Variants of Polynomial Chaos Methods in Uncertainty Quantification

A Thesis

submitted in partial fulfillment of the requirements for the award of the degree of

Doctor of Philosophy

in

School of Mathematics

by

Navjot Kaur
Reg no: 901711005

under the supervison of

Dr. Kavita
Assistant Professor
School of Mathematics

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

THAPAR INSTITUTE OF ENGINEERING AND TECHNOLOGY
PATIALA-147004, PUNJAB, INDIA
October 7, 2021

Certificate

I, hereby certify that the work, which is being presented in the thesis, entitled **"Variants of Polynomial Chaos Methods in Uncertainty Quantification"**, in partial fulfillment of the requirements for the award of the degree of **Doctor of Philosophy** and submitted to the institution is an authentic record of my own work carried out during the period **27 July, 2017** to **October 7, 2021** under the supervision of **Dr. Kavita**, Assistant Professor, School of Mathematics , Thapar Institute of Engineering and Technology, Patiala, Punjab, India.

Date: 13|10|2021

Navjot Kaur .

(**Navjot Kaur**)
Candidate

It is certified that the above statement made by the candidate is correct to the best of my knowledge.

Date: 13.10.2021

Kavita

Dr. Kavita
Supervisor

.....dedicated to my parents

Acknowledgements

This is unbelievable that I have finally summed up all my hardwork into a Ph.D. thesis. This is indeed priceless. It is strenuous hard work of four years. These years grew me into a better person both scientifically and personally with increased confidence, patience, skills and knowledge. In the beginning, the journey seemed very challenging and tough but as time passed, everything seemed to get in place. This journey would have been impossible without the support of few people which I would like to thank here. Indeed, it is a humbling experience to acknowledge those people who have, mostly out of kindness, helped along the journey of my Ph.D. I am indebted to so many for their encouragement and support.

First and foremost, I would like to thank the Almighty, who has always showered countless blessings, knowledge, and opportunities upon me. A sincere thanks to my supervisor *Dr. Kavita* for always believing in me and giving me accessible guidance and support throughout my Ph.D. journey. While continuously providing constructive suggestions, her guidance has always left me with much space for exploring new directions. Her immense knowledge and plentiful experience has always encouraged me in all the time of my academic research and personal life. There have been numerous times when I remember I felt disheartened and stumped about the direction of my research but, a meeting with her would bring back all the enthusiasm and raise my spirits up. I feel incredibly privileged to have her as my supervisor.

I am thankful to Thapar Institute of Engineering and Technology, Patiala authorities for providing me the necessary facilities for the smooth completion of my Ph.D. I would like to give special thanks to my dissertation committee members: *Dr. Harish Garg, Dr. Sapna Sharma* and *Dr. R.K. Sharma* for their insightful comments and encouragement. Their suggestions brought in threads of thought that made my research so much richer, and my dissertation something I can be proud of having written. I express my gratitude to all the faculty members and staff of the School of Mathematics, Thapar Institute of Engineering and Technology, Patiala, for their kind support.

My deepest thanks to my friends and lab mates- *Deepika Sharma, Sukhbir, Gimmi, Kiran, Simar, Kulbir, Manpreet* and *Nikita* who have always been a major source of support and for the cherished time spent together in the lab. Thanks guys for always being there for me. A special thanks to my Ph.D. buddy *Gagan*, a source of great emotional support, who always managed to make me feel special and with whom I had the best lunch breaks of

my life.

I owe so much thanks to my parents *Jagdev Singh* and *Bhupinder Kaur* for always encouraging me to be an independent thinker, and having confidence in my abilities to go after new things that inspired me. Their constant support and confidence in me enhanced my ability to get through the ups and downs of my bumpy academic career and succeed in the end. I am grateful to my sister *Kiran* for her support, understanding, and love when it was most required and my little nephew *Sidak* for always cheering me up with his cute little smile whenever I felt low. A special thanks to my brother *Balkaran* and my brother-in-law *Parvinder Singh* for always motivating me to work hard. I owe my deepest gratitude to my chacha ji *Gunjit Singh Tiwana*, chachi ji *Inderpal Kaur* and bhui ji *Manjit Kaur* for always encouraging me to achieve my dreams. A sincere thanks to my parents-in-law *Jasveer Kaur* and *Lal Singh* for understanding, loving and supporting me to finish this piece of work.

Finally, I would like to thank *Sukhveer*, my best friend, a great companion who always supported, encouraged, entertained and helped me in keeping things going and for always showing how proud he is of me.

Navjot Kaur

Abstract

The area of uncertainty quantification (UQ) has acquired a lot of importance in the past few years. The eagerness to achieve precision has driven today's world to quantify the uncertainties present in various physical and engineering problems. In order to have a better understanding of the stochastic approaches, a current state-of-the-art review of the numerical methods for stochastic computations is presented. In this thesis, a brief account of the related work of various authors to numerically solve stochastic partial differential equations (PDEs) by using several approaches is reviewed. The framework of the methods is discussed along with their algorithms, literature, comparison analysis, strengths, and weaknesses. An illustrative example of a simple ordinary differential equations (ODEs) with uncertain parameter is discussed and is compared with three main methods-Monte Carlo, polynomial chaos and stochastic collocation method. We initially started with the traditional polynomial chaos method which involves Hermite polynomials and united it with the summation by parts-simultaneous approximation terms (SBP-SAT) operators in order to acquire the stability conditions for the Dirichlet boundary conditions (BCs). Spatial derivatives are approximated by SBP operators and SATs are used to enforce BCs by ensuring stable solutions.

As our aim was to develop variants of polynomial chaos methods in engineering problems, so, a new class of wavelets known as B-spline wavelets is introduced into the area of UQ. On the basis of order of B-spline wavelets, linear and cubic Wiener semi-orthogonal B-spline generalized polynomial chaos (gPC) is developed. The advantages of B-spline wavelet based gPC are

- Usually, gPC may have slow convergence or fails to converge in problems which consists of discontinuities or sharp dependence on the random space even in short-time integration. However, wavelets being local waves are known for expressing discontinuities or steep gradients more accurately than the global basis.

- Wavelet methods are generally known for their self adaptive nature which makes it a good choice for the numerical solution of a stochastic PDEs. The self adaptivity property comes from the good localization properties of wavelets which are seen both in space and frequency. This makes B-spline wavelets an interesting tool for adaptivity to explore it in stochastic PDEs.

- Although there is wide research available for wavelets, yet the theory of wavelets for numerically solving stochastic PDEs in UQ is in its emerging phase. The beauty of

semi-orthogonal compactly supported B-spline wavelets is that they have finite support, both even and odd symmetry and simple analytical expressions, ideal attributes of a basis function.

Adaptive schemes perform refinements where most needed in order to reduce the computational effort. Such local refinements plays a vital role when the system dynamics indicate steep dependences on the random parameters. One such method known as wavelet optimized finite difference method is employed in collaboration with B-spline wavelet based gPC to find the adaptive solutions of stochastic PDEs with periodic BCs. Moreover, we introduce another method known as adaptive wavelet collocation gPC which involves the concept of autocorrelation functions of compactly supported Daubechies scaling functions. In particular, this collocation method seems well suited to the treatment of Dirichlet's BCs.

We have implemented the above developed methods on number of differential equations from engineering problems. In this thesis, we have illustrated simple second order ODEs with discontinuous solution, coupled ODEs such as stochastic linear oscillator problem, stochastic Kraichan-Orszag problem, linear as well as non-linear PDEs. Moreover, we have practiced our method on real life epidemic situation consisting of influenza in boys boarding school in England, 1978 and Ebola in Liberia, 2014. Real data is considered from British Medical Journal and World Health Organization. Moreover, in this, we have compared the linear and cubic B-spline wavelets and it has been observed that linear B-spline wavelets show faster results as compared to cubic B-spline wavelets.

The wavelet optimized finite difference B-spline wavelet gPC is tested on stochastic heat equation and stochastic Burger's equation. Further, the adaptive wavelet collocation based gPC is tested on elliptic problem, advection equation and non-linear Burger's equation. The CPU time analysis of the methods reveals that the methods are quite efficient. The postprocessing step which includes first order moment (mean) and second order moment (variance) is performed for each test problem.

List of Publications

International Journal

1. N. Kaur and K. Goyal, *Uncertainty propagation using Wiener-Linear B-spline wavelet expansion*, Computers & Mathematics with Applications, 79(9), 2598-2623, 2020, doi: https://doi.org/10.1016/j.camwa.2019.11.021
   (**SCI: Impact Factor: 3.476**).

2. N. Kaur and K. Goyal, *Hybrid Hermite polynomial chaos SBP-SAT technique for stochastic advection-diffusion equations*, International Journal of Modern Physics C, 31(9), 2020, pp. 2050128 (33 pages), doi: https://doi.org/10.1142/S0129183120501284
   (**SCI: Impact Factor: 1.17**).

3. N. Kaur and K. Goyal, *Uncertainty Quantification of Stochastic Epidemic SIR Models Using B-spline Polynomial Chaos*, Regular and Chaotic Dynamics, 26(1), 22-38, 2021, doi: https://doi.org/10.1134/S1560354721010020
   (**SCI: Impact Factor: 1.421**).

Submitted after Revision

1. N. Kaur and K. Goyal, *An Adaptive Wavelet Optimized Finite Difference B-spline Polynomial Chaos Method for Random Partial Differential Equations*, Applied Mathematics and Computation **(SCI: Impact factor: 4.091)**.

Communicated Papers

1. N. Kaur and K. Goyal, *"Stochastic Approaches to Uncertainty Quantification: A review", Applications of Mathematics.*

2. N. Kaur and K. Goyal, *"An Adaptive Stochastic Investigation of Partial Differential Equations Using Wavelet Collocation Generalized Polynomial Chaos Method", SIAM Journal on Numerical Analysis.*

International Conference

1. N. Kaur and K. Goyal, *Uncertainty propagation using Wiener-Linear B-spline wavelet expansion* presented in International Conference on Differential Equations and Control Problems: Modeling, Analysis and Computations (ICDECP) held at IIT Mandi, India, June 17-19, 2019.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Uncertainties can be seen almost everywhere ranging from a layman using weighing machine to a scientist precisely measuring weight of nano-particles. Therefore, mathematical models which are used to represent these real-life engineering systems are subject to a large range of uncertainties, thus, leading to the increased interest of researchers and scientists in the field of Uncertainty Quantification (UQ). Over the past few decades, due to the growth of statistical mechanics, computational power, simulations and testing, there has been a progressive trend in science towards making robust and powerful models even in the presence of uncertainty or randomness in problems. Every design such as an airplane or a new building shows a model-based estimation that the airplane will fly safely or the building will stand. Hence, besides solely validating a model, prediction in engineering design presumes that resources of imagination, money, and time can be confidently spent in order to produce something strong, reliable and economical. Mathematical models together with computer models are typically made to take decisions, calculate risks and frame decisions accordingly. One challenging thing what researchers face is that higher the complexity of the system becomes, harder it takes the computer model to present it accurately. Therefore, we focus on UQ of complex systems which consists of tough mathematical framework such as systems of differential equations (DEs) where computation of even a single deterministic inversion of a model is expensive.

UQ is the science of identification, quantitative characterization and minimization of uncertainties in both real world and computational applications. It tries to find out how likely certain outcomes are if some details of the system are not completely known. UQ basically is the amalgamation of three fields - mathematics, statistics, and engineering. On a whole, it solicits to handle the problems consisting of real world variability and probabilistic behavior into system analysis and engineering. The quantification of uncertainties is the ever-growing demand of the scientific field, for eg. prediction of uncertainty in production of oil from oil reservoirs, reliable prediction in weather forecasting, the influence of random heat conductivity and capacity on the solution of transient heat conduction and so on. UQ basically involves three steps in quantifying uncertainty in engineering problems [1, 2].

These are as follows:

1. **Characterization of uncertainties** - The first step aims at using the power of mathematical statistics for characterizing the uncertain features linked to the given system as one random variable (RV), more than one RV, random matrices, random fields, or random operators.

2. **Propagation of uncertainties** - The second step focuses on propagating characterization of inputs into characterization of outputs. There are number of ways to carry out this, for instance, by employing Monte Carlo (MC) sampling, stochastic polynomial chaos expansion (PCE), stochastic collocation method (SCM) and so on.

3. **Quantification of uncertainties** - Lastly, in this step, the aim is to make the model beneficial in the design and analysis part of engineering system under study. It can be achieved by performing sensitivity analysis (SA) [3] for the reduction of uncertainties, by utilizing decision-theoretic methods for validation, or by using optimization methods for improving the design such as variance-based SA, differentiation-based SA, Sobol, ANOVA etc.

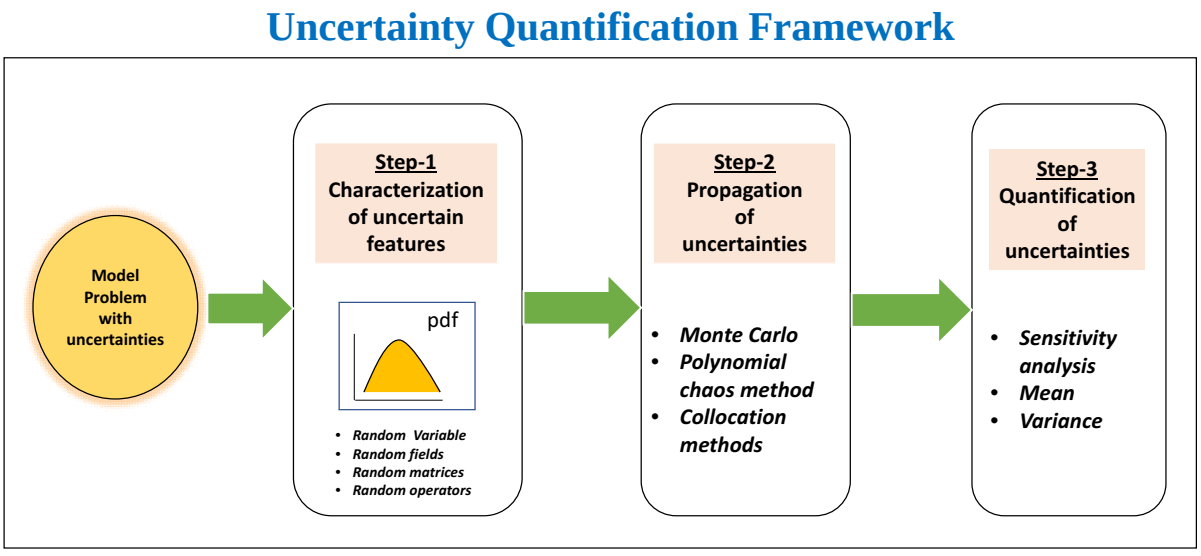


Figure 1.1: Framework of Uncertainty Quantification

1.1 Basic Concepts with Definitions

1.1.1 Uncertainty and Errors

While dealing with the field of UQ, one must be familiar with the difference between errors and uncertainties that can occur in a model [4] as they are distinct concepts which are often confused with one another. *Uncertainty* is considered as a potential deficiency in any stage of the modeling process which is basically in virtue of lack of information. It is generally represented by some type of probability distribution of the occurrence. As we know, the main cause of uncertainty is incomplete information, so it can be minimized by strengthening the knowledge base. On the other hand, *errors* are said to be recognizable deficiencies in any stage of a modeling process which are not a consequence of lack of knowledge. Moreover, they are divided into acknowledged and unacknowledged errors. Acknowledged errors arise from physical approximations made for the simplification of the modeling of a physical process; a defined level of iterative convergence of a numerical method; finite precision arithmetic in a system; transformation of the governing partial differential equations (PDEs) into discrete equations whereas unacknowledged errors arise from mistakes, or blunders. In numerical analysis, numerical methods are designed to solve a problem using numerical approximations as it becomes time consuming to solve it analytically and hence round off errors, truncation errors are introduced in the computational part. So, on a whole, uncertainties are inherently physical whereas errors are purely mathematical in its essence. Although uncertainties influence the output of a computational model, but they should not be mistaken as errors. Moreover, errors can be handled and minimized to a negligible level if the analyst is familiar with the numerical methods and algorithms to be used.

1.1.2 Types of Uncertainties

Uncertainty is further classified either as aleatory, epistemic, or a combination of both [5]. *Aleatory* (Latin word alea meaning dice) uncertainty also referred as non-reducible uncertainty which emerges due to intrinsic variation or physical randomness or due to spatial or temporal variations which cannot be reduced. For eg., different trucks manufactured from a single production line are not exactly the same, a fair dice in which the outcome of a dice throw varies naturally between one to six and is unpredictable, weather or the height of individuals in a population which cannot be minimized by collecting more information. In simple words, we can say an uncertainty which is impossible to reduce is characterized

as aleatory. Hence, it is defined either by a cumulative distribution function (CDF) or a probability density function (PDF). On the other side, *epistemic* uncertainty arises due to incomplete knowledge of the expert, or team of analysts, implementing the simulation and modeling process. These uncertainties are introduced through poor data, deficient measurements, or missing data. However, it can be eliminated by adding sufficient knowledge (which charges resources and time). Moreover, it is traditionally described either as a PDF which provides degree of belief of the analyst or an interval with no associated PDF. Also, these quantities have a fixed value in an analysis, but that fixed value is not known to us. For instance, the elastic modulus for the material in a particular component is probably fixed but poorly known or unknown.

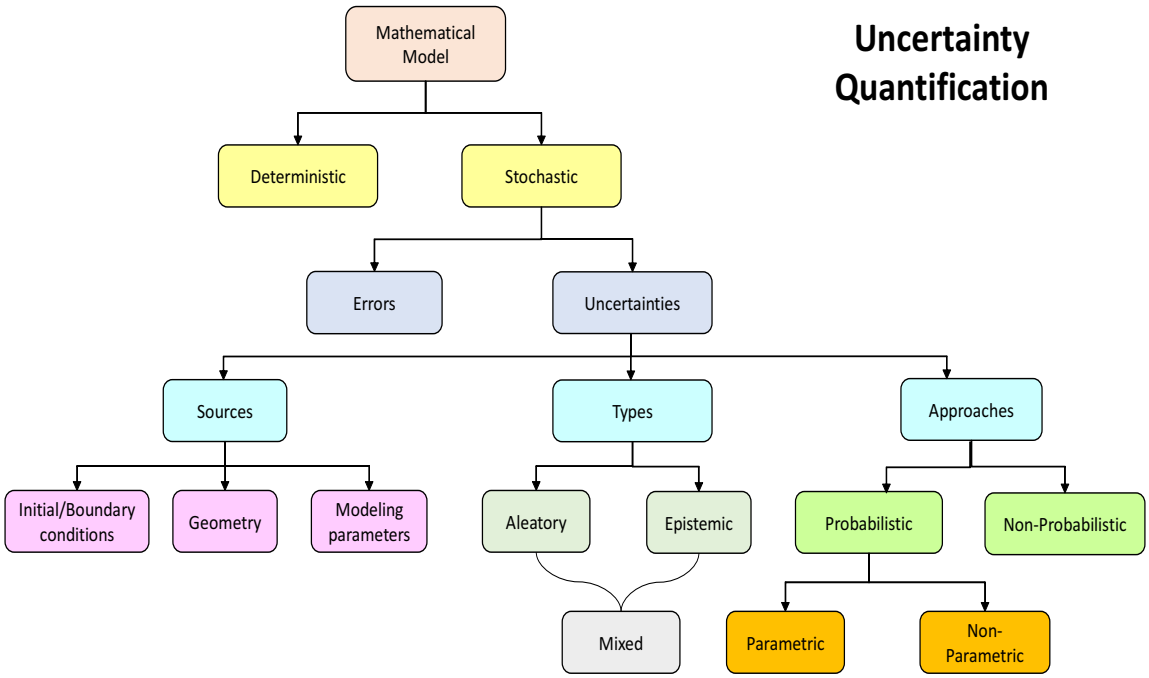


Figure 1.2: UQ Hierachy

1.1.3 Mathematical Models

A mathematical model is an abstract model that employs mathematical concepts to describe the behavior of a real system. Most physical processes are represented using mathematical models with the help of DEs and on the basis of uncertainties and errors, these mathematical models are differentiated into deterministic and stochastic models. In *deterministic models*, the solution comes out to be unique due to the known parameters

governing the equations. This drawback usually raises a practical problem as nature is inherently heterogeneous and the system is often evaluated at small discrete number of locations. Thus, it becomes difficult to rely on solutions of deterministic models even if the physical processes are simple and understandable by deterministic equations. This is basically due to the input parameters, initial and boundary conditions, model geometry etc being not well known or are never known clearly. However, *stochastic methods* can be considered as a tool which collaborates statistics, physics, and uncertainty within a logical theoretical framework. Statistical distributions are used to describe the unknown parameters. On the other hand, the distinct variables depicting the problem are linked to the random model parameters and to each other through (deterministic) physical laws. Therefore, the resulting models take the shape of *stochastic* ordinary DEs (ODEs) or PDEs [6]. Thus, a stochastic model produces numerous equally likely solutions, which enables the modeler to assess the inherent uncertainty of the modeled physical system. However, stochastic models are computationally complex to execute than the deterministic models as they require great computational and statistical expertise. Stochastic modeling is heavily used in various disciplines such as insurance industry, linguistics, stock investing, biology, statistics, and quantum physics.

1.1.4 Probabilistic and Non-Probabilistic Approaches

The literature consists of many frameworks for handling the uncertainties present in a system based on probabilistic and non-probabilistic approaches. *Probabilistic approach* deals with the probability theory by modeling the uncertainties present in a physical system as random mathematical objects whereas *non-probabilistic approaches* are still under development and generally result in interval computations that are expensive when applied to large problems with many variables. This approach generally consists of interval analysis, imprecise probabilities, evidence theory [7], possibility theory [8], and fuzzy set theory [9]. Obviously, from the late 19th century until the late 20th century, the principal theory in quantifying uncertainty in mathematical models had been the probability theory. Probability has a long history going back to the time of Cardano in the 16th century, when gamblers acknowledged the rules of probability in games of chance.

Moreover, probabilistic approaches are further categorized into *parametric* and *non-parametric* approaches [10]. *Parametric* approaches consists of uncertainties in some or all of its parameters by characterizing them as RVs. On the other hand, *non-parametric* approaches are those where uncertain features of the model cannot be related with uncertainties in some or all of the parameters. For eg., when uncertainties originate from modeling sim-

plifications and assumptions or simply contains modeling errors. It was initially given by Soize [11] for UQ where the model uncertainties induced by the modeling errors were taken into consideration that could not be treated by parametric approaches. In computations, uncertainty can arise due to constitutive laws, modeling assumptions, domain geometry, inputs, model structure, model parameters, initial/boundary conditions.

1.1.5 Quantities of interest

The end results after propagating the uncertainties present in a system are the quantities of interest or the statistical descriptors which will express the influence of input uncertainties on the output. The analysts are mainly interested in getting information about the statistical data of the considered quantity, for instance its mean and variance, which partially gives knowledge about the uncertainty characteristics. Moments are a very useful way of describing a RV, since we can characterize the behavior of a RVs using few concepts.

The most commonly used moment is probably the first moment, which is also called the expectation or the mean value. This typically aims at describing the most likely value to be taken by the variable, hence we call it expectation. Let us suppose, we want to measure the speed of wind outside. As we know, due to natural variations or because of measuring instrument, we won't be able to get a steady value. Thus, we get different results after repeated measurements which is quite natural. In such cases, it is best to take more and more readings and calculate its average/mean in order to get the best estimate of the true value. Higher the number of readings you take, closer you get to the true value.

If we get different results after repeated measurements, the next thing which we look up at is the spread of the values. The width of the spread will tell us the uncertainty of the measuring instrument. For instance, a large spread generally means a single reading is quite different from the other readings. Standard deviation (S.D.) comes into form for quantifying this spread or simply we can say it describes how far the variation is around the mean. It basically gives knowledge about how different the individual readings are from the average of the set.

1.1.6 Books and Softwares

There are a number of books available for UQ [10,12–18] which are helpful in understanding the implementation of UQ approaches to various applications. Moreover, there are already many mature UQ software packages developed by the experts in this field. Few of them are Design Analysis Kit for OpTimizAtion (DAKOTA), MIT Uncertainty Quantification

Library (MUQ), an Open source for the treatment of Uncertainties (OpenTURNS), UQ Laboratory (UQLab), UQpy, UQ Toolkit (UQTk), SmartUQ, Uncertainpy, Chaospy which are helpful in the propagation of uncertainties present in a system.

