Acta Numerica (2014), pp. 521–650
doi:10.1017/S0962492914000075

Stochastic finite element methods for partial differential equations with random input data*

Max D. Gunzburger
Department of Scientific Computing,
Florida State University, Tallahassee, Florida 32306, USA
E-mail: mgunzburger@fsu.edu
https://www.sc.fsu.edu/~gunzburg

Clayton G. Webster
Department of Computational and Applied Mathematics,
Oak Ridge National Laboratory, Oak Ridge, Tennessee 37831, USA
E-mail: webstercg@ornl.gov
http://www.csm.ornl.gov/~cgwebster

Guannan Zhang
Department of Computational and Applied Mathematics,
Oak Ridge National Laboratory, Oak Ridge, Tennessee 37831, USA
E-mail: zhangg@ornl.gov
http://www.csm.ornl.gov/~gz3

The quantification of probabilistic uncertainties in the outputs of physical, biological, and social systems governed by partial differential equations with random inputs require, in practice, the discretization of those equations. Stochastic finite element methods refer to an extensive class of algorithms for the approximate solution of partial differential equations having random input data, for which spatial discretization is effected by a finite element method. Fully discrete approximations require further discretization with respect to solution dependences on the random variables. For this purpose several approaches have been developed, including intrusive approaches such as stochastic Galerkin methods, for which the physical and probabilistic degrees of freedom are coupled, and non-intrusive approaches such as stochastic sampling and interpolatory-type stochastic collocation methods, for which the physical and probabilistic degrees of freedom are uncoupled. All these method classes are surveyed in this article, including some novel recent developments. Details about the construction of the various algorithms and about theoretical error estimates and complexity analyses of the algorithms are provided. Throughout, numerical examples are used to illustrate the theoretical results and to provide further insights into the methodologies.

* Colour online for monochrome figures available at journals.cambridge.org/anu.

CONTENTS

PART ONE
Introduction

Mathematical modelling and computer simulations are tools widely used to predict the behaviour of scientific and engineering systems and to assess risk and inform decision making in manufacturing, service, economic, public policy, military, and many other venues. Such predictions are obtained by constructing mathematical models whose solutions describe the phenomenon of interest and then using computational methods to approximate the outputs of the models. Thus, the solution of a mathematical model can be viewed as a mapping from available input information onto a desired output of interest; predictions obtained through computational simulations are merely approximations of the images of the inputs, that is, of the output of interest.

There are several causes for possible discrepancies between observations and approximate solutions obtained via computer simulations. The mathematical model may not, and usually does not, provide a totally faithful description of the phenomenon being modelled. In many, if not most applications, a hierarchy of models of increasing fidelity is available. In the partial differential equation (PDE) setting of this article, errors also arise because the mathematical models must be discretized to enable a computer simulation. Again, hierarchies of discretization methods having increasing fidelity are also available. It is usually the case that higher-fidelity models require larger computational resources to obtain the same discretization error. Because of limits in time, hardware capability, and costs, a user usually has to strike a balance between the fidelity of the model and the accuracy of the discretization method used.

Discretization errors can be controlled and reduced by using sophisticated techniques such as *a posteriori* error estimation coupled with mesh adaptivity (Ainsworth and Oden 2000, Babuška and Strouboulis 2001, Eriksson, Estep, Hansbo and Johnson 1995, Moon, von Schwerin, Szepessy and Tempone 2006, Verfürth 1996, Johnson 2000). Modelling errors are more difficult to quantify and control and are thus usually identified by comparison with observations, although analytical approaches have been developed for some settings (Oden and Vemaganti 2000, Oden, Prudhomme, Hammerand and Kuczma 2001, Oden and Prudhomme 2002, Braack and Ern 2003, Romkes and Oden 2004, Oden *et al.* 2005*a*, Oden, Prudhomme and Bauman 2005*b*). Of course, one should always make sure that discretization errors are sufficiently small that they do not dominate and therefore obscure possible modelling errors. These and other efforts at identifying and controlling modelling and discretization errors have increased the accuracy of computational predictions as well as our confidence in them.

However, other critical issues involving the mathematical modelling/computational simulation combination have not been so adequately addressed. Perhaps foremost among these are the ever-present uncertainties in model inputs, that is, even if the form of a mathematical model is accepted as being correct, the model inputs are not known with exactitude. Thus, model input uncertainty is a third source of discrepancy between simulation outputs and observations. Algorithms that can be used to help account for output uncertainties is the central subject of this article.

In the PDE setting, model input uncertainties appear in coefficients, forcing terms, boundary and initial condition data, geometry, *etc.* (Babuška and Chleboun 2002, Babuška and Chleboun 2003, Tartakovsky and Broyda 2011, Fichtl, Prinja and Warsa 2009, Babuška and Oden 2006). Input uncertainties may be due to incomplete knowledge that, in principle, could be remedied through additional measurements or improved measuring devices but for which such remedies are too costly or impractical to apply. For example, the highly heterogeneous subsurface properties in groundwater flow simulations can only be measured at relatively few locations, so at other locations these properties are subject to uncertainty. Incomplete knowledge can also be forced into a model due to lack of computational resources. For example, although turbulent flows are generally thought of as being adequately modelled by the Navier–Stokes equations, in many practical situations one cannot use that model because the grids necessary to adequately approximate solutions are so fine that the resulting computational cost is prohibitive; in such cases, the unresolved scales are sometimes modelled via the addition of uncertainties into the model. Uncertainties due to incomplete knowledge are referred to as being *epistemic*. Additional examples of such uncertainties include the mechanical properties of many biomaterials, polymeric fluids, or composite materials and initial data for weather forecasting.

In other situations, uncertainty is due to an inherent variability in the system that cannot be reduced by additional experimentation or improvements in measuring devices. Such uncertainties are referred to as being *aleatoric*. Examples include unexpected fluctuations induced in a flow field around an aircraft wing by wind gusts or on a structure by seismic vibrations. When sufficient data are available, probability distributions can be used to fully characterize such uncertainties in a statistical manner so that the uncertainty can be modelled as a random process.

Discussions about both types of sources of uncertainties are given in a general setting by Cullen and Frey (1999), and for some applications to solid mechanics, climate modelling, and turbulent flow by Ben-Haim (1996), Mrczyk (1997), Elishakoff (1999), Melchers (1999), Elishakoff and Ren (2003), Oden, Belytschko, Babuška and Hughes (2003), Reilly *et al.* (2001), Lucor, Meyers and Sagaut (2007), Cheung *et al.* (2011) and Pope (1981, 1982).

In practice we have to deal with both types of uncertainty, that is, we are faced with the task of *uncertainty quantification* (UQ), which is a broadly used term that encompasses a variety of methodologies including uncertainty characterization and propagation, parameter estimation/model calibration, and error estimation. Simply put, the goal of UQ is to learn about the uncertainties in system outputs of interest, given information about the uncertainties in the system inputs. Given that this task is crucial to assessing risks, robust design, and many other areas of scientific and engineering enquiry, it is not surprising that the development of UQ methodologies within those communities, as well as among computational mathematicians, has been and remains a very active area of research. There are, in fact, several approaches being followed for quantifying uncertainties, including the following.

- *Worst-case-scenario* (or anti-optimization) methods (Hlaváček, Chleboun and Babuška 2004, Babuška, Nobile and Tempone 2005*a*), which are useful in cases where we know only a little information about the uncertainty in the input data, namely that the input data lie in a functional set that might well be infinite-dimensional.
- *Probabilistic* methods, which use statistical characterizations of uncertainties, such as probability density functions or expected values, variances, correlation functions, and statistical moments (Ghanem and Spanos 2003, Kleiber and Hien 1992, Benth and Gjerde 1998*a*, Benth and Gjerde 1998*b*, Ghanem and Red-Horse 1999, Glimm *et al.* 2003, Xiu and Karniadakis 2002*a*, Schwab and Todor 2003*b*, Schwab and Todor 2003*a*, Xiu and Karniadakis 2003, Soize 2003, Lucor, Xiu, Su and Karniadakis 2003, Lucor and Karniadakis 2004, Le Maître, Knio, Najm and Ghanem 2004*a*, Le Maître, Najm, Ghanem and Knio 2004*b*, Soize and Ghanem 2004, Babuška, Tempone and Zouraris 2004, Narayanan and Zabaras 2004, Zabaras and Samanta 2004, Lu and Zhang 2004, Xiu and Tartakovsky 2004, Regan, Ferson and Berleant 2004, Babuška, Tempone and Zouraris 2005*b*, Keese and Matthies 2005, Matthies and Keese 2005, Frauenfelder, Schwab and Todor 2005, Soize 2005, Rubinstein and Choudhari 2005, Narayanan and Zabaras 2005*b*, Narayanan and Zabaras 2005*a*, Mathelin, Hussaini and Zang 2005, Roman and Sarkis 2006, Webster 2007, Lin, Tartakovsky and Tartakovsky 2010, Xiu 2009, Nobile and Tempone 2009, Doostan and Iaccarino 2009, Ma and Zabaras 2009, Beck, Nobile, Tamellini and Tempone 2011, Elman, Miller, Phipps and Tuminaro 2011, Gunzburger, Trenchea and Webster 2013, Eldred, Webster and Constantine 2008, Burkardt, Gunzburger and Webster 2007, Nobile, Tempone and Webster 2008*a*, Nobile, Tempone and Webster 2008*b*, Nobile, Tempone and Webster 2007, Agarwal and Aluru 2009, Barth, Schwab and Zollinger 2011, Gunzburger and

Labovsky 2011, Doostan, Ghanem and Red-Horse 2007, Dauge and Stevenson 2010, Stoyanov and Webster 2014, Gunzburger, Jantsch, Teckentrup and Webster 2014, Zhang and Gunzburger 2012, Zhang, Webster and Gunzburger 2014).

- *Bayesian inference and optimization* (Webster, Zhang and Gunzburger 2013, Zhang *et al.* 2013, Box 1973, Lemm 2003, Beck and Au 2002, Yuen and Beck 2003, Ching and Beck 2004, Wang and Zabaras 2005, Marzouk and Xiu 2009, Cheung *et al.* 2011, Marzouk, Najm and Rahn 2007, Babuška, Nobile and Tempone 2008, Cheung and Beck 2010, Muto and Beck 2008), estimating calibration parameters from noisy experimental data (Kennedy and O'Hagan 2001, Higdon *et al.* 2004, Qian *et al.* 2006, Bayarri *et al.* 2007, Qian and Wu 2008, Joseph and Melkote 2009, Chang and Joseph 2013, Joseph 2013, Tuo and Wu 2013).
- *Measure-theoretic approaches*, which approximate densities through closed systems of PDEs (Breidt, Butler and Estep 2011, Tartakovsky and Broyda 2011, Pope 1981, Pope 1982).
- *Knowledge-based* methods, which characterize uncertainties using fuzzy sets (Bernardini 1999, Dubois and Prade 2000), evidence theory (Oberkampf, Helton and Sentz 2001, Kramosil 2001, Ferson *et al.* 2003), and subjective probability (Vick 2002, Helton 1997).

All five approaches can be applied directly or indirectly to PDEs with uncertain input data. Despite the large effort represented by these citations, it is widely recognized that we have not reached the end of research in UQ. New, more effective methods for treating uncertainty are still needed and will become increasingly important in virtually all branches of engineering and science (Babuška, Nobile and Tempone 2007*b*, Phipps, Eldred, Salinger and Webster 2008, Dongarra *et al.* 2013).

A crucial, yet often complicated, ingredient that all approaches to UQ must incorporate is a proper description of the uncertainty in system parameters and external environments. All such uncertainties can be included in mathematical models adopting the probabilistic approach, provided enough information is available for an accurate statistical characterization of the data. The mathematical model may depend on a set of distinct uncertain parameters that may be represented as random variables with a given joint probability distribution. In other situations, the input data may vary randomly from one point of the physical domain to another and from one time instant to another. In these cases, uncertainty in the inputs should rather be described in terms of random fields. Approaches to describing *correlated* random fields include Karhunen–Loève expansions (Loève 1977, 1978) (or Fourier–Karhunen–Loève expansion: Li *et al.* 2007) and expansions in terms of global orthogonal polynomials (Wiener 1938, Ghanem and Spanos 2003, Xiu and Karniadakis 2002*b*). Both types of expansion

represent a random field in terms of an infinite number of random variables and require that the random field has a bounded second statistical moment. Other nonlinear expansions (Grigoriu 2002) and transformations (Matthies and Keese 2005, Winter and Tartakovsky 2002) have been considered. Whereas all these expansions are infinite, realizations often vary slowly in space and time, and thus only a few terms are typically needed to accurately approximate the random field (Babuška, Liu and Tempone 2003, Frauenfelder *et al.* 2005). Therefore, in this article, *we consider probabilistic representations of uncertainties in mathematical models consisting of a system of stochastic partial differential equations*[1] (SPDEs) *having coefficients and source terms that are described by a finite-dimensional random vector*, either because the problem itself can be described by a finite number of random variables or because inputs are modelled as truncated expansions of random fields.

The outline of this article is as follows. In the rest of Part 1 we provide a brief discussion of uncertainty quantification in the SPDE setting and an overview of numerical methods for SPDEs.

In Part 2 we provide a generalized mathematical description of SPDEs, establish the notation used throughout, and introduce stochastic finite element methods and stochastic Galerkin methods. We also introduce the notions of semi-discrete and fully discrete stochastic approximation and state assumptions about the parametrization of random inputs which prove useful for transforming a given SPDE into a deterministic parametric one.

In Part 3 we consider sampling-based SFEMs by introducing a general framework that incorporates all stochastic sampling methods, and also explain how stochastic sampling methods fit into the framework of stochastic Galerkin methods. This part also includes an error analysis of both the discretization and sampling errors associated with the use of classical Monte Carlo sampling methods. Also shown is how the overall computational complexity can be reduced through the use of multilevel Monte Carlo methods. The part ends with an overview of other stochastic sampling methods.

In Part 4 we consider problems for which the solution of the SPDE has very smooth dependence on the input random variables. We present SFEMs that approximate solutions using global approximations in parameter space. We first introduce several choices of multivariate polynomial spaces that result in global stochastic Galerkin methods and global stochastic collocation methods. A generalized sparse grid interpolatory approximation is presented, followed by a detailed convergence analysis with respect to the

[1] In some circles, the nomenclature 'stochastic partial differential equations' is reserved for a specific class of PDEs having random inputs, driven by uncorrelated stochastic processes. Here, for the sake of economy of notation, we use this terminology to refer to any PDE having random inputs.

total number of collocation points. We conclude the part with a numerical example that provides a setting for the comparison of the total computational complexity of global stochastic Galerkin and stochastic collocation methods.

In Part 5 we consider problems for which the solution of the SPDE may have irregular dependence on the input random variables, as a result of which the global approximations discussed in Part 4 are usually not appropriate. As an alternative, we present SFEMs that use locally supported piecewise polynomial spaces for both spatial and stochastic discretization. We then extend this concept to include adaptive hierarchical stochastic collocation methods and provide a novel acceleration technique to reduce the computational complexity of obtaining fully discrete approximations. We also provide a detailed error estimate and complexity analysis for our new approach.

In the Appendices we provide a brief review of the essential concepts, definitions, and results from probability theory and stochastic processes.

Three comments about the content of this article are called for. First, throughout, we ignore the temporal dependence of solutions of SPDEs, that is, we assume that coefficients, forcing functions, *etc.*, and therefore solutions, only depend on spatial variables and random parameters. We do this merely for economy of notation. Almost all discussions extend to problems that also involve temporal dependences. Second, throughout, we only consider finite element methods for effecting the spatial discretization of SPDEs. Most of the discussions also apply to finite difference, finite volume, and spectral methods for spatial discretization. Third, throughout, we treat problems having random inputs that consist of a finite number of parameters or are correlated random fields. So as not to completely ignore the important class of problems having *white noise* inputs, we provide, in the last Appendix, a brief discussion about how a white noise random field, which is an infinite stochastic process, can be discretized.

1.1. Uncertainty quantification

In the SPDE setting, *uncertainty quantification is the task of determining, given statistical information about the inputs of an SPDE, statistical information about an output of interest that depends on the solution of the SPDE.* Outputs of interest could be the solution of the PDE itself, but more often take the form of functionals of that solution. If $u(\mathbf{x}, \boldsymbol{y})$ denotes the solution of the SPDE, where $\boldsymbol{y}$ denotes a vector of random parameters, examples of outputs of interest include the spatial average of u over the spatial domain D,

$$G_u(\boldsymbol{y}) = \frac{1}{|D|} \int_D u(\mathbf{x}, \boldsymbol{y}) \, \mathrm{d}\mathbf{x},$$

where $|D|$ denotes the volume of D, or the maximum value of u over D,

$$G_u(\boldsymbol{y}) = \max_{\mathbf{x}\in D} u(\mathbf{x}, \boldsymbol{y}).$$

The desired statistical information comes in the form of expected values, variances, or higher statistical moments of the output of interest. In the first case, we would then have the quantity of interest

$$\mathbb{E}[G_u(\boldsymbol{y})] = \int_\Gamma G_u(\boldsymbol{y})\rho(\boldsymbol{y})\,\mathrm{d}\boldsymbol{y}, \tag{1.1.1}$$

where Γ denotes the parameter domain and $\rho(\boldsymbol{y})$ the joint probability density function for the random vector $\boldsymbol{y}$ of input parameters. Another quantity of interest is event probabilities. For example, one might want to determine the probability that a scalar output of interest $G_u(\boldsymbol{y})$ is greater than some threshold value τ. This probability can be expressed as

$$\int_\Gamma 1_{G_u(\boldsymbol{y})>\tau}\rho(\boldsymbol{y})\,\mathrm{d}\boldsymbol{y},$$

where we have the indicator function

$$1_{G_u(\boldsymbol{y})>\tau} = \begin{cases} 1 & \text{if } G_u(\boldsymbol{y}) > \tau, \\ 0 & \text{otherwise.} \end{cases}$$

In practice, one cannot determine the exact solution of the SPDE, so approximate solutions are used instead when evaluating an output of interest. A further approximation step occurs because integrals over the parameter domain Γ usually have to be approximated using a quadrature rule.

Given that for us the outputs of interest depend on the solution of an SPDE, estimating the accuracy of approximations of those solution is required to ascertain information about the accuracy of approximations of a quantity of interest. Thus, in this article, *we focus on the approximation of solutions of SPDEs.* It is then a straightforward matter, at least conceptually, to obtain information about the accuracy of approximations of quantities of interest.

1.2. An overview of numerical methods for SPDEs

Monte Carlo methods (see, *e.g.*, Fishman 1996) are the most popular approach for approximating expected values and other statistical moments of quantities of interest of the solution to an SPDE. Monte Carlo methods are based on independent realizations of the input parameters; approximations of the expectation or other quantities of interest are obtained by averaging over the corresponding realizations of that quantity. Thus, the method requires a deterministic PDE solution for each realization. The resulting numerical error is proportional to $1/\sqrt{M}$, where M denotes the number of realizations, thus requiring a very large number of SPDE

solutions to achieve small errors. In particular cases, convergence can be improved with the use of important sampling (Jouini, Cvitanić and Musiela 2001, Novak 1988, Traub and Werschulz 1998), multilevel methods (Barth and Lang 2012, Barth, Lang and Schwab 2013, Barth *et al.* 2011, Cliffe, Giles, Scheichl and Teckentrup 2011, Giles 2008), and other means.

Other ensemble-based methods such as quasi-Monte Carlo sequences, Latin hypercube sampling, lattice rules, and orthogonal arrays (see, *e.g.*, Niederreiter 1992, Helton and Davis 2003 and the references therein) have been devised to produce 'faster' convergence rates, for example, proportional to $(\log(M)^{r(N)}/M)$, where $r(N) > 0$ grows with the number N of random variables. Another sampling approach is provided by centroidal Voronoi tessellations (Du, Faber and Gunzburger 1999, Romero, Gunzburger, Burkardt and Peterson 2006, Saka, Gunzburger and Burkardt 2007, Du, Gunzburger and Ju 2002, Du and Gunzburger 2002*a*, Du and Gunzburger 2002*b*, Du, Gunzburger and Ju 2003*a*, Du, Gunzburger and Ju 2003*b*, Du and Gunzburger 2003, Romero *et al.* 2003*a*, Romero, Gunzburger, Burkardt and Peterson 2003*b*, Romero, Burkardt, Gunzburger and Peterson 2005, Du, Gunzburger, Ju and Wang 2006, Burkardt, Gunzburger and Lee 2006*a*, Burkardt, Gunzburger and Lee 2006*b*, Ju, Gunzburger and Zhao 2006, Ringler, Ju and Gunzburger 2008, Nguyen *et al.* 2009, Du, Gunzburger and Ju 2010, Jacobsen *et al.* 2013, Womeldorff, Peterson, Gunzburger and Ringler 2013)

In the past decade, other approaches have been proposed that, in some situations, often feature much faster convergence rates. These include spectral (global) *stochastic Galerkin methods* (Ghanem and Spanos 2003, Ghanem and Red-Horse 1999, Xiu and Karniadakis 2002*a*, Babuška *et al.* 2004, Babuška *et al.* 2005*b*, Deb 2000, Deb, Babuška and Oden 2001, Frauenfelder *et al.* 2005, Matthies and Keese 2005, Le Maître *et al.* 2004*a*, Roman and Sarkis 2006), *stochastic collocation methods* (Babuška, Nobile and Tempone 2007*a*, Tatang 1995, Mathelin *et al.* 2005, Xiu and Hesthaven 2005, Nobile *et al.* 2008*a*, Nobile *et al.* 2008*b*), and *perturbation, Neumann, and Taylor expansion methods* (Gaudagnini and Neumann 1999, Winter and Tartakovsky 2002, Babuška and Chatzipantelidis 2002, Karniadakis *et al.* 2005, Todor 2005, Winter, Tartakovsky and Guadagnini 2002, Lu and Zhang 2004, Kleiber and Hien 1992). These approaches transform the original stochastic problem into a deterministic one with a large number of parameters and differ in the choice of polynomial bases and the corresponding approximating spaces used to effect approximation in the probability domain. Additional details can be found in recent work by Le Maître and Knio (2010), Xiu (2009) and Nobile and Tempone (2009). These methods use standard approximations in physical space, such as a finite element method, and globally defined polynomial approximation in the probability domain, either by full polynomial spaces (Xiu and Karniadakis 2002*a*, Matthies and

Keese 2005, Ghanem 1999), tensor product polynomial spaces (Babuška *et al.* 2004, Frauenfelder *et al.* 2005, Roman and Sarkis 2006), or sparse tensor product polynomials (Cohen, DeVore and Schwab 2011, Xiu and Hesthaven 2005, Webster 2007, Nobile *et al.* 2008*a*, Nobile *et al.* 2008*b*, Beck, Nobile, Tamellini and Tempone 2014, Beck, Tempone and Nobile 2012).

In Ghanem and Red-Horse (1999) and Ghanem and Spanos (2003), formal Wiener chaos expansions in terms of Hermite polynomials are used. A similar approach using general orthogonal polynomials, sometimes referred to as *polynomial chaos*, is described in Xiu and Karniadakis (2002*a*). Generally, these techniques are *intrusive* in the sense that they are non-ensemble-based methods, that is, they require the solution of discrete systems that couple all spatial and probabilistic degrees of freedom. Variations, including non-intrusive polynomial chaos methods (Acharjee and Zabaras 2007, Hosder and Walters 2007, Eldred *et al.* 2008), have been developed that decouple the stochastic and spatial degrees of freedom by exploiting the orthogonality of the basis and using appropriate quadrature rules.

Recently, global stochastic collocation methods based on either full or sparse tensor product approximation spaces (Babuška *et al.* 2007*a*, Ganapathysubramanian and Zabaras 2007, Nobile *et al.* 2008*a*, Nobile *et al.* 2008*b*, Mathelin *et al.* 2005, Xiu and Hesthaven 2005) have gained considerable attention. As shown in Babuška *et al.* (2007*a*), stochastic collocation methods can essentially match the fast convergence of intrusive polynomial chaos methods, even coinciding with them in particular cases. The major difference between the two approaches is that stochastic collocation methods are ensemble-based, *non-intrusive* approaches that achieve fast convergence rates by exploiting the inherent regularity of PDE solutions with respect to parameters. Compared to non-intrusive polynomial chaos methods, they also require fewer assumptions about the underlying SPDE. Stochastic collocation methods can also be viewed as stochastic Galerkin methods in which one employs an interpolatory basis built from the zeros of orthogonal polynomials with respect to the joint probability density function of the input random variables. For additional details about the relations between polynomial chaos methods and stochastic collocation methods see Le Maître and Knio (2010) and Xiu (2010), for example, and for computational comparisons between the two approaches see Elman *et al.* (2011), Beck *et al.* (2011) and Jantsch, Webster and Zhang (2014).

To achieve increased rates of convergence, most polynomial chaos and stochastic collocation approaches described above are based on global polynomial approximations that take advantage of smooth behaviour of the solution in the multi-dimensional parameter space. Hence, when there are steep gradients, sharp transitions, bifurcations, or finite discontinuities (*e.g.*, piecewise processes) in stochastic space, these methods converge very slowly or even fail to converge. Such problems often arise in scientific and

engineering problems due to the highly complex nature of most physical or biological phenomena. To be effective, refinement strategies must be guided by accurate estimations of errors (both local and global) while not expending significant computational effort approximating an output of interest within each random dimension. The resulting explosion in computational effort as the number of random parameters increases is referred to as the *curse of dimensionality*; not surprisingly, there have been many methods proposed to counteract this curse.

The first type involves domain decomposition approaches, such as the 'multi-element' method presented in Foo and Karniadakis (2010), Wan and Karniadakis (2009) and Foo, Wan and Karniadakis (2008), which decomposes each parameter dimension into subdomains and then uses tensor products to reconstruct the entire parameter space. This method has been successfully applied to moderate dimension problems, but the tensor product decomposition inevitably falls prey to the curse of dimensionality. Similarly, a tensor-product-based multi-resolution approximation, by virtue of a Galerkin projection onto a Wiener–Haar basis, is developed in Le Maître and Knio (2010) and Le Maître *et al.* (2004*a*). This approach provides significant improvements over global polynomial chaos expansions. However, in terms of robustness, dimension scaling is not possible due to the resulting dense coupled system and the lack of any rigorous criteria for triggering refinement.

Elman and Miller (2011), Ma and Zabaras (2009, 2010) and Jakeman, Archibald and Xiu (2011) apply an adaptive sparse grid stochastic collocation strategy that follows the work of Griebel (1998), Gerstner and Griebel (2003) and Klimke and Wohlmuth (2005); piecewise multi-linear hierarchical h-type finite elements basis functions are used, similar to those constructed in the physical domain. These approaches utilize the hierarchical surplus as an error indicator to automatically detect regions of importance (*e.g.*, discontinuities) in the stochastic parameter space and adaptively refine the collocation points in this region. To this end, grids are constructed in an adaptation process steered by the indicator in such a way that a prescribed global error tolerance is attained. This goal, however, might be achieved using more points than necessary due to the instability of this multi-scale basis. To address this issue, Gunzburger, Webster and Zhang (2014) have introduced a novel multi-dimensional multi-resolution adaptive wavelet stochastic collocation method, with the desirable multi-scale stability of the hierarchical coefficients guaranteed as a result of the wavelet basis having the Riesz property. This property provides an additional lower-bound estimate for the wavelet coefficients that are used to guide the adaptive grid refinement, resulting in the approximation requiring a significantly reduced number of deterministic simulations for both smooth and irregular stochastic solutions.

PART TWO
Stochastic finite element methods

2.1. Partial differential equations with random input data

We begin by focusing our attention on a possibly nonlinear elliptic operator $\mathcal{L}$, defined on a domain $D \subset \mathbb{R}^d$, $d = 1, 2$ or 3, having boundary ∂D. The operator $\mathcal{L}$ has a coefficient $a(\mathbf{x}, \omega)$ with $\mathbf{x} \in D$ and $\omega \in \Omega$, where $(\Omega, \mathcal{F}, \mathbb{P})$ denotes a complete probability space. Here, Ω is the set of outcomes, $\mathcal{F} \subset 2^\Omega$ is the σ-algebra of events, and $\mathbb{P} : \mathcal{F} \to [0, 1]$ is a probability measure. Analogously, the forcing term $f = f(\mathbf{x}, \omega)$ can be assumed random as well.

Consider the following stochastic elliptic boundary value problem. Find a random function $u : \overline{D} \times \Omega \to \mathbb{R}$ such that $\mathbb{P}$-almost everywhere in Ω, that is, almost surely, we have that

$$\mathcal{L}(a)(u) = f \quad \text{in } D \tag{2.1.1}$$

equipped with suitable boundary conditions.

We let $W(D)$ denote a Banach space of functions $v : D \to \mathbb{R}$ and define the stochastic Banach spaces

$$L^q_{\mathbb{P}}\big(\Omega; W(D)\big) := \Big\{ v : \Omega \to W(D) \mid v \text{ is strongly measurable and } \int_\Omega \|v(\cdot, \omega)\|^q_{W(D)} \, \mathrm{d}\mathbb{P}(\omega) < +\infty \Big\}$$

for $q \in [1, \infty)$ and

$$L^\infty_{\mathbb{P}}\big(\Omega; W(D)\big) := \Big\{ v : \Omega \to W(D) \mid v \text{ is strongly measurable and } \mathbb{P} - \operatorname*{ess\,sup}_{\omega \in \Omega} \|v(\cdot, \omega)\|^2_{W(D)} < +\infty \Big\}.$$

Of particular interest is the space $L^2_{\mathbb{P}}(\Omega; W(D))$ consisting of Banach-valued functions that have finite second stochastic moments.

We make the following assumptions.

Assumption 2.1.1.

(a) The solution to (2.1.1) has realizations in the Banach space $W(D)$, that is, $u(\cdot, \omega) \in W(D)$ almost surely and, for all $\omega \in \Omega$,

$$\|u(\cdot, \omega)\|_{W(D)} \le C \|f(\cdot, \omega)\|_{W^*(D)},$$

where $W^*(D)$ denotes the dual space of $W(D)$ and C denotes a constant having value independent of the realization $\omega \in \Omega$.

(b) The forcing term $f \in L^2_{\mathbb{P}}(\Omega; W^*(D))$ is such that the solution u is uniquely defined and bounded in $L^2_{\mathbb{P}}(\Omega; W(D))$.

Two examples of problems posed in this setting are as follows.

Example 2.1.2. The *linear* second-order elliptic problem

$$\begin{aligned} -\nabla \cdot \big(a(\mathbf{x},\omega)\nabla u(\mathbf{x},\omega)\big) &= f(\mathbf{x},\omega) \quad \text{in } D\times\Omega, \\ u(\mathbf{x},\omega) &= 0 \qquad\qquad \text{on } \partial D\times\Omega, \end{aligned} \tag{2.1.2}$$

with $a(\mathbf{x},\omega)$ uniformly bounded from above and below, that is,

$$\begin{aligned} &\text{there exist } a_{\min}, a_{\max} \in (0,\infty) \text{ such that} \\ &\mathbb{P}\big(\omega\in\Omega:\ a(\mathbf{x},\omega)\in[a_{\min},a_{\max}]\ \forall\,\mathbf{x}\in\overline{D}\big)=1, \end{aligned}$$

and $f(\mathbf{x},\omega)$ square-integrable with respect to $\mathbb{P}$, that is,

$$\int_D \mathbb{E}[f^2]\,\mathrm{d}\mathbf{x} = \int_D\int_\Omega f^2(\mathbf{x},\omega)\,\mathrm{d}\mathbb{P}(\omega)\,\mathrm{d}\mathbf{x} < \infty,$$

such that Assumptions 2.1.1(a,b) are satisfied with $W(D) = H_0^1(D)$; see Babuška *et al.* (2007*a*).

Example 2.1.3. Similarly, for $s\in\mathbb{N}_+$, the *nonlinear* second-order elliptic problem

$$\begin{aligned} -\nabla \cdot \big(a(\mathbf{x},\omega)\nabla u(\mathbf{x},\omega)\big) + u(\mathbf{x},\omega)|u(\mathbf{x},\omega)|^s &= f(\mathbf{x},\omega) \quad \text{in } D\times\Omega, \\ u(\mathbf{x},\omega) &= 0 \qquad\qquad \text{on } \partial D\times\Omega, \end{aligned} \tag{2.1.3}$$

with $a(\mathbf{x},\omega)$ uniformly bounded from above and below and $f(\mathbf{x},\omega)$ square-integrable with respect to $\mathbb{P}$ such that Assumptions 2.1.1(a,b) are satisfied with $W(D) = H_0^1(D)\cap L^{s+2}(D)$; see Webster (2007).

2.2. Parametrization of random inputs

Because the two sources of stochasticity, namely the random fields $a(\mathbf{x},\omega)$ and $f(\mathbf{x},\omega)$, are seldom related to each other, it is reasonable to assume that they are defined on two independent probability spaces $(\Omega_a,\mathcal{F}_a,\mathbb{P}_a)$ and $(\Omega_f,\mathcal{F}_f,\mathbb{P}_f)$, respectively. Then the solution u is defined on the product probability space $(\Omega,\mathcal{F},\mathbb{P}) = (\Omega_a\times\Omega_f, \mathcal{F}_a\times\mathcal{F}_f, \mathbb{P}_a\times\mathbb{P}_f)$ and $\omega = (\omega_a,\omega_f)\in\Omega$, where $\omega_a\in\Omega_a$ and $\omega_f\in\Omega_f$. Thus, a and f are essentially functions of ω_a and ω_f, respectively.

In many applications, the source of randomness can be approximated using just a finite number of uncorrelated, or even independent, random variables. As such, similar to Babuška *et al.* (2007*a*), Nobile *et al.* (2008*a*) and Nobile *et al.* (2008*b*), we make the following assumptions regarding the stochastic input data, that is, the random coefficient $a(\mathbf{x},\omega_a)$ in $\mathcal{L}$ and the right-hand side $f(\mathbf{x},\omega_f)$.

Assumption 2.2.1. The random input data of the PDE in (2.1.1) satisfy the following.

(a) The functions $a(\mathbf{x}, \omega_a)$ and $f(\mathbf{x}, \omega_f)$ are bounded from above and below with probability 1, that is, for the right-hand side $f(\mathbf{x}, \omega_f)$, there exists $f_{\min} > -\infty$ and $f_{\max} < \infty$ such that

$$\mathbb{P}\big(\omega_f \in \Omega_f : f_{\min} \le f(\mathbf{x}, \omega_f) \le f_{\max} \ \forall\, \mathbf{x} \in \overline{D}\big) = 1, \tag{2.2.1}$$

and similarly for the random coefficient $a(\mathbf{x}, \omega_a)$.

(b) The input data $a(\mathbf{x}, \omega_a)$ and $f(\mathbf{x}, \omega_f)$ have the form

$$\begin{aligned} a(\mathbf{x}, \omega_a) &= a\big(\mathbf{x}, \boldsymbol{y}_a(\omega_a)\big) \quad \text{in } \overline{D} \times \Omega_a, \\ f(\mathbf{x}, \omega_f) &= f\big(\mathbf{x}, \boldsymbol{y}_f(\omega_f)\big) \quad \text{in } \overline{D} \times \Omega_f, \end{aligned} \tag{2.2.2}$$

where, with $N_a \in \mathbb{N}_+$, $\boldsymbol{y}_a(\omega_a) = \big(y_{a,1}(\omega_a), \dots, y_{a,N_a}(\omega_a)\big)$ is a vector of real-valued *uncorrelated* random variables and likewise for $\boldsymbol{y}_f(\omega_f) = \big(y_{f,1}(\omega_f), \dots, y_{f,N_f}(\omega_f)\big)$ with $N_f \in \mathbb{N}_+$.

(c) The random functions $a\big(\mathbf{x}, \boldsymbol{y}_a(\omega_a)\big)$ and $f\big(\mathbf{x}, \boldsymbol{y}_f(\omega_f)\big)$ are σ-measurable with respect to $\boldsymbol{y}_a$ and $\boldsymbol{y}_f$, respectively.

We next provide two examples of random input data that satisfy Assumption 2.2.1. Without lost of generality, we only consider the coefficient $a(\mathbf{x}, \omega_a)$ in the examples.

Example 2.2.2 (piecewise constant random fields). Assume that the spatial domain D is the union of non-overlapping subdomains D_n, $n = 1, \dots, N_a$. Then consider a coefficient $a(\mathbf{x}, \omega_a)$ that is a random constant in each subdomain D_n, that is, $a(\mathbf{x}, \omega_a)$ is the piecewise constant function

$$a(\mathbf{x}, \omega_a) = a_0 + \sum_{n=1}^{N_a} a_n y_{a,n}(\omega_a) 1_{D_n}(\mathbf{x}),$$

where a_n, $n = 0, \dots, N$, denote constants, 1_{D_n} denotes the indicator function of the set $D_n \subset D$, and the random variables $y_{a,n}(\omega_a)$, $n = 1, \dots, N$, are bounded and independent. Note that Assumption 2.2.1 requires restrictions on the constants a_n and the bounds on the random variables $y_{a,n}(\omega_a)$; in practice, such restrictions would be deduced from the physics of the problem.

Example 2.2.3 (Karhunen–Loève expansions). According to Mercer's theorem (Theorem B.1), any second-order correlated random field $a(\mathbf{x}, \omega_a)$ with continuous covariance function $\mathbb{COV}(\mathbf{x}_1, \mathbf{x}_2)$ can be represented as an infinite sum of random variables. One commonly used example is the Karhunen–Loève expansion discussed in Appendix B.1. In this case, the random field $a(\mathbf{x}, \omega_a)$ can be approximated by a truncated Karhunen–Loève

expansion having the form

$$a(\mathbf{x},\omega_a) \approx a_{N_a}(\mathbf{x},\omega_a) = \mathbb{E}[a(\mathbf{x},\cdot)] + \sum_{n=1}^{N_a} \sqrt{\lambda_n} b_n(\mathbf{x}) y_{a,n}(\omega_a),$$

where λ_n and $b_n(\mathbf{x})$ for $n = 1,\dots,N_a$ are the dominant eigenvalues and corresponding eigenfunctions for the covariance function and $y_{a,n}(\omega_a)$ for $n = 1,\dots,N_a$ denote uncorrelated real-valued random variables. Note that if the process is Gaussian, then the random variables $\{y_{a,n}\}_{n=1}^{N_a}$ are standard independent identically distributed random variables. In addition, we would like to keep the property that a random input coefficient is bounded away from zero. To do this, we instead expand the logarithm of the random filed so that $a_{N_a}(\mathbf{x},\omega)$ has the form

$$a_{N_a}(\mathbf{x},\omega_a) = a_{\min} + \mathrm{e}^{\sum_{n=1}^{N_a} \sqrt{\lambda_n} b_n(\mathbf{x}) y_{a,n}(\omega_a)}, \tag{2.2.3}$$

where $a_{\min} > 0$ is the lower bound on a.

Assumption 2.2.1 and the Doob–Dynkin lemma (Lemma A.12) guarantee that $a(\mathbf{x}, \boldsymbol{y}_a(\omega_a))$ is a Borel-measurable function of the random vector $\boldsymbol{y}_a$ and likewise for f with respect to $\boldsymbol{y}_f$. As mentioned above, the random fields a and f are independent because of their physical properties, so that $\boldsymbol{y}_a$ and $\boldsymbol{y}_f$ are independent random vectors. Thus, we relabel the elements of the two random vectors and define $\boldsymbol{y} = (y_1,\dots,y_N) = (\boldsymbol{y}_a, \boldsymbol{y}_f)$, where $N = N_a + N_f$. By definition, the random variables $\{y_n\}_{n=1}^N$ are mappings from the product sample space Ω to the real space $\mathbb{R}^N$, so we let $\Gamma_n = y_n(\Omega) \subset \mathbb{R}$ denote the image of the random variable y_n, and set $\Gamma = \prod_{n=1}^N \Gamma_n$, where $N \in \mathbb{N}_+$. If the distribution measure of $\boldsymbol{y}(\omega)$ is absolutely continuous with respect to the Lebesgue measure, there exists a joint probability density function (PDF) for $\{y_n\}_{n=1}^N$ denoted by

$$\rho(\boldsymbol{y}) : \Gamma \to \mathbb{R}_+ \quad \text{with } \rho(\boldsymbol{y}) \in L^\infty(\Gamma).$$

Thus, based on Assumption 2.2.1, the probability space $(\Omega, \mathcal{F}, \mathbb{P})$ is mapped to $(\Gamma, \mathcal{B}(\Gamma), \rho(\boldsymbol{y})\,\mathrm{d}\boldsymbol{y})$, where $\mathcal{B}(\Gamma)$ is the Borel σ-algebra on Γ and $\rho(\boldsymbol{y})\,\mathrm{d}\boldsymbol{y}$ is the finite measure.

Remark 2.2.4. What is the form of the joint density function $\rho : \Gamma \to \mathbb{R}_+$? According to the definition of $\mathbb{P} = \mathbb{P}_a \times \mathbb{P}_f$, for some element $A \in \mathcal{F}$, it is not true that $\mathbb{P}(A) = \mathbb{P}_a(A)\mathbb{P}_f(A)$. However, in the image space $(\Gamma, \mathcal{B}^d, \rho\,\mathrm{d}\boldsymbol{y})$, if we denote $\rho = \rho_a \times \rho_f$, because of Fubini's theorem, it is true that for any $\boldsymbol{y} = (\boldsymbol{y}_a, \boldsymbol{y}_f) \in \Gamma$ we have $\rho(\boldsymbol{y}) = \rho_a(\boldsymbol{y}_a)\rho_f(\boldsymbol{y}_f)$. In fact, in the product probability space $(\Omega, \mathcal{F}, \mathbb{P})$, by Fubini's theorem, a multiple integral can be converted to an iterative integral, that is, for a function

$\varphi(\omega) = \varphi\big(\boldsymbol{y}_a(\omega_a), \boldsymbol{y}_f(\omega_f)\big)$ with $\omega \in \Omega_a \times \Omega_f$, we have

$$\mathbb{E}[\varphi] = \int_{\Omega_a \times \Omega_f} \varphi \,\mathrm{d}(\mathbb{P}_a \times \mathbb{P}_f) = \int_{\Omega_a} \int_{\Omega_f} \varphi(\omega_a, \omega_f) \,\mathrm{d}\mathbb{P}_f(\omega_f) \,\mathrm{d}\mathbb{P}_a(\omega_a).$$

Then, mapping the right-hand side to the space $(\Gamma, \mathcal{B}^d, \rho \,\mathrm{d}\boldsymbol{y})$, we obtain

$$\mathbb{E}[\varphi] = \int_{\Omega_a} \int_{\Omega_f} \varphi(\omega_a, \omega_f) \,\mathrm{d}\mathbb{P}_f(\omega_f) \,\mathrm{d}\mathbb{P}_a(\omega_a) = \int_{\Gamma} \varphi \rho_a \rho_f \,\mathrm{d}\boldsymbol{y},$$

so that indeed $\rho(\boldsymbol{y}) = \rho_a(\boldsymbol{y}_a)\rho_f(\boldsymbol{y}_f)$.

2.3. Stochastic finite element methods

We present a generalized framework for *stochastic finite element methods* (SFEMs), which are finite-element-based spatial semi-discretizations of an SPDE.[2] We also prepare the way for the discussion in Section 2.4 of *stochastic Galerkin methods* (SGMs) which, in our context, are SFEMs for which parameter space discretization is also effected using a Galerkin method. Both discussions rely on a weak formulation for SPDEs.

2.3.1. Galerkin weak formulation of stochastic partial differential equations

Analogous to $L^q_{\mathbb{P}}\big(\Omega; W(D)\big)$ and $L^\infty_{\mathbb{P}}\big(\Omega; W(D)\big)$, we define $L^q_\rho\big(\Gamma; W(D)\big)$ and $L^\infty_\rho\big(\Gamma; W(D)\big)$ as

$$L^q_\rho\big(\Gamma; W(D)\big) := \Big\{ v : \Gamma \to W(D) \mid v \text{ is strongly measurable and } \int_\Gamma \|v(\cdot, \boldsymbol{y})\|^q_{W(D)} \rho(\boldsymbol{y}) \,\mathrm{d}\boldsymbol{y} < +\infty \Big\} \tag{2.3.1}$$

and

$$L^\infty_\rho\big(\Gamma; W(D)\big) := \Big\{ v : \Gamma \to W(D) \mid v \text{ is strongly measurable and } \rho(\boldsymbol{y}) \,\mathrm{d}\boldsymbol{y} - \operatorname*{ess\,sup}_{\boldsymbol{y} \in \Gamma} \|v(\cdot, \boldsymbol{y})\|^2_{W(D)} < +\infty \Big\}. \tag{2.3.2}$$

Based on the discussions in Section 2.2, the solution u of an SPDE can be expressed as $u(\mathbf{x}, \omega) = u\big(\mathbf{x}, y_1(\omega), \ldots, y_N(\omega)\big)$. Then, it is natural to treat $u(\mathbf{x}, \boldsymbol{y})$, a function of d spatial variables and N random parameters, as a function of $d + N$ variables. This leads us to consider a *Galerkin weak formulation* of the SPDE, with respect to both physical and parameter

[2] Recall that, to economize notation, we refer to any partial differential equation with random inputs as a stochastic partial differential equation (SPDE).

space, in the following form. Seek $u \in W(D) \otimes L^q_\rho(\Gamma)$ such that

$$\sum_{k=1}^{K} \int_\Gamma \int_D S_k(u;\boldsymbol{y}) T_k(v) \rho(\boldsymbol{y}) \,\mathrm{d}\mathbf{x}\,\mathrm{d}\boldsymbol{y} \tag{2.3.3}$$
$$= \int_\Gamma \int_D v f(\mathbf{x},\boldsymbol{y}) \rho(\boldsymbol{y}) \,\mathrm{d}\mathbf{x}\,\mathrm{d}\boldsymbol{y} \quad \text{for all } v \in W(D) \otimes L^q_\rho(\Gamma),$$

where $S_k(\cdot,\cdot)$, $k = 1,\dots,K$, are in general nonlinear operators and $T_k(\cdot,\cdot)$, $k = 1,\dots,K$, are linear operators.

Example 2.3.1. A weak formulation of the stochastic PDE in (2.1.3) is given by

$$\int_\Gamma \int_D \big(a(\boldsymbol{y})\nabla u\big) \cdot \nabla v \rho(\boldsymbol{y}) \,\mathrm{d}\mathbf{x}\,\mathrm{d}\boldsymbol{y} + \int_\Gamma \int_D \big(u(\boldsymbol{y})|u(\boldsymbol{y})|^s\big) v \rho(\boldsymbol{y}) \,\mathrm{d}\mathbf{x}\,\mathrm{d}\boldsymbol{y}$$
$$= \int_\Gamma \int_D f(\boldsymbol{y}) v \rho(\boldsymbol{y}) \,\mathrm{d}\mathbf{x}\,\mathrm{d}\boldsymbol{y} \quad \text{for all } v \in H^1_0(D) \otimes L^q_\rho(\Gamma),$$

where we omit reference to the dependence of a, f, u, and v on the spatial variable $\mathbf{x}$ for the sake of economizing notation. For the first term on the left-hand side, we have the linear operators $S_1(u,\boldsymbol{y}) = a(\boldsymbol{y})\nabla u$ and $T_1(v) = \nabla v$; for the second term, we have the nonlinear operator $S_2(u,\boldsymbol{y}) = u(\boldsymbol{y})|u(\boldsymbol{y})|^s$ and the linear operator $T_2(v) = v$.

For our purposes, and without loss of generality, it suffices to consider the single term form of (2.3.3), that is,

$$\int_\Gamma \int_D S(u;\boldsymbol{y}) T(v) \rho(\boldsymbol{y}) \,\mathrm{d}\mathbf{x}\,\mathrm{d}\boldsymbol{y} \tag{2.3.4}$$
$$= \int_\Gamma \int_D v f(\boldsymbol{y}) \rho(\boldsymbol{y}) \,\mathrm{d}\mathbf{x}\,\mathrm{d}\boldsymbol{y} \quad \text{for all } v \in W(D) \otimes L^q_\rho(\Gamma),$$

where $T(\cdot)$ is a linear operator and, in general, $S(\cdot)$ is a nonlinear operator and where again we have suppressed explicit reference to dependences on the spatial variable $\mathbf{x}$.

2.3.2. Spatial finite element semi-discretization

Any method for the approximate solution of an SPDE that uses a finite element method to effect discretization with respect to the spatial variable $\mathbf{x}$ is referred to as a *stochastic finite element method* (SFEM). We assume that all methods discussed in this work use the same finite element method for this purpose. Details about the finite element methods discussed in this article may be found in Brenner and Scott (2008) and Ciarlet (1978), for example.

Let $\mathcal{T}_h$ denote a conforming triangulation of D with maximum mesh size $h > 0$ and let $W_h(D) \subset W(D)$ denote a finite element space, parametrized by $h \to 0$, constructed using the triangulation $\mathcal{T}_h$. Let $\{\phi_j(\mathbf{x})\}_{j=1}^{J_h}$ denote a basis for $W_h(D)$ so that J_h denotes the dimension of $W_h(D)$. We introduce the *semi-discrete* approximation $u_{J_h}(\mathbf{x}, \boldsymbol{y}) \in W_h(D) \otimes L^q_\rho(\Gamma)$ having the form

$$u_{J_h}(\mathbf{x}, \boldsymbol{y}) = \sum_{j=1}^{J_h} c_j(\boldsymbol{y})\phi_j(\mathbf{x}). \tag{2.3.5}$$

At each point in $\boldsymbol{y} \in \Gamma$, the coefficients $c_j(\boldsymbol{y})$, and thus u_{J_h}, are determined by solving the problem

$$\begin{aligned} &\int_\Gamma \int_D S\Big(\sum_{j=1}^{J_h} c_j(\boldsymbol{y})\phi_j(\mathbf{x}); \boldsymbol{y}\Big) T(v)\rho(\boldsymbol{y})\,\mathrm{d}\mathbf{x}\,\mathrm{d}\boldsymbol{y} \\ &\quad = \int_\Gamma \int_D v f(\boldsymbol{y})\rho(\boldsymbol{y})\,\mathrm{d}\mathbf{x}\,\mathrm{d}\boldsymbol{y} \quad \text{for all } v \in W_h(D) \otimes L^q_\rho(\Gamma) \end{aligned} \tag{2.3.6}$$

or, equivalently,

$$\begin{aligned} &\int_D S\Big(\sum_{j=1}^{J_h} c_j(\boldsymbol{y})\phi_j(\mathbf{x}); \boldsymbol{y}\Big) T(\phi_{j'})\,\mathrm{d}\mathbf{x} \\ &\quad = \int_D \phi_{j'} f(\boldsymbol{y})\,\mathrm{d}\mathbf{x} \quad \text{for } j' = 1, \ldots, J_h. \end{aligned} \tag{2.3.7}$$

What this means is that *to obtain the semi-discrete approximation* $u_{J_h}(\mathbf{x}, \boldsymbol{y})$ *at any specific point* $\boldsymbol{y}_0 \in \Gamma$*, one only has to solve a* deterministic *finite element problem by fixing* $\boldsymbol{y} = \boldsymbol{y}_0$ *in* (2.3.7). The subset of Γ in which (2.3.7) has no solution has zero measure with respect to $\rho\,\mathrm{d}\boldsymbol{y}$. For convenience, we assume that the coefficient a and the forcing term f in (2.1.1) admit a smooth extension on $\rho\,\mathrm{d}\boldsymbol{y}$-zero measure sets. Then, (2.3.7) can be extended a.e. in Γ with respect to the Lebesgue measure, instead of the measure $\rho\,\mathrm{d}\boldsymbol{y}$.

2.4. Stochastic Galerkin methods

Stochastic Galerkin finite element methods, which we will refer to simply as *stochastic Galerkin methods* (SGMs), are SFEMs for which discretization with respect to the parameter vector $\boldsymbol{y} \in \Gamma$ is effected using a Galerkin method. Coupled with the spatial finite element discretization in the domain D, this results in a full discretization of the Galerkin weak formulation (2.3.4) in the product domain $D \times \Gamma$.

To this end, let $\mathcal{P}(\Gamma) \subset L^q_\rho(\Gamma)$ denote a finite-dimensional subspace and let $\{\psi_m(\boldsymbol{y})\}_{m=1}^M$ denote a basis for $\mathcal{P}(\Gamma)$ so that $M =$ the dimension of $\mathcal{P}(\Gamma)$. We seek a *fully discrete* approximation of the solution $u(\mathbf{x}, \boldsymbol{y})$ of

(2.3.4) having the form

$$u_{J_h,M}(\mathbf{x},\boldsymbol{y}) = \sum_{m=1}^{M}\sum_{j=1}^{J_h} c_{jm}\phi_j(\mathbf{x})\psi_m(\boldsymbol{y}) \ \in W_h(D)\times\mathcal{P}(\Gamma), \tag{2.4.1}$$

where the coefficients c_{jm}, and thus $u_{J_h,M}$, are determined by solving the problem

$$\begin{aligned}\int_\Gamma\int_D S(u_{J_h,M};\boldsymbol{y})T(v)\rho(\boldsymbol{y})\,\mathrm{d}\mathbf{x}\,\mathrm{d}\boldsymbol{y} \\ = \int_\Gamma\int_D vf(\boldsymbol{y})\rho(\boldsymbol{y})\,\mathrm{d}\mathbf{x}\,\mathrm{d}\boldsymbol{y} \quad \text{for all } v\in W_h(D)\times\mathcal{P}(\Gamma)\end{aligned} \tag{2.4.2}$$

or, equivalently,

$$\begin{aligned}&\int_\Gamma\int_D S\Big(\sum_{m=1}^{M}\sum_{j=1}^{J_h} c_{jm}\phi_j(\mathbf{x})\psi_m(\boldsymbol{y}),\boldsymbol{y}\Big)T\big(\phi_{j'}(\mathbf{x})\big)\psi_{m'}(\boldsymbol{y})\rho(\boldsymbol{y})\,\mathrm{d}\mathbf{x}\,\mathrm{d}\boldsymbol{y}\\ &\quad = \int_\Gamma\int_D \phi_{j'}(\mathbf{x})\psi_{m'}(\boldsymbol{y})f(\mathbf{x},\boldsymbol{y})\rho(\boldsymbol{y})\,\mathrm{d}\mathbf{x}\,\mathrm{d}\boldsymbol{y}\\ &\qquad\qquad \text{for } j'\in\{1,\dots,J_h\} \text{ and } m'\in\{1,\dots,M\},\end{aligned} \tag{2.4.3}$$

where we have used the fact that $T(\cdot)$ is linear and contains no derivatives with respect to $\boldsymbol{y}$.

