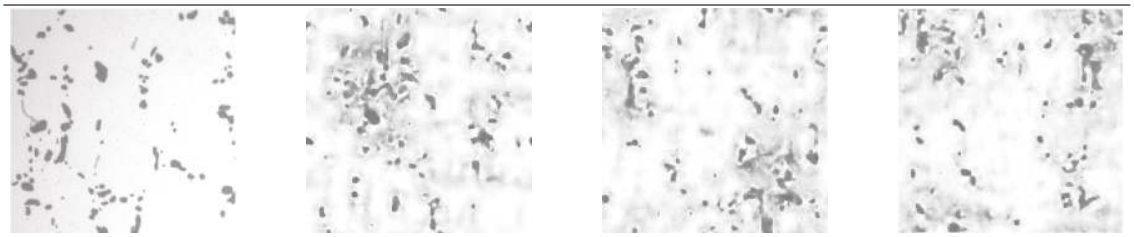


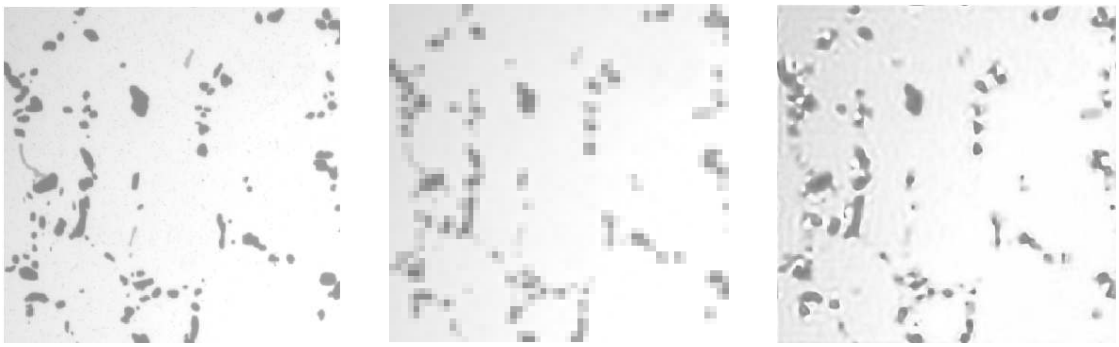

Figure 6.7: Block diagram of synthesis procedure with particle filter. Block diagram of the recursive micrograph synthesis procedure along with the density-based Monte Carlo filter.

high resolution calibrating micrographs available at locations A and B as shown in figures 6.1 and 6.2. The procedure can be repeated to construct high resolution micrographs at any other location. Two cases are considered to construct high resolution micrographs. In the first case (figures 6.8, 6.10 and 6.12), a parametric texture model with the synthesis procedure described in section "Micrograph synthesis procedure" is used to construct a high resolution micrograph, whereas in the second case (figures 6.9, 6.11 and 6.13), the parametric texture model along with the density-based Monte Carlo filter described in section "Density-based Monte Carlo filter" is used to estimate a high resolution micrograph.



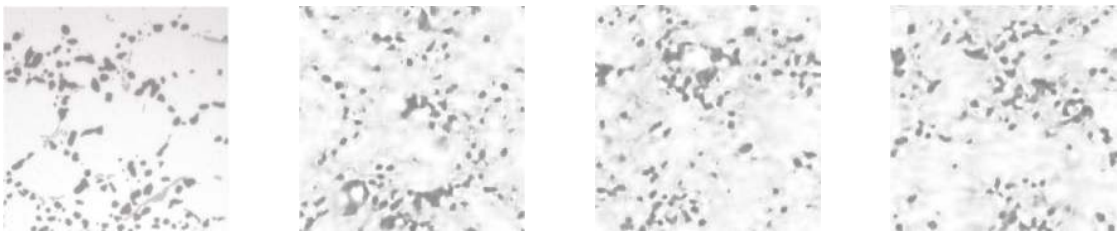
(a) High resolution A (b) micrograph-1 (c) micrograph-2 (d) micrograph-3

Figure 6.8: Micrograph at A synthesized without particle filter. Micrographs from left to right show respectively, the experimental high resolution micrograph, synthesized micrographs without particle filter which are statistically equivalent with high resolution micrograph at location **A**.