1.2 Numerical Problem- A simple ODE

Rather than getting into the detailed discussion of the stochastic UQ approaches available for the propagation of uncertainties, we first try to understand how uncertainty is introduced into a simple mathematical model. As we know, a simple deterministic ODE has the following form

$$\begin{aligned} \frac{du(t)}{dt} &= -\alpha u(t) \\ u(0) &= \beta. \end{aligned} \tag{1.2.1}$$

Here, randomness can be introduced into the parameters coefficients or into the initial conditions (ICs). Now, we move from deterministic ODE to stochastic ODE by considering α to be the random parameter. The introduction of randomness into deterministic ODE turns it into a stochastic ODE and hence, we have

$$\frac{du(t,\xi)}{dt} = -\alpha(\xi)u(t,\xi) \tag{1.2.2}$$

$$u(0,\xi) = \beta. \tag{1.2.3}$$

where ξ be a RV with some associated PDF or CDF. After getting familiar with the stochastic ODE, we further aim at obtaining the analytical solution of the stochastic ODE as follows:

$$\begin{aligned} \frac{du(t,\xi)}{dt} &= -\alpha(\xi)u(t,\xi) \\ \frac{du(t,\xi)}{u(t,\xi)} &= -\alpha(\xi)dt \\ \int_{u(0,\xi)}^{u(t,\xi)} \frac{du(t,\xi)}{u(t,\xi)} &= -\int_0^t \alpha(\xi)dt \\ ln(u(t,\xi))\Big|_{u(0,\xi)}^{u(t,\xi)} &= -\alpha(\xi)t\Big|_0^t \\ ln(u(t,\xi)) - ln(u(0,\xi)) &= -\alpha(\xi)t \\ ln\left(\frac{u(t,\xi)}{u(0,\xi)}\right) &= -\alpha(\xi)t \end{aligned}$$

$$\frac{u(t,\xi)}{u(0,\xi)} = e^{-\alpha(\xi)t}$$
$$u(t,\xi) = u(0,\xi)e^{-\alpha(\xi)t}$$

Since, this is a simple ODE, so it was easy to find out the analytical solution. However, it becomes difficult to find out the analytical solution of the complex DEs. As we already know the ICs $u(0,\xi) = \beta$, therefore the analytical solution comes out to be

$$u(t,\xi) = \beta e^{-\alpha(\xi)t} \tag{1.2.4}$$

1.3 Propagation of uncertainties

In this section, we review the approaches available in literature which handle the stochastic models by propagating the uncertainties. We will summarize the methodologies of the numerical schemes [19] such as MC, PCE and SCM which are useful in approximating the stochastic models.

1.3.1 Monte Carlo Method

MC method is inherently the most common and simple technique used in literature to propagate the uncertainties present in a system. It was in 1873, when MC method was used to experimentally determine the value of π by throwing matchsticks. Later on, this method was further developed and has been applied in nuclear physics and operation research to compute probabilistic problems. The name and systematic developments of MC method dates from about 1944. The term MC has been coined by Stanislav Ulam and Nicholas Metropolis during the Manhattan Project in the forties and was first published in 1949 [20]. The detailed historical and theoretical aspects of MC methods have been explicitly explained by Hammersley and Handscomb [21].

The algorithm of MC method is as follows:

Algorithm 1.1 MC Method

1: Generate N realizations of the random parameter on the basis of PDF.
2: Corresponding to each realization, get a deterministic solution.
3: Obtain N deterministic solutions corresponding to N deterministic system of equations by using any available deterministic solver.
4: Compute the quantities of interest.

The advantages of using MC method are:

1. This method has an easy implementation and can be employed to any kind of problem covering nonlinear systems as well.
2. An estimate of the error is achievable as part of the solution process.
3. It is considered to be the last resort for high dimensional problems as it is away from the curse of dimensionality.
4. It also shares the advantageous property of not requiring modification of the codes of the original system and thus leads to be a *non-intrusive approach*.

However, it also has few limitations such as

1. It needs thousands or sometimes millions of samples in order to reach to the required accuracies.
2. It has notoriously slow convergence rate of $O(N^{-1/2})$.
3. It is computationally expensive.
4. Also, MC method is a very time-consuming method due to high processing time of a single realization or due to requirement of large number of realizations to get to an exact result, which makes it difficult to apply it in complex simulations.

MC methods have been used in various applications in the context of quantifying uncertainties present in a system. Americo Cunha Jr. et al. quantified uncertainty using MC method in cloud computing setting [22]. MC methods [20, 23] have been applied for many years to conduct simulations when randomness was considered. In 2011, Siva et al. [24] studied the influence of uncertainties to power and thrust coefficients of a helicopter with the help of MC method. In 2014, Dörr et al. [25] demonstrated the use of different MC sample generators in the context of numerical UQ. Moreover, they gave a brief introduction on the statistical fundamentals of the MC uncertainty framework.

1.3.1.1 Variants of MC method

Due to the limitations of MC method, researchers have worked on developing its various extensions. Therefore, there are a number of variants of MC methods and only a few are discussed here. In 1979, stratified and Latin Hypercube sampling (LHS) [26] were examined as an alternative to simple random sampling in MC simulations. The main feature of LHS [27] is that, in contrast to simple random sampling, it simultaneously stratifies on all input dimensions. Further, MC method has been extended to Quasi- MC

(QMC) [28], Multilevel MC (MLMC) [29] and Multilevel QMC (MLQMC) [30]. Multilevel methods [31] have turned out to considerably minimize the asymptotic computational effort of the traditional MC during UQ problems. It raises only proportionally to the price of solving the deterministic problem in specific situations. These methods depend on smart combination of large number of computationally inexpensive low resolution samples and a comparatively small number of computationally more expensive higher resolution samples.

On the other hand, quasi random sequences are considered as the deterministic substitutes to pseudo-random sequences which are employed for QMC method. Pseudo-random sequences are known for imitating the features of random sequences whereas quasi-random sequences are good at producing better uniformity and therefore, grab faster convergence for quadrature formulae [32]. Hence, QMC makes use of inputs which are uniformly spaced from already sampled quasi-random sequences to evaluate the statistical descriptors. Thus, it provides an improved rate of convergence than MC along with considerable minimization of the computational cost. Further, MLMC and MLQMC needs a hierarchy of meshes defined a prior for its effective functioning. In MLMC method, coarsest grids covers up most of the uncertainty, and hence, the cost associated along with the amount of realizations required on the finest grids is dramatically decreased [33].

In 2010, Cliffe et al. [31] demonstrated the effectiveness of MLMC method over the traditional MC method by applying it on elliptic PDEs in groundwater flow. In 2011, Barth et al. [34] proposed MLMC finite element method which helps in the reduction of computational cost. One of its application is in stochastic elliptic equation with random coefficients by assuming smoother coefficient fields. In 2012, Mishra et al. [35] extended the MLMC algorithm to approximate the stochastic shallow water equations with random field source terms by taking into account one and two dimensions. In 2013, Müller et al. [36] studied the performance of MLMC in combination with a streamline-based solver to examine uncertain Buckley-Leverett transport and two phase flow in random porous media. In 2016, Mishra et al. proposed MLMC finite volume method for the uncertain acoustic wave propagation. In 2017, Scheichl et al. [37] provided complexity analysis as well as full convergence of the ratio estimator where MC, QMC or MLMC methods are used as estimators. Recently, Croci et al. [38] compared the performance of MC, QMC and MLMC in uncertain coefficients obtained from MRI-studies of the glymphatic system in human beings. In most of the research papers associated with UQ, it has been observed that MC method [39] has been compared with other UQ methods such as PCE, SCM etc., and these are discussed in the later section.

1.3.1.2 Quantities of interest

The expectation of the solution by MC method is simply computed as

$$\mathbb{E}[u] = \frac{1}{N}\sum_{i=1}^{N} u_i. \tag{1.3.1}$$

On the other hand, variance is computed as

$$\sigma^2(u) = \mathbb{E}[u^2] - (\mathbb{E}[u])^2 = \frac{1}{N}\sum_{i=1}^{N} u_i^2 - \left(\frac{1}{N}\sum_{i=1}^{N} u_i\right)^2. \tag{1.3.2}$$

1.3.2 Spectral expansions

Spectral methods are known for describing problems of increasing complexity by taking up different series expansions with improved analytical capacity. Let us now discuss in detail Karhunen-Loéve expansion, PCEs and its variants.

1.3.2.1 Karhunen-Loéve (K-L) expansion

As we know, MC method involves quite large number of points for sampling so as to achieve a good approximation. So, we have another theoretically more appealing approach which begins by expanding the functions in a Fourier-type series as follows:

$$u(x,\theta) = \sum_{i=0}^{\infty} \sqrt{\lambda_i}\xi_i(\theta) f_i(x) \tag{1.3.3}$$

where λ_i is some constant, $\{f_i(x)\}$ is an orthonormal set of deterministic functions, θ be the space of random events Ω and $\{\xi(\theta)\}$ is a uncorrelated set of RVs to be determined. This expansion is known as K-L expansion and it was derived independently by many investigators [40–42].

Let us denote random process by $u(x,\theta)$ where x is the position vector defined over the space $\mathcal{D}$. Suppose $\tilde{u}(x)$ be the expected value of $u(x,\theta)$ and $\mathcal{C}(x_1,x_2)$ be the covariance function which satisfies few properties such as boundedness, symmetric and positive definiteness. Therefore, we have the spectral representation of covariance kernel as follows [43]

$$\mathcal{C}(x_1,x_2) = \sum_{i=0}^{\infty} \lambda_i f_i(x_1) f_i(x_2) \tag{1.3.4}$$

where λ_i denotes the eigenvalue of the covariance kernel and $f_i(x)$ is its eigenvector. Specifically, they satisfy the integral equation

$$\int_{\mathcal{D}} \mathcal{C}(x_1, x_2) f_i(x) dx_1 = \lambda_i f_i(x_2) \tag{1.3.5}$$

Moreover, the eigenfunctions form a complete set and are orthogonal due to the positive definiteness and the symmetry of the covariance kernel. So, we can normalize it in the following way

$$\int_{\mathcal{D}} f_i(x) f_j(x) dx = \delta_{ij} \tag{1.3.6}$$

where δ_{ij} is the Kronecker delta. Therefore, we have

$$u(x, \theta) = \tilde{u}(x) + \gamma(x, \theta) \tag{1.3.7}$$

where $\gamma(x, \theta)$ is a process with zero mean and $\mathcal{C}(x_1, x_2)$. Now, we can expand $\gamma(x, \theta)$ using eigenfunctions $f_i(x)$ as

$$\gamma(x, \theta) = \sum_{i=0}^{\infty} \xi_i(\theta) \sqrt{\lambda_i} f_i(x). \tag{1.3.8}$$

such that

$$\langle \xi_i(\theta) \rangle = 0, \quad \langle \xi_i(\theta) \xi_j(\theta) \rangle = \delta_{ij}$$

After inserting Eq. (1.3.8) in Eq. (1.3.7), we have random process as follows:

$$u(x, \theta) = \tilde{u}(x) + \sum_{i=0}^{\infty} \xi_i(\theta) \sqrt{\lambda_i} f_i(x). \tag{1.3.9}$$

Truncating the series in Eq. (1.3.9) to N, we get

$$u(x, \theta) = \tilde{u}(x) + \sum_{i=0}^{N} \xi_i(\theta) \sqrt{\lambda_i} f_i(x). \tag{1.3.10}$$

which is the required solution. For Gaussian random field, $\xi(\theta)$ are taken to be independent standard normal RVs. Otherwise, the joint distribution is almost impossible to obtain. Thus, the K-L expansion [44] is primarily suitable for the discretization of Gaussian fields. However, Phoon et al. [45] employed K-L expansion for the direct modeling of non-Gaussian random fields. Later on, Matthies and Keese [46] performed K-L expansion on the non-

Gaussian field and projected the RVs engaged in the expansion to an independent Gaussian RV space.

The advantages of K-L expansion are

1. It converts stochastic equations into deterministic equations.
2. It is optimal in accordance with the minimization of mean square error.

The disadvantages of K-L expansion are

1. Non-availability of analytical solutions of most covariance functions.
2. Evaluation of covariance eigen-pairs is computationally expensive.
3. For the calculation of the K-L basis, a large eigenvalue problem needs to be solved, even if the spatial discretization is only moderately fine.

We will next discuss the formulation of the generalization of K-L expansion known as PCE when the covariance function is not known in advance.

1.3.2.2 Polynomial Chaos expansion method

As discussed earlier, we know, the covariance function is required for the implementation of the K-L expansion. Hence, it cannot be applied for the solution process in the absence of covariance function and their corresponding eigenfunctions. In such situations, we need an alternative expansion such that basis consists of random functions which are known prior and the deterministic coefficients can be evaluated by minimizing some error norm arising from finite representation. PCEs are Fourier like series expansions that describe randomness in stochastic dynamical systems and hence fall in the category of spectral methods. PCE was first developed by Norbert Wiener [47] in 1938 which begins by expanding random processes of second order on the basis of orthogonal polynomials. Basically, processes with finite variance are known as second-order random processes and are applicable to most of the physical processes. Hermite polynomials were the first polynomials employed to handle stochastic models with Gaussian RVs.

Consider a probability space $(\Omega, \Sigma, \mathbb{P})$, such that the sample space is denoted by Ω, Σ indicates the σ-algebra on Ω, and $\mathbb{P}$ be the probability measure (Ω, Σ). Suppose we have set of standard Gaussian RV on Ω, i.e., $\{\xi_i(\theta)\}_{i=1}^{\infty}$. So, in accordance with the Cameron-Martin theorem [48], Hermite chaos expansion converges to any stochastic process in L_2 sense i.e., it can be practiced for the representation of any stochastic process with finite

variance. Therefore, $u(x,\theta)$ can be described as follows:

$$\begin{aligned} u(x,\theta) &= a_0 H_0 + \sum_{i_1=1}^{\infty} a_{i_1} H_1(\xi_{i_1}) + \sum_{i_1=1}^{\infty}\sum_{i_2=1}^{i_1} a_{i_1 i_2} H_2(\xi_{i_1},\xi_{i_2}) \\ &\quad + \sum_{i_1=1}^{\infty}\sum_{i_2=1}^{i_1}\sum_{i_3=1}^{i_2} a_{i_1 i_2 i_3} H_3(\xi_{i_1},\xi_{i_2},\xi_{i_3}) + \dots \end{aligned} \tag{1.3.11}$$

The above expression replaces the continuous integrals by summation and is basically the discrete version of the actual Wiener PCE. In Eq. (1.3.11), H_n signifies Hermite polynomials and ξ signifies the standard Gaussian RV with unit variance and zero mean. The vector $(\xi_{i_1},\xi_{i_2},\dots,\xi_{i_n})$ represents n independent RVs. Together $H_n(\xi_{i_1},\xi_{i_2},\dots,\xi_{i_n})$ denotes the Hermite-chaos of order n. We can also write Eq. (1.3.11) for notational convenience as

$$u(x,\theta) = \sum_{k=0}^{\infty} \hat{a}_k \Psi_k(\xi). \tag{1.3.12}$$

where there is one to one correspondence between $\Psi_k(\xi)$ and $H_n(\xi_{i_1},\xi_{i_2},\dots,\xi_{i_n})$. Also, the Hermite chaos expansion is effectively used for stochastic DEs with both Gaussian and certain types of non-Gaussian inputs [49–52]. But the optimal exponential convergence rate will not be realized for general non-Gaussian random inputs [53]. In fact, we come across cases where the convergence rate is extremely deteriorated.

Distribution	Density function	Polynomial	Support range	Weight function
Normal	$\frac{1}{\sqrt{2\pi}} e^{\frac{-x^2}{2}}$	Hermite	$(-\infty,\infty)$	$e^{\frac{-x^2}{2}}$
Beta	$\frac{(1-x)^{\alpha}(1+x)^{\beta}}{2^{\alpha+\beta+1}B(\alpha+1,\beta+1)}$	Jacobi	$[-1,1]$	$(1-x)^{\alpha}(1+x)^{\beta}$
Uniform	$\frac{1}{2}$	Legendre	$[-1,1]$	1
Exponential	e^{-x}	Laguerre	$(0,\infty)$	e^{-x}
Gamma	$\frac{x^{\alpha}e^{-x}}{\Gamma(\alpha+1)}$	Generalized Laguerre	$(0,\infty)$	$x^{\alpha}e^{-x}$

Figure 1.3: Connection between Wiener-Askey PCE and their continuous RVs.

Later on, the Wiener-Askey PCE known as generalized PCE (gPC) [54] was introduced as a generalization of the initial Wiener PCE. In gPC, the choice of polynomials is not only

limited to Hermite polynomials instead one can choose the orthogonal polynomials from the Askey-scheme such as beta, Gamma, uniform, binomial, poisson, hypergeometric and so on. Now, we can express random process $u(x,\theta)$ using gPC as follows:

$$\begin{aligned} u(x,\theta) &= d_0\Gamma_0 + \sum_{i_1=1}^{\infty} d_{i_1}\Gamma_1(\varsigma_{i_1}(\theta)) + \sum_{i_1=1}^{\infty}\sum_{i_2=1}^{i_1} d_{i_1 i_2}\Gamma_2(\varsigma_{i_1}(\theta),\varsigma_{i_2}(\theta)) \\ &\quad + \sum_{i_1=1}^{\infty}\sum_{i_2=1}^{i_1}\sum_{i_3=1}^{i_2} d_{i_1 i_2 i_3}\Gamma_3(\varsigma_{i_1}(\theta),\varsigma_{i_2}(\theta),\varsigma_{i_3}(\theta)) + \dots \end{aligned} \tag{1.3.13}$$

where Γ_i denotes any of the polynomials from Wiener Askey scheme and $\Gamma_n(\varsigma_{i_1},\varsigma_{i_2},\dots,\varsigma_{i_n})$ describes Wiener polynomial chaos of order n. Again, we can rewrite Eq. (1.3.13) as

$$u(x,\theta) = \sum_{k=0}^{\infty} \hat{d}_k \Psi_k(\varsigma). \tag{1.3.14}$$

where there is a one-to-one correspondence between the $\Psi_k(\varsigma)$ and $\Gamma_n(\varsigma_{i_1},\varsigma_{i_2},\dots,\varsigma_{i_n})$. These orthogonal polynomials $\Psi_k(\varsigma)$ have the following properties:

$$\begin{aligned} \langle \Psi_0(\varsigma) \rangle &= 1, \\ \langle \Psi_i(\varsigma) \rangle &= 0, \quad i > 0, \\ \langle \Psi_i(\varsigma), \Psi_j(\varsigma) \rangle &= \delta_{ij}, \quad \text{for} \quad j,k > 0. \end{aligned}$$

where $\langle \cdot,\cdot \rangle$ denotes the inner product of the variables. Fig.1.3 gives the connection between the continuous RVs and the Wiener-Askey polynomials. It can be noticed from this figure that the weighting functions of some polynomials from the Wiener-Askey scheme is similar to probability function of random distributions. Accordingly, we can pick out the kind of variables ς in the polynomials with respect to the given random distribution. Usually, we assume random inputs to be fully correlated or independent as PCE require statistically independent RVs. In practice, the PCE is truncated in order p as well as in dimension n and thus, we get the resulting finite PCE as follows:

$$u(x,\theta) = \sum_{k=0}^{P} \hat{d}_k \Psi_k(\varsigma_1,\varsigma_2,\dots,\varsigma_n), \tag{1.3.15}$$

which contains number of terms as follows:

$$P + 1 = \frac{(n+p)!}{n!p!}.$$

The advantages of PCE are

1. It achieves fast convergence due to spectral expansion.
2. It is a efficient method due to its orthogonality property.

The disadvantages of PCE are

1. It suffers from 'curse of dimensionality' which translates into the resolution of very large systems of equations or equivalently countless calls to the deterministic solver.
2. It becomes inappropriate and computationally expensive when the dimension of RVs is large.
3. It may diverge for long-time integration.
4. It may become inefficient for problems with singularity or low regularity in the parametric space.

Throughout this thesis, the dimension of ς is taken to be one.

1.3.2.3 Wavelet based gPC

Usually, gPC may have slow convergence or fails to converge in problems which consists of discontinuities or sharp dependence on the random space even in short-time integration. Therefore, to enhance the performance of gPC, a more robust and efficient scheme based on wavelets is introduced into the field of UQ. Wavelets being local waves are known for expressing discontinuities or steep gradients more accurately than the global basis. The first wavelet based gPC was developed in 2004 and named as Wiener Haar gPC [55] which addressed problems related to discontinuities in stochastic space. Haar wavelets are popular for their high aptitude in the modeling of nonlinear functions with fast variations and reducing the Gibbs phenomenon in the presence of discontinuities. Moreover, they are defined in a hierarchy on various resolution levels, exemplifying progressively finer features of the solution with increasing resolution. They are said to be localized due to non-overlapping support within each resolution level. In wavelet based gPC, we replace the gPC basis with the wavelet basis by employing scaling and wavelet functions of the wavelets. In both Wiener-Legendre and Wiener-Hermite expansions, it has been analyzed that the coefficients of the higher order terms progressively gain and lose dominance over the lower-order coefficients which makes global basis to fail.

In 2006, Petit et al. [56] examined simulation of stochastic aero-elastic limit cycles and analyzed that Wiener-Haar expansion entirely eliminates the waste of energy at large times for

the sinusoidal process as well as for the response of a nonlinear system. In 2007, Le Maître et al. [57] introduced another efficient wavelet based PCE by employing multi-wavelets which greatly improved the robustness of the Galerkin process in the presence of discontinuities. They proposed adaptive methods for multi-resolution analysis, which relies on a multi-wavelet basis of compact piecewise-smooth polynomial functions. In 2009, Najm et al. [58] utilized intrusive multi-wavelet PCE UQ method for the first time within the framework of non-isothermal ignition of a hydrocarbon chemical system. In 2012, Nechak et al. [59] exploited Wiener-Haar model to approximate the vibratory level of the considered brake system by taking into account the random friction coefficient. Haar wavelets minimize the Gibbs phenomenon in the nearness of discontinuities in the random dimension. In 2013, Sahai and Pasini [60] used Haar-wavelet expansion to grab the discontinuities in the probability distributions of the output variables in hybrid dynamical systems. In 2014, Wang and Sarris [61] proposed a new intrusive polynomial chaos finite-difference time-domain method for UQ in electromagnetic time-domain modeling by employing Haar wavelet basis. In 2018, Shi et al. [62] developed a data-driven temporal-dependency Haar expansions UQ approach by decomposing temporal dependency of load uncertainties over differing time-scales. They used integration of two-dimensional Haar expansions with MC estimators.

1.3.2.4 Multi element based gPC

Another method for handling discontinuities is the multi element gPC (ME-gPC) proposed by Wan and Karniadakis [63–65] which breaks down the space of random inputs into small baby elements. Next, it creates a new RV in each element and after that gPC is applied again. Also, it can preserve a relative low polynomial order for gPC in each element as the degree of perturbation in each element is minimized proportionally to the size of random elements. This ME-gPC can reach to *hp*-convergence (as in spectral elements for spatial discretization), where p is the PCE order and h is decided by the size of random elements. In 2005, Wan and Karniadakis [65] formulated an adaptive ME-gPC method to handle the discontinuities and long-term integration in stochastic DEs with *hp*-discretization of the random space. Moreover, in 2006, they both [66] employed ME-gPC method to address stochastic inputs with arbitrary probability measures. In 2006, Wan and Karniadakis [67] presented an adaptive ME-gPC method, which can attain *hp*-convergence in random space along with its applications. In 2009, Bruno et al. [68] proposed a ME-gPC method in viscous fluid flow together with an efficient deterministic computational approach in order to achieve accurate approximations of the output RVs at a reasonable cost. In 2010, Prempraneerach [69] achieved effective resolution of stochastic discontinuous solutions by

the ME decomposition of random space. Moreover, the ME-gPC is shown to be very efficient to predict the friction-induced vibrations in a nonlinear uncertain dry friction system [70, 71]. Trinh et al. [72] have used ME-gPC to perform stability analysis of a clutch system with a significant reduction of the computational cost in comparison with the standard gPC. Recently, Snoun et al. [73] applied ME-gPC based method to locate the discontinuity as well as to predict the propensity of the system to undergo an harmless steady-state regime.