In general, the integrals in (2.4.3) cannot be evaluated exactly, so quadrature rules must be invoked to effect the approximate evaluation of both the integrals over Γ and D. However, because we assume that all methods discussed treat all aspects of the spatial discretization in the same manner, we focus on the integral over Γ and do not explicitly write down quadrature rules for the integral over D. As such, for some choice of quadrature points $\{\widehat{\boldsymbol{y}}_r\}_{r=1}^R$ in Γ and quadrature weights $\{w_r\}_{r=1}^R$, we have that(2.4.3) is further discretized, resulting in

$$\begin{aligned}&\sum_{r=1}^{R} w_r\rho(\widehat{\boldsymbol{y}}_r)\psi_{m'}(\widehat{\boldsymbol{y}}_r)\\ &\qquad\times\int_D S\Big(\sum_{m=1}^{M}\sum_{j=1}^{J_h} c_{jm}\phi_j(\mathbf{x})\psi_m(\widehat{\boldsymbol{y}}_r),\widehat{\boldsymbol{y}}_r\Big)T\big((\phi_{j'}(\mathbf{x})\big)\,\mathrm{d}\mathbf{x}\\ &\qquad\qquad = \sum_{r=1}^{R} w_r\rho(\widehat{\boldsymbol{y}}_r)\psi_{m'}(\widehat{\boldsymbol{y}}_r)\int_D \phi_{j'}(\mathbf{x})f(\mathbf{x},\widehat{\boldsymbol{y}}_r)\,\mathrm{d}\mathbf{x}\\ &\qquad\qquad\qquad \text{for } j'\in\{1,\dots,J_h\} \text{ and } m'\in\{1,\dots,M\}.\end{aligned} \tag{2.4.4}$$

In general, the discrete problem (2.4.4) is a fully coupled system of J_hM equations in J_hM degrees of freedom c_{jm}, $j=1,\dots,J_h$ and $m=1\dots,M$.

Thus, the fully discrete approximation $u_{J_h,M}(\mathbf{x},\boldsymbol{y})$ of the solution $u(\mathbf{x},\boldsymbol{y})$ of the SPDE can be obtained by solving the *single deterministic problem* (2.4.4).

All approaches discussed in Parts 3, 4 and 5 can be viewed as being special cases of SGMs; they differ in the choices made for the parameter domain approximating space, $\mathcal{P}(\Gamma)$, for the basis $\{\psi_m(\boldsymbol{y})\}_{m=1}^M$, and for the quadrature rule $\{\widehat{\boldsymbol{y}}_r, w_r\}_{r=1}^R$.

PART THREE
Stochastic sampling methods

3.1. General stochastic sampling methods

Stochastic sampling methods (SSMs) for determining statistical information about solutions of SPDEs with parametrized random inputs proceed by first

– choosing M points $\{\boldsymbol{y}_m\}_{m=1}^M$ in the parameter domain $\Gamma \subseteq \mathbb{R}^N$

and then

– determining a spatial finite element approximate solution $u_{J_h}(\mathbf{x};\boldsymbol{y}_m)$ of the SPDE for each chosen parameter point $\boldsymbol{y}_m$.

To be precise, with $\{\phi_j(\mathbf{x})\}_{j=1}^{J_h}$ denoting a finite element basis used for spatial approximation, for each parameter point $\boldsymbol{y}_m$, $m = 1,\dots,M$, we have the approximation

$$u_{J_h}(\mathbf{x};\boldsymbol{y}_m) = \sum_{j=1}^{J_h} c_{jm}\phi_j(\mathbf{x}) \tag{3.1.1}$$

of the solution $u(\mathbf{x},\boldsymbol{y}_m)$ of the SPDE at the parameter point $\boldsymbol{y}_m$. For $m = 1,\dots,M$, the coefficients c_{jm}, $j = 1,\dots,J_h$, are determined from the M *uncoupled* finite element systems

$$\begin{aligned}\int_D S\Big(\sum_{j=1}^{J_h} c_{jm}(\boldsymbol{y}_m)\phi_j(\mathbf{x});\boldsymbol{y}_m\Big)T\big(\phi_{j'}(\mathbf{x})\big)\,\mathrm{d}\mathbf{x} \\ = \int_D f(\mathbf{x},\boldsymbol{y}_m)\phi_{j'}(\mathbf{x})\,\mathrm{d}\mathbf{x} \quad \text{for } j' = 1,\dots,J_h.\end{aligned} \tag{3.1.2}$$

The attraction of SSMs as embodied by (3.1.1) and (3.1.2) is that, clearly, they are embarrassingly easy to implement: one merely wraps a parameter sampling method around a deterministic legacy code for a deterministic partial differential equation, resulting in a code that is also embarrassingly

easy to parallelize. Clearly, after a spatial approximation method is chosen, an approximation of the type (3.1.1) is completely defined by simply specifying how one samples the parameter points $\{\boldsymbol{y}_m\}_{m=1}^M$ in Γ.

As always, there are three types of inputs to consider: a finite set of random parameters, correlated random fields, and uncorrelated random fields. The last two are infinite stochastic processes that require the additional step of approximation in terms of a finite number of random parameters. For the sake of economy of exposition, in this section, we lump the first two into the same discussion. Uncorrelated random fields cannot be lumped into the same discussion, so they are treated separately in Appendix C.

Specifically, under the assumptions discussed in Section 2.2, we consider direct sampling approximations of the solution $u(\mathbf{x}, \boldsymbol{y})$ of an SPDE with parametrized stochastic inputs when *the number N of parameters is finite and is not dependent on the spatial grid size.* Clearly, such a situation arises in problems defined in terms of a finite number of random input parameters. It can seemingly also arise whenever infinite representations, such as Karhunen–Loève expansions, of correlated random field inputs are truncated after the first N terms. However, because both spatial and temporal approximations are present, the value of N may have to be adjusted as spatial grids are refined, to ensure that the error due to truncation of infinite stochastic representations is commensurate with the errors due to spatial approximation. We refer to this case as 'weakly dependent', in contrast to the white noise case considered in Appendix C, for which the dependence of N on the spatial grid size is much more direct. In this section, we ignore the weak dependence completely and assume that N is fixed independent of the spatial grid.

The most used class of SSMs is that of Monte Carlo methods (MCMs), which correspond to drawing independent and identically distributed (i.i.d.) random samples $\boldsymbol{y}_m \in \Gamma$ from the probability density function (PDF) $\rho(\boldsymbol{y})$. As is well known, MCMs result in errors that are statistically $O\big(1/\sqrt{M}\big)$, that is, Monte Carlo methods converge very slowly; this is a significant disadvantage in light of the need to solve the SPDE for every sample point $\boldsymbol{y}_m$ taken. However, the $O\big(1/\sqrt{M}\big)$ convergence behaviour of MCMs holds true for any N, that is, the performance of MCMs is insensitive to the dimension of the parameter space. In contrast, the convergence behaviour of most methods, including most methods discussed in Section 3.5 as well as those of Parts 4 and 5, deteriorate as the dimension N of the parameter space increases; this is a stark manifestation of the curse of dimensionality. As a result, if N is sufficiently large, MCMs require a smaller computational effort for the same accuracy than do most other methods discussed in this article. Thus, one can view most efforts to develop new methods, including most of those in Section 3.5 and Parts 4 and 5, as attempts to increase the value of N at which MCMs start winning.

MCMs have the additional important advantage over all other methods of near-universal applicability in the sense that their performance is not affected by the smoothness – or lack thereof – of $u(\mathbf{x}, \boldsymbol{y})$ or $u_{J_h}(\mathbf{x}, \boldsymbol{y})$. This is in contrast to the polynomial-based methods discussed in Parts 4 and 5, which do require some degree of smoothness with respect to $\boldsymbol{y}$ to achieve their advertised accuracy. Thus, in some cases, for example when $u(\mathbf{x}, \boldsymbol{y})$ or $u_{J_h}(\mathbf{x}, \boldsymbol{y})$ are discontinuous functions of $\boldsymbol{y}$, MCMs may converge as fast or faster than other methods. Note that the slow convergence of MCMs, even for smooth dependences on $\boldsymbol{y}$, is, in fact, a negative consequence of the insensitivity of the method to smooth dependence on $\boldsymbol{y}$, that is, MCMs converge in the same way irrespective of that smoothness.

The very slow convergence of MCMs has given rise to extensive efforts directed at inventing other simple sampling strategies that improve on the $O(1/\sqrt{M})$ behaviour of MCMs, and for which the growth of the error with respect to increasing N is manageable for at least moderately large values of N. We briefly consider some such methods in Section 3.5. Of course, it also gives rise to an interest in developing more complex discretization strategies such as those discussed in Parts 4 and 5, which include, among other methods, additional examples of SSMs.

Before moving on to the discussion of specific methods, we first discuss the connection between SSMs and stochastic Galerkin methods (SGMs) discussed in Section 2.4. Despite the connection between them, which is established in Section 3.2, we will then take the more traditional and straightforward approach discussed so far for defining SSMs, that is, simply choose a set of sample parameter points in the parameter domain Γ and then solve the SPDE for each of these points.

3.2. The relation between stochastic sampling and stochastic Galerkin methods

There are insights to be gained, especially in comparing SSMs to other approaches for approximately solving SPDEs, by showing that SSMs can be placed into the SGM framework, albeit for a specific choice of basis functions.

We begin by choosing, in $L^q_\rho(\Gamma)$, the approximating parameter space to consist of global polynomials. For a basis, we choose the Lagrange fundamental polynomials $\{\ell_m(\boldsymbol{y})\}_{m=1}^M$ based on a set of interpolating points $\{\boldsymbol{y}_m\}_{m=1}^M$ in Γ. These basis functions satisfy the 'delta property'

$$\ell_m(\boldsymbol{y}_{m'}) = \delta_{mm'} \quad \text{for } m, m' = 1, \ldots, M. \tag{3.2.1}$$

Thus, the parameter approximating space is now $\mathcal{P}(\Gamma) := \operatorname{span}\{\ell_m(\boldsymbol{y})\}_{m=1}^M$

and the Lagrange polynomial interpolant is given by

$$u_{J_h,M}(\mathbf{x},\boldsymbol{y}) = \sum_{m=1}^{M} u_{J_h}(\mathbf{x},\boldsymbol{y}_m)\ell_m(\boldsymbol{y}) = \sum_{j=1}^{J_h}\sum_{m=1}^{M} c_{jm}\phi_j(\mathbf{x})\ell_m(\boldsymbol{y}), \tag{3.2.2}$$

where, as always, $\{\phi_j(\mathbf{x})\}_{j=1}^{J_h}$ denotes the finite element basis used to effect spatial discretization. The discrete system (2.4.4) from which the coefficients c_{jm}, $j = 1\ldots, J_h$ and $m = 1,\ldots,M$, are determined is now given by

$$\begin{aligned}\sum_{r=1}^{R} w_r\rho(\widehat{\boldsymbol{y}}_r)\ell_{m'}(\widehat{\boldsymbol{y}}_r) \\ \times \int_D S\Big(\sum_{m=1}^{M}\sum_{j=1}^{J_h} c_{jm}\phi_j(\mathbf{x})\ell_m(\widehat{\boldsymbol{y}}_r),\widehat{\boldsymbol{y}}_r\Big)T\big((\phi_{j'}(\mathbf{x})\big)\,\mathrm{d}\mathbf{x} \\ = \sum_{r=1}^{R} w_r\rho(\widehat{\boldsymbol{y}}_r)\ell_{m'}(\boldsymbol{y}_r)\int_D \phi_{j'}(\mathbf{x})f(\mathbf{x},\widehat{\boldsymbol{y}}_r)\,\mathrm{d}\mathbf{x} \\ \text{for } j' \in \{1,\ldots,J_h\} \text{ and } m' \in \{1,\ldots,M\}.\end{aligned} \tag{3.2.3}$$

At this juncture, we have two sets of points:

- the set $\{\boldsymbol{y}_m\}_{m=1}^{M}$ of interpolation points used to construct the Lagrange interpolant of the solution of the SPDE (see (3.2.2));
- the set $\{\widehat{\boldsymbol{y}}_r\}_{r=1}^{R}$ of quadrature points used to approximate parameter integrals appearing in the discretization of the SPDE (see (3.2.3)).

Suppose we choose the two sets to be the same. In this case, because of the delta property of the Lagrange fundamental polynomials, it is easy to see that (3.2.3) reduces to (3.1.2). Thus, we have shown that SSMs are SGMs for which:

- approximation with respect to the random parameters is effected using interpolatory polynomial approximations with Lagrange fundamental polynomial bases;
- the interpolation points are also used as quadrature points for approximating parameter integrals in the stochastic Galerkin equations.

3.3. Classical Monte Carlo sampling

The classical MCM approximation $u^{\mathrm{MC}}_{J_h,M}(\mathbf{x})$ of the solution of an SPDE is defined by

$$u^{\mathrm{MC}}_{J_h,M}\big(\mathbf{x};\{\boldsymbol{y}_m\}_{m=1}^{M}\}\big) = \frac{1}{M}\sum_{m=1}^{M} u_{J_h}(\mathbf{x},\boldsymbol{y}_m) \quad \text{for all } \mathbf{x}\in D,$$

where the i.i.d. sample points $\{\boldsymbol{y}_m\}_{m=1}^M$ in Γ are drawn from the PDF $\rho(\boldsymbol{y})$ and, for each sample point $\boldsymbol{y}_m \in \Gamma$, $u_{J_h}(\mathbf{x}, \boldsymbol{y}_m)$ denotes the solution (3.1.1) of the deterministic finite element system (3.1.2). Note that $u^{\mathrm{MC}}_{J_h,M}\big(\mathbf{x}; \{\boldsymbol{y}_m\}_{m=1}^M\}\big)$ is itself random, in fact, it is a function of MN random parameters, namely the N components of the M random vectors $\boldsymbol{y}_m$; each of these vectors also has the PDF $\rho(\boldsymbol{y})$. We then have that, for each[3] $\mathbf{x} \in D$,

$$\mathbb{E}\big[u^{\mathrm{MC}}_{J_h,M}\big] = \mathbb{E}\left[\frac{1}{M}\sum_{m=1}^{M} u_{J_h}(\boldsymbol{y}_m)\right] = \frac{1}{M}\sum_{m=1}^{M} \mathbb{E}\big[u_{J_h}(\boldsymbol{y}_m)\big] = \mathbb{E}\big[u_{J_h}(\boldsymbol{y})\big],$$

that is, the Monte Carlo approximation is *unbiased.* The goal is to derive an estimate for $\mathbb{E}\big[\|u^{\mathrm{MC}}_{J_h,M} - \mathbb{E}[u(\boldsymbol{y})]\|_{\widetilde{W}(D)}\big]$, where $\widetilde{W}(D)$ is a spatial function space that is appropriate for the SPDE considered. We have that

$$\begin{aligned} &\big\|u^{\mathrm{MC}}_{J_h,M} - \mathbb{E}[u(\boldsymbol{y})]\big\|_{\widetilde{W}(D)} \\ &\quad\le \underbrace{\big\|\mathbb{E}[u_{J_h}(\boldsymbol{y})] - \mathbb{E}[u(\boldsymbol{y})]\big\|_{\widetilde{W}(D)}}_{\text{error due to spatial discretization}} + \underbrace{\big\|u^{\mathrm{MC}}_{J_h,M} - \mathbb{E}[u_{J_h}(\boldsymbol{y})]\big\|_{\widetilde{W}(D)}}_{\text{error due to Monte Carlo sampling}}. \end{aligned} \tag{3.3.1}$$

Thus, the error is estimated by separately estimating the errors due to spatial discretization and Monte Carlo sampling.

3.3.1. Spatial discretization error

We have assumed that for any chosen $\boldsymbol{y}$, the spatial approximation $u_{J_h}(\boldsymbol{y})$ of the solution of the SPDE is obtained using a finite element method. Then, for any $\boldsymbol{y} \in \Gamma$, the spatial error $\|u - u_{J_h}\|_{W(D)}$ can often be approximated by means of traditional finite element analyses. For second-order elliptic partial differential equations (PDEs) with homogeneous Dirichlet boundary conditions, under standard assumptions on the spatial domain D and the data, one can choose

$$\widetilde{W}(D) = W(D) = H^1_0(D) \quad \text{or} \quad \widetilde{W}(D) = L^2(D) = H^0(D) \subset W(D),$$

that is, we can measure the error in either the $H^1_0(D)$- or $L^2(D)$-norms. One can then construct $u_{J_h}(\cdot, \boldsymbol{y}) \in W_h(D) \subset H^1_0(D)$, where $W_h(D)$ denotes a standard finite element space of continuous piecewise polynomials of degree at most p based on a regular triangulation $\mathcal{T}_h$ of the spatial domain D with maximum mesh spacing parameter $h := \max_{\tau\in\mathcal{T}_h} \mathrm{diam}(\tau)$. We then have the error estimate (Brenner and Scott 2008, Ciarlet 1978)

$$\|u_{J_h}(\cdot, \boldsymbol{y}) - u(\cdot, \boldsymbol{y})\|_{H^s(D)} \le C_f h^{p+1-s} \|u(\cdot, \boldsymbol{y})\|_{H^{p+1}(D)} \tag{3.3.2}$$

[3] Again, when there is no danger of ambiguity, we suppress explicit reference to dependences on the spatial variable $\mathbf{x}$. For the same reason, we sometimes simply write $u^{\mathrm{MC}}_{J_h,M}$ for $u^{\mathrm{MC}}_{J_h,M}\big(\mathbf{x}; \{\boldsymbol{y}_m\}_{m=1}^M\}\big)$.

for $s = 0, 1$ and for a.e. $\boldsymbol{y} \in \Gamma$, where $C_f > 0$ can be chosen independent of $\boldsymbol{y}$ and h. For finite element error estimates under less rigid conditions, see Grisvard (1985), for example.

We also have that, with $\nabla_1 = \nabla$ and $\nabla_0 =$ the identity operator,

$$\begin{aligned}
\|\mathbb{E}[u_{J_h}(\boldsymbol{y})] - \mathbb{E}[u(\boldsymbol{y})]\|^2_{H^s(D)} &= \int_D \left| \nabla_s \int_\Gamma \big(u_{J_h}(\boldsymbol{y}) - u(\boldsymbol{y})\big)\rho(\boldsymbol{y})\,\mathrm{d}\boldsymbol{y} \right|^2 \mathrm{d}\mathbf{x} \\
&\le \int_D \left| \int_\Gamma \nabla_s \big(u_{J_h}(\boldsymbol{y}) - u(\boldsymbol{y})\big)\rho(\boldsymbol{y})\,\mathrm{d}\boldsymbol{y} \right|^2 \mathrm{d}\mathbf{x} \\
&\le \int_D \int_\Gamma \big|\nabla_s \big(u_{J_h}(\boldsymbol{y}) - u(\boldsymbol{y})\big)\big|^2 \rho(\boldsymbol{y})\,\mathrm{d}\boldsymbol{y}\,\mathrm{d}\mathbf{x} \\
&= \int_\Gamma \left(\int_D \big|\nabla_s \big(u_{J_h}(\boldsymbol{y}) - u(\boldsymbol{y})\big)\big|^2 \,\mathrm{d}\mathbf{x} \right) \rho(\boldsymbol{y})\,\mathrm{d}\boldsymbol{y} \\
&= \mathbb{E}[\|u_{J_h}(\boldsymbol{y}) - u(\boldsymbol{y})\|^2_{H^s(D)}]
\end{aligned}$$

for $s = 0$ or 1. Then, combining with (3.3.2), we have that, for $s = 0, 1$,

$$\|\mathbb{E}[u_{J_h}(\boldsymbol{y})] - \mathbb{E}[u(\boldsymbol{y})]\|_{H^s(D)} \le C_f h^{p+1-s} \big(\mathbb{E}\big[\|u\|^2_{H^{p+1}(D)}\big]\big)^{1/2}. \tag{3.3.3}$$

3.3.2. Monte Carlo sampling error

We introduce the shorthand $\overline{u}_{J_h} = \mathbb{E}[u_{J_h}(\boldsymbol{y})]$ and $\widetilde{u}_{J_h,m} = u_{J_h}(\boldsymbol{y}_m)$. Then

$$\begin{aligned}
&\mathbb{E}\big[\|u^{\mathrm{MC}}_{J_h,M} - \mathbb{E}[u_{J_h}(\boldsymbol{y})]\|^2_{H^1(D)}\big] = \mathbb{E}\big[\|u^{\mathrm{MC}}_{J_h,M} - \overline{u}_{J_h}\|^2_{H^1(D)}\big] \\
&= \mathbb{E}\left[\left\| \frac{1}{M} \sum_{m=1}^M \widetilde{u}_{J_h,m} - \overline{u}_{J_h} \right\|^2_{H^1(D)} \right] \\
&= \mathbb{E}\left[\int_D \left| \frac{1}{M} \sum_{m=1}^M \nabla\big(\widetilde{u}_{J_h,m} - \overline{u}_{J_h}\big) \right|^2 \mathrm{d}\mathbf{x} \right] \\
&= \frac{1}{M^2} \mathbb{E}\left[\int_D \left(\sum_{m=1}^M \nabla\big(\widetilde{u}_{J_h,m} - \overline{u}_{J_h}\big) \right) \cdot \left(\sum_{m'=1}^M \nabla\big(\widetilde{u}_{J_h,m'} - \overline{u}_{J_h}\big) \right) \mathrm{d}\mathbf{x} \right] \\
&= \frac{1}{M^2} \mathbb{E}\left[\int_D \sum_{m=1}^M \nabla\big(\widetilde{u}_{J_h,m} - \overline{u}_{J_h}\big) \cdot \nabla\big(\widetilde{u}_{J_h,m} - \overline{u}_{J_h}\big)\,\mathrm{d}\mathbf{x} \right] \\
&\quad + \frac{1}{M^2} \mathbb{E}\left[\int_D \sum_{\substack{m=1 \\ m \ne m'}}^M \sum_{m'=1}^M \nabla\big(\widetilde{u}_{J_h,m} - \overline{u}_{J_h}\big) \cdot \nabla\big(\widetilde{u}_{J_h,m'} - \overline{u}_{J_h}\big)\,\mathrm{d}\mathbf{x} \right]
\end{aligned}$$

$$= \frac{1}{M^2}\int_D \mathbb{E}\left[\sum_{m=1}^{M} \left|\nabla\big(\widetilde{u}_{J_h,m} - \overline{u}_{J_h}\big)\right|^2\right] \mathrm{d}\mathbf{x}$$

$$+ \frac{1}{M^2}\int_D \sum_{\substack{m=1\\ m\neq m'}}^{M}\sum_{m'=1}^{M} \mathbb{E}\big[\nabla\big(\widetilde{u}_{J_h,m} - \overline{u}_{J_h}\big)\cdot\nabla\big(\widetilde{u}_{J_h,m'} - \overline{u}_{J_h}\big)\big]\,\mathrm{d}\mathbf{x}$$

$$= \frac{1}{M^2}\int_D \mathbb{E}\left[\sum_{m=1}^{M} \left|\nabla\big(\widetilde{u}_{J_h,m} - \mathbb{E}[\widetilde{u}_{J_h,m}]\big)\right|^2\right] \mathrm{d}\mathbf{x}$$

$$+ \frac{1}{M^2}\int_D \sum_{\substack{m=1\\ m\neq m'}}^{M}\sum_{m'=1}^{M} \mathbb{E}\big[\nabla\big(\widetilde{u}_{J_h,m} - \mathbb{E}[\widetilde{u}_{J_h,m}]\big)\cdot\nabla\big(\widetilde{u}_{J_h,m'} - \mathbb{E}[\widetilde{u}_{J_h,m'}]\big)\big]\,\mathrm{d}\mathbf{x}$$

$$= \frac{1}{M^2}\int_D \mathbb{E}\left[\sum_{m=1}^{M} \left|\nabla\big(\widetilde{u}_{J_h,m} - \mathbb{E}[\widetilde{u}_{J_h,m}]\big)\right|^2\right] \mathrm{d}\mathbf{x}$$

$$= \frac{1}{M}\int_D \mathbb{E}\Big[\left|\nabla\big(u_{J_h}(\boldsymbol{y}) - \mathbb{E}[\nabla u_{J_h}(\boldsymbol{y})]\big)\right|^2\Big]\,\mathrm{d}\mathbf{x}$$

$$= \frac{1}{M}\int_D \mathbb{E}\Big[\left|\nabla u_{J_h}(\boldsymbol{y}) - \mathbb{E}[\nabla u_{J_h}(\boldsymbol{y})]\right|^2\Big]\,\mathrm{d}\mathbf{x} = \frac{1}{M}\int_D \sigma\big(\nabla u_{J_h}(\boldsymbol{y})\big)\,\mathrm{d}\mathbf{x}.$$

In a similar but simpler manner, we have that

$$\mathbb{E}\big[\|u^{\mathrm{MC}}_{J_h,M} - \mathbb{E}[u_{J_h}(\boldsymbol{y})]\|^2_{L^2(D)}\big] = \frac{1}{M}\int_D \sigma\big(u_{J_h}(\boldsymbol{y})\big)\,\mathrm{d}\mathbf{x}.$$

Then, for $s = 0$ or 1, we have

$$\begin{aligned}\mathbb{E}\big[\|u^{\mathrm{MC}}_{J_h,M} - \mathbb{E}[u_{J_h}(\boldsymbol{y})]\|_{H^s(D)}\big] &\le \big(\mathbb{E}\big[\|u^{\mathrm{MC}}_{J_h,M} - \mathbb{E}[u_{J_h}(\boldsymbol{y})]\|^2_{H^s(D)}\big]\big)^{1/2}\\ &= \frac{1}{\sqrt{M}}\left(\int_D \sigma\big(\nabla_s u_{J_h}(\boldsymbol{y})\big)\,\mathrm{d}\mathbf{x}\right)^{1/2}.\end{aligned} \tag{3.3.4}$$

Substituting (3.3.3) and (3.3.4) into (3.3.1), we obtain the estimate for the combined spatial discretization and sampling error given by

$$\begin{aligned}\mathbb{E}\big[\|u^{\mathrm{MC}}_{J_h,M} - \mathbb{E}[u(\boldsymbol{y})]\|_{H^s(D)}\big] \le C_f h^{p+1-s}\big(\mathbb{E}\big[\|u\|^2_{H^{p+1}(D)}\big]\big)^{1/2}\\ + \frac{1}{\sqrt{M}}\left(\int_D \sigma\big(\nabla_s u_{J_h}(\boldsymbol{y})\big)\,\mathrm{d}\mathbf{x}\right)^{1/2} \quad \text{for } s = 0 \text{ or } 1,\end{aligned} \tag{3.3.5}$$

where, again, $\nabla_1 = \nabla$ and ∇_0 denotes the identity operator.

To illustrate how the estimate (3.3.5) is used to relate the number of MC samples M to the spatial grid size h, we assume that:

- for each $y \in \Gamma$, $u(\boldsymbol{y}) \in H^{p+1}(D)$;
- a finite element space consisting of piecewise polynomials of degree p is used for spatial discretization;
- for $s = 0$ or 1, the variance of $\nabla_s u_{J_h}(\boldsymbol{y})$ is bounded.

Then, it follows that there exist constants $C_{\text{space}}(p, s, u)$ and $C_{\text{sampling}}(s, u_{J_h})$ such that

$$\mathbb{E}\big[\|u^{\text{MC}}_{J_h,M} - \mathbb{E}[u(\boldsymbol{y})]\|_{H^s(D)}\big] \le C_{\text{space}} h^{p+1-s} + \frac{C_{\text{sampling}}}{\sqrt{M}}. \tag{3.3.6}$$

Then, if we wish the spatial discretization error and sampling error contributions to the total error to be balanced, we would arrange things such that

$$\frac{C_{\text{sampling}}}{\sqrt{M}} \approx C_{\text{space}} h^{p+1-s},$$

and, to achieve that balance, we would need

$$M = O(h^{-2(p+1-s)})$$

samples.

For example, for $h = 0.01$, if we choose $p = 1$ (a piecewise linear finite element space) and $s = 1$ (we are interested in $H^1(D)$ errors), we need $M \approx 10^4$. On the other hand, if we choose $p = 2$ (a piecewise quadratic finite element space) and $s = 0$ (we are interested in $L^2(D)$ errors), we need $M \approx 10^{12}$.

Of course, the values of C_{space} and C_{sampling} influence the error, so that good estimates for these constants are important for minimizing the number of samples needed to render the sampling error to be balanced with the spatial discretization error. However, it is clear that the slow convergence of MCMs can result in a large value of M, that is, a large number of PDE solves, for even moderate values of the grid size h.

Note that even if we choose $s = 1$ and $p = 2$, we need $M \approx 10^8$ so that just by using quadratic instead of linear finite element spaces, the number of MCM samples needed to balance the spatial and sampling errors is squared. This rapid growth with respect to spatial accuracy has the consequence that, in practice, usually an insufficient number of samples are taken to render the sampling error comparable to the spatial discretization error; this is certainly true if one uses higher-order accurate finite element spatial approximations.

From (3.3.4) and (3.3.6), it is clear that smaller $\int_D \sigma\big(\nabla_s u_{J_h}(\boldsymbol{y})\big)\, \mathrm{d}\mathbf{x}$ means that a smaller number of samples is needed to make the sampling error

commensurate with the spatial discretization error, that is, smaller variances require less sampling.

The estimate (3.3.5) is in terms of expectations. In practice, the error behaves erratically as M increases; for example, it is certainly not monotone with increasing M and may, in fact, at times increase dramatically as M is incremented upwards.

3.4. Multilevel Monte Carlo methods

The sampling error estimate (3.3.4) involves the number of samples M and an average variance determined from the finite element approximation of the solution of the SPDE. To obtain smaller errors, one can attempt to devise methods that converge faster with respect to M or one can try to do something to reduce the variance. The former approach is considered in Section 3.5, whereas the latter is the subject of this section. There have been many efforts (*e.g.*, Hammersley and Handscomb 1964, Kahn and Marshall 1953, Press, Teukolsky, Vetterling and Flannery 2007, Ripley 1987, Rubinstein 1981, Smith, Shafi and Gao 1997, Srinivasan 2002) devoted to variance reduction in the MCM framework. Here, we briefly discuss multilevel Monte Carlo methods (MLMCMs) (Barth *et al.* 2011, Barth *et al.* 2013, Barth and Lang 2012, Charrier, Scheichl and Teckentrup 2013, Giles 2008, Ketelsen, Scheichl and Teckentrup 2013, Cliffe *et al.* 2011), which are intimately connected to spatial discretizations.

Starting with a coarse spatial grid with grid size h_0, we determine a set of increasingly finer spatial grids by subdivision so that, for some integer $K > 1$,

$$h_l = h_{l-1}/K \quad \text{or} \quad h_l = K^{-l}h_0 \quad \text{for } l = 0, \ldots, L.$$

At each level l, we have a spatially approximate solution $u_{h_l}(\boldsymbol{y})$ of the SPDE. Obviously, the approximate solution $u_{h_L}(\boldsymbol{y})$ on the finest grid can be written as

$$u_{h_L}(\boldsymbol{y}) = u_{h_0}(\boldsymbol{y}) + \sum_{l=1}^{L} \big(u_{h_l}(\boldsymbol{y}) - u_{h_{l-1}}(\boldsymbol{y})\big). \tag{3.4.1}$$

We express this more economically as

$$u_{h_L}(\boldsymbol{y}) = \sum_{l=0}^{L} \Delta_{h_l}(\boldsymbol{y}), \tag{3.4.2}$$

where

$$\Delta_{h_0}(\boldsymbol{y}) = u_{h_0}(\boldsymbol{y}) \quad \text{and} \quad \Delta_{h_l}(\boldsymbol{y}) = u_{h_l}(\boldsymbol{y}) - u_{h_{l-1}}(\boldsymbol{y}) \quad \text{for } l = 1, \ldots, L.$$

For each $l = 1, \ldots, L$, we determine an MCM approximation of $\Delta u_{h_l}(\boldsymbol{y})$

using M_l samples, that is, we have

$$\Delta^{\mathrm{MC}}_{h_l,M_l}\big(\{\boldsymbol{y}_{m_l}\}_{m_l=1}^{M_l}\big) = \frac{1}{M_l}\sum_{m_l=0}^{M_l}\Delta_{h_l}(\boldsymbol{y}_{m_l}).$$

The MLMCM approximation of $u_{h_L}(\boldsymbol{y})$ is then defined as

$$u^{\mathrm{MLMC}}_{h_L,M}\big(\cup_{l=0}^{L}\{\boldsymbol{y}_{m_l}\}_{m_l=1}^{M_l}\big) = \sum_{l=0}^{L}\Delta^{\mathrm{MC}}_{h_l,M_l}\big(\{\boldsymbol{y}_{m_l}\}_{m_l=1}^{M_l}\big), \tag{3.4.3}$$

where the total number of samples taken is $M = \sum_{l=0}^{L} M_l$. Note that we do not apply the MCM to any $u_{h_l}(\boldsymbol{y})$ for $l > 0$, but rather to the differences $\Delta_{h_l}(\boldsymbol{y}) = u_{h_l}(\boldsymbol{y}) - u_{h_{l-1}}(\boldsymbol{y})$.

As for the MCM, we split the error into spatial and sampling errors so that

$$\begin{aligned}&\mathbb{E}\big[\|u^{\mathrm{MLMC}}_{h_L,M} - \mathbb{E}[u(\boldsymbol{y})]\|_{H^s(D)}\big]\\ &\quad\le \underbrace{\|\mathbb{E}[u_{h_L}(\boldsymbol{y})] - \mathbb{E}[u(\boldsymbol{y})]\|_{H^s(D)}}_{\substack{\text{error due to}\\ \text{spatial discretization}}} + \underbrace{\mathbb{E}\big[\|u^{\mathrm{MLMC}}_{h_L,M} - \mathbb{E}[u_{h_L}(\boldsymbol{y})]\|_{H^s(D)}\big]}_{\substack{\text{error due to}\\ \text{multilevel Monte Carlo sampling}}}.\end{aligned}$$

The spatial error does not depend on what sampling method we use, so it is the same as that for the MCM method, and again we have, from (3.3.3), that

$$\|\mathbb{E}[u_{h_L}(\boldsymbol{y})] - \mathbb{E}[u(\boldsymbol{y})]\|_{H^s(D)} = O(h_L^{\alpha}) \quad \text{with } \alpha = p + 1 - s$$

for $s = 0$ or 1, where p denotes the degree of the piecewise polynomials used to effect the spatial finite element discretization. If we want half the error $\varepsilon/2$ to be due to spatial discretization, we have that

$$h_L = O(\varepsilon^{1/\alpha}). \tag{3.4.4}$$

The sampling error is now the sum of the errors due to the $(L+1)$ MCM approximations, that is, noting that

$$\mathbb{E}[u_{h_L}(\boldsymbol{y})] = \mathbb{E}\left[\sum_{l=0}^{L}\Delta_{h_l}(\boldsymbol{y})\right] = \sum_{l=0}^{L}\mathbb{E}[\Delta_{h_l}(\boldsymbol{y})],$$

using (3.3.4) and (3.4.2) and setting

$$\overline{\sigma}_l = \int_D \sigma\big(\nabla_s\Delta_{h_l}(\boldsymbol{y})\big)\,\mathrm{d}\mathbf{x} \quad \text{for } l = 0,\dots,L,$$

we have

$$\begin{aligned}&\big(\mathbb{E}\big[\|u^{\mathrm{MLMC}}_{h_L,M} - \mathbb{E}[u_{h_L}(\boldsymbol{y})]\|_{H^s(D)}\big]\big)^2\\ &\quad\le \mathbb{E}\big[\|u^{\mathrm{MLMC}}_{h_L,M} - \mathbb{E}[u_{h_L}(\boldsymbol{y})]\|^2_{H^s(D)}\big]\end{aligned}$$

$$
\begin{aligned}
&= \mathbb{E}\left[\left\|\sum_{l=0}^{L}\left(\Delta_{h_l,M_l}^{\mathrm{MC}} - \mathbb{E}[\Delta_{h_l}(\boldsymbol{y})]\right)\right\|_{H^s(D)}^2\right] \\
&\le (L+1)\sum_{l=0}^{L}\mathbb{E}\left[\left\|\frac{1}{M_l}\sum_{m_l=1}^{M_l}\Delta_{h_l}(\boldsymbol{y}_{m_l}) - \mathbb{E}\big[\Delta_{h_l}(\boldsymbol{y})\big]\right\|_{H^s(D)}^2\right] \\
&\le (L+1)\sum_{l=0}^{L}\frac{\overline{\sigma}_l}{M_l}.
\end{aligned}
$$

We next consider how to choose the number of samples M_l, $l = 0, \ldots, L$, to use for each of the $L+1$ MCM calculations. We do this by minimizing the total cost of those $L+1$ calculations. Let $\mathcal{C}_l$ denote the cost incurred to determine the approximate solutions of the PDE using the grid of size h_l. Then, the total sampling cost is given by

$$
\mathcal{C}_{\mathrm{sampling}} = \sum_{l=0}^{L} M_l \mathcal{C}_l. \tag{3.4.5}
$$

We also want roughly half the total error to be due to the sampling error, so we want

$$
\mathbb{E}\big[\|u_{h_L,M}^{\mathrm{MLMC}} - \mathbb{E}[u_{h_L}(\boldsymbol{y})]\|_{H^s(D)}\big] \approx \frac{\varepsilon}{2},
$$

which we can guarantee by setting

$$
(L+1)\sum_{l=0}^{L}\frac{\overline{\sigma}_l}{M_l} = \frac{\varepsilon^2}{4}. \tag{3.4.6}
$$

Thus, we choose $\{M_l\}_{l=0}^{L}$ by minimizing the total sampling cost (3.4.5) subject to the constraint (3.4.6). This results in the choice

$$
M_l = \left\lceil \frac{4(L+1)}{\varepsilon^2}\left(\frac{\overline{\sigma}_l}{\mathcal{C}_l}\right)^{1/2}\sum_{l=0}^{L}\sqrt{\mathcal{C}_i\overline{\sigma}_l}\right\rceil^{+},
$$

where $\lceil\cdot\rceil^{+}$ denotes rounding to the nearest larger integer, and the total sampling cost

$$
\mathcal{C}_{\mathrm{sampling}} = \frac{4(L+1)}{\varepsilon^2}\left(\sum_{l=0}^{L}\sqrt{\mathcal{C}_i\overline{\sigma}_l}\right)^2.
$$

We assume that the cost of solving the PDE increases and the average variance decreases as the grid size decreases. Specifically, we assume that for some positive constants β, γ, C, and $C_{\overline{\sigma}}$, we have

$$
\mathcal{C}_l = C h_l^{-\gamma} \quad \text{and} \quad \overline{\sigma}_l = C_{\overline{\sigma}}\, h_l^{\beta} \tag{3.4.7}
$$

so that, for some positive constant C,

$$\mathcal{C}_{\text{sampling}} \approx \frac{C}{\varepsilon^2}\left(\sum_{l=0}^{L} h_l^{\frac{\beta-\gamma}{2}}\right)^2.$$

If $\beta > \gamma$, that is, if, as l increases, the variance integral $\overline{\sigma}_l$ decreases faster than the cost $\mathcal{C}_l$ increases, then the $l = 0$ term in the right-hand side of (3.4.7) dominates and

$$\mathcal{C}_{\text{sampling}} = O(\varepsilon^{-2}). \tag{3.4.8}$$

On the other hand, if $\gamma > \beta$ so that $\overline{\sigma}_l$ decreases more slowly than $\mathcal{C}_l$ increases, then the $l = L$ term dominates and we have

$$\mathcal{C}_{\text{sampling}} = O(\varepsilon^{-2} h_L^{\beta-\gamma}) = O(\varepsilon^{-2-\frac{\gamma-\beta}{\alpha}}), \tag{3.4.9}$$

where we have used (3.4.4) to relate h_L to ε. Because we have equilibrated the sampling and spatial discretization errors, the relations (3.4.8) and (3.4.9) also hold for the total cost, that is, we have

$$\mathcal{C}_{\text{MLMC}} = \mathcal{C}_{\text{spatial}} + \mathcal{C}_{\text{sampling}} = \begin{cases} O(\varepsilon^{-2}) & \text{if } \beta > \gamma, \\ O\big(\varepsilon^{-2-\frac{\gamma-\beta}{\alpha}}\big) & \text{if } \beta < \gamma. \end{cases}$$

For the MCM applied on the finest grid with grid size h_L, the cost $\mathcal{C}_{\text{MC}} = O(\varepsilon^{-2-\frac{\gamma}{\alpha}})$ so that

$$\frac{\mathcal{C}_{\text{MLMC}}}{\mathcal{C}_{\text{MC}}} = \begin{cases} O\big(\frac{\varepsilon^{-2}}{\varepsilon^{-2-\gamma/\alpha}}\big) = O(\varepsilon^{\gamma/\alpha}) & \text{if } \beta > \gamma, \\ O\big(\frac{\varepsilon^{-2-(\gamma-\beta)/\alpha}}{\varepsilon^{-2-\gamma/\alpha}}\big) = O(\varepsilon^{\beta/\alpha}) & \text{if } \beta < \gamma. \end{cases}$$

Thus, we see that in either case, the MLMCM results in a reduction in cost compared to the MCM.

Thus, we have seen that the key to the greater efficiency of the multilevel Monte Carlo method compared to the Monte Carlo method is writing the approximate solution of the SPDE as the telescoping sum (3.4.1), which is based on a set of successively refined grids. As a result, we have the following:

- one has to do relatively lots of sampling when the realizations of the solution of the PDE are relatively cheap, that is, M_l is large when h_l is large;
- one has to do relatively little sampling when the realizations of the solution of the PDE are relatively expensive, that is, M_l is small when h_l is small.

3.5. Other sampling methods

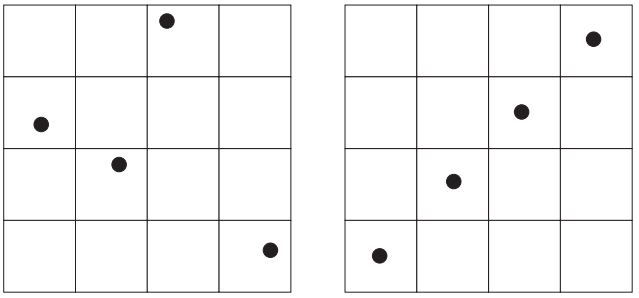
The slow convergence of the Monte Carlo method has led to a huge effort aimed at defining sampling methods that result in faster convergence. Here, we give a very brief review of some of these sampling methods, many of which, unlike the Monte Carlo approach, are purely deterministic. Among the sampling methods we do not describe are stratified, orthogonal, importance, and lattice sampling. Descriptions of these methods are readily available in the literature: see, for example, Hammersley and Handscomb (1964), Kahn and Marshall (1953), Press, Teukolsky, Vetterling and Flannery (2007), Ripley (1987), Rubinstein (1981), Smith, Shafi and Gao (1997) and Srinivasan (2002). We will also encounter additional sampling approaches elsewhere in this article.

3.5.1. Uniform sampling in hypercubes

Quasi-Monte Carlo sequences (QMC). The descriptor 'sequences' refers to the fact that the QMC points are sampled one at a time so that an M-point set retains all the points of the $M-1$ point set; in this regard, QMC and MC sampling are alike. Unlike MC, the QMC samples are deterministically defined. Many QMC sequences have been defined, including the Faure, Halton, Niederreiter, and Sobol sequences, to name just a few. As an example, consider *Halton sequences*, which are determined according to the following procedure. Given a prime number p, any $m \in \mathbb{N}$ can be represented as $m = \sum_i m_i p^i$ for some m_i. Define the mapping H_p from $\mathbb{N}$ to $[0,1]$ by $H_p(m) = \sum_i m_i/p^{i+1}$. Then the Halton sequence of of M points in N dimensions is given by $\{H_{p_1}(m), H_{p_2}(m), \ldots, H_{p_N}(m)\}_{m=1}^{M}$, where $\{p_n\}_{n=1}^{N}$ is a set of N prime numbers.

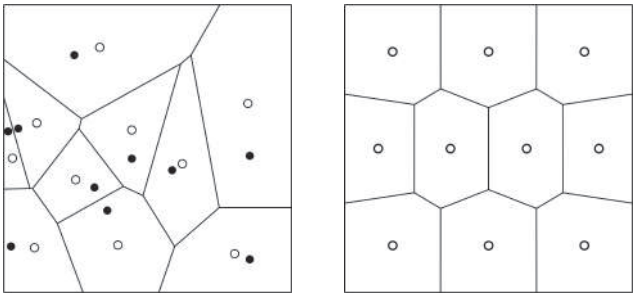
Hammersley sampling. Hammersley sampling is also deterministic, but although it relies on the Halton sequence, it is not itself sequential; thus, it does not fall within the class of QMC methods. Hammersley sampling in the unit hypercube in $\mathbb{R}^N$ proceeds as follows. The first coordinate of the sample points is determined by a uniform partition of the unit interval; the remaining coordinates are determined from an $(N-1)$-dimensional Halton sequence.

Latin hypercube sampling. Many variations of LHS sampling have been developed; here we describe the basic technique. A set of LHS sample points in the unit hypercube in $\mathbb{R}^N$ are determined probabilistically and non-sequentially by the following process. First, the unit cube is divided into M^N cubical bins, that is, into M bins in each of the N coordinate directions. Then, M of the cubical bins are chosen according to N random permutations of $\{1, 2, \ldots, M\}$. Finally, a random point is sampled within each of the



(a) (b)

Figure 3.5.1. $M = 4$ point LHS samples in $N = 2$ dimensions. (a) The cubical bins are determined from the permutations $\{3, 2, 4, 1\}$ and $\{4, 2, 1, 3\}$. (b) The cubical bins are determined from the permutations $\{1, 2, 3, 4\}$ and $\{1, 2, 3, 4\}$.



(a) (b)

Figure 3.5.2. (a) Ten randomly selected points in a square (dots) and the centres of mass (circles) of the corresponding Voronoi regions. (b) A 10-point CVT in a square. The circles are simultaneously the generators of the Voronoi tessellation and the centres of mass of the corresponding Voronoi cells.

M cubical bins so chosen; alternatively, one can simply choose the centre points of those bins. Two sample LHS sample sets are given in Figure 3.5.1.

Centroidal Voronoi tessellations. CVTs are a non-sequential and deterministic sampling technique. A CVT point set has the property that each point is simultaneously the generator of a Voronoi tessellation and the centre of mass of its Voronoi cell. General point sets do not have this property, so CVT point sets have to be constructed. The simplest construction algorithm, known as Lloyd's method, is an iterative method that proceeds as follows. First, select M points in the unit hypercube; for example, they could be any of the other point sets discussed. Then construct the Voronoi tessellation of the unit hypercube corresponding to the selected points. Then, the point set is replaced by the centres of mass of the Voronoi cells. These two steps, that is, Voronoi tessellation construction followed by centre of

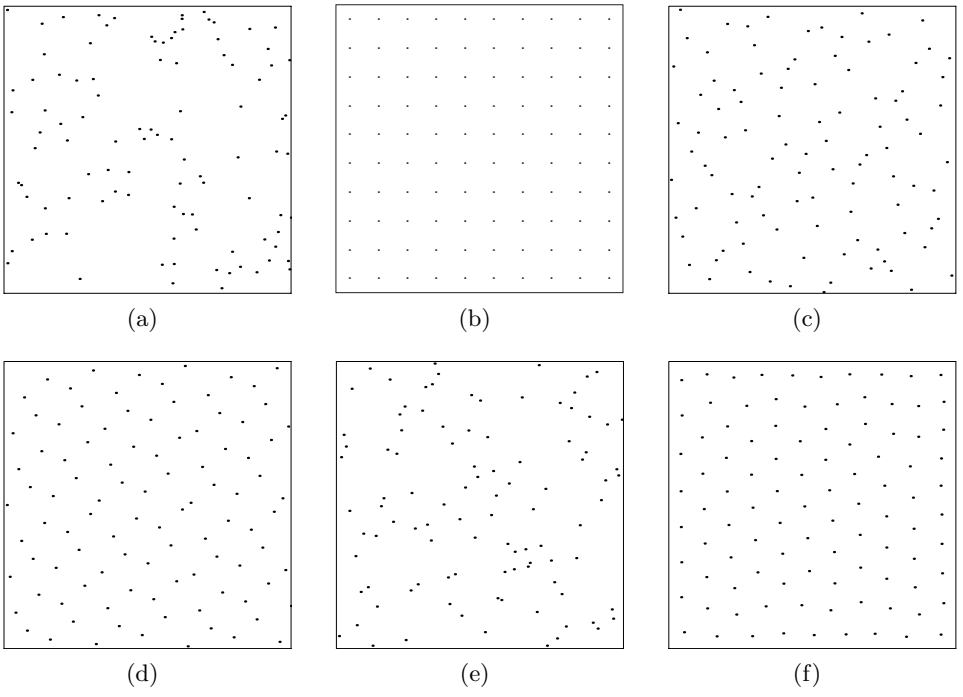


Figure 3.5.3. $M = 100$ point samples. (a) Monte Carlo, (b) tensor product, (c) Halton, (d) Hammersley, (e) Latin hypercube, and (f) centroidal Voronoi tessellation.

mass determination, are repeated until convergence is achieved. Both steps within an iteration of Lloyd's method are very difficult to achieve, even for $N = 3$ or 4. Fortunately, purely probabilistic, embarrassingly parallelizable algorithms have been devised for constructing CVTs. Note that the end product is still deterministic: the probabilistic aspect is restricted to the construction of the CVT. Regardless, CVT point sets are considerably more expensive to determine than most others. However, the cost remains negligible compared to the cost of even a single typical computational solution of a PDE. Figure 3.5.2 first illustrates that general point sets do not possess the CVT property, that is, the generators of the Voronoi cells do not coincide with the centres of mass of the cells. The figure also illustrates the CVT property of CVT point sets.

Figure 3.5.3 illustrates uniform point sets for five sampling methods discussed so far, along with a Cartesian tensor product arrangement. The 'uniform' MCM point set is determined by sampling from the uniform PDF; of course, actual realizations of MCM point sets are far from uniform, as illustrated by Figure 3.5.3(a). The tensor product, Halton, Hammersley, LHS, and CVT point sets are 'uniform' by construction, although the actual uniformity varies between the various choices. Certainly, the tensor

product set is the most 'uniform', but of course, the number of points M is restricted to be an integer power of the parameter dimension N. Note that Hammersley looks more uniform than Halton, which is why this variant of Halton was developed. Visually, the CVT point set is second in 'uniformity', a fact that is confirmed by using quantitative measures of uniformity such as the variance in the spacing between points.

Do any of these point sets improve on the convergence rate of the MCM method for integration? For example, for QMC sequences, it can be shown that the error estimate 'improves' from $1/\sqrt{M}$ to $(\log M)^N/M$. Here, we see another manifestation of the curse of dimensionality. For low-dimensional problems, that is, for N small, one does indeed see an improvement from $1/\sqrt{M}$ to close to $1/M$ convergence. But, for large enough N, the logarithmic term dominates the M term, so that the estimate predicts that the QMC method will lose to MCM in such cases.

3.5.2. Non-uniform sampling in hypercubes

So far we have considered uniformly distributed point sets. For any of the sampling methods, sets of points according to a given joint PDF $\rho(\boldsymbol{y})$ can be constructed from the uniform point sets by appropriate mapping.

For MCM sampling, the sampling itself can incorporate the density function, for instance by using a rejection method. One such method proceeds as follows. We set $\rho_{\max} = \max_{\boldsymbol{y}\in\Gamma} \rho(\boldsymbol{y})$. We then sample a point $\boldsymbol{y} \in \Gamma$ according to the uniform PDF. We also sample a point $\widehat{\boldsymbol{y}} \in [0,1]$ according to the one-dimensional uniform PDF. If $\widehat{\boldsymbol{y}} < \rho(\boldsymbol{y})/\rho_{\max}$, then $\boldsymbol{y}$ is accepted as one of the M desired sample points. Otherwise it is rejected, and we continue the process until we obtain M sample points. The rejection method may also be applied to QMC and Hammersley sampling.

For LHS sampling, the PDF can also be incorporated into the construction algorithm. Suppose the parameters are independent so that the joint PDF

$$\rho(\boldsymbol{y}) = \prod_{n=1}^{N} \rho_n(y_n)$$

for N one-dimensional PDFs $\{\rho_n(y_n)\}_{n=1}^N$. Then, for each coordinate direction $n \in \{0, 1, \ldots, N\}$, we partition the unit interval $[0,1]$ into the subintervals $\cup_{m=1}^M [y_{n,m-1}, y_{n,m}]$, where $0 = y_{n,0} < y_{n,1} < \cdots < y_{n,M-1} < y_{n,M} = 1$. Standard LHS sampling as described in Section 3.5.1 uses uniform partitions of the unit interval. However, one could also choose a partition that respects the PDF. For example, for each $n = 1, \ldots, N$, we should choose the subintervals $\{[y_{n,m-1}, y_{n,m}]\}_{m=1}^M$ so that

$$\int_{y_{n,m-1}}^{y_{n,m}} \rho(y_n)\, \mathrm{d}y_n$$

is independent of n', that is, so the probability that a sample point is in a subinterval is the same for all subintervals.