(a) High resolution A (b) Low resolution A (c) Estimated A

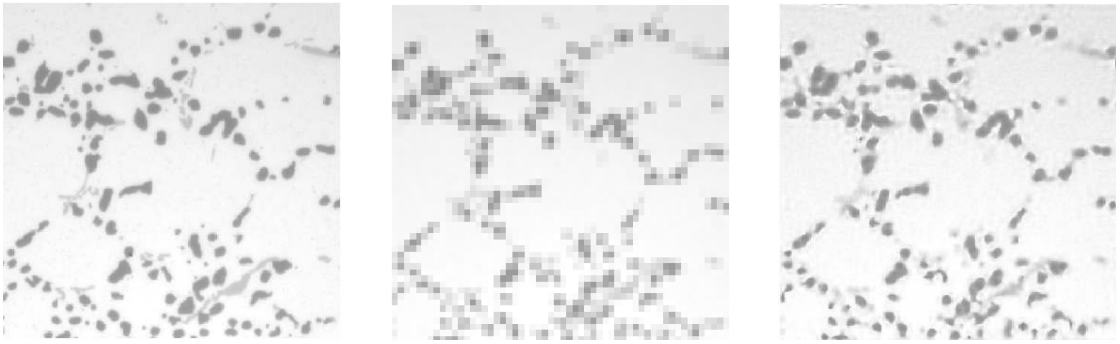
Figure 6.9: Micrograph at A synthesized with particle filter. From 6.9**(a)** to 6.9**(c)**, experimental high resolution micrograph, experimental low resolution micrograph and high resolution micrograph synthesized with texture model and particle filter at location A.



(a) High resolution B (b) micrograph-1 (c) micrograph-2 (d) micrograph-3

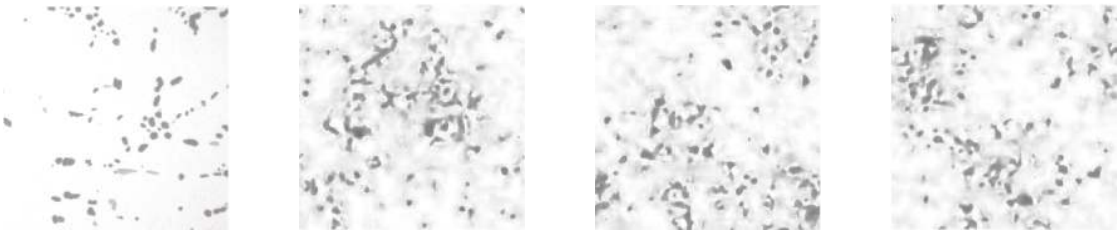
Figure 6.10: Micrograph at B synthesized without particle filter. Micrographs from left to right show respectively, the experimental high resolution micrograph, synthesized micrographs without particle filter which are statistically equivalent with high resolution micrograph at location **B**.

When the particle filter is not used, the synthesis procedure does not make use of the low resolution micrographs available at a given location. It uses only the parameters of the texture model obtained from the calibrating micrographs. The synthesis process with distinct initial guesses results in statistically equivalent high resolution micrographs, that have visually distinct microstructural features. To validate the synthesis procedure, high resolution micrographs are synthesized at locations A and B where experimental high resolution micrographs are available. Figure 6.8(a)



(a) High resolution B (b) Low resolution B (c) Estimated B

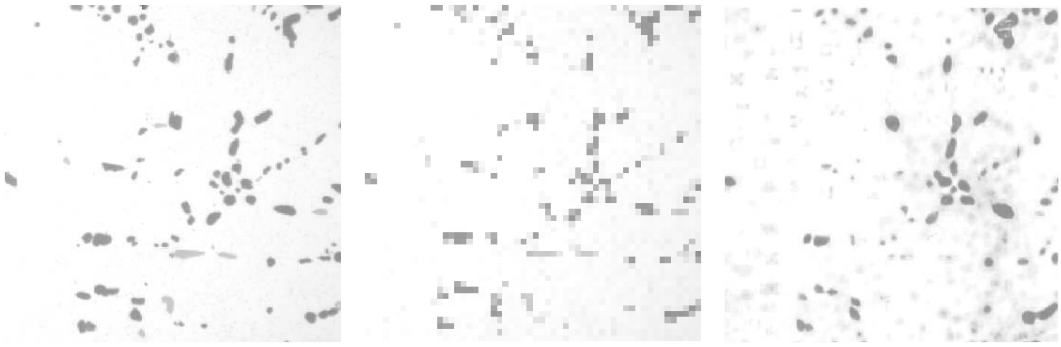
Figure 6.11: Micrograph at B synthesized with particle filter. From6.11**(a)** to 6.11**(c)**, experimental high resolution micrograph, experimental low resolution micrograph and high resolution micrograph synthesized with texture model and particle filter at location B.