1.3.2.5 Computation of coefficients

The unknown deterministic coefficients of the PCE representation can be calculated in two ways-intrusive or non-intrusive. These schemes are explained below.

1. **Intrusive Galerkin projection**

   For the propagation of uncertainty in computational simulations, PCE method starts by substituting the PCE of uncertain parameters and variables in the governing equations. Then, the unknown deterministic coefficients are calculated by projecting the resulting equations onto the basis functions for various modes. This approach is known as the *intrusive approach* which needs modification of deterministic codes. As the modification of the existing codes can be challenging to implement, expensive and time consuming, therefore, in complex computational problems, this approach may be inconvenient or difficult to apply. Since for executing the quadratic term and cubic term, we use multiplicative tensor and triple tensor product respectively whereas in complex situations, we need higher order tensor product which are difficult to calculate.
   This thesis focuses on variants of PCE methods with projection through Galerkin method, so, we must know how the implementation of PCE works out with intrusive Galerkin method. The algorithm of PCE is as follows:

Algorithm 1.2 PCE method-*Intrusive approach*

1: Select the polynomial in accordance with the PDF of the RV.
2: Insert the PCEs of uncertain parameters and solution u.
3: Truncate the polynomial order.
4: Apply the Galerkin projection in order to get the deterministic system of equations.
5: Compute the statistical descriptors.

We consider the same stochastic ODE i.e., Eq. (1.2.2) with uncertain parameter α.

The PCE of solution u and α is written as follows:

$$u(t,\xi) = \sum_{i=0}^{P} u_i(t)\psi_i(\xi), \qquad \alpha(\xi) = \sum_{i=0}^{P} \alpha_i \psi_i(\xi). \tag{1.3.16}$$

After that, we insert the PCE of u and α in Eq. (1.2.2)

$$\frac{d}{dt}\Big(\sum_{i=0}^{P} u_i(t)\psi_i(\xi)\Big) = -\sum_{i=0}^{P} \alpha_i\psi_i(\xi) \sum_{j=0}^{P} u_j(t)\psi_j(\xi),$$
$$\frac{d}{dt}\Big(\sum_{i=0}^{P} u_i(t)\psi_i(\xi)\Big) = -\sum_{i=0}^{P}\sum_{j=0}^{P} \alpha_i u_j(t)\psi_i(\xi)\psi_j(\xi)$$

Next, we take inner product with respect to ψ_k, i.e., applying stochastic Galerkin projection as follows:

$$\frac{d}{dt}\Big(\sum_{i=0}^{P} u_i(t)\langle\psi_i(\xi)\psi_k(\xi)\rangle\Big) = -\sum_{i=0}^{P}\sum_{j=0}^{P} \alpha_i u_j(t)\langle\psi_i(\xi)\psi_j(\xi)\psi_k(\xi)\rangle$$
$$\frac{du_k(t)}{dt} = -\sum_{i=0}^{P}\sum_{j=0}^{P} \alpha_i u_j(t)\frac{\langle\psi_i(\xi)\psi_j(\xi)\psi_k(\xi)\rangle}{\langle\psi_i(\xi)\psi_k(\xi)\rangle}$$
$$\frac{du_k(t)}{dt} = -\frac{1}{k!}\sum_{i=0}^{P}\sum_{j=0}^{P} \alpha_i u_j(t)\mathscr{C}_{ijk}, \quad k = 0,1,\ldots,P \tag{1.3.17}$$

where C_{ijk} is the multiplicative tensor which is given by

$$\mathscr{C}_{ijk} = \langle\psi_i(\xi)\psi_j(\xi)\psi_k(\xi)\rangle \tag{1.3.18}$$

Now, we look at the IC and follow the same procedure as earlier.

$$\sum_{i=0}^{P} u_i(0)\psi_i(\xi) = \beta = 1. \tag{1.3.19}$$
$$\sum_{i=0}^{P} u_i(0)\langle\psi_i(\xi)\psi_k(\xi)\rangle = 1.$$

This implies that $u_0 = 1,\ u_1 = 0, \ldots, u_P = 0$.

1.3.2.6 Quantities of interest

PCEs, on the other hand, shares the advantage of requiring no extra computational effort for calculating the output statistical descriptors. The mean of the PCE can be estimated as follows:

$$\begin{aligned}\mathbb{E}(u)(t) &= \left\langle \sum_{i=0}^{P} u_i(t)\psi_i, 1 \right\rangle, \\ &= \sum_{i=0}^{P} u_i(t)\langle \psi_i, 1\rangle, \\ &= u_0(t)\langle \psi_0, 1\rangle + u_1(t)\langle \psi_1, 1\rangle + u_2(t)\langle \psi_2, 1\rangle + ... + u_P(t)\langle \psi_P, 1\rangle, \\ &= u_0(t). \end{aligned} \tag{1.3.20}$$

Since, we have a result in accordance with the orthogonality of the polynomials to ψ_0, i.e., $\langle \psi_i, 1\rangle = 0$ for $i > 1$. So, without loss of generality, we assume $\psi_0 \equiv 1$. With the expectation defined, we move forward to defining the centered moments. Variance is something which tells us the deviation of output from the mean. Therefore, we have

$$\begin{aligned}\sigma^2(u) &= \mathbb{E}(u^2)(t) - (\mathbb{E}(u)(t))^2, \\ &= \left\langle \sum_{i=0}^{P}\sum_{j=0}^{P} u_i(t)u_j(t)\psi_i, \psi_j \right\rangle - u_0^2(t), \\ &= \sum_{i=0}^{P}\sum_{j=0}^{P} u_i(t)u_j(t)\langle \psi_i, \psi_j\rangle - u_0^2(t), \\ &= \sum_{i=0}^{P} u_i^2(t)\langle \psi_i, \psi_i\rangle - u_0^2(t), \\ &= \sum_{i=1}^{P} u_i^2(t)\langle \psi_i, \psi_i\rangle. \\ &= \sum_{i=1}^{P} u_i^2(t). \end{aligned} \tag{1.3.21}$$

1.3.2.7 Literature - Stochastic Galerkin Method

The initial work of Wiener on homogenous chaos was carried forward in the decade of 1960s in turbulence [74–76]. However, it doesn't gained much popularity for a long period of time due to its slow convergence for turbulent field [77–79]. Further, Le

Maître et al. [80, 81] applied it to the thermo-fluid applications by considering low Mach number flow. Later on, the theory of PCE was extended to gPC [52, 54, 82] by Xiu et al. for examining different classes of orthogonal polynomials in the Askey scheme together with their associated stochastic processes. In 2003, Xiu and Karniadakis [83] applied gPC for the solution of two dimensional unsteady heat transfer problem with random input parameters and the results were validated through MC method. Then in the same year, Debusschere et al. [84] applied PCE to the electrochemical flow in micro-fluidic system. Further, Reagan et al. [85] employed PCE in a Hydrogen-Oxygen reaction system in supercritical water conditions. Moreover, they implemented an intrusive UQ technique [86] along with a comprehensive formulation in reacting flow by taking in a fully spectral chemical source term. Also, Reagan et al. [87] gave a new PC based framework for the assessment of high-order stochastic sensitivities as well as predicted model output uncertainties. They also applied both non-intrusive spectral projection construction and traditional SA for homogeneous ignition. Later on, in 2004, PCE was combined with wavelets [55, 88] by taking uniform random inputs in order to handle the discontinuities for which standard gPC ceases to converge.

Lucor et al. [89] represented the random input process using K-L expansion and solved the model using gPC and compared it with the MC simulations. There are many research papers available in literature where K-L spectral decomposition has been incorporated into the PCE [89–92]. In 2005, Chen et al. [93] studied the stochastic isentropic flow in dual-throat nozzle with equal throat areas using PCE and compared it with MC simulations. In 2006, Hou et al. [94] engaged Wiener chaos for solving the stochastic Burger equation together with Navier Stokes equation modeling the roll-up of thin vorticity layers through the Kelvin-Helmholtz instability. Further, Hover and Triantafyllou [95] exercised PCE to the control problems for evaluating transient response and long term stability. Over the past few decades, gPC scheme has been effectively practiced in various engineering problems such as random vibrations [96], soil mechanics [97], elasticity problems [98, 99], reference evapotranspiration [100], radio indoor channels [101], fluid dynamics [102], air quality models [103], prediction of vehicle dynamics [104–106], multiphysics microfluidics [84], chemical reactions [107, 108] and many more. "A. Sandu et al. [104] exploited gPC theoretically and gave the computational aspects for modeling complex nonlinear multi-body dynamic systems in the presence of parametric and external uncertainty. Meanwhile, the same research group [105] studied the numerical investigations of a quarter car model. Later on, Cheng and Sandu [103] applied PCE for UQ by incorporating uncertainties in air quality models namely emissions and wind fields." Later on, in 2008, Witteveen

and Bijl [109] applied gPC to the stochastic advection diffusion problems including both 1D and 2D pipe flows. Further, Gottlieb and Xiu [110] studied the properties of PCE in hyperbolic problems by taking into account a simple scalar wave equation model with uncertain wave speed. Furthermore, on the basis of PCE, Lovett et al. [111] derived the formal modeling theory so as to develop a computer-aided design simulator for stochastic linear networks.

In 2010, Soize and Desceliers [112] presented a new technique for the high dimensional case in order to compute the high number of independent realizations of a PCEs in view of an arbitrary measure along with the preservation of the orthogonality properties. Later on, Gerritsma et al. [113] considered time-dependent gPC which highlights the way the PDF of the solution emerges as a function of time. Further, Shen et al. [114] presented PCE so as to quantify the growth of randomness in the Lorenz system. In the meantime, Prabhakar et al. [115] analyzed the stochastic Mars entry, descent, and landing problem by employing both gPC and MC methods. Then, Jiang and Li [116] developed a more computationally efficient hybrid framework via gPC and spectral decomposition for quantifying the propagation of the uncertainty in state trajectories of a low-lift Mars entry vehicle by considering uncertain dynamical parameters and ICs.

In 2011, Panayirci and Schuëller [117] discussed the latest developments focussing on the application of the PCE method particularly within the computational solid mechanics. Later on, in 2012, Santonja and Chen-Charpentier [118] set forth the use of PCE in the field of epidemics by considering stochastic epidemiological mathematical models based on ODEs. Further, Sepahvand [119] investigated the application of gPC in free vibration of orthotropic plates with random moduli. Furthermore, for the high dimensional case, Perrin et al. [120] developed a new methodology for the identification of PCE along with performing relevant convergence analysis with respect to an arbitrary measure. Then, Shi and Zhang [121] gave the error analysis of gPC for non-linear random ODEs, which highlights that the error basically depends on the numerical and projection error. In 2012, Pulch [122] combined advantages from both the intrusive and the non-intrusive approach to boundary value problems of systems of stochastic ODEs. In 2012, Pascual and Adhikar [123] propagated both parametric and non-parametric uncertainties in elliptic PDEs in such a way that parametric uncertainty is modelled using random field and K-L expansion is used to discretise it.

The theory and methods of PCE are not well understood by most statisticians, so, in 2013, O'Hagan [124] and Kim et al. [125] presented a theoretical and historical

overview on PC methods respectively. Also, Desai et al. [126] performed a Gauss-Hermite quadrature based PCE for parametric uncertainties taken by considering modeling and measurement inaccuracies. In the meantime, Kim et al. [127] combined gPC with convex relaxation methods by incorporating both model uncertainty and exogenous disturbances to deal with a stochastic system model. Further, Austin and Sarris [128] expanded time domain electric and magnetic fields using gPC in stochastic microwave circuits and showed that the results coincide strongly with the MC simulations.

Motivated by hierarchical UQ, in 2014, Zhang et al. [129] proposed a formulation to ascertain gPC basis functions and Gauss quadrature rules from surrogate models. Later on, Navarro et al. [130] constructed a basis of the probability space of orthogonal polynomials for general multivariate distributions with correlations between the uncertain input. Further, an efficient long-time uncertainty propagation method has been presented by Luchtenburg et al. [131] in dynamical systems by taking into account random parameters and ICs. The intermediate short-time flow maps are approximated by spectral polynomial bases, as in the gPC method, and long-time flow map is constructed by using flow map composition. In 2015, Delgado and Kumar [132] formulated a Galerkin UQ method for the traditional Biot model of poroelasticity. In 2016, Vittaldev et al. [133] united PCE with Gaussian mixture models to handle the random spacecraft model with initial Gaussian errors. Later on, Daròczy et al. [134] applied PCE to transient, turbulent flow simulations of a variable-speed H-Darrieus turbine, by considering randomness in angular velocity and in preset pitch angle. Further, Jia et al. [135] integrated PCE method with MC simulations in the SACROC unit to quantify uncertainty of CO_2 storage in various phases. Furthermore, Du and Roblin [136] combined PCE and spherical modes expansion method to establish a compact and accurate statistical model by minimizing the experimental design cost.

In 2017, Savin and Faverjon [137] provided the analytical formulas for the third-order moments of Jacobi, generalized Laguerre and Hermite polynomials along with their easy MATLAB implementation. These formulas are highly useful in the post processing step of the PCE. Later on, Slim et al. [138] proposed a skilled surrogate modeling scheme on the basis of PCE to determine the field quality by taking into account tolerances and misalignments along with SA. Further, Mühlpfordt et al. [139] provides a way to choose the PCE order in such a way that the truncation error fades away. In the recent years, Xu et al. [140] utilized gPC as well as ME-gPC methods for the quantification of randomness in the statistical power system

dynamic analysis. Moreover, these two PC-based methods significantly minimize the computational effort in comparison to the traditional MC method. Recently, Hu et al. [141] analytically handled parametric uncertainty in ion channel models of mouse ventricular cell by combining gPC with Galerkin projection and further propagated the randomness across various organizational levels of tissue and cell.

2. **Non-Intrusive Polynomial chaos (NIPC)**

There are number of non-intrusive PC approaches available in the literature which are based on sampling [85, 142, 143] or quadrature methods [142, 144] to determine the projected polynomial coefficients. However, there is another computationally less expensive approach, known as NIPC approach, which is neither dependent on sampling nor on the quadrature methods for determining the PCE coefficients. It was first introduced by Walters [145] and has a straightforward implementation in any stochastic problem. The method is known for its non-intrusive nature, i.e., no modification to the deterministic code is required in any form. Walters employed NIPC to a stochastic heat transfer problem with geometric uncertainty. Later on, Hosder et al. [146] presented this computationally inexpensive NIPC method for the propagation of uncertainty in fluid dynamics. The algorithm of the NIPC is as below:

Algorithm 1.3 Non-intrusive PC

1: For any stochastic variable u

$$u(x,\xi) = \sum_{i=0}^{P} u_i(x)\psi_i(\xi)$$

where $\xi = (\xi_1, \xi_2, \ldots, \xi_n)$ be $n-$dimensional RV.

2: We choose $P+1$ vectors $\xi_i = \{\xi_1, \xi_2, \ldots, \xi_n\}_i$, $i = 0, 1, \ldots, P$ in random space.

3: Now, evaluate the deterministic code at these points.

4: The known solutions at left hand side at chosen random points leads to a linear system of equations as follows:

$$\begin{pmatrix} \psi_0(\xi_0) & \psi_1(\xi_0) & \cdots & \psi_P(\xi_0) \\ \psi_0(\xi_1) & \psi_1(\xi_1) & \cdots & \psi_P(\xi_1) \\ \vdots & \vdots & \ddots & \vdots \\ \psi_0(\xi_P) & \psi_1(\xi_P) & \cdots & \psi_P(\xi_P) \end{pmatrix} \begin{pmatrix} u_0(x) \\ u_1(x) \\ \vdots \\ u_P(x) \end{pmatrix} = \begin{pmatrix} u(x,\xi_0) \\ u(x,\xi_1) \\ \vdots \\ u(x,\xi_P) \end{pmatrix}$$

In this method, the deterministic code is used like a black box and needs a single LU decomposition on a relatively small matrix, thus, requiring minimally $P+1$ deter-

ministic evaluations to approximate quantities of interest. However, it can be found to converge to the projected PCE coefficients under certain conditions. More generally, the solution acquired by this approach is not unique, since one can arbitrarily choose the random vectors for estimating the required deterministic solutions. Due to this fact, the present method should be interpreted as a model for evaluating the projected coefficients. It should be noted that the above expression does not require that the number of arbitrarily selected random vectors be equal to P; if this is not true, valid solutions can still be obtained by solving the system in the least squares sense.

In 2007, Loeven et al. [147] compared probabilistic collocation method with the Galerkin PCE method, the NIPC method and the SCM. Further, Li et al. [148] converted stochastic ODEs of a robust trajectory optimization problem into an equivalent deterministic ODEs by employing NIPC. Later on, Weise et al. [149] proposed NIPC for transcranial magnetic stimulation for analyzing the impact of randomness in the electrical conductivity of biological tissues on the induced electric field. Cooper et al. [150] applied NIPC and concluded that it is an effective tool for reducing simulation time costs when studying parametric roll, and potentially other ship dynamics phenomena. Moreover, Liang et al. [151] implemented NIPC together with nonlinearity reduction technique into the robust design method proposed by the EU project DESICOS, to determine the knock-down factor for an unstiffened CFRP cylindrical shell. Recently, Liu and Sun [152] combined NIPC with two scale fatigue damage model to predict the probabilistic fatigue life for high cycle.

3. **Non-Intrusive Spectral Projection (NISP)**

   In this approach, we perform multiple simulations on the collocation points and then estimate the coefficients of the PCE by applying special techniques. In order to obtain the coefficients, spectral projection is carried out, where the responses are projected onto the basis functions using orthogonal polynomials and the inner products.

   The advantage of this method is that we can use the legacy codes with varying parameters in order to obtain the statistics and spectral coefficients. It needs to be noticed that NISP not only provides sensitivity information but it also propagates the whole probabilistic representation of the model inputs to the model outputs. Further, according to the chosen order of the PCE, NISP also gives us information about the higher-order sensitivity. Moreover, higher-order effects are not put together into a single coefficient, but are studied individually and in terms of parameter-parameter interactions. The result is a complete PC representation of a complex system response behavior that can be used to generate PDFs of model output. It can be

easily applied to almost any deterministic codes. However, it needs implementation of sophisticated sampling methods in situations when the number of uncertain inputs increases such that it becomes computationally competitive with the intrusive spectral projection approach. The algorithm of NISP is as follows:

Algorithm 1.4 Non-intrusive Spectral Projection

1: Define the PDF of the random model parameters

2: Write down the corresponding PCE of each of the parameters.

$$\alpha = \sum_{k=0}^{P} \alpha_k \psi_k, \quad u(x) = \sum_{k=0}^{P} u_k(x)\psi_k$$

3: Bring out the samples of ξ, $\{\xi^j\}_{j=1}^{N}$ in accordance with the sampling scheme of interest.

4: Then evaluate $\alpha^j = \sum_{k=0}^{P} \alpha_k \psi_k(\xi^j)$ for each sample ξ^j such that $u^j = \mathfrak{M}(\alpha^j)$

5: Next, numerically compute the expectations for the Galerkin projection

$$\alpha_k = \frac{\langle \alpha \psi_k \rangle}{\langle \psi_k^2 \rangle}$$

$$u_k = \frac{\langle u \psi_k \rangle}{\langle \psi_k^2 \rangle} \quad \forall \quad k \in 0, 1, \ldots, P$$

6: Evaluate the above expectations over a large number of samples N and find polynomial coefficients u_k of the solution

$$u_k = \frac{\sum_{m=1}^{N} (u)_m (\psi_k)_m}{\sum_{m=1}^{N} (\psi_k^2)_m} \quad \forall \quad k \in 0, 1, \ldots, P$$

7: Assemble the PCE after computing u_k values,

$$u = \sum_{k=0}^{P} u_k \psi_k(\xi)$$

Reagana et al. implemented NISP scheme to propagate parametric uncertainty in reacting-flow simulations by using LHS to estimate the coefficients [85]. Later on, Alexanderian et al. [153] applied NISP scheme, on the basis of a Smolyak sparse quadrature grid, for deriving the PC representation of the stochastic response of selected quantities of interest. Mendes et al. [154] addressed the stochastic heat transfer problems using NISP approach.

1.3.3 Stochastic collocation Method

The idea behind SCM is that it requires only a collocation grid definition on which Lagrange interpolation functions are formed for known coefficients. It employs Gauss nodes and weights from the same optimal orthogonal polynomials as in gPC for carrying out collocation. This method was first given by Mathelin and Hussaini [155] in 2003. They [144] have shown a substantial decrease in computational time in comparison to the Galerkin PCE method for a quasi-1D nozzle flow.

The construction of this method is with reference to the polynomial interpolation in the multi dimensional space. The SCM results in a set of decoupled deterministic equations, making the method non-intrusive. For each Legendre collocation point, the problem is solved deterministically. The distribution function of the solution is reconstructed using polynomial interpolation. The mean and variance are obtained using Gauss-Legendre quadrature.

The first step is the projection of distribution function of the RV from $[0, 1]$ on the domain to $[-1, 1]$, which is called the ξ domain, by the linear transformation

$$F_\xi(\xi) = 2\mathcal{F}_\xi(\xi) - 1 \tag{1.3.22}$$

where $F_\xi(\xi)$ is the projection distribution function on the ξ domain $[-1, 1]$ and $\mathcal{F}_\xi(\xi)$ is the distribution function on $[0, 1]$. The ξ domain is considered as the stochastic domain which is defined according to the standard domain of orthogonal polynomials $[-1, 1]$. From that very ξ domain, N_p collocation points ξ_i are taken. For function approximation, Mathelin and Hussaini uses Gauss-Legendre quadrature points and Lagrange interpolating polynomials of order $N_p - 1$. So, the solution is represented as

$$u(x, t, \xi) = \sum_{i=1}^{N_p} u_i(x, t)\mathcal{L}_i(\xi) \tag{1.3.23}$$

where $\mathcal{L}_i(\xi)$ are the interpolating polynomials such that

$$\mathcal{L}_i(\xi_j) = \delta_{ij}, \quad 1 \leq i, j \leq N_p \tag{1.3.24}$$

Simply, Eq. (1.3.22) is applied to the model problem and then Eq. (1.3.23) is inserted. After that, Galerkin projection is applied with respect to $\mathcal{L}_k(\xi)$ and it is approximated using Gauss quadrature which involves quadrature weights corresponding to collocation points.

The algorithm of the SCM is as follows:

Algorithm 1.5 Stochastic Collocation Method

1: Select the polynomial in accordance with the PDF of the RV.
2: Evaluate the nodes and weights according to the Gauss-quadrature.
3: Treat nodes as the value of the RV.
4: Solve the deterministic system using a deterministic solver.
5: Compute the statistical descriptors using the weights of the quadrature.

This approach is quite promising when dealing with low dimensions in the input random fields and also the solution process is highly parallelizable. While choosing the collocation points, we have a number of collocation methods available in literature which include tensor product [156], Smolyak method [157], Stroud 2 or 3 cubature method [158], adaptive Stroud method [159].

1.3.3.1 Quantities of interest

The expectation of the solution by SCM is evaluated in the following way

$$\mathbb{E}[u] = \sum_{i=1}^{N_p} u_i w_i \tag{1.3.25}$$

where w_i are the weights of the quadrature. On the other hand, variance of the SCM is calculated by using the weights of the quadrature as follows:

$$\sigma^2(u)(t) = \mathbb{E}[u^2] - (\mathbb{E}[u])^2 = \sum_{i=1}^{N_p} (u_i^2 - \mathbb{E}[u])^2 w_i \tag{1.3.26}$$

1.3.3.2 Literature

In 2005, Xiu and Hesthaven [160] proposed high-order SCM approach with several feasible constructions. Moreover, numerical examples were executed in order to check the efficiency and accuracy of the method. In 2007, Baroth et al. [161] developed an efficient stochastic finite element method on the basis of Lagrange interpolating polynomials. Also, a comparison analysis is performed with other methods for the resolution of two nonlinear stochastic mechanical problems. In 2009, Bonnet et al. [162] applied SCM to numerically compute the electromagnetic field from a low number of empty cavity simulations. In 2010, Tang and Zhou [163] illustrated that the rate of convergence is dependent on the monotonicity of the solution which is governed by the random wave speed and the boundary and initial data.