For CVT sampling, the PDF can be easily incorporated into the construction process. In fact, by definition, the CVT sample points are centres of mass of their Voronoi cells, so that one need only incorporate the PDF into the centre of mass calculation.

3.5.3. Latinization of point sets

By construction and as illustrated in Figure 3.5.1, LHS sample points have the desirable feature that there is exactly one sample point in each subinterval in each of the N coordinate directions. As a result, the projection of the M sample points onto any lower-dimensional face of the hypercube results in M distinct points. Contrast this with the tensor product point set, as illustrated in Figure 3.5.3. Note that for this two-dimensional illustration, projecting the 100 points to any side of the square results in only 10 distinct points. The other point sets illustrated in Figure 3.5.3 fall somewhere in between the LHS and tensor product cases. In fact, QMC and Hammersley sampling were partly devised to prevent the type of serious coalescence that occurs in the tensor product case. CVT point sets, on the other hand, do suffer from the clustering (although not from the total coalescence) of points when projected to lower-dimensional faces. The clustering of point sets when projected onto lower-dimensional faces is considered undesirable because it may result in a loss of accuracy in quadrature rules, for example.

On the other hand, LHS point sets usually have inferior volumetric coverage: see the 'holes' in the coverage of the LHS point set in Figure 3.5.3, especially compared to the CVT point set. Ideally, we would like both, that is, the LHS property and good volumetric coverage. This can be achieved because any point set can be transformed into an LHS by a simple procedure referred to as *Latinization* (Saka *et al.* 2007, Romero *et al.* 2006). Thus, for example, a CVT point set can be transformed into an LHS point set. We now describe the Latinization procedure for arbitrary point sets.

Suppose we are given a point set $P = \{\boldsymbol{y}_m\}_{m=1}^M$ in the unit hypercube. For any $n \in \{1, 2, \ldots, N\}$, we define:

- the *nth reordering* of P to be the point set $\{R_n \boldsymbol{y}_m\}_{m=1}^M$ obtained by reordering P according to the value of the nth coordinates of the $\boldsymbol{y}_m$ (ties can be arbitrarily broken);
- the *nth shift* of P to be the point set $\{S_n \boldsymbol{y}_m\}_{m=1}^M$ such that

$$S_n \boldsymbol{y}_{m,n'} = \begin{cases} \boldsymbol{y}_{m,n'} & \text{if } n' \neq n, \\ \frac{m - U_{n'm}}{M} & \text{if } n' = n, \end{cases} \tag{3.5.1}$$

  where $\boldsymbol{y}_{m,n}$ denotes the nth component of the vector $\boldsymbol{y}_m$ and $U_{n'm}$ denotes a uniform random variable taking values on the unit interval.

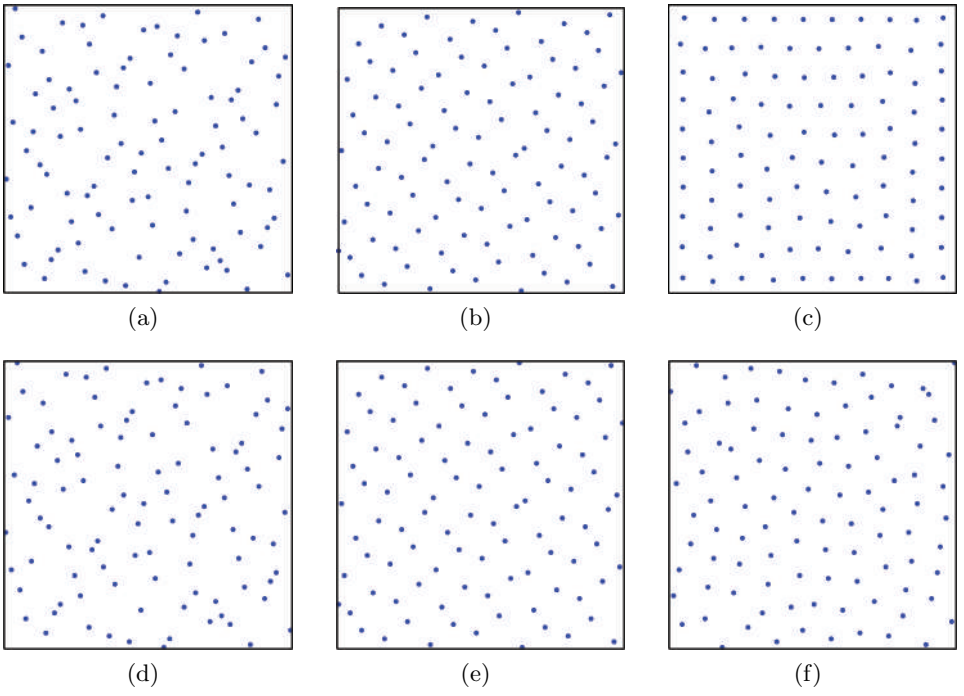


Figure 3.5.4. 100 points in the square. (a) Halton, (b) Hammersley, and (c) centroidal Voronoi sample points. (d,e,f) Latinized versions of the corresponding sample points in (a,b,c).

Then, starting with any point set $\{\boldsymbol{y}_m\}_{m=1}^M$, the corresponding Latinized point set $\{L\boldsymbol{y}_m\}_{m=1}^M$ is given by

$$L\boldsymbol{y}_m = \left(\prod_{n=1}^{N}(S_n R_n)\right)\boldsymbol{y}_m \quad \text{for } m = 1, \ldots, M.$$

By construction, the Latinized point set is an LHS. The nth shift moves the reordered points parallel to the nth axis while preserving the nth coordinate ordering of the points. Latinization is the result of applying the shift to all coordinates.

Illustrations of three Latinized point sets are provided in Figure 3.5.4. We see that Latinization does somewhat harm the 'uniformity' of the point sets. This is the cost of transforming the point sets into Latinized versions. On the other hand, the harm does not appear to be great, especially for the CVT point set.

PART FOUR
Global polynomial stochastic approximation

4.1. Preliminaries

For certain classes of problems, the solution of a partial differential equation (PDE) may have a very smooth dependence on the input random variables, and thus it is reasonable to use a global polynomial approximation in the parameter space $L^2_\rho(\Gamma)$. For example, it is known (Babuška *et al.* 2007*a*, Beck *et al.* 2012, Cohen *et al.* 2011) that the solution of a linear elliptic PDE with diffusivity coefficient and/or forcing term, described as truncated expansions of random fields, depends analytically on the input random variables $y_n \in \Gamma_n$, $n = 1\ldots, N$. In general, throughout this section we make the following assumption concerning the regularity of the solution to (2.3.4).

Assumption 4.1.1. For $n = 1, \ldots, N$, let

$$\Gamma_n^* = \prod_{\substack{j=1 \\ j\neq n}}^{N} \Gamma_j,$$

and let $\boldsymbol{y}_n^*$ denote an arbitrary element of Γ_n^*. Then there exist constants λ and $\tau_n \geq 0$ and regions $\Sigma_n \equiv \{z \in \mathbb{C},\ \ \mathrm{dist}(z, \Gamma_n) \leq \tau_n\}$ in the complex plane for which

$$\max_{\boldsymbol{y}_n^* \in \Gamma_n^*} \max_{z\in\Sigma_n} \big\| u(\cdot, \boldsymbol{y}_n^*, z)\big\|_{W(D)} \leq \lambda,$$

that is, the solution $u(\mathbf{x}, \boldsymbol{y}_n^*, y_n)$ admits an analytic extension $u(\mathbf{x}, \boldsymbol{y}_n^*, z)$, $z \in \Sigma_n \subset \mathbb{C}$.

Example 4.1.2. It has been proved (Babuška *et al.* 2007*a*) that the linear problem (2.1.2) satisfies the analyticity result stated in Assumption 4.1.1. For example, if we take the diffusivity coefficient as the truncated nonlinear expansion

$$a(\mathbf{x}, \omega) \approx a_{\min} + \exp\bigg\{ b_0(\mathbf{x}) + \sum_{n=1}^{N} \sqrt{\lambda_n} b_n(\mathbf{x}) y_n(\omega) \bigg\}, \tag{4.1.1}$$

where $\mathbb{VAR}[y_n] = \lambda_n$ and (λ_n, b_n) are eigenpairs of the covariance operator associated to the random field $a(\mathbf{x}, \omega)$ (see Nobile *et al.* 2008*a* for details), then a suitable analyticity region $\Sigma(\Gamma_n; \tau_n)$ is given by

$$\tau_n = \frac{1}{4\sqrt{\lambda_n}\|b_n\|_{L^\infty(D)}}. \tag{4.1.2}$$

Observe that, because $\sqrt{\lambda_n}\|b_n\|_{L^\infty(D)} \to 0$ for a regular enough covariance function (Frauenfelder *et al.* 2005), the analyticity region increases as n increases. This fact naturally introduces an anisotropic behaviour with respect to y_n, $n = 1, \ldots, N$.

The analytic dependence of the solution with respect to the random input parameters, required by Assumption 4.1.1, has also been verified for the nonlinear elliptic problem (2.1.3) (Webster 2007) and even for the Navier–Stokes equations (Tran, Trenchea and Webster 2012). In such situations, global stochastic Galerkin (SGM) or stochastic collocation (SCM) methods, with the former involving a projection onto an orthogonal basis and the latter involving a multi-dimensional interpolation, feature faster convergence rates than do classical sampling methods.

4.2. Stochastic global polynomial subspaces

We seek to further approximate the semi-discrete spatial finite element approximation $u_{J_h}(\mathbf{x}, \boldsymbol{y})$ given in (2.3.5) by discretizing with respect to $\boldsymbol{y}$ using global polynomials. Motivated by the goal of reducing the curse of dimensionality, here we follow Beck *et al.* (2011) in defining several choices for the multivariate polynomial spaces as alternatives to the standard isotropic tensor product space. Each choice is realized through a particular choice for the basis $\{\psi_m(\boldsymbol{y})\}_{m=1}^M$ used to define the fully discrete approximation $u_{J_h M}(\mathbf{x}, \boldsymbol{y})$ given in (2.4.1) and which is determined by solving (2.4.3).

Let $p \in \mathbb{N}$ denote a single index denoting the polynomial order of the associated approximation and consider a sequence of increasing multi-index sets $\mathcal{J}(p)$ such that $\mathcal{J}(0) = \{(0, \ldots, 0)\}$ and $\mathcal{J}(p) \subseteq \mathcal{J}(p+1)$. Let $\mathcal{P}_{\mathcal{J}(p)}(\Gamma) \subset L^2_\rho(\Gamma)$ denote the multivariate polynomial space over Γ corresponding to the index set $\mathcal{J}(p)$, defined by

$$\mathcal{P}_{\mathcal{J}(p)}(\Gamma) = \operatorname{span}\left\{ \prod_{n=1}^{N} y_n^{p_n} \,\middle|\, \boldsymbol{p} \in \mathcal{J}(p), y_n \in \Gamma_n \right\}. \tag{4.2.1}$$

We set $M_p = \dim\{\mathcal{P}_{\mathcal{J}(p)}\}$. The fully discrete global polynomial approximation is now denoted by $u_{J_h M_p} \in W_h(D) \bigotimes \mathcal{P}_{\mathcal{J}(p)}(\Gamma)$.

Several choices for the index set and the corresponding polynomial spaces are available (Beck *et al.* 2011, Cohen *et al.* 2011, Todor 2005, Frauenfelder *et al.* 2005). The most obvious one is the tensor product (TP) polynomial space, defined by choosing $\mathcal{J}(p) = \{\boldsymbol{p} \in \mathbb{N}^N \mid \max_n p_n \le p\}$. In this case $M_p^{TP} = (p+1)^N$, which results in an explosion in computational effort for higher dimensions. For the same value of p, the same nominal rate of convergence is achieved at a substantially lower costs by the total degree (TD) polynomial spaces, for which

$$\mathcal{J}(p) = \left\{ \boldsymbol{p} \in \mathbb{N}^N \,\middle|\, \sum_{n=1}^{N} p_n \le p \right\},$$

and $M_p^{TD} = (N+p)!/(N!\,p!)$. Table 4.2.1 provides a comparison of the TP and TD choices and also provides stark evidence of the curse of dimensionality.

Table 4.2.1. A comparison of the $M_p = \dim\{\mathcal{P}_{\mathcal{J}(p)}\}$ degrees of freedom for the total degree (TD) and tensor product (TP) polynomial spaces, where $N = \dim(\Gamma)$ is the number of random variables and p is the maximal degree of polynomials.

N	p	M_p using TD basis	M_p using TP basis
3	3	20	64
	5	56	216
5	3	56	1 024
	5	252	7 776
10	3	286	1 048 576
	5	3 003	60 046 176
20	3	1 771	$> 1 \times 10^{12}$
	5	53 130	$> 3 \times 10^{15}$
100	3	176 851	$> 1 \times 10^{60}$
	5	96 560 646	$> 6 \times 10^{77}$

Other subspaces having dimension smaller than the TP subspace include hyperbolic cross (HC) polynomial spaces, for which

$$\mathcal{J}(p) = \left\{ \boldsymbol{p} \in \mathbb{N}^N \;\middle|\; \sum_{n=1}^{N} \log_2(p_n + 1) \leq \log_2(p+1) \right\},$$

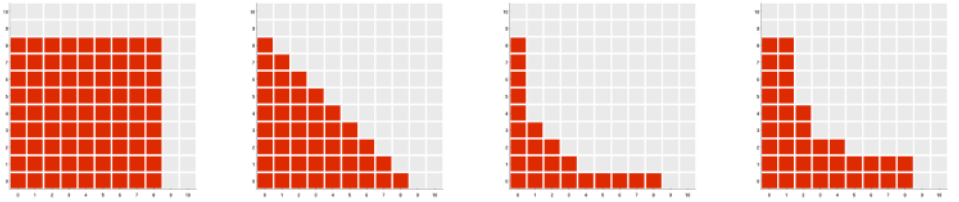
and sparse Smolyak (SS) polynomial spaces, for which

$$\mathcal{J}(p) = \left\{ \boldsymbol{p} \in \mathbb{N}^N \;\middle|\; \sum_{n=1}^{N} \gamma(p_n) \leq \gamma(p) \right\} \tag{4.2.2}$$

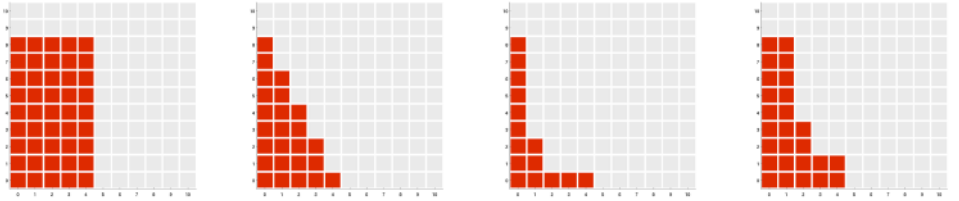
$$\text{with } \gamma(p) = \begin{cases} 0 & \text{for } p = 0, \\ 1 & \text{for } p = 1, \\ \lceil \log_2(p) \rceil & \text{for } p \geq 2. \end{cases}$$

SS polynomial spaces are not typically used in SGMs but are the natural choice in sparse grid SCM methods, as first described in Smolyak (1963) and used in Nobile *et al.* (2008*a*, 2008*b*). Finally, as illustrated by Example 4.1.2, it is often the case that the stochastic input data exhibits anisotropic behaviour with respect to the 'directions' y_n, $n = 1, \ldots, N$. To exploit this effect, it is necessary to approximate $u_{J_h M_p}$ in an anisotropic polynomial space. Following Nobile *et al.* (2008*b*), we introduce a vector

$$\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_N) \in \mathbb{R}_+^N$$



(a)



(b)

Figure 4.2.1. For a finite-dimensional Γ with $N = 2$ and a fixed polynomial index $p = 8$, we plot (a) the indices $(p_1, p_2) \in \mathcal{J}(8)$ corresponding to the isotropic TP, TD, HC, and SS polynomial spaces, and (b) the indices $(p_1, p_2) \in \mathcal{J}_{\boldsymbol{\alpha}}(8)$ with $\alpha_1/\alpha_2 = 2$ corresponding to the anisotropic TP, TD, HC, and SS polynomial spaces.

of positive weights and define $\alpha_{\min} = \min_{n=1,\ldots,N} \alpha_n$. The anisotropic versions $\mathcal{J}_{\boldsymbol{\alpha}}(p)$ of the aforementioned polynomial spaces are described in Beck *et al.* (2011). Here, we consider the anisotropic SS polynomial space given by

$$\mathcal{J}_{\boldsymbol{\alpha}}(p) = \left\{ \boldsymbol{p} \in \mathbb{N}^N \,\middle|\, \sum_{n=1}^{N} \alpha_n \, \gamma(p_n) \leq \alpha_{\min} \, \gamma(p) \right\}.$$

For $N = 2$ dimensions, Figure 4.2.1 provides examples of both isotropic and anisotropic TP, TD, HC, and SS polynomial spaces, where we chose $p = 8$ and $\boldsymbol{\alpha} = (2, 1)$.

In Sections 4.3 and 4.4 we provide the generalized construction of global SGMs and the global SCMs, respectively, of the approximation $u_{J_h M_p}$. In Section 4.5 we use an example to compare the computational complexity of the two approaches.

4.3. Global stochastic Galerkin methods

In this section we focus on the fully discrete approximation (2.4.3) in the subspace $W_h(D) \bigotimes \mathcal{P}_{\mathcal{J}(p)}(\Gamma)$, where both spatial and stochastic approximation are effected by a Galerkin method. We use standard locally supported piecewise polynomial finite element bases in $W_h(D)$ but, taking advantage of Assumption 4.1.1, we use global (orthonormal) polynomial bases in $\mathcal{P}_{\mathcal{J}(p)}(\Gamma)$. As such, it is entirely natural to call these approaches global

Table 4.3.1. Relationship between standard continuous probability density functions and the Askey scheme of continuous hyper-geometric polynomials.

Distribution	PDF	Polynomial family	Support
normal	$\frac{1}{\sqrt{2\pi}}\mathrm{e}^{\frac{-y^2}{2}}$	Hermite $H_{p,n}(y)$	$[-\infty,\infty]$
uniform	$\frac{1}{2}$	Legendre $P_{p,n}(y)$	$[-1,1]$
beta	$\frac{(1-y)^\alpha(1+y)^\beta}{2^{\alpha+\beta+1}B(\alpha+1,\beta+1)}$	Jacobi $P_{p,n}^{(\alpha,\beta)}(y)$	$[-1,1]$
exponential	e^{-y}	Laguerre $L_{p,n}(y)$	$[0,\infty]$
gamma	$\frac{y^\alpha \mathrm{e}^{-y}}{\Gamma(\alpha+1)}$	generalized Laguerre $L_{p,n}^{(\alpha)}(y)$	$[0,\infty]$

stochastic Galerkin finite element methods, which we abbreviate to *global stochastic Galerkin methods* (gSGMs), and to treat the solution as a function of $d+N$ variables, that is, of the d spatial variables and N random parameters. Generally, these techniques are *intrusive* in the sense that they are non-ensemble-based methods, requiring the solution of a discrete system that couples all spatial and probabilistic degrees of freedom.

Specifically, let

$$\{\psi_{p,n}(y_n)\}_{p=0}^{\infty} \subset L^2_{\rho_n}(\Gamma_n)$$

denote a set of $L^2_{\rho_n}$-orthonormal polynomials in Γ_n. That is, recalling that $\rho(\boldsymbol{y}) = \prod_{n=1}^N \rho_n(y_n)$, we have, for all $n = 1,\dots,N$ and $p, p' = 0, 1, 2, \dots,$

$$\int_{\Gamma_n} \psi_{p,n}(y_n)\psi_{p',n}(y_n)\rho_n(y_n)\,\mathrm{d}y_n = \delta_{pp'}.$$

Table 4.3.1 lists examples of polynomial families that provide orthonormal bases with respect to several continuous PDFs. The listed families are derived from the family of hypergeometric orthonormal polynomials known as the Askey scheme (Askey and Wilson 1985), of which the Hermite polynomials employed by Wiener (1938) are one member.

Multivariate $L^2_\rho(\Gamma)$-orthonormal polynomials are defined as tensor products of the univariate polynomials with $\boldsymbol{p} \in \mathcal{J}(p)$, that is,

$$\psi_{\boldsymbol{p}}(\boldsymbol{y}) = \prod_{n=1}^{N} \psi_{p_n,n}(y_n).$$

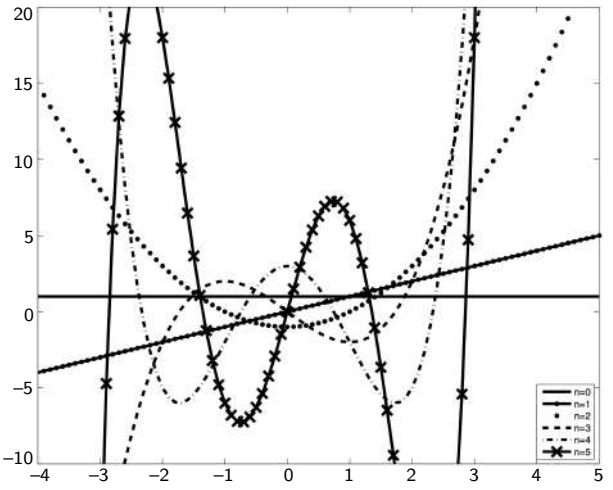


Figure 4.3.1. For $(N = 1)$-dimensional Γ and $-4 \leq y \leq 5$, we plot the first six Hermite polynomials $H_0(y) = 1$, $H_1(y) = y$, $H_2(y) = y^2 - 1$, $H_3(y) = y^3 - 3y$, $H_4(y) = y^4 - 6y^2 + 3$, and $H_5(y) = y^5 - 10y^3 + 15y$.

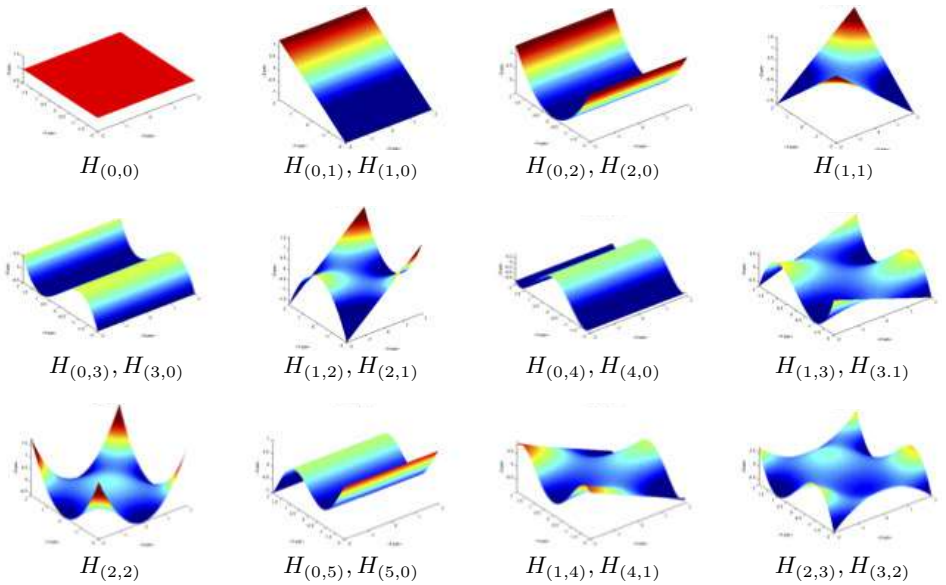


Figure 4.3.2. The two-dimensional Hermite polynomials $H_{\boldsymbol{p}}(\boldsymbol{y})$ such that $p_1 + p_2 \leq 5$.

One- and two-dimensional Hermite polynomials are shown in Figures 4.3.1 and 4.3.2, respectively.

Having chosen the bases $\{\phi_j(\mathbf{x})\}_{j=1}^{J_h} \in W_h(D)$ and

$$\{\psi_{\boldsymbol{p}(\boldsymbol{y})}\}_{\boldsymbol{p}\in\mathcal{J}(p)} \in \mathcal{P}_{\mathcal{J}(p)}(\Gamma),$$

the gSGM approximation is defined by

$$u^{gSG}_{J_h M_p}(\mathbf{x},\boldsymbol{y}) = \sum_{\boldsymbol{p}\in\mathcal{J}(p)} u_{\boldsymbol{p}}(\mathbf{x})\psi_{\boldsymbol{p}}(\boldsymbol{y}) = \sum_{\boldsymbol{p}\in\mathcal{J}(p)}\sum_{j=1}^{J_h} u_{\boldsymbol{p},j}\phi_j(\mathbf{x})\psi_{\boldsymbol{p}}(\boldsymbol{y}), \qquad (4.3.1)$$

where, after a reordering of the index set $\mathcal{J}(p)$, is of the form (2.4.1) with $M = M_p$. Solving for the coefficients $\{u_{\boldsymbol{p},j}\}$, $\boldsymbol{p} \in \mathcal{J}(p), j = 1,\dots,J_h$, requires the substitution of the approximation (4.3.1) into the weak formulation (2.4.3), resulting in a (possibly nonlinear) coupled system of size $J_h M_p \times J_h M_p$. Given that $\{\psi_{\boldsymbol{p}}\}_{\boldsymbol{p}\in\mathcal{J}(p)}$ is an orthonormal basis, it is easy to show that the first two moments are given by

$$\mathbb{E}\big[u^{gSG}_{J_h M_p}\big](\mathbf{x}) = u_{\mathbf{0}}(\mathbf{x}) \quad \text{and}$$
$$\mathbb{VAR}\big[u^{gSG}_{J_h M_p}\big](\mathbf{x}) = \sum_{\boldsymbol{p}\in\mathcal{J}(p)} u^2_{\boldsymbol{p}}(\mathbf{x}) - \mathbb{E}\big[u^{gSG}_{J_h M_p}\big]^2(\mathbf{x}).$$

Example 4.3.1. This example provides a detailed description of the construction of a gSGM for the linear elliptic problem (2.1.2) with $f(\mathbf{x},\omega) = f(\mathbf{x})$ and $a(\mathbf{x},\omega) = a(\mathbf{x},\boldsymbol{y}(\omega))$ for $\boldsymbol{y} \in \Gamma$. For $W_h \subset H^1_0(D)$, the general semi-discrete approximation (2.3.5) is given as follows. Find $u_{J_h} \in W_h(D) \bigotimes L^2_\rho(\Gamma)$ such that

$$\int_D a(\mathbf{x},\boldsymbol{y})\nabla u_{J_h}(\mathbf{x},\boldsymbol{y})\cdot\nabla v_{J_h}(\mathbf{x})\,\mathrm{d}\mathbf{x} \qquad (4.3.2)$$
$$= \int_D f(\mathbf{x}) v_{J_h}(\mathbf{x})\,\mathrm{d}\mathbf{x} \quad \rho\text{-a.e. in } \Gamma$$

for all $v_{J_h} \in W_h(D)$. The solution $u_{J_h}(\mathbf{x},\boldsymbol{y})$ of (4.3.2) satisfies Assumption 4.1.1 and is uniquely defined for almost every $\boldsymbol{y} \in \Gamma$.

Let $\{\phi_j\}_{j=1}^{J_h}$ denote a finite element basis for $W_h(D)$ such that $\phi_j(\mathbf{x}_{j'}) = \delta_{jj'}$ for all $j = 1,\dots,J_h$, where $\{\mathbf{x}_j\}_{j=1}^{J_h}$ denotes the grid nodes, and consider the semi-discrete approximation given by $u_{J_h}(\mathbf{x},\boldsymbol{y}) = \sum_{j=1}^{J_h} u_j(\boldsymbol{y})\phi_j(\mathbf{x})$. For any $\boldsymbol{y} \in \Gamma$, let $\boldsymbol{u}(\boldsymbol{y}) = [u_1(\boldsymbol{y}), u_2(\boldsymbol{y}),\dots,u_{J_h}(\boldsymbol{y})]^T$ be the vector of nodal values of $u_{J_h}(\mathbf{x},\boldsymbol{y})$. Then the semi-discrete problem (4.3.2) can be written algebraically as

$$\boldsymbol{A}(\boldsymbol{y})\boldsymbol{u}(\boldsymbol{y}) = \boldsymbol{f} \quad \rho\text{-a.e. in } \Gamma,$$

where $\boldsymbol{f}_j = \int_D f(\mathbf{x})\phi_j(\mathbf{x})\,\mathrm{d}\mathbf{x}$ for $j = 1, \ldots, J_h$ and, for $j, j' = 1, \ldots, J_h$,

$$\boldsymbol{A}_{j,j'}(\boldsymbol{y}) = \int_D a(\mathbf{x}, \boldsymbol{y})\ \nabla\phi_j(\mathbf{x}) \cdot \nabla\phi_{j'}(\mathbf{x})\,\mathrm{d}\mathbf{x}. \tag{4.3.3}$$

The fully discrete approximation of (2.1.2) directly follows from (2.4.2), as follows. Find $u^{gSG}_{J_h M_p} \in W_h(D) \bigotimes \mathcal{P}_{\mathcal{J}(p)}(\Gamma)$ such that

$$\begin{aligned}\int_\Gamma \int_D a(\mathbf{x}, \boldsymbol{y})\, \nabla u^{gSG}_{J_h M_p}(\mathbf{x}, \boldsymbol{y}) \cdot \nabla v_{J_h M_p}(\mathbf{x}, \boldsymbol{y}) \rho(\boldsymbol{y})\,\mathrm{d}\mathbf{x}\,\mathrm{d}\boldsymbol{y} \\ = \int_\Gamma \int_D v_{J_h M_p}(\mathbf{x}, \boldsymbol{y}) f(\mathbf{x}) \rho(\boldsymbol{y})\,\mathrm{d}\mathbf{x}\,\mathrm{d}\boldsymbol{y}\end{aligned} \tag{4.3.4}$$

for all $v_{J_h M_p} \in W_h(D) \bigotimes \mathcal{P}_{\mathcal{J}(p)}(\Gamma)$. Let $\boldsymbol{u_p} = [u_{\boldsymbol{p},1}, \ldots, u_{\boldsymbol{p},J_h}]$ denote the vector of nodal values of the finite element solution corresponding to the $\boldsymbol{p}$th stochastic mode. Then, substituting (4.3.1) into (4.3.4), *i.e.*, performing a Galerkin projection onto the span of $\{\psi_{\boldsymbol{p}}\}_{\boldsymbol{p}\in\mathcal{J}(p)}$, yields the following linear algebraic system: for all $\boldsymbol{p} \in \mathcal{J}(p)$,

$$\sum_{\boldsymbol{p}'\in\mathcal{J}(p)} \underbrace{\left(\int_\Gamma \boldsymbol{A}(\boldsymbol{y})\psi_{\boldsymbol{p}}(\boldsymbol{y})\psi_{\boldsymbol{p}'}(\boldsymbol{y})\rho(\boldsymbol{y})\,\mathrm{d}\boldsymbol{y}\right)}_{\boldsymbol{K}_{\boldsymbol{p},\boldsymbol{p}'}} \boldsymbol{u}_{\boldsymbol{p}'} = \underbrace{\int_\Gamma \boldsymbol{f}\psi_{\boldsymbol{p}}(\boldsymbol{y})\rho(\boldsymbol{y})\,\mathrm{d}\boldsymbol{y}}_{\boldsymbol{F}_{\boldsymbol{p}}}. \tag{4.3.5}$$

Note that coefficient matrix $\boldsymbol{K}$ of the system (4.3.5) consists of $(M_p)^2$ block matrices, each of size $J_h \times J_h$, that is, the size of $\boldsymbol{A}(\boldsymbol{y})$. In some cases, such as when $a(\mathbf{x}, \boldsymbol{y})$ can be represented as a linear function of the random variables y_n, $n = 1, \ldots, N$, the matrix $\boldsymbol{K}$ can have an extremely sparse block structure. However, in other cases for which $a(\mathbf{x}, \boldsymbol{y})$ is a nonlinear function of the random variables, for example for lognormal random fields, $\boldsymbol{K}$ is extremely dense. As such, the preconditioning of the system (4.3.5) is a very active area of research (Desceliers, Ghanem and Soize 2005, Eiermann, Ernst and Ullmann 2007, Elman, Ernst and O'Leary 2001, Elman *et al.* 2011, Ernst, Powell, Silvester and Ullmann 2009, Ernst and Ullmann 2010, Ghanem and Kruger 1996, Ghanem and Spanos 1991, Gordon and Powell 2012, Jin, Cai and Li 2007, Parks *et al.* 2006, Pellissetti and Ghanem 2000, Powell and Elman 2009, Powell and Ullmann 2010, Simoncini and Szyld 2007, Ullmann 2010, Ullmann, Elman and Ernst 2012).

Even in the case of a sparse gSGM matrix $\boldsymbol{K}$, it is impractical to form and store the matrix explicitly. Typically, matrix-free methods are applied to solve the linear system without ever having to store $\boldsymbol{K}$ in memory, as described in Pellissetti and Ghanem (2000). Depending on the form of the coefficient $a(\mathbf{x}, \boldsymbol{y})$, certain choices can be made to reduce this complexity by decoupling the stochastic and spatial components, by writing $\boldsymbol{K}$ as a series of random variables multiplied by several deterministic stiffness matrices. Even so, this approach requires us to rewrite the Galerkin solver for each

new choice of $a(\mathbf{x}, \boldsymbol{y})$. A more convenient and robust choice is to perform an 'offline' projection of $a(\mathbf{x}, \boldsymbol{y})$ onto $\text{span}\{\psi_{\boldsymbol{p}}(\boldsymbol{y})\}_{\boldsymbol{p}\in\mathcal{J}(p)}$, and then exploit the three-term relation of orthonormal polynomials (Ghanem and Spanos 1991, Gautschi 2004) when constructing $\boldsymbol{K}$. This approach can be used regardless of the form of the stochastic coefficient and is used to compare the computational complexity of gSGMs with the methods discussed in Section 4.5.

4.4. Global stochastic collocation methods

Similar to Section 4.3, we again focus our attention on the construction of the fully discrete approximation (2.4.3) in the subspace $W_h(D) \bigotimes \mathcal{P}_{\mathcal{J}(p)}(\Gamma)$. However, rather than making use of a Galerkin projection in both the deterministic and stochastic domains, in this section we instead collocate the semi-discrete approximation $u_{J_h}(\cdot, \boldsymbol{y})$ given in (2.3.5) on an appropriate set of points $\{\boldsymbol{y}_m\}_{m=1}^M \in \Gamma$ to determine M solutions $\{u_{J_h}(\cdot, \boldsymbol{y}_m)\}_{m=1}^M \in \Gamma$. One can then use these solutions to construct a global, possibly interpolatory, polynomial to define the fully discrete approximation $u_{J_h M}^{gSC}(\mathbf{x}, \boldsymbol{y})$. We refer to this process as global stochastic collocation finite element methods, or in short, as global stochastic collocation methods (gSCMs).

Clearly, gSCMs are *non-intrusive* in that the solution of (2.4.3) naturally decouples into a series of M deterministic solves, each of size $J_h \times J_h$. In this sense, gSCMs are another example of stochastic sampling methods.

In Sections 4.4.1 and 4.4.3 we describe the construction of two gSCMs, one based on Lagrange interpolation and the other on non-intrusive projections onto an orthonormal basis.

4.4.1. Global Lagrange interpolation in the parameter space

Interpolatory approximations in the parameter space start with the selection of a set of distinct points $\{\boldsymbol{y}_m\}_{m=1}^M \in \Gamma$ and a set of basis functions[4] $\{\psi_m(\boldsymbol{y})\}_{m=1}^M \in \mathcal{P}_{\mathcal{J}(p)}(\Gamma)$. Then, we seek an approximation

$$u_{J_h M}^{gSC} \in W_h(D) \bigotimes \mathcal{P}_{\mathcal{J}(p)}(\Gamma)$$

of the form

$$u_{J_h M}^{gSC}(\mathbf{x}, \boldsymbol{y}) = \sum_{m=1}^{M} c_m(\mathbf{x}) \psi_m(\boldsymbol{y}). \tag{4.4.1}$$

[4] In general, the number of points and number of basis functions do not have to be the same, *e.g.*, for Hermite interpolation. However, because here we only consider *Lagrange interpolation*, we let M denote both the number of points and the cardinality of the basis.

The Lagrange interpolant is defined by taking M realizations $u_{J_h}(\mathbf{x}, \boldsymbol{y}_m)$ of the finite element approximation of the solution $u(\mathbf{x}, \boldsymbol{y}_m)$ of (2.1.1), that is, one solves the finite element approximation for each of the interpolation points in the set $\{\boldsymbol{y}_m\}_{m=1}^M$. Then, the coefficient functions $\{c_m(\mathbf{x})\}_{m=1}^M$ are determined by imposing the interpolation conditions

$$\sum_{m=1}^{M} c_m(\mathbf{x})\psi_m(\boldsymbol{y}_{m'}) = u_{J_h}(\mathbf{x}, \boldsymbol{y}_{m'}) \quad \text{for } m' = 1, \ldots, M. \tag{4.4.2}$$

Thus, each of the coefficient functions $\{c_m(\mathbf{x})\}_{m=1}^M$ is a linear combination of the finite element data $\{u_{M_h}(\mathbf{x}, \boldsymbol{y}_m)\}_{m=1}^M$; the specific linear combinations are determined in the usual manner from the entries of the inverse of the $M \times M$ interpolation matrix $\boldsymbol{L}$ having entries $L_{m',m} = \psi_m(\boldsymbol{y}_{m'})$, $m, m' = 1, \ldots, M$. The sparsity and conditioning of $\boldsymbol{L}$ heavily depend on the choice of basis; that choice could result in matrices that range from fully dense to diagonal and from highly ill-conditioned to perfectly well-conditioned.

The main attraction of interpolatory approximations of parameter dependences is that it effects a complete decoupling of the spatial and probabilistic degrees of freedom. Clearly, once the interpolation points $\{\boldsymbol{y}_m\}_{m=1}^M$ are chosen, we can solve M deterministic finite element problems, one for each parameter point $\boldsymbol{y}_m$, with total disregard to what basis $\{\psi_m(\boldsymbol{y})\}_{m=1}^M$ we choose to use. Then, the coefficients $\{c_m(\mathbf{x})\}_{m=1}^M$ defining the approximation (4.4.1) are found from the interpolation conditions in (4.4.2); it is only in this last step that the choice of stochastic basis enters into the picture. Note that this decoupling property makes the implementation of Lagrange interpolatory approximations of parameter dependences almost as trivial as it is for Monte Carlo sampling. However, if that dependence is smooth, as described by Assumption 4.1.1, because of the higher accuracy of global polynomial approximations in the space $\mathcal{P}_{\mathcal{J}(p)}(\Gamma)$, interpolatory approximations require substantially fewer sampling points to achieve a desired error tolerance.

Given a set of interpolation points, to complete the set-up of a Lagrange interpolation problem, one has to choose a basis. The simplest and most popular choice is to use Lagrange fundamental polynomials, that is, polynomials that possess a *delta property* $\psi_{m'}(\boldsymbol{y}_m) = \delta_{m'm}$, where $\delta_{m'm}$ denotes the Kronecker delta. In this case, the interpolating conditions (4.4.2) reduce to $c_m(\mathbf{x}) = u_{J_h}(\mathbf{x}, \boldsymbol{y}_m)$ for $m = 1, \ldots, M$, that is, the interpolation matrix $\boldsymbol{L}$ is simply the $M \times M$ identity matrix. In this sense, the use of Lagrange polynomial bases can be viewed as resulting in pure sampling methods, much the same as Monte Carlo methods, but instead of randomly sampling in the parameter space Γ, the sample points are deterministically structured. Mathematically, using the Lagrange fundamental polynomial basis $\{\psi_m\}_{m=1}^M$, this ensemble-based approach results in the fully discrete

approximation of the solution $u(\mathbf{x}, \boldsymbol{y})$ of the PDE given by

$$u^{gSC}_{J_h M}(\mathbf{x}, \boldsymbol{y}) = \sum_{m=1}^{M} u_{J_h}(\mathbf{x}, \boldsymbol{y}_m)\psi_m(\boldsymbol{y}). \tag{4.4.3}$$

We note that the construction of multivariate Lagrange fundamental polynomials for a general set of interpolations points is not an easy matter. Fortunately, there exist means for so doing: see Sauer and Xu (1995).

4.4.2. Generalized sparse grid construction

We follow Beck *et al.* (2011) and Nobile *et al.* (2008*a*, 2008*b*) to describe a generalized version of the Smolyak sparse grid gSCM first described in Smolyak (1963) for interpolation and quadrature. For each $n = 1, \ldots, N$, let $l_n \in \mathbb{N}_+$ denote the one-dimensional level of approximation and let $\{y^{(l_n)}_{n,k}\}_{k=1}^{m(l_n)} \subset \Gamma_n$ denote a sequence of one-dimensional interpolation points in Γ_n. Here, $m(l) : \mathbb{N}_+ \to \mathbb{N}_+$ is such that $m(0) = 0$, $m(1) = 1$, and $m(l) < m(l+1)$ for $l = 2, 3, \ldots$, so that $m(l)$ strictly increases with l; $m(l_n)$ defines the the total number of collocation points at level l_n. For a univariate function $v \in C^0(\Gamma_n)$, we introduce, for $n = 1, \ldots, N$, a sequence of one-dimensional Lagrange interpolation operators $\mathscr{U}_n^{m(l_n)} : C^0(\Gamma_n) \to \mathcal{P}_{m(l_n)-1}(\Gamma_n)$ given by

$$\mathscr{U}_n^{m(l_n)}[v](y_n) = \sum_{k=1}^{m(l_n)} v\big(y^{(l_n)}_{n,k}\big)\psi^{(l_n)}_{n,k}(y_n) \quad \text{for } l_n = 1, 2, \ldots, \tag{4.4.4}$$

where $\psi^{(l_n)}_{n,k} \in \mathcal{P}_{m(l_n)-1}(\Gamma_n)$, $k = 1, \ldots, m(l_n)$, are Lagrange fundamental polynomials of degree $p_{l_n} = m(l_n) - 1$ such that

$$\psi^{(l_n)}_{n,k}(y_n) = \prod_{\substack{k'=1 \\ k' \neq k}}^{m(l_n)} \frac{\big(y_n - y^{(l_n)}_{n,k'}\big)}{\big(y^{(l_n)}_{n,k} - y^{(l_n)}_{n,k'}\big)}.$$

Using the convention that $\mathscr{U}_n^{m_0} = 0$, we introduce the difference operator given by

$$\Delta_n^{m(l_n)} = \mathscr{U}_n^{m(l_n)} - \mathscr{U}_n^{m_{l_n-1}}. \tag{4.4.5}$$

For the multivariate case, we let $\boldsymbol{l} = (l_1, \ldots, l_N) \in \mathbb{N}_+^N$ denote a multi-index and let $L \in \mathbb{N}_+$ denote the total level of the sparse grid approximation. Then, for each $n = 1, \ldots, N$, we exploit the operator (4.4.5) to form the *N-dimensional hierarchical surplus operator* defined by

$$\Delta^m = \bigotimes_{n=1}^{N} \Delta_n^{m(l_n)} \tag{4.4.6}$$

and, from (4.4.5) and (4.4.6), the *Lth level generalized sparse grid operator* given by

$$\mathscr{I}_L^{m,g} = \sum_{g(\boldsymbol{l})\le L} \bigotimes_{n=1}^{N} \Delta_n^{m(l_n)}, \tag{4.4.7}$$

where $g : \mathbb{N}_+^N \to \mathbb{N}$ is another strictly increasing function that defines the mapping between the multi-index $\boldsymbol{l}$ and the level L used to construct the sparse grid. Finally, given the functions m and g and a level L, we can construct the generalized sparse grid approximation of u_{J_h} as

$$\begin{aligned} u_{J_h M_L}^{gSC} &= \mathscr{I}_L^{m,g}[u_{J_h}] \\ &= \sum_{L-N+1\le g(\boldsymbol{l})\le L} \sum_{\substack{\boldsymbol{k}\in\{0,1\}^N \\ g(\boldsymbol{l}+\boldsymbol{k})\le L}} (-1)^{|\boldsymbol{k}|} \bigotimes_{n=1}^{N} \mathscr{U}_n^{m(l_n)}[u_{J_h}]. \end{aligned} \tag{4.4.8}$$

The fully discrete gSCM (4.4.8) requires the independent evaluation of the finite element approximation $u_{J_h}(\mathbf{x}, \boldsymbol{y})$ on a deterministic set of *distinct collocation points* given by

$$\mathscr{H}_L^{m,g} = \bigcup_{g(\boldsymbol{l})\le L} \bigotimes_{n=1}^{N} \{y_{n,k}^{(l_n)}\}_{k=1}^{m(l_n)}$$

having cardinality M_L, that is, we have $M = M_L$ in (2.4.3). Moreover, the construction of the sparse grid approximation naturally enables the evaluation of moments through simple sparse grid quadrature formulas, for example,

$$\mathbb{E}\big[u_{J_h M_L}^{gSC}\big](\mathbf{x}) = \sum_{m=1}^{M_L} u_{J_h}(\mathbf{x}, \boldsymbol{y}_m) \underbrace{\int_\Gamma \psi_m(\boldsymbol{y})\rho(\boldsymbol{y})\,\mathrm{d}\boldsymbol{y}}_{\text{precomputed weights}} = \sum_{m=1}^{M_L} u_{J_h}(\mathbf{x}, \boldsymbol{y}_m) w_m$$

and

$$\mathbb{VAR}\big[u_{J_h M_L}^{gSC}\big](\mathbf{x}) = \sum_{m}^{M_L} \widetilde{w}_m u_{J_h}^2(\mathbf{x}, \boldsymbol{y}_m) - \mathbb{E}\big[u_{J_h M_L}^{SC}\big]^2(\mathbf{x}),$$

where $\widetilde{w}_m = \mathbb{VAR}[\psi_m(\boldsymbol{y})]$, $m = 1, \ldots, M_L$.

To compare the gSCM based on the generalized sparse grid construction with the gSGM approximation (4.3.1), Beck *et al.* (2011) constructed the underlying polynomial space associated with the approximation (4.4.8) for particular choices of m, g, and L. Let $m^{-1}(k) = \min\{l \in \mathbb{N}_+ \mid m(l) \ge k\}$ denote the left inverse of m such that $m^{-1}\big(m(l)\big) = l$ and $m\big(m^{-1}(k)\big) \ge k$. Then, let $\boldsymbol{s}(\boldsymbol{l}) = \big(m(l_1), \ldots, m(l_N)\big)$ and define the polynomial index set

$$\mathcal{J}^{m,g}(L) = \big\{\boldsymbol{p} \in \mathbb{N}^N : g\big(\boldsymbol{m}^{-1}(\boldsymbol{p}+\mathbf{1})\big) \le L\big\}.$$

With this definition in hand, we recall the following proposition, whose proof

can be found in Beck *et al.* (2011, Proposition 1), which characterizes the underlying polynomial space of the sparse grid approximation $\mathscr{I}_L^{m,g}[u_{J_h}]$.

Proposition 4.4.1. Let $m : \mathbb{N}_+ \to \mathbb{N}_+$ and $g(\boldsymbol{l}) : \mathbb{N}_+^N \to \mathbb{N}$ denote strictly increasing functions, as described above, and let

$$\{y_{n,k}^{(l)}\}_{k=1}^{m(l)} \subset \Gamma_n$$

denote arbitrary distinct points used in (4.4.4) to determine $\mathscr{U}_n^{m(l)}$, $l = 1, 2, \ldots$. Then,

(1) for any function $v \in C^0(\Gamma)$, the approximation $\mathscr{I}_L^{m,g}[v] \in \mathcal{P}_{\mathcal{J}^{m,g}(L)}(\Gamma)$;
(2) for all $v \in \mathcal{P}_{\mathcal{J}^{m,g}(p)}(\Gamma)$, we have $\mathscr{I}_L^{m,g}[v] = v$.

With Proposition 4.4.1 in hand, we are in position to relate the sparse grid approximation $\mathscr{I}_L^{m,g}$ with the corresponding polynomial subspaces defined in Section 4.2, that is, $\mathcal{P}_{\mathcal{J}^{m,g}(L)}(\Gamma)$ with $m(l) = l$ and

$$\begin{aligned} g(\boldsymbol{l}) &= \max_{n=1,\ldots,N}(l_n - 1) \leq L, \\ g(\boldsymbol{l}) &= \sum_{n=1}^{N}(l_n - 1) \leq L, \\ g(\boldsymbol{l}) &= \sum_{n=1}^{N}\log_2(l_n) \leq \log_2(L+1) \end{aligned}$$

for the tensor product, total degree, and hyperbolic cross polynomial subspaces, respectively. However, the most widely used polynomial subspace is the sparse Smolyak given by (4.2.2), which, in the context of the sparse grid approximation, is defined by

$$m(1) = 1, \quad m(l) = 2^{l-1} + 1, \quad \text{and} \quad g(\boldsymbol{l}) = \sum_{n=1}^{N}(l_n - 1). \tag{4.4.9}$$

Moreover, similar to the anisotropic polynomial spaces described in Section 4.2, the generalized gSCM enables anisotropic refinement with respect to the direction y_n by incorporating a weight vector $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_N) \in \mathbb{R}_+^N$ into the mapping $g : \mathbb{N}_+^N \to \mathbb{N}$, for example, $g(\boldsymbol{l}) = \sum_{n=1}^{N} \alpha_n(l_n - 1) \leq \alpha_{\min} L$ in (4.4.9). Anisotropic refinement will be discussed further in the sections that follow but first, we describe two choices of points used for (4.4.8), namely the Clenshaw–Curtis and Gauss points. See Trefethen (2008) for an insightful comparison of quadrature formulas based on these points.

Remark 4.4.2. Recall that the number of distinct nodes on the sparse grid $\mathscr{H}_L^{m,g}$ is denoted by M_L, which corresponds to the number of basis functions in (4.4.8) and the number of evaluations of the finite element approximation, that is, $M = M_L$ in (2.4.3). However, in general, the number

of probabilistic degrees of freedom $M_p = \dim(\mathcal{P}_{\mathcal{J}^{m,g}(L)}(\Gamma))$, in the approximation $u^{gSG}_{J_h M_p}$, is much smaller. Nonetheless, as we describe in Section 4.5, in order to compare gSCMs and gSGMs fairly we have to take into account the total computational cost required to achieve a desired tolerance. As we will show, our cost analysis is based entirely on the total number of matrix–vector products required by the conjugate gradient solution of the underlying Galerkin and collocation systems.

Clenshaw–Curtis points on bounded hypercubes

Without loss of generality, assume that $\Gamma_n = [-1, 1]$. The Clenshaw–Curtis points are the extrema of Chebyshev polynomials (CC) (see Clenshaw and Curtis 1960) given by, for any choice of $m(l) > 1$,

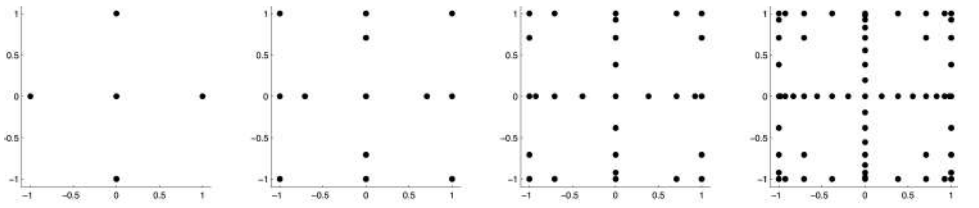
$$y_k^{(l)} = -\cos\left(\frac{\pi(k-1)}{m(l)-1}\right) \quad \text{for } k = 1, \ldots, M_l. \tag{4.4.10}$$

In addition, we set $y_1^{(l)} = 0$ if $m(l) = 1$ and choose the multi-index map g as well as the number of points $m(l)$, $l > 1$, at each level as in (4.4.9). We note that this particular choice corresponds to the most commonly used sparse grid approximation, because it leads to nested sequences of points, *i.e.*,

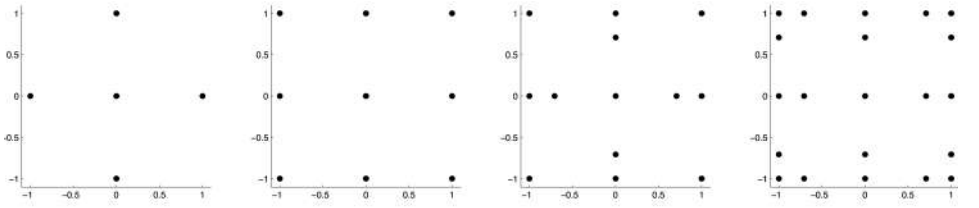
$$\{y_k^{(l)}\}_{k=1}^{m(l)} \subset \{y_k^{(l+1)}\}_{k=1}^{m(l+1)},$$

so that the sparse grids are also nested, that is, $\mathscr{H}_L^{m,g} \subset \mathscr{H}_{L+1}^{m,g}$.