(a) High resolution C (b) micrograph-1 (c) micrograph-2 (d) micrograph-3

Figure 6.12: Micrograph at C synthesized without particle filter. Micrographs from left to right show respectively, the experimental high resolution micrograph, synthesized micrographs without particle filter which are statistically equivalent with high resolution micrograph at location **C**. High resolution micrograph at location **C** is used only for comparison and is not used in the synthesis process.

shows a high resolution micrograph obtained experimentally at location A and figures 6.8(b) – 6.8(d) show synthesized micrographs whose joint statistics are the same as those of 6.8(a) but with visibly distinct microstructural features. Similarly figure 6.10(a) shows an experimental high resolution micrograph at location B and figures 6.10(b) – 6.10(d) show corresponding synthesized micrographs. Parameters of the texture model for the synthesis of these micrographs at locations A and B are obtained using high resolution micrographs available at these locations.



(a) High resolution C (b) Low resolution C (c) Estimated C

Figure 6.13: Micrograph at C synthesized with particle filter. From 6.13**(a)** to 6.13**(c)**, experimental high resolution micrograph, experimental low resolution micrograph and high resolution micrograph synthesized with texture model and particle filter at location **C**. High resolution micrograph at location **C** is used only for comparison and is not used in the synthesis process.

---

In case of location C, on the other hand, we assume that an experimental high resolution micrograph is not available and we aim to synthesize one. The parameters of the texture model for C are obtained from a low resolution micrograph at C and high resolution micrographs available at locations A and B. Pixel statistics and low-pass band statistics which account for total gray scale levels are computed from low resolution micrograph at C and the rest of the parameters which determine high resolution feature are computed as averages of the parameters from A and B. Figure 6.12(a) shows experimental high resolution micrograph and figures 6.12(b) – 6.12(d) show corresponding synthesized micrographs at location C. Here high resolution micrograph at location C (figure 6.12(a)) is used only for comparison and is not used in the synthesis process.

In the second case, high resolution micrographs of the microstructure are estimated in the Bayesian context using the density-based Monte Carlo filter (section "Density-based Monte Carlo filter") where the texture synthesis process (section "Micrograph synthesis procedure") is treated as a dynamical system and the experimental low resolution micrograph at a given location is

treated as measurement used to update the dynamical state. It is reiterated that the high resolution micrograph at C is synthesized using the low resolution micrograph at C and high resolution micrographs at A and B together with a texture model and a particle filter. The parameters of the texture model for A, B and C are obtained similarly to before. For estimating high resolution micrographs at locations A, B and C, experimental low resolution micrographs at corresponding locations are used as measurements in the filtering algorithm. Low resolution micrographs are also used as the statistical average of the initial probability density function in the algorithm. The algorithm convergence is slowed for other choices of initial average. Measurement and process noises are assumed to be zero mean Gaussian random fields with $0.01$ coefficient of variation. In the particle filter algorithm, $25$ Monte Carlo samples are used to estimate the high resolution micrographs. Figure 6.9(a) and figure 6.9(b) show high and low resolution micrographs obtained experimentally at location A, figure 6.9(c) is the estimated high resolution micrograph at location A using density-based Monte Carlo filter along with the texture synthesis procedure. Figures 6.11(a)– 6.11(c) show the experimental high resolution, experimental low resolution, estimated high resolution micrographs at location B. At location C, figure 6.13(a) and figure 6.13(b) are experimental high and low resolution micrographs, figure 6.13(c) is the estimated high resolution micrograph. It should be noted that the high resolution features of micrograph at location C (figure 6.13(c)), are recovered using information from experimental low resolution micrograph at C and experimental high resolution micrographs at A and B.

It is again worth comparing figure 6.12 showing the synthesis of high resolution micrograph using only the parametric texture model with figure 6.13 showing high resolution micrograph synthesized using texture model along with the particle filter. It can be clearly observed that the estimated high resolution micrograph with particle filter (figure 6.13(c)) provides information

about the local microstructure that cannot be gleaned from previous approaches. Resolving this level of detail is paramount in physical problems where bifurcation is initiated by spatially local events.

## 6.6 Appendix

Here, we provide the pseudo-code for high resolution micrograph synthesis using the parametric texture model and density based Monte-Carlo filter. A few high resolution micrographs are decomposed using steerable pyramid method and the parameters of the texture model are obtained as joint statistics of the decomposed subbands described in "Parameter estimation sketch".