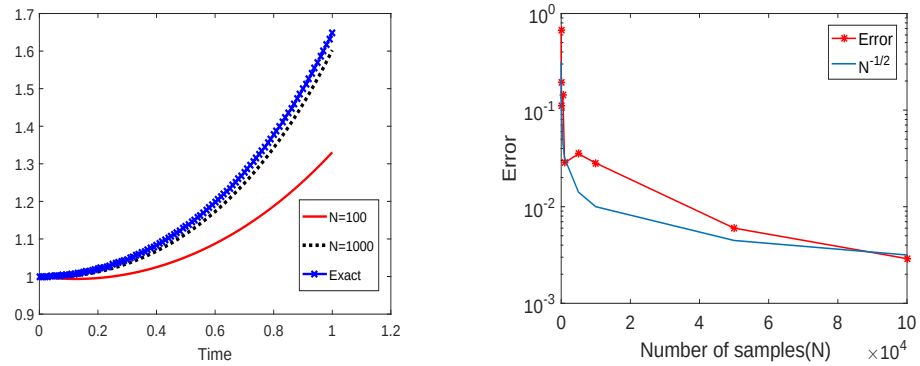
In the same year, Riahi et al. [164] proposed SCM by incorporating uncertain parameters in order to solve mixed mode fatigue crack growth problems. Moreover, Bressolette et al. [165] proposed SCM for large classes of mechanical problems, such as dynamic, static, and spectral problems.

In 2011, Fichtl and Prinja [166] applied SCM method to numerically evaluate the linear transport equation by considering the atomic density of the medium to be a continuous random function of position with log-normal probability distribution. Later on, in 2013, Motamed et al. [167] solved random second order wave equation with uncertain wave speed in a heterogeneous random medium by employing SCM. In 2015, DeGennaro et al. [168] applied PCEs based on SCM along with an application to airfoil icing. The standard probabilistic collocation method suffers from the curse of dimensionality. When there are more than one random parameters, it is difficult to compute even low order approximations within reasonable time. A problem can be solved by the concept of sparse grid method developed by Smolyak [169]. In situations when the number of RVs is slightly large, the best choice is to consider sparse grid tensor product spaces which can be further investigated in [160,170–172]. For incorporating uncertainties in hemodynamic simulations, a sparse grid SCM technique [173] has been employed for choosing collocation points and interpolates in the stochastic space and perform a carefully chosen set of cardiovascular simulations. In 2007, Ganapathysubramanian and Zabaras [174] employed sparse grid collocation method to effectively solve natural convection problems which consists of large stochastic dimensions. In 2014, He et al. [175] developed a sparse grid SCM for the reliability analysis of structures with random parameters and loads.

1.3.4 Comparison Analysis

In this section, we have compared MC, PCE and SCM. First of all, we will apply MC method to the test problem discussed earlier in section 1.2 by taking into account normally distributed α, i.e., $\alpha \backsim N(\mu, \sigma^2) \backsim N(0, 1)$. We have performed MC simulations by taking two different values of N as 100 and 1000 which can be seen in Fig 1.4a. It has been noticed from Fig 1.4a that the mean plots of $N = 1000$ matches with the exact solution whereas for $N = 100$, the solution is far from the exact solution. In Fig 1.4b, error of the mean curves has been displayed along with the curve $\sqrt{N}$. We analyze that as we increase the value of N, the error of the mean curve starts decreasing which is numerically expected.

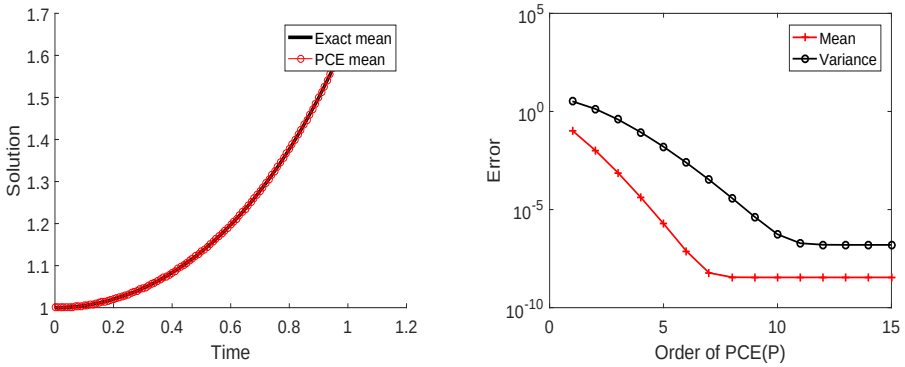
Now, we perform PCE simulations on the same problem by taking normally distributed α. Fig 1.5a shows the mean of the solution for polynomial order $P = 3$ and it has been noticed



(a) Mean at $t = 10$ (b) Error of the mean

Figure 1.4: Results of test problem using MC method with $dt = 0.01$.

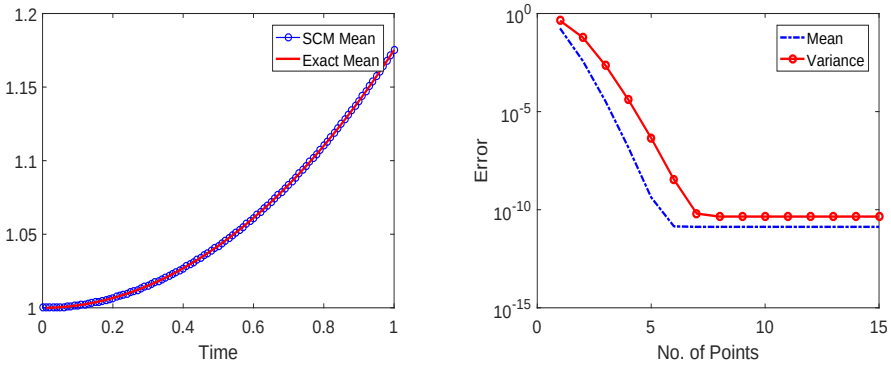
that PCE performs much better than the MC method. The mean by the PCE matches exactly with the exact mean at a very lower PCE order. We have also plotted the error of the mean and variance curves in Fig 1.5b and it has been analyzed that the error is reduced at a fast rate. It has been demonstrated that PCE exhibits considerable computational advantages over the traditional MC analysis and can be treated as a powerful tool in the field of UQ.



(a) Mean at $t = 1(P{=}3)$ (b) Error Mean and Variance

Figure 1.5: Results of test problem using PCE method with $dt = 0.01$.

Now, we apply SCM to the test problem with α to be uniformly distributed, i.e., $\alpha \backsim U(a, b) \backsim U(-1, 1)$. Fig 1.6a displays the mean by SCM by considering 15 points and it has been noticed that the mean by SCM agrees strongly by the exact mean. Also, in Fig 1.6b, the error of the mean and variance curves is highlighted and it has been analyzed that after 6 points, the error is coming out to be same for both mean and variance. It has been analyzed that MC method requires large number of samples to reach to the exact solution whereas PCE and SCM respectively requires much lower polynomial chaos order and nodes. It has been concluded that the error of mean and variance is most reduced by



(a) Mean at $t = 1(P{=}15)$ (b) Error Mean and Variance

Figure 1.6: Results of test problem using SCM with $dt = 0.01$.

the SCM.

1.4 Organisation of the thesis

The research work in this thesis aims at developing new techniques for solving stochastic PDEs numerically. The proposed work summarizes these approaches which are structured into six chapters. These are as follows:

Chapter 1 is the current chapter where we have given present state-of-the-art of the methods developed in propagating the uncertainties present in a system. The aim is to give the reader an exhaustive description of the stochastic approaches available for solving stochastic models in the field of UQ.

In Chapter 2, Hermite polynomial chaos is united with summation-by-parts (SBP)- simultaneous approximation terms (SATs) technique to solve the advection-diffusion equations with random Dirichlet boundary conditions (BCs). PCE with Hermite basis is employed to separate the randomness, then SBP operators are used to approximate the differential operators and SATs are used to enforce BCs by ensuring the stability. For each chaos coefficient, time integration is done using fourth order Runge-Kutta method (RK4). Statistical moments namely mean and variance are computed using PCE coefficients without any extra computational effort. The method is applied on three test problems for validation. The study of advection diffusion equation has relatively became an active research area in the field of UQ due to its numerous real life applications. The first two test problems are stochastic advection equations on $\mathbb{R}$ without any boundary and third problem is stochastic advection-diffusion equation on [0,2] with Dirichlet BCs. For the third problem, we have obtained a range of permissable parameters for a stable numerical solution.

In Chapter 3, we have developed a scheme combining gPC representation and B-spline wavelets. We have constructed a semi-orthogonal compactly supported B-spline wavelets for the bounded interval $[0, 1]$ which have been used for PCE of possible stochastic processes. To compute the deterministic coefficients of expansion, we have applied Galerkin projection on uncertain data and the solution variables. Then, to ascertain the behaviour of the random process, the system of equations obtained from projection has been integrated using RK4 method. To handle the nonlinearity, we have compared Galerkin projection with pseudo-spectral projection. From the three engineering problems, we conclude that Galerkin approximation performs better in comparison to pseudo-spectral approach which is numerically expected. Also, it has been observed that the wavelet function based expansion shows superior results as compared to scaling function based expansion.

In Chapter 4, we have studied the real life epidemic situations. As we know, they are modeled using system of DEs by considering deterministic parameters. However, in reality, the transmission parameters involved in such models experience a lot of variations and it is not possible to compute them exactly. In this chapter, we apply B-spline wavelet based gPC to analyze possible stochastic epidemic processes. Furthermore, SA has been performed to investigate the behavior of randomness in a simple epidemic model. It has been analyzed that linear B-spline wavelet basis shows accurate results by involving fewer PCE in comparison to cubic B-spline wavelets. The developed method has been employed on two real disease outbreaks, firstly, Influenza that affected British boarding school for boys in North England in 1978, and secondly, for the Ebola virus in Liberia in 2014. Real data from the British Medical Journal (Influenza) and World Health Organization (Ebola) has been incorporated into the Susceptible Infected Recovered (SIR) epidemic model. It has been observed that the numerical results obtained by the proposed method are quite satisfactory.

In Chapter 5, wavelet optimized finite difference B-spline polynomial chaos method is proposed for solving stochastic PDEs. The gPC is applied by considering linear B-spline wavelet basis. Then, stochastic Galerkin projection is executed for evaluating the deterministic coefficients of the gPC. In the next step, the system of equations is discretized by using Crank-Nicolson (CN) scheme for time integration and for approximating the differential operators, central finite difference (FD) matrices on irregular grids are considered. An adaptive grid is generated using the linear B-spline gPC for optimizing the numerical solution. The method is then tested on three problems namely heat equation with uncertain ICs, Burger's equation with uncertain ICs and Burger's equation with random viscosity as well as uncertain ICs. For the three test problems, grid modifications are displayed by taking periodic BCs into consideration. Mean and S.D. are plotted for each test prob-

lem. Moreover, for the third test problem, computational time comparison is performed by computing CPU time taken by the proposed method and the CPU time taken by the FD method on a uniform grid.

In Chapter 6, wavelet collocation approach has been proposed in collaboration with the gPC for numerically solving the stochastic PDEs. This approach is based on the concept of autocorrelation functions of compactly supported Daubechies scaling functions. First of all, we make use of linear B-spline basis in gPC. Then, in order to separate the randomness, Galerkin projection is executed on uncertain data and solution variables. After that, connection coefficients are calculated using Daubechies wavelets for approximating the differential operators. Moreover, fast algorithms are known to speed up the numerical scheme, so, we have executed a class of fast algorithms, on the basis of fast wavelet transform. Also, in order to handle the condition number, we have used a good diagonal preconditioning technique which makes the condition number of the matrices bounded. As we know, the grid on which the solution is computed is not stationary but is generated throughout the time integration process depending on the features of the numerical solution at the current time. Thus, we execute adaptive scheme which will focus on capturing the essential features of the solution by reducing the CPU time. Further, the method has been tested along with adaptivity on three stochastic PDEs with uncertain initial data and the results obtained from the proposed method are quite promising.

Chapter 2

Hybrid Hermite polynomial chaos SBP-SAT technique[1]

Advection is the transport of a substance or quantity by bulk motion of a fluid. The properties of that substance are carried with it. On the other hand, diffusion is the movement of a substance from an area of high concentration to a area of low concentration. Diffusion is a latin word "*diffundere*" which means to 'spread out'. For example, transport of ink dumped into a river. As the river flows, ink will move downstream in a pulse via advection, as the water's movement itself transports the ink. If added to a lake without significant bulk water flow, the ink would simply disperse outwards from its source in a diffusive manner, which is not advection. Note that as it moves downstream, the "pulse" of ink will also spread via diffusion. The sum of these processes is called convection.

The advection diffusion equation [176] is amongst the most widely used equation in applied sciences and engineering. It has been employed to describe the transport of pollutants in water and air, contaminant dispersion in shallow water [177], radon transport in soil, heat transfer in draining film [109], solute transport in a randomly heterogeneous aquifer [178], chemical (solute) transport in groundwater [179], flow in realistic discrete fracture network model [180], sediment transport [181], a jet diffusion flame in a combustor [182], tracer dispersion in porous media [183, 184], spatial rainfall intensity and speed of individual rain cells [185], fluid flow [186], pattern formation [187] and so on. In such flows, field parameters are often influenced by uncertainty due to lack of understanding of natural open channel properties including ICs and BCs, roughness coefficient and geometry.

SBP-SATs [188–194] is a numerical method which ensures accuracy, simplicity and stability. SBP operators simply estimate the derivatives involved in the initial boundary value problems (IBVP). They do not essentially impose BCs [190]. Therefore, most of SBP-based discretizations depend on SATs for imposing the BCs. SATs are the penalty terms that enforce boundary data weakly so as to achieve stability. The procedure begins with the continuous PDEs and the energy method is employed to ascertain data (forcing function,

[1]The content of this chapter is published as "Hybrid Hermite polynomial chaos SBP-SAT technique for stochastic advection-diffusion equations", *International Journal of Modern Physics C*, 31(9), 2020, 2050128 (33 pages), (**SCI: Impact Factor: 1.17**)

initial and BCs) so that the problem becomes well-posed. Further, this favors the SATs to be designed for the semi-discrete equations. To achieve stability [195] of the semi discrete form, additional conditions on the SATs are specified using energy method. SBP operators are generally one-dimensional [188–191] and can be employed to multidimensional problems using a Kronecker product formulation [192–194]. They are apt for smooth problems such as advection-diffusion equations.

M. Jardak et al. [196] proposed a new algorithm by combining Wiener-Hermite functionals and the Fourier collocation to solve the stochastic advection equation with uncertain transport velocity. N. Li et al. [197] used gPC representation for stochastic advection-diffusion with random diffusivity. M. El-Amrani et al. [198] solved the stochastic advection-diffusion equation using PCE together with the method of characteristics. M. Zahri and K. Al Madinah [199] solved two dimensional advection-diffusion equation with additive random excitations by using the method of lines. Many engineering problems of solid and fluid mechanics have employed widely used PC representation for quantifying the uncertainties, for example, flow in porous media [200], in mass spring and heat transfer problems [201] etc. Pettersson et al. [202] applied PCE approach on the Burger's equation with random boundary and ICs by employing SBP, artificial dissipation and SAT operators. Later on, Pettersson et al. [17] examined the usefulness of PCE and SBP-SAT operators to acquire energy estimates and stability for hyperbolic PDEs. Moreover, they applied PCE together with SBP-SAT technique on linearized symmetrized Navier-Stokes equations [203] in one dimension. Wahlsten and Nordström [204] constructed stable FD scheme of high-order on SBP-SAT form for stochastic non-linear viscous Burger's equation.

Till now, stability conditions have been acquired only for mixed BCs in literature and no one gave the stability conditions for Dirichlet BCs. We focus on acquiring these conditions for Dirichlet BCs in the third test problem. Thus, we are basically extending the existing method with novelty in the stability conditions for the Dirichlet BCs. In this chapter, we focus on presenting a stable numerical technique for random advection diffusion equation with Dirichlet BCs.

2.1 Hybrid Hermite polynomial chaos SBP-SAT technique (HPSBPSAT)

In this section, we will explain HPSBPSAT to solve the PDEs with randomness. The algorithm is as below:

Algorithm 2.1 HPSBPSAT

1: Hermite polynomials are used to approximate the unknown u in terms of unknown deterministic coefficients and stochastic basis functions, i.e., Hermite chaos polynomials.
2: The unknown deterministic coefficients are computed through Galerkin projection.
3: Further, SBP operators are used to approximate differential operators in space.
4: To impose BCs, the SBP operators are associated with SATs and the stability is imposed using energy estimate.
5: Then the time integration is performed using RK4 method.
6: Finally, in the end, we post-process the problem by estimating the mean and variance.

2.1.1 Hermite chaos

In this chapter, we have considered the Gaussian distributed inputs. According to Askey scheme which provides the relationship between various polynomial basis and probability distributions, Hermite polynomial basis are optimal basis with respect to the convergence for a Gaussian distributed input. The similarity between the weight function and the actual density for a Gaussian variable suggests that Hermite polynomials are easy to use when modeling Gaussian RVs. Therefore, we discuss the Hermite chaos briefly ahead. Considering the dimension of RV as $n = 1$ and therefore writing the PCE as

$$u(x,\xi) = \sum_{k=0}^{P} u_k \Psi_k(\xi). \tag{2.1.1}$$

where u_k are the coefficients, $\Psi_k(\xi)$ are the Hermite basis functions and ξ is the second order Gaussian RV. Now let us define inner product $\langle \cdot, \cdot \rangle$ and the norm $\| \cdot \|$ by $\langle u, v \rangle = \int_\Omega uvf(\xi)d\xi$ and $\|u\|^2 = \int_\Omega u^2 f(\xi)d\xi < \infty$, where $f(\xi)$ is the PDF. One dimensional Hermite polynomials of order m are given as follows:

$$\Psi_m(\xi) = H_m(\xi) = \sum_{k=0}^{\left[\frac{m}{2}\right]} \frac{(-1)^k m!}{2^k k!(m-2k)!} \xi^{m-2k}, \qquad m = 0, 1, \ldots,$$

where

$$\left[\frac{m}{2}\right] = \begin{cases} \frac{m}{2}, & m \text{ is even} \\ \frac{m-1}{2}, & m \text{ is odd} \end{cases}$$

Using the above relation, one can easily calculate $H_0(\xi) = 1, H_1(\xi) = \xi, H_2(\xi) = \xi^2 - 1, H_3(\xi) = \xi^3 - 3\xi, H_4(\xi) = \xi^4 - 6\xi^2 + 3$, and so on. Also, we can define Hermite polynomials as orthogonal polynomials in the following way

$$H_m(\xi) \;=\; (-1)^m e^{\xi^2/2} \frac{d^m}{d\xi^m} e^{-\xi^2/2}, \tag{2.1.2}$$

with weight function

$$\mathbf{w}(\xi) \;=\; \frac{1}{\sqrt{2\pi}} e^{-\xi^2/2}. \tag{2.1.3}$$

The inner product for the Hermite polynomial basis $H_m(\xi)$ of a Gaussian distributed RV ξ is defined as follows:

$$\langle H_i(\xi) H_j(\xi) \rangle = \int_{-\infty}^{\infty} H_i(\xi) H_j(\xi) \mathbf{w}(\xi) d\xi = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-\xi^2/2} H_i(\xi) H_j(\xi) d\xi = \delta_{ij} i!. \tag{2.1.4}$$

where δ_{ij} represents Kronecker delta. The above result can be verified by

$$\begin{aligned}
\langle H_1(\xi) H_2(\xi) \rangle &= \int_{-\infty}^{\infty} H_1(\xi) H_2(\xi) \mathbf{w}(\xi) d\xi, \\
&= \int_{-\infty}^{\infty} (-1)^1 e^{\xi^2/2} \frac{d}{d\xi} (e^{-\xi^2/2}) (-1)^2 e^{\xi^2/2} \frac{d^2}{d\xi^2} (e^{-\xi^2/2}) \frac{e^{-\xi^2/2}}{\sqrt{2\pi}} d\xi, \\
&= -\int_{-\infty}^{\infty} e^{\xi^2/2} e^{-\xi^2/2} \frac{\xi}{\sqrt{2\pi}} (-\xi^2 e^{-\xi^2/2} + e^{-\xi^2/2}) d\xi, \\
&= -\int_{-\infty}^{\infty} \frac{\xi}{\sqrt{2\pi}} (-\xi^2 e^{-\xi^2/2} + e^{-\xi^2/2}) d\xi, \\
&= -\int_{-\infty}^{\infty} \frac{\xi e^{-\xi^2/2}}{\sqrt{2\pi}} (-\xi^2 + 1) d\xi = 0. \\
\langle H_2(\xi) H_2(\xi) \rangle &= \int_{-\infty}^{\infty} H_2(\xi) H_2(\xi) \mathbf{w}(\xi) d\xi, \\
&= \int_{-\infty}^{\infty} \left((-1)^2 e^{\xi^2/2} \frac{d^2}{d\xi^2} (e^{-\xi^2/2}) \right)^2 \frac{e^{-\xi^2/2}}{\sqrt{2\pi}} d\xi, \\
&= \int_{-\infty}^{\infty} e^{\xi^2} (e^{-\xi^2/2} - \xi^2 e^{-\xi^2/2})^2 \frac{e^{-\xi^2/2}}{\sqrt{2\pi}} d\xi, \\
&= \int_{-\infty}^{\infty} \frac{(1-\xi^2)^2 e^{-\xi^2/2}}{\sqrt{2\pi}} d\xi = 2.
\end{aligned}$$

The triple product [205] of Hermite polynomials is defined as follows:

$$\langle H_p(\xi)H_q(\xi)H_r(\xi)\rangle = \begin{cases} 0, & \text{if } \ p+q+r \text{ is odd or } \max(p,q,r) > s \\ \frac{\sqrt{p!q!r!}}{(s-p)!(s-q)!(s-r)!}, & \text{otherwise.} \end{cases} \tag{2.1.5}$$

where $s = \frac{p+q+r}{2}$. Varying the values of p, q, r over $0, 1, 2$, we obtain the following values.

$$\begin{aligned} &\langle H_0(\xi)H_0(\xi)H_0(\xi)\rangle = 1, \quad \langle H_0(\xi)H_1(\xi)H_0(\xi)\rangle = 0, \quad \langle H_0(\xi)H_2(\xi)H_0(\xi)\rangle = 0, \\ &\langle H_1(\xi)H_0(\xi)H_0(\xi)\rangle = 0, \quad \langle H_1(\xi)H_1(\xi)H_0(\xi)\rangle = 1, \quad \langle H_1(\xi)H_2(\xi)H_0(\xi)\rangle = 0, \\ &\langle H_2(\xi)H_0(\xi)H_0(\xi)\rangle = 0, \quad \langle H_1(\xi)H_2(\xi)H_0(\xi)\rangle = 0, \quad \langle H_2(\xi)H_2(\xi)H_0(\xi)\rangle = 1. \end{aligned}$$

2.1.2 SBP-SAT operators

SBP operator is a discrete analogous of the continuous integration by parts (IBP) operator. In this section, we will explain IBP and SBP operators simultaneously. We move from continuous case to semi-discrete case using SBP operator and then we find the energy estimates for stability analysis in both the cases. First of all, we need to give expressions for the inner product and norm for continuous as well as discrete case.