It is important to note that we are interested in optimal approximation for relatively large parameter space dimension N. However, even though this CC choice of points results in a significantly reduced number of points used by $\mathscr{I}_L^{m,g}$, that number of points eventually increases exponentially fast with N. With this in mind, we consider an alternative to the standard Clenshaw–Curtis (CC) family of rules, which attempts to retain the advantages of nestedness while reducing the excessive growth described above. To achieve this, we use the fact that the CC interpolant is exact in the polynomial space $\mathcal{P}_{m(l)-1}$ to drop, in each direction, the requirement that the function m be strictly increasing. Instead, we define a new mapping $\widetilde{m}(l) : \mathbb{N}_+ \to \mathbb{N}_+$ such that $\widetilde{m}(l) \leq \widetilde{m}(l+1)$ and $\widetilde{m}(l) = \widetilde{m}(k)$, where $k = \text{argmin}\{k' \mid 2^{k'-1} \geq L\}$. In other words, we simply re-use the current rule for as many levels as possible, until we properly include the total degree subspace. Figure 4.4.1 shows the difference between the standard CC sparse grid and the 'slow growth' CC (sCC) sparse grid for $l = 1, 2, 3, 4$. Figure 4.4.2 shows the corresponding polynomial accuracy of the CC and sCC sparse grids when used in a quadrature rule approximation (as opposed to an interpolant) of an integral in $C^0(\Gamma)$. Note that the concept of 'slow growth' can also be applied to other nested one-dimensional rules, including, for example, the Gauss–Patterson points (Gerstner and Griebel 1998).

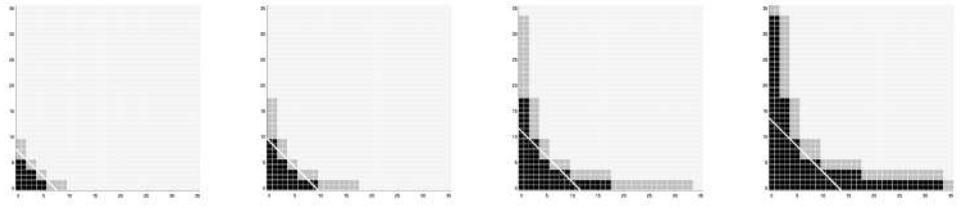


(a)

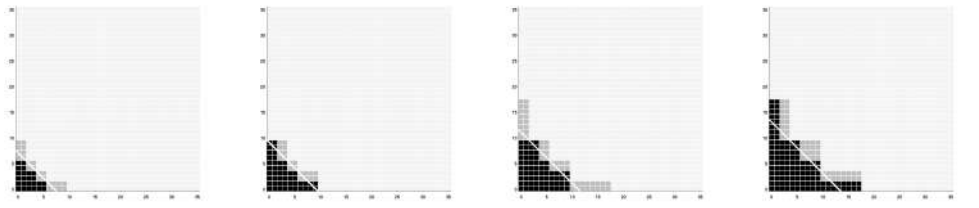


(b)

Figure 4.4.1. For $\Gamma = [-1,1] \times [-1,1]$, the sparse grids corresponding to levels $L = 1, 2, 3, 4$ with (a) standard Clenshaw–Curtis points and (b) slow-growth Clenshaw–Curtis points.



(a)



(b)

Figure 4.4.2. For $\Gamma = [-1,1] \times [-1,1]$, the polynomial subspaces associated with integrating a function $u \in C^0(\Gamma)$, using sparse grids corresponding to levels $L = 3, 4, 5, 6$ with (a) standard Clenshaw–Curtis points and (b) slow-growth Clenshaw–Curtis points.

Gaussian points in bounded or unbounded hypercubes

The Gaussian points $\{y_{n,k}^{(l_n)}\}_{k=1}^{m(l_n)} \subset \Gamma_n$ are the zeros of the orthogonal polynomials with respect to some positive weight function. In general, they are not nested. The natural choice for the weight function is the PDF $\rho(\boldsymbol{y})$ of the random variables $\boldsymbol{y}$. However, in the general multivariate case, if the random variables y_n are not independent, the PDF $\rho(\boldsymbol{y})$ does not factorize, that is,

$$\rho(\boldsymbol{y}) \neq \prod_{n=1}^{N} \rho_n(y_n).$$

As a result, we first introduce an auxiliary probability density function $\widehat{\rho}(\boldsymbol{y}) : \Gamma \to \mathbb{R}^+$ defined by

$$\widehat{\rho}(\boldsymbol{y}) = \prod_{n=1}^{N} \widehat{\rho}_n(y_n) \quad \text{for all } \boldsymbol{y} \in \Gamma, \quad \text{and such that } \left\| \frac{\rho}{\widehat{\rho}} \right\|_{L^\infty(\Gamma)} < \infty.$$

Note that $\widehat{\rho}(\boldsymbol{y})$ factorizes so that it can be viewed as a joint PDF for N independent random variables,

For each parameter dimension $n = 1, \ldots, N$, let the $m(l_n)$ Gaussian points be the roots of the $m(l_n)$ degree polynomial that is $\widehat{\rho}_n$-orthogonal to all polynomials of degree $m(l_n - 1)$ on the interval Γ_n. The auxiliary density $\widehat{\rho}$ should be chosen as close as possible to the true density ρ so that the quotient $\rho/\widehat{\rho}$ is not too large.

Selection of the anisotropic weights

The ability to treat the stochastic dimensions differently is a necessity because many practical problems exhibit highly anisotropic behaviour, for example, the size τ_n of the analyticity region (4.1.2) associated to each random variable y_n increases with n.

We assume that the solution to our problem has analytic dependence with respect to each of the random variables, that is, it satisfies Assumption 4.1.1. In such a case, we know that if we approximate the dependence on each random variable with polynomials, the best approximation error decays exponentially fast with respect to the polynomial degree. More precisely, for a bounded region Γ_n and a univariate analytic function, we recall the following lemma, whose proof can be found in Babuška *et al.* (2007*a*, Lemma 7) and which is an immediate extension of the result given in DeVore and Lorentz (1993, Chapter 7, Section 8).

Lemma 4.4.3. Given a function $v \in C^0(\Gamma_n; W(D))$, which admits an analytic extension in the region of the complex plane

$$\Sigma(\Gamma_n; \tau_n) = \{z \in \mathbb{C}, \ \operatorname{dist}(z, \Gamma_n) \leq \tau_n\}, \quad \text{for some } \tau_n > 0,$$

then

$$E_{m(l_n)} \equiv \min_{w \in \mathcal{P}_{m(l_n)}} \|v - w\|_{C_n^0} \leq \frac{2}{\mathrm{e}^{2r_n} - 1} \mathrm{e}^{-2M_{l_n} r_n} \max_{z \in \Sigma(\Gamma_n; \tau_n)} \|v(z)\|_{W(D)}$$

with

$$0 < r_n = \frac{1}{2} \log\left(\frac{2\tau_n}{|\Gamma_n|} + \sqrt{1 + \frac{4\tau_n^2}{|\Gamma_n|^2}} \right). \tag{4.4.11}$$

A related result with weighted norms holds for unbounded random variables whose probability density decays like the Gaussian density at infinity: see Babuška *et al.* (2007*a*)

In the multivariate case, the size τ_n of the analyticity region depends, in general, on the direction n. As a consequence, the decay coefficient r_n will also depend on the direction. The key idea of the anisotropic sparse gSCM in Nobile *et al.* (2008*a*) is to place more points in directions having slower convergence rate, that is, with smaller value for r_n. In particular, we link the weights α_n with the rate of exponential convergence in the corresponding direction by

$$\alpha_n = r_n \quad \text{for all } n = 1, 2, \ldots, N. \tag{4.4.12}$$

Let

$$\underline{\alpha} = \underline{r} = \min_{n=0,1,\ldots,N} \{r_n\} \quad \text{and} \quad \mathscr{R}(N) = \sum_{n=1}^{N} r_n. \tag{4.4.13}$$

As we observe in Remark 4.4.8, the choice $\boldsymbol{\alpha} = \boldsymbol{r}$ is optimal with respect to the error bound derived in Theorem 4.4.4. Note that we have now transformed the problem of choosing $\boldsymbol{\alpha}$ into one of estimating the decay coefficients $\boldsymbol{r} = (r_1, \ldots, r_N)$. Nobile *et al.* (2008*a*, Section 2.2) have given two rigorous estimation strategies: the first uses *a priori* knowledge about the error decay in each direction, whereas the second uses *a posteriori* information obtained from computations and fits the values of $\boldsymbol{r}$.

An illustration of the salubrious effect on the resulting sparse grid by taking this anisotropy into account is given in Figure 4.4.3.

Sparse grid gSCM error estimates

Global sparse grid Lagrange interpolation gSCMs can be used to approximate the solution $u \in C^0(\Gamma; W(D))$ using finitely many function values. By Assumption 4.1.1, u admits an analytic extension. Furthermore, each function value is computed by means of a finite element technique. Recall that the fully discrete approximation is defined as $u_{J_h M_p}^{gSC} = \mathscr{I}_L^{m,g}[u_{J_h}]$, where the operator $\mathscr{I}_L^{m,g}$ is defined in (4.4.7). Our aim is to provide *a priori* estimates for the total error

$$\epsilon = u - \mathscr{I}_L^{m,g}[u_{J_h}].$$

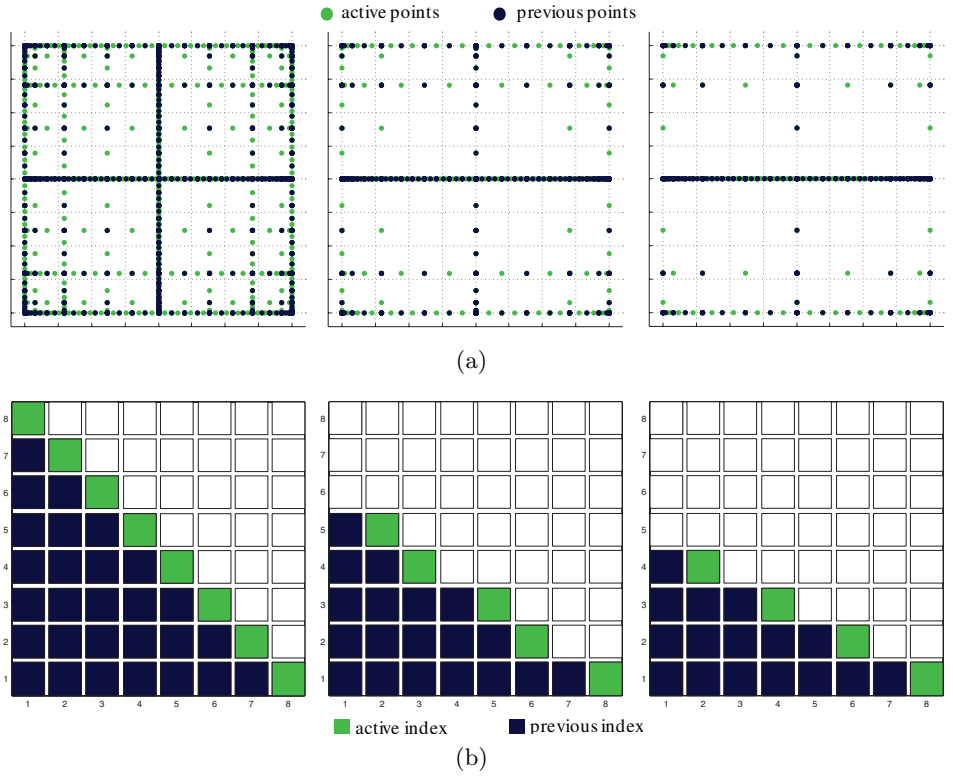


Figure 4.4.3. For $\Gamma = [0,1]\times[0,1]$ and $L = 7$, we plot (a) the anisotropic sparse grids with $\alpha_2/\alpha_1 = 1$ (isotropic), $\alpha_2/\alpha_1 = 3/2$, and $\alpha_2/\alpha_1 = 2$ utilizing the Clenshaw–Curtis points, and (b) the corresponding indices (l_1, l_2) such that $\alpha_1(l_1 - 1) + \alpha_2(l_2 - 1) \le \alpha_{\min} L$.

We will investigate the error[5]

$$\left\|u - \mathscr{I}_L^{m,g}[u_{J_h}]\right\| \le \underbrace{\left\|u - u_{J_h}\right\|}_{\text{(I)}} + \underbrace{\left\|u_{J_h} - \mathscr{I}_L^{m,g}[u_{J_h}]\right\|}_{\text{(II)}} \tag{4.4.14}$$

evaluated in the natural norm $L^2_\rho(\Gamma; W(D))$. By controlling the error in this natural norm, we also control the error in the expected value of the solution, for example,

$$\|\mathbb{E}[\epsilon]\|_{W(D)} \le \mathbb{E}\big[\|\epsilon\|_{W(D)}\big] \le \|\epsilon\|_{L^2_\rho(\Gamma;W(D))}.$$

[5] If the stochastic data, *i.e.*, a and/or f, are not an exact representation but are instead an approximation in terms of N random variables, *e.g.*, arising from a suitable truncation of infinite representations of random fields, then there would be an additional error $\|u - u_N\|$ to consider. This contribution to the total error was considered in Nobile *et al.* (2008*b*, Section 4.2).

The quantity (I) accounts for the error with respect to the spatial grid size h, that is, the finite element error. It is estimated using standard approximability properties of the finite element space $W_h(D)$ and the spatial regularity of the solution u; see, for example, Brenner and Scott (2008) and Ciarlet (1978). Specifically, we have

$$\|u - u_{J_h}\|_{L^2_\rho(\Gamma;W(D))} \le h^s \left(\int_\Gamma C_\pi(\boldsymbol{y}) C(s; u(\boldsymbol{y}))^2 \rho(\boldsymbol{y}) \, \mathrm{d}y \right)^{1/2}.$$

Our primary concern will be to analyse the approximation error (II), that is,

$$\left\| u_{J_h} - \mathscr{I}_L^{m,g}[u_{J_h}] \right\|_{L^2_\rho(\Gamma;W(D))}, \tag{4.4.15}$$

for the Clenshaw–Curtis points using the anisotropic sparse grid approximation with $m(l)$ and g defined as follows:

$$m(1) = 1, \quad m(l) = 2^{l-1} + 1, \quad \text{and} \quad g(\boldsymbol{l}) = \sum_{n=1}^{N} \alpha_n (l_n - 1) \le \alpha_{\min} L. \tag{4.4.16}$$

Convergence for other isotropic and anisotropic choices for $m(l)$ and g, as well as for the sparse grid generated from the Gaussian points, are considered in Nobile *et al.* (2008*a*, 2008*b*).

Under the very reasonable assumption that the semi-discrete finite element solution u_{J_h} admits an analytic extension as described in Assumption 4.1.1, with the same analyticity region as for u, the behaviour of the error (4.4.15) will be analogous to

$$\left\| u - \mathscr{I}_L^{m,g}[u] \right\|_{L^2_\rho(\Gamma;W(D))}.$$

For this reason, in the following analysis we consider the latter.

Recall that even though in the global estimate (4.4.14) it is enough to bound the approximation error (II) in the $L^2_\rho(\Gamma; W(D))$-norm, we consider the more stringent $L^\infty(\Gamma; W(D))$-norm. In our notation the norm $\|\cdot\|_{\infty,n}$ is shorthand for $\|\cdot\|_{L^\infty(\Gamma_n;W(D))}$ and similarly, $\|\cdot\|_{\infty,N}$ is shorthand for $\|\cdot\|_{L^\infty(\Gamma;W(D))}$.

The multi-dimensional error estimate $\left\| u - \mathscr{I}_L^{m,g}[u] \right\|$ is constructed from a sequence of one-dimensional estimates and a tight bound on the number of distinct nodes on the sparse grid $\mathscr{H}_L^{m,g}$. To start with, we again let E_m denote the best approximation error, as in Lemma 4.4.3, to functions $u \in C^0(\Gamma_n; W(D))$ by polynomial functions $w \in \mathcal{P}_M$. Because the interpolation formula $\mathscr{U}_n^{M_{l_n}}$, $n = 1, \ldots, N$, is exact for polynomials in $\mathcal{P}_{m(l_n)-1}$, we can apply the general formula

$$\left\| u - \mathscr{U}^{m(l_n)}(u) \right\|_{\infty,n} \le \left(1 + \Lambda_{m(l_n)}\right) E_{m(l_n)-1}(u), \tag{4.4.17}$$

where Λ_m denotes the Lebesgue constant for the points (4.4.10). In this case, it is known that

$$\Lambda_m \leq \frac{2}{\pi}\log(m-1)+1 \tag{4.4.18}$$

for $M_{l_n} \geq 2$; see Dzjadyk and Ivanov (1983). On the other hand, using Lemma 4.4.3, the best approximation to functions $u \in C^0(\Gamma_n; W(D))$ that admit an analytic extension as described by Assumption 4.1.1 is bounded by

$$E_{m(l_n)}(u) \leq \frac{2}{\mathrm{e}^{2r_n}-1}\,\mathrm{e}^{-2m(l_n)r_n}\theta(u), \tag{4.4.19}$$

where

$$\theta(u) = \max_{1\leq n\leq N}\max_{y_n^*\in\Gamma_n^*}\max_{z\in\Sigma(\Gamma_n;\tau_n)} \|u(z)\|_{W(D)}.$$

For $n = 1, 2, \ldots, N$, let

$$I_n : C^0(\Gamma_n; W(D)) \to C^0(\Gamma_n; W(D))$$

denote the one-dimensional identity operator, and use (4.4.17)–(4.4.19) to obtain the estimates

$$\left\|(I_n - \mathscr{U}_n^{m(l_n)})(u)\right\|_{\infty,n} \leq \frac{4}{\mathrm{e}^{2r_n}-1}\,l_n\,\mathrm{e}^{-r_n 2^{l_n}}\,\theta(u)$$

and

$$\left\|(\Delta_n^{m(l_n)})(u)\right\|_{\infty,n} \leq \frac{8}{\mathrm{e}^{2r_n}-1}\,l_n\,\mathrm{e}^{-r_n 2^{l_n-1}}\theta(u). \tag{4.4.20}$$

Because the value $\theta(u)$ affects the error estimates as a multiplicative constant, from here on we assume it to be one without any loss of generality.

The next theorem provides an error bound in terms of the total number M_L of Clenshaw–Curtis collocation points. The details of the proof can be found in Nobile *et al.* (2008*a*, Section 3.1.1) and are therefore omitted. A similar result holds for the sparse grid $\mathscr{I}_L^{m,g}$ using Gaussian points, and can found in Nobile *et al.* (2008*a*, Section 3.1.2).

Theorem 4.4.4. Let $u \in L^2_\rho(\Gamma; W(D))$ and let the functions m and g satisfy (4.4.16) with weights $\alpha_n = r_n$. Then, for the gSCM approximation based on the Clenshaw–Curtis points, we have the following estimates.

- *Algebraic convergence* $\left(0 \leq L \leq \frac{\mathscr{R}(N)}{\underline{r}\log(2)}\right)$:

$$\|(I_N - \mathscr{A}_{\boldsymbol{\alpha}}(L,N))(u)\|_{L^\infty(\Gamma^N;W(D))} \leq \widehat{C}(\boldsymbol{r},N)M_L^{-\mu_1} \tag{4.4.21}$$

$$\text{with } \mu_1 = \frac{\underline{r}\,(\log(2)\,\mathrm{e}-1/2)}{\log(2)+\sum_{n=1}^N \underline{r}/g(n)}.$$

- *Sub-exponential convergence* $\big(L > \frac{\mathscr{R}(N)}{\underline{r}\log(2)}\big)$:

$$\begin{aligned} &\|(I_N - \mathscr{A}_{\boldsymbol{\alpha}}(L,N))(u)\|_{L^\infty(\Gamma^N;W(D))} \\ &\quad \le \widehat{C}(\boldsymbol{r},N)\, M_L^{\frac{\mu_2}{2}}\, \exp\Big(-\mathscr{R}(N) M_L^{\frac{\log(2)}{\mathscr{R}(N)}\mu_2}\Big) \\ &\quad \text{with } \mu_2 = \frac{\underline{r}}{\big(\log(2) + \sum_{n=1}^N \underline{r}/g(n)\big)}, \end{aligned} \tag{4.4.22}$$

  where the constant $\widehat{C}(\boldsymbol{r},N)$, defined in Nobile *et al.* (2008*a*, (3.14)), is independent of M_L.

Remark 4.4.5. The estimate (4.4.22) may be improved when $L \to \infty$. Such an asymptotic estimate is obtained using the better counting result described in Nobile *et al.* (2008*a*, Remark 3.7).

Remark 4.4.6. We observe that sub-exponential rate of convergence is always faster than the algebraic one when $L > \mathscr{R}(N)/(\underline{r}\log(2))$. However, this estimate is of little practical relevance since in practical computations such high L is seldom reached.

Remark 4.4.7 (on the curse of dimensionality). Suppose the stochastic input data are truncated expansions of random fields and that we are able to estimate the values $\{r_n\}_{n=1}^\infty$. Whenever the sum $\sum_{n=1}^\infty \underline{r}/r_n$ is finite, the algebraic exponent in (4.4.21) does not deteriorate as the truncation dimension N increases. This condition is satisfied, for example, by the problem discussed in Section 4.5. This is a clear advantage compared to the isotropic Smolyak method studied in Nobile *et al.* (2008*b*) because we have $r_n \to +\infty$ and we can show that $\widehat{C}(\boldsymbol{r},N)$ does not deteriorate with N, that is, it is bounded, and therefore *the method does not suffer from the curse of dimensionality.* In fact, in such a case, we can work directly with the anisotropic Smolyak formula in infinite dimensions, that is, $\sum_{n=1}^\infty (l_n - 1) r_n \le L\underline{r}$.

The condition $\sum_{n=1}^\infty \underline{r}/r_n < \infty$ is clearly sufficient to break the curse of dimensionality. In that case, even an anisotropic full tensor approximation also breaks the curse of dimensionality.

The algebraic exponent for the convergence of the anisotropic full tensor approximation again deteriorates with the value of $\sum_{n=1}^\infty \underline{r}/r_n$, but the constant for such convergence is

$$\sum_{n=1}^N \frac{2}{\mathrm{e}^{2r_n} - 1}.$$

This constant is worse than the one corresponding to the anisotropic Smolyak approximation $\widehat{C}(\boldsymbol{r},N)$.

On the other hand, by considering the case where all r_n are equal, and using the results derived in Nobile *et al.* (2008*b*), we see that our estimate of

the algebraic convergence exponent is not sharp. We expect the anisotropic Smolyak method to break the curse of dimensionality for a wider set of problems, that is, the condition $\sum_{n=1}^{\infty} \underline{r}/r_n < \infty$ does not seem to be necessary to break the curse of dimensionality. This is in agreement with Remark 4.4.5.

Remark 4.4.8 (optimal choice of weights $\boldsymbol{\alpha}$). Looking at the exponential term $\mathrm{e}^{-h(\boldsymbol{l},d)}$, where $h(\boldsymbol{l},d) = \sum_{n=1} r_n 2^{l_n-1}$, which determines the rate of convergence, we can try to choose the weight $\boldsymbol{\alpha}$ as the solution to the optimization problem

$$\max_{\substack{\boldsymbol{\alpha}\in\mathbb{R}_+^N \\ |\boldsymbol{\alpha}|=1}} \min_{\widetilde{g}(\boldsymbol{l})\le \underline{\alpha} L} h(\boldsymbol{l}, N),$$

where $\widetilde{g}(\boldsymbol{l}) = \sum_{n=1}^{N} \alpha_n (l_n - 1)$. This problem has the solution $\boldsymbol{\alpha} = \boldsymbol{r}$ and hence our choice of weights (4.4.12) is optimal.

4.4.3. Non-intrusive spectral projection onto an orthonormal basis

The interpolatory approaches described in Section 4.4.1 evaluate the semi-discrete approximation $u_{J_h}(\cdot, \boldsymbol{y}_m)$ given by (2.3.5) on an appropriate set of points $\{\boldsymbol{y}_m\}_{m=1}^{M} \in \Gamma$ and then apply a global, possibly interpolatory, polynomial to construct the approximation $u_{J_h M_L}^{gSC}(\mathbf{x}, \boldsymbol{y})$. However, the resulting interpolant is not L_ρ^2-optimal with respect to the selected function basis and only seeks to match the value of $u_{J_h}(\cdot, \boldsymbol{y}_m)$ at the collocation points. On the other hand, the Galerkin projection methods described in Section 4.3 produce an optimal approximation, but require the solution of a fully coupled $J_h M_p \times J_h M_p$ system of equations.

Given a set of basis functions $\psi_{\boldsymbol{p}}(\boldsymbol{y})$, the non-intrusive orthonormal approximation approach[6] (Reagana, Najm, Ghanem and Knio 2003, Ghanem and Spanos 2003, Migliorati, Nobile, Von Schwerin and Tempone 2013, Eldred *et al.* 2008) seeks to construct a fully discrete approximation, denoted by $u_{J_h M}^{SC}(\mathbf{x}, \boldsymbol{y})$, similar to (4.3.1), but which uses fewer samples from the semi-discrete approximation. As such, it is a variation of the stochastic collocation approach for which the coefficients $u_{\boldsymbol{p}}(\mathbf{x})$ are chosen to minimize the $L_\rho^2(\Gamma; W_h(D))$ error norm

$$\left\| u(\mathbf{x}, \boldsymbol{y}) - \sum_{\boldsymbol{p}\in\mathcal{J}(p)} u_{\boldsymbol{p}}(\mathbf{x}) \psi_{\boldsymbol{p}}(\boldsymbol{y}) \right\|_{L_\rho^2(\Gamma; W(D))}, \tag{4.4.23}$$

and hence they must satisfy the linear system of equations

$$\sum_{\boldsymbol{p}\in\mathcal{J}(p)} \int_\Gamma u_{\boldsymbol{p}}(\mathbf{x}) \psi_{\boldsymbol{p}}(\boldsymbol{y}) \psi_{\boldsymbol{p}'}(\boldsymbol{y}) \rho(\boldsymbol{y}) \,\mathrm{d}\boldsymbol{y} = \int_\Gamma u(\mathbf{x}, \boldsymbol{y}) \psi_{\boldsymbol{p}'}(\boldsymbol{y}) \rho(\boldsymbol{y}) \,\mathrm{d}\boldsymbol{y} \tag{4.4.24}$$

[6] This approach is often referred to as the *non-intrusive polynomial chaos* (NIPC) method.

for all $\boldsymbol{p}' \in \mathcal{J}(p)$. If the system (4.4.24) is written in matrix form, the left-hand side corresponds to the mass matrix associated with the basis and furthermore, if $\{\psi_{\boldsymbol{p}}(\boldsymbol{y})\}_{\boldsymbol{p}\in\mathcal{J}(p)}$ are L^2_ρ-orthonormal, that is,

$$\int_\Gamma \psi_{\boldsymbol{p}}(\boldsymbol{y})\psi_{\boldsymbol{p}'}(\boldsymbol{y})\rho(\boldsymbol{y})\,\mathrm{d}\boldsymbol{y} = \delta_{\boldsymbol{p}\boldsymbol{p}'},$$

then (4.4.24) decouples so that

$$\begin{aligned} u_{\boldsymbol{p}}(\mathbf{x}) &= \int_\Gamma u(\mathbf{x},\boldsymbol{y})\psi_{\boldsymbol{p}}(\boldsymbol{y})\rho(\boldsymbol{y})\,\mathrm{d}\boldsymbol{y} \approx \int_\Gamma u_{J_h}(\mathbf{x},\boldsymbol{y})\psi_{\boldsymbol{p}}(\boldsymbol{y})\rho(\boldsymbol{y})\,\mathrm{d}\boldsymbol{y} \\ &= \sum_{j=1}^{J_h}\Bigg(\sum_{m=1}^{M} w_m u(\mathbf{x}_j,\boldsymbol{y}_m)\psi_{\boldsymbol{p}}(\boldsymbol{y}_m)\Bigg)\phi_j(\mathbf{x}), \end{aligned} \tag{4.4.25}$$

where $\{w_m,\boldsymbol{y}_m\}_{m=1}^M$ are the selected quadrature weights and points. The main challenge of this non-intrusive approach is that the integral for the coefficients $u_{\boldsymbol{p}}(\mathbf{x})$, given by (4.4.25), can be very high-dimensional. However, we can make use of the sampling methods described in Sections 3.3–3.5 or the sparse grid methods described in Section 4.4.1, which combat the curse of dimensionality and produce accurate high-dimensional quadrature rules. Thus, using only a set of samples from the semi-discrete approximation, we obtain the 'near-optimal[7]' $L^2_\rho(\Gamma)$ projection.

Least-squares methods

The least-squares method (LS) (see, *e.g.*, Le Maître and Knio 2010 and the references therein) is a statistical approach for minimizing the discrepancy between an approximation and a set of samples. Given a number of samples from $u(\mathbf{x},\boldsymbol{y}_m) \approx u_{J_h}(\mathbf{x},\boldsymbol{y}_m)$, $m = 1,\dots,M$, the LS approach seeks an approximation that solves the optimization problem

$$\min_{u_{\boldsymbol{p}}(\mathbf{x})} F(u_{\boldsymbol{p}}(\mathbf{x})) = \min_{u_{\boldsymbol{p}}(\mathbf{x})} \sum_{m}^{M}\Bigg(u(\mathbf{x},\boldsymbol{y}_m) - \sum_{\boldsymbol{p}\in\mathcal{J}(p)} u_{\boldsymbol{p}}(\mathbf{x})\psi_{\boldsymbol{p}}(\boldsymbol{y}_m)\Bigg)^2. \tag{4.4.26}$$

The minimum is attained at $u_{\boldsymbol{p}}(\mathbf{x})$, which satisfies the system of linear equations

$$0 = \frac{\partial F(u_{\boldsymbol{p}}(\mathbf{x}))}{u_{\boldsymbol{p}'}} = -2\sum_{m}^{M}\Bigg(u(\mathbf{x},\boldsymbol{y}_m) - \sum_{\boldsymbol{p}\in\mathcal{J}(p)} u_{\boldsymbol{p}}(\mathbf{x})\psi_{\boldsymbol{p}}(\boldsymbol{y}_m)\Bigg)\psi_{\boldsymbol{p}'}(\boldsymbol{y}_m) \tag{4.4.27}$$

for $\boldsymbol{p}' \in \mathcal{J}(p)$. If the number of samples and the number of basis functions are equal, then (4.4.27) is equivalent to the interpolation problem of finding

[7] The approximation is optimal only if the right-hand side integrals are computed exactly. In practice, the quadrature error will contaminate the approximation and can potentially dominate all other sources of numerical error.

$u_{\boldsymbol{p}}(\mathbf{x})$ that satisfy

$$u_{J_h M}^{SC}(\mathbf{x}, \boldsymbol{y}_m) = \sum_{\boldsymbol{p} \in \mathcal{J}(p)} u_{\boldsymbol{p}}(\mathbf{x}) \psi_{\boldsymbol{p}}(\boldsymbol{y}_m). \tag{4.4.28}$$

However, in order to avoid the potential ill-conditioning of the interpolation matrix, the LS approach utilizes more samples than basis functions, that is, it over-determines the system. On the other hand, if the samples are taken according to a Monte Carlo strategy, then the corresponding LS approximation is asymptotically equivalent to (4.4.24) (Pukelsheim 1993). Other sampling techniques can be utilized as well: see, for example, Hardin and Sloane (1993) and Pukelsheim (1993). In effect, the LS approach is a compromise between the pointwise interpolation approach and the global optimal projection method.

Compressed sensing

Compressed sensing (CS) (see, *e.g.*, Doostan and Owhadi 2011, Mathelin and Gallivan 2010 and the references therein) is a model reduction approach that assumes $u(\mathbf{x}, \boldsymbol{y})$ can be well approximated by only a small number of basis functions. In other words, given an approximation of the form (4.3.1), there are two sets of coefficients $u_{\boldsymbol{p}}^{\epsilon_0}(\mathbf{x})$ and $u_{\boldsymbol{p}}^{\epsilon_1}(\mathbf{x})$ such that

$$\sum_{\boldsymbol{p} \in \mathcal{J}(p)} u_{\boldsymbol{p}}(\mathbf{x}) \psi_{\boldsymbol{p}}(\boldsymbol{y}) = \sum_{\boldsymbol{p} \in \mathcal{J}(p)} u_{\boldsymbol{p}}^{\epsilon_0}(\mathbf{x}) \psi_{\boldsymbol{p}}(\boldsymbol{y}) + \sum_{\boldsymbol{p} \in \mathcal{J}(p)} u_{\boldsymbol{p}}^{\epsilon_1}(\mathbf{x}) \psi_{\boldsymbol{p}}(\boldsymbol{y}),$$

where $\|u_{\boldsymbol{p}}^{\epsilon_0}(\mathbf{x})\|_{L_\rho^0(\Gamma)}$ is small (*i.e.*, most of the $u_{\boldsymbol{p}}^{\epsilon_0}(\mathbf{x})$ are zero) and

$$\|u_{\boldsymbol{p}}^{\epsilon_1}(\mathbf{x})\|_{W(D)} < \epsilon \quad \text{for all } \boldsymbol{p} \in \mathcal{J}(p).$$

The CS approach considers a set of samples $u(\mathbf{x}, \boldsymbol{y}_m)$ and seeks to find the coefficients $u_{\boldsymbol{p}}^{\epsilon_0}(\mathbf{x})$. The optimization problem can be written as

$$\begin{cases} \min\limits_{u_{\boldsymbol{p}}^{\epsilon_0}(\mathbf{x})} \|u_{\boldsymbol{p}}^{\epsilon_0}(\mathbf{x})\|_{L_\rho^0(\Gamma)} \\ \text{subject to } \sum\limits_{m}^{M} \left(u(\mathbf{x}, \boldsymbol{y}_m) - \sum\limits_{\boldsymbol{p} \in \mathcal{J}(p)} u_{\boldsymbol{p}}^{\epsilon_0}(\mathbf{x}) \psi_{\boldsymbol{p}}(\boldsymbol{y}) \right)^2 \leq \epsilon. \end{cases} \tag{4.4.29}$$

The $L_\rho^0(\Gamma)$-optimization problem is NP-hard and hence infeasible in many circumstances. The common practice is to replace (4.4.29) with an equivalent $L_\rho^1(\Gamma)$ problem, that is,

$$\begin{cases} \min\limits_{u_{\boldsymbol{p}}^{\epsilon_0}(\mathbf{x})} \|u_{\boldsymbol{p}}^{\epsilon_0}(\mathbf{x})\|_{L_\rho^1(\Gamma)} \\ \text{subject to } \sum\limits_{m}^{M} \left(u(\mathbf{x}, \boldsymbol{y}_m) - \sum\limits_{\boldsymbol{p} \in \mathcal{J}(p)} u_{\boldsymbol{p}}^{\epsilon_0}(\mathbf{x}) \psi_{\boldsymbol{p}}(\boldsymbol{y}) \right)^2 \leq \epsilon. \end{cases} \tag{4.4.30}$$

If sufficiently sparse $u_{\boldsymbol{p}}^{\epsilon_0}(\mathbf{x})$ exists, then a solution to (4.4.30) exists for a very small number of samples and hence the dominant coefficients can be found at a cost that is highly reduced when compared to the previous methods described above.

4.5. Computational complexity comparisons

We now focus our attention on a comparison of gSGM, using an orthogonal basis, with gSCM, using a Lagrange basis, for solving the stochastic linear elliptic problem described in Example 2.1.2 in two spatial dimensions. We consider the problem first used in Nobile *et al.* (2008*a*, 2008*b*) and used in many subsequent research papers.

The problem is to solve

$$\begin{aligned} -\nabla \cdot (a(\cdot,\omega)\nabla u(\cdot,\omega)) &= f(\cdot,\omega) \quad \text{in } D\times\Omega, \\ u(\cdot,\omega) &= 0 \qquad\quad \text{on } \partial D\times\Omega, \end{aligned} \tag{4.5.1}$$

with $D=[0,b]^2$. We choose the deterministic load

$$f(x_1,x_2,\omega)=\cos(x_1)\sin(x_2)$$

and the random coefficient $a(\mathbf{x},\omega)$ with one-dimensional spatial dependence given by

$$\log(a(\mathbf{x},\omega)-0.5)=1+y_1(\omega)\left(\frac{\sqrt{\pi}C}{2}\right)^{1/2}+\sum_{n=2}^{N}\zeta_n\,\varphi_n(\mathbf{x})\,y_n(\omega), \tag{4.5.2}$$

where

$$\zeta_n:=\left(\sqrt{\pi}C\right)^{1/2}\exp\left(\frac{-\left(\lfloor\frac{n}{2}\rfloor\pi C\right)^2}{8}\right) \quad \text{if } n>1 \tag{4.5.3}$$

and

$$\varphi_n(\mathbf{x}):=\begin{cases}\sin\left(\dfrac{\lfloor\frac{n}{2}\rfloor\pi x_1}{C_p}\right) & \text{if } n \text{ even},\\[2ex] \cos\left(\dfrac{\lfloor\frac{n}{2}\rfloor\pi x_1}{C_p}\right) & \text{if } n \text{ odd}.\end{cases} \tag{4.5.4}$$

For $x_1\in[0,b]$, let C_l denote a desired physical correlation length for the random field a, meaning that the random variables $a(x_1,\omega)$ and $a(x_1',\omega)$ become essentially uncorrelated for $|x_1-x_1'|\gg C_l$. Then, the parameter C_p in (4.5.4) can be taken as $C_p=\max\{b,2C_l\}$ and the parameter C in (4.5.2) and (4.5.3) is given by $C=C_l/C_p$. Expression (4.5.2) represents a possible truncation of a one-dimensional random field with stationary covariance

$$\mathbb{COV}[\log(a-0.5)](x_1,x_1')=\exp\left\{-\frac{(x_1-x_1')^2}{C_l^2}\right\}.$$

In this example, the random variables $\{y_n(\omega)\}_{n=1}^{\infty}$ are independent, have zero mean and unit variance, that is, $\mathbb{E}[y_n] = 0$ and $\mathbb{E}[y_n y_{n'}] = \delta_{nn'}$ for $n, n' \in \mathbb{N}_+$, and are uniformly distributed in the interval $[-\sqrt{3}, \sqrt{3}]$.

Because the random variables y_n are uniformly distributed, the orthogonal polynomials in the gSGM correspond to the Legendre polynomials. Moreover, due to the boundedness of y_n, we can use the Gauss–Legendre or the Clenshaw–Curtis points. The finite element space for the spatial discretization is the span of continuous functions that are piecewise polynomials of degree two over a uniform triangulation of D with 4 225 spatial unknowns.

We next compare the cost associated with setting up and solving the fully discrete approximations $u^{gSG}_{J_h M_p}$ and $u^{gSC}_{J_h M_L}$ described in Sections 4.3 and 4.4, respectively.

4.5.1. The cost of constructing the Galerkin and collocation systems

In order to construct a highly sparse, symmetric, and positive definite coupled system of algebraic system of equations (4.3.5), describing the gSGM approximation of (4.5.1), we first need to project $a(\mathbf{x}, \boldsymbol{y})$ onto the orthonormal basis, that is, use the non-intrusive spectral projection described in Section 4.4.3, and by letting ϵ_{SG} be the error in SG approximation, we must choose $q \in \mathbb{N}$ such that

$$\left| a(\mathbf{x}, \boldsymbol{y}) - \sum_{\boldsymbol{q} \in \mathcal{J}(q)} a_{\boldsymbol{q}}(\mathbf{x}) \psi_{\boldsymbol{q}}(\boldsymbol{y}) \right| < \epsilon_{SG}, \tag{4.5.5}$$

where

$$a_{\boldsymbol{q}}(\mathbf{x}) = \int_{\Gamma} a(\mathbf{x}, \boldsymbol{y}) \psi_{\boldsymbol{q}}(\boldsymbol{y}) \rho(\boldsymbol{y}) \, \mathrm{d}\boldsymbol{y}.$$

This requires the evaluation of an N-dimensional quadrature due to the exponential expansion of $a(\mathbf{x}, \boldsymbol{y})$. Substituting $a(\mathbf{x}, \boldsymbol{y}) = \sum_{\boldsymbol{q} \in \mathcal{J}(q)} a_{\boldsymbol{q}}(\mathbf{x}) \psi_{\boldsymbol{q}}(\boldsymbol{y})$ into (4.3.3) yields, for all $j, j' = 1 \ldots, J_h$,

$$\begin{aligned} \boldsymbol{A}_{j,j'}(\boldsymbol{y}) &= \sum_{\boldsymbol{q} \in \mathcal{J}(q)} \psi_{\boldsymbol{q}}(\boldsymbol{y}) \int_D a_{\boldsymbol{q}}(\mathbf{x}) \, \nabla \phi_j(\mathbf{x}) \cdot \nabla \phi_{j'}(\mathbf{x}) \, \mathrm{d}\mathbf{x} \\ &= \sum_{\boldsymbol{q} \in \mathcal{J}(q)} \psi_{\boldsymbol{q}}(\boldsymbol{y}) [\boldsymbol{A}_{\boldsymbol{q}}]_{j,j'}, \end{aligned} \tag{4.5.6}$$

where $[\boldsymbol{A}_{\boldsymbol{q}}]_{j,j'} = \int_D a_{\boldsymbol{q}}(\mathbf{x}) \, \nabla \phi_j(\mathbf{x}) \cdot \nabla \phi_{j'}(\mathbf{x}) \, \mathrm{d}\mathbf{x}$ can be computed componentwise by utilizing a quadrature rule over J_h elements on the mesh $\mathcal{T}_h$.

Given a sufficiently resolved stochastic finite element stiffness matrix $\boldsymbol{A}(\boldsymbol{y}) = \sum_{\boldsymbol{q} \in \mathcal{J}(q)} [\boldsymbol{A}_{\boldsymbol{q}}] \psi_{\boldsymbol{q}}(\boldsymbol{y})$, we substitute $\boldsymbol{A}(\boldsymbol{y})$ into (4.3.5) and obtain,

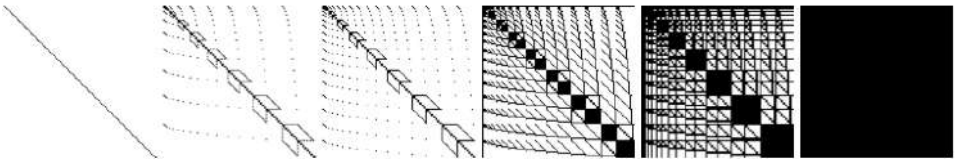


Figure 4.5.1. Structure of the matrix $\boldsymbol{K}$ in (4.5.9), where we use a $p = 4$ approximation in the sparse Smolyak polynomial subspace but increase the order of the data projection, that is, $q = 0, 1, 2, 4, 5, 9$. In each case, the matrix is a 145×145 block matrix, but the sparsity ratio decreases until the matrix is full at order $q = 9$.

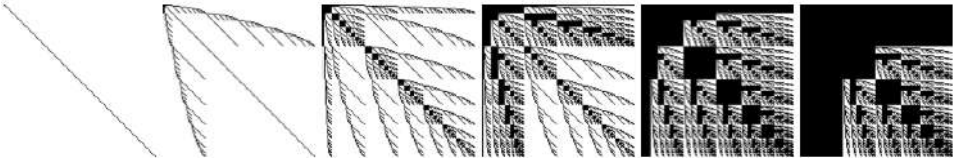


Figure 4.5.2. Structure of the matrix $\boldsymbol{K}$ in (4.5.9), where we use a $p = 3$ approximation in the total degree polynomial subspace but increase the order of the data projection, that is, $q = 0, 1, 2, 4, 5$. In each case, the matrix is a 165×165 block matrix, but the sparsity ratio decreases until the matrix eventually becomes full.

for all $\boldsymbol{p}' \in \mathcal{J}(p)$,

$$\sum_{\boldsymbol{p}\in\mathcal{J}(p)} \sum_{\boldsymbol{q}\in\mathcal{J}(q)} \left[\int_\Gamma [\boldsymbol{A}_{\boldsymbol{q}}] \psi_{\boldsymbol{q}}(\boldsymbol{y}) \psi_{\boldsymbol{p}'}(\boldsymbol{y}) \psi_{\boldsymbol{p}}(\boldsymbol{y}) \rho(\boldsymbol{y}) \, \mathrm{d}\boldsymbol{y} \right] \boldsymbol{u}_{\boldsymbol{p}} = \boldsymbol{F}_{\boldsymbol{p}'}, \tag{4.5.7}$$

where $\boldsymbol{u}_{\boldsymbol{p}}$ and $\boldsymbol{F}_{\boldsymbol{p}}$ are introduced in Example (4.3.1). By defining

$$[\boldsymbol{G}_{\boldsymbol{q}}]_{\boldsymbol{p}',\boldsymbol{p}} = \int_\Gamma \psi_{\boldsymbol{q}} \psi_{\boldsymbol{p}'} \psi_{\boldsymbol{p}} \rho(\boldsymbol{y}) \, \mathrm{d}\boldsymbol{y} \quad \text{and} \quad \boldsymbol{K} = \sum_{\boldsymbol{q}\in\mathcal{J}(q)} [\boldsymbol{G}_{\boldsymbol{q}}] \bigotimes [\boldsymbol{A}_{\boldsymbol{q}}], \tag{4.5.8}$$

where $[\boldsymbol{G}_{\boldsymbol{q}}] \bigotimes [\boldsymbol{A}_{\boldsymbol{q}}]$ denotes the Kronecker product of $[\boldsymbol{G}_{\boldsymbol{q}}]$ and $[\boldsymbol{A}_{\boldsymbol{q}}]$, we obtain the gSGM coupled system of equations, namely,

$$\boldsymbol{K}\boldsymbol{u} = \boldsymbol{F}, \tag{4.5.9}$$

with $\boldsymbol{K}$ symmetric and positive definite.

Figures 4.5.1 and 4.5.2 display the effect of fixing the projection order p of the solution but increasing the order q of the data projection: the matrix $\boldsymbol{K}$ loses its sparsity as q increases. These increases would be required if the data were highly nonlinear in order to minimize the error of the projection as in (4.5.5).

Each coefficient matrix $[\boldsymbol{A}_{\boldsymbol{q}}]$ in the expansion of the operator $\boldsymbol{A}(\boldsymbol{y})$ requires $N_e N_q * N_a$ evaluations of the coefficient $a(\mathbf{x}, \boldsymbol{y})$, where N_e is the number of finite elements (*i.e.*, the cardinality of τ_h for a given h), N_q is the number of quadrature points per element, and N_a denotes the number

of quadrature points used to approximate the integral

$$\int_\Gamma a(\mathbf{x}, \boldsymbol{y})\psi_{\boldsymbol{q}}(\boldsymbol{y})\rho(\boldsymbol{y})\,\mathrm{d}\boldsymbol{y}.$$

In the case that $a(\mathbf{x}, \omega)$ is affine, $N_a \approx (N+1) * N_{a_1}$, where N_{a_1} is the number of quadrature points used in one dimension. Then, the *set-up* cost for constructing the non-zero entries of the matrix $\boldsymbol{K}$ is given by

$$W^{SG}_{\text{set-up}} \approx M_p N_e N_q * N_a, \tag{4.5.10}$$

where we recall that $M_p = \dim\{\mathcal{P}_{\mathcal{J}(p)}\}$.

On the other hand, for the gSCM, we must construct M_L finite element systems, requiring work on the order of

$$W^{SC}_{\text{set-up}} \approx M_L N_e N_q. \tag{4.5.11}$$

Note that even set-up costs can dramatically affect the total computational cost; we do not take this into account in the results shown in Figure 4.5.3.

4.5.2. Cost of solving the Galerkin and collocation systems

For the solution of the stochastic Galerkin system, we use a preconditioned conjugate gradient (CG) method. Previous efforts (Elman *et al.* 2011, Beck *et al.* 2011) have performed similar computational comparisons, and use the work of solving one deterministic finite element problem as a metric for measuring the total computational cost of both the gSGM and gSCM. On the other hand, our cost analysis is based entirely on the number of matrix–vector products involved per CG iteration of the gSGM and gSCM solutions. This metric enables a truly fair comparison of the overall computational complexity associated with both approaches.

Given an expansion of the form (4.5.6) of the operator $\boldsymbol{A}(\boldsymbol{y})$ and the Kronecker product form of the SG operator $\boldsymbol{K}$, we define

$$N_G = \sum_{\boldsymbol{q}\in\mathcal{J}(q)} \text{number of non-zeros in } [\boldsymbol{G}_{\boldsymbol{q}}]$$

to be the total number of non-zeros in the matrices $\{[\boldsymbol{G}_{\boldsymbol{q}}]\}_{\boldsymbol{q}\in\mathcal{J}(q)}$. At each iteration of the preconditioned CG (PCG) method, each non-zero block in $\boldsymbol{K}$ in (4.5.9) implies a matrix–vector product of the form (4.3.5). Therefore, our cost estimate for the stochastic Galerkin method, based on the sparsity of the spectral Galerkin system, is then given by

$$W^{SG}_{\text{solve}} \approx N_G N_{\text{iter}}, \tag{4.5.12}$$

where N_{iter} is the number of PCG iterations. As the density of the Galerkin system $\boldsymbol{K}$ increases, more matrix–vector products are required in order to iterate the PCG method.

On the other hand, for the gSCM, the total cost of constructing the fully discrete approximation $u^{SC}_{J_h M_p}$ is defined as

$$W^{SC}_{\text{solve}} \approx \sum_{k=1}^{M_L} N_k,$$

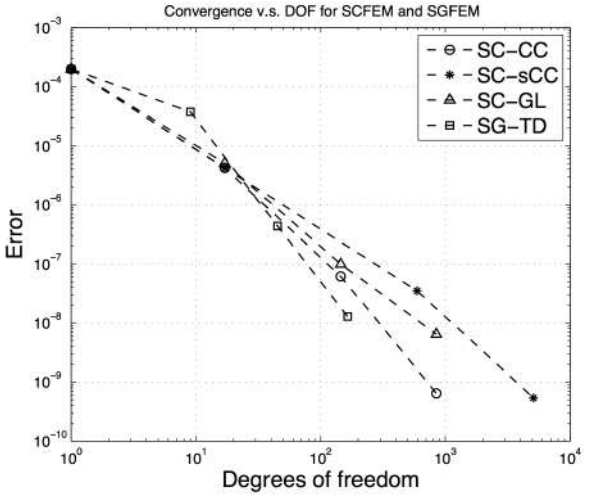
which is the number of iterations required by the PCG method to solve all M_L distinct required finite element simulations; here N_k denotes the number of iterations the PCG method requires to solve the kth realization.

To study the convergence of both the gSGM and the sparse grid gSCM, we consider a problem with a fixed dimension $N = 8$ and correlation length $C = 1/64$, and investigate the behaviour as the order p (Galerkin) and the level L (collocation) increase, respectively. We note that this is essentially an isotropic problem, that is, almost all y_n, $n = 1, \ldots, 8$, have equal weight in the solution. Thus, we will only consider the behaviour with respect to the isotropic polynomial subspaces described in Section 4.2.

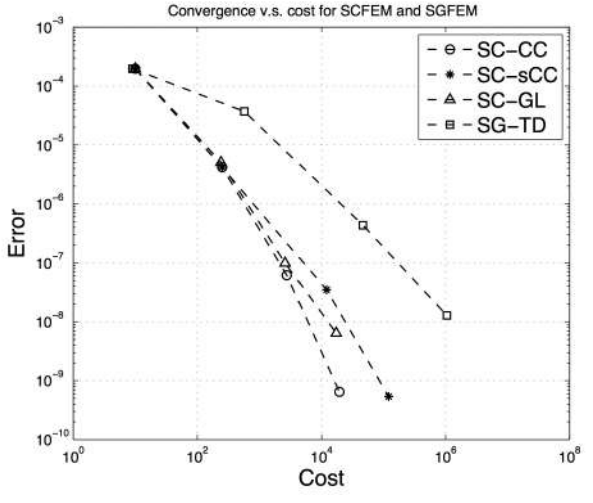
Because we do not know the exact solution for this problem, we check the convergence of the expected value of the solution with respect to a 'highly enriched' solution, which we consider close enough to the exact one. To construct this 'exact' solution u_{ex}, we make use of the isotropic sparse grid gSCEM in the sparse Smolyak subspace with $L = 8$, which uses more than 20 000 Clenshaw–Curtis points. We approximate the computational error for the gSGM with $p = 0, 1, \ldots$ and for the gSCM with $L = 0, 1, \ldots$, as

$$\|\mathbb{E}[\epsilon_{SG}]\| \approx \|\mathbb{E}[u_{ex} - u^{gSG}_{J_h M_p}]\| \quad \text{and} \quad \|\mathbb{E}[\epsilon_{SC}]\| \approx \|\mathbb{E}[u_{ex} - u^{gSC}_{J_h M_L}]\|, \tag{4.5.13}$$

where $u^{gSG}_{J_h M_p}$ and $u^{gSC}_{J_h M_L}$ are given by (4.3.1) and (4.4.8), respectively. For the solution of the stochastic Galerkin and collocation systems, we used preconditioned conjugate gradient (PCG). To ensure that we are fair to both methods and that we do not over-resolve either system, we set the tolerance in the solvers to be $\|u_{ex} - u^{SG}_{J_h M_p}\|/10$ and $\|u_{ex} - u^{SC}_{J_h M_L}\|/10$ respectively. Figure 4.5.3(a) displays the convergence of the stochastic Galerkin and collocation methods against the total number of stochastic degrees of freedom (SDOF) for both methods. For the stochastic Galerkin method, we take the SDOF to be the dimension of the stochastic spectral polynomial basis M_p for a given solution projection order. For the stochastic collocation method, we take the SDOF to be the total number of sample points M_L used to obtain the solution at a given level. For the Galerkin case, we considered two polynomial subspaces, described in Section 4.2. In particular, we project the SG approximation onto both the total degree subspace and the sparse Smolyak subspace, using an orthonormal expansion in the Legendre basis. For this particular example, the Smolyak subspace is impractical due to the fast growth of the SDOF. In the collocation approximation, we approximate $\mathscr{I}^{m,g}_L[u_{J_h}]$ using m and g defined by (4.4.9) using the Clenshaw–Curtis (CC),



(a)



(b)

Figure 4.5.3. For $\Gamma = [0,1]^8$ so that $N = 8$ and for correlation length $C = 1/64$, (a) convergence of the gSGM and gSCM versus the stochastic degrees of freedom (DOF), and (b) convergence of the gSGM and gSCM versus the total computational cost. The SG approximation uses projections onto the isotropic total degree (TD) and sparse Smolyak (SS) polynomial subspaces spanned by the Legendre polynomials. For the sparse grid SC approximation, the Gauss–Legendre (GL), Clenshaw–Curtis (CC), and slow-growth CC (sCC) sparse grid points are used.