**Algorithm 3** Recursive micrograph syntheis procedure

1. Compuete parameters $p$ from sample micrographs
2. Assume initial guess $x_0$
3. Compute parameters $\tilde{p}$ from $x_0$
4. **while** $\tilde{p} \neq p$ **do**
5. Decompose $x_k$ into subbands, $h_0$, $\{b_{s,k}, s = 1, \cdots, 4, k = 1, \cdots, 4\}$ and $\{l_0, l_1, l_2, l_3, l_4\}$
6. Compute the parameters $\tilde{p}$ from above subbands
7. From coarse to fine scale, impose desired statistics and reconstruct $l_0$
8. Impose auto-correlation on $l_0$
9. Impose skewness and kurtosis on $l_0$
10. Impose variance on $h_0$
11. Construct finescale micrograph $x_{k+1} = l_0 + h_0$
12. Impose pixel level statistics on $x_{k+1}$
13. **end while**.

**Algorithm 4** Density based Monte-Carlo filter

1. Draw initial samples $x_{i,0}$ from density $P(x_0)$

2. **for** $k$ = 1 to $K$ **do**

3. Compute $x_{i,k} = f(x_{i,k-1} + \eta_{i,k-1})$, where $f$ is micrograph synthesis algorinthm 3

4. Compute wights $w_{i,k}$ from equation 6.21

5. **end for**

6. Estimate $\hat{x}$ from equation 6.20 and variance from equation 6.24.

# Bibliography


Adams, B. M., Dalbey, K. R., Eldred, M. S., Gay, D. M., Swiler, L. P., Bohnhoff, W. J., Eddy, J. P., Haskell, K., and Hough, P. D. 2010. *DAKOTA, A Multilevel Parallel Object-Oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quantification, and Sensitivity Analysis*. Sandia National Laboratories, tech. rep.sand2010-2183 edition.

Babuška, I., Tempone, R., and Zouraris, G. E. 2004. Galerkin finite element approximations of stochastic elliptic partial differential equations. *SIAM J. Numer. Math.*, 42:800–825.

Berthod, M., Kato, Z., Yu, S., and Zerubia, J. 1996. Bayesian image classification using markov random fields. *Image and Vision Computing*, 14:285–295.

Betchen, L., Straatman, A. G., and Thompson, B. E. 2006. A nonequilibrium finite-volume model for conjugate fluid/porous/solid domains. *Numerical Heat Transfer, Part A*, 49:543–565.

Brooks, A. N. and Hughes, T. J. 1982. Streamline upwind/petrov-galerkin formulations for convection dominated flows with particular emphasis on the incompressible navier-stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 32:199–259.

C. Gear, J. H., Kevrekidis, P., Kevrekidis, I., Runborg, O., and Theodoropoulos, C. 2003. Equation-free, coarse-grained multiscale computation: Enabling mocroscopic simulators to perform system-level analysis. *Commun. Math. Sci.*, 1:715–762.

Chandesris, M., Serre, G., and Sagaut, P. 2006. A macroscopic turbulence model for flow in porous media suited for channel, pipe and rod bundle flows. *International Journal of Heat and Mass Transfer*, 49:27392750.

Choi, H. and Baraniuk, R. G. 2001. Multiscale image segmentation using wavelet-domain hidden markov models. *IEEE Transactions on Image Processing*, 10:1309–1321.

Dass, S. C. and Nair, V. N. 2003. Edge detection, spatial smoothing, and image reconstruction with partially observed multivariate data. *Journal of the American Statistical Association*, 98:77–89.

Doostan, A., Ghanem, R., and Red-Horse, J. 2007. Stochastic model reduction for chaos representations. *Computer Methods in Applied Mechanics and Engineering*, 196:3951–3966.

Doostan, A. and Iaccarino, G. 2009. A least-squares approximation of partial differential equations with high-dimensional random inputs. *Journal of Computational Physics*, 228:4332–4345.

Doostan, A. and Owhadi, H. 2011. A non-adapted sparse approximation of pdes with stochastic inputs. *Journal of Computational Physics*, 230:3015–3034.

Doucet, A., Godsill, S., and Andrieu, C. 2000. On sequential monte carlo sampling methods for bayesian filtering. *Statistics and computing*, 10:197–208.

Elman, H. C., Miller, C. W., Phipps, E. T., and Tuminaro, R. S. 2011. Assessment of collocation and galerkin approaches to linear diffusion equations with random data. *International Journal for Uncertainty Quantifications*, 1:19–34.