Let $u, v \in L^2[a, b]$ be continuous real valued functions. The inner product and corresponding norm for the continuous case is given as

$$\begin{aligned} \langle u, v\rangle &= \int_a^b uv \ dx, \\ \|u\|^2 &= \langle u, u\rangle. \end{aligned} \tag{2.1.6}$$

Now, we will discretize the domain $a \leq x \leq b$ using $\mathcal{M}+1$ equidistant grid points,

$$x_i = a + (i-1)h, \quad i = 1, 2, ...\mathcal{M}, \quad h = \frac{b-a}{\mathcal{M}-1}.$$

The function u is numerically approximated at grid point x_i and is denoted by $\mathbf{u_i}$ and $\mathbf{u^T} = [\mathbf{u_0}, \mathbf{u_1}, \mathbf{u_2}, ...\mathbf{u}_\mathcal{M}]$ denotes discrete solution vector. Now, the inner product and norm for discrete real valued vector functions $u, v \in \mathbb{R}^n$ by

$$\begin{aligned} \langle u, v\rangle_\mathcal{P} &= u^T\mathcal{P}v, \\ \|u\|^2_\mathcal{P} &= u^T\mathcal{P}u. \end{aligned} \tag{2.1.7}$$

where $\mathcal{P} = \mathcal{P}^T > 0$ is a symmetric, positive definite $(\mathcal{M}+1) \times (\mathcal{M}+1)$ matrix [190]. After that, let us introduce some matrices and vectors which will be used later. For instance,

$$e_0 = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, e_{\mathcal{M}} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}, E_0 = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & \ddots & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & \cdots & 0 \end{bmatrix}, E_{\mathcal{M}} = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 0 & \ddots & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & \cdots & 1 \end{bmatrix}. \tag{2.1.8}$$

Now, we will give detailed explanation of how to use the SBP-SAT technique. Let us take a simple hyperbolic scalar equation

$$u_t + u_x = 0 \quad \text{(without considering BCs)} \tag{2.1.9}$$

As we know

$$\begin{aligned} \frac{d}{dt}\|u\|^2 &= \frac{d}{dt}\langle u, u\rangle, \\ \frac{d}{dt}\|u\|^2 &= \frac{d}{dt}\int_a^b u(t)u(t)\ dt, \\ \frac{d}{dt}\|u\|^2 &= \int_a^b uu_t\ dt + \int_a^b u_t u\ dt, \\ \frac{d}{dt}\|u\|^2 &= \langle u, u_t\rangle + \langle u_t, u\rangle. \end{aligned} \tag{2.1.10}$$

From Eq. (2.1.9) we get

$$u_t = -u_x. \tag{2.1.11}$$

Put Eq. (2.1.11) into Eq. (2.1.10)

$$\begin{aligned} \frac{d}{dt}\|u\|^2 =& \langle u, -u_x\rangle + \langle -u_x, u\rangle, \\ =& -\int_a^b uu_x dx - \int_a^b u_x u dx, \end{aligned}$$

$$
\begin{aligned}
&= u^2\Big|_a^b + \int_a^b u_x u dx - \int_a^b u_x u dx, \qquad (\because \text{IBP}) \\
&= u^2(x=b) - u^2(x=a).
\end{aligned}
$$

The next step is to move from continuous to semidiscrete approximation of the same problem by using SBP operator $\mathfrak{D}^{(1)} = \mathcal{P}^{-1}Q$ for the first derivative. $\mathfrak{D}^{(1)}$ is a SBP operator with $Q + Q^T = B$, where $B = diag(-1, ..., 0, 1)$. Using $\mathfrak{D}^{(1)}$, we have semi-discretized approximation of our problem as

$$
\mathbf{u_t} + \mathfrak{D}^{(1)}\mathbf{u} = 0. \tag{2.1.12}
$$

Using Eq. (2.1.12), we can find out the discrete analogous as follows

$$
\begin{aligned}
\frac{d}{dt}\|\mathbf{u}\|_{\mathcal{P}}^2 &= \langle \mathbf{u}, \mathbf{u_t}\rangle_{\mathcal{P}} + \langle \mathbf{u_t}, \mathbf{u}\rangle_{\mathcal{P}}, \\
&= \langle \mathbf{u}, -\mathcal{P}^{-1}Q\mathbf{u}\rangle_{\mathcal{P}} + \langle -\mathcal{P}^{-1}Q\mathbf{u}, \mathbf{u}\rangle_{\mathcal{P}}, \\
&= -\langle \mathbf{u}, \mathcal{P}^{-1}Q\mathbf{u}\rangle_{\mathcal{P}} - \langle \mathcal{P}^{-1}Q\mathbf{u}, \mathbf{u}\rangle_{\mathcal{P}}, \\
&= -\mathbf{u}^T\mathcal{P}\mathcal{P}^{-1}Q\mathbf{u} - (\mathcal{P}^{-1}Q\mathbf{u})^T\mathcal{P}\mathbf{u}, \\
&= -\mathbf{u}^TQ\mathbf{u} - \mathbf{u}^TQ^T(\mathcal{P}^{-1})^T\mathcal{P}\mathbf{u}, \\
&= -\mathbf{u}^TQ\mathbf{u} - \mathbf{u}^TQ^T(\mathcal{P}^{-1}\mathcal{P})^T\mathbf{u}, \quad (\because \mathcal{P} \text{ is a symmetric matrix}) \\
&= -\mathbf{u}^TQ\mathbf{u} - \mathbf{u}^TQ^T\mathbf{u}, \\
&= -\mathbf{u}^T(Q + Q^T)\mathbf{u}, \\
&= -\begin{bmatrix} \mathbf{u_0} & \mathbf{u_1} & \cdots & \mathbf{u_M} \end{bmatrix}
\begin{bmatrix} -1 & 0 & \cdots & \cdots & 0 \\ 0 & 0 & & & 0 \\ \vdots & & \ddots & & \vdots \\ \vdots & & & 0 & \vdots \\ 0 & \cdots & \cdots & 0 & 1 \end{bmatrix}
\begin{bmatrix} \mathbf{u_0} \\ \mathbf{u_1} \\ \vdots \\ \vdots \\ \mathbf{u_M} \end{bmatrix}, \\
&= \mathbf{u}_0^2 - \mathbf{u}_{\mathcal{M}}^2.
\end{aligned} \tag{2.1.13}
$$

In case a PDE contains a second order derivative ($\mathfrak{D}^{(2)}$), it is approximated in two ways.

1) One way is the utilization of first derivative approximation $\mathfrak{D}^{(1)}$ twice, and

2) Second way is to use $\mathfrak{D}^{(2)} = \mathcal{P}^{-1}(-M + B\mathscr{D})$ where $M + M^T \geq 0$, $\mathscr{D}$ includes

estimation of first derivative operator at the boundary in the following way

$$\mathscr{D} = \begin{bmatrix} d_1 & d_2 & d_3 & \cdots & \\ & 1 & & & \\ & & \ddots & & \\ & & & 1 & \\ & \cdots & -d_3 & -d_2 & -d_1 \end{bmatrix}, \tag{2.1.14}$$

Now, let us move forward to SAT for imposing BCs so that energy estimate is obtained and SBP property is preserved. We have different SAT approximations of a PDE for different BCs. Mattson et al. [190] explains the SBP-SAT of an advection-diffusion equation with Robin BCs. Below, we discuss the SAT procedure of a simple hyperbolic scalar equation with Dirichlet BC,

$$\begin{aligned} &\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} = 0, \quad 0 \le x \le 1, \qquad t \ge 0, \\ &u(0,t) = g_0(t). \end{aligned} \tag{2.1.15}$$

Applying IBP on Eq. (2.1.15), we get

$$\begin{aligned} \frac{d}{dt}\|u\|^2 &= -u^2(x=b) + u^2(x=a), \\ &= -u^2(x=1) + g_0^2. \end{aligned} \tag{2.1.16}$$

The discrete approximation of Eq. (2.1.15) using SBP-SAT technique for BCs is given by

$$\begin{aligned} \frac{\partial \mathbf{u}}{\partial t} + \mathfrak{D}^{(1)}\mathbf{u} &= -\mathcal{P}^{-1}\tau\{E_0\mathbf{u} - e_0 g_0(t)\}, \\ \frac{\partial \mathbf{u}}{\partial t} + \mathcal{P}^{-1}Q\mathbf{u} &= -\mathcal{P}^{-1}\tau\{E_0\mathbf{u} - e_0 g_0(t)\}, \\ \frac{\partial \mathbf{u}}{\partial t} &= -\mathcal{P}^{-1}Q\mathbf{u} - \mathcal{P}^{-1}\tau\{E_0\mathbf{u} - e_0 g_0(t)\}. \end{aligned} \tag{2.1.17}$$

where E_0 and e_0 are given in Eq. (2.1.8). Further,

$$\begin{aligned} \frac{\partial}{\partial t}\|\mathbf{u}\|_{\mathcal{P}}^2 &= \langle \mathbf{u}, \mathbf{u_t}\rangle_{\mathcal{P}} + \langle \mathbf{u_t}, \mathbf{u}\rangle_{\mathcal{P}}, \\ &= \big\langle \mathbf{u}, -\mathcal{P}^{-1}Q\mathbf{u} - \mathcal{P}^{-1}\tau\{E_0\mathbf{u} - e_0 g_0(t)\}\big\rangle_{\mathcal{P}} + \big\langle -\mathcal{P}^{-1}Q\mathbf{u} - \mathcal{P}^{-1}\tau\{E_0\mathbf{u} - e_0 g_0(t)\}, \mathbf{u}\big\rangle_{\mathcal{P}}, \\ &= -\big\langle \mathbf{u}, \mathcal{P}^{-1}Q\mathbf{u} + \mathcal{P}^{-1}\tau\{E_0\mathbf{u} - e_0 g_0(t)\}\big\rangle_{\mathcal{P}} \\ &- \big\langle \mathcal{P}^{-1}Q\mathbf{u} + \mathcal{P}^{-1}\tau\{E_0\mathbf{u} - e_0 g_0(t)\}, \mathbf{u}\big\rangle_{\mathcal{P}}, \end{aligned}$$

$$
\begin{aligned}
&= -\mathbf{u}^T\mathcal{P}\mathcal{P}^{-1}(Q\mathbf{u}+\tau E_0\mathbf{u}-\tau e_0 g_0(t))-(\mathcal{P}^{-1}Q\mathbf{u}+\mathcal{P}^{-1}\tau\{E_0\mathbf{u}-e_0g_0(t)\})^T\mathcal{P}\mathbf{u},\\
&= -\mathbf{u}^T(Q\mathbf{u}+\tau E_0\mathbf{u}-\tau e_0g_0(t))-(\mathcal{P}^{-1}Q\mathbf{u})^T\mathcal{P}\mathbf{u}-(\mathcal{P}^{-1}\tau E_0\mathbf{u})^T\mathcal{P}\mathbf{u}\\
&+ (\mathcal{P}^{-1}\tau e_0g_0(t))^T\mathcal{P}\mathbf{u},\\
&= -\mathbf{u}^TQ\mathbf{u}-\mathbf{u}^T\tau E_0\mathbf{u}+\mathbf{u}^T\tau e_0g_0(t)-\mathbf{u}^TQ^T(\mathcal{P}^{-1})^T\mathcal{P}\mathbf{u}-\mathbf{u}^T(E_0)^T\tau^T(\mathcal{P}^{-1})^T\mathcal{P}\mathbf{u}\\
&+ (g_0(t))^T(e_0)^T\tau^T(\mathcal{P}^{-1})^T\mathcal{P}\mathbf{u},\\
&= -\mathbf{u}^TQ\mathbf{u}-\mathbf{u}^T\tau E_0\mathbf{u}+\mathbf{u}^T\tau e_0g_0(t)-\mathbf{u}^TQ^T\mathbf{u}-\mathbf{u}^T(E_0)^T\tau^T\mathbf{u}+(g_0(t))^T(e_0)^T\tau^T\mathbf{u},\\
&= -\mathbf{u}^T(Q+Q^T)\mathbf{u}-\mathbf{u}^T\tau E_0\mathbf{u}+\mathbf{u}^T\tau e_0g_0(t)-\mathbf{u}^TE_0\tau\mathbf{u}\\
&+ (g_0(t))^T(e_0)^T\tau\mathbf{u}, \quad (\because (E_0)^T=E_0 \text{ and } \tau \text{ is scalar})\\
&= -\underbrace{\mathbf{u}^T(Q+Q^T)\mathbf{u}}_{(I)}-\underbrace{(\mathbf{u}^T\tau E_0\mathbf{u}+\mathbf{u}^TE_0\tau\mathbf{u})}_{(II)}+\underbrace{\mathbf{u}^T\tau e_0g_0(t)+(g_0(t))^T(e_0)^T\tau\mathbf{u}}_{(III)}. \quad (2.1.18)
\end{aligned}
$$

Now, we will solve part $(I),(II),(III)$ of Eq. (2.1.18) as follows:

$$
\begin{aligned}
\text{Part } I = -\mathbf{u}^T(Q+Q^T)\mathbf{u} &= -\begin{bmatrix}\mathbf{u_0} & \mathbf{u_1} & \dots & \mathbf{u}_{\mathcal{M}}\end{bmatrix}\begin{bmatrix}-1 & & & & \\ & 0 & & & \\ & & \ddots & & \\ & & & 0 & \\ & & & & 1\end{bmatrix}\begin{bmatrix}\mathbf{u_0}\\ \mathbf{u_1}\\ \vdots\\ \mathbf{u}_{\mathcal{M}}\end{bmatrix},\\
&= -\begin{bmatrix}\mathbf{u_0} & \mathbf{u_1} & \dots & \mathbf{u}_{\mathcal{M}}\end{bmatrix}\begin{bmatrix}-\mathbf{u_0}\\ 0\\ \vdots\\ 0\\ \mathbf{u}_{\mathcal{M}}\end{bmatrix},\\
&= -(-\mathbf{u_0}^2+\mathbf{u}_{\mathcal{M}}{}^2),\\
&= \mathbf{u_0}^2-\mathbf{u}_{\mathcal{M}}{}^2. \quad (2.1.19)
\end{aligned}
$$

$$
\begin{aligned}
\text{Part } II &= \mathbf{u}^T\tau E_0\mathbf{u}+\mathbf{u}^TE_0\tau\mathbf{u}, \quad (\text{Since } \tau \text{ is a scalar}, \therefore \tau E_0 = E_0\tau)\\
&= 2\mathbf{u}^T\tau E_0\mathbf{u},\\
&= 2\begin{bmatrix}\mathbf{u_0} & \mathbf{u_1} & \dots & \mathbf{u}_{\mathcal{M}}\end{bmatrix}\begin{bmatrix}\tau & 0 & \cdots & 0\\ 0 & 0 & & \vdots\\ \vdots & & \ddots & \vdots\\ 0 & \cdots & \cdots & 0\end{bmatrix}\begin{bmatrix}\mathbf{u_0}\\ \mathbf{u_1}\\ \vdots\\ \mathbf{u}_{\mathcal{M}}\end{bmatrix},
\end{aligned}
$$

$$= 2\begin{bmatrix}\mathbf{u_0} & \mathbf{u_1} & \dots & \dots & \mathbf{u}_{\mathbb{M}}\end{bmatrix}\begin{bmatrix}\tau\mathbf{u_0}\\ 0\\ \vdots\\ 0\end{bmatrix},$$

$$= 2\tau\mathbf{u_0^2}. \tag{2.1.20}$$

$$\begin{aligned}
\text{Part } III &= \mathbf{u}^T\tau e_0 g_0(t) + (g_0(t))^T(e_0)^T\tau\mathbf{u},\\
&= \mathbf{u}^T\tau e_0 g_0(t) + g_0(t)(e_0)^T\tau\mathbf{u},\\
&= \begin{bmatrix}\mathbf{u_0} & \mathbf{u_1} & \dots & \mathbf{u}_{\mathbb{M}}\end{bmatrix}\tau\begin{bmatrix}1\\0\\\vdots\\0\end{bmatrix}g_0(t) + g_0(t)\begin{bmatrix}1 & 0 & \dots & 0\end{bmatrix}\tau\begin{bmatrix}\mathbf{u_0}\\\mathbf{u_1}\\\vdots\\\mathbf{u}_{\mathbb{M}}\end{bmatrix},\\
&= \begin{bmatrix}\mathbf{u_0} & \mathbf{u_1} & \dots & \mathbf{u}_{\mathbb{M}}\end{bmatrix}\begin{bmatrix}\tau g_0(t)\\0\\\vdots\\0\end{bmatrix} + \begin{bmatrix}\tau g_0(t) & 0 & \dots & 0\end{bmatrix}\begin{bmatrix}\mathbf{u_0}\\\mathbf{u_1}\\\vdots\\\mathbf{u}_{\mathbb{M}}\end{bmatrix},\\
&= \mathbf{u_0}\tau g_0(t) + \tau\mathbf{u_0}g_0(t),\\
&= 2\mathbf{u_0}\tau g_0(t).
\end{aligned} \tag{2.1.21}$$

Therefore, Eq. (2.1.19), (2.1.20), (2.1.21) are substituted in Eq. (2.1.18) to get

$$\begin{aligned}
\frac{\partial}{\partial t}\|\mathbf{u}\|_{\mathbb{P}}^2 &= (\mathbf{u_0^2} - \mathbf{u}_{\mathbb{M}}{}^{\mathbf{2}}) - 2\tau\mathbf{u_0^2} + 2\mathbf{u_0}\tau g_0(t),\\
&= \frac{\tau^2}{2\tau-1}g_0(t)^2 - \mathbf{u}_{\mathbb{M}}^2 - (2\tau-1)\left(\mathbf{u}_0 - \frac{\tau}{2\tau-1}g_0(t)\right)^2.
\end{aligned} \tag{2.1.22}$$

After solving we have acquired the energy method in Eq. (2.1.22) of our hyperbolic problem. For the stability, we need $\tau > \frac{1}{2}$ in expression (2.1.22) as per the following theorem.

Theorem : Consider the IBVP [17]

$$\begin{aligned}
u_t + \mathcal{Q}(x,t,\partial_x)u = F(x,t), \quad 0 \le x \le 1, \quad t \ge 0,\\
u(x,0) = f(x),\\
L_0(t,\partial_x)u(0,t) = g_0(t),\\
L_1(t,\partial_x)u(1,t) = g_1(t),
\end{aligned} \tag{2.1.23}$$

where $u = (u_0, ..., u_{\mathcal{M}})^T$ is the solution vector and $\mathcal{Q}$ is a differential operator. Further, BCs are defined by differential operators $\mathcal{L}_0$ and $\mathcal{L}_1$. Moreover, $g_0(t), g_1(t)$ gives the boundary data of the problem, $f(x)$ gives the initial data, and $F(x,t)$ denotes forcing function.
We now formulate the semi-discrete approximation of Eq. (2.1.23) by

$$\begin{aligned} \mathbf{u}_t + \tilde{\mathcal{Q}}(x,t)\mathbf{u} &= \mathbf{F} + \mathbf{S}, \\ \mathbf{u}(0) &= \mathbf{f}, \end{aligned}$$

where $\tilde{\mathcal{Q}}$ represents the discrete approximation of $\mathcal{Q}$. $\mathbf{S} = \mathbf{S}(g_0, g_1)$ is the SAT which enforce BCs weakly. Now, assume BCs to be homogenous, i.e., $\mathbf{g_0} = \mathbf{g_1} = 0$. Let $\mathbf{f}$ be the projection of C^∞ function that vanishes at boundaries. The semi-discretization is called stable if for all $h \leq h_0$,

$$\|\mathbf{u}\|_h \leq K e^{\alpha_d t} \|\mathbf{f}\|_h. \tag{2.1.24}$$

holds and K, α_d, h_0 are constants that do not depend on $\mathbf{f}$. Till now, SBP-SAT operators were dealing with a single equation. When a problem involves system of equations, SBP-SAT technique is applied as multiblock using Kronecker product. Let $\mathcal{K}$ be a $\mathfrak{m} \times \mathfrak{n}$ matrix and $\mathcal{L}$ be a $\mathfrak{p} \times \mathfrak{q}$ matrix, then we define the Kronecker product $\mathcal{K} \otimes \mathcal{L}$ as

$$\mathcal{K} \otimes \mathcal{L} = \begin{pmatrix} [\mathcal{K}]_{1,1}\mathcal{L} & \cdots & [\mathcal{K}]_{1,\mathfrak{n}}\mathcal{L} \\ \vdots & \ddots & \vdots \\ [\mathcal{K}]_{\mathfrak{m},1}\mathcal{L} & \cdots & [\mathcal{K}]_{\mathfrak{m},\mathfrak{n}}\mathcal{L} \end{pmatrix}. \tag{2.1.25}$$

Kronecker product consists of very interesting properties such as

1) $(\mathcal{K} \otimes \mathcal{L})(\mathfrak{M} \otimes \mathcal{N}) = (\mathcal{K}\mathfrak{M} \otimes \mathcal{L}\mathcal{N})$, where $\mathcal{K}$ and $\mathfrak{M}$ have same dimension and so does $\mathcal{L}$ and $\mathcal{N}$,

2) $(\mathcal{K} \otimes \mathcal{L})^T = (\mathcal{K}^T \otimes \mathcal{L}^T)$,

3) $(\mathcal{K} \otimes \mathcal{L})^{-1} = (\mathcal{K}^{-1} \otimes \mathcal{L}^{-1})$ only if $\mathcal{K}$ and $\mathcal{L}$ are invertible.

Having obtained the parameters, we have performed time integration using the RK4 method. After time integration, we obtain our solution in the form

$$u(x,t,\xi) = \sum_{k=0}^{P} u_k(x,t)\Psi_k(\xi). \tag{2.1.26}$$

2.1.3 Post Processing

Without depending on the selection of basis $\{\Psi\}_{i=0}^{\infty}$, mean and variance of $u(x,t,\xi)$ can be described by using the coefficients of the PCE as

$$\begin{aligned} &\mathbb{E}(u(x,t,\xi)) = u_0(x,t), \\ &\sigma^2(u(x,t,\xi)) = \sum_{i=1}^{\infty} u_i^2(x,t). \end{aligned} \tag{2.1.27}$$

The detailed explanation of the above formulas has been given in subsubsection (1.3.2.6). One can straightforwardly calculate the statistics of the uncertain output with polynomial series that are orthogonal with respect to their PDF. From Eq. (2.1.27), we can note that the coefficients of PCE are only required for calculating the statistical descriptors of the uncertain output, which does not require additional computational cost. One can also estimate the higher order moments straightforwardly with the coefficients of the expansion by deriving from its analytical formula.

After evaluating the inner product exactly, the mean of the output is not affected by the truncation; it does however involve aliasing errors when the inner products are approximated by quadrature rules. The series which is truncated underrates the variance whereas in case of small magnitude of the higher order terms, the error stays little. Hence, variance is good at indicating whether the series includes sufficient terms or not.

2.2 Numerical Results

In this section, we have applied HPSBPSAT on the advection-diffusion and simple advection equations.

2.2.1 Stochastic Advection equation-I

Let us consider the motion of a scalar u advected by an uncertain velocity field along with an IC which is as follows [206]:

$$\begin{aligned} &\frac{\partial u}{\partial t} + v(x,t,\xi)\frac{\partial u}{\partial x} = 0, \qquad t \geq 0, \qquad x \in \mathbb{R} \\ &u(x,0) = sin(x). \end{aligned} \tag{2.2.1}$$

Here, we will take the randomness in the advection velocity $v(\xi)$ by considering it as lognormal RV with $\xi \backsim N(2, 0.5)$. Therefore,

$$v(\xi) = c_1 + c_2 e^{\xi}.$$

The analytical solution of the above advection problem is

$$u(x, t, \xi) = -sin(v(\xi)t - x).$$

Now, we apply HPSBPSAT on our problem. First of all, we write the PCEs of $u(x, t, \xi)$ and $v(x, t, \xi)$ as

$$u(x, t, \xi) = \sum_{i=0}^{P} u_i(x, t)\Psi_i(\xi), \qquad v(x, t, \xi) \ = \sum_{i=0}^{P} v_i \Psi_i(\xi), \tag{2.2.2}$$

Then we insert these PCEs in Eq. (2.2.1) as

$$\frac{\partial}{\partial t}\left(\sum_{i=0}^{P} u_i(x, t)\Psi_i(\xi)\right) + \sum_{j=0}^{P} v_j \Psi_j(\xi) \frac{\partial}{\partial x}\left(\sum_{i=0}^{P} u_i(x, t)\Psi_i(\xi)\right) = 0.$$

Further, Galerkin projection is applied by taking inner product with respect to $\Psi_k(\xi)$. Therefore, we get

$$\sum_{i=0}^{P} \frac{\partial u_i(x, t)}{\partial t} \langle \Psi_i(\xi)\Psi_k(\xi)\rangle + \sum_{i=0}^{P}\sum_{j=0}^{P} v_j \frac{\partial u_i(x, t)}{\partial x} \langle \Psi_j(\xi)\Psi_i(\xi)\Psi_k(\xi)\rangle = 0,$$

$$\frac{\partial u_k(x, t)}{\partial t} + \sum_{i=0}^{P}\sum_{j=0}^{P} v_j \frac{\partial u_i(x, t)}{\partial x} \frac{\langle \Psi_j(\xi)\Psi_i(\xi)\Psi_k(\xi)\rangle}{\langle \Psi_k^2(\xi)\rangle} = 0, \quad \text{where} \quad k = 0, 1, \ldots, P.$$

$$\frac{\partial u_k(x, t)}{\partial t} + \sum_{i=0}^{P}\sum_{j=0}^{P} v_j \frac{\partial u_i(x, t)}{\partial x} \frac{\langle \Psi_j(\xi)\Psi_i(\xi)\Psi_k(\xi)\rangle}{k!} = 0, \quad \text{where} \quad k = 0, 1, \ldots, P.$$

Taking $u = [u_0, u_1, \ldots, u_P]$ as the vector of gPC coefficients. We have,

$$\frac{\partial u^P(x, t)}{\partial t} + V \frac{\partial u^P(x, t)}{\partial x} = 0, \tag{2.2.3}$$

Time step size(dt)	RK4	CPU time (sec)	Crank Nicolson	CPU time(sec)
0.01	0.0162	3.208206	4.5638e-05	4.908624
0.005	0.0080	4.261247	4.1350e-05	7.548784
0.001	0.0016	12.491318	4.1072e-05	28.868491
0.0005	7.8002e-04	22.630850	4.1072e-05	55.600763
0.0001	1.3788e-04	104.937106	4.1073e-05	289.117093

Table 2.1: l_2 norm of the error of the mean curves of Test problem 1 with $c_1 = 1$ and $c_2 = 0.01$ along with CPU time for RK4 and CN scheme at different time step sizes and fixed m = 250.

where

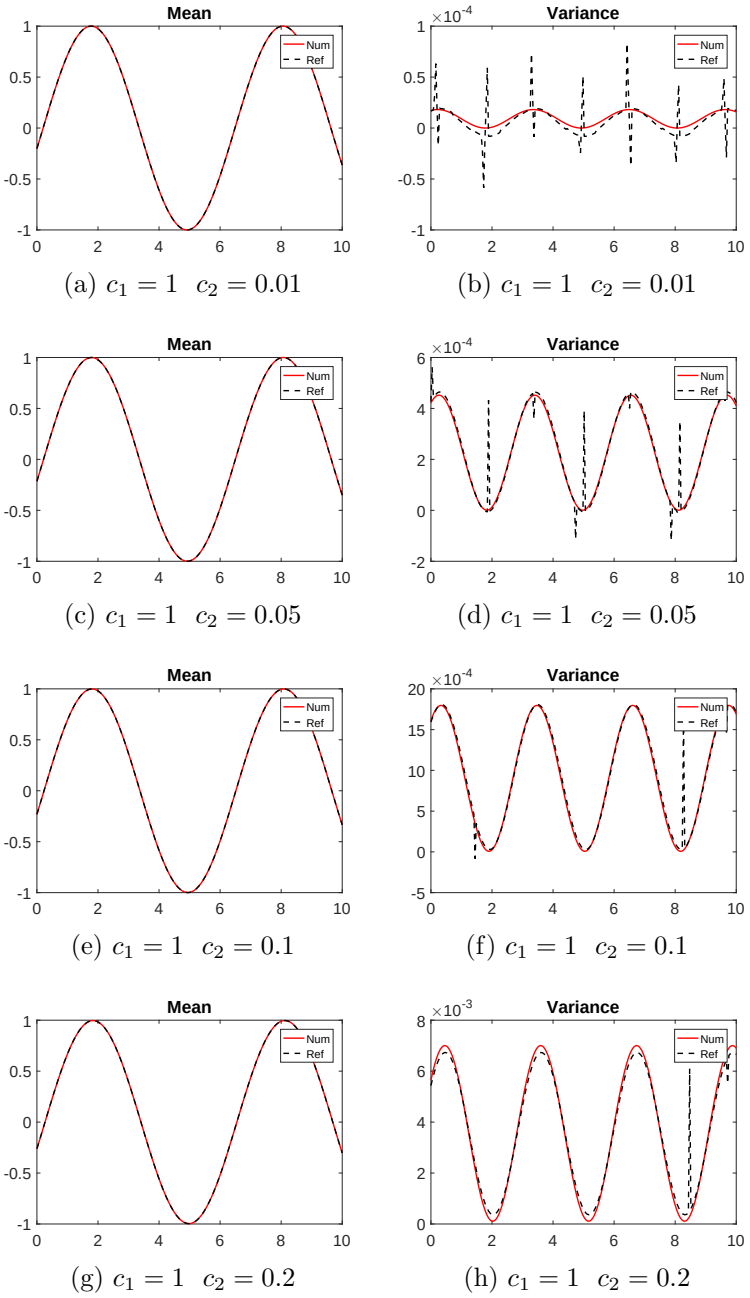
$$V = \sum_{k=0}^{P} v_j \frac{\langle \Psi_j(\xi)\Psi_i(\xi)\Psi_k(\xi)\rangle}{k!} = 0, \quad \text{where} \quad i, j = 0, 1, \ldots, P.$$

Since, we are considering the problem without BCs, therefore, we will only use the SBP technique as follows:

$$\frac{\partial u}{\partial t} + (\mathcal{P}^{-1}Q \otimes V)u = 0. \tag{2.2.4}$$

We have applied SBP operators of order 6 in our simulations. After applying RK4 method for the time integration, the last step of HPSBPSAT algorithm is post-processing the solution. The graphs of mean and variance are plotted to know the uncertainty characteristics. In this problem, we have taken distinct values of c_1 and c_2. Fig. 2.1 and Fig. 2.2 shows the expectation and variance for different values of c_1 and c_2 at time $t = 0.2$ and $dt = 0.0001$. The space is discretized into m = 250 grid points and the order of the PCE is taken as $P = 4$ in our simulations. In all the plots, we notice that our numerical results of the mean agree perfectly well with the mean of the reference solution whereas the variance decreases as we decrease the values of the parameter c_2 from 0.2 to 0.01. In Fig. 2.2, we observe that when c_2 is small ($c_2 = 0.01$), variance is of the range 10^{-5} whereas when $c_2 = 0.5$, the variance of the range 10^{-2}, i.e., variance is increasing as we increase the value of c_2. This is theoratically expected, as with increase in randomness the variance is supposed to be increased. Also, in Fig. 2.2 we can see that if we take $c_2 = 0$, i.e., there is no randomness, thus, the variance is zero which is theoretically expected.

The main advantages of RK4 method used for time integration is that it is easy to implement and is a stable explicit scheme. All of this was recognized a long time ago by researchers, and the classical Runge-Kutta method was suggested in 1895. RK4 is 4th order convergent, with an error $O(h^4)$, and can be used for accurate computations. In table 2.1, we have performed comparison between explicit RK4 method and the implicit



(a) $c_1 = 1 \quad c_2 = 0.01$ (b) $c_1 = 1 \quad c_2 = 0.01$

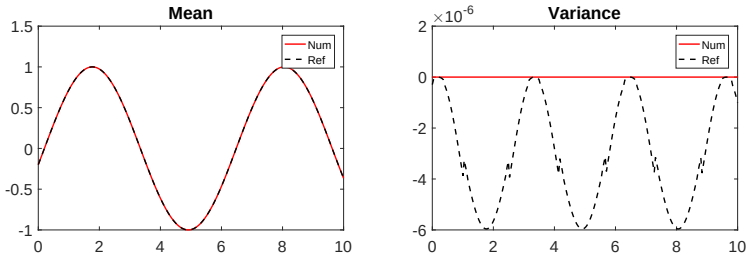
(c) $c_1 = 1 \quad c_2 = 0.05$ (d) $c_1 = 1 \quad c_2 = 0.05$

(e) $c_1 = 1 \quad c_2 = 0.1$ (f) $c_1 = 1 \quad c_2 = 0.1$

(g) $c_1 = 1 \quad c_2 = 0.2$ (h) $c_1 = 1 \quad c_2 = 0.2$

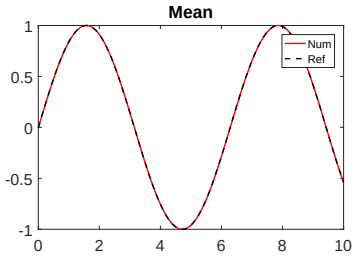
Figure 2.1: Mean (Left) and Variance (Right) of the solution of Test problem 1 at $t = 0.2$ for different values of c_1 and c_2.

CN scheme by computing the l_2 norm of the error of the mean curves of test problem 1 at different time step sizes $dt = 0.01, 0.005, 0.001, 0.0005, 0.0001$ and it has been analyzed that computational cost of CN scheme is high when compared with the RK4. As we decrease the time step size, it has been noticed that RK4 takes less CPU time in comparison to CN scheme. Implicit methods are recognized as better tools for time integration in order to have high accuracy and convergence but they lag behind when considering the

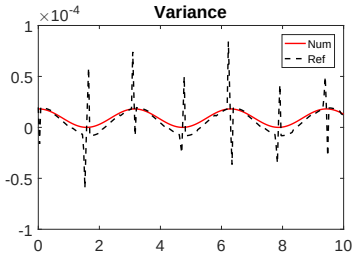


(a) $c_1 = 1 \quad c_2 = 0$

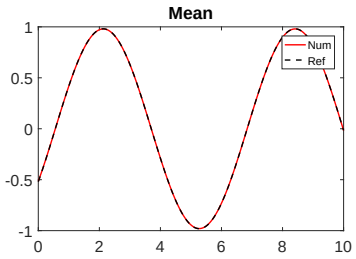
(b) $c_1 = 1 \quad c_2 = 0.1$



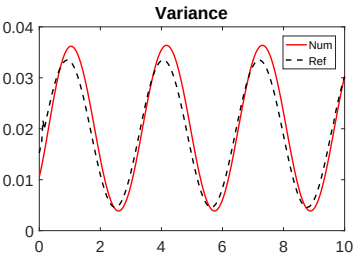
(c) $c_1 = 0 \quad c_2 = 0.01$



(d) $c_1 = 0 \quad c_2 = 0.01$



(e) $c_1 = 2 \quad c_2 = 0.5$



(f) $c_1 = 2 \quad c_2 = 0.5$

Figure 2.2: Mean (Left) and Variance(Right) of Test Problem 1 at $t = 0.2$.

computational effort. Keeping computational cost in mind, RK4 has been applied for time integration as it gives promising results at a much lower computational cost.

2.2.2 Stochastic Advection equation-II

Again, we are considering the same stochastic advection equation [206]

$$\frac{\partial u}{\partial t} + v(x,t,\xi)\frac{\partial u}{\partial x} = 0, \qquad t \geq 0, \qquad x \in \mathbb{R} \tag{2.2.5}$$

but with different IC

$$u(x,0) = \exp(-200(x-0.25)^2).$$

Here, advection velocity $v(\xi)$ is taken to be uncertain by considering it as Gaussian RV

with lognormal distribution $N(2, 0.5)$. Therefore,

$$v(\xi) = c_1 + c_2 e^{\xi}.$$

The analytical solution of the above problem comes out to be

$$u(x, t, \xi) = \exp(-200(v(\xi)t - x + 0.25)^2).$$

According to HPSBPSAT, the PCE of Eq.(2.2.5) is same as

$$\frac{\partial u_k(x,t)}{\partial t} + \sum_{i=0}^{P}\sum_{j=0}^{P} v_j \frac{\partial u_i(x,t)}{\partial x}\frac{\langle \Psi_j(\xi)\Psi_i(\xi)\Psi_k(\xi)\rangle}{k!} = 0, \quad \text{where} \quad k = 0, 1, \ldots, P.$$

Taking $u = [u_0, u_1, \ldots, u_P]$ as the vector of gPC coefficients. We have,

$$\frac{\partial u^P(x,t)}{\partial t} + V\frac{\partial^P(x,t)}{\partial x} = 0, \tag{2.2.6}$$

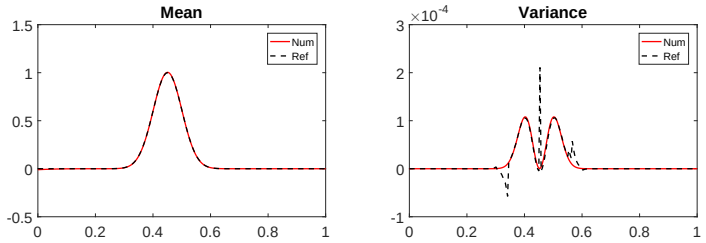
where

$$V = \sum_{k=0}^{P} v_j(x,t)\frac{\langle \Psi_j(\xi)\Psi_i(\xi)\Psi_k(\xi)\rangle}{k!} = 0, \quad \text{where} \quad i, j = 0, 1, \ldots, P.$$

Again, we will apply only SBP operators as Eq. (2.2.5) consists of no boundary, i.e.,

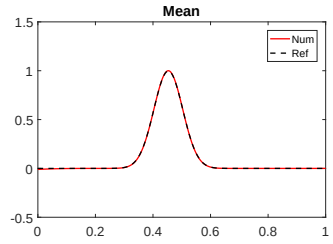
$$\frac{\partial u}{\partial t} + (\mathcal{P}^{-1}Q \otimes V)u = 0. \tag{2.2.7}$$

Mean and variance plots of the problem are shown in Fig. 2.3. Here, we have fixed the value of $c_1 = 1$ and have chosen different values of the parameter c_2 as $0.002, 0.01, 0.05$ and 0.5. As we increase the value of the parameter c_2, the mean of our problem resembles exactly with the mean of the reference solution. On the other hand, the variance increases as we increase the value of the parameter c_2 whereas we notice that the number of peaks in variance plots in Fig. 2.3 are increasing slightly with the increase in the value of c_2. Also, variance of the reference solution is showing sharp peaks whereas we have smooth curves of variance of the numerical solution. In Fig. 2.3, we observe that when c_2 is small ($c_2 = 0.002$), variance is of the range 10^{-4} whereas when $c_2 = 0.5$, the variance of the range 0.1, i.e., variance is increasing as we increase the value of c_2. This is theoretically expected, as with increase in randomness the variance is supposed to increase. Also, in Fig. 2.3 we can see that variance is zero as there is no randomness when we take $c_2 = 0$.

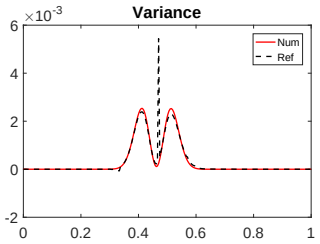


(a) $c_2 = 0.002$

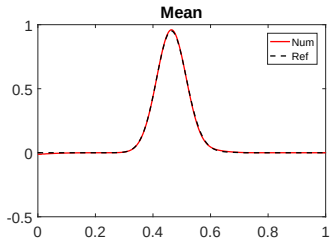
(b) $c_2 = 0.002$



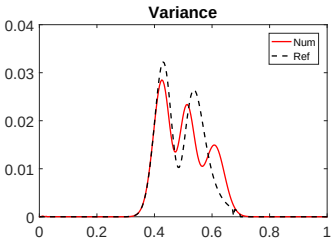
(c) $c_2 = 0.01$



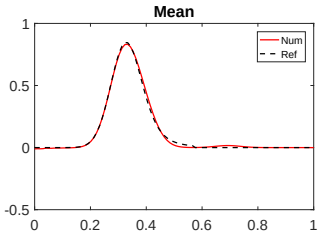
(d) $c_2 = 0.01$



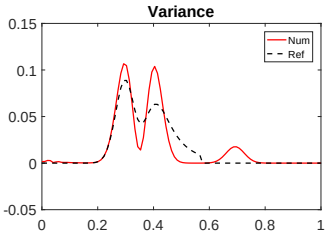
(e) $c_2 = 0.05$



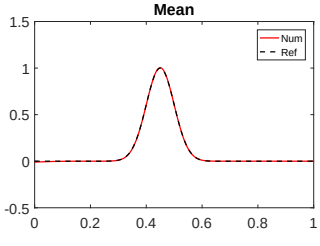
(f) $c_2 = 0.05$



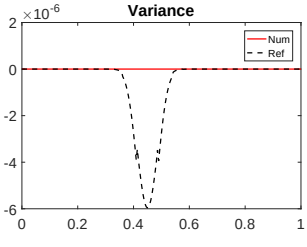
(g) $c_2 = 0.5$



(h) $c_2 = 0.5$



(i) $c_2 = 0$



(j) $c_2 = 0$

Figure 2.3: Mean (Left) and Variance(Right) of Test problem 2 at $t = 0.2$ with different values of parameter c_2 and $c_1 = 1$.

2.2.3 Advection-Diffusion equation

We now consider one dimensional advection-diffusion equation [198] in space domain $\mathcal{D} = [0, 2]$ and time domain T=(0,t]

$$\frac{\partial u}{\partial t} + v(x,t,\xi)\frac{\partial u}{\partial x} - \mu(\xi)\frac{\partial^2 u}{\partial x^2} = 0 \quad \text{in } \mathcal{D} \times T \times \Omega. \tag{2.2.8}$$

where $\mu(\xi)$ is the diffusion coefficient, $v(x,t,\xi)$ denotes velocity field and $u(x,t,\xi)$ is the required solution. The deterministic IC of the problem is

$$u(x,0,\xi) = u^0(x,\xi) = \begin{cases} 1, & x \in [a,b] \\ 0 & \text{otherwise} \end{cases}$$

and Dirichlet BCs are taken to be homogeneous, i.e.,

$$u(x=0,t,\xi) = 0, \quad u(x=2,t,\xi) = 0.$$

The velocity field and the viscosity coefficient are taken to be stochastic with lognormal distribution as follows:

$$v(x,t,\xi) = c_1 + c_2 e^{\xi},$$
$$\mu(\xi) = c_3 + c_4 e^{\xi}.$$

The analytical solution of the above stochastic advection-diffusion equation is given by

$$\bar{u}(x,t,\xi) = \frac{1}{2}\left(\text{erf}\left(\frac{x - v(\xi)t - a}{\sqrt{4\mu(\xi)t}}\right) - \text{erf}\left(\frac{x - v(\xi)t - b}{\sqrt{4\mu(\xi)t}}\right)\right),$$

where $\text{erf}(x)$ is the error function given by

$$\text{erf}(x) = \frac{1}{\sqrt{\pi}}\int_0^x e^{-s^2} ds. \tag{2.2.9}$$

According to HPSBPSAT, we insert the following PCEs of $u(x,t,\xi)$, $v(\xi)$, $\mu(\xi)$ in Eq. (6.2.8) as we did earlier

$$u(x,t,\xi) = \sum_{i=0}^{P} u_i(x,t)\Psi_i(\xi),$$
$$v(x,t,\xi) = \sum_{i=0}^{P} v_i\Psi_i(\xi),$$

$$\mu(\xi) = \sum_{i=0}^{P} \mu_i \Psi_i(\xi). \tag{2.2.10}$$

After inserting PCEs in the problem, we get

$$\frac{\partial}{\partial t}\left(\sum_{i=0}^{P} u_i(x,t)\Psi_i(\xi)\right) + \sum_{j=0}^{P} v_j \Psi_j(\xi) \frac{\partial}{\partial x}\left(\sum_{i=0}^{P} u_i(x,t)\Psi_i(\xi)\right) - \sum_{j=0}^{P} \mu_j \Psi_j(\xi) \frac{\partial^2}{\partial x^2}\left(\sum_{i=0}^{P} u_i(x,t)\Psi_i(\xi)\right) = 0. \tag{2.2.11}$$

Further, Galerkin projection is applied by taking inner product with respect to $\Psi_k(\xi)$ as follows:

$$\begin{aligned}
\sum_{i=0}^{P} \frac{\partial u_i(x,t)}{\partial t}\langle \Psi_i(\xi)\Psi_k(\xi)\rangle + \sum_{i=0}^{P}\sum_{j=0}^{P} v_j(x,t)\frac{\partial u_i(x,t)}{\partial x}\langle \Psi_j(\xi)\Psi_i(\xi)\Psi_k(\xi)\rangle & \\
- \sum_{i=0}^{P}\sum_{j=0}^{P} \mu_j \frac{\partial^2 u_i(x,t)}{\partial x^2}\langle \Psi_j(\xi)\Psi_i(\xi)\Psi_k(\xi)\rangle &= 0, \\
\frac{\partial u_k(x,t)}{\partial t} + \sum_{i=0}^{P}\sum_{j=0}^{P} v_j \frac{\partial u_i(x,t)}{\partial x}\frac{\langle \Psi_j(\xi)\Psi_i(\xi)\Psi_k(\xi)\rangle}{\langle \Psi_k^2(\xi)\rangle} & \\
- \sum_{i=0}^{P}\sum_{j=0}^{P} \mu_j \frac{\partial^2 u_i(x,t)}{\partial x^2}\frac{\langle \Psi_j(\xi)\Psi_i(\xi)\Psi_k(\xi)\rangle}{\langle \Psi_k^2(\xi)\rangle} &= 0, \\
\frac{\partial u_k(x,t)}{\partial t} + \sum_{i=0}^{P}\sum_{j=0}^{P} v_j \frac{\partial u_i(x,t)}{\partial x}\frac{\langle \Psi_j(\xi)\Psi_i(\xi)\Psi_k(\xi)\rangle}{k!} & \\
- \sum_{i=0}^{P}\sum_{j=0}^{P} \mu_j \frac{\partial^2 u_i(x,t)}{\partial x^2}\frac{\langle \Psi_j(\xi)\Psi_i(\xi)\Psi_k(\xi)\rangle}{k!} &= 0,
\end{aligned}$$

where $k = 0, 1, \ldots, P$. Now, insert the PCEs in the same way into the IC

$$\sum_{i=0}^{P} u_i(x,0)\Psi_i(\xi) = \sum_{i=0}^{P} u_i^0(x)\Psi_i(\xi).$$

Taking inner product with respect to $\Psi_k(\xi)$

$$\begin{aligned}
\sum_{i=0}^{P} u_i(x,0)\langle \Psi_i(\xi)\Psi_k(\xi)\rangle &= \sum_{i=0}^{P} u_i^0(x)\langle \Psi_i(\xi)\Psi_k(\xi)\rangle, \\
\sum_{k=0}^{P} u_k(x,0)\langle \Psi_k^2(\xi)\rangle &= \sum_{k=0}^{P} u_k^0(x)\langle \Psi_k^2(\xi)\rangle,
\end{aligned}$$

$$u_k(x,0) \;=\; u_k^0(x), \quad \text{where} \quad k = 0, 1, \ldots, P. \tag{2.2.12}$$

In a similar fashion, we can insert PCEs into the BCs

$$\sum_{i=0}^{P} u_i(0,t)\Psi_i(\xi) = \sum_{i=0}^{P} u_i(2,t)\Psi_i(\xi) = 0. \tag{2.2.13}$$

Further, take Galerkin projection by taking inner product with respect to $\Psi_k(\xi)$

$$\begin{aligned}
\sum_{i=0}^{P} u_i(0,t)\langle\Psi_i(\xi)\Psi_k(\xi)\rangle &= \sum_{i=0}^{P} u_i(2,t)\langle\Psi_i(\xi)\Psi_k(\xi)\rangle = 0, \\
\sum_{k=0}^{P} u_k(0,t)\langle\Psi_k^2(\xi)\rangle &= \sum_{k=0}^{P} u_k(2,t)\langle\Psi_k^2(\xi)\rangle = 0, \\
u_k(0,t) &= u_k(2,t) = 0, \quad \text{where} \quad k = 0, 1, \ldots, P.
\end{aligned}$$

Summing up by taking $u = [u_0, u_1, \ldots, u_P]$ as the vector of gPC coefficients. We have,

$$\begin{aligned}
\frac{\partial u^P(x,t)}{\partial t} + V\frac{\partial u^P(x,t)}{\partial x} &= B\frac{\partial^2 u^P(x,t)}{\partial^2 x}, \\
u^P(x,0) &= (u^0(x))^P, \\
u^P(0,t) = 0,\ u^P(2,t) &= 0.
\end{aligned} \tag{2.2.14}$$

$$\begin{aligned}
\text{where} \quad V &= \sum_{k=0}^{P} v_j(x,t)\frac{\langle\Psi_j(\xi)\Psi_i(\xi)\Psi_k(\xi)\rangle}{k!} = 0, \quad \text{where} \;\; i,j = 0,1,\ldots,P. \\
B &= \sum_{k=0}^{P} \mu_j(x,t)\frac{\langle\Psi_j(\xi)\Psi_i(\xi)\Psi_k(\xi)\rangle}{k!} = 0, \quad \text{where} \;\; i,j = 0,1,\ldots,P.
\end{aligned}$$

Now we will apply SBP-SAT technique on our problem and will find out the conditions for achieving stability. K. Kormann and M. Kronbichler [207] explains how to implement Dirichlet BCs in advection diffusion equation. The SBP-SAT technique of Eq.(2.2.14) is given by

$$\begin{aligned}
\frac{\partial u}{\partial t} + (\mathcal{P}^{-1}Q \otimes V)u - (\mathcal{P}^{-1}(-M + \tilde{B}\mathscr{D}) \otimes B)u \\
&= (\mathcal{P}^{-1} \otimes I)(E_0 \otimes \Sigma_0^I)(u - g_0) \\
&+ (\mathcal{P}^{-1} \otimes I)(\mathscr{D}^T \otimes I)(E_0 \otimes \Sigma_0^V)(u - g_0) \\
&+ (\mathcal{P}^{-1} \otimes I)(E_N \otimes \Sigma_N^I)(u - g_1) \\
&+ (\mathcal{P}^{-1} \otimes I)(\mathscr{D}^T \otimes I)(E_N \otimes \Sigma_N^V)(u - g_1).
\end{aligned} \tag{2.2.15}$$

We can see that the right hand side is simply discretizing the PDE whereas the left hand side is enforcing Dirichlet BCs.

Proposition 2.2.1 *The scheme is stable if* $\Sigma_N^V = -B$, $\Sigma_0^V = B$, $\Sigma_0^I \leq -\frac{V}{2}$ *and* $\Sigma_N^I \leq \frac{V}{2}$.

Proof : Multiplying Eq. (2.2.15) by $u^T(\mathcal{P} \otimes I)$ and setting $\tilde{B} = E_N - E_0$ and BCs to be homogeneous in Eq. (2.2.15), we get

$$\begin{aligned} u^T(\mathcal{P} \otimes I)\frac{\partial u}{\partial t} + u^T(\mathcal{P} \otimes I)(\mathcal{P}^{-1}Q \otimes V)u &= u^T(\mathcal{P} \otimes I)(\mathcal{P}^{-1}(-M + (E_N - E_0)\mathscr{D}) \otimes B)u \\ &+ u^T(\mathcal{P} \otimes I)(\mathcal{P}^{-1} \otimes I)(E_0 \otimes \Sigma_0^I)(u - 0) \\ &+ u^T(\mathcal{P} \otimes I)(\mathcal{P}^{-1} \otimes I)(\mathscr{D}^T \otimes I)(E_0 \otimes \Sigma_0^V)(u - 0) \\ &+ u^T(\mathcal{P} \otimes I)(\mathcal{P}^{-1} \otimes I)(E_N \otimes \Sigma_N^I)(u - 0) \\ &+ u^T(\mathcal{P} \otimes I)(\mathcal{P}^{-1} \otimes I)(\mathscr{D}^T \otimes I)(E_N \otimes \Sigma_N^V)(u - 0). \end{aligned}$$

Using the property $(\mathcal{K} \otimes \mathcal{L})(\mathfrak{M} \otimes \mathcal{N}) = (\mathcal{K}\mathfrak{M} \otimes \mathcal{L}\mathcal{N})$, we get

$$\begin{aligned} u^T(\mathcal{P} \otimes I)\frac{\partial u}{\partial t} + u^T(\mathcal{P}\mathcal{P}^{-1}Q \otimes V)u &= u^T(\mathcal{P}\mathcal{P}^{-1}(-M + (E_N - E_0)\mathscr{D}) \otimes B)u \\ &+ u^T(\mathcal{P}\mathcal{P}^{-1} \otimes I)(E_0 \otimes \Sigma_0^I)u \\ &+ u^T(\mathcal{P}\mathcal{P}^{-1} \otimes I)(\mathscr{D}^T \otimes I)(E_0 \otimes \Sigma_0^V)u \\ &+ u^T(\mathcal{P}\mathcal{P}^{-1} \otimes I)(E_N \otimes \Sigma_N^I)u \\ &+ u^T(\mathcal{P}\mathcal{P}^{-1} \otimes I)(\mathscr{D}^T \otimes I)(E_N \otimes \Sigma_N^V)u, \end{aligned}$$

$$\begin{aligned} u^T(\mathcal{P} \otimes I)\frac{\partial u}{\partial t} + u^T(Q \otimes V)u = u^T((-M + (E_N - E_0)\mathscr{D}) \otimes B)u + u^T(I \otimes I)(E_0 \otimes \Sigma_0^I)u \\ + u^T(I \otimes I)(\mathscr{D}^T \otimes I)(E_0 \otimes \Sigma_0^V)u + u^T(I \otimes I)(E_N \otimes \Sigma_N^I)u \\ + u^T(I \otimes I)(\mathscr{D}^T \otimes I)(E_N \otimes \Sigma_N^V)u, \end{aligned}$$

$$\begin{aligned} u^T(\mathcal{P} \otimes I)\frac{\partial u}{\partial t} + u^T(Q \otimes V)u &= -u^T(M \otimes B)u + u^T(E_N\mathscr{D} \otimes B)u - u^T(E_0\mathscr{D} \otimes B)u \\ &+ u^T(E_0 \otimes \Sigma_0^I)u + u^T(\mathscr{D}^T E_0 \otimes \Sigma_0^V)u + u^T(E_N \otimes \Sigma_N^I)u \\ &+ u^T(\mathscr{D}^T E_N \otimes \Sigma_N^V)u. \end{aligned} \tag{2.2.16}$$

By setting $\Sigma_0^V = B$ and $\Sigma_N^V = -B$, four terms on right side of Eq. (2.2.16) cancel down.

Also, set $u^T(Q \otimes V)u = \frac{1}{2}u^T((E_N - E_0) \otimes V)u$, we get

$$\begin{aligned} u^T(\mathcal{P} \otimes I)\frac{\partial u}{\partial t} + \frac{1}{2}u^T((E_N - E_0) \otimes V)u &= -u^T(M \otimes B)u + u^T(E_0 \otimes \Sigma_0^I)u \\ &+ u^T(E_N \otimes \Sigma_N^I)u, \\ u^T(\mathcal{P} \otimes I)\frac{\partial u}{\partial t} + u^T\left(E_N \otimes \frac{V}{2}\right)u - u^T\left(E_0 \otimes \frac{V}{2}\right)u &= -u^T(M \otimes B)u \\ &+ u^T(E_0 \otimes \Sigma_0^I)u + u^T(E_N \otimes \Sigma_N^I)u, \end{aligned}$$

Therefore, we get

$$\begin{aligned} u^T(\mathcal{P} \otimes I)\frac{\partial u}{\partial t} &= -u^T(M \otimes B)u + u^T\left(E_0 \otimes \left(\Sigma_0^I + \frac{V}{2}\right)\right)u \\ &+ u^T\left(E_N \otimes \left(\Sigma_N^I - \frac{V}{2}\right)\right)u. \end{aligned} \tag{2.2.17}$$

Now, to obtain the conditions for stability of the problem, we set $\Sigma_0^I = -\delta_1 V$ and $\Sigma_N^I = -\delta_2 V$. Next, we derive the energy estimate by taking the transpose of Eq. (2.2.17)

$$\begin{aligned} \left(\frac{\partial u}{\partial t}\right)^T(\mathcal{P} \otimes I)^T u &= -u^T(M \otimes B)^T u + u^T\left(E_0 \otimes \left(-\delta_1 V + \frac{V}{2}\right)\right)^T u \\ &+ u^T\left(E_N \otimes \left(-\delta_2 V - \frac{V}{2}\right)\right)^T u, \\ &= -u^T(M \otimes B)^T u - u^T\left(E_0 \otimes \left(\delta_1 V - \frac{V}{2}\right)\right)^T u \\ &- u^T\left(E_N \otimes \left(\delta_2 V + \frac{V}{2}\right)\right)^T u, \end{aligned} \tag{2.2.18}$$

and then adding Eq. (2.2.17) to Eq. (2.2.18) as follows

$$\begin{aligned} u^T(\mathcal{P} \otimes I)\frac{\partial u}{\partial t} + \left(\frac{\partial u}{\partial t}\right)^T(\mathcal{P} \otimes I)^T u = -u^T(M \otimes B)u - u^T(M \otimes B)^T u \\ - u^T\left(E_0 \otimes \left(\delta_1 V - \frac{V}{2}\right)\right)u - u^T\left(E_0 \otimes \left(\delta_1 V - \frac{V}{2}\right)\right)^T u \\ - u^T\left(E_N \otimes \left(\delta_2 V + \frac{V}{2}\right)\right)u - u^T\left(E_N \otimes \left(\delta_2 V + \frac{V}{2}\right)\right)^T u, \end{aligned}$$

As we know

$$\left\langle u, \frac{\partial u}{\partial t}\right\rangle_{\mathcal{P}\otimes I} + \left\langle \frac{\partial u}{\partial t}, u\right\rangle_{\mathcal{P}\otimes I} = \frac{\partial}{\partial t}\|u\|^2{}_{\mathcal{P}\otimes I}. \tag{2.2.19}$$

Since B and V are both symmetric matrices, therefore, $B = B^T$ and $V = V^T$ and hence, we get

$$\frac{\partial}{\partial t}\|u\|^2_{\mathcal{P}\otimes I} = -u^T((M+M^T)\otimes B)u - u^T\left(E_0 \otimes \left(\delta_1 V - \frac{V}{2}\right)\right)u - u^T\left(E_0 \otimes \left(\delta_1 V - \frac{V}{2}\right)\right)^T u$$
$$-u^T\left(E_N \otimes \left(\delta_2 V + \frac{V}{2}\right)\right)u - u^T\left(E_N \otimes \left(\delta_2 V + \frac{V}{2}\right)\right)^T u,$$

$$\begin{aligned}\frac{\partial}{\partial t}\|u\|^2_{\mathcal{P}\otimes I} &= -u^T((M+M^T)\otimes B)u - u^T(E_0 \otimes (2\delta_1 V - V))u - u^T(E_N \otimes (2\delta_2 V + V))u,\\ &= -u^T((M+M^T)\otimes B)u - u^T(E_0 \otimes (2\delta_1 - 1)V)u - u^T(E_N \otimes (2\delta_2 + 1)V)u,\\ &= -u^T((M+M^T)\otimes B)u + u_0^T(1-2\delta_1)v)u_0 + u_N^T(-1-2\delta_2)v)u.\end{aligned}$$

Therefore, we have

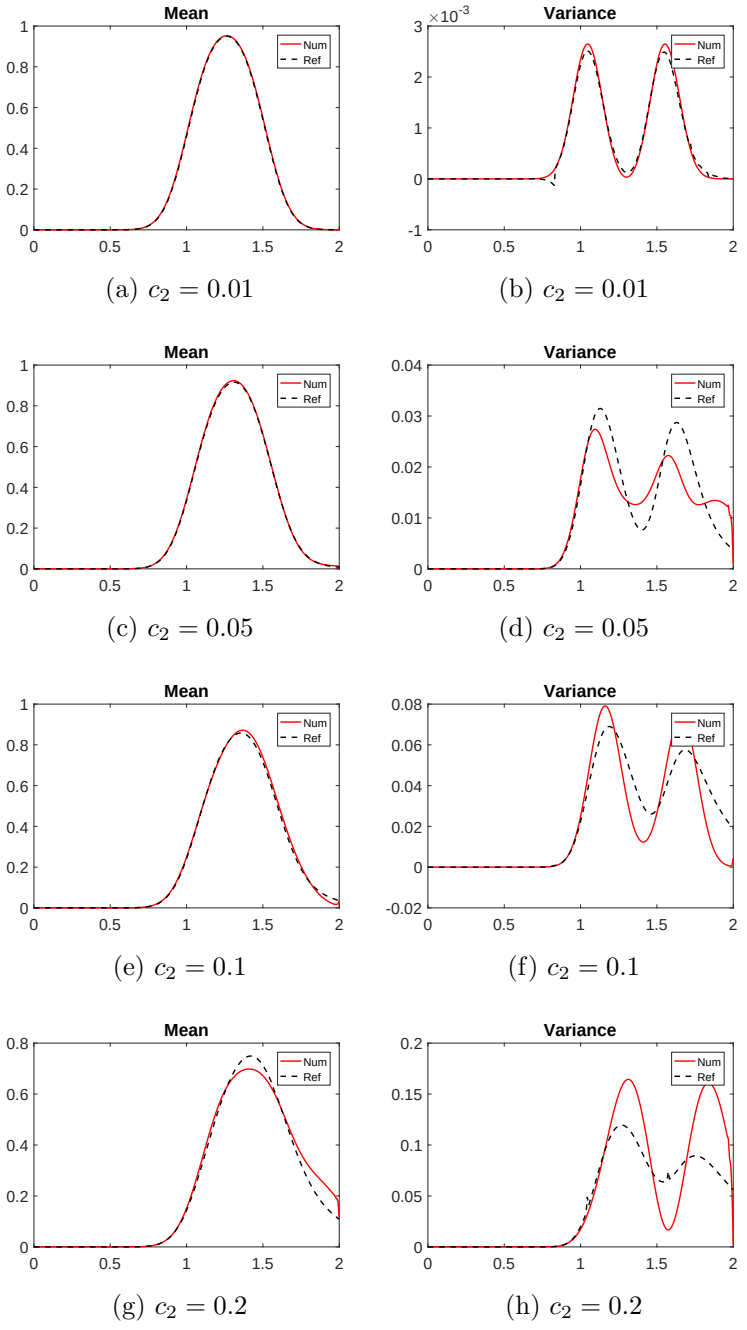
$$\begin{aligned}-2\delta_1 + 1 \leq 0 \quad \text{and} \quad -1 - 2\delta_2 \leq 0,\\ \delta_1 \geq \frac{1}{2} \quad \text{and} \quad \delta_2 \geq -\frac{1}{2}.\end{aligned} \tag{2.2.20}$$

Since $M + M^T$ and B are positive definite, the Eq. (2.2.20) with $\delta_1 \geq \frac{1}{2}$ and $\delta_2 \geq -\frac{1}{2}$, i.e., $\Sigma_0^I \leq -\frac{V}{2}$ and $\Sigma_N^I \leq \frac{V}{2}$, proves the scheme (2.2.15) is stable.

Using the above obtained parameters for stability and setting $c_4 = 0$, $a = 0.2$, $b = 0.7$, we have performed the simulations. Also, we set the mean velocity field with $c_1 = 1$. Basically, the parameters c_2 and c_4 are of utmost importance as they will be utilized to manage the amplitude of the stochastic perturbations. We have used two different values of mean diffusion coefficients $c_3 = 10^{-3}$ and $c_3 = 10^{-2}$ in our simulations. We have discretized the spatial grid into 250 grid points and have displayed the numerical results at time $t = 0.8$ with $dt = 0.00001$. Fig. 2.4 and Fig. 2.5 shows results of mean and variance for the parameters $c_3 = 10^{-2}$ and $c_3 = 10^{-3}$ respectively. In Fig. 2.4, we notice that for the small values of the parameter c_2, mean plots are perfectly resembling with the mean of the reference solution but as we are increasing the value of c_2 such as setting the value of $c_2 = 0.2$, the mean is showing a little deviation. Also, variance is increasing from a range of 10^{-3} to 0.15 as we are increasing the value of parameter c_2 which is theoretically expected. Therefore, we conclude that variance is directly proportional to the randomness. Higher the randomness, more will be the variance.

In Fig. 2.5 with $c_2 = 10^{-3}$, the mean plots match exactly with the analytical solution for small values of $c_2 = 0.01$ and $c_2 = 0.05$, but as we are increasing the value of c_2 from

0.1 to 0.2, the mean is showing changes. Also, the variance in fig. 2.5 is increasing with the increase in values of parameter c_2, but at higher values of the parameter $c_2 = 0.2$, the variance is showing oscillations at the right side.



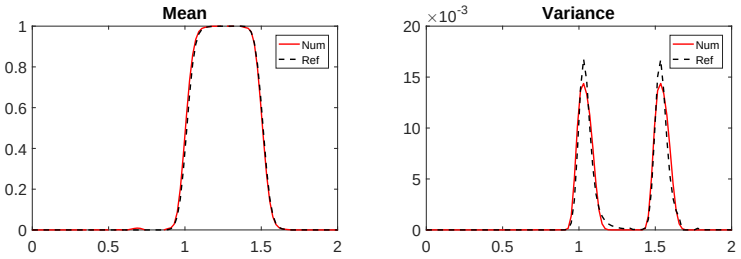
(a) $c_2 = 0.01$ (b) $c_2 = 0.01$

(c) $c_2 = 0.05$ (d) $c_2 = 0.05$

(e) $c_2 = 0.1$ (f) $c_2 = 0.1$

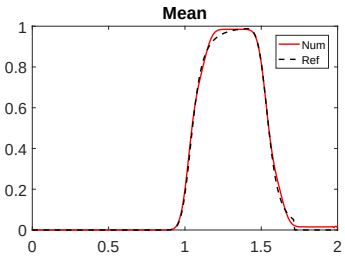
(g) $c_2 = 0.2$ (h) $c_2 = 0.2$

Figure 2.4: Mean (Left) and Variance (Right) of Test problem 3 at $t = 0.8$ with different values of parameter c_2 and $c_3 = 10^{-2}$.

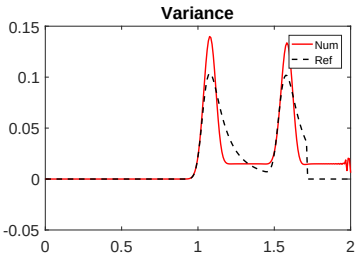


(a) $c_2 = 0.01$

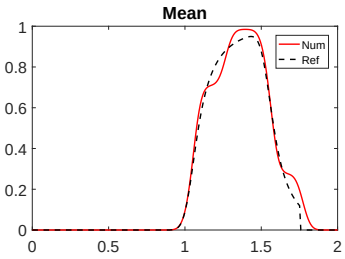
(b) $c_2 = 0.01$



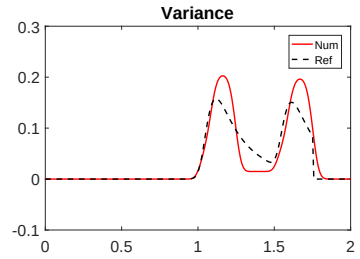
(c) $c_2 = 0.05$



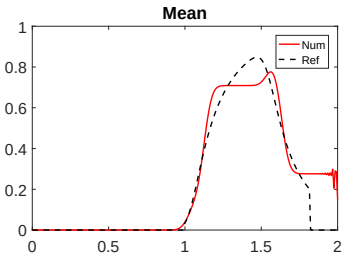
(d) $c_2 = 0.05$



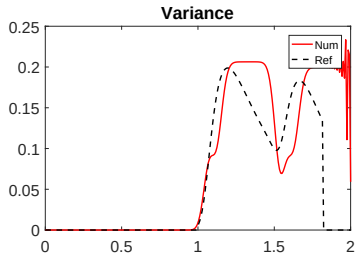
(e) $c_2 = 0.1$



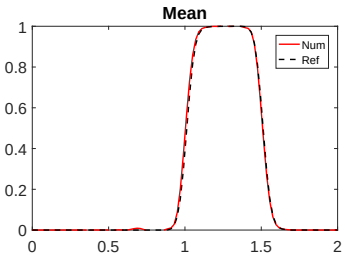
(f) $c_2 = 0.1$



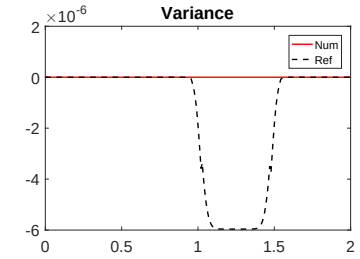
(g) $c_2 = 0.2$



(h) $c_2 = 0.2$



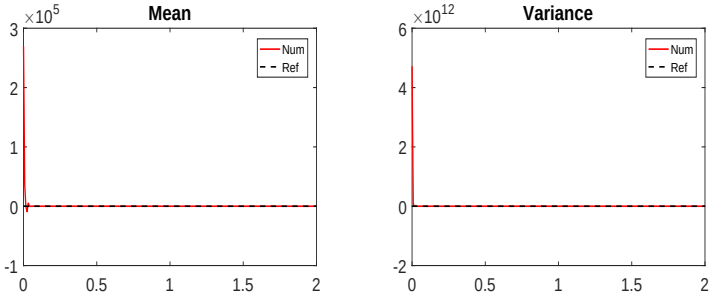
(i) $c_2 = 0$ and $c_4 = 0$



(j) $c_2 = 0$ and $c_4 = 0$

Figure 2.5: Mean (Left) and Variance (Right) of Test problem 3 at $t = 0.8$ with different values of parameter c_2 and $c_3 = 10^{-3}$.

In Fig. 2.6, we have displayed the mean and variance of the unstable case of the above problem at $t = 0.02$ with parameters $c_2 = 0.2$ and $c_3 = 10^{-3}$. For the unstable case, we have taken 250 spatial grid points and took the values $\Sigma_0^I = V$ and $\Sigma_N^I = V$ which makes the procedure unstable. In the unstable case, the mean and the variance are increasing very rapidly with increase in time. Moreover, if we take another case with $c_2 = 0$ and $c_4 = 0$, i.e., introducing no randomness, then the variance is zero which is theoretically expected as shown in the last case of fig. 2.5. We have compared the results of third test problem with the existing literature by following paper titled "A new stochastic approach for advection-diffusion problems with uncertain parameters [198]". It has been noticed that the proposed method shows better results than the method implemented in [198] which employs PCE with the method of characteristics for solution of advection diffusion equation. After applying the proposed method on the three test problems, we will now discuss



(a) $c_2 = 0.2$ (b) $c_2 = 0.2$

Figure 2.6: Mean (Left) and Variance (Right) of the unstable case of Test problem 3 at $t = 0.02$ with parameters $c_2 = 0.2$ and $c_3 = 10^{-3}$ by taking $\Sigma_0^I = V$ and $\Sigma_N^I = V$.

the advantages as well as the disadvantages of the proposed method. The advantages are as below:

1) HPSBPSAT combines the strengths of both PCE and SBP-SAT operators. Spatial derivatives are approximated by SBP operators by ensuring stable solutions. Their practicality lies in the scope of expressing energy decay with respect to known boundary values, in the same manner as in the continuous case and in order to have a complete statistical knowledge of the solution, PCE requires a single simulation which is indeed time saving.

2) HPSBPSAT provides easy access to the statistics of the random inputs including moments and the PDF, providing an expansion where the zero-index term contains the solution mean.

Further, the proposed method has few limitations such as

1) It suffers from the curse of dimensionality which means the count of random parameters has an effect on both the computational cost and model complexity of any PC-based methods, and, when sufficiently high n is considered, the efficiency of such approaches can be greatly reduced.

2) As the proposed method is intrusive in nature, thus, it demands modification in the codes which may turn out to be troublesome, time consuming and expensive for various problems.

2.3 Conclusion

In this chapter, Hybrid Hermite polynomial chaos SBP-SAT technique has been applied to advection and advection-diffusion problems with uncertain input data. With the help of Hermite polynomial chaos, the randomness is separated, and with the help of SBP operators, differential operators are approximated. After that, SAT are used to impose the BCs ensuring stability, and finally RK4 is used for time integration. The first two test problems are simple advection equations on $\mathbb{R}$ without any boundary and therefore, require only SBP operators whereas the third test problem is a stochastic advection-diffusion equation with uncertain Dirichlet BCs. It needs SAT treatment along with SBP operator for enforcing the BCs. A range of permissable parameters for a stable numerical solution are obtained for the third problem. The statistical data of the solution, i.e., mean and variance are determined from the coefficients of the PCE without requiring any additional cost.

Chapter 3

Uncertainty propagation using Wiener-Linear B-spline wavelet expansion[1]

Wavelet is a wave-like oscillation with an amplitude that begins at zero, increases and then decreases back to zero. It is typically visualized as a brief oscillation like one recorded by a seismograph or heart monitor. Wavelets being local waves are known for expressing discontinuities or steep gradients more accurately than the global basis. On the other hand, gPC has a slow convergence or sometimes fails to converge even in short-time integration.

The theory of wavelets has become comparatively upcoming and a new area in the field of UQ. The development of new wavelet based gPC has become a topic of recent research. To the best of our knowledge, Haar wavelets [55] and multiwavelets [88] are the only wavelets which have been explored in this field. According to literature, semi-orthogonal B-spline wavelets [208, 209] have never been used as the gPC in the area of UQ till now. Chui and Quak [210] constructed the first spline multi-resolution analysis for the interval [0,1]. Later on, Quak and Weyrich [211] presented the reconstruction and decomposition algorithms for spline wavelets on a bounded interval. Apart from that, they also gave the construction of dual B-splines and dual B-wavelets.

B-spline wavelets are further divided according to the order n_B such as linear B-spline wavelets ($n_B = 2$), quadratic B-spline wavelets ($n_B = 3$), cubic B-spline wavelets ($n_B = 4$), quartic B-spline wavelets ($n_B = 5$) and so on. Further, we can categorize the wavelets into orthogonal and semi-orthogonal on the basis of property orthogonality. However, orthogonal wavelets are either non-symmetric in some situations or have infinite support. That is the reason why they become poor choice for characterization of a function. On the other hand, semi-orthogonal wavelets have both even and odd symmetry, finite support, and simple analytical expressions for a basis function. Therefore, we aim to develop the semi-orthogonal B-spline wavelet based gPC in this chapter.

The main contribution of this chapter is the development of B-spline wavelet based gPC for handling stochastic DEs. Till now, linear B-spline wavelets have never been studied

[1]The content of this chapter is published as "Uncertainty propagation using Wiener-Linear B-spline wavelet expansion", *Computers & Mathematics with Applications*, 79(9), 2598-2623, 2020, **(SCI: Impact Factor: 3.476).**

as gPC in the field of UQ. We have applied Galerkin and pseudo-spectral approach to handle nonlinearity. From literature, we have analyzed that Galerkin projection shows superior results in comparison to pseudo-spectral approach [212]. Also, wavelet function approximation displays better results than scaling function approximation. Moreover, Wiener B-spline (WBe) gPC show better approximations when compared with Wiener Legendre (WLe) gPC in research paper entitled "Uncertainty propagation using Wiener-Haar expansions [55]".

3.1 B-spline Expansion

In this section, the construction of WBe representation of a random process is explained. Let us first discuss the decomposition of probability space with a single dimension by employing the B-spline wavelets.

3.1.1 Probability space

Let us take θ as the RV and $\wp(x)$ be the probability for $\theta < x$. Moreover, suppose that $\wp(x)$ is a continuous strictly increasing function of x defined on a real interval (a, b) such that $-\infty \leq a < b \leq \infty$, i.e.,

$$\wp : [a, b] \rightarrow [0, 1] \quad \text{with} \quad \wp(a) = 0, \quad \wp(b) = 1.$$

Note that, it can be extended to the infinite case, but in this thesis, we will only consider cases where a and b are finite. After defining the probability $\wp(x)$, we move forward to the PDF of x on (a, b) which is defined as follows:

$$pdf(x) = \begin{cases} \frac{d\wp(x)}{dx} > 0, & \forall \ \ x \in (a, b), \\ 0, & \forall \ \ x \notin (a, b). \end{cases}$$

According to the assumed properties of $\wp(x)$ and by the inverse function theorem of continuous strictly monotonic functions (which gives a sufficient condition for a function to be invertible in a neighborhood of a point in its domain: namely, that its derivative is continuous and non-zero at the point), it follows that $\forall\ y \in [0, 1]$, there is a unique $x \in [a, b]$ such that

$$\wp(x) = y.$$

Hence, we specify a one-to-one mapping

$$\wp^{-1} : [0, 1] \rightarrow [a, b],$$

such that

$$\wp^{-1}(y) = x \in [a, b].$$

3.1.2 Multi-resolution analysis

The backbone of wavelet theory is MRA that express a function $f(x) \in L^2(\mathbb{R})$ at various levels of details. A multi resolution approximation of $L^2(\mathbb{R})$ is specified by a set of nested subspaces $\mathcal{V}_j$ in $L^2(\mathbb{R})$

$$\mathcal{V}_j \subset \mathcal{V}_{j+1}, \forall j \in \mathbb{Z} \tag{3.1.1}$$

which satisfies the below mentioned properties:

$$\begin{aligned} &\bigcup_{j \in Z} \mathcal{V}_j \text{ is dense in } L^2(\mathbb{R}), \\ &\bigcap_{j \in Z} \mathcal{V}_j = \emptyset, \\ &f(x) \in \mathcal{V}_j \Longleftrightarrow f(2x) \in \mathcal{V}_{j+1}, \forall j \in \mathbb{Z} \end{aligned} \tag{3.1.2}$$

For each j, $\exists$ a scaling function $\phi(x)$ such that

$$\{\phi_{j,k}(x)\}_{k \in \mathbb{Z}} = \{\phi(2^j x - k)\}_{k \in \mathbb{Z}} \tag{3.1.3}$$

is a basis for $\mathcal{V}_j$. Moreover, $\exists$ an orthogonal complementary subspace $\mathcal{W}_j$ of $\mathcal{V}_j$ in $\mathcal{V}_{j+1}$

$$\mathcal{V}_{j+1} = \mathcal{V}_j \bigoplus \mathcal{W}_j \tag{3.1.4}$$

as well as a wavelet function $\psi(x)$, such that

$$\{\psi_{j,k}(x)\}_{k \in \mathbb{Z}} = \{\psi(2^j x - k)\}_{k \in \mathbb{Z}} \tag{3.1.5}$$

is a basis for $\mathcal{W}_j$, thus, $\{\psi_{j,k}(x)\}_{j,k \in \mathbb{Z}}$ is a basis for $L^2(\mathbb{R})$.

Moreover, we have the following results

$$\begin{aligned} \mathcal{V}_{j_1} \bigcap \mathcal{V}_{j_2} &= \mathcal{V}_{j_2},\ j_1 > j_2, \\ \mathcal{W}_{j_1} \bigcap \mathcal{W}_{j_2} &= 0,\ j_1 \neq j_2, \\ \mathcal{V}_{j_1} \bigcap \mathcal{W}_{j_2} &= 0,\ j_1 = j_2, \end{aligned} \tag{3.1.6}$$

Further, semi-orthogonal B-spline wavelets are known to generate the above MRA with some important properties such as

1. **Vanishing moments**: Vanishing moment of a wavelet of order r is given as

$$\int_{-\infty}^{\infty} x^p \psi(x) dx = 0, \quad p = 0, \ldots, r-1 \tag{3.1.7}$$

   All wavelets must satisfy the above condition for $p = 0$.

2. **Semi-orthogonality**: The wavelets $\psi_{j,k}$ form a semi-orthogonal basis if

$$\langle \psi_{j,k}, \psi_{s,i} \rangle = 0;\ j \neq s;\ \forall\ j, k, s, i \in \mathbb{Z} \tag{3.1.8}$$

3.1.3 B-spline scaling and wavelet functions

In this section, we derive the scaling and wavelet functions of compactly supported semi-orthogonal B-spline wavelets of order n_B. Throughout this chapter, we focus on second order B-spline wavelets and acquire the expressions for their scaling and wavelet functions. Generally, we have $n_B - 1$ boundary scaling functions at zero and one and $2^\jmath - n_B + 1$ inner scaling functions. Also, we have $n_B - 1$ boundary wavelets at zero and one and $2^\jmath - 2n_B + 2$ inner wavelets [213] .

3.1.3.1 Algorithm

We follow the below mentioned algorithm for acquiring the expressions of scaling and wavelet functions of B-spline wavelets of any order.

1) We initially decide scale $\jmath$ of B-spline wavelets. For this, we need to satisfy the following condition

$$2^\jmath \geq 2n_B - 1, \tag{3.1.9}$$

so as to have atleast one inner wavelet. After that, at any scale $\jmath$, we have the discretization step as $1/2^{\jmath}$ with $\jmath > 0$ which follows that there will be $\mathrm{s} = 2^{\jmath}$ number of segments in $[0, 1]$.

2) Let $\jmath \in \mathbb{Z}$, let $\{\mathfrak{t}_k^{\jmath}\}_{k=-n_B+1}^{2^{\jmath}+n_B-1}$ be a knot sequence with n_B-tuple knots at 0 and 1, and simple knots inside the unit interval as given below

$$\begin{cases} \mathfrak{t}_{-n_B+1}^{\jmath} = \mathfrak{t}_{-n_B+2}^{\jmath} = \ldots = \mathfrak{t}_0^{\jmath} = 0, \\ \mathfrak{t}_k^{\jmath} = \frac{k}{2^{\jmath}}, \quad (k = 1, \ldots, 2^{\jmath} - 1) \\ \mathfrak{t}_{2^{\jmath}}^{\jmath} = \mathfrak{t}_{2^{\jmath}+1}^{\jmath} = \ldots = \mathfrak{t}_{2^{\jmath}+n_B-1}^{\jmath} = 1. \end{cases} \tag{3.1.10}$$

3) We now define the B-spline ($n_B \geq 2$) for the above knot sequence (3.1.10) in the following way

$$\mathfrak{B}_{n_B,\jmath,k}(x) = (\mathfrak{t}_{k+n_B}^{\jmath} - \mathfrak{t}_k^{\jmath}) \times [\mathfrak{t}_k^{\jmath}, \mathfrak{t}_{k+1}^{\jmath}, \ldots, \mathfrak{t}_{k+n_B}^{\jmath}]_{\mathfrak{t}} (\mathfrak{t} - x)_+^{n_B-1},$$

where $[\cdot, \ldots, \cdot]_t$ is the n_B^{th} order divided difference of $(\mathfrak{t} - x)_+^{n_B-1}$ with respect to $\mathfrak{t}$ and $(x)_+$=max$(0, x)$. Also, truncated power function is given by

$$(\mathfrak{t} - x)_+^{n_B} = \begin{cases} (\mathfrak{t} - x)^{n_B}, & \text{if } \mathfrak{t} > x, \\ 0, & \text{otherwise.} \end{cases}$$

4) Suppose $\jmath_0$ be the scale for which the condition (3.1.9) is satisfied. Then, for each $\jmath \geq \jmath_0$, the scaling functions $\phi_{n_B,\jmath,k}$ of order n_B are defined as

$$\phi_{n_B,\jmath,k}(x) = \begin{cases} \mathfrak{B}_{n_B,\jmath_0,k}(2^{\jmath-\jmath_0}x), & k = -n_B + 1, \ldots, -1, \\ \mathfrak{B}_{n_B,\jmath_0,2^{\jmath}-n_B-k}(1 - 2^{\jmath-\jmath_0}x), & k = 2^{\jmath} - n_B + 1, \ldots, 2^{\jmath} - 1, \\ \mathfrak{B}_{n_B,\jmath_0,0}(2^{\jmath-\jmath_0}x - 2^{-\jmath_0}k), & k = 0, \ldots, 2^{\jmath} - n_B. \end{cases} \tag{3.1.11}$$

5) Now, we derive the B-spline wavelet functions [214] using the two scale relation given by

$$\begin{aligned} \psi_{n_B,\jmath,i-n_B} &= \sum_{k=i}^{2i+2n_B-2} r_{i,k} \mathfrak{B}_{n_B,\jmath,k-n_B}, \quad i = 1, \ldots, n_B - 1, \\ \psi_{n_B,\jmath,i-n_B} &= \sum_{k=2i-n_B}^{2i+2n_B-2} r_{i,k} \mathfrak{B}_{n_B,\jmath,k-n_B}, \quad i = n_B, \ldots, \mathrm{s} - n_B + 1, \\ \psi_{n_B,\jmath,i-n_B} &= \sum_{k=2i-n_B}^{\mathrm{s}+i+n_B-1} r_{i,k} \mathfrak{B}_{n_B,\jmath,k-n_B}, \quad i = \mathrm{s} - n_B + 2, \ldots, \mathrm{s}, \end{aligned}$$

where $r_{i,k} = r_{k-2i}$.

6) We can express compactly supported B-wavelet functions in [0,1] as follows:

$$\psi_{n_B,\jmath,i}(x) = \begin{cases} \psi_{n_B,\jmath_0,i}(2^{\jmath-\jmath_0}x), & i = -n_B+1,\ldots,-1, \\ \psi_{n_B,2^{\jmath}-2n_B+1-i,i}(1-2^{\jmath-\jmath_0}x), & i = 2^{\jmath}-2n_B+2,\ldots,2^{\jmath}-n_B, \\ \psi_{n_B,\jmath_0,0}(2^{\jmath-\jmath_0}x-2^{-\jmath_0}i), & i = 0,\ldots,2^{\jmath}-2n_B+1. \end{cases} \tag{3.1.12}$$

Therefore, we have $(2^{\jmath}-2n_B+2)$ inner wavelets and n_B-1 boundary wavelets at each boundary in the bounded interval $[a,b]$.

3.1.3.2 Compactly supported semi-orthogonal linear B-spline scaling and wavelet functions

By applying the above algorithm, we acquire the linear B-spline wavelets by putting order of B-spline as $n_B = 2$. So, we initially start with the smallest octave level $\jmath = \jmath_0 = 2$ which satisfies Eq. (3.1.9) as $4 \geq 3$. Therefore, for $\jmath_0 = 2$, there will be one boundary scaling function at each boundary and three inner scaling functions.

Therefore, inner scaling B-spline functions [215] with $x_\jmath = 2^{\jmath}x$ for $k = 0,\ldots,2^{\jmath}-2$ are given by

$$\phi_{\jmath,k}(x) = \begin{cases} x_\jmath - k, & k \leq x_\jmath \leq k+1, \\ 2-(x_\jmath-k), & k+1 \leq x_\jmath \leq k+2, \\ 0, & \text{otherwise.} \end{cases}$$

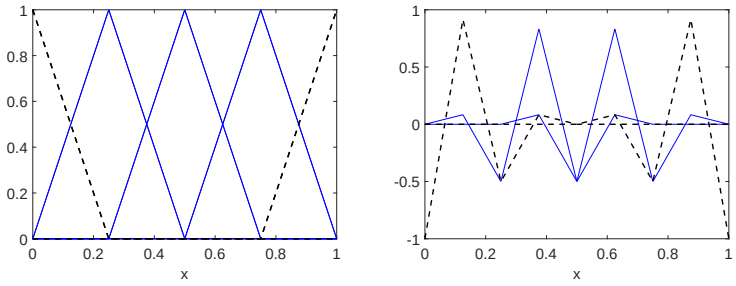
Left boundary scaling function for $k=-1$ is given by

$$\phi_{\jmath,k}(x) = \begin{cases} 2-(x_\jmath-k), & 0 \leq x_\jmath \leq 1, \\ 0, & \text{otherwise.} \end{cases}$$

Right boundary scaling function for $k = 2^{\jmath}-1$ is given by

$$\phi_{\jmath,k}(x) = \begin{cases} x_\jmath - k, & k \leq x_\jmath \leq k+1, \\ 0, & \text{otherwise.} \end{cases}$$

For $\jmath_0 = 2$, we have two inner wavelet functions and one boundary wavelet function at each boundary that means we have total of four wavelet functions. The second order inner



(a) (b)

Figure 3.1: a) Linear B-spline scaling functions b) Linear B-spline wavelet functions

B-spline wavelets for $k = 0, \ldots, 2^j - 3$ are given as follows:

$$\psi_{j,k}(x) = \frac{1}{6}\begin{cases} x_j - k, & k \le x_j \le k + 0.5, \\ 4 - 7(x_j - k), & k + 0.5 \le x_j \le k + 1, \\ -19 + 16(x_j - k), & k + 1 \le x_j \le k + 1.5, \\ 29 - 16(x_j - k), & k + 1.5 \le x_j \le k + 2, \\ -17 + 7(x_j - k), & k + 2 \le x_j \le k + 2.5, \\ 3 - (x_j - k), & k + 2.5 \le x_j \le k + 3, \\ 0, & \text{otherwise.} \end{cases}$$

Left boundary wavelet function for $k = -1$ is given by

$$\psi_{j,k}(x) = \frac{1}{6}\begin{cases} -6 + 23x_j, & 0 \le x_j \le 0.5, \\ 14 - 17x_j, & 0.5 \le x_j \le 1, \\ -10 + 7x_j, & 1 \le x_j \le 1.5, \\ 2 - x_j, & 1.5 \le x_j \le 2, \\ 0, & \text{otherwise.} \end{cases}$$

Right boundary wavelet function for $k = 2^j - 2$ is given by

$$\psi_{j,k}(x) = \frac{1}{6}\begin{cases} 2 - (k + 2 - x_j), & k \le x_j \le k + 0.5, \\ -10 + 7(k + 2 - x_j), & k + 0.5 \le x_j \le k + 1, \\ 14 - 17(k + 2 - x_j), & k + 1 \le x_j \le k + 1.5, \\ -6 + 23(k + 2 - x_j), & k + 1.5 \le x_j \le k + 2, \\ 0, & \text{otherwise.} \end{cases}$$

We have plotted the scaling functions of linear B-spline wavelets in Fig. 3.1a where the solid line represents the inner scaling functions and dotted line represents the boundary scaling functions. Similarly, in Fig. 3.1b, we have plotted the wavelet functions of linear

B-spline wavelets where the solid line represents the inner wavelet functions and dotted line represents the boundary wavelet functions.

3.1.3.3 B-spline dual scaling functions and dual wavelets

The concept of dual scaling and dual wavelet functions for B-splines was given by Chui and Wang [216]. Later on, Quak and Weyrich [211] addressed the construction of dual scaling $\tilde{\phi}_{\jmath,k}$ and dual wavelet functions $\tilde{\psi}_{\jmath,k}$ for B-spline wavelets. The functions $\tilde{\phi}_{\jmath,k} \in \mathcal{V}^{\jmath}_{[0,1]}$ with $k = -n_B + 1, \ldots, 2^{\jmath} - 1$ are called dual scaling functions to $\phi_{\jmath,\tilde{k}} \in \mathcal{V}^{\jmath}_{[0,1]}$ if

$$\langle \tilde{\phi}_{\jmath,k}, \phi_{\jmath,\tilde{k}} \rangle = \int_0^1 \tilde{\phi}_{\jmath,k}(x)\phi_{\jmath,\tilde{k}}(x)dx = \delta_{k\tilde{k}} \quad \text{where} \quad k, \tilde{k} = -n_B + 1, \ldots, 2^{\jmath} - 1. \tag{3.1.13}$$

The functions $\tilde{\psi}_{\jmath,k} \in \mathcal{W}^{\jmath}_{[0,1]}$ with $k = -n_B + 1, \ldots, 2^{\jmath} - n_B$ are called dual wavelet functions to $\psi_{\jmath,\tilde{k}} \in \mathcal{W}^{\jmath}_{[0,1]}$ if

$$\langle \tilde{\psi}_{\jmath,k}, \psi_{\jmath,\tilde{k}} \rangle = \int_0^1 \tilde{\psi}_{\jmath,k}(x)\psi_{\jmath,\tilde{k}}(x)dx = \delta_{k\tilde{k}} \quad \text{where} \quad k, \tilde{k} = -n_B + 1, \ldots, 2^{\jmath} - n_B.$$

It has been observed that on each level, the dual scaling and dual wavelet functions span the same space as scaling and wavelet functions respectively. Since $\tilde{\phi}_{\jmath,k} \in \mathcal{V}^{\jmath}_{[0,1]}$, it holds that

$$\tilde{\phi}_{\jmath,k}(x) = \sum_{l=-n_B+1}^{2^{\jmath}-1} g^{\jmath}_{k,l}\phi_{\jmath,l}(x) \quad (k = -n_B + 1, \ldots, 2^{\jmath} - 1), \tag{3.1.14}$$

with

$$\mathbf{g}^{\jmath}_{\mathbf{k}} = (g^{\jmath}_{k,-n_B+1}, \ldots, g^{\jmath}_{k,2^{\jmath}-1})^T. \tag{3.1.15}$$

Now, to compute the coefficients $g^{\jmath}_{k,l}$, we take inner product of Eq. (3.1.14) with respect to $\phi_{\jmath,\tilde{k}}$ with $\tilde{k} = -n_B + 1, \ldots, 2^{\jmath} - 1$ as follows:

$$\langle \tilde{\phi}_{\jmath,k}(x), \phi_{\jmath,\tilde{k}}(x) \rangle = \sum_{l=-n_B+1}^{2^{\jmath}-1} g^{\jmath}_{k,l} \langle \phi_{\jmath,l}(x)\phi_{\jmath,\tilde{k}}(x) \rangle \quad (k = -n_B + 1, \ldots, 2^{\jmath} - 1),$$

and using the orthonormality condition for dual scaling functions from Eq. (3.1.13), we get

$$\mathbf{g}^{\jmath}_{\mathbf{k}} = (\mathrm{C}^{\jmath})^{-1}\mathbf{e}_{\mathbf{k}} \quad (k = -n_B + 1, \ldots, 2^{\jmath} - 1).$$

where $\mathbf{e_k}$ is the kth unit vector and

$$\mathrm{C}^{\jmath} = (c^{\jmath}_{k,l})^{2^{\jmath}-1}_{k,l=-n_B+1}, \qquad c^{\jmath}_{k,l} = \langle \phi_{\jmath,k}, \phi_{\jmath,l} \rangle.$$

Observe that local support of scaling functions means that

$$c^{\jmath}_{k,l} = 0 \quad \text{if} \quad |k-l| > n_B - 1.$$

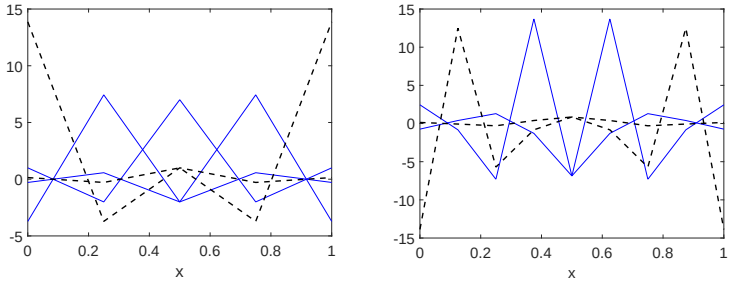
So, from here, we are able to compute the dual scaling functions. Similarly, we can compute the dual wavelet functions

$$\tilde{\psi}_{\jmath,k}(x) = \sum_{l=-n_B+1}^{2^{\jmath}-n_B} h^{\jmath}_{k,l} \psi_{\jmath,l}(x) \quad (k = -n_B+1, \ldots, 2^{\jmath} - n_B) \tag{3.1.16}$$

with

$$\mathbf{h}^{\jmath}_{\mathbf{k}} = (h^{\jmath}_{k,-n_B+1}, \ldots, h^{\jmath}_{k,2^{\jmath}-n_B})^T.$$

Now, to compute the coefficients $h^{\jmath}_{k,l}$, we take inner product of Eq. (3.1.16) with respect



(a) (b)

Figure 3.2: a) Linear dual B-spline scaling functions b) Linear dual B-spline wavelet functions

to $\psi_{\jmath,\tilde{k}}$ with $\tilde{k} = -n_B + 1, \ldots, 2^j - n_B$

$$\langle \tilde{\psi}_{\jmath,k}(x), \psi_{\jmath,\tilde{k}} \rangle = \sum_{l=-n_B+1}^{2^{\jmath}-n_B} h^{\jmath}_{k,l} \langle \psi_{\jmath,l}(x) \psi_{\jmath,\tilde{k}} \rangle \quad (k = -n_B+1, \ldots, 2^{\jmath} - n_B).$$

and using the orthonormality condition for dual scaling functions from Eq. (3.1.14), we get

$$\mathbf{h}^{\jmath}_{\mathbf{k}} = (\mathrm{D}^{\jmath})^{-1} \mathbf{e_k} \quad (k = -n_B+1, \ldots, 2^{\jmath} - n_B).$$

where $\mathbf{e_k}$ is the kth unit vector and

$$\mathrm{D}^{\jmath} = (\mathrm{d}^{\jmath}_{k,l})^{2^{\jmath}-n_B}_{k,l=-n_B+1}, \qquad \mathrm{d}^{\jmath}_{k,l} = \langle \psi_{\jmath,k}, \psi_{\jmath,l} \rangle.$$

Observe that local support of wavelets means that

$$\mathrm{d}^{\jmath}_{k,l} = 0 \quad \text{if} \quad |k-l| > 2n_B - 2.$$

Following this procedure, we analyze that we have dual scaling and dual wavelet functions corresponding to each scaling and wavelet functions. These are plotted in Fig. 3.2 (a), the solid lines represent the inner dual scaling functions and dotted lines represent the boundary dual scaling functions. Similarly, Fig. 3.2 (b), the solid lines describe the inner dual wavelet functions and dotted line describes the boundary dual wavelet functions. So, we have equal number of scaling