Gauss–Legendre (GL), and the slow-growth CC (sCC) sparse grid points. As we expect, as the tolerance increases, the gSGM approximations in the TD subspace require less SDOF than any of the gSCM approximations.

However, Figure 4.5.3(b) compares the total computational cost as described above. The results reveal that all three sparse grid gSCM approximations dramatically outperform the gSGM, even the widely used gSGM based on TD subspaces. Of course, this is for one particular example and choice of stochastic input data. However, as the problems become ever more nonlinear, we expect the results to become even more favourable to the gSCM.

PART FIVE
Local piecewise polynomial stochastic approximation

To realize their high accuracy, the stochastic Galerkin and stochastic collocation methods in Part 4, based on the use of global polynomials as discussed, require high regularity of the solution $u(\mathbf{x}, \boldsymbol{y})$ with respect to the random parameters $\{y_n\}_{n=1}^N$. They are therefore ineffective for the approximation of solutions that have irregular dependence with respect to those parameters. Motivated by finite element methods (FEMs) for spatial approximation, an alternative and potentially more effective approach for approximating irregular solutions is to use *locally supported piecewise polynomial* approaches for approximating the solution dependence on the random parameters. To achieve greater accuracy, global polynomial approaches increase the polynomial degree; piecewise polynomial approaches instead keep the polynomial degree fixed but refine the grid used to define the approximation space.

To set the stage, in Section 5.1, we use standard FEMs commonly used for spatial approximation and apply them to parameter space approximation. We show that such approaches are especially vulnerable to the curse of dimensionality, so we then consider more judicious choices of piecewise polynomial bases.

5.1. Stochastic Galerkin methods with piecewise polynomial bases

In this section, we consider the use of standard FEMs with locally supported piecewise polynomial bases for approximation with respect to the parameters $\boldsymbol{y}$. Spatial discretization is effected by using a finite element space defined on D consisting of continuous piecewise polynomial functions on a conforming triangulation $\mathcal{T}_h$ of D with maximum mesh size $h > 0$. Because both spatial and parameter discretizations utilize piecewise

polynomial bases, such methods can be viewed as direct extensions of standard FEMs to the product domain $D \times \Gamma$. For each $\boldsymbol{y} \in \Gamma$, the spatial discretization error of the semi-discrete solution $u_{J_h}(\mathbf{x}, \boldsymbol{y})$ can be estimated using standard FEM error analyses. For example, for second-order elliptic PDEs with homogeneous Dirichlet boundary conditions, under standard assumptions on the spatial domain D and the data, the spatial discretization error is given by (3.3.2).

To define a finite element space with respect to the parameters $\boldsymbol{y} \in \Gamma$, we start by partitioning the domain Γ. For simplicity, we assume Γ is bounded. For a PDF with unbounded support, for example, a Gaussian PDF, an appropriate truncation can be applied such that the integral of the PDF over the domain exterior to Γ is much smaller than the desired error of the approximate solution. Without further loss of generality, we assume Γ is the hypercube $[-1, 1]^N$. Then, for a prescribed grid size[8] $\widetilde{h}$, a partition $\mathcal{T}_{\widetilde{h}}$ of the parameter domain Γ into the finite number of disjoint, covering N-dimensional boxes $\gamma_{\widetilde{m}} = \prod_{n=1}^{N} [a_n^{\widetilde{m}}, b_n^{\widetilde{m}}]$ with $\widetilde{m} = 1, \ldots, \widetilde{M}$ is defined; we have $\Gamma = \bigcup_{\widetilde{m}=1}^{\widetilde{M}} \gamma_{\widetilde{m}}$ with the number of elements $\widetilde{M} = (2/\widetilde{h})^N$. Then, a finite element subspace $Z_M \subset L^2_\rho(\Gamma)$ consisting of piecewise polynomial functions of degree less than or equal to p is defined with respect to the partition $\mathcal{T}_{\widetilde{h}}$.

If functions belonging to Z_M are continuous on Γ, basis functions can be chosen that have support over a small number of the elements $\gamma_{\widetilde{m}}$. In this case, the dimension M of Z_M, that is, the number of parameter degrees of freedom, is given by[9]

$$M = \left(p\widetilde{M}^{1/N} + 1\right)^N = \left(p\frac{2}{\widetilde{h}} + 1\right)^N. \tag{5.1.1}$$

Alternatively, if functions belonging to Z_M are discontinuous across the faces of the elements $\gamma_{\widetilde{m}}$, then the basis functions can be chosen to have support over only a single element. In this case, we have that the number of parameter degrees of freedom is given by

$$M = \widetilde{M}\left(\frac{(N+p)!}{N!p!}\right) = \left(\frac{2}{\widetilde{h}}\right)^N \left(\frac{(N+p)!}{N!p!}\right). \tag{5.1.2}$$

According to (5.1.2), the number of degrees of freedom associated with the element $\gamma_{\widetilde{m}}$, $\widetilde{m} = 1, \ldots, \widetilde{M}$, is given by $M_{\widetilde{m}} := (N+p)!/(N!p!)$ and the

[8] Of course, $\widetilde{h}$ is chosen so that $2/\widetilde{h}$ is an integer; $\widetilde{h}^{-1}$ a power of 2 is a common choice. Also, there are no additional difficulties (other than more complicated notation) engendered by the use of non-uniform grid spacings in each parameter direction.

[9] Here, we assume that the finite element space Z_M is a Lagrange finite element space, *i.e.*, a finite element space for which the degrees of freedom are nodal values.

index set of the basis functions corresponding to that element is given by

$$I_{\widetilde{m}} = \big\{m = 1, \dots, M \mid \operatorname{supp}(\psi_m(\boldsymbol{y})) \subset \gamma_{\widetilde{m}}\big\} \quad \text{for } \widetilde{m} = 1, \dots, \widetilde{M};$$

the cardinality of $I_{\widetilde{m}}$ is $M_{\widetilde{m}}$. Then, due to the fact that

$$\psi_{\widetilde{m}'}(\boldsymbol{y}) = 0 \quad \text{for } \widetilde{m}' \in I_{\widetilde{m}} \text{ and } \boldsymbol{y} \notin \gamma_{\widetilde{m}},$$

the coupled $J_h M \times J_h M$ system (2.4.3) uncouples into $\widetilde{M}$ systems, each of size $J_h M_{\widetilde{m}} \times J_h M_{\widetilde{m}}$, that is, for each $\widetilde{m} = 1, \dots, \widetilde{M}$, we have the system

$$\begin{aligned}\int_D \int_{\gamma_{\widetilde{m}}} \rho(\boldsymbol{y}) S\Big(\sum_{j=1}^{J_h} \sum_{\widetilde{m} \in I_{\widetilde{m}}} c_{j\widetilde{m}} \phi_j(\mathbf{x}) \psi_{\widetilde{m}}(\boldsymbol{y}), \boldsymbol{y}\Big) T\big(\phi_{j'}(\mathbf{x})\big) \psi_{\widetilde{m}'}(\boldsymbol{y}) \, \mathrm{d}\mathbf{x} \, \mathrm{d}\boldsymbol{y} \\ = \int_D \int_{\gamma_{\widetilde{m}}} \rho(\boldsymbol{y}) \phi_{j'}(\mathbf{x}) \psi_{\widetilde{m}'}(\boldsymbol{y}) f(\boldsymbol{y}) \, \mathrm{d}\mathbf{x} \, \mathrm{d}\boldsymbol{y} \end{aligned} \tag{5.1.3}$$

for $j = 1, \dots, J_h$ and $\widetilde{m}' \in I_{\widetilde{m}}$.

If Z_M is chosen as the piecewise constant finite element space with respect to $\mathcal{T}_{\widetilde{h}}$, that is, $p = 0$, then $M = \widetilde{M}$ and the single basis function corresponding to the $\widetilde{m}$th element is given by

$$\psi_{\widetilde{m}}(\boldsymbol{y}) = \begin{cases} 1 & \text{if } \boldsymbol{y} \in \gamma_{\widetilde{m}}, \\ 0 & \text{otherwise.} \end{cases}$$

If, to approximate the integrals with respect to Γ appearing in (2.4.3), we choose an M-point quadrature rule $\{\boldsymbol{y}_{\widetilde{m}}, w_{\widetilde{m}}\}_{\widetilde{m}=1}^{M}$ such that each element $\gamma_{\widetilde{m}}$, $\widetilde{m} = 1, \dots, \widetilde{M} = M$, contains one and only one of the quadrature points $\{\boldsymbol{y}_{\widetilde{m}}\}_{\widetilde{m}=1}^{M}$, we have that $M_{\widetilde{m}} = 1$, $I_{\widetilde{m}} = \widetilde{m}$, and

$$\psi_{\widetilde{m}}(\boldsymbol{y}_{\widetilde{m}'}) = \delta_{\widetilde{m}\widetilde{m}'} \quad \text{for } \widetilde{m}, \widetilde{m}' = 1, \dots, \widetilde{M} = M. \tag{5.1.4}$$

As a result, with

$$u_{J_h}(\mathbf{x}, \boldsymbol{y}_{\widetilde{m}}) = \sum_{j=1}^{J_h} c_{j\widetilde{m}} \phi_j(\mathbf{x}) \quad \text{for } \widetilde{m} = 1, \dots, \widetilde{M} = M,$$

the decoupled SFEM system (5.1.3) for the element $\gamma_{\widetilde{m}}$, $\widetilde{m} = 1, \dots, \widetilde{M} = M$, reduces to the single $J_h \times J_h$ deterministic finite element system

$$\int_D S\big(u_{J_h}(\mathbf{x}, \boldsymbol{y}_{\widetilde{m}}), \boldsymbol{y}_{\widetilde{m}}\big) T\big(\phi_{j'}(\mathbf{x})\big) \, \mathrm{d}\mathbf{x} = \int_D \phi_{\widetilde{j'}}(\mathbf{x}) f(\boldsymbol{y}_{\widetilde{m}}) \, \mathrm{d}\mathbf{x},$$

for $j' = 1, \dots, J_h$, from which the coefficients $c_{j\widetilde{m}}$, $j = 1, \dots, J_h$ are determined. Thus, we have a total uncoupling of the spatial and parameter degrees of freedom, that is, we merely have to solve a *sequence* of $M = \widetilde{M}$ deterministic finite element problems of size $J_h \times J_h$ to determine $\{u_{J_h}(\mathbf{x}, \boldsymbol{y}_{\widetilde{m}})\}_{\widetilde{m}=1}^{\widetilde{M}}$, that is, to determine all the coefficients c_{jm}, $j = 1 \dots, J_h$

and $m = 1, \ldots, M$ appearing in the fully discrete approximation (2.4.1). Clearly, this is an example of another stochastic sampling method.

Although this approach is clearly straightforward to implement using existing deterministic FEM software as a black box, in practice it is useful only for problems having a small number of random input parameters, that is, only if the parameter dimension N is small. In fact, for both continuous and discontinuous bases, the degrees of freedom given in (5.1.1) and (5.1.2), respectively, increase exponentially as the N increases. For example, for the piecewise constant case just discussed, the quadrature or sample points $\{\boldsymbol{y}_{\widetilde{m}}\}_{\widetilde{m}=1}^{M}$ form a tensor product grid, that is, an $M = (2/\widetilde{h})^N$-point Cartesian grid with $2/\widetilde{h}$ points in each parameter direction. As a means to alleviate the curse of dimensionality while retaining the decoupling property that leads to (5.1.3), we next discuss a hierarchical sparse grid stochastic collocation method based on piecewise hierarchical bases integrated into the SFEM framework.

5.2. Hierarchical stochastic collocation methods

We now introduce several types of one-dimensional piecewise hierarchical polynomial bases (Bungartz and Griebel 2004, Griebel 1998) which are the foundation of hierarchical sparse grid stochastic collocation methods.

5.2.1. One-dimensional piecewise linear hierarchical interpolation

We begin with the one-dimensional hat function having support $[-1, 1]$ defined by

$$\psi(y) = \max\{0\,,\, 1 - |y|\},$$

from which an arbitrary hat function with support $(y_{l,i} - \widetilde{h}_l, y_{l,i} + \widetilde{h}_l)$ can be generated by dilation and translation, that is,

$$\psi_{l,i}(y) := \psi\left(\frac{y + 1 - i\widetilde{h}_l}{\widetilde{h}_l}\right),$$

where l denotes the resolution level, $\widetilde{h}_l = 2^{-l+1}$ for $l = 0, 1, \ldots$ denotes the grid size of the level l grid for the interval $[-1, 1]$, and $y_{l,i} = i\,\widetilde{h}_l - 1$ for $i = 0, 1, \ldots, 2^l$ denotes the grid points of that grid. The basis function $\psi_{l,i}(y)$ has local support and is centred at the grid point $y_{l,i}$; the number of grid points in the level l grid is $2^l + 1$.

With $Z = L^2_\rho(\Gamma)$, a sequence of subspaces $\{Z_l\}_{l=0}^{\infty}$ of Z of increasing dimension $2^l + 1$ can be defined as

$$Z_l = \operatorname{span}\{\psi_{l,i}(y) \mid i = 0, 1, \ldots, 2^l\} \quad \text{for } l = 0, 1, \ldots.$$

The sequence is dense in Z, that is, $\cup_{l=0}^{\infty} Z_l = Z$, and nested:

$$Z_0 \subset Z_1 \subset \cdots \subset Z_l \subset Z_{l+1} \subset \cdots \subset Z.$$

Each of the subspaces $\{Z_l\}_{l=0}^{\infty}$ is the standard finite element subspace of continuous piecewise linear polynomial functions on $[-1, 1]$ that is defined with respect to the grid having grid size $\widetilde{h}_l$. The set $\{\psi_{l,i}(y)\}_{i=0}^{2^l}$ is the standard nodal basis for the space Z_l.

An alternative to the nodal basis $\{\psi_{l,i}(y)\}_{i=0}^{2^l}$ for Z_l is a *hierarchical* basis, which we now construct, starting with the hierarchical index sets

$$B_l = \big\{ i \in \mathbb{N} \;\big|\; i = 1, 3, 5, \ldots, 2^l - 1 \big\} \quad \text{for } l = 1, 2, \ldots$$

and the sequence of hierarchical subspaces defined by

$$W_l = \operatorname{span}\big\{ \psi_{l,i}(y) \mid i \in B_l \big\} \quad \text{for } l = 1, 2, \ldots.$$

Due to the nesting property of $\{Z_l\}_{l=0}^{\infty}$, we have that $Z_l = Z_{l-1} \oplus W_l$ and $W_l = Z_l / \oplus_{l'=0}^{l-1} Z_{l'}$ for $l = 1, 2, \ldots$. We also have the hierarchical subspace splitting of Z_l given by

$$Z_l = Z_0 \oplus W_1 \oplus \cdots \oplus W_l \quad \text{for } l = 1, 2, \ldots.$$

Then, the *hierarchical basis* for Z_l is given by

$$\big\{ \psi_{0,0}(y), \psi_{0,1}(y) \big\} \cup \big(\cup_{l'=1}^{l} \{ \psi_{l',i}(y) \}_{i \in B_{l'}} \big). \tag{5.2.1}$$

It is easy to verify that, for each l, the subspaces spanned by the hierarchical and the nodal basis bases are the same, that is, they are both bases for Z_l.

The nodal basis $\{\psi_{L,i}(y)\}_{i=0}^{2^L}$ for Z_L possesses the delta property, that is, $\psi_{L,i}(y_{L,i'}) = \delta_{i,i'}$ for $i, i' \in \{0, \ldots, 2^L\}$. The hierarchical basis (5.2.1) for Z_L possesses only a partial delta property; specifically, the basis functions corresponding to a specific level possess the delta property with respect to its own level and coarser levels, but not with respect to finer levels, that is, for $l = 0, 1, \ldots, L$ and $i \in B_l$ we have

$$\begin{aligned}
&\text{for } 0 \le l' < l, \quad \psi_{l,i}(y_{l',i'}) = 0 \qquad \text{for all } i' \in B_{l'}, \\
&\text{for } l' = l, \qquad\quad \psi_{l,i}(y_{l,i'}) = \delta_{i,i'} \quad \text{for all } i' \in B_{l'}, \\
&\text{for } l < l' \le L, \quad \psi_{l,i}(y_{l',i'}) \neq 0 \qquad \text{for all } i' \in B_{l'}.
\end{aligned} \tag{5.2.2}$$

A comparison between the linear hierarchical polynomial basis and the corresponding nodal basis for $L = 3$ is given in Figure 5.2.1.

For each grid level l, the interpolant of a function $g(y)$ in the subspace Z_l in terms of the its nodal basis $\{\psi_{l,i}(y)\}_{i=0}^{2^l}$ is given by

$$\mathcal{I}_l\big(g(y)\big) = \sum_{i=0}^{2^l} g(y_{l,i}) \psi_{l,i}(y). \tag{5.2.3}$$

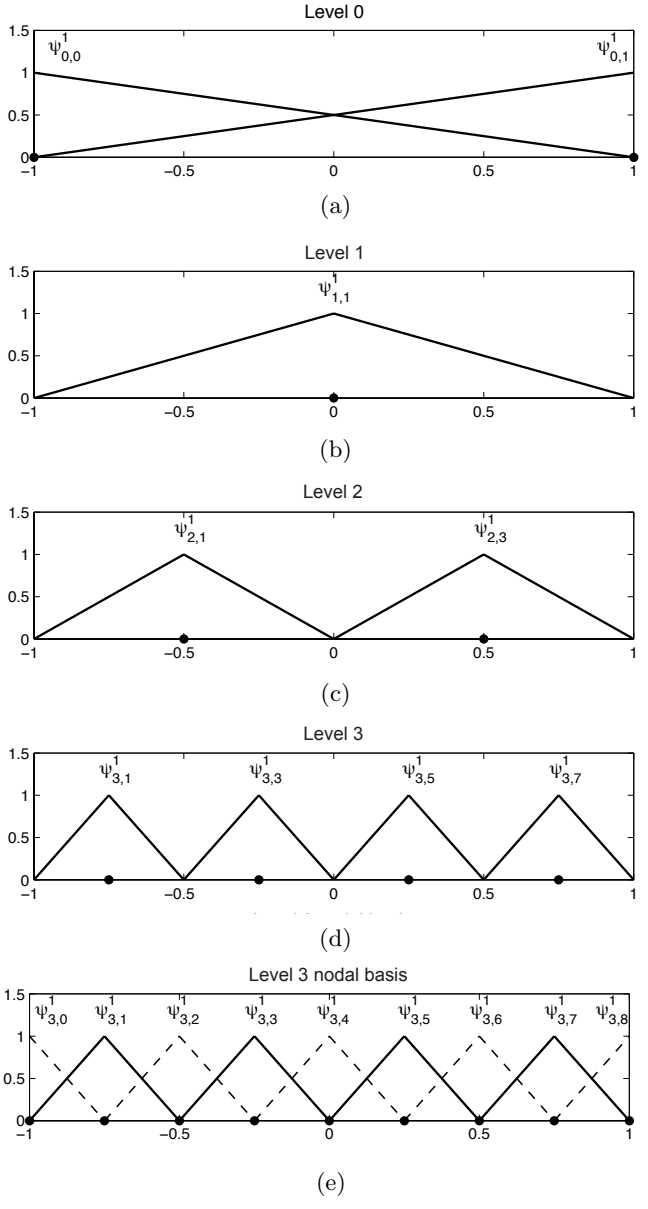


Figure 5.2.1. Piecewise linear polynomial bases for $L = 3$. (a–d) The basis functions for Z_0, W_1, W_2, and W_3, respectively. The hierarchical basis for Z_3 is the union of the functions in (a–d). (e) The nodal basis for Z_3.

Due to the nesting property $Z_l = Z_{l-1} \oplus W_l$, it is easy to see that $\mathcal{I}_{l-1}(g) = \mathcal{I}_l\big(\mathcal{I}_{l-1}(g)\big)$, based on which we define the incremental interpolation operator

$$\begin{aligned}\Delta_l(g) &= \mathcal{I}_l(g) - \mathcal{I}_{l-1}(g) = \mathcal{I}_l\big(g - \mathcal{I}_{l-1}(g)\big) \\ &= \sum_{i=0}^{2^l} \big[g(y_{l,i}) - \mathcal{I}_{l-1}\big(g\big)(y_{l,i})\big]\psi_{l,i}(y) \\ &= \sum_{i\in B_l} \big[g(y_{l,i}) - \mathcal{I}_{l-1}\big(g\big)(y_{l,i})\big]\psi_{l,i}(y) = \sum_{i\in B_l} c_{l,i}\psi_{l,i}(y),\end{aligned} \tag{5.2.4}$$

where $c_{l,i} = g(y_{l,i}) - \mathcal{I}_{l-1}\big(g(y_{l,i})\big)$. Note that $\Delta_l(g)$ only involves the basis functions for W_l for $l \geq 1$. Because $\Delta_l(g)$ essentially approximates the difference between g and the interpolant $\mathcal{I}_{l-1}(g)$ on level $l-1$, the coefficients $\{c_{l,i}\}_{i\in B_l}$ are referred to as the *surpluses* on level l.

The interpolant $\mathcal{I}_l(g)$ for any level $l > 0$ can be decomposed in the form

$$\mathcal{I}_l(g) = \mathcal{I}_{l-1}(g) + \Delta_l(g) = \cdots = \mathcal{I}_0(g) + \sum_{l'=1}^{l} \Delta_{l'}(g). \tag{5.2.5}$$

The delta property of the nodal basis implies that the interpolation matrix is diagonal. The interpolation matrix for the hierarchical basis is not diagonal, but the partial delta property (5.2.2) implies that it is triangular, so the coefficients in the interpolant can be solved for explicitly. This can also be seen from the definition (5.2.4) for $\Delta_l(\cdot)$ and the recursive form of $\mathcal{I}_l(\cdot)$ in (5.2.5), for which the surpluses can be computed explicitly.

5.2.2. Multi-dimensional hierarchical sparse grid interpolation

We now consider the interpolation of a multivariate function $g(\boldsymbol{y})$ defined, again without loss of generality, over the unit hypercube $\Gamma = [-1, 1]^N \subset \mathbb{R}^N$. The one-dimensional hierarchical polynomial basis (5.2.1) can be extended to the N-dimensional parameter domain Γ using tensorization. Specifically, the N-variate basis function $\psi_{\mathbf{l},\mathbf{i}}(\boldsymbol{y})$ associated with the point $\boldsymbol{y}_{\mathbf{l},\mathbf{i}} = (y_{l_1,i_1}, \ldots, y_{l_N,i_N})$ is defined using tensor products, that is,

$$\psi_{\mathbf{l},\mathbf{i}}(\boldsymbol{y}) := \prod_{n=1}^{N} \psi_{l_n,i_n}(y_n),$$

where $\{\psi_{l_n,i_n}(y_n)\}_{n=1}^N$ are the one-dimensional hierarchical polynomials associated with the point $y_{l_n,i_n} = i_n\widetilde{h}_{l_n} - 1$ with $\widetilde{h}_{l_n} = 2^{-l_n+1}$ and $\mathbf{l} = (l_1, \ldots, l_N)$ is a multi-index indicating the resolution level of the basis function. The N-dimensional hierarchical incremental subspace $W_{\mathbf{l}}$ is defined by

$$W_{\mathbf{l}} = \bigotimes_{n=1}^{N} W_{l_n} = \operatorname{span}\{\psi_{\mathbf{l},\mathbf{i}}(\boldsymbol{y}) \mid \mathbf{i} \in B_{\mathbf{l}}\},$$

where the multi-index set $B_{\mathbf{l}}$ is given by

$$B_{\mathbf{l}} := \left\{ \mathbf{i} \in \mathbb{N}^N \;\middle|\; \begin{array}{ll} i_n \in \{1,3,5,\ldots,2^{l_n}-1\} & \text{for } n=1,\ldots,N \quad \text{if } l_n > 0 \\ i_n \in \{0,1\} & \text{for } n=1,\ldots,N \quad \text{if } l_n = 0 \end{array} \right\}.$$

Similar to the one-dimensional case, a sequence of subspaces, again denoted by $\{Z_l\}_{l=0}^{\infty}$, of the space $Z := L^2_\rho(\Gamma)$ can be constructed as

$$Z_l = \bigoplus_{l'=0}^{l} W_{l'} = \bigoplus_{l'=0}^{l} \bigoplus_{\alpha(\mathbf{l}')=l'} W_{\mathbf{l}'},$$

where the key is how the mapping $\alpha(\mathbf{l})$ is defined because it defines the incremental subspaces $W_{l'} = \oplus_{\alpha(\mathbf{l}')=l'} W_{\mathbf{l}'}$. For example, $\alpha(\mathbf{l}) = \max_{n=1,\ldots,N} l_n$ leads to a full tensor product space whereas $\alpha(\mathbf{l}) = |\mathbf{l}| = l_1 + \cdots + l_N$ leads to a sparse polynomial space. As discussed in Section 5.1, because the full tensor product space suffers dramatically from the curse of dimensionality as N increases, this choice is not feasible for even moderately high-dimensional problems. Thus, we only consider the sparse polynomial space obtained by setting $\alpha(\mathbf{l}) = |\mathbf{l}|$.

The level l hierarchical sparse grid interpolant of the multivariate function $g(\boldsymbol{y})$ is then given by

$$\begin{aligned} g_l(\boldsymbol{y}) &:= \sum_{l'=0}^{l} \sum_{|\mathbf{l}'|=l'} (\Delta_{l'_1} \otimes \cdots \otimes \Delta_{l'_N}) g(\boldsymbol{y}) \qquad (5.2.6) \\ &= g_{l-1}(\boldsymbol{y}) + \sum_{|\mathbf{l}'|=l} (\Delta_{l'_1} \otimes \cdots \otimes \Delta_{l'_N}) g(\boldsymbol{y}) \\ &= g_{l-1}(\boldsymbol{y}) + \sum_{|\mathbf{l}'|=l} \sum_{\mathbf{i} \in B_{\mathbf{l}'}} \left[g(\boldsymbol{y}_{\mathbf{l}',\mathbf{i}}) - g_{l'-1}(\boldsymbol{y}_{\mathbf{l}',\mathbf{i}}) \right] \psi_{\mathbf{l}',\mathbf{i}}(\boldsymbol{y}) \\ &= g_{l-1}(\boldsymbol{y}) + \sum_{|\mathbf{l}'|=l} \sum_{\mathbf{i} \in B_{\mathbf{l}'}} c_{\mathbf{l}',\mathbf{i}}\, \psi_{\mathbf{l}',\mathbf{i}}(\boldsymbol{y}), \end{aligned}$$

where $c_{\mathbf{l}',\mathbf{i}} = g(\boldsymbol{y}_{\mathbf{l}',\mathbf{i}}) - g_{l'-1}(\boldsymbol{y}_{\mathbf{l}',\mathbf{i}})$ is the multi-dimensional hierarchical surplus. This interpolant is a direct extension, via the Smolyak algorithm, of the one-dimensional hierarchical interpolant. Analogous to (5.2.4), the definition of the surplus $c_{\mathbf{l}',\mathbf{i}}$ is based on the facts that $g_l(g_{l-1}(\boldsymbol{y})) = g_{l-1}(\boldsymbol{y})$ and $g_{l-1}(\boldsymbol{y}_{\mathbf{l}',\mathbf{i}}) - g(\boldsymbol{y}_{\mathbf{l}',\mathbf{i}}) = 0$ for $|\mathbf{l}'| = l$. In this case, we denote by $\mathcal{H}_{\mathbf{l}}(\Gamma) = \{\boldsymbol{y}_{\mathbf{l},\mathbf{i}} \mid \mathbf{i} \in B_{\mathbf{l}}\}$ the set of sparse grid points corresponding to subspace $W_{\mathbf{l}}$. Then, the sparse grid corresponding to the interpolant g_l is given by

$$\mathcal{H}_l^N(\Gamma) = \cup_{l'=0}^{l} \cup_{|\mathbf{l}'|=l'} \mathcal{H}_{\mathbf{l}'}(\Gamma).$$

We have that $\mathcal{H}_l^N(\Gamma)$ is also nested, *i.e.*, $\mathcal{H}_{l-1}^N(\Gamma) \subset \mathcal{H}_l^N(\Gamma)$. In Figure 5.2.2 we plot the structure of a level $l = 2$ sparse grid in $N = 2$ dimensions,

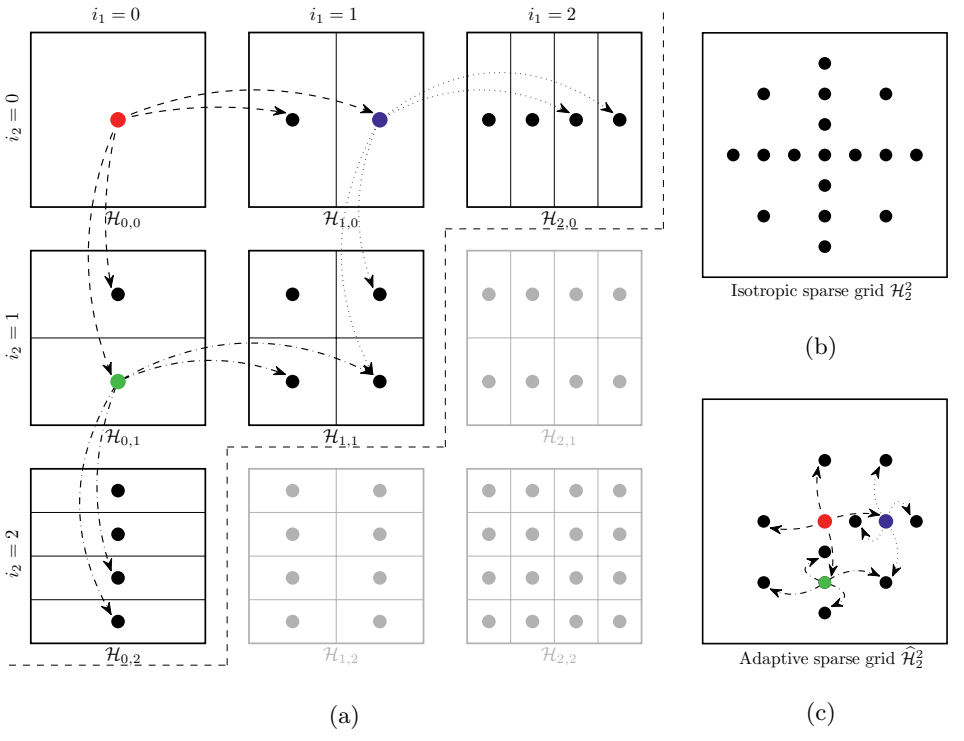


Figure 5.2.2. (a) Nine tensor product subgrids for level $l = 0, 1, 2$ of which only the six subgrids for which $l_1' + l_2' \leq l = 2$ are chosen to appear in the level $l = 2$ isotropic sparse grid $\mathcal{H}_2^2(\Gamma)$ (b) containing 17 points. With adaptivity, only points that correspond to a large surplus lead to two child points added in each direction, resulting in the adaptive sparse grid $\widetilde{\mathcal{H}}_2^2(\Gamma)$ (c) containing 12 points.

without consideration of boundary points. The nine subgrids $\mathcal{H}_{\mathbf{l}'}(\Gamma)$ in Figure 5.2.2(a) correspond to the nine multi-index sets $B_{\mathbf{l}'}$, where

$$\mathbf{l}' \in \{(0,0),(0,1),(0,2),(1,0),(1,1),(1,2),(2,0),(2,1),(2,2)\}.$$

The level $l = 2$ sparse grid $\mathcal{H}_2^2(\Gamma)$ in Figure 5.2.2(b) includes only six of the nine subgrids, with the three subgrids depicted in grey not included because they fail the criterion $|\mathbf{l}'| \leq l' = 2$. Moreover, due to the nesting property of the hierarchical basis, $\mathcal{H}_2^2(\Gamma)$ has only 17 points, as opposed to the 49 points of the full tensor product grid.

5.2.3. Multi-dimensional hierarchical sparse grid interpolation

We now use the hierarchical sparse grid interpolation method of Section 5.2.2 to approximate the parameter dependence of the solution $u(\mathbf{x}, \boldsymbol{y})$ of an SPDE. Specifically, the basis $\{\psi_m(\boldsymbol{y})\}_{m=1}^M$ entering into the fully discrete

approximation (2.4.1) is chosen to be the hierarchical basis defined in Section 5.2.2. Here, we use the indexing of that section because it more easily handles the hierarchical nature of the hierarchical basis. In this case, the fully discrete approximate solution takes the form

$$u_{J_h M_L}(\mathbf{x}, \boldsymbol{y}) = \sum_{l=0}^{L} \sum_{|\mathbf{l}|=l} \sum_{\mathbf{i} \in B_{\mathbf{l}}} c_{\mathbf{l},\mathbf{i}}(\mathbf{x}) \psi_{\mathbf{l},\mathbf{i}}(\boldsymbol{y}), \tag{5.2.7}$$

where the coefficients are now functions of $\mathbf{x}$ to reflect that dependence of the function $u_{J_h M_L}(\mathbf{x}, \boldsymbol{y})$. In the usual manner, those coefficients are given in terms of the spatial finite element basis $\{\phi_j(\mathbf{x})\}_{j=1}^{J_h}$ by $c_{\mathbf{l},\mathbf{i}}(\mathbf{x}) = \sum_{j=1}^{J_h} c_{j,\mathbf{l},\mathbf{i}} \phi_j(\mathbf{x})$ so that, from (5.2.7), we obtain

$$\begin{aligned} u_{J_h M_L}(\mathbf{x}, \boldsymbol{y}) &= \sum_{l=0}^{L} \sum_{|\mathbf{l}|=l} \sum_{\mathbf{i} \in B_{\mathbf{l}}} \left(\sum_{j=1}^{J_h} c_{j,\mathbf{l},\mathbf{i}} \phi_j(\mathbf{x}) \right) \psi_{\mathbf{l},\mathbf{i}}(\boldsymbol{y}) \\ &= \sum_{j=1}^{J_h} \left(\sum_{l=0}^{L} \sum_{|\mathbf{l}|=l} \sum_{\mathbf{i} \in B_{\mathbf{l}}} c_{j,\mathbf{l},\mathbf{i}} \psi_{\mathbf{l},\mathbf{i}}(\boldsymbol{y}) \right) \phi_j(\mathbf{x}). \end{aligned} \tag{5.2.8}$$

The number of parameter degrees of freedom M_L of $u_{J_h M_L}$ is equal to the number of the grid points of the sparse grid $\mathcal{H}_L^N(\Gamma)$.

We next explain how the coefficients $c_{j,\mathbf{l},\mathbf{i}}$ in (5.2.8) are determined. In general, after running the deterministic FEM solver for all the sparse grid points, we obtain the dataset

$$J_h(\mathbf{x}_j, \boldsymbol{y}_{\mathbf{l},\mathbf{i}}) \quad \text{for } j = 1, \ldots, J_h \text{ and } |\mathbf{l}| \le L, \ \mathbf{i} \in B_{\mathbf{l}}.$$

Then, it is easy to see from (5.2.8) that, for fixed j, $\{c_{j,\mathbf{l},\mathbf{i}}\}_{|\mathbf{l}| \le L, \mathbf{i} \in B_{\mathbf{l}}}$ can be obtained by solving the linear system

$$\begin{aligned} u_{J_h M_L}(\mathbf{x}_j, \boldsymbol{y}_{\mathbf{l}',\mathbf{i}'}) &= \sum_{l=0}^{L} \sum_{|\mathbf{l}|=l} \sum_{\mathbf{i} \in B_{\mathbf{l}}} c_{j,\mathbf{l},\mathbf{i}} \psi_{\mathbf{l},\mathbf{i}}(\boldsymbol{y}_{\mathbf{l}',\mathbf{i}'}) \\ &= u_{J_h}(\mathbf{x}_j, \boldsymbol{y}_{\mathbf{l}',\mathbf{i}'}) \quad \text{for } |\mathbf{l}'| \le L, \ \mathbf{i}' \in B_{\mathbf{l}}. \end{aligned} \tag{5.2.9}$$

Thus, the approximation $u_{J_h M_L}(\mathbf{x}, \boldsymbol{y})$ can be obtained by solving J_h linear systems. However, because the hierarchical bases $\psi_{\mathbf{l},\mathbf{i}}(\boldsymbol{y})$ satisfy $\psi_{\mathbf{l},\mathbf{i}}(\boldsymbol{y}_{\mathbf{l}',\mathbf{i}'}) = 0$ if $l' \le l$ (this is a consequence of the one-dimensional partial delta property), the coefficient $c_{j,\mathbf{l}',\mathbf{i}'}$ in the the system (5.2.9) corresponding to the sparse grid point $\boldsymbol{y}_{\mathbf{l}',\mathbf{i}'}$ on level L, that is, for $|\mathbf{l}'| = L$, it reduces to

$$\begin{aligned} c_{j,\mathbf{l}',\mathbf{i}'} &= u_{J_h}(\mathbf{x}_j, \boldsymbol{y}_{\mathbf{l}',\mathbf{i}'}) - \sum_{l=0}^{L-1} \sum_{|\mathbf{l}|=l} \sum_{\mathbf{i} \in B_{\mathbf{l}}} c_{j,\mathbf{l},\mathbf{i}} \psi_{\mathbf{l},\mathbf{i}}(\boldsymbol{y}_{\mathbf{l}',\mathbf{i}'}) \\ &= u_{J_h}(\mathbf{x}_j, \boldsymbol{y}_{\mathbf{l}',\mathbf{i}'}) - u_{J_h M_{L-1}}(\mathbf{x}_j, \boldsymbol{y}_{\mathbf{l}',\mathbf{i}'}), \end{aligned} \tag{5.2.10}$$

so that the linear system becomes a triangular system and all the coefficients can be computed explicitly by recursively using (5.2.10). Note that (5.2.10) is consistent with the definition of the $c_{\mathbf{l},\mathbf{i}}(\mathbf{x})$ given in (5.2.6).

5.3. Adaptive hierarchical stochastic collocation method

By virtue of the hierarchical surpluses $c_{j,\mathbf{l},\mathbf{i}}$, the approximation in (5.2.8) can be represented in a hierarchical manner, that is,

$$u_{J_h M_L}(\mathbf{x}, \boldsymbol{y}) = u_{J_h M_{L-1}}(\mathbf{x}, \boldsymbol{y}) + \Delta u_{J_h M_L}(\mathbf{x}, \boldsymbol{y}), \tag{5.3.1}$$

where $u_{J_h M_{L-1}}(\mathbf{x}, \boldsymbol{y})$ is the sparse grid approximation in Z_{L-1} and

$$\Delta u_{J_h M_L}(\mathbf{x}, \boldsymbol{y})$$

is the hierarchical surplus interpolant in the subspace W_L. According to the analysis in Bungartz and Griebel (2004), for smooth functions, the surpluses $c_{j,\mathbf{l},\mathbf{i}}$ of the sparse grid interpolant $u_{J_h M_L}$ in (5.2.8) tend to zero as the interpolation level l goes to infinity. For example, in the context of using piecewise linear hierarchical bases and assuming the spatial approximation $u_{J_h}(\mathbf{x}, \boldsymbol{y})$ of the solution has bounded second-order weak derivatives with respect to $\boldsymbol{y}$, that is, $u_{J_h}(\mathbf{x}, \boldsymbol{y}) \in W_h(D) \otimes H^2_\rho(\Gamma)$, then the surplus $c_{j,\mathbf{l},\mathbf{i}}$ can be bounded as

$$|c_{j,\mathbf{l},\mathbf{i}}| \leq C 2^{-2\cdot|\mathbf{l}|} \quad \text{for } \mathbf{i} \in B_{\mathbf{l}} \text{ and } j = 1, \ldots, J_h, \tag{5.3.2}$$

where the constant C is independent of the level l. Furthermore, the smoother the target function is, the faster the surplus decays. This provides a good avenue for constructing adaptive sparse grid interpolants using the magnitude of the surplus as an error indicator, especially for irregular functions having, for example, steep slopes or jump discontinuities.

We first focus on the construction of one-dimensional adaptive grids and then extending the adaptivity to multi-dimensional sparse grids. As shown in Figure 5.3.1, the one-dimensional hierarchical grid points have a tree-like structure. In general, a grid point $y_{l,i}$ on level l has two children, namely $y_{l+1,2i-1}$ and $y_{l+1,2i+1}$ on level $l+1$. Special treatment is required when moving from level 0 to level 1, where we only add a single child $y_{1,1}$ on level 1. On each successive interpolation level, the basic idea of adaptivity is to use the hierarchical surplus as an error indicator to detect the smoothness of the target function and refine the grid by adding two new points on the next level for each point for which the magnitude of the surplus is larger than the prescribed error tolerance. For example, in Figure 5.3.1 we illustrate the six-level adaptive grid for interpolating the function $g(y) = \exp[-(y-0.4)^2/0.0625^2]$ on $[0,1]$ with error tolerance 0.01. From level 0 to level 2, because the magnitude of every surplus is larger than 0.01, two points are added for each grid point on levels 0 and 2; as

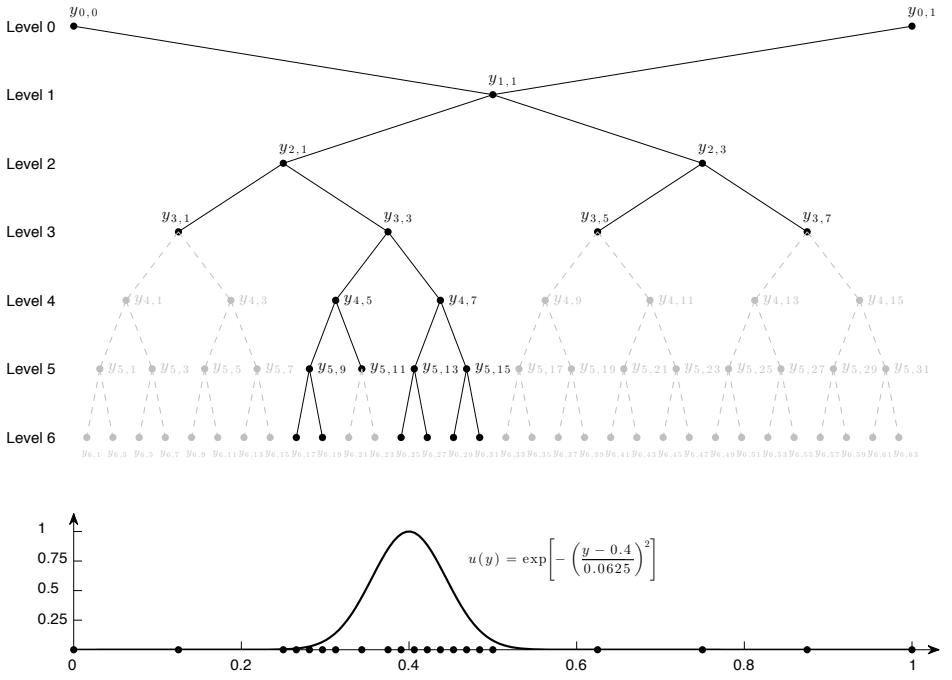


Figure 5.3.1. A six-level adaptive sparse grid for interpolating the one-dimensional function $g(y) = \exp[-(y-0.4)^2/0.0625^2]$ on $[0,1]$ with the error tolerance of 0.01. The resulting adaptive sparse grid has only 21 points (the black points) whereas the full grid has 65 points (the black and grey points).

mentioned above, only one point is added for each grid point on level 1. However, on level 3, there is only one point, namely $y_{3,3}$, whose surplus has magnitude larger than 0.01, so only two new points are added on level 4. If we continue through levels 5 and 6, we end up with the six-level adaptive grid with only 21 points (points in black in Figure 5.3.1), whereas the six-level non-adaptive grid has a total of 65 points (points in black and grey in Figure 5.3.1).

It is trivial to extend this adaptive approach from one dimension to a multi-dimensional adaptive sparse grid. In general, as shown in Figure 5.2.2, in N dimensions a grid point has $2N$ children which are also its neighbour points. However, note that the children of a parent point correspond to hierarchical basis functions on the next interpolation level, so we can build the interpolant $u_{J_h M_L}$ in (5.2.8) from level $L-1$ to level L by only adding those points on level L whose parents have surpluses greater than the prescribed tolerance. Because at each sparse grid point $\boldsymbol{y}_{\mathbf{l},\mathbf{i}}$ we have j surpluses $c_{j,\mathbf{l},\mathbf{i}}$, the error indicator is set to the maximum magnitude of the j

surpluses, that is, to $\max_{j=1,\ldots,J_h} |c_{j,\mathbf{l},\mathbf{i}}|$. In this way, we can refine the sparse grid locally, resulting in an adaptive sparse grid which is a subgrid of the corresponding isotropic sparse grid, as illustrated by Figure 5.2.2(c). The solution of the corresponding adaptive hSGSC approach is represented by

$$u^{\varepsilon}_{J_h M_L}(\mathbf{x}, \boldsymbol{y}) = \sum_{l=0}^{L} \sum_{|\mathbf{l}|=l} \sum_{\mathbf{i} \in B^{\varepsilon}_{\mathbf{l}}} \left(\sum_{j=1}^{J_h} c_{j,\mathbf{l},\mathbf{i}} \phi_j(\mathbf{x}) \right) \psi_{\mathbf{l},\mathbf{i}}(\boldsymbol{y}), \tag{5.3.3}$$

where the multi-index set $B^{\varepsilon}_{\mathbf{l}} \subset B_{\mathbf{l}}$ is defined by

$$B^{\varepsilon}_{\mathbf{l}} = \Big\{ \mathbf{i} \in B_{\mathbf{l}} \;\Big|\; \max_{j=1,\ldots,J_h} |c_{j,\mathbf{l},\mathbf{i}}| \geq \varepsilon \Big\}.$$

Note that $B^{\varepsilon}_{\mathbf{l}}$ is an optimal multi-index set that contains only the indices of the basis functions with surplus magnitudes larger than the tolerance ε. However, in practice, we also need to run the deterministic FEM solver at a certain number of grid points $\boldsymbol{y}_{\mathbf{l},\mathbf{i}}$ with $\max_{j=1,\ldots,J_h} |c_{j,\mathbf{l},\mathbf{i}}| < \varepsilon$ in order to detect when mesh refinement can stop. For example, in Figure 5.3.1, the points $y_{3,1}$, $y_{3,5}$, $y_{3,7}$, and $y_{5,11}$ are of this type. In this case, the number of degrees of freedom in (5.3.3) is usually smaller than the necessary number of executions of the deterministic FEM solver.

5.3.1. Relation between hierarchical stochastic collocation methods and stochastic Galerkin methods

For the hSGSC method, the fully discrete approximation is constructed according to the Lagrange interpolation rule, that is,

$$u_{J_h M_L}(\mathbf{x}, \boldsymbol{y}_m) = u_{J_h}(\mathbf{x}, \boldsymbol{y}_m) \quad \text{for } m = 1, \ldots, M_L.$$

Here, we show that hSGSC approximation also satisfies the variational form (2.4.3).

A quadrature rule is needed to approximate the integrals over Γ in the variational form (2.4.3). Here, we choose the rule $\{w_r, \boldsymbol{y}_r\}_{r=1}^{R}$ such that the quadrature points are the same as the sparse grid points. As such, the variational form (2.4.3) becomes the $J_h M_L \times J_h M_L$ system of equations

$$\begin{aligned} &\sum_{r=1}^{R} w_r \rho(\boldsymbol{y}_r) \psi_{m'}(\boldsymbol{y}_r) \\ &\qquad \times \int_D S\Big(\sum_{j=1}^{J_h} \sum_{m=1}^{M_L} c_{jm} \phi_j(\mathbf{x}) \psi_m(\boldsymbol{y}_r), \boldsymbol{y}_r \Big) T\big(\phi_{j'}(\mathbf{x})\big) \, \mathrm{d}\mathbf{x} \\ &\qquad\qquad = \sum_{r=1}^{R} w_r \rho(\boldsymbol{y}_r) \psi_{m'}(\boldsymbol{y}_r) \int_D \phi_{j'}(\mathbf{x})(\boldsymbol{y}_r) f(\mathbf{x}, \boldsymbol{y}_r) \, \mathrm{d}\mathbf{x}, \end{aligned} \tag{5.3.4}$$

for $j' \in \{1, \ldots, J_h\}$ and $m' \in \{1, \ldots, M_L\}$, and $\{w_r\}_{r=1}^{R}$ denotes a set of quadrature weights. Note that an appropriate quadrature rule is also needed to discretize the spatial integral over D, but we do not write it out explicitly because it is not germane to the current discussion.

In this case, that is, if $R = M_L$ and $\boldsymbol{y}_r = \boldsymbol{y}_m$ for $r = m$, it is easy to see that if c_{jm}, $j = 1, \ldots, J_h$ and $m = 1, \ldots, M_L$, satisfy

$$\int_D S\Big(\sum_{j=1}^{J_h}\sum_{m=1}^{M_L} c_{jm}\phi_j(\mathbf{x})\psi_m(\boldsymbol{y}_{m'}), \boldsymbol{y}_{m'}\Big)T\big(\phi_{j'}(\mathbf{x})\big)\,\mathrm{d}\mathbf{x} \tag{5.3.5}$$
$$= \int_D \phi_{j'}(\mathbf{x})\psi_m(\boldsymbol{y}_{m'})f(\mathbf{x}, \boldsymbol{y}_{m'})\,\mathrm{d}\mathbf{x},$$

for $j' \in \{1, \ldots, J_h\}$ and $m' \in \{1, \ldots, M_L\}$, then they also solve (5.3.4). If the system (5.3.4) has a unique solution, that solution is given by the solution of (5.3.5). Substituting

$$u_{jm'} = \sum_{m=1}^{M_L} c_{jm}\psi_m(\boldsymbol{y}_{m'}) \quad \text{for } j = 1, \ldots, J_h \text{ and } m' = 1, \ldots, M_L \tag{5.3.6}$$

into (5.3.5), we obtain

$$\int_D S\Big(\sum_{j=1}^{J_h} u_{jm'}\phi_j(\mathbf{x}), \boldsymbol{y}_{m'}\Big)T\big(\phi_{j'}(\mathbf{x})\big)\,\mathrm{d}\mathbf{x} \tag{5.3.7}$$
$$= \int_D \phi_{j'}(\mathbf{x})\psi_m(\boldsymbol{y}_{m'})f(\mathbf{x}, \boldsymbol{y}_{m'})\,\mathrm{d}\mathbf{x} \quad \text{for } j' \in \{1, \ldots, J_h\}$$

which is just the deterministic FEM problem at the point $\boldsymbol{y}_{m'}$ for $m' = 1, \ldots, M_L$ in parameter space. In order to compute $u_{jm'}$ for $j = 1, \ldots, J_h$ and $m' = 1, \ldots, M_L$, we need to solve M_L systems at each $y_{m'}$, each of size $J_h \times J_h$. After that, since there are only basis functions $\psi_m(\boldsymbol{y})$ involved in (5.3.6), for each $j \in \{1, \ldots, J_h\}$, $\{c_{km}\}_{m=1}^{M_L}$ can be obtained by substituting the values $\{u_{jm'}\}_{m'=1}^{M_L}$ into (5.3.6) and solving the linear system. By noting the fact that

$$u_{jm'} = u_{J_h}(\mathbf{x}_j, \boldsymbol{y}_{m'}),$$

it is easy to see that the system (5.3.6) is equivalent to the system (5.2.9) for computing the coefficients of $u_{J_h M_L}(\mathbf{x}, \boldsymbol{y})$. Therefore, the solution of the hSGSC method is also the solution of the variational form in (2.4.2).

Furthermore, with a proper reordering, the property (5.2.9) of the linear hierarchical basis gives rise to the property that

$$\psi_m(\boldsymbol{y}_{m'}) = 0 \quad \text{for } m > m', \tag{5.3.8}$$

and then the system (5.3.6) becomes

$$u_{jr} = \sum_{m=1}^{m'-1} c_{jm}\psi_m(\boldsymbol{y}_{m'}) + c_{jm'} \quad \text{for } j = 1, \ldots, J_h. \tag{5.3.9}$$

In this case, the resulting system becomes a lower triangular system so that all the coefficients c_{jr} can be computed explicitly. This is also consistent with the formula in (5.2.10).

5.3.2. Other choices of hierarchical basis

High-order hierarchical polynomial basis

One can generalize the piecewise linear hierarchical polynomials to high-order hierarchical polynomials (Bungartz and Griebel 2004). The goal is to construct polynomial basis functions of order p, denoted by $\psi_{l,i}^p(y)$, without enlarging the support $[y_{l,i} - \widetilde{h}_l, y_{l,i} + \widetilde{h}_l]$ or increasing the degrees of freedom in the support. As shown in Figure 5.2.1, for $l \geq 0$, a piecewise linear polynomial $\psi_{l,i}(y)$ is defined based on three supporting points, that is, $y_{l,i}$ and its two ancestors which are also the endpoints of the support $[y_{l,i} - \widetilde{h}_l, y_{l,i} + \widetilde{h}_l]$. For $p \geq 2$, it is well known that we need $p+1$ supporting points to define a Lagrange interpolating polynomial of order p. To achieve the goal, at each grid point $y_{l,i}$ we borrow additional ancestors outside $[y_{l,i} - \widetilde{h}_l, y_{l,i} + \widetilde{h}_l]$ to help build a higher-order Lagrange polynomial; then, the desired polynomial $\psi_{l,i}^p(y)$ is defined by restricting the resulting polynomial to the support $[y_{l,i} - \widetilde{h}_l, y_{l,i} + \widetilde{h}_l]$. The constructions of the cubic polynomial $\psi_{2,3}^3(y)$ and the quartic polynomial $\psi_{3,1}^4(y)$ are illustrated in Figure 5.3.2(b). For the cubic polynomial associated with $y_{2,3}$, we introduce the additional ancestor $y_{0,0}$ to define a cubic Lagrange polynomial; for the quartic polynomial associated with $y_{3,1}$, two more ancestors $y_{1,1}$ and $y_{0,1}$ are added. After the construction of the cubic and quartic polynomials, we retain only the part within the support (solid curves) and cut out the parts outside the support (dashed curves). Using this strategy, we can construct high-order bases while retaining the hierarchical structure and, more importantly, the property (5.1.4) as in the linear case. It should be noted that because a total of p ancestors are needed, a polynomial of order p cannot be defined earlier than level $p-1$. In other words, at level L, the maximum order of polynomials is $p = L+1$. For example, a quartic polynomial basis of level 3 is plotted in Figure 5.3.2(a), where linear, quadratic, and cubic polynomials are used on levels 0, 1 and 2 due to the lack of ancestors. We observe that there are multiple types of basis functions on each level when $p \geq 3$ because of the different distributions of supporting points for different grid points. In general, the hierarchical basis of order $p > 1$ contains 2^{p-2} types

Table 5.3.1. Supporting points for high-order hierarchical bases ($p = 2, 3, 4$).

Order	Grid point $y_{l,i}$	Supporting points of $\psi^p_{l,i}(y)$
$p=2$	$l \geq 1, \mod(i,2) = 1$	$y_{l,i} - \widetilde{h}_l,\ y_{l,i},\ y_{l,i} + \widetilde{h}_l$
$p=3$	$l \geq 2, \mod(i,4) = 1$	$y_{l,i} - \widetilde{h}_l,\ y_{l,i},\ y_{l,i} + \widetilde{h}_l,\ y_{l,i} + 3\widetilde{h}_l$
	$l \geq 2, \mod(i,4) = 3$	$y_{l,i} - 3\widetilde{h}_l,\ y_{l,i} - \widetilde{h}_l,\ y_{l,i},\ y_{l,i} + \widetilde{h}_l$
$p=4$	$l \geq 3, \mod(i,8) = 1$	$y_{l,i} - \widetilde{h}_l,\ y_{l,i},\ y_{l,i} + \widetilde{h}_l,\ y_{l,i} + 3\widetilde{h}_l,\ y_{l,i} + 7\widetilde{h}_l$
	$l \geq 3, \mod(i,8) = 3$	$y_{l,i} - 3\widetilde{h}_l,\ y_{l,i} - \widetilde{h}_l,\ y_{l,i},\ y_{l,i} + \widetilde{h}_l,\ y_{l,i} + 5\widetilde{h}_l$
	$l \geq 3, \mod(i,8) = 5$	$y_{l,i} - 5\widetilde{h}_l,\ y_{l,i} - \widetilde{h}_l,\ y_{l,i},\ y_{l,i} + \widetilde{h}_l,\ y_{l,i} + 3\widetilde{h}_l$
	$l \geq 3, \mod(i,8) = 7$	$y_{l,i} - 7\widetilde{h}_l,\ y_{l,i} - 3\widetilde{h}_l,\ y_{l,i} - \widetilde{h}_l,\ y_{l,i},\ y_{l,i} + \widetilde{h}_l$

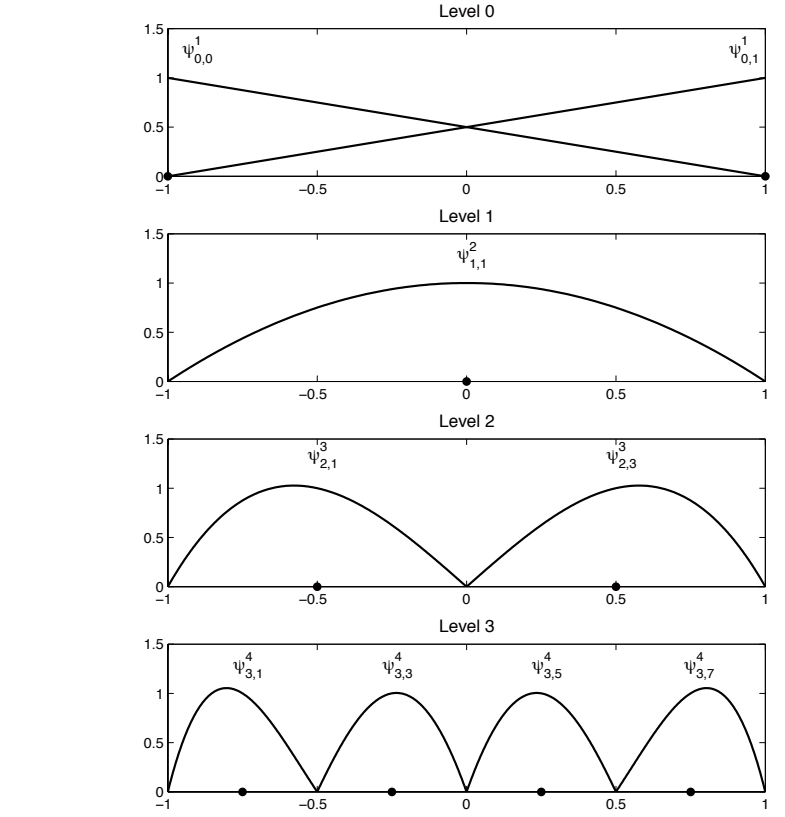
of pth-order polynomials. In Table 5.3.1 we list the supporting points used to define the hierarchical polynomial bases of order $p = 2, 3, 4$.

Wavelet basis

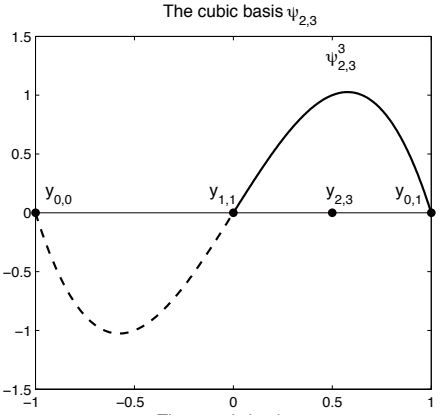
Besides the hierarchical bases discussed above, wavelets form another important family of basis functions which can provide a stable subspace splitting because of their Riesz property. In the following, let us briefly mention the second-generation wavelets constructed using the lifting scheme discussed in Sweldens (1996, 1998). Second-generation wavelets are a generalization of biorthogonal wavelets that is easier to apply for functions defined on bounded domains. The lifting scheme (Sweldens 1996, 1998) is a tool for constructing second-generation wavelets that are no longer dilates and translates of a single scaling function. The basic idea behind lifting is to start with simple multi-resolution analysis and gradually build a multi-resolution analysis with specific, *a priori* defined properties. The lifting scheme can be viewed as a process of taking an existing wavelet and modifying it by adding linear combinations of the scaling function at the coarse level. In the context of the piecewise linear basis, the second-generation wavelet on level $l \geq 1$, denoted by $\psi^{\mathrm{w}}_{l,i}(y)$, is constructed by 'lifting' the piecewise linear basis $\psi_{l,i}(y)$ as

$$\psi^{\mathrm{w}}_{l,i}(y) := \psi_{l,i}(y) + \sum_{i'=0}^{2^{l-1}} \beta^{i'}_{l,i} \psi_{l-1,i'}(y),$$

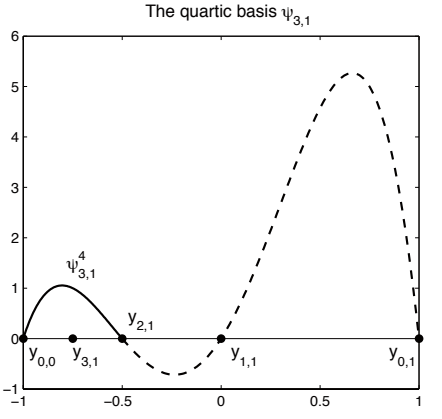
where, for $i = 0, \ldots, 2^{l-1}$, $\psi_{l-1,i}(y)$ are the nodal polynomials on level $l-1$ and the weights $\beta^j_{l,i}$ in the linear combination are chosen in such a way that the wavelet $\psi^{\mathrm{w}}_{l,i}(y)$ has more vanishing moments than $\psi_{l,i}(y)$ and thus provides a stabilization effect. Specifically, in the bounded domain $[-1, 1]$,



(a)



(b)



(c)

Figure 5.3.2. (a) Quartic hierarchical basis functions, where linear, quadratic, and cubic basis functions are used on levels 0, 1 and 2, respectively. Quartic basis functions appear beginning with level 3. (b,c) Construction of a cubic hierarchical basis function and a quartic hierarchical basis function.

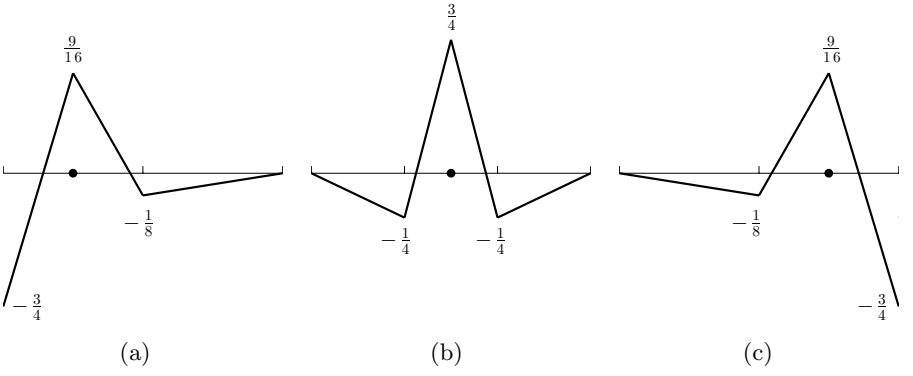


Figure 5.3.3. (a) Left-boundary wavelet, (b) central wavelet, (c) right-boundary wavelet.

we have three types of linear lifting wavelets:

$$\begin{aligned}
\psi^{\mathrm{w}}_{l,i} &:= \psi_{l,i} - \frac{1}{4}\psi_{l-1,\frac{i-1}{2}} - \frac{1}{4}\psi_{l-1,\frac{i+1}{2}} \quad \text{for } 1 < i < 2^l - 1, \ \ i \text{ odd}, \\
\psi^{\mathrm{w}}_{l,i} &:= \psi_{l,i} - \frac{3}{4}\psi_{l-1,\frac{i-1}{2}} - \frac{1}{8}\psi_{l-1,\frac{i+1}{2}} \quad \text{for } i = 1, \\
\psi^{\mathrm{w}}_{l,i} &:= \psi_{l,i} - \frac{1}{8}\psi_{l-1,\frac{i-1}{2}} - \frac{3}{4}\psi_{l-1,\frac{i+1}{2}} \quad \text{for } i = 2^l - 1,
\end{aligned} \tag{5.3.10}$$

where the three equations define the central 'mother' wavelet, the left-boundary wavelet, and the right-boundary wavelet, respectively. We illustrate the three lifting wavelets in Figure 5.3.3. For additional details, see Sweldens (1996).

Note that the property given in (5.2.2) is not valid for the lifting wavelets in (5.3.10) because neighbouring wavelets at the same level have overlapping support. As a result, the coefficient matrix of the linear system (5.2.9) is no longer triangular. Thus, J_h linear systems, each of size $M_L \times M_L$, need to be solved to obtain the surpluses in (5.2.8). However, note that for the second-generation wavelet defined in (5.3.10), the interpolation matrix is well-conditioned. See Gunzburger *et al.* (2014) for details.

5.4. Hierarchical acceleration of stochastic collocation methods

In the framework of stochastic finite element methods, the computational complexity is dominated by the cost of solving the $J_h M \times J_h M$ linear system (2.4.3) to obtain the coefficients c_{jm} in (2.4.1). When using a non-intrusive method such as the hSGSC method, the coupled $J_h M \times J_h M$ linear system decouples to M smaller linear systems, each of which leads to the solution of the deterministic PDE at one of the M collocation points in parameter

space. Because the M linear systems are independent and deterministic, they can be solved separately using classic FEM solvers, providing an easy path for parallelization compared to intrusive methods such as the stochastic Galerkin approach. However, the executions of iterative FEM solvers for those linear systems still dominate the total computational cost, especially for some complex physical problems such as turbulence flow models. In this section, we focus on further improving the computational efficiency of the hSGSC method by proposing a hierarchical acceleration approach to reduce the total number of iterations needed for solving the M decoupled linear systems. The key idea is to exploit the hierarchical structure to take advantage of the approximation of the current level to predict better initial guesses for the iterative solvers used to solve the deterministic systems at the sparse grid points on the next level.

We denote the decoupled linear system at the sparse grid point $\boldsymbol{y}_{\mathbf{l},\mathbf{i}}$ by

$$\boldsymbol{A}_{\mathbf{l},\mathbf{i}}\boldsymbol{u}_{\mathbf{l},\mathbf{i}} = \boldsymbol{f}_{\mathbf{l},\mathbf{i}}, \tag{5.4.1}$$

where $\boldsymbol{A}_{\mathbf{l},\mathbf{i}}$ denotes the $J_h \times J_h$ finite element system matrix,

$$\boldsymbol{f}_{\mathbf{l},\mathbf{i}} = (f_{1,\mathbf{l},\mathbf{i}}, \ldots, f_{J_h,\mathbf{l},\mathbf{i}})^\top$$

denotes the right-hand side vector, and $\boldsymbol{u}_{\mathbf{l},\mathbf{i}} = (u_{1,\mathbf{l},\mathbf{i}}, \ldots, u_{J_h,\mathbf{l},\mathbf{i}})^\top$ denotes the vector of coefficients that serve to define the deterministic FEM solution for the parameter point $\boldsymbol{y}_{\mathbf{l},\mathbf{i}}$. Specifically, in the rest of this section, we assume that the linear system in (5.4.1) is symmetric positive definite and, as an example to provide a concrete context, choose the well-known conjugate gradient (CG) method for its solution. We then have the well-known error estimate

$$\|\boldsymbol{e}^k_{\mathbf{l},\mathbf{i}}\|_{\boldsymbol{A}_{\mathbf{l},\mathbf{i}}} \le 2\left(\frac{\sqrt{\kappa_{\mathbf{l},\mathbf{i}}}-1}{\sqrt{\kappa_{\mathbf{l},\mathbf{i}}}+1}\right)^k \|\boldsymbol{e}^0_{\mathbf{l},\mathbf{i}}\|_{\boldsymbol{A}_{\mathbf{l},\mathbf{i}}},$$

where $\kappa_{\mathbf{l},\mathbf{i}}$ denotes the condition number of the system matrix $\boldsymbol{A}_{\mathbf{l},\mathbf{i}}$ and $\boldsymbol{e}^k_{\mathbf{l},\mathbf{i}} = \boldsymbol{u}_{\mathbf{l},\mathbf{i}} - \boldsymbol{u}^k_{\mathbf{l},\mathbf{i}}$ denotes the error of the output $\boldsymbol{u}^k_{\mathbf{l},\mathbf{i}}$ from the kth iteration of the CG simulation. With a prescribed accuracy $\varepsilon > 0$, the semi-discrete solution $u_{J_h}(\mathbf{x}, \boldsymbol{y}_{\mathbf{l},\mathbf{i}})$ is approximated by

$$u_{J_h}(\mathbf{x}, \boldsymbol{y}_{\mathbf{l},\mathbf{i}}) = \sum_{j=1}^{J_h} u_{j,\mathbf{l},\mathbf{i}}\phi_j(\mathbf{x}) \approx \widetilde{u}_{J_h}(\mathbf{x}, \boldsymbol{y}_{\mathbf{l},\mathbf{i}}) = \sum_{j=1}^{J_h} \widetilde{u}_{j,\mathbf{l},\mathbf{i}}\phi_j(\mathbf{x}),$$

where

$$\widetilde{\boldsymbol{u}}_{\mathbf{l},\mathbf{i}} = (\widetilde{u}_{1,\mathbf{l},\mathbf{i}}, \ldots, \widetilde{u}_{J_h,\mathbf{l},\mathbf{i}})^\top$$

is the output of the CG solver that satisfies $\|\boldsymbol{u}_{\mathbf{l},\mathbf{i}} - \widetilde{\boldsymbol{u}}_{\mathbf{l},\mathbf{i}}\|_{\boldsymbol{A}_{\mathbf{l},\mathbf{i}}} \le \varepsilon$. In this respect, the traditional strategy to improve the convergence rate is to develop preconditioners to reduce the condition number $\kappa_{\mathbf{l},\mathbf{i}}$. However, the quality of the initial guess also affects the convergence of the CG solver; a good

prediction of the solution $\boldsymbol{u}_{\mathbf{l},\mathbf{i}}$ will dramatically reduce the number of iterations necessary to reduce the error below a prescribed tolerance. From the formula in (5.2.10) for computing surpluses, we have

$$u_{j,\mathbf{l},\mathbf{i}} = u_{J_h,M_{L-1}}(\mathbf{x}_j, \boldsymbol{y}_{\mathbf{l},\mathbf{i}}) + c_{j,\mathbf{l},\mathbf{i}} \quad \text{for } j = 1, \ldots, J_h \text{ and } \boldsymbol{y}_{\mathbf{l},\mathbf{i}} \in W_L,$$

where $u_{J_h,M_{L-1}}(\mathbf{x}_j, \boldsymbol{y}_{\mathbf{l},\mathbf{i}})$ can be treated as a prediction of $u_{j,\mathbf{l},\mathbf{i}}$ at the new added grid point $y_{\mathbf{l},\mathbf{i}}$ on level L. The corresponding surplus is simply the error of such prediction. Then, due to the property in (5.3.2) that the surplus will decay to zero as the level increases, the quality of the prediction will become better and better. Therefore, at each new added point $\boldsymbol{y}_{\mathbf{l},\mathbf{i}}$ on level L, the initial guess of the linear system (5.4.1) is defined by

$$\widetilde{\boldsymbol{u}}^0_{\mathbf{l},\mathbf{i}} := \big(u_{J_h,M_{L-1}}(\mathbf{x}_1, \boldsymbol{y}_{\mathbf{l},\mathbf{i}}), \ldots, u_{J_h,M_{L-1}}(\mathbf{x}_{J_h}, \boldsymbol{y}_{\mathbf{l},\mathbf{i}})\big)^\top,$$

and we expect the necessary number of iterations to become smaller as the level $|\mathbf{l}|$ increases. We denote by $\widetilde{u}_{J_h,M_L}(\mathbf{x}, \boldsymbol{y})$ the approximate solution to $u_{J_h,M_L}(\mathbf{x}, \boldsymbol{y})$ obtained by the CG solver. To evaluate the efficiency of the hSGSC method, we describe the total computational cost for constructing $\widetilde{u}_{J_h,M_L}(\mathbf{x}, \boldsymbol{y})$ by

$$\mathcal{C}_{\text{total}} := \sum_{l=0}^{L} \sum_{|\mathbf{l}|=l} \sum_{\mathbf{i} \in B_{\mathbf{l}}} \mathcal{M}_{\mathbf{l},\mathbf{i}}, \tag{5.4.2}$$

where $\mathcal{M}_{\mathbf{l},\mathbf{i}}$ is the number of iterations used in the CG simulation to solve the deterministic FEM problem at the grid point $\boldsymbol{y}_{\mathbf{l},\mathbf{i}}$. So $\mathcal{C}_{\text{total}}$ is the total number of iterations for building $\widetilde{u}_{J_h,M_L}(\mathbf{x}, \boldsymbol{y})$. Now we apply our method to solve the second-order elliptic PDE in order to demonstrate the performance of the acceleration technique.

Example 5.4.1. We consider the two-dimensional Poisson equation with stochastic diffusivity and forcing term, that is,

$$\begin{aligned} \nabla \cdot (a(\mathbf{x}, \boldsymbol{y}) \nabla u(\mathbf{x}, \boldsymbol{y})) &= f(\mathbf{x}, \boldsymbol{y}) \quad \text{in } [0,1]^2 \times \Gamma, \\ u(\mathbf{x}, \boldsymbol{y}) &= 0 \qquad\quad \text{on } \partial D \times \Gamma, \end{aligned}$$

where κ and f are the nonlinear functions of the random vector $\boldsymbol{y}$ given by

$$a(\mathbf{x}, \boldsymbol{y}) = 0.1 + \exp\big[y_1 \cos(\pi x_1) + y_2 \sin(\pi x_2)\big],$$

and

$$f(\mathbf{x}, \boldsymbol{y}) = 10 + \exp\big[y_3 \cos(\pi x_1) + y_4 \sin(\pi x_2)\big],$$

where y_n for $n = 1, 2, 3, 4$ are independent and identically distributed random variables following the uniform distribution $U([-1, 1])$. To investigate the convergence of the approximate solution with respect to the random

Table 5.4.1. The computational cost and savings of the hSGSC method with acceleration for Example 5.4.1.

Basis type	Error	# SG points	hSGSC cost	hSGSC+acceleration cost	hSGSC+acceleration saving
linear	1.0×10^{-2}	377	13 841	7 497	45.8%
	1.0×10^{-3}	1 893	81 068	38 670	52.2%
	1.0×10^{-4}	7 777	376 287	167 832	55.3%
quadratic	1.0×10^{-3}	701	29 874	11 877	60.2%
	1.0×10^{-4}	2 285	110 744	36 760	66.8%
	1.0×10^{-5}	6 149	329 294	100 420	69.5%
cubic	1.0×10^{-4}	1 233	59 344	23 228	60.8%
	1.0×10^{-5}	3 233	172 845	57 777	66.5%
	1.0×10^{-6}	7 079	415 760	129 433	68.8%

variables, the error is measured by

$$e = \mathbb{E}\bigg[\int_D \big(u_{J_h}(\mathbf{x}, \boldsymbol{y}) - \widetilde{u}_{J_h M_L}\big)(\mathbf{x}, \boldsymbol{y})\, \mathrm{d}\mathbf{x}\bigg],$$

where the true solution $u_{J_h}(\mathbf{x}, \boldsymbol{y})$ is obtained by using a sufficiently fine sparse grid with the tolerance for adaptivity set to 10^{-8}; the tolerance for the CG solver is set to $\tau = 10^{-15}$. The deterministic FEM solver for computing $u_{J_h}(\mathbf{x}, \boldsymbol{y})$ for each $\boldsymbol{y}$ is constructed based on a triangulation with 2500 elements. For the hSGSC approximation, we fix $L = 20$, which is large enough, and vary the tolerance to increase the accuracy of the interpolant. The computational cost is measured by the total number of iterations of the CG solver. In Table 5.4.1, we list the computational costs of the standard and accelerated hSGSC methods for linear, quadratic, and cubic polynomial (in $\boldsymbol{y}$) bases. As expected, the hSGSC provides significant savings in the cost by using a more accurate initial guess for the CG solver. Note also that for the same accuracy, approximation with higher-order bases dramatically reduces the number of sparse grid points, resulting in further savings in the total cost. In fact, because the solution $u(\mathbf{x}, \boldsymbol{y})$ is analytic with respect to the random variables y_n, $n = 1, \ldots, N$, the acceleration based on sparse grid interpolation with a *global* polynomial basis is more accurate and efficient. Such results can be found in Jantsch *et al.* (2014).

5.5. Error estimate and complexity analysis

In this section, we rigorously analyse the approximation errors and the complexities of the standard and accelerated hSGSC method in order to demonstrate the improved efficiency of the proposed acceleration technique. For simplicity, we only consider the isotropic sparse grid interpolation given in (5.2.8), with a linear hierarchical basis ($p = 1$), for solving the second-order elliptic PDE with homogeneous Dirichlet boundary condition given in (2.1.2). However, the analyses in this section can be extended, without any essential difficulty, to adaptive hSGSC methods for more complicated PDEs. The deterministic FEM systems are solved by the conjugate gradient method.

We start by defining several notations used in the following derivation. Let $\widetilde{u}_{J_h,M_L}(\mathbf{x},\boldsymbol{y})$ denote the approximation to $u_{J_h,M_L}(\mathbf{x},\boldsymbol{y})$ obtained using the conjugate gradient method to solve the linear system (5.4.1). At each sparse grid point $\boldsymbol{y}_{\mathbf{l},\mathbf{i}} \in \mathcal{H}_L^N(\Gamma)$, the error from the CG simulation is represented by

$$\boldsymbol{e}_{\mathbf{l},\mathbf{i}} := \begin{pmatrix} u_{J_h}(\mathbf{x}_1,\boldsymbol{y}_{\mathbf{l},\mathbf{i}}) - \widetilde{u}_{J_h}(\mathbf{x}_1,\boldsymbol{y}_{\mathbf{l},\mathbf{i}}) \\ \vdots \\ u_{J_h}(\mathbf{x}_{J_h},\boldsymbol{y}_{\mathbf{l},\mathbf{i}}) - \widetilde{u}_{J_h}(\mathbf{x}_{J_h},\boldsymbol{y}_{\mathbf{l},\mathbf{i}}) \end{pmatrix}, \tag{5.5.1}$$

which is a $J_h \times 1$ vector. The maximum error of all CG simulations is defined by

$$e_{\mathrm{cg}} := \max_{\mathbf{i}\in B_{\mathbf{l}},|\mathbf{l}|\le L} \|\boldsymbol{e}_{\mathbf{l},\mathbf{i}}\|_2, \tag{5.5.2}$$

where $\|\cdot\|_2$ is the l^2-norm of the vector $\boldsymbol{e}_{\mathbf{l},\mathbf{i}}$. Similarly, we define

$$\kappa := \max_{\mathbf{i}\in B_{\mathbf{l}},|\mathbf{l}|\le L} \kappa_{\mathbf{l},\mathbf{i}}, \quad \tau_0 := \max_{\mathbf{i}\in B_{\mathbf{l}},|\mathbf{l}|\le L} \|\boldsymbol{e}^0_{\mathbf{l},\mathbf{i}}\|_2,$$

where $\kappa_{\mathbf{l},\mathbf{i}}$ and $\boldsymbol{e}^0_{\mathbf{l},\mathbf{i}}$ are the condition number of $\boldsymbol{A}_{\mathbf{l},\mathbf{i}}$ and the initial error of the CG simulation at $\boldsymbol{y}_{\mathbf{l},\mathbf{i}}$, respectively.

As mentioned in Section 2, we assume that the coefficient a and the forcing term f admit a smooth extension on the $\rho\,\mathrm{d}\boldsymbol{y}$-zero measure sets. Then (2.3.7) can be extended a.e. in Γ with respect to the Lebesgue measure (instead of the measure $\rho\,\mathrm{d}\boldsymbol{y}$). Thus, we estimate the error between u and $\widetilde{u}_{J_h,M_L}$ in the norm $\|\cdot\|_{L^2(D\times\Gamma)}$. The error of the approximate solution $\widetilde{u}_{J_h,M_L}(\mathbf{x},\boldsymbol{y})$ is given in the following lemma.

Lemma 5.5.1. For the second-order elliptic PDE with homogeneous Dirichlet boundary conditions in (2.1.2), the approximate solution $\widetilde{u}_{J_h,M_L}$ is constructed using the hSGSC method and the conjugate gradient solver.

Then the error $e = u - \widetilde{u}_{J_h,M_L}$ is bounded by

$$\begin{aligned}\|e\|_{L^2(D\times\Gamma)} &= \|u - \widetilde{u}_{J_h,M_L}\|_{L^2(D\times\Gamma)} \qquad (5.5.3)\\ &\le C_{\text{fem}} \cdot h^{r+1} + C_{\text{sg}} \cdot 2^{-2L} \sum_{n=0}^{N-1} \binom{L+N-1}{n} + 2^N \binom{L+N}{N} e_{\text{cg}},\end{aligned}$$

where $u \in H^{r+1}(D) \otimes L^2(\Gamma)$, the constant C_{fem} is independent of h and the random vector $\boldsymbol{y}$, the constant C_{sg} is independent of the level L and the dimension N.

Proof. It is easy to see that the total error can be split into

$$e = u - \widetilde{u}_{J_h,M_L} = \underbrace{u - u_{J_h}}_{e_1} + \underbrace{u_{J_h} - u_{J_h,M_L}}_{e_2} + \underbrace{u_{J_h,M_L} - \widetilde{u}_{J_h,M_L}}_{e_3}. \qquad (5.5.4)$$

The first term $e_1 = u - u_{J_h}$ is the FEM error from the spatial discretization, which is given by

$$\|e_1\|_{L^2(D\times\Gamma)} = \|u - u_{J_h}\|_{L^2(D\times\Gamma)} \le C_{\text{fem}} \cdot h^{r+1}, \qquad (5.5.5)$$

where $u(\mathbf{x}, \boldsymbol{y}) \in H^{r+1}(D) \otimes L^2(\Gamma)$ and the constant C_{fem} is independent of the mesh size h and the random vector $\boldsymbol{y}$. Next, according to the analyses in Bungartz and Griebel (2004), the error e_2 is bounded by

$$\|e_2\|_{L^2(D\times\Gamma)} \le C_{\text{sg}} \cdot 2^{-2L} \sum_{n=0}^{N-1} \binom{L+N-1}{n}, \qquad (5.5.6)$$

where the constant C_{sg} is independent of L and N.

We observe from the expression in (5.2.8) that both $u_{J_h,M_L}(\mathbf{x}, \boldsymbol{y})$ and $\widetilde{u}_{J_h,M_L}(\mathbf{x}, \boldsymbol{y})$ are linear combinations of continuous basis functions such that they are in the space $L^\infty(D\times\Gamma)$ and $\|e_3\|_{L^2(D\times\Gamma)} \le \|e_3\|_{L^\infty(D\times\Gamma)}$. Thus, we instead estimate the e_3 in the L^∞-norm. By substituting $u_{J_h,M_L} - \widetilde{u}_{J_h,M_L}$ into (5.2.6) and taking the L^∞-norm, we have

$$\begin{aligned}&\|e_3\|_{L^\infty(D\times\Gamma)} \qquad (5.5.7)\\ &\quad\le \max_{(\mathbf{x},\boldsymbol{y})\in D\times\Gamma} \left| \sum_{l=0}^{L} \sum_{|\mathbf{l}|=l} \left(\Delta^{l_1} \otimes \cdots \otimes \Delta^{l_N}\right)(u_{J_h} - \widetilde{u}_{J_h,M_L}) \right|\\ &\quad= \max_{(\mathbf{x},\boldsymbol{y})\in D\times\Gamma} \left| \sum_{l=0}^{L} \sum_{|\mathbf{l}|=l} \sum_{\boldsymbol{\alpha}\in\{0,1\}^N} (-1)^{|\boldsymbol{\alpha}|} \bigotimes_{n=1}^{N} \mathcal{I}_{l_n-\alpha_n}(u_{J_h} - \widetilde{u}_{J_h,M_L}) \right|\\ &\quad\le \sum_{l=0}^{L} \sum_{|\mathbf{l}|=l} \sum_{\boldsymbol{\alpha}\in\{0,1\}^N} \max_{(\mathbf{x},\boldsymbol{y})\in D\times\Gamma} \left| \bigotimes_{n=1}^{N} \mathcal{I}_{l_n-\alpha_n}(u_{J_h} - \widetilde{u}_{J_h,M_L}) \right|,\end{aligned}$$

where $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_N)$ is a multi-index for which each entry is 0 or 1. So

there are a total of 2^N combinations. Then, for a fixed $\mathbf{l}$ with $|\mathbf{l}| \le L$ and a fixed $\boldsymbol{\alpha} \in \{0,1\}^N$, we have the following estimate:

$$\begin{aligned}
&\max_{(\mathbf{x},\boldsymbol{y})\in D\times\Gamma}\left|\bigotimes_{n=1}^{N}\mathcal{I}_{l_n-\alpha_n}(u_{J_h}-\widetilde{u}_{J_h,M_L})(\mathbf{x},\boldsymbol{y})\right| \\
&= \max_{(\mathbf{x},\boldsymbol{y})\in D\times\Gamma}\left|\sum_{i_1=0}^{2^{l_1-\alpha_1}}\cdots\sum_{i_N=0}^{2^{l_N-\alpha_N}}\left\{\sum_{j=1}^{J_h}\left[u_{J_h}(\mathbf{x}_j,\boldsymbol{y}_{\mathbf{l},\mathbf{i}})-\widetilde{u}_{J_h}(\mathbf{x}_j,\boldsymbol{y}_{\mathbf{l},\mathbf{i}})\right]\phi_j(\mathbf{x})\right\}\psi_{\mathbf{l},\mathbf{i}}(\boldsymbol{y})\right| \\
&\le \max_{(\mathbf{x},\boldsymbol{y})\in D\times\Gamma}\left|\sum_{i_1=0}^{2^{l_1-\alpha_1}}\cdots\sum_{i_N=0}^{2^{l_N-\alpha_N}}\left[\|e_{\mathbf{l},\mathbf{i}}\|_\infty\cdot\sum_{j=1}^{J_h}\phi_j(\mathbf{x})\right]\psi_{\mathbf{l},\mathbf{i}}(\boldsymbol{y})\right| \\
&\le \max_{\boldsymbol{y}\in\Gamma}\left|\sum_{i_1=0}^{2^{l_1-\alpha_1}}\cdots\sum_{i_N=0}^{2^{l_N-\alpha_N}}\|e_{\mathbf{l},\mathbf{i}}\|_2\cdot\psi_{\mathbf{l},\mathbf{i}}(\boldsymbol{y})\right| \\
&\le \max_{|\mathbf{l}|\le L,\mathbf{i}\in B_{\mathbf{l}}}\|e_{\mathbf{l},\mathbf{i}}\|_2 = e_{\mathrm{cg}}.
\end{aligned}$$

By substituting the above estimate into (5.5.7), we obtain

$$\begin{aligned}
\|e_3\|_{L^\infty(D\times\Gamma)} &\le \sum_{l=0}^{L}\sum_{|\mathbf{l}|=l}\sum_{\boldsymbol{\alpha}\in\{0,1\}^N} e_{\mathrm{cg}} \le \sum_{l=0}^{L}2^N\binom{l+N-1}{N-1}e_{\mathrm{cg}} \\
&\le 2^N\sum_{n=N-1}^{N-1+L}\binom{L+N}{N}e_{\mathrm{cg}} = 2^N\binom{L+N}{N}e_{\mathrm{cg}},
\end{aligned}$$

which concludes the proof. □

Next, we analyse the cost of constructing $\widetilde{u}_{J_h,M_L}(\mathbf{x},\boldsymbol{y})$ with the prescribed error being $\varepsilon > 0$. According to the error estimate in Lemma 5.5.1, a sufficient condition of $\|e\|_{L^2(D\times\Gamma)} \le \varepsilon$ is that

$$\|e_1\|_{L^2(D\times\Gamma)} \le C_{\mathrm{fem}}\cdot h^{r+1} \le \frac{\varepsilon}{3}, \tag{5.5.8}$$

$$\|e_2\|_{L^2(D\times\Gamma)} \le C_{\mathrm{sg}}\cdot 2^{-2L}\sum_{n=0}^{N-1}\binom{L+N-1}{n} \le \frac{\varepsilon}{3}, \tag{5.5.9}$$

and

$$\|e_3\|_{L^2(D\times\Gamma)} \le 2^N\binom{L+N}{N}e_{\mathrm{cg}} \le \frac{\varepsilon}{3}. \tag{5.5.10}$$

Let $\mathcal{C}_{\min}$ in (5.4.2) represent the *minimum cost*, that is, the minimum number of CG iterations, to satisfy the inequalities (5.5.8), (5.5.9) and (5.5.10). The goal is to estimate an upper bound on $\mathcal{C}_{\min}$. Note that, for fixed dimension N, level L and mesh size h, the total cost $\mathcal{C}_{\mathrm{total}}$ is determined by

solving the inequality (5.5.10). The bigger are L and $1/h$, the higher the cost is. Thus, the estimation of $\mathcal{C}_{\text{total}}$ has two steps. Given N and ε, we first estimate the maximum h to achieve (5.5.8) and the minimum L to achieve (5.5.9); and then substitute the obtained values into (5.5.10) to obtain an upper bound on $\mathcal{C}_{\min}$.

To perform the first step, we need to relate the numbers of degrees of freedom of Z_L and W_l for $l \le L$, denoted by $|Z_L|$ and $|W_l|$, respectively. The estimation of $|Z_L|$ has been studied in Bungartz and Griebel (2004) and Nobile *et al.* (2008*a*), but the estimate in Nobile *et al.* (2008*a*) is not sufficiently sharp and the estimate in Bungartz and Griebel (2004) does not concern $|W_l|$. In the following lemma, we provide estimates of $|W_l|$ which directly lead to an estimate of $|Z_L|$.

Lemma 5.5.2. The dimensions of the subspaces W_l and Z_L for $N \ge 2$, that is, the numbers of grid points in $\Delta\mathcal{H}_l^N$ and $\mathcal{H}_L^N$, are bounded by

$$|W_l| \le 2^l \binom{l+N-1}{N-1} \le 2^l \left(\frac{l+N-1}{N-1}\right)^{N-1} \mathrm{e}^{N-1}, \tag{5.5.11}$$

and correspondingly,

$$|Z_L| \le 2^{L+1} \binom{L+N-1}{N-1} \le 2^{L+1} \left(\frac{L+N-1}{N-1}\right)^{N-1} \mathrm{e}^{N-1}, \tag{5.5.12}$$

where $0 \le l \le L$.

Proof. By using the formula (5.2.6) and exploiting the nested structure of the sparse grid, the dimension of Z_L can be represented by

$$|Z_L| = \sum_{l=0}^{L} |W_l| = \sum_{l=0}^{L} \sum_{|\mathbf{l}|=l} \prod_{n=1}^{N} (m_{l_n} - m_{l_n-1}), \tag{5.5.13}$$

where $m_{l_n} = 2^{l_n} + 1$ is the number of grid points involved in the one-dimensional interpolant $\mathcal{I}_{l_n}(\cdot)$ in (5.2.3) and $m_{-1} = 0$. In the case of using the linear hierarchical basis shown in Figure 5.2.1, then $m_{l_n} - m_{l_n-1} = 2^{l_n-1}$ for $l_n \ge 1$. We now derive an upper bound for $|W_l|$. Note that there are $\binom{N-1+l}{N-1}$ ways to form the sum l with $N-1+l$ non-negative integers, so we have

$$|W_l| = \prod_{n=1}^{N} (m_{i_n} - m_{i_n-1}) \binom{N-1+l}{N-1} \le 2^l \frac{(N-1+l)!}{(N-1)! \cdot l!}. \tag{5.5.14}$$

Using Stirling's approximation of a factorial in the inequality form

$$d_n \le n! \le d_n \left(1 + \frac{1}{4n}\right) \quad \text{with } d_n = \sqrt{2\pi n} \left(\frac{n}{\mathrm{e}}\right)^n, \quad n \in \mathbb{N}^+, \tag{5.5.15}$$

we obtain that

$$
\begin{aligned}
|W_l| &\le 2^l \left(1 + \frac{1}{4(N-1+l)}\right) \frac{d_{N-1+l}}{d_{N-1} \cdot d_l} \qquad (5.5.16)\\
&= 2^l \frac{\left(1 + \frac{1}{4(N-1+l)}\right)\sqrt{N-1+l}}{\sqrt{2\pi l(N-1)}} \left(\frac{N-1+l}{N-1}\right)^{N-1} \left(\frac{N-1+l}{l}\right)^{l}\\
&\le 2^l \left(\frac{l+N-1}{N-1}\right)^{N-1} \left(1 + \frac{N-1}{l}\right)^{l}\\
&\le 2^l \left(\frac{l+N-1}{N-1}\right)^{N-1} \mathrm{e}^{N-1}.
\end{aligned}
$$

This concludes the proof for $|W_l|$ and the estimate of $|Z_L|$ can be obtained immediately from the estimate of $|W_l|$. □

Similar to the analyses in Wasilkowski and Woźniakowski (1995), now we solve the equation (5.5.9) to find an upper bound for L such that the error of the isotropic sparse grid interpolation u_{J_h,M_L} is smaller than the prescribed accuracy $\varepsilon/3$.

Lemma 5.5.3. For $\varepsilon < 3C_{\mathrm{sg}}$ in (5.5.9), the accuracy $\|e_2\|_{L^2(D\times\Gamma)} \le \varepsilon/3$ can be achieved with level L bounded by

$$
L \le L_k = \frac{t_k N}{2\ln 2} + 1 \quad \text{with} \quad s = \frac{2\,\mathrm{e}}{\ln 2}\left(\frac{3C_{\mathrm{sg}}}{\varepsilon}\right)^{\frac{1}{N}}, \qquad (5.5.17)
$$

where $\{t_k\}_{k=0}^{\infty}$ is a monotonically decreasing sequence defined by

$$
t_k = \ln(t_{k-1}s) \quad \text{with} \quad t_0 = \frac{\mathrm{e}}{\mathrm{e}-1}\ln s. \qquad (5.5.18)
$$

Proof. We observe that the value of the minimal solution of the inequality (5.5.9) has two possibilities, that is, $L < N$ and $L \ge N$. In the former case, all values bigger than N are also solutions of (5.5.9). Hence, we assume the solution of (5.5.9) is bigger than N. It is also observed that if $L \ge N$, then

$$
\sum_{k=0}^{N-1} \binom{L+N-1}{k} \le N\binom{L+N-1}{N-1} \le N\binom{L+N}{N} \le N\left(\frac{2L}{N}\right)^{N} \mathrm{e}^{N}. \qquad (5.5.19)
$$

Thus, instead of solving (5.5.9) directly, we solve its sufficient condition as follows:

$$
C_{\mathrm{sg}} 2^{-2L} N \left(\frac{2L}{N}\right)^{N} \mathrm{e}^{N} \le \frac{\varepsilon}{3} \quad \text{and} \quad L \ge N, \qquad (5.5.20)
$$

Now we define $L = tN/\ln 4$ in (5.5.8). Then the inequality has the following

sufficient conditions:

$$\left(\frac{2L}{N}\right)^N \mathrm{e}^N \left(\frac{3NC_{\mathrm{sg}}}{\varepsilon}\right) \le 2^{2L} \tag{5.5.21}$$

$$\Longleftarrow \left(\frac{t}{\ln 2}\right)^N \mathrm{e}^N \left(\frac{3NC_{\mathrm{sg}}}{\varepsilon}\right) \le 4^{\frac{t}{\ln 4}N}$$

$$\Longleftarrow \left(\frac{t\,\mathrm{e}}{\ln 2}\right)\left(\frac{3NC_{\mathrm{sg}}}{\varepsilon}\right)^{\frac{1}{N}} \le 4^{\frac{t}{\ln 4}}$$

$$\Longleftarrow \ln t + \ln\left[\frac{\mathrm{e}}{\ln 2}\left(\frac{3C_{\mathrm{sg}}}{\varepsilon}\right)^{\frac{1}{N}} N^{\frac{1}{N}}\right] \le t$$

$$\Longleftarrow \ln t + \ln\left[\frac{2\,\mathrm{e}}{\ln 2}\left(\frac{3C_{\mathrm{sg}}}{\varepsilon}\right)^{\frac{1}{N}}\right] \le t.$$

Then we have

$$t \ge \ln t + \ln s \quad \text{with } s = \frac{2\,\mathrm{e}}{\ln 2}\left(\frac{3C_{\mathrm{sg}}}{\varepsilon}\right)^{\frac{1}{N}} \tag{5.5.22}$$

where $s > 1$ under the assumption of this lemma. By defining $t_0 = \frac{\mathrm{e}}{\mathrm{e}-1}\ln s$, it is easy to verify that

$$t_0 - \ln s = \frac{1}{\mathrm{e}-1}\ln s \ge 1 + \ln\left(\frac{1}{\mathrm{e}-1}\ln s\right) = \ln\left(\frac{\mathrm{e}}{\mathrm{e}-1}\ln s\right) = \ln t_0, \tag{5.5.23}$$

such that the inequality (5.5.8) is satisfied. Furthermore, for $k \ge 0$, $t_k = \ln(t_{k-1}s) \le t_{k-1}$ is also the solution of (5.5.8) due to the fact that

$$\ln t_k + \ln s = \ln(\ln t_{k-1} + \ln s) + \ln s \le \ln t_{k-1} + \ln s = \ln(t_{k-1}s) = t_k. \tag{5.5.24}$$

Thus, the sequence $\{t_k\}_{k=0}^{\infty}$ monotonically converges to a unique solution t^* such that $t^* = \ln t^* + \ln s$. □

To achieve the accuracy required in (5.5.10), we need to estimate the maximum error e_{cg} of the CG simulations. By definition of e_{cg}, we have

$$e_{\mathrm{cg}} = \max_{\mathbf{i}\in B_{\mathbf{l}},|\mathbf{l}|\le L} \|e_{\mathbf{l},\mathbf{i}}\|_2 \le \max_{\mathbf{i}\in B_{\mathbf{l}},|\mathbf{l}|\le L} \frac{1}{\sqrt{\lambda_{\mathbf{l},\mathbf{i}}}} \|\boldsymbol{e}_{\mathbf{l},\mathbf{i}}^{J_{\mathbf{l},\mathbf{i}}}\|_{A_{\mathbf{l},\mathbf{i}}} \tag{5.5.25}$$

$$\le \max_{\mathbf{i}\in B_{\mathbf{l}},|\mathbf{l}|\le L} \frac{2}{\sqrt{\lambda_{\mathbf{l},\mathbf{i}}}} \left(\frac{\sqrt{\kappa_{\mathbf{l},\mathbf{i}}}-1}{\sqrt{\kappa_{\mathbf{l},\mathbf{i}}}+1}\right)^{J_{\mathbf{l},\mathbf{i}}} \cdot \|\boldsymbol{e}_{\mathbf{l},\mathbf{i}}^0\|_{A_{\mathbf{l},\mathbf{i}}}$$

$$\le \max_{\mathbf{i}\in B_{\mathbf{l}},|\mathbf{l}|\le L} 2\sqrt{\kappa}_{\mathbf{l},\mathbf{i}} \left(\frac{\sqrt{\kappa_{\mathbf{l},\mathbf{i}}}-1}{\sqrt{\kappa_{\mathbf{l},\mathbf{i}}}+1}\right)^{J_{\mathbf{l},\mathbf{i}}} \cdot \|\boldsymbol{e}_{\mathbf{l},\mathbf{i}}^0\|_2$$

$$\le 2\sqrt{\kappa}\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^J \cdot \tau_0,$$

where $\lambda_{\mathbf{l},\mathbf{i}}$ and $\kappa_{\mathbf{l},\mathbf{i}}$ are the smallest eigenvalue and the condition number of the matrix $A_{\mathbf{l},\mathbf{i}}$, respectively, and $J_{\mathbf{l},\mathbf{i}}$ is the iteration number of the CG simulation conducted at the sparse grid point $\boldsymbol{y}_{\mathbf{l},\mathbf{i}} \in \mathcal{H}_L^N(\Gamma)$. The constant J is defined by

$$J := \min_{\mathbf{i}\in B_{\mathbf{l}},|\mathbf{l}|\le L} J_{\mathbf{l},\mathbf{i}}. \tag{5.5.26}$$

It should be noted that the condition numbers $\kappa_{\mathbf{l},\mathbf{i}}$ play an important role in estimating the number of iterations J. The value of J will dramatically grow as the value κ increases. However, in practice, many types of preconditioners can used to reduce the condition numbers of the M deterministic FEM system. In general, we assume that the upper bound κ of all the condition numbers $\kappa_{\mathbf{l},\mathbf{i}}$ can be bounded or represented by a function of mesh size h, denoted by $\overline{\kappa}(h) \ge \kappa$. On the other hand, to satisfy the condition (5.5.8), h can be represented by $h \le \varepsilon/(3C_{\text{fem}})$, such that κ can be bounded by a function of ε, that is,

$$\kappa \le \overline{\kappa}\left(\frac{3C_{\text{fem}}}{\varepsilon}\right). \tag{5.5.27}$$

Note that different preconditioners will lead to different forms of $\overline{\kappa}(\cdot)$. Since estimating the dependence of κ on ε is not our goal in this article, we use $\overline{\kappa}(\cdot)$ in (5.5.27) to represent the dependence of κ on ε in the following derivation. The estimation of $\mathcal{C}_{\text{min}}$ for standard hSGSC method without acceleration is given below.

Theorem 5.5.4. Under Lemmas 5.5.2 and 5.5.3, the minimum cost $\mathcal{C}_{\text{min}}$ for building a standard hSGSC approximation $\widetilde{u}_{J_h,M_L}$ satisfying (5.5.8), (5.5.9) and (5.5.10) can be bounded by

$$\begin{aligned}\mathcal{C}_{\text{min}} \le \frac{\alpha_1}{N}\left[\alpha_2 + \alpha_3\frac{\log_2\left(\frac{3C_{\text{sg}}}{\varepsilon}\right)}{N}\right]^{\alpha_4 N}\left(\frac{3C_{\text{sg}}}{\varepsilon}\right)^{\alpha_5} \\ \times \frac{1}{\log_2\left(\frac{\sqrt{\overline{\kappa}}+1}{\sqrt{\overline{\kappa}}-1}\right)}\left[\alpha_6\log_2\left(\frac{3C_{\text{sg}}}{\varepsilon}\right) + \log_2(\sqrt{\overline{\kappa}}\tau_0) + \alpha_7 N + \alpha_8\right],\end{aligned} \tag{5.5.28}$$

where the constants $\alpha_1, \ldots, \alpha_8$ are defined by

$$\begin{aligned}&\alpha_1 = 2, \quad \alpha_2 = \frac{2\,\mathrm{e}^2}{(\mathrm{e}-1)}\log_2\left(\frac{2\,\mathrm{e}}{\ln 2}\right), \quad \alpha_3 = \frac{2\,\mathrm{e}^2}{(\mathrm{e}-1)}, \quad \alpha_4 = \frac{3}{2}, \quad \alpha_5 = \frac{1}{2}, \\ &\alpha_6 = \frac{\mathrm{e}}{\mathrm{e}-1}, \quad \alpha_7 = \frac{\mathrm{e}}{\mathrm{e}-1}\log_2\left(\frac{2\,\mathrm{e}}{\ln 2}\right) + 1, \quad \alpha_8 = \log_2\left(\frac{2}{C_{\text{sg}}}\right).\end{aligned} \tag{5.5.29}$$

Proof. By definition in (5.4.2), the minimum cost $\mathcal{C}_{\text{min}}$ to achieve (5.5.8), (5.5.9) and (5.5.10), can be bounded by

$$\mathcal{C}_{\text{min}} \le |Z_L| J(\tau_0, \varepsilon, \overline{\kappa}, L_k, N), \tag{5.5.30}$$

where the L_k are determined from Lemma 5.5.3, and $J(\tau_0, \varepsilon, \overline{\kappa}, L_k, N)$ is the necessary number of iterations of the CG simulation at each sparse grid point to achieve the accuracy $\varepsilon/3$ in (5.5.10) for fixed N, ε and the initial CG error τ_0. Thus, $J(\tau_0, \varepsilon, \overline{\kappa}, L_k, N)$ is represented by substituting (5.5.25) into (5.5.10),

$$J(\tau_0, \varepsilon, \overline{\kappa}, L_k, N) = \frac{\frac{1}{2}\log_2(\overline{\kappa}) + \log_2\left[\frac{3\cdot 2^{N+1}\tau_0}{\varepsilon}\binom{L_k+N}{N}\right]}{\log_2\left(\frac{\sqrt{\overline{\kappa}}+1}{\sqrt{\overline{\kappa}}-1}\right)}, \tag{5.5.31}$$

where we temporarily treat J as a positive real number for convenience, and the desired iteration number is $\lceil J \rceil$.

As for the initial error τ_0, we set $u^0_{\mathbf{l},\mathbf{i}} = 0$ at each point $\boldsymbol{y}_{\mathbf{l},\mathbf{i}}$ in the context of a standard hSGSC method, so that the error is given by

$$\boldsymbol{e}^0_{\mathbf{l},\mathbf{i}} = (u_{J_h}(\mathbf{x}_1, \boldsymbol{y}_{\mathbf{l},\mathbf{i}}), \ldots, u_{J_h}(\mathbf{x}_{J_h}, \boldsymbol{y}_{\mathbf{l},\mathbf{i}}))^\top.$$

Substituting L_0 obtained in Lemma 5.5.3, we have

$$\begin{aligned}
&\log_2\left[\frac{3\cdot 2^{N+1}\tau_0}{\varepsilon}\binom{L_0+N}{N}\right] \qquad (5.5.32)\\
&\quad\le \log_2\left(\frac{3\cdot 2^{N+1}\tau_0}{\varepsilon}\right) + \log_2\left(\frac{\varepsilon}{3NC_{\mathrm{sg}}}2^{2L_0}\right)\\
&\quad= \log_2\left(\frac{2^{N+1}\tau_0}{C_{\mathrm{sg}}N}\right) + 2L_0\\
&\quad\le \log_2\left(\frac{2^{N+1}\tau_0}{C_{\mathrm{sg}}N}\right) + \frac{\mathrm{e}N}{\mathrm{e}-1}\log_2\left[\frac{2\,\mathrm{e}}{\ln 2}\left(\frac{3C_{\mathrm{sg}}}{\varepsilon}\right)^{\frac{1}{N}}\right]\\
&\quad\le N + \frac{\mathrm{e}N}{\mathrm{e}-1}\log_2\left[\frac{2\,\mathrm{e}}{\ln 2}\left(\frac{3C_{\mathrm{sg}}}{\varepsilon}\right)^{\frac{1}{N}}\right] + \log_2\left(\frac{2\tau_0}{C_{\mathrm{sg}}}\right)\\
&\quad= \frac{\mathrm{e}}{\mathrm{e}-1}\log_2\left(\frac{3C_{\mathrm{sg}}}{\varepsilon}\right) + N\left[\frac{\mathrm{e}}{\mathrm{e}-1}\log_2\left(\frac{2\,\mathrm{e}}{\ln 2}\right)+1\right] + \log_2\left(\frac{2\tau_0}{C_{\mathrm{sg}}}\right)\\
&\quad= \alpha_6\log_2\left(\frac{3C_{\mathrm{sg}}}{\varepsilon}\right) + \alpha_7 N + \alpha_8 + \log_2(\tau_0).
\end{aligned}$$

On the other hand, substituting L_1 into the upper bound on Z_{L_1}, we have

$$\begin{aligned}
|Z_{L_1}| &\le 2^{L_1+1}\binom{L_1+N-1}{N-1} \le 2^{L_1+1}\binom{L_1+N}{N} \qquad (5.5.33)\\
&\le 2^{L_1+1}\left(\frac{\varepsilon}{3NC_{\mathrm{sg}}}\right)2^{2L_1} \le \left(\frac{\varepsilon}{NC_{\mathrm{sg}}}\right)2^{\frac{3t_1 N}{2\ln 2}}\\
&= \left(\frac{\varepsilon}{NC_{\mathrm{sg}}}\right)2^{\frac{3\ln(t_0 h)N}{2\ln 2}} = \left(\frac{\varepsilon}{NC_{\mathrm{sg}}}\right)t_0^{\frac{3}{2}N}\left[\frac{2\,\mathrm{e}}{\ln 2}\left(\frac{3C_{\mathrm{sg}}}{\varepsilon}\right)^{\frac{1}{N}}\right]^{\frac{3}{2}N}
\end{aligned}$$

$$
\begin{aligned}
&= \left(\frac{\varepsilon}{NC_{\mathrm{sg}}}\right)\left(\frac{\mathrm{e}}{\mathrm{e}-1}\ln h\right)^{\frac{3}{2}N}\left(\frac{2\,\mathrm{e}}{\ln 2}\right)^{\frac{3}{2}N}\left(\frac{3C_{\mathrm{sg}}}{\varepsilon}\right)^{\frac{3}{2}} \\
&= \frac{2}{N}\left(\frac{3C_{\mathrm{sg}}}{\varepsilon}\right)^{\frac{1}{2}}\left\{\frac{2\,\mathrm{e}^2}{\mathrm{e}-1}\log_2\left[\frac{2\,\mathrm{e}}{\ln 2}\left(\frac{3C_{\mathrm{sg}}}{\varepsilon}\right)^{\frac{1}{N}}\right]\right\}^{\frac{3}{2}N} \\
&= \frac{2}{N}\left\{\frac{2\,\mathrm{e}^2}{\mathrm{e}-1}\log_2\left(\frac{2\,\mathrm{e}}{\ln 2}\right)+\frac{2\,\mathrm{e}^2}{\mathrm{e}-1}\frac{\log_2\left(\frac{3C_{\mathrm{sg}}}{\varepsilon}\right)}{N}\right\}^{\frac{3}{2}N}\left(\frac{3C_{\mathrm{sg}}}{\varepsilon}\right)^{\frac{1}{2}} \\
&= \alpha_1\left\{\alpha_2+\alpha_3\frac{\log_2\left(\frac{3C_{\mathrm{sg}}}{\varepsilon}\right)}{N}\right\}^{\alpha_4 N}\left(\frac{3C_{\mathrm{sg}}}{\varepsilon}\right)^{\alpha_5}.
\end{aligned}
$$

Hence, by substituting (5.5.31), (5.5.32), and (5.5.33) into (5.5.30), the proof is finished. □

Now we analyse the computational cost of the accelerated hSGSC method. Unlike the standard hSGSC method, where the initial error τ_0 is of the same scale as the maximum value of u_{J_h} in $D\times\Gamma$, in accelerated hSGSC, for each new added sparse grid point $\boldsymbol{y}_{\mathbf{l},\mathbf{i}}$ with $L=|\boldsymbol{i}|\geq 1$, the initial guess $u^j_{\mathbf{l},\mathbf{i}}$ is first predicted by the interpolated value $\widetilde{u}_{J_h,M_{L-1}}(\mathbf{x},\boldsymbol{y}_{\mathbf{l},\mathbf{i}})$. In this case, the error bound shown in (5.5.3) is still valid, but we can obtain a sharper bound for the error $u_{J_h}(\mathbf{x},\boldsymbol{y}_{\mathbf{l},\mathbf{i}})-\widetilde{u}_{J_h,M_{L-1}}(\mathbf{x},\boldsymbol{y}_{\mathbf{l},\mathbf{i}})$ at each sparse grid point on level L. The result is shown in the following lemma.

Lemma 5.5.5. Using the isotropic sparse grid interpolation in (5.2.8), at a sparse grid point $\boldsymbol{y}_{\mathbf{l},\mathbf{i}}$ with $L=|\mathbf{l}|\geq 1$ and $\mathbf{i}\in B_{\mathbf{l}}$, the error $u_{J_h}(\mathbf{x},\boldsymbol{y}_{\mathbf{l},\mathbf{i}})-\widetilde{u}_{J_h,M_{L-1}}(\mathbf{x},\boldsymbol{y}_{\mathbf{l},\mathbf{i}})$ can be bounded by

$$
\left\|u_{J_h}(\mathbf{x},\boldsymbol{y}_{\mathbf{l},\mathbf{i}})-\widetilde{u}_{J_h,M_{L-1}}(\mathbf{x},\boldsymbol{y}_{\mathbf{l},\mathbf{i}})\right\|_{L^2(D)}\leq C_{\mathrm{surp}}2^{-2L}+2^N e_{\mathrm{cg}}, \tag{5.5.34}
$$

where $C_{\mathrm{surp}}>0$ is independent of L and e_{cg} is the maximum error of the CG simulations.

Proof. As in (5.5.3), we split the error into two parts, that is,

$$
\begin{aligned}
&u_{J_h}(\boldsymbol{y}_{\mathbf{l},\mathbf{i}})-\widetilde{u}_{J_h,M_{L-1}}(\boldsymbol{y}_{\mathbf{l},\mathbf{i}}) \\
&\quad=\underbrace{u_{J_h}(\boldsymbol{y}_{\mathbf{l},\mathbf{i}})-u_{J_h,M_{L-1}}(\boldsymbol{y}_{\mathbf{l},\mathbf{i}})}_{e_1}+\underbrace{u_{J_h,M_{L-1}}(\boldsymbol{y}_{\mathbf{l},\mathbf{i}})-\widetilde{u}_{J_h,M_{L-1}}(\boldsymbol{y}_{\mathbf{l},\mathbf{i}})}_{e_2},
\end{aligned} \tag{5.5.35}
$$

where e_1 is the definition of the hierarchical surplus, whose bound has been proved in Lemma 3.6 of Bungartz and Griebel (2004), that is,

$$
\|e_1\|_{L^\infty(D)}\leq 2^{-N}\cdot\|u_{J_h}\|_{L^\infty(D)\otimes L^2(\Gamma)}\cdot 2^{-2|\mathbf{i}|}=C_{\mathrm{surp}}2^{-2L}; \tag{5.5.36}
$$

and e_2 measures the error between the exact prediction and the perturbed one. To estimate e_2, we need to extend the formula for calculating surpluses

given in Bungartz and Griebel (2004) by including the sparse grid points on the boundary. Based on Lemma 3.2 of Bungartz and Griebel (2004), we can see that for each grid point $(\mathbf{x}_j, \boldsymbol{y}_{\mathbf{l},\mathbf{i}})$ for $j = 1, \ldots, J_h$ and $|\mathbf{i}| \geq 1$, its surplus $w_{j,\mathbf{l},\mathbf{i}}$ can be computed from the solution u_{J_h} in the following way:

$$w_{j,\mathbf{l},\mathbf{i}} = \mathcal{A}_{\mathbf{l},\mathbf{i}}(u_{J_h}(\mathbf{x}_j, \cdot)) = \left(\prod_{n=1}^{N} \mathcal{A}_{l_n,i_n}\right)(u_{J_h}(\mathbf{x}_j, \cdot)), \tag{5.5.37}$$

where $\mathcal{A}_{\mathbf{l},\mathbf{i}}(\cdot)$ is an N-dimensional stencil, which gives us the coefficients for a linear combination of the nodal values of the solution u_{J_h} to compute $w_{\mathbf{l},\mathbf{i}}$. Specifically, $\mathcal{A}_{\mathbf{l},\mathbf{i}}$ is product of N one-dimensional stencils $\mathcal{A}_{l_n,i_n}$ for $n = 1, \ldots, N$, defined by

$$\begin{aligned}\mathcal{A}_{l_n,i_n}(u_{J_h}(\mathbf{x}_j, \cdot)) =& \left[-\frac{1}{2} \quad 1 \quad -\frac{1}{2}\right]_{\boldsymbol{y}_{l_n,i_n}} (u_{J_h}(\mathbf{x}_j, \cdot)) \\ =& -\frac{1}{2} u_{J_h}(\mathbf{x}_j, \boldsymbol{y}_{\mathbf{l},\mathbf{i}} - \widetilde{h}_{l_n}\mathbf{1}_n) + u_{J_h}(\mathbf{x}_j, \boldsymbol{y}_{\mathbf{l},\mathbf{i}}) \\ & - \frac{1}{2} u_{J_h}(\mathbf{x}_j, \boldsymbol{y}_{\mathbf{l},\mathbf{i}} + \widetilde{h}_{l_n}\mathbf{1}_n)\end{aligned} \tag{5.5.38}$$

where $\mathbf{1}_n$ is a vector of zeros except for the nth entry, which is one, and $\widetilde{h}_{l_n}$ is a scalar equal to a half of the length of the support of the basis function $\psi_{\mathbf{l},\mathbf{i}}(\boldsymbol{y})$ in the nth direction. It is easy to see that the sum of the absolute values of the coefficients of $\mathcal{A}_{\mathbf{l},\mathbf{i}}(\cdot)$ is equal to 2^N. Note that all the involved grid points in (5.5.38) belong to $\mathcal{H}^N_{L-1}(\Gamma)$ except for $\boldsymbol{y}_{\mathbf{l},\mathbf{i}}$. Thus, due to the fact that

$$|u_{J_h}(\mathbf{x}_j, \boldsymbol{y}_{\mathbf{l},\mathbf{i}}) - \widetilde{u}_{J_h}(\mathbf{x}_j, \boldsymbol{y}_{\mathbf{l},\mathbf{i}})| \leq e_{\mathrm{cg}} \quad \text{for } j = 1, \ldots, J_h,$$

the error e_2 can be estimated by

$$\begin{aligned}\|e_2\|_{L^2(D)} &= \left\|\mathcal{A}_{\mathbf{l},\mathbf{i}}(u_{J_h} - \widetilde{u}_{J_h}) - (u_{J_h}(\boldsymbol{y}_{\mathbf{l},\mathbf{i}}) - \widetilde{u}_{J_h}(\boldsymbol{y}_{\mathbf{l},\mathbf{i}}))\right\|_{L^2(D)} \\ &\leq 2^N e_{\mathrm{cg}}.\end{aligned} \tag{5.5.39}$$

□

Theorem 5.5.6. Under Lemmas 5.5.2 and 5.5.3, the total cost $\mathcal{C}_{\mathrm{total}}$ in (5.4.2) for building isotropic sparse grid approximation $\widetilde{u}_{J_h,M_L}$ with accuracy ε using accelerated hSGSC method is bounded by

$$\begin{aligned}\mathcal{C}_{\min} \leq \alpha_1 &\left[\alpha_2 + \alpha_3 \frac{\log_2\left(\frac{3C_{\mathrm{sg}}}{\varepsilon}\right)}{N}\right]^{\alpha_4 N} \left(\frac{3C_{\mathrm{sg}}}{\varepsilon}\right)^{\alpha_5} \\ &\times \frac{1}{\log_2\left(\frac{\sqrt{\kappa}+1}{\sqrt{\kappa}-1}\right)} \left[2N - \log_2(N) + \alpha_9 + \log_2(\sqrt{\kappa})\right],\end{aligned} \tag{5.5.40}$$

where the constants $\alpha_1, \ldots, \alpha_5$ are defined as in Theorem 5.4.2 and α_9 is

defined by

$$\alpha_9 = \log_2\left(\frac{C_{\text{surp}}}{C_{\text{sg}}}\right) + 3. \tag{5.5.41}$$

Proof. In the case of $L = L_1$, according to the definition in (5.4.2), $\mathcal{C}_{\min}$ can be decomposed as

$$\mathcal{C}_{\min} \le \sum_{l=0}^{L_1} |W_l| J(\tau_0^l, \varepsilon, \overline{\kappa}, L_1, N), \tag{5.5.42}$$

where J is defined as in (5.5.31). Based on Lemma 5.5.5, we define the initial searching interval τ_0^l on level l by

$$\tau_0^l = C_{\text{surp}} 2^{-2l} + 2^N e_{\text{cg}}, \tag{5.5.43}$$

where e_{cg} is defined in (5.5.25). For sufficiently small ε, the logarithmic function in (5.5.42) is positive. By defining

$$\widehat{J}(\tau_0^l, \varepsilon, L, N) := \log_2\left[\frac{3 \cdot 2^{N+1}\tau_0^l}{\varepsilon}\binom{L+N}{N}\right],$$

we have

$$\mathcal{C}_{\min} \le \zeta(\mathcal{C}_{\min}^1 + \mathcal{C}_{\min}^2),$$

where

$$\mathcal{C}_{\min}^1 := \frac{1}{2}\log_2(\overline{\kappa})|Z_L|, \quad \mathcal{C}_{\min}^2 := \sum_{l=0}^{L_1} |W_l| \widehat{J}(\tau_0^l, \varepsilon, L_1, N),$$

$$\text{and} \quad \zeta := \frac{1}{\log_2\left(\frac{\sqrt{\overline{\kappa}}+1}{\sqrt{\overline{\kappa}}-1}\right)},$$

where $\mathcal{C}_{\min}^1$ can be obtained directly from the estimate of Z_L. Thus we focus on estimating $\mathcal{C}_{\min}^2$. Substituting τ_0^l into (5.5.42), we obtain

$$\begin{aligned}
\mathcal{C}_{\min}^2 &\le \sum_{l=0}^{L_1} 2^l \binom{l+N-1}{N-1} \\
&\quad \times \log_2\left[\frac{3 \cdot 2^{N+1}}{\varepsilon}\binom{L_1+N}{N}\left(C_{\text{surp}}2^{-2l} + 2^N e_{\text{cg}}\right)\right] \\
&= \sum_{l=0}^{L_1} 2^l \binom{l+N-1}{N-1} \\
&\quad \times \log_2\left[\frac{3 \cdot 2^{N+1}}{\varepsilon}\binom{L_1+N}{N}\left(C_{\text{surp}}2^{-2l} + \frac{\varepsilon}{3\binom{L_1+N}{N}}\right)\right] \\
&= \sum_{l=0}^{L_1} 2^l \binom{l+N-1}{N-1} \log_2\left[\frac{3 \cdot 2^{N+1} C_{\text{surp}} 2^{-2l}}{\varepsilon}\binom{L_1+N}{N}\right] + N
\end{aligned}$$

$$
\begin{aligned}
&\le \sum_{l=0}^{L_1} 2^l \binom{l+N-1}{N-1} \log_2\left[\frac{2^{N+1} C_{\mathrm{surp}} 2^{2(L_1-l)}}{\varepsilon} \frac{\varepsilon}{N C_{\mathrm{sg}}}\right] + N \\
&= \sum_{l=0}^{L_1} 2^l \binom{l+N-1}{N-1} \left[2(L_1-l) + \log_2\left(\frac{C_{\mathrm{surp}}}{C_{\mathrm{sg}}}\right) + 2N + 1 - \log_2(N)\right] \\
&\le \binom{L_1+N}{N} \sum_{l=0}^{L_1} (L_1-l) 2^l \\
&\qquad + 2^{L_1+1} \binom{L_1+N}{N} \left[\log_2\left(\frac{C_{\mathrm{surp}}}{C_{\mathrm{sg}}}\right) + 2N + 1 - \log_2(N)\right] \\
&\le 2^{L_1+1} \binom{L_1+N}{N} \left[\log_2\left(\frac{C_{\mathrm{surp}}}{C_{\mathrm{sg}}}\right) + 2N + 3 - \log_2(N)\right] \\
&\le \alpha_1 \left[\alpha_2 + \alpha_3 \frac{\log_2\left(\frac{3C_{\mathrm{sg}}}{\varepsilon}\right)}{N}\right]^{\alpha_4 N} \left(\frac{3C_{\mathrm{sg}}}{\varepsilon}\right)^{\alpha_5} \left[2N - \log_2(N) + \alpha_9\right].
\end{aligned}
$$

Thus, substituting the estimates of $\mathcal{C}_{\min}^2$ and $|Z_L|$ into (5.5.42), the proof is complete. □

Remark 5.5.7. Theorems 5.5.4 and 5.5.6 tell us that the cost of the hSGSC method is mainly determined by the number of sparse grid points M_L, the condition numbers of the relevant finite element system, and the initial guesses of the CG simulations. Asymptotically, the growth rate of M_L is characterized by the constants α_4 and α_5, and the cost due to inaccurate initial guesses is of order $\log_2(1/\varepsilon)$. Note that the use of acceleration techniques with accurate initial guesses will reduce the total cost by a factor $\log_2(1/\varepsilon)$ asymptotically, which is consistent with the numerical results given in Example 5.4.1.

APPENDIX

A. Brief review of probability theory

Essential concepts and definitions of probability theory required in this work are reviewed in this section. Following Rudin (1987), Loève (1977, 1978) and Rao and Swift (2006), we first provide a very brief introduction to the measure-theoretic foundations of probability theory and then explore several important concepts such as real-valued random variables and vectors, the notion of moment operators, and stochastic processes. Further concepts in probability theory can be found in several references: see, for example, Taylor (1997), Loève (1977) and Grigoriu (2002).

A.1. The notion of measurability

The class of continuous functions plays a fundamental role in topological theory. It has several elementary properties in common with measurable functions that play an essential role in integration theory. In what follows, we present an abstract setting to emphasize the analogies between the concepts *topological spaces, open sets* and *continuous functions* with *measurable spaces, measurable sets* and *measurable functions*. Here Ω is defined as a non-empty set with a finite or infinite (countable[10] or uncountable) number of elements ω.

Definition A.1 (topological space). A *topology* $\mathcal{F}$ on a non-empty set Ω is a collection of subsets of Ω such that

(i) $\emptyset \in \mathcal{F}$ and $\Omega \in \mathcal{F}$,

(ii) if $A_i \in \mathcal{F}$ for $i = 1, \ldots, n$, then $\bigcap_{i=1}^{n} A_i \in \mathcal{F}$,

(iii) if $A_\alpha \in \mathcal{F}$ for $\alpha \in \mathcal{A}$, for an arbitrary index set $\mathcal{A}$, then $\bigcup_{\alpha\in\mathcal{A}} A_\alpha \in \mathcal{F}$,

where the members of $\mathcal{F}$ are called the *open sets* of Ω and the ordered pair $(\Omega, \mathcal{F})$ is called a *topological space*.

Definition A.2 (σ-algebra and measurable space). A collection $\mathcal{F}$ of subsets of a non-empty set Ω is called a *σ-algebra* of Ω if $\mathcal{F}$ satisfies

(i) $\Omega \in \mathcal{F}$,

(ii) if $A \in \mathcal{F}$, then $A^c \in \mathcal{F}$, where $A^c = \Omega \backslash A$ is the complement of A in Ω,

(iii) if $\{A_n\}_{n=1}^{\infty} \subset \mathcal{F}$, then $\bigcup_{n=1}^{\infty} A_n \in \mathcal{F}$,

in which case the ordered pair $(\Omega, \mathcal{F})$ is called a *measurable space* and the members of $\mathcal{F}$ are called the *measurable sets* in Ω.

Definition A.3 (measurable function). Let $(\Omega, \mathcal{F})$ and (Υ, Σ) denote measurable spaces. Then, a function $\mu : \Omega \to \Upsilon$ is measurable if, for every $A \in \Upsilon$, the pre-image of A under μ is in $\mathcal{F}$, that is,

$$\mu^{-1}(A) \equiv \big\{\omega \in \Omega \mid \mu(\omega) \in A\big\} \subset \mathcal{F}.$$

Definition A.4 (positive measure and measure space). Let $(\Omega, \mathcal{F})$ be a measurable space. A function $\mu : \mathcal{F} \to [0, \infty]$ is called a *positive measure*[11] if μ satisfies the following.

[10] A set S is *countable* if all its elements can be indexed by natural numbers in a one-to-one fashion, *i.e.*, there exists a function $f : \mathbb{N} \to S$ such that $S = \big\{f(n) : n \in \mathbb{N}\big\}$ and, if $f(n_1) = f(n_2)$, then $n_1 = n_2$. A set is at most countable if it is either finite, that is, it can be 'counted' using $\{1, 2, \ldots, n\}$ for some n, or countable, that is, it can be counted using $\mathbb{N}$.

[11] What we call a *positive measure* is usually just referred to as a measure. If $\mu(A) = 0$ for every $A \in \mathcal{F}$ then, by our definition, μ is a positive measure.

(i) Non-negativity: for all $A \in \mathcal{F}$, $\mu(A) \geq 0$.

(ii) Null empty set: $\mu(\emptyset) = 0$.

(iii) Countable additivity: if $A_1, A_2, \ldots \in \mathcal{F}$ and $A_i \cap A_j = \emptyset$ for $i \neq j$, then

$$\mu\left(\bigcup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} \mu(A_i).$$

The triple $(\Omega, \mathcal{F}, \mu)$ is called a *measure space*.

Remark A.5. Measure spaces are often referred to as 'ordered triples' $(\Omega, \mathcal{F}, \mu)$, where Ω is a set, $\mathcal{F}$ is a σ-algebra in Ω, and μ is a measure defined on $\mathcal{F}$. Similarly, measurable spaces are often referred to as 'ordered pairs' $(\Omega, \mathcal{F})$. These conventions make common sense and are logically correct, even though they are somewhat redundant. For example, given the aforementioned ordered pair, the set Ω is merely the largest member of $\mathcal{F}$; hence, given $\mathcal{F}$, we can construct Ω. Moreover, by definition, every measure takes a σ-algebra as its domain so that, given a measure μ, we can deduce the σ-algebra $\mathcal{F}$ in which μ is defined and we also know the set Ω in which $\mathcal{F}$ is a σ-algebra. It is therefore admissible to use the expressions 'let μ be a measure' or 'let μ be a measure on Ω' if we choose to emphasize the set, or even 'let μ be a measure on $\mathcal{F}$' if we want to emphasize the σ-algebra. The customary approach, which is logically rather meaningless, is to say 'let Ω be a measure space', even though it is understood that there is a measure defined on $\mathcal{F}$ in Ω and it is the measure that we are mathematically interested in.

Borel σ-algebras

The Borel σ-algebra is an important example of a σ-algebra that is used in the theory of functions, Lebesgue integration, and probability. Before giving a definition, we state a classical theorem showing that σ-algebras exist in great profusion.

Theorem A.6. Let Ω be a set and $\mathcal{V}$ is a non-empty collection of subsets of Ω. There exists a smallest σ-algebra, denoted by $\sigma(\mathcal{V})$, in Ω such that $\mathcal{V} \subset \sigma(\mathcal{V})$, namely

$$\sigma(\mathcal{V}) := \bigcap \{\mathcal{F} : \mathcal{F} \text{ is a } \sigma\text{-algebra of } \Omega, \mathcal{V} \subset \mathcal{F}\},$$

which is also called the σ-algebra generated by $\mathcal{V}$.

We now let Ω be a topological space. By Theorem A.6, if $\mathcal{V}$ is a collection of all open sets (or, equivalently, all closed sets) of Ω, then the smallest σ-algebra $\mathcal{B} = \sigma(\mathcal{V})$ called the *Borel σ-algebra* on Ω. The elements of $B \in \mathcal{B}$ are called the *Borel sets*, which can be formed from open sets (or, equivalently, from closed sets) through operations of countable intersection,

countable union, and relative complement. Because $\mathcal{B}$ is a σ-algebra, we may regard $(\Omega, \mathcal{B})$ as a measurable space, with the Borel sets playing the role of the measurable sets. If $\mu : \Omega \to \Upsilon$ is a continuous mapping of Ω, where Υ is another topological space, then from the definitions we obtain that $\mu^{-1}(A) \in \mathcal{B}$ for every open set $A \in \Upsilon$. In conclusion, every continuous mapping of Ω is Borel-measurable. Borel-measurable mappings are often referred to as Borel mappings or Borel functions.

A.2. Probability spaces and random variables

Probability measure

Basically, probability is the numerical measure of uncertainty of outcomes of an action or experiment. The actual assignment of these values should be based on experience and should generally be verifiable when the experiment is, if possible, repeated under essentially the same conditions. To build an axiomatic representation we first represent all possible outcomes of an experiment as distinct points of a non-empty set. Since the collection of all such possibilities can be infinitely large, various combinations of them, useful to the experiments, have to be considered. We then define combinations of such outcomes as *events* and consider an algebra of events as the primary datum which includes everything of conceivable use for an experiment. Finally, each event is assigned a numerical measure corresponding to the 'quantity' of uncertainty in such a way that this uncertainty has natural additive and consistency properties. Mathematically, this axiomatic formulation was created by Kolmogorov; the analytical structure is what we next describe.

Definition A.7 (probability measure and probability space). Let $(\Omega, \mathcal{F})$ be a measurable space representing all possible outcomes of an experiment, where the members of the σ-algebra $\mathcal{F}$, called *events*, are collections of outcomes of the experiment. $\mathbb{P} : \mathcal{F} \to [0, 1]$ is called a *probability measure*, or simply a *probability*, if it is a measure on $(\Omega, \mathcal{F})$ satisfying

$$\mathbb{P}(A) \geq 0 \quad \text{for all } A \in \mathcal{F} \quad \text{and} \quad \mathbb{P}(\Omega) = 1.$$

A *probability space* is the triple $(\Omega, \mathcal{F}, \mathbb{P})$.

Thus, a probability space is a finite measure space whose measure function is normalized so that the entire space has measure one. The space $(\Omega, \mathcal{F}, \mathbb{P})$ is called a *complete probability space* if $\mathcal{F}$ contains all the subsets A of Ω with $\mathbb{P}$-outer measure zero, that is, with

$$\mathbb{P}^*(A) = \inf\{\mathbb{P}(F) : F \in \mathcal{F}, A \subset F\} = 0.$$

Any probability space can be made complete by adding to $\mathcal{F}$ all the sets

with outer measure being zero and by extending $\mathbb{P}$ accordingly.[12] Similarly, the subsets A of Ω which belong to $\mathcal{F}$ are called $\mathcal{F}$-measurable. However, in the probability context, the interpretation of these events is different. For example, when we write $\mathbb{P}(A)$, what we mean is 'the probability that the event A occurs'. In particular, if $\mathbb{P}(A) = 1$ we say that 'A occurs with probability 1' or 'almost surely' (a.s.).

Conditional probability

Let $(\Omega, \mathcal{F}, \mathbb{P})$ denote a probability space and let $A_1, A_1 \in \mathcal{F}$ be events with $P(A_1) > 0$ and $P(A_2) > 0$. Denote the intersection $A_1 \cap A_2$ by the 'product' A_1A_2. Then the ratio $P(A_1A_2)/P(A_1)$ is called the *conditional probability* of A_2 given A_1, or simply the probability of A_2 given A_1, and is denoted by $P(A_2|A_1)$ so that

$$P(A_1A_2) = P(A_1)P(A_2|A_1). \tag{A.1}$$

Then, by induction, for $A_1, A_2, \ldots, A_N \in \mathcal{F}$ we obtain the chain rule

$$P\left(\bigcap_{i=1}^{N} A_i\right) = P(A_1)P(A_2|A_1)\cdots P(A_1A_2\cdots A_{N-1}|A_N). \tag{A.2}$$

Moreover, if $\bigcup_i A_i = \Omega$ with $A_iA_{j\neq i} = \emptyset$ and A_i and $B \in \mathcal{F}$, we have that

$$P(B) = P(\Omega B) = \sum_i P(A_iB). \tag{A.3}$$

Then, the total probability rule follows from (A.1), namely

$$P(B) = \sum_i P(A_i)P(B|A_i). \tag{A.4}$$

Finally, using (A.1)–(A.4), we arrive at *Bayes' theorem*:

$$P(A_j|B) = \frac{P(A_j)P(B|A_j)}{\sum_i P(A_i)P(B|A_i)}.$$

Random variables and their probability distributions

Definition A.8 (random variables). Let $(\Omega, \mathcal{F}, \mathbb{P})$ denote a probability space. A function $X : \Omega \to \mathbb{R}$ is a *random variable* if X satisfies

$$X^{-1}(\mathcal{B}) \subset \mathcal{F} \quad \text{or} \quad X^{-1}(A) := \{\omega \in \Omega \mid X(\omega) \in A\} \in \mathcal{F},$$

where $\mathcal{B}$ is the Borel σ-algebra of $\mathbb{R}$, and $A = (-\infty, x), x \in \mathbb{R}$.

Thus, a random variable is a function from the abstract set Ω to the real space, where each outcome $\omega \in \Omega$ is assigned a real number $X(\omega) \in \mathbb{R}$.

[12] In this article, we assume that all probability spaces are complete.

A random variable is of real interest when related to its image measure or to the distribution function in the context of probability theory.

Definition A.9 (image measure). Let $(\Omega, \mathcal{F}, \mathbb{P})$ denote a probability space and $X : \Omega \to \mathbb{R}$ denote a random variable. The *image measure* of X, denoted by $\mathbb{P}_X$, is a measure on the Borel space $(\mathbb{R}, \mathcal{B})$ defined by

$$\mathbb{P}_X(A) := \mathbb{P}(X^{-1}(A)) = \mathbb{P}(\{\omega \in \Omega | X(\omega) \in A\}) \quad \text{for all } A \in \mathcal{B},$$

where $\mathbb{P}_X$ is also a probability measure.

Note that the σ-algebra $\{X^{-1}(A) \mid A \in \mathcal{B}\}$ is a subset of $\mathcal{F}$ and only characterizes the probabilistic events related to the random vector. Thus, such a σ-algebra is usually referred to as the σ-algebra generated by X, and is denoted by $\sigma(X)$. The restriction of $\mathbb{P}$ on $\sigma(X)$, that is, $\mathbb{P}_X$, only describes the probability law related to X. Moreover, $\mathbb{P}_X$ determines a unique distribution function in $\mathbb{R}$.

Definition A.10 (distribution functions). If $X : \Omega \to \mathbb{R}$ is a random variable on $(\Omega, \mathcal{F}, \mathbb{P})$, then its *distribution function* is a mapping $F_X : \mathbb{R} \to \mathbb{R}^+$, defined by

$$F_X(x) = \mathbb{P}_X(X \le x) = \mathbb{P}(\{\omega \in \Omega | X(\omega) \le x\}) \quad \text{for all } x \in \mathbb{R},$$

which is right-continuous and monotonically increasing and satisfies

$$\lim_{x \to +\infty} F_X(x) = 1, \quad \lim_{x \to -\infty} F_X(x) = 0,$$
$$\mathbb{P}_X(a < x \le b) = F_X(b) - F_X(a) \ge 0, \quad \text{for all } a \le b \in \mathbb{R},$$
$$\mathbb{P}_X(a \le x < b) = F_X(b) - F_X(a) + \mathbb{P}_X(x = a) - \mathbb{P}_X(x = b).$$

From Definitions A.9 and A.10, we see that a random variable uniquely determines its image measure $\mathbb{P}_X$ and the distribution function F_X. Conversely, F_X uniquely determines a measure $\mathbb{P}_X$, but there exist multiple random variables having the same image measure $\mathbb{P}_X$. Those random variables are called *identically distributed* random variables.

Definition A.11 (probability density functions). Let X denote a random variable in $(\Omega, \mathcal{F}, \mathbb{P})$ and let F_X be its probability distribution function which is absolutely continuous in $\mathbb{R}$. Then, there exists an integrable function $f_X(x)$, referred to as the *probability density function* of X such that

$$F_X(b) - F_X(a) = \int_a^b f_X(x)\,\mathrm{d}x, \quad a \le b.$$

$f_X(x)$ is, in fact, the Radon–Nikodym derivative of F_X.

The Doob–Dynkin lemma

The following measurability result is extremely useful; it is a special case of a result usually referred to as the *Doob–Dynkin lemma.*

Lemma A.12 (Doob–Dynkin). Let $(\Omega, \mathcal{F})$ and $(\Theta, \mathcal{A})$ denote measure spaces and let $X : \Omega \to \Theta$ be measurable. Then, a function $Y : \Omega \to \mathbb{R}$ is $\sigma(X)$-measurable if and only if there exists a function $g : \Theta \to \mathbb{R}$ such that $Y = g(X)$.

A number of specializations of this result are possible. If the measure space $\Theta = \mathbb{R}$ and we use the Borel σ-algebra $\mathcal{A} = \mathcal{B}$ of $\mathbb{R}$, then there is a Borel-measurable $g : \mathbb{R} \to \mathbb{R}$, which satisfies the requirements. This yields the following result.

Corollary A.13. Let Ω be a measure space. If $X, Y : \Omega \to \mathbb{R}$ are two given measurable functions, then Y is $\sigma(X)$-measurable if and only if there exists a Borel-measurable function $g : \mathbb{R} \to \mathbb{R}$ such that $Y = g(X)$.

If $\mathcal{A}$ is replaced by the larger σ-algebra of all (completion of $\mathcal{A}$) Lebesgue-measurable subsets of $\mathbb{R}$, then g will be a Lebesgue-measurable function.

A.3. Integration and moment operators of random variables

Integrability

For a random variable $X : (\Omega, \mathcal{F}) \to (\mathbb{R}, \mathcal{B})$, the integral of X with respect to $\mathbb{P}$ over a subdomain $D \subset \Omega$ is defined by

$$\int_D X(\omega)\mathbb{P}(\mathrm{d}\omega) = \int_\Omega \mathcal{I}_D(\omega)X(\omega)\mathbb{P}(\mathrm{d}\omega),$$

where $\mathcal{I}_D(\omega)$ is the characteristic function of D. If such an integral exists and is finite, then X is $\mathbb{P}$-*integrable over* D.

Moments of a random variable

Definition A.14 (expectation). Let X denote a random variable on a probability space $(\Omega, \mathcal{F}, \mathbb{P})$. Then

$$\mathbb{E}(X) := \int_\Omega X(\omega)\mathbb{P}(\mathrm{d}\omega)$$

is called the *expectation* of X. If X is $\mathbb{P}$-integrable over Ω, then the expectation of X is finite.

Based on this definition, we see that for any Borel-measurable function $Y : \mathbb{R} \to \mathbb{R}$ which is $\mathbb{P}_X$-integrable, we have

$$\mathbb{E}(Y \circ X) = \int_{\mathbb{R}} Y \, \mathrm{d}\mathbb{P}_X.$$

In particular, when $Y = X$, the expectation of X can be represented by an integral over $\mathbb{R}$, that is,

$$\mathbb{E}(X) = \int_{\mathbb{R}} X \, \mathbb{P}_X(\mathrm{d}X).$$

Definition A.15 (the space $L^q(\Omega)$). In the probability space $(\Omega, \mathcal{F}, \mathbb{P})$, for $q > 1$, we denote by $L^q(\Omega)$ the collection of random variables X defined on $(\Omega, \mathcal{F}, \mathbb{P})$ such that

$$\mathbb{E}\big[|X|^q\big] \leq \infty.$$

Definition A.16 (moments of order q). Let X denote a random variable on a probability space $(\Omega, \mathcal{F}, \mathbb{P})$ and $Y = X^q$ for $q \geq 1$. The expectation of Y is called the *moment of order* q of X, and is given by

$$\mathbb{E}(X^q) := \int_\Omega X^q(\omega)\mathbb{P}(\mathrm{d}\omega) = \int_{\mathbb{R}} x^q \,\mathrm{d}F_X(x), \tag{A.5}$$

where $x \in \mathbb{R}$. If $X \in L^q(\Omega)$, then its moment of order q is finite.

A.4. Random vectors and their probability distributions

Similar to the definition of a random variable, a random vector defined on the probability space $(\Omega, \mathcal{F}, \mathbb{P})$ is a vector of scalar random variables, denoted by $\boldsymbol{X}(\omega) = (X_1(\omega), \ldots, X_N(\omega))$, whose components are defined on the same probability space $(\Omega, \mathcal{F}, \mathbb{P})$. The *image measure* of $\boldsymbol{X}$, denoted by $\mathbb{P}_{\boldsymbol{X}}$, is a measure on the Borel space $(\mathbb{R}^N, \mathcal{B}^N)$ defined by

$$\mathbb{P}_{\boldsymbol{X}}(A) := \mathbb{P}(\boldsymbol{X}^{-1}(A)) = \mathbb{P}(\{\omega \in \Omega | \boldsymbol{X}(\omega) \in A\}) \quad \text{for all } A \in \mathcal{B}^N.$$

Definition A.17 (joint and marginal distribution functions). The *joint distribution function* of $\boldsymbol{X}$ is the direct extension of the distribution function of the scalar random variable, that is,

$$F_{\boldsymbol{X}}(\boldsymbol{x}) = \mathbb{P}(\{\omega \in \Omega | X_1(\omega) \leq x_1, \ldots, X_N(\omega) \leq x_N\}) \text{ for all } \boldsymbol{x} \in \mathbb{R}^N. \tag{A.6}$$

The *marginal distribution function* of X_n, denoted by $F_{X_n}(x_n)$, is defined by

$$F_{X_n}(x_n) = F_{\boldsymbol{X}}(\infty, \ldots, \infty, x_n, \infty, \ldots, \infty).$$

Definition A.18 (joint and marginal density functions). The *joint density function* of $\boldsymbol{X}$ is the direct extension of the definition of the density function of the scalar random variable, that is, the Radon–Nikodym derivative of $F_{\boldsymbol{X}}$ represented by

$$f_{\boldsymbol{X}}(\boldsymbol{x}) := \frac{\partial^N F_{\boldsymbol{X}}(\boldsymbol{x})}{\partial x_1 \cdots \partial x_N}.$$

The *marginal density function* of X_n, denoted by $f_{X_n}(x_n)$, is defined by

$$f_{X_n}(x_n) := \int_{\mathbb{R}^{N-1}} F_{\boldsymbol{X}}(x_1, \ldots, x_N)\,\mathrm{d}x_1 \cdots \mathrm{d}x_{n-1}\,\mathrm{d}x_{n+1} \cdots \mathrm{d}x_N.$$

A.5. Independence and correlation of random variables

Definition A.19 (independent events). A family of events $\{A_i\}_{i\in I}$ in $\mathcal{F}$ is called *independent* with respect to the measure $\mathbb{P}$ if, for every non-empty, finite index set $\{i_1,\dots,i_n\}\subset I$, we have

$$\mathbb{P}(A_{i_1}\cap\cdots\cap A_{i_n}) = \mathbb{P}(A_{i_1})\times\cdots\times\mathbb{P}(A_{i_n}).$$

Definition A.20 (independent random variables). Let $\{X_n\}_{n=1}^N$ denote N random variables in $(\Omega,\mathcal{F},\mathbb{P})$. If, for any $\boldsymbol{x}=(x_1,\dots,x_N)\in\mathbb{R}^N$,

$$\mathbb{P}\Big(\bigcap_{n=1}^N\{X_n\le x_n\}\Big) = \prod_{n=1}^N \mathbb{P}(X_n\le x_n), \tag{A.7}$$

then $\{X_n\}_{n=1}^N$ are called *independent* random variables.

An equivalent definition of independent random variables is that X_n for $n=1,\dots,N$ are independent if and only if their joint distribution is the product of their marginal distributions, that is,

$$F_{\boldsymbol{X}}(\boldsymbol{x}) = \prod_{n=1}^N F_{X_n}(x_n).$$

On the other hand, if the joint density function $f_{\boldsymbol{X}}(\boldsymbol{x})$ exists, then it also satisfies the product rule, that is,

$$f_{\boldsymbol{X}}(\boldsymbol{x}) := \prod_{n=1}^N f_{X_n}(x_n).$$

Definition A.21 (covariance). Let $\boldsymbol{X}=(X_1,\dots,X_N)^\top$ denote an N-dimensional random vector on a probability space $(\Omega,\mathcal{F},\mathbb{P})$. Then the matrix

$$\mathbb{COV}(\boldsymbol{X}) := \mathbb{E}\big[(\boldsymbol{X}-\mathbb{E}(\boldsymbol{X}))(\boldsymbol{X}-\mathbb{E}(\boldsymbol{X}))^\top\big]\in\mathbb{R}^{N\times N}$$

is called the *covariance* matrix of the random vector $\boldsymbol{X}$. The covariance $\mathbb{COV}(\boldsymbol{X})<\infty$ if and only if $\boldsymbol{X}$ is square-integrable. Each of the diagonal entries of $\mathbb{COV}(\boldsymbol{X})(\boldsymbol{X})$ is called the *variance* of X_n for $n=1,\dots,N$ and is denoted by

$$\mathbb{VAR}(X_n) := \mathbb{E}\big[(X_n-\mathbb{E}(X_n))^2\big] = \mathbb{E}(X_n^2)-\mathbb{E}(X_n)^2.$$

Each of the off-diagonal entries

$$c_{ij} = \mathbb{E}[(X_i-\mathbb{E}(X_i))(X_j-\mathbb{E}(X_j))]$$

is the covariance of (X_i,X_j). If $c_{ij}=0$ for $i\neq j$, then X_i and X_j are said to be *uncorrelated*.

A.6. Product probability space

We now consider of a family of probability spaces $(\Omega_k, \mathcal{F}_k, \mathbb{P}_k)$ for $k = 1, \ldots, Kd$, from which we would like to build a product space $(\Omega, \mathcal{F}, \mathbb{P})$ represented by

$$(\Omega, \mathcal{F}, \mathbb{P}) := \prod_{i=1}^{d} (\Omega_i, \mathcal{F}_i, \mathbb{P}_i),$$

where Ω, $\mathcal{F}$, and $\mathbb{P}$ are the product sample spaces, product σ-algebra, and product measure, respectively. Their definitions are given below.

Definition A.22 (product sample space). The product space Ω is defined by

$$\Omega := \Omega_1 \times \cdots \times \Omega_K = \{(\omega_1, \ldots, \omega_K) |\ \omega_k \in \Omega_k \quad \text{for } k = 1, \ldots, K\}.$$

Definition A.23 (product σ-algebra). Let $(\Omega_k, \mathcal{F}_k)$, for $k = 1, \ldots, K$, denote K measure spaces and define the collection of subsets $\mathcal{C}$ in $\prod_{k=1}^{K} \Omega_k$ by

$$\mathcal{C} := \left\{ \prod_{i=1}^{d} A_k \,\middle|\, A_k \in \mathcal{F}_k \quad \text{for } k = 1, \ldots, K \right\}.$$

Then, the *product σ-algebra* is the σ-algebra generated by $\mathcal{C}$, that is,

$$\mathcal{F} := \mathcal{F}_1 \times \cdots \times \mathcal{F}_d = \sigma(\mathcal{C}).$$

Theorem A.24 (product probability measure). Let $(\Omega_k, \mathcal{F}_k, \mathbb{P}_k)$, for $k = 1, \ldots, K$, denote K probability spaces. Then there exists a unique product probability measure $\mathbb{P}$ defined on the product σ-algebra $\otimes_{k=1}^{K} \mathcal{F}_k$ satisfying

$$\mathbb{P}\left(\prod_{k=1}^{K} A_k \right) = \mathbb{P}_1(A_1) \cdots \mathbb{P}_K(A_K) \quad \text{for } A_k \in \mathcal{F}_k, \quad k = 1, \ldots, K.$$

For a general event $A \in \mathcal{F} = \otimes_{k=1}^{K} \mathcal{F}_k$, $\mathbb{P}(A)$ is defined by

$$\mathbb{P}(A) = \int_{\Omega_{i_K}} \left(\cdots \left(\int_{\Omega_{i_1}} \mathcal{I}_A(\omega_1, \ldots, \omega_K) \mathbb{P}_{i_1}(\mathrm{d}\omega_{i_1}) \right) \cdots \right) \mathbb{P}_{i_d}(\mathrm{d}\omega_{i_d}).$$

where $(i_1, \ldots, i_K)$ is an arbitrary reordering of $(1, \ldots, K)$ and $\mathcal{I}_A$ is the characteristic function of the event A.

Theorem A.25 (Fubini's theorem). Let $(\Omega_k, \mathcal{F}_k, \mathbb{P}_k)$, for $k = 1, \ldots, K$, denote K probability spaces and let $(\Omega, \mathcal{F}, \mathbb{P})$ be the product probability space. If f is a measurable function on $\mathcal{F} = \mathcal{F}_1 \times \cdots \times \mathcal{F}_K$ and is integrable

with respect to the product measure $\mathbb{P} = \mathbb{P}_1 \times \cdots \times \mathbb{P}_K$, then

$$\int_{\Omega_1 \times \cdots \times \Omega_K} f(\omega_1, \ldots, \omega_K)\ \mathrm{d}(\mathbb{P}_1 \times \cdots \times \mathbb{P}_K)$$
$$= \int_{\Omega_{i_K}} \left(\cdots \left(\int_{\Omega_{i_1}} f(\omega_1, \ldots, \omega_K) \mathbb{P}_{i_1}(\mathrm{d}\omega_{i_1}) \right) \cdots \right) \mathbb{P}_{i_d}(\mathrm{d}\omega_{i_K}),$$

where $(i_1, \ldots, i_K)$ is an arbitrary reordering of $(1, \ldots, K)$.

B. Random fields

In this article we consider numerical methods for partial differential equations with random input data whose solutions are functions of spatial and random variables. Thus, the notion of random variables need to be extended by incorporating a spatial dependence. For convenience, we use the notations D to represent the spatial domain and $\mathbf{x} = (x_1, \ldots, x_d)$ to represent the spatial coordinates. Then, in the probability space $(\Omega, \mathcal{F}, \mathbb{P})$, a stochastic process is a collection of random variables

$$\{a(\mathbf{x}, \omega), \mathbf{x} \in D, \omega \in \Omega\}. \tag{B.1}$$

The term 'random field' usually refers to a stochastic process taking values in a Euclidean space[13] $\mathbb{R}^d$, $d = 1, 2, 3$. Because $\mathbf{x}$ is used to represent the spatial coordinates, we use 'random field' to refer to the process defined in (B.1). A random field can be viewed in two ways:

- for a fixed $\mathbf{x} \in D$, $a(\mathbf{x}, \cdot)$ is a random variable in Ω;
- for a fixed $\omega \in \Omega$, $a(\cdot, \omega)$ is a realization of the random field in D.

It is natural and useful to study the statistics of a random field. For example, the *expectation* of a random field $a(\mathbf{x}, \omega)$ is given by

$$\overline{a}(\mathbf{x}) := \mathbb{E}[a(\mathbf{x}, \cdot)] \quad \text{for each } \mathbf{x} \in D$$

and the *covariance function* is given by

$$\mathbb{COV}(\mathbf{x}, \mathbf{x}') := \mathbb{E}\big[(a(\mathbf{x}, \cdot) - \overline{a}(\mathbf{x}))(a(\mathbf{x}', \cdot) - \overline{a}(\mathbf{x}'))\big]$$

for each pair $\mathbf{x}, \mathbf{x}' \in D$.

B.1. Karhunen–Loève expansions

Given a collection of real-valued functions $\{b_n(\mathbf{x})\}_{n=1}^{\infty}$ defined for $\mathbf{x} \in D$ and a collection of uncorrelated random variables

$$\{\widehat{\xi}_n(\omega)\}_{n=1}^{\infty}$$

[13] Time-dependent random fields are in even greater use.

with, for convenience, mean zero and variances $\{\sigma_n^2\}_{n=1}^{\infty}$, the linear combination

$$a(\mathbf{x},\omega) = \sum_{n=1}^{\infty} b_n(\mathbf{x})\widehat{\xi}_n(\omega) \tag{B.2}$$

is a random field that can be used as a simple way to represent a given correlated random field as an infinite sum involving uncorrelated random variables. The covariance function of the random field (B.2) is given by

$$\mathbb{COV}_a(\mathbf{x},\mathbf{x}') = \sum_{n=1}^{\infty} \sigma_n^2 b_n(\mathbf{x}) b_n(\mathbf{x}'). \tag{B.3}$$

The above structure is attractive because the random variables ξ_n are uncorrelated or independent so they are easy to handle in practice.

If we set $\xi_n(\omega) = \frac{1}{\sigma_n}\xi_n(\omega)$, then $\{\xi_n(\omega)\}_{n=1}^{\infty}$ are a collection of uncorrelated random variables having mean zero and variance 1. Also, (B.2) is now given by

$$a(\mathbf{x},\omega) = \sum_{n=1}^{\infty} \sigma_n b_n(\mathbf{x})\xi_n(\omega).$$

If the functions $\{b_n(\mathbf{x})\}_{n=1}^{\infty}$ are orthonormal, that is, if

$$\int_D b_n(\mathbf{x}) b_{n'}(\mathbf{x})\,\mathrm{d}\mathbf{x} = \delta_{nn'},$$

then

$$\begin{aligned}\int_D \mathbb{COV}_a(\mathbf{x},\mathbf{x}')b_n(\mathbf{x}')\,\mathrm{d}\mathbf{x} &= \int_D \left(\sum_{n'=1}^{\infty} \sigma_{n'}^2 b_{n'}(\mathbf{x})b_{n'}(\mathbf{x}')\right) b_n(\mathbf{x}')\,\mathrm{d}\mathbf{x} \\ &= \sum_{n'=1}^{\infty} \sigma_{n'}^2 b_{n'}(\mathbf{x}) \int_D b_{n'}(\mathbf{x}')b_n(\mathbf{x}')\,\mathrm{d}\mathbf{x} \\ &= \sum_{n'=1}^{\infty} \sigma_{n'}^2 b_{n'}(\mathbf{x})\delta_{nn'} = \sigma_n^2 b_n(\mathbf{x}).\end{aligned}$$

Thus, we see that σ_n^2 and $b_n(\mathbf{x})$, $n = 1, 2, \ldots$, are the eigenpairs of the correlation function $\mathbb{COV}_a(\mathbf{x},\mathbf{x}')$.

What we have shown is that, given the covariance function $\mathbb{COV}_a(\mathbf{x},\mathbf{x}')$ of a random field $a(\mathbf{x},\omega)$, random field can be expressed as the infinite sum

$$a(\mathbf{x},\omega) = \sum_{n=1}^{\infty} \sqrt{\lambda_n} b_n(\mathbf{x})\xi_n(\omega), \tag{B.4}$$

where $\{\lambda_n, b_n(\mathbf{x})\}_{n=1}^{\infty}$ denote the eigenpairs of the given covariance function and $\{\xi_n(\omega)\}_{n=1}^{\infty}$ are uncorrelated random variables with mean zero and unit variance. The expansion (B.4) is well known as the Karhunen–Loève (KL)

expansion (Loève 1977) of a random field. KL expansions are also known as proper orthogonal decompositions (POD) and principal component analyses (PCA) in finite-dimensional settings.

Truncated KL expansions provide a means for approximating random fields, that is, we have

$$a(\mathbf{x},\omega) \approx a_N(\mathbf{x},\omega) = \sum_{n=1}^{N} \sqrt{\lambda_n} b_n(\mathbf{x}) \xi_n(\omega).$$

We have that

$$\mathbb{COV}_{a_N}(\mathbf{x},\mathbf{x}') = \sum_{n=1}^{N} \lambda_n b_n(\mathbf{x}) b_n(\mathbf{x}').$$

The convergence of $\mathbb{COV}_{a_N}$ to $\mathbb{COV}_a$ as $N \to \infty$ is guaranteed by Mercer's theorem.

Theorem B.1 (Mercer's theorem). Let $D \subset \mathbb{R}^d$ be closed, let μ be a strictly positive Borel measure on D, let $\mathbb{COV}_a(\mathbf{x},\mathbf{x}')$ be a continuous function on $D \times D$ which is *symmetric*,

$$\mathbb{COV}_a(\mathbf{x},\mathbf{x}') = \mathbb{COV}_a(\mathbf{x}',\mathbf{x}) \quad \text{for all } \mathbf{x},\mathbf{x}' \in D,$$

non-negative definite,

$$\int_D \int_D \mathbb{COV}_a(\mathbf{x},\mathbf{x}') v(\mathbf{x}') v(\mathbf{x}) \, \mathrm{d}\mathbf{x}' \, \mathrm{d}\mathbf{x} \geq 0 \quad \text{for all } v(\mathbf{x}),$$

and *square-integrable*,

$$\int_D \int_D C_a(\mathbf{x},\mathbf{x}')^2 \, \mathrm{d}\mu(\mathbf{x}) \, \mathrm{d}\mu(\mathbf{x}') < \infty.$$

Then, we have that

$$\lim_{N\to\infty} \max_{(\mathbf{x},\mathbf{x}')\in D\times D} \left| \mathbb{COV}_a(\mathbf{x},\mathbf{x}') - \sum_{n=1}^{N} \lambda_n b_n(\mathbf{x}) b_n(\mathbf{x}') \right| = 0,$$

where λ_n and $b_n(\mathbf{x})$ are the eigenpairs of $\mathbb{COV}_a(\mathbf{x},\mathbf{x}')$.

Moreover, the truncation error decreases monotonically with the number of terms in the expansion and the convergence is inversely proportional to the correlation length and depends on the regularity of the covariance kernel. The decay of the eigenvalues λ_n is given in the following theorem.

Theorem B.2. Let $\mathbb{COV}_a(\mathbf{x},\mathbf{x}') \in L^2(D \times D)$ be piecewise analytic on $D \times D$ and let $\{\lambda_n\}_{n=1}^{\infty}$ be the eigenvalue sequence. Then, there exist constants $c_1 > 0$ and $c_2 > 0$ that are independent of n such that

$$0 \leq \lambda_n \leq c_1 \exp(-c_2 \, n^{1/N}) \quad \text{for all } n \geq 1.$$

Theorem B.3. Let $\mathbb{COV}_a(\mathbf{x}, \mathbf{x}') \in L^2(D \times D)$ be piecewise H_0^k with $k > 0$ and let $\{\lambda_n\}_{n=1}^{\infty}$ be the eigenvalue sequence. Then, there exists a constant $c_3 > 0$ that is independent of n such that

$$0 \leq \lambda_n \leq c_3 \; n^{-k/N} \quad \text{for all } n \geq 1.$$

C. White noise inputs

The value of a white noise random field at every point in space is independently chosen according to a centred[14] Gaussian PDF with variance σ^2. The covariance function corresponding to a white noise random field is given by

$$\mathbb{COV}_{\text{white}}(\mathbf{x}, \mathbf{x}') = \sigma^2 \delta(\mathbf{x} - \mathbf{x}'), \tag{C.1}$$

where $\delta(\cdot)$ denotes the Dirac delta function. Thus, the variance $\mathbb{VAR}_{\text{white}}(\mathbf{x})$ of white noise is infinite, so white noise cannot describe a real process. Notwithstanding this observation, white noise random fields are the most common random inputs used in the partial differential equation (PDE) setting. White noise random fields are infinite stochastic processes so that, in any computer simulation, they have to be approximated in terms of a finite number of random parameters. Among the means available for defining *discretized white noise* in the PDE setting, the most popular are grid-based methods.

To define a single realization of grid-based discretized white noise, we first subdivide the spatial domain[15] D into N non-overlapping, covering subdomains $\{D_n\}_{n=1}^N$. Then, for some constants $\{b_n\}_{n=1}^N$, we seek a *piecewise constant approximation* of a white noise random field of the form

$$\eta_{\text{white}}^N(\mathbf{x}; \boldsymbol{y}) = \sum_{n=1}^{N} b_n 1_n(\mathbf{x}) y_n, \tag{C.2}$$

where, for each $n = 1, \ldots, N$, $1_n(\mathbf{x})$ denotes the indicator function corresponding to the subdomain D_n, y_n denotes an i.i.d. random number drawn from a standard Gaussian PDF, and $\boldsymbol{y} = (y_1, y_2, \ldots, y_N)^\top$ denotes the vector or random samples.

The covariance of $\eta_{\text{white}}^N(\mathbf{x}; \boldsymbol{y})$ is given by

$$\begin{aligned}
\mathbb{COV}_{\text{white}}^N(\mathbf{x}, \mathbf{x}') &= \int_\Gamma \eta_{\text{white}}^N(\mathbf{x}; \boldsymbol{y}) \eta_{\text{white}}^N(\mathbf{x}'; \boldsymbol{y}) \rho_G(\boldsymbol{y}) \, \mathrm{d}\boldsymbol{y} \\
&= \int_\Gamma \sum_{n=1}^{N} \sum_{n'=1}^{N} b_n 1_n(\mathbf{x}) y_n b_{n'} 1_{n'}(\mathbf{x}') y_{n'} \rho_G(\boldsymbol{y}) \, \mathrm{d}\boldsymbol{y}
\end{aligned}$$

[14] We assume, without loss of generality, that the random field has zero expected value.

[15] Defining discretized white noise with respect to spatial subdomains is well suited to finite element and finite volume spatial discretizations of PDEs. For finite difference methods, a node-based discretization is more appropriate.

$$
\begin{aligned}
&= \sum_{n=1}^{N} \sum_{n'=1}^{N} b_n 1_n(\mathbf{x}) b_{n'} 1_{n'}(\mathbf{x}') \int_{\Gamma} y_n y_{n'} \rho_G(\boldsymbol{y}) \, \mathrm{d}\boldsymbol{y} \\
&= \sum_{n=1}^{N} \sum_{n'=1}^{N} b_n 1_n(\mathbf{x}) b_{n'} 1_{n'}(\mathbf{x}') \delta_{nn'} \\
&= \sum_{n=1}^{N} b_n^2 1_n(\mathbf{x}) 1_n(\mathbf{x}'),
\end{aligned}
$$

so that, for $n = 1, \ldots, N$,

$$
\mathbb{COV}^N_{\text{white}}(\mathbf{x}, \mathbf{x}') = \begin{cases} b_n^2 & \text{if } \mathbf{x}, \mathbf{x}' \in D_n, \\ 0 & \text{if } \mathbf{x} \in D_n \text{ and } \mathbf{x}' \notin D_n. \end{cases} \tag{C.3}
$$

Because pointwise values of the covariance (C.1) are not defined, we determine, for $n = 1, \ldots, N$, the coefficient b_n by matching the averages of (C.1) and (C.3) over D_n. This results in $b_n^2 |D_n| = \sigma^2$, where $|D_n|$ denotes the volume of D_n. Then, from (C.2), the *discretized white noise random field is given by*

$$
\eta^N_{\text{white}}(\mathbf{x}; \boldsymbol{y}) = \sigma \sum_{n=1}^{N} \frac{1}{\sqrt{|D_n|}} \chi_n(\mathbf{x}) y_n,
$$

so that, via discretization, white noise has been reduced to the case of N random parameters. Note that *the number of random parameters N is intimately tied to the spatial grid size.* In fact, for ordinary meshes in domains $D \subset \mathbb{R}^d$, we have that $N = O(\frac{1}{h^d})$. This should be contrasted with the coloured noise case, for which there is at most a weak connection between the number of parameters and the spatial grid size.

In one dimension, for a uniform grid of size h, we have the well-known formula

$$
\eta^N_{\text{white}}(x; \boldsymbol{y}) = \frac{\sigma}{\sqrt{h}} \sum_{n=1}^{N} \chi_n(x) y_n
$$

for approximating a white noise random field. This formula is especially well known when x is interpreted as a time variable.

The variance of the discretized white noise random field $\eta^N_{\text{white}}(x; \boldsymbol{y})$ is given by, for $n = 1, \ldots, N$,

$$
\mathbb{VAR}^N_{\text{white}}(\mathbf{x}) = \frac{\sigma^2}{|D_n|} \quad \text{for } \mathbf{x} \in D_n
$$

so that, unlike for white noise itself, *the variance of discretized white noise is finite.* However, note that as the grid size is reduced, that is, as $|D_n| \to 0$, we do have that $\mathbb{VAR}^N_{\text{white}}(\mathbf{x}) \to \infty$. Furthermore, although a white noise random field is uncorrelated, *the discretized white noise is a correlated*

random field. In fact, we have that

$$\mathbb{COV}^N_{\text{white}}(\mathbf{x},\mathbf{x}') = \begin{cases} \frac{\sigma^2}{|D_n|} & \text{if } \mathbf{x},\mathbf{x}' \in D_n, \\ 0 & \text{if } \mathbf{x} \in D_n \text{ and } \mathbf{x}' \notin D_n, \end{cases}$$

so that the correlation function for the discretized white noise random field $\eta^N_{\text{white}}(x;\boldsymbol{y})$ is given by, for $n = 1,\ldots,N$,

$$\begin{aligned} \mathbb{COR}^N_{\text{white}}(\mathbf{x},\mathbf{x}') &= \frac{\mathbb{COV}^N_{\text{white}}(\mathbf{x},\mathbf{x}')}{\sqrt{\mathbb{VAR}^N_{\text{white}}(\mathbf{x})\mathbb{VAR}^N_{\text{white}}(\mathbf{x}')}} \\ &= \begin{cases} 1 & \text{if } \mathbf{x},\mathbf{x}' \in D_n, \\ 0 & \text{if } \mathbf{x} \in D_n \text{ and } \mathbf{x}' \notin D_n. \end{cases} \end{aligned}$$

Thus, all pairs of points within a subdomain D_n are perfectly correlated, whereas any two points in different subdomains are uncorrelated.

In what sense is the discretized white noise field $\eta^N_{\text{white}}(x;\boldsymbol{y})$ an approximation to a white noise field? Certainly, pointwise convergence or even $L^2(D)$ convergence is out of the question because a white noise field is not square-integrable. What can be shown is convergence of the covariance function in the sense that

$$\lim_{N\to\infty,\, \max_{n=1,\ldots,N}|D_n|\to 0} \mathbb{COV}^N_{\text{white}}(\mathbf{x},\mathbf{x}') = \sigma^2\delta(\mathbf{x}-\mathbf{x}') = \mathbb{COV}_{\text{white}}(\mathbf{x},\mathbf{x}').$$

This result follows because, for any function $g(\mathbf{x})$ that is continuous over D, we have that

$$\lim_{N\to\infty} \frac{\sigma^2}{|D_{n_N}|} \int_{D_{n_N}} g(\mathbf{x})\,\mathrm{d}\mathbf{x} = \sigma^2 g(\mathbf{x}'),$$

where $\{D_{n_N}\}_{N\to\infty}$ denotes a sequence of subdomains that contain a point $\mathbf{x}' \in D$ and such that $\max_{n=1,\ldots,N}|D_n| \to 0$.

Acknowledgements

We would like to thank Drs John Burkardt and Miroslav Stoyanov, as well as Mr Nick Dexter, for generating several insightful plots used throughout.

The preparation of the article as well as the research of the authors on topics related to this article were supported in part by the Office of Science of the US Department of Energy under grant numbers DE-SC0010678, ERKJ259, and ERKJE45; by the US Air Force Office of Scientific Research under grant numbers FA9550-11-1-0149 and 1854-V521-12; and by the Laboratory Directed Research and Development program at the Oak Ridge National Laboratory which is operated by UT-Battelle, LLC, for the US Department of Energy under Contract DE-AC05-00OR22725.

REFERENCES[16]

S. Acharjee and N. Zabaras (2007), 'A non-intrusive stochastic Galerkin approach for modeling uncertainty propagation in deformation processes', *Comput. Struct.* **85**, 244–254.

N. Agarwal and N. R. Aluru (2009), 'A domain adaptive stochastic collocation approach for analysis of MEMS under uncertainties', *J. Comput. Phys.* **228**, 7662.

M. Ainsworth and J.-T. Oden (2000), *A Posteriori Error Estimation in Finite Element Analysis*, Wiley.

J. Dongarra, J. Hittinger, J. Bell, L. Chacon, R. Falgout, M. Heroux, P. Hovland, E. Ng, C. Webster, and S. Wild (2013), Applied mathematics research for exascale computing. Technical report, US Department of Energy.

R. Askey and J. A. Wilson (1985), *Some Basic Hypergeometric Orthogonal Polynomials that Generalize Jacobi Polynomials*, Vol. 319 of *Memoirs of the American Mathematical Society*, AMS.

I. M. Babuška and P. Chatzipantelidis (2002), 'On solving elliptic stochastic partial differential equations', *Comput. Methods Appl. Mech. Engrg* **191**, 4093–4122.

I. M. Babuška and J. Chleboun (2002), 'Effects of uncertainties in the domain on the solution of Neumann boundary value problems in two spatial dimensions', *Math. Comp.* **71**, 1339–1370.

I. M. Babuška and J. Chleboun (2003), 'Effects of uncertainties in the domain on the solution of Dirichlet boundary value problems', *Numer. Math.* **93**, 583–610.

I. M. Babuška and J. T. Oden (2006), 'The reliability of computer predictions: Can they be trusted?', *Internat. J. Numer. Anal. Model.* **3**, 255–272.

I. M. Babuška and T. Strouboulis (2001), *The Finite Element Method and its Reliability*, Numerical Mathematics and Scientific Computation, Oxford Science Publications.

I. M. Babuška, K. M. Liu and R. Tempone (2003), 'Solving stochastic partial differential equations based on the experimental data', *Math. Models Methods Appl. Sci.* **13**, 415–444.

I. M. Babuška, F. Nobile and R. Tempone (2005*a*), 'Worst-case scenario analysis for elliptic problems with uncertainty', *Numer. Math.* **101**, 185–219.

I. M. Babuška, F. Nobile and R. Tempone (2007*a*), 'A stochastic collocation method for elliptic partial differential equations with random input data', *SIAM J. Numer. Anal.* **45**, 1005–1034.

I. Babuška, F. Nobile and R. Tempone (2007*b*), 'Reliability of computational science', *Numer. Methods Partial Diff. Equations* **23**, 753–784.

I. M. Babuška, F. Nobile and R. Tempone (2008), 'A systematic approach to model validation based on Bayesian updates and prediction related rejection criteria', *Comput. Methods Appl. Mech. Engrg* **197**, 2517–2539.

[16] The URLs cited in this work were correct at the time of going to press, but the publisher and the authors make no undertaking that the citations remain live or are accurate or appropriate.

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

I. M. Babuška, R. Tempone and G. E. Zouraris (2005*b*), 'Solving elliptic boundary value problems with uncertain coefficients by the finite element method: The stochastic formulation', *Comput. Methods Appl. Mech. Engrg* **194**, 1251–1294.

A. Barth and A. Lang (2012), 'Multilevel Monte Carlo method with applications to stochastic partial differential equations', *Internat. J. Comput. Math.* **89**, 2479–2498.

A. Barth, A. Lang and C. Schwab (2013), 'Multilevel Monte Carlo method for parabolic stochastic partial differential equations', *BIT Numer. Math.* **53**, 3–27.

A. Barth, C. Schwab and N. Zollinger (2011), 'Multi-level Monte Carlo finite element method for elliptic PDEs with stochastic coefficients', *Numer. Math.* **119**, 123–161.

M. J. Bayarri, J. O. Berger, R. Paulo, J. Sacks, J. Cafeo, J. Cavendish, C. H. Lin and J. Tu (2007), 'A framework for validation of computer models', *Technometrics* **49**, 138–154.

J. L. Beck and S. K. Au (2002), 'Bayesian updating of structural models and reliability using Markov chain Monte Carlo simulation', *J. Engrg Mech.* **128**, 380–391.

J. Beck, F. Nobile, L. Tamellini and R. Tempone (2011), Stochastic spectral Galerkin and collocation methods for PDEs with random coefficients: A numerical comparison. In *Spectral and High Order Methods for Partial Differential Equations*, Vol. 76 of *Lecture Notes in Computational Science and Engineering*, Springer, pp. 43–62.

J. Beck, F. Nobile, L. Tamellini and R. Tempone (2014), 'Convergence of quasi-optimal stochastic Galerkin methods for a class of PDEs with random coefficients', *Comput. Math. Appl.* **67**, 732–751.

J. Beck, R. Tempone and F. Nobile (2012), 'On the optimal polynomial approximation of stochastic PDEs by Galerkin and collocation methods', *Math. Models Methods Appl. Sci.* **22**, 1250023.

Y. Ben-Haim (1996), *Robust Reliability in the Mechanical Sciences*, Springer.

F. E. Benth and J. Gjerde (1998*a*), 'Convergence rates for finite element approximations of stochastic partial differential equations', *Stochastics Stochastic Rep.* **63**, 313–326.

F. E. Benth and J. Gjerde (1998*b*), Numerical solution of the pressure equation for fluid flow in a stochastic medium. In *Stochastic Analysis and Related Topics VI: Geilo, 1996*, Vol. 42 of *Progress in Probability*, Birkhäuser, pp. 175–186.

A. Bernardini (1999), What are the random fuzzy sets and how to use them for uncertainty modelling in engineering systems? In *Whys and Hows in Uncertainty Modelling: Probability, Fuzziness and Anti-Optimization* (I. Elishakoff, ed.), Vol. 388 of *CISM Course and Lectures*, Springer, pp. 63–125.

G. E. P. Box (1973), *Bayesian Inference in Statistical Analysis*, Wiley.

M. Braack and A. Ern (2003), '*A posteriori* control of modeling errors and discretization errors', *Multiscale Model. Simul.* **1**, 221–238.

J. Breidt, T. Butler and D. Estep (2011), 'A measure-theoretic computational method for inverse sensitivity problems I: Method and analysis', *SIAM J. Numer. Anal.* **49**, 1836–1859.

S. C. Brenner and L. R. Scott (2008), *The Mathematical Theory of Finite Element Methods*, Springer.

H.-J. Bungartz and M. Griebel (2004), Sparse grids. In *Acta Numerica*, Vol. 13, Cambridge University Press, pp. 1–123.

J. Burkardt, M. Gunzburger and H.-C. Lee (2006*a*), 'Centroidal Voronoi tessellation-based reduced-order modeling of complex systems', *SIAM J. Sci. Comput.* **28**, 459–484.

J. Burkardt, M. Gunzburger and H.-C. Lee (2006*b*), 'POD and CVT-based reduced-order modeling of Navier–Stokes flows', *Comp. Meth. Appl. Mech. Engrg* **196**, 337–355.

J. Burkardt, M. Gunzburger and C. G. Webster (2007), 'Reduced order modeling of some nonlinear stochastic partial differential equations', *Internat. J. Numer. Anal. Model.* **4**, 368–391.

C.-J. Chang and V. Joseph (2013), 'Model calibration through minimal adjustments', *Technometrics*, published online.

J. Charrier, R. Scheichl and A. Teckentrup (2013), 'Finite element error analysis of elliptic PDEs with random coefficients and its application to multilevel Monte Carlo methods', *SIAM J. Numer. Anal.* **51**, 322–352.

S. H. Cheung and J. L. Beck (2010), Comparison of different model classes for Bayesian updating and robust predictions using stochastic state-space system models. In *Safety, Reliability and Risk of Structures, Infrastructures and Engineering Systems*, CRC Press, pp. 1–8.

S. H. Cheung, T. A. Oliver, E. E. Prudencio, S. Prudhomme and R. D. Moser (2011), 'Bayesian uncertainty analysis with applications to turbulence modeling', *Reliab. Engrg System Safety* **96**, 1137–1149.

J. Ching and J. L. Beck (2004), 'Bayesian analysis of the phase II IASC–ASCE structural health monitoring experimental benchmark data', *J. Engrg Mech.* **130**, 1233–1244.

P. G. Ciarlet (1978), *The Finite Element Method for Elliptic Problems*, North-Holland.

C. W. Clenshaw and A. R. Curtis (1960), 'A method for numerical integration on an automatic computer', *Numer. Math.* **2**, 197–205.

K. A. Cliffe, M. B. Giles, R. Scheichl and A. L. Teckentrup (2011), 'Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients', *Computing and Visualization in Science* **14**, 3–15.

A. Cohen, R. DeVore and C. Schwab (2011), 'Analytic regularity and polynomial approximation of parametric and stochastic elliptic PDE's', *Anal. Appl.* **9**, 11–47.

A. C. Cullen and H. C. Frey (1999), *Probabilistic Techniques Exposure Assessment*, Plenum.

M. Dauge and R. Stevenson (2010), 'Sparse tensor product wavelet approximation of singular functions', *SIAM J. Math. Anal.* **42**, 2203–2228.

M.-K. Deb (2000), Solution of stochastic partial differential equations (SPDEs) using Galerkin method: Theory and applications. PhD thesis, The University of Texas at Austin.

M. K. Deb, I. M. Babuška and J. T. Oden (2001), 'Solution of stochastic partial differential equations using Galerkin finite element techniques', *Comput. Methods Appl. Mech. Engrg* **190**, 6359–6372.

C. Desceliers, R. Ghanem and C. Soize (2005), 'Polynomial chaos representation of a stochastic preconditioner', *Internat. J. Numer. Methods Engrg* **64**, 618–634.

R. A. DeVore and G. G. Lorentz (1993), *Constructive Approximation*, Vol. 303 of *Grundlehren der Mathematischen Wissenschaften*, Springer.

A. Doostan and G. Iaccarino (2009), 'A least-squares approximation of partial differential equations with high-dimensional random inputs', *J. Comput. Phys.* **228**, 4332–4345.

A. Doostan and H. Owhadi (2011), 'A non-adapted sparse approximation of PDEs with stochastic inputs', *J. Comput. Phys.* **230**, 3015–3034.

A. Doostan, R. Ghanem and J. Red-Horse (2007), 'Stochastic model reduction for chaos representations', *Comput. Methods Appl. Mech. Engrg* **196**, 3951–3966.

Q. Du and M. Gunzburger (2002*a*), 'Grid generation and optimization based on centroidal Voronoi tessellations', *Appl. Math. Comput.* **133**, 591–607.

Q. Du and M. Gunzburger (2002*b*), Model reduction by proper orthogonal decomposition coupled with centroidal Voronoi tessellation. In *Proc. FEDSM'02*, ASME.

Q. Du and M. Gunzburger (2003), Centroidal Voronoi tessellation based proper orthogonal decomposition analysis. In *Control and Estimation of Distributed Parameter Systems* (W. Desch *et al.*, eds), Birkhäuser.

Q. Du, V. Faber and M. Gunzburger (1999), 'Centroidal Voronoi tessellations: Applications and algorithms', *SIAM Review* **41**, 637–676.

Q. Du, M. Gunzburger and L. Ju (2002), 'Probabilistic algorithms for centroidal Voronoi tessellations and their parallel implementation', *Parallel Comput.* **28**, 1477–1500.

Q. Du, M. Gunzburger and L. Ju (2003*a*), 'Constrained centroidal Voronoi tessellations for surfaces', *SIAM J. Sci. Comput.* **24**, 1488–1506.

Q. Du, M. Gunzburger and L. Ju (2003*b*), 'Voronoi-based finite volume methods, optimal Voronoi meshes, and PDEs on the sphere', *Comput. Methods Appl. Mech. Engrg* **192**, 3933–3957.

Q. Du, M. Gunzburger and L. Ju (2010), 'Advances in studies and applications of centroidal Voronoi tessellations', *Numer. Math. Theor. Meth. Appl.* **3**, 119–142.

Q. Du, M. Gunzburger, L. Ju and X. Wang (2006), 'Centroidal Voronoi tessellation algorithms for image compression, segmentation, and multichannel restoration', *J. Math. Imag. Vision* **24**, 177–194.

D. Dubois and H. Prade, eds (2000), *Fundamentals of Fuzzy Sets*, Vol. 7 of *Handbooks of Fuzzy Sets*, Kluwer.

V. K. Dzjadyk and V. V. Ivanov (1983), 'On asymptotics and estimates for the uniform norms of the Lagrange interpolation polynomials corresponding to the Chebyshev nodal points', *Analysis Mathematica* **9**, 85–97.

M. Eiermann, O. G. Ernst and E. Ullmann (2007), 'Computational aspects of the stochastic finite element method', *Computing and Visualization in Science* **10**, 3–15.

M. Eldred, C. G. Webster and P. G. Constantine (2008), Evaluation of non-intrusive approaches for Wiener–Askey generalized polynomial chaos. AIAA paper 1892.

I. Elishakoff and Y. Ren (2003), *Finite Element Methods for Structures With Large Variations*, Oxford University Press.

I. Elishakoff, ed. (1999), *Whys and Hows in Uncertainty Modelling: Probability, Fuzziness and Anti-Optimization*, Vol. 388 of *CISM Course and Lectures*, Springer.

H. Elman and C. Miller (2011), Stochastic collocation with kernel density estimation. Technical report, Department of Computer Science, University of Maryland.

H. C. Elman, O. G. Ernst and D. P. O'Leary (2001), 'A multigrid method enhanced by Krylov subspace iteration for discrete Helmholtz equations', *SIAM J. Sci. Comput.* **23**, 1291–1315.

H. C. Elman, C. W. Miller, E. T. Phipps and R. S. Tuminaro (2011), 'Assessment of collocation and Galerkin approaches to linear diffusion equations with random data', *Internat. J. Uncertainty Quantification* **1**, 19–33.

K. Eriksson, D. Estep, P. Hansbo and C. Johnson (1995), Introduction to computational methods for differential equations. In *Theory and Numerics of Ordinary and Partial Differential Equations*, Vol. IV of *Advances in Numerical Analysis*, Oxford University Press, pp. 77–122.
Theory and Numerics of Ordinary and Partial Differential Equations (Advances in Numerical Analysis Vol. 4) by M. Ainsworth and M. Marletta (20 Jul 1995)

O. G. Ernst and E. Ullmann (2010), 'Stochastic Galerkin matrices', *SIAM J. Matrix Anal. Appl.* **31**, 1848–1872.

O. G. Ernst, C. E. Powell, D. J. Silvester and E. Ullmann (2009), 'Efficient solvers for a linear stochastic Galerkin mixed formulation of diffusion problems with random data', *SIAM J. Sci. Comput.* **31**, 1424–1447.

S. Ferson, V. Kreinovich, L. Ginzburg, D. Mayers and K. Sentz (2003), Constructing probability boxed and Demster–Shafer structures. Sandia Report SAND 2002-4015, Sandia National Laboratories.

E. D. Fichtl, A. K. Prinja and J. S. Warsa (2009), Stochastic methods for uncertainty quantification in radiation transport. In *International Conference on Mathematics, Computational Methods and Reactor Physics.*

G. Fishman (1996), *Monte Carlo: Concepts, Algorithms, and Applications*, Springer Series in Operations Research and Financial Engineering, Springer.

J. Foo and G. E. Karniadakis (2010), 'Multi-element probabilistic collocation method in high dimensions', *J. Comput. Phys.* **229**, 1536–1557.

J. Foo, X. Wan and G. Karniadakis (2008), 'The multi-element probabilistic collocation method (ME-PCM): Error analysis and applications', *J. Comput. Phys.* **227**, 9572–9595.

P. Frauenfelder, C. Schwab and R. A. Todor (2005), 'Finite elements for elliptic problems with stochastic coefficients', *Comput. Methods Appl. Mech. Engrg* **194**, 205–228.

B. Ganapathysubramanian and N. Zabaras (2007), 'Sparse grid collocation schemes for stochastic natural convection problems', *J. Comput. Phys.* **225**, 652–685.

A. Gaudagnini and S. Neumann (1999), 'Nonlocal and localized analysis of conditional mean steady state flow in bounded, randomly nonuniform domains. Part 1: Theory and computational approach. Part 2: Computational examples', *Water Resour. Res.* **35**, 2999–3039.

W. Gautschi (2004), *Orthogonal Polynomials: Computation and Approximation*, Numerical Mathematics and Scientific Computation, Oxford Science Publications.

T. Gerstner and M. Griebel (1998), 'Numerical integration using sparse grids', *Numer. Algorithms* **18**, 209–232.

T. Gerstner and M. Griebel (2003), 'Dimension-adaptive tensor-product quadrature', *Computing* **71**, 65–87.

R. Ghanem (1999), 'Ingredients for a general purpose stochastic finite elements implementation', *Comput. Methods Appl. Mech. Engrg* **168**, 19–34.

R. Ghanem and J. Red-Horse (1999), 'Propagation of probabilistic uncertainty in complex physical systems using a stochastic finite element approach', *Physica* D **133**, 137–144.

R. Ghanem and P. D. Spanos (2003), *Stochastic Finite Elements: A Spectral Approach*, revised edition, Dover.

R. G. Ghanem and R. M. Kruger (1996), 'Numerical solution of spectral stochastic finite element systems', *Comput. Methods Appl. Mech. Engrg* **129**, 289–303.

R. G. Ghanem and P. D. Spanos (1991), *Stochastic Finite Elements: A Spectral Approach*, Springer.

M. B. Giles (2008), 'Multilevel Monte Carlo path simulation', *Operations Research* **56**, 607–617.

J. Glimm, S. Hou, Y.-H. Lee, D. H. Sharp and K. Ye (2003), Solution error models for uncertainty quantification. In *Advances in Differential Equations and Mathematical Physics: Birmingham, AL, 2002*, Vol. 327 of *Contemporary Mathematics*, AMS, pp. 115–140.

A. Gordon and C. Powell (2012), 'On solving stochastic collocation systems with algebraic multigrid', *IMA J. Numer. Anal.* **32**, 1051–1070.

M. Griebel (1998), 'Adaptive sparse grid multilevel methods for elliptic PDEs based on finite differences', *Computing* **61**, 151–179.

M. Grigoriu (2002), *Stochastic Calculus: Applications in Science and Engineering*, Birkhäuser.

P. Grisvard (1985), *Elliptic Problems in Non-Smooth Domains*, Pitman.

M. Gunzburger and A. Labovsky (2011), 'Effects of approximate deconvolution models on the solution of the stochastic Navier–Stokes equations', *J. Comput. Math.* **29**, 131–140.

M. Gunzburger, P. Jantsch, A. Teckentrup and C. G. Webster (2014), 'A multilevel stochastic collocation method for partial differential equations with random input data', *SIAM J. Uncertainty Quantification*, submitted.

M. Gunzburger, C. Trenchea and C. G. Webster (2013), 'A generalized stochastic collocation approach to constrained optimization for random data identification problems', *Numerical Methods for PDEs*, submitted.

M. Gunzburger, C. G. Webster and G. Zhang (2014), An adaptive wavelet stochastic collocation method for irregular solutions of stochastic partial differential equations with random input data. In *Sparse Grids and Applications: Munich 2012*, Vol. 97 of *Lecture Notes in Computational Science and Engineering*, Springer, pp. 137–170.

J. Hammersley and D. Handscomb (1964), *Monte Carlo Methods*, Halsted.

R. Hardin and N. Sloane (1993), 'A new approach to the construction of optimal designs', *J. Statist. Planning Inference* **37**, 339–369.

J. C. Helton (1997), 'Analysis in the presence of stochastic and subjective uncertainties', *J. Statist. Comput. Simulation* **57**, 3–76.

J. C. Helton and F. J. Davis (2003), 'Latin hypercube sampling and the propagation of uncertainty in analyses of complex systems', *Reliab. Engrg System Safety* **81**, 23–69.

D. Higdon, M. Kennedy, J. C. Cavendish, J. A. Cafeo and R. D. Ryne (2004), 'Combining field data and computer simulations for calibration and prediction', *SIAM J. Sci. Comput.* **26**, 448–466.

J. Hlaváček, I. Chleboun and I. M. Babuška (2004), *Uncertain Input Data Problems and the Worst Scenario Method*, Elsevier.

S. Hosder and R. W. Walters (2007), A non-intrusive polynomial chaos method for uncertainty propagation in CFD simulations. In *44th AIAA Aerospace Sciences Meeting*.

D. Jacobsen, M. Gunzburger, T. Ringler, J. Burkardt and J. Peterson (2013), 'Parallel algorithms for planar and spherical Delaunay construction with an application to centroidal Voronoi tessellations', *Geo. Mod. Develop.* **6**, 1427–1466.

J. D. Jakeman, R. Archibald and D. Xiu (2011), 'Characterization of discontinuities in high-dimensional stochastic problems on adaptive sparse grids', *J. Comput. Phys.* **230**, 3977–3997.

P. Jantsch, C. Webster and G. Zhang (2014), A hierarchical stochastic collocation method for adaptive acceleration of PDEs with random input data. ORNL Technical Report.

C. Jin, X. Cai and C. Li (2007), 'Parallel domain decomposition methods for stochastic elliptic equations', *SIAM J. Sci. Comput.* **29**, 2096–2114.

C. Johnson (2000), Adaptive computational methods for differential equations. In *ICIAM 99: Edinburgh*, Oxford University Press, pp. 96–104.

V. R. Joseph (2013), 'A note on nonnegative DoIt approximation', *Technometrics* **55**, 103–107.

V. R. Joseph and S. N. Melkote (2009), 'Statistical adjustments to engineering models', *J. Quality Technology* **41**, 362–375.

E. Jouini, J. Cvitanić and M. Musiela, eds (2001), *Option Pricing, Interest Rates and Risk Management*, Cambridge University Press.

L. Ju, M. Gunzburger and W. Zhao (2006), 'Adaptive finite element methods for elliptic PDEs based on conforming centroidal Voronoi–Delaunay triangulations', *SIAM J. Sci. Comput.* **28**, 2023–2053.

H. Kahn and A. Marshall (1953), 'Methods of reducing sample size in Monte Carlo computations', *J. Oper. Res. Soc. Amer.* **1**, 263–271.

G. Karniadakis, C.-H. Su, D. Xiu, D. Lucor, C. Schwab and R. Todor (2005), Generalized polynomial chaos solution for differential equations with random inputs. SAM Report 2005-01, ETH Zürich.

A. Keese and H. G. Matthies (2005), 'Hierarchical parallelisation for the solution of stochastic finite element equations', *Comput. Struct.* **83**, 1033–1047.

M. C. Kennedy and A. O'Hagan (2001), 'Bayesian calibration of computer models' (with discussion), *J. Royal Statist. Soc.* B **63**, 425–464.

C. Ketelsen, R. Scheichl and A. L. Teckentrup (2013), A hierarchical multilevel Markov chain Monte Carlo algorithm with applications to uncertainty quantification in subsurface flow. arXiv:1303.7343

M. Kleiber and T.-D. Hien (1992), *The Stochastic Finite Element Method*, Wiley.

A. Klimke and B. Wohlmuth (2005), 'Algorithm 847: Spinterp: Piecewise multilinear hierarchical sparse grid interpolation in MATLAB', *ACM Trans. Math. Software* **31**, 561–579.

I. Kramosil (2001), *Probabilistic Analysis of Belief Functions*, Kluwer.

O. P. Le Maître and O. M. Knio (2010), *Spectral Methods for Uncertainty Quantification: With Applications to Computational Fluid Dynamics*, Springer.

O. P. Le Maître, O. M. Knio, H. N. Najm and R. G. Ghanem (2004*a*), 'Uncertainty propagation using Wiener–Haar expansions', *J. Comput. Phys.* **197**, 28–57.

O. P. Le Maître, H. N. Najm, R. G. Ghanem and O. M. Knio (2004*b*), 'Multiresolution analysis of Wiener-type uncertainty propagation schemes', *J. Comput. Phys.* **197**, 502–531.

J. C. Lemm (2003), *Bayesian Field Theory*, Johns Hopkins University Press.

C. F. Li, Y. T. Feng, D. R. J. Owen, D. F. Li and I. M. Davis (2007), 'A Fourier–Karhunen–Loève discretization scheme for stationary random material properties in SFEM', *Internat. J. Numer. Methods Engrg.* **73**, 1942–1965.

G. Lin, A. M. Tartakovsky and D. M. Tartakovsky (2010), 'Uncertainty quantification via random domain decomposition and probabilistic collocation on sparse grids', *J. Comput. Phys.* **229**, 6995–7012.

M. Loève (1977), *Probability Theory I*, fourth edition, Vol. 45 of *Graduate Texts in Mathematics*, Springer.

M. Loève (1978), *Probability Theory II*, fourth edition, Vol. 46 of *Graduate Texts in Mathematics*, Springer.

Z. Lu and D. Zhang (2004), 'A comparative study on uncertainty quantification for flow in randomly heterogeneous media using Monte Carlo simulations and conventional and KL-based moment-equation approaches', *SIAM J. Sci. Comput.* **26**, 558–577.

D. Lucor and G. E. Karniadakis (2004), 'Predictability and uncertainty in flow-structure interactions', *Eur. J. Mech. B Fluids* **23**, 41–49.

D. Lucor, J. Meyers and P. Sagaut (2007), 'Sensitivity analysis of large-eddy simulations to subgrid-scale-model parametric uncertainty using polynomial chaos', *J. Fluid Mech.* **585**, 255–279.

D. Lucor, D. Xiu, C.-H. Su and G. E. Karniadakis (2003), 'Predictability and uncertainty in CFD', *Internat. J. Numer. Methods Fluids* **43**, 483–505.

X. Ma and N. Zabaras (2009), 'An adaptive hierarchical sparse grid collocation algorithm for the solution of stochastic differential equations', *J. Comput. Phys.* **228**, 3084–3113.

X. Ma and N. Zabaras (2010), 'An adaptive high-dimensional stochastic model representation technique for the solution of stochastic partial differential equations', *J. Comput. Phys.* **229**, 3884–3915.

Y. Marzouk and D. Xiu (2009), 'A stochastic collocation approach to Bayesian inference in inverse problems', *Commun. Comput. Phys.* **6**, 826–847.

Y. M. Marzouk, H. N. Najm and L. A. Rahn (2007), 'Stochastic spectral methods for efficient Bayesian solution of inverse problems', *J. Comput. Phys.* **224**, 560–586.

L. Mathelin and K. Gallivan (2010), 'A compressed sensing approach for partial differential equations with random input data', *Comput. Methods Appl. Mech. Engrg*, submitted.

L. Mathelin, M. Y. Hussaini and T. A. Zang (2005), 'Stochastic approaches to uncertainty quantification in CFD simulations', *Numer. Algorithms* **38**, 209–236.

H. G. Matthies and A. Keese (2005), 'Galerkin methods for linear and nonlinear elliptic stochastic partial differential equations', *Comput. Methods Appl. Mech. Engrg* **194**, 1295–1331.

R. E. Melchers (1999), *Structural Reliability, Analysis and Prediction*, Wiley.

G. Migliorati, F. Nobile, E. Von Schwerin and R. Tempone (2013), 'Approximation of quantities of interest in stochastic PDEs by the random discrete L^2 projection on polynomial spaces', *SIAM J. Sci. Comput.* **35**, A1440–A1460.

K.-S. Moon, E. von Schwerin, A. Szepessy and R. Tempone (2006), An adaptive algorithm for ordinary, stochastic and partial differential equations. In *Recent Advances in Adaptive Computation*, Vol. 381 of *Contemporary Mathematics*, AMS, pp. 369–388.

J. Mrczyk, ed. (1997), *Computational Mechanics in a Meta Computing Perspective*, Center for Numerical Methods in Engineering, Barcelona.

M. Muto and J. L. Beck (2008), 'Bayesian updating and model class selection for hysteretic structural models using stochastic simulation', *J. Vibration Control* **14**, 7–34.

V. A. B. Narayanan and N. Zabaras (2004), 'Stochastic inverse heat conduction using a spectral approach', *Internat. J. Numer. Methods Engrg* **60**, 1569–1593.

V. A. B. Narayanan and N. Zabaras (2005*a*), 'Variational multiscale stabilized FEM formulations for transport equations: Stochastic advection–diffusion and incompressible stochastic Navier–Stokes equations', *J. Comput. Phys.* **202**, 94–133.

V. A. B. Narayanan and N. Zabaras (2005*b*), 'Using stochastic analysis to capture unstable equilibrium in natural convection', *J. Comput. Phys.* **208**, 134–153.

H. Nguyen, J. Burkardt, M. Gunzburger, L. Ju and Y. Saka (2009), 'Constrained CVT meshes and a comparison of triangular mesh generators', *Comp. Geom. Theo. Appl.* **42**, 1–19.

H. Niederreiter (1992), *Random Number Generation and Quasi-Monte Carlo Methods*, Vol. 63 of *CBMS–NSF Regional Conference Series in Applied Mathematics*, SIAM.

F. Nobile and R. Tempone (2009), 'Analysis and implementation issues for the numerical approximation of parabolic equations with random coefficients', *Internat. J. Numer. Methods Engrg* **80**, 979–1006.

F. Nobile, R. Tempone and C. G. Webster (2007), The analysis of a sparse grid stochastic collocation method for partial differential equations with high-dimensional random input data. Technical Report SAND2007-8093, Sandia National Laboratories.

F. Nobile, R. Tempone and C. G. Webster (2008*a*), 'A sparse grid stochastic collocation method for partial differential equations with random input data', *SIAM J. Numer. Anal.* **46**, 2309–2345.

F. Nobile, R. Tempone and C. G. Webster (2008*b*), 'An anisotropic sparse grid stochastic collocation method for partial differential equations with random input data', *SIAM J. Numer. Anal.* **46**, 2411–2442.

E. Novak (1988), 'Stochastic properties of quadrature formulas', *Numer. Math.* **53**, 609–620.

W. L. Oberkampf, J. C. Helton and K. Sentz (2001), Mathematical representation of uncertainty. AIAA paper 2001-1645.

J. T. Oden and S. Prudhomme (2002), 'Estimation of modeling error in computational mechanics', *J. Comput. Phys.* **182**, 496–515.

J. T. Oden and K. S. Vemaganti (2000), 'Estimation of local modeling error and goal-oriented adaptive modeling of heterogeneous materials I: Error estimates and adaptive algorithms', *J. Comput. Phys.* **164**, 22–47.

J. T. Oden, I. M. Babuška, F. Nobile, Y. Feng and R. Tempone (2005*a*), 'Theory and methodology for estimation and control of errors due to modeling, approximation, and uncertainty', *Comput. Methods Appl. Mech. Engrg* **194**, 195–204.

J. T. Oden, T. Belytschko, I. Babuška and T. J. R. Hughes (2003), 'Research directions in computational mechanics', *Comput. Methods Appl. Mech. Engrg* **192**, 913–922.

J. T. Oden, S. Prudhomme and P. Bauman (2005*b*), 'On the extension of goal-oriented error estimation and hierarchical modeling to discrete lattice models', *Comput. Methods Appl. Mech. Engrg* **194**, 3668–3688.

J. T. Oden, S. Prudhomme, D. C. Hammerand and M. S. Kuczma (2001), 'Modeling error and adaptivity in nonlinear continuum mechanics', *Comput. Methods Appl. Mech. Engrg* **190**, 6663–6684.

M. Parks, E. De Sturler, G. Mackey, D. Johnson and S. Maiti (2006), 'Recycling Krylov subspaces for sequences of linear systems', *SIAM J. Sci. Comput.* **28**, 1651–1674.

M. F. Pellissetti and R. G. Ghanem (2000), 'Iterative solution of systems of linear equations arising in the context of stochastic finite elements', *Adv. Engineering Software* **31**, 607–616.

E. Phipps, M. Eldred, A. Salinger and C. Webster (2008), Capabilities for uncertainty in predictive science. Technical Report SAND2008-6527, Sandia National Laboratories.

S. Pope (1981), 'Transport equation for the joint probability density function of velocity and scalars in turbulent flow', *Phys. Fluids* **24**, 588–596.

S. Pope (1982), 'The application of PDF transport equations to turbulent reactive flows', *J. Non-Equil. Thermody.* **7**, 1–14.

C. E. Powell and H. C. Elman (2009), 'Block-diagonal preconditioning for spectral stochastic finite-element systems', *IMA J. Numer. Anal.* **29**, 350–375.

C. E. Powell and E. Ullmann (2010), 'Preconditioning stochastic Galerkin saddle point systems', *SIAM J. Matrix Anal. Appl.* **31**, 2813–2840.

W. Press, S. Teukolsky, W. Vetterling and B. Flannery (2007), *Numerical Recipes: The Art of Scientific Computing*, Cambridge University Press.

F. Pukelsheim (1993), *Optimal Design of Experiments*, SIAM.

Z. Qian and C. F. J. Wu (2008), 'Bayesian hierarchical modeling for integrating low-accuracy and high-accuracy experiments', *Technometrics* **50**, 192–204.

Z. Qian, C. Seepersad, R. Joseph, J. Allen and C. F. J. Wu (2006), 'Building surrogate models based on detailed and approximate simulations', *ASME J. Mech. Design* **128**, 668–677.

M. M. Rao and R. J. Swift (2006), *Probability Theory with Applications*, Vol. 582 of *Mathematics and its Applications*, second edition, Springer.

M. T. Reagana, H. N. Najm, R. G. Ghanem and O. M. Knio (2003), 'Uncertainty quantification in reacting-flow simulations through non-intrusive spectral projection', *Combustion and Flame* **132**, 545–555.

H. M. Regan, S. Ferson and D. Berleant (2004), 'Equivalence of methods for uncertainty propagation of real-valued random variables', *Internat. J. Approx. Reason.* **36**, 1–30.

J. Reilly, P. H. Stone, C. E. Forest, M. D. Webster, H. D. Jacoby and R. G. Prinn (2001), 'Uncertainty and climate change assessments', *Science* **293**, 430–433.

T. Ringler, L. Ju and M. Gunzburger (2008), 'A multi-resolution method for climate system modeling: Application of spherical centroidal Voronoi tessellations', *Ocean Dyn.* **58**, 475–498.

B. Ripley (1987), *Stochastic Simulation*, Wiley.

L. Roman and M. Sarkis (2006), 'Stochastic Galerkin method for elliptic SPDEs: A white noise approach', *Discrete Contin. Dyn. Syst.* B **6**, 941–955.

V. Romero, J. Burkardt, M. Gunzburger and J. Peterson (2005), Initial evaluation of pure and Latinized centroidal Voronoi tessellation for non-uniform statistical sampling. In *Sensitivity Analysis of Model Output*, Los Alamos National Laboratory, pp. 380–401.

V. Romero, J. Burkardt, M. Gunzburger, J. Peterson and K. Krishnamurthy (2003*a*), Initial application and evaluation of a promising new sampling method for response surface generation: Centroidal Voronoi tessellations. In *Proc. 44th AIAA/AME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, pp. 1488–1506. AIAA paper 2003-2008.

V. Romero, M. Gunzburger, J. Burkardt and J. Peterson (2003*b*), Initial evaluation of centroidal Voronoi tessellation method for statistical sampling and function integration. In *Fourth International Symposium on Uncertainty Modeling and Analysis*, ISUMA, pp. 174–183.

V. Romero, M. Gunzburger, J. Burkardt and J. Peterson (2006), 'Comparison of pure and "Latinized" centroidal Voronoi tessellation against other statistical sampling methods', *Reliab. Engrg System Safety* **91**, 1266–1280.

A. Romkes and J. T. Oden (2004), 'Adaptive modeling of wave propagation in heterogeneous elastic solids', *Comput. Methods Appl. Mech. Engrg* **193**, 539–559.

R. Rubinstein (1981), *Simulation and the Monte Carlo Method*, Wiley.

R. Rubinstein and M. Choudhari (2005), 'Uncertainty quantification for systems with random initial conditions using Wiener–Hermite expansions', *Stud. Appl. Math.* **114**, 167–188.

W. Rudin (1987), *Real and Complex Analysis*, third edition, McGraw-Hill.

Y. Saka, M. Gunzburger and J. Burkardt (2007), 'Latinized, improved LHS, and CVT point sets in hypercubes', *Internat. J. Numer. Anal. Model.* **4**, 729–743.

T. Sauer and Y. Xu (1995), 'On multivariate Lagrange interpolation', *Math. Comp.* **64**, 1147–1170.

C. Schwab and R.-A. Todor (2003*a*), 'Sparse finite elements for elliptic problems with stochastic loading', *Numer. Math.* **95**, 707–734.

C. Schwab and R. A. Todor (2003*b*), 'Sparse finite elements for stochastic elliptic problems: Higher order moments', *Computing* **71**, 43–63.

V. Simoncini and D. B. Szyld (2007), 'Recent computational developments in krylov subspace methods for linear systems', *Numer. Linear Algebra Appl.* **14**, 1–59.

P. Smith, M. Shafi and H. Gao (1997), 'Quick simulation: A review of importance sampling techniques in communication systems', *IEEE J. Select. Areas Commun.* **15**, 597–613.

S. Smolyak (1963), 'Quadrature and interpolation formulas for tensor products of certain classes of functions', *Dokl. Akad. Nauk SSSR* **4**, 240–243 (English translation).

C. Soize (2003), 'Random matrix theory and non-parametric model of random uncertainties in vibration analysis', *J. Sound Vibration* **263**, 893–916.

C. Soize (2005), 'Random matrix theory for modeling uncertainties in computational mechanics', *Comput. Methods Appl. Mech. Engrg* **194**, 1333–1366.

C. Soize and R. Ghanem (2004), 'Physical systems with random uncertainties: Chaos representations with arbitrary probability measure', *SIAM J. Sci. Comput.* **26**, 395–410.

R. Srinivasan (2002), *Importance sampling: Applications in Communications and Detection*, Springer.

M. Stoyanov and C. G. Webster (2014), 'A gradient-based sampling approach for stochastic dimension reduction for partial differential equations with random input data', *Internat. J. Uncertainty Quantification*, to appear.

W. Sweldens (1996), 'The lifting scheme: A custom-design construction of biorthogonal wavelets', *Appl. Comput. Harmon. Anal.* **3**, 186–200.

W. Sweldens (1998), 'The lifting scheme: A construction of second generation wavelets', *SIAM J. Math. Anal.* **29**, 511–546.

D. M. Tartakovsky and S. Broyda (2011), 'PDF equations for advective–reactive transport in heterogeneous porous media with uncertain properties', *J. Contaminant Hydrology* **120/121**, 129–140.

M. Tatang (1995), Direct incorporation of uncertainty in chemical and environmental engineering systems. PhD thesis, MIT.
J. C. Taylor (1997), *An Introduction to Measure and Probability*, Springer.
R. A. Todor (2005), Sparse perturbation algorithms for elliptic PDE's with stochastic data. Dissertation 16192, ETH Zürich.
H. Tran, C. Trenchea and C. G. Webster (2012), Convergence analysis of global stochastic collocation methods for Navier–Stokes with random input data. Technical Report ORNL/TM-2014/000, Oak Ridge National Laboratory. Submitted to *SIAM J. Uncertainty Quantification.*
J. F. Traub and A. G. Werschulz (1998), *Complexity and Information*, Cambridge University Press.
L. N. Trefethen (2008), 'Is Gauss quadrature better than Clenshaw–Curtis?', *SIAM Review* **50**, 67–87.
R. Tuo and C. F. J. Wu (2013), A theoretical framework for calibration in computer models: Parametrization, estimation and convergence properties. Technical report, Georgia Tech.
E. Ullmann (2010), 'A Kronecker product preconditioner for stochastic Galerkin finite element discretizations', *SIAM J. Sci. Comput.* **32**, 923–946.
E. Ullmann, H. C. Elman and O. G. Ernst (2012), 'Efficient iterative solvers for stochastic Galerkin discretizations of log-transformed random diffusion problems', *SIAM J. Sci. Comput.* **34**, A659–A682.
R. Verfürth (1996), *A Review of A Posteriori Error Estimation and Adaptive Mesh Refinement Techniques*, Wiley-Teubner.
S. G. Vick (2002), *Degrees of Belief: Subjective Probability and Engineering Judgment*, American Society of Civil Engineers.
X. Wan and G. E. Karniadakis (2009), 'Solving elliptic problems with non-Gaussian spatially-dependent random coefficients', *Comput. Methods Appl. Mech. Engrg* **198**, 1985–1995.
J. Wang and N. Zabaras (2005), 'Hierarchical Bayesian models for inverse problems in heat conduction', *Inverse Problems* **21**, 183–206.
G. W. Wasilkowski and H. Woźniakowski (1995), 'Explicit cost bounds of algorithms for multivariate tensor product problems', *J. Complexity* **11**, 1–56.
C. G. Webster (2007), Sparse grid stochastic collocation techniques for the numerical solution of partial differential equations with random input data. PhD thesis, Florida State University.
C. G. Webster, G. Zhang and M. Gunzburger (2013), 'An adaptive sparse-grid iterative ensemble Kalman filter approach for parameter field estimation', *Internat. J. Comput. Math.*, to appear.
N. Wiener (1938), 'The homogeneous chaos', *Amer. J. Math.* **60**, 897–936.
C. L. Winter and D. M. Tartakovsky (2002), 'Groundwater flow in heterogeneous composite aquifers', *Water Resour. Res.* **38**, 23.
C. L. Winter, D. M. Tartakovsky and A. Guadagnini (2002), 'Numerical solutions of moment equations for flow in heterogeneous composite aquifers', *Water Resour. Res.* **38**, 13.
G. Womeldorff, J. Peterson, M. Gunzburger and T. Ringler (2013), 'Unified matching grids for multidomain multiphysics simulations', *SIAM J. Sci. Comput.* **35**, A2781–A2806.

D. Xiu (2009), 'Fast numerical methods for stochastic computations: A review', *Commun. Comput. Phys.* **5**, 242–272.

D. Xiu (2010), *Numerical Methods for Stochastic Computations: A Spectral Method Approach*, Princeton University Press.

D. Xiu and J. Hesthaven (2005), 'High-order collocation methods for differential equations with random inputs', *SIAM J. Sci. Comput.* **27**, 1118–1139.

D. Xiu and G. E. Karniadakis (2002*a*), 'Modeling uncertainty in steady state diffusion problems via generalized polynomial chaos', *Comput. Methods Appl. Mech. Engrg* **191**, 4927–4948.

D. Xiu and G. E. Karniadakis (2002*b*), 'The Wiener–Askey polynomial chaos for stochastic differential equations', *SIAM J. Sci. Comput.* **24**, 619–644.

D. Xiu and G. E. Karniadakis (2003), 'Modeling uncertainty in flow simulations via generalized polynomial chaos', *J. Comput. Phys.* **187**, 137–167.

D. Xiu and D. M. Tartakovsky (2004), 'A two-scale nonperturbative approach to uncertainty analysis of diffusion in random composites', *Multiscale Model. Simul.* **2**, 662–674.

K. V. Yuen and J. L. Beck (2003), 'Updating properties of nonlinear dynamical systems with uncertain input', *J. Engrg Mech.* **129**, 9–20.

N. Zabaras and D. Samanta (2004), 'A stabilized volume-averaging finite element method for flow in porous media and binary alloy solidification processes', *Internat. J. Numer. Methods Engrg* **60**, 1103–1138.

G. Zhang and M. Gunzburger (2012), 'Error analysis of a stochastic collocation method for parabolic partial differential equations with random input data', *SIAM J. Numer. Anal.* **50**, 1922–1940.

G. Zhang, D. Lu, M. Ye, M. Gunzburger and C. Webster (2013), 'An adaptive sparse-grid high-order stochastic collocation method for Bayesian inference in groundwater reactive transport modeling', *Water Resour. Res.* **49**, 6871–6892.

G. Zhang, C. Webster, M. Gunzburger and J. Burkardt (2014), A hyper-spherical adaptive sparse-grid method for high-dimensional discontinuity detection. ORNL Technical Report.