Fan, G. and Xia, X. G. 2003. Wavelet-based texture analysis and synthesis using hidden markov models. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 50:106–120.

Fleuret, F. and Geman, D. 2001. Coarse-to-fine face detection. *International Journal of Computer Vision*, 41:85–107.

Geman, S. and Geman, D. 1984. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Pat. Anal. Mach. Intell*, 6:721–741.

Geman, S. and Graffigne, C. 1986. Markov random field image models and their applications to computer vision. In *Proceedings of International Congress of mathematicians*, pages 1496–1517, Berkeley, California, USA.

Ghanem, R. 1999. Ingredients for a general purpose stochastic finite elements implementation. *Computer Methods in Applied Mechanics and Engineering*, 168:19–34.

Ghanem, R. and Spanos, P. 1991. *Stochastic Finite Elements: A Spectral Approach*. Springer-Verlag.

Ghanem, R. G. and Doostan, A. 2006. On the construction and analysis of stochastic models: characterization and propagation of the errors associated with limited data. *Journal of Computational Physics*, 213:63–81.

Ghosh, S., Valiveti, D., Harris, S. J., and Boileau, J. 2006. A domain partitioning based preprocessor for multi-scale modelling of cast aluminium alloys. *Modelling and simulation in materials science and engineering*, 14:1363–1396.

Ghosh, S. J., Manohar, C. S., and Roy, D. 2008. A sequential importance sampling filter with a new proposal distribution for state and parameter estimation of nonlinear dynamical systems. *Proc. R. Soc. A*, 464:25–47.

Gordon, N. J., Salmond, D. J., and Smith, A. F. M. 1993. Novel approach to nonlinear/non-gaussian bayesian state estimation. *IEEE Trans. Radar Signal Process.*, 140:107–113.

Groeber, M., Ghosh, S., Uchic, M. D., and Dimiduk, D. M. 2008a. A framework for automated analysis and simulation of 3d polycrystalline microstructures.: Part 1: Statistical characterization. *Acta Materialia*, 56:12571273.

Groeber, M., Ghosh, S., Uchic, M. D., and Dimiduk, D. M. 2008b. A framework for automated analysis and simulation of 3d polycrystalline microstructures. part 2: Synthetic structure generation. *Acta Materialia*, 56:12741287.

Holden, H., Oksendal, B., Uboe, J., and Zhang, T. 1996. *Stochastic Partial Differential Equations: a modeling, white noise functional approach*. Birkhauser Boston Inc., Cambridge, MA, USA.

Janson, S. 1999. *Gaussian Hilbert Spaces*. Cambridge University Press.

Julesz, B. 1962. Visual pattern discrimination. *IRE Trans. Info Theory*, IT-8:84–92.

Kato, Z. and Pong, T. C. 2000. A markov random field image segmentation model for color textured images. *Image and Vision Computing*, 24:1103–1114.

Lehmann, T. M., Gönner, C., and K.Spitzer 1999. Survey: Interpolation methods in medical image processing. *IEEE Transactions on Medical Imaging*, 18:1049–1075.

Manohar, C. S. and Roy, D. 2006. Monte carlo filters for identification of nonlinear structural dynamical systems. *Sadhana*, 31(4):399–427.

Nakayama, A., Kuwahara, F., Umemoto, T., and Hayashi, T. 2002. Heat and fluid flow within an anisotropic porous medium. *Transactions of ASME*, 124:746–753.

Niezgoda, S. R., Yabansu, Y. C., and Kalidindi, S. R. 2011. Understanding and visualizing microstructure and microstructure variance as a stochastic process. *Acta Materialia*, 59:63876400.

Nobile, F., Tempone, R., and Webster, C. G. 2008. A sparse grid stochastic collocation method for partial differential equations with random input data. *SIAM Journal of Numerical Analysis*, 46:2309–2345.

Pan, R. and Reeves, S. J. 2006. Efficient huber-markov edge-preserving image restoration. *IEEE Transactions on Image Processing*, 15:3728–3735.

Portilla, J. and Simoncelli, E. 2000. A parametric texture model based on joint statistics of complex wavelet coefficients. *International Journal of Computer Vision*, 40(1):49–70.

Powell, C. and Elman, H. 2009. Block-diagonal preconditioning for spectral stochastic finite element systems. *IMA Journal of Numerical Analysis*, 29:350–375.

Rollett, A. D., Lee, S. B., Campman, R., and Rohrer, G. 2007. Three-dimensional characterization of microstructure by electron back-scatter diffraction. *Annual Review of Materials Research*, 37:627–658.

Rosseel, E. and Vandewalle, S. 2010. Iterative solvers for the stochastic finite element method. *SIAM J. Sci. Comput*, 32:372–397.

Saad, Y. 1996. *Iterative Methods for Sparse Linear Systems*. SIAM.

Schmidt, R. C., Belcourt, K., Clarno, K. T., Hooper, R., Humphries, L. L., Lorber, A. L., Pryor, R. J., and Spotz, W. F. 2010. *Foundational Development of an Advanced Nuclear Reactor Integrated Safety Code*. Sandia National Labs, sand2010-0878 edition.

Schultz, R. and Stevenson, R. L. 1994. A bayesian approach to image expansion for improved definition. *IEEE Transactions on Image Processing*, 3:233–242.

Schwab, C. and Gittelson, C. 2011. Sparse tensor discretizations of high-dimensional parametric and stochastic pdes. *Acta Numerica*, 20:291–467.

Shadid, J., Salinger, A., Pawlowski, R., Lin, P., Hennigan, G., Tuminaro, R., and Lehoucq, R. 2006. Large-scale stabilized fe computational analysis of nonlinear steady-state transport/reaction systems. *Comput. Methods Appl. Mech. Engrg.*, 195:1846–1871.

Simoncelli, E. P. 2009. matlabpyrtools code.

Simoncelli, E. P. and Freeman, W. 1995. The steerable pyramid: A flexible architecture for multi-scale derivative computation. In *In Second Int'l Conf. on Image Proc.*, volume III, pages 444–447, Washington, DC,. IEEE Sig. Proc. Society.

Simoncelli, E. P., Freeman, W., Adelson, E., and Heeger, D. 2000. Shiftable multi-scale transforms. *IEEE Trans Information Theory*, 38(2):587–607.

Singh, H., Gokhale, A. M., Mao, Y., and Spowart, J. E. 2006. Computer simulations of realistic microstructures of discontinuously reinforced aluminum alloy (dra) composites. *Acta Materialia*, 54:21312143.

Smolyak, S. 1963. Quadrature and interpolation formulas for tensor products of certain classes of functions. *Dokl. Akad. Nauk SSSR*, 148:1042–1043.

Tanizaki, H. 1996. *Nonlinear filters: estimation and applications*. Springer Verlag, Berlin, second edtion edition.

Tewari, A., Gokhale, A. M., Spowart, J. E., and Miracle, D. B. 2004. Quantitative characterization of spatial clustering in three-dimensional microstructures using two-point correlation functions. *Acta Materialia*, 52:307319.

Tezduyar, T. E. 1992. Stabilizedfinite element formulations for incompressibleflow computation. *Advances in Applied Mechanics*, 28:1–44.

Tipireddy, R., Nasrellah, H. A., and Manohar, C. S. 2009. A kalman filter based strategy for linear structural system identification based on multiple static and dynamic test data. *Probabilistic Engineering Mechanics*, 24:60–74.

Ullmann, E. 2010. A kronecker product preconditioner for stochastic galerkin finite element discretizations. *SIAM J. Sci. Comput.*, 32(2):923–946.

Valiveti, D. M. and Ghosh, S. 2007. Morphology based domain partitioning of multi-phase materials: A preprocessor for multi-scale modelling. *Int. J. Numer. Meth. Engng*, 69:1717–1754.

Weiner, N. 1963. The homogeneous chaos. *American Journal of Mathematics*, 60:897–936.

Wu, Y. N., Zhu, S. C., and Liu, X. 2000. Equivalence of julesz ensembles and FRAME models. *International Journal of Computer Vision*, 38:247–265.

Xiu, D. and Karniadakis, G. 2002. The wiener-askey polynomial chaos for stochastic differential equations. *SIAM J. Sci. Comput.*, 24:619–644.

Zhang, X., Lam, K. M., and Shen, L. 2005. Image magnification based on adaptive mrf model parameter estimation. In *Proceedings of 2005 International Symposium on Intelligent Signal Proceeding and Communication Systems*, pages 653–656, Hong Kong.

Zhu, S., Wu, Y. N., and Mumford, D. 1998. Filters, random fields and maximum entropy (FRAME):towards the unified theory for texture modeling. *International Journal of Computer Vision*, 27(2):107–126.

Zhu, S. C. and Mumford, D. 1997. Prior learning and gibbs reaction-diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:1236–1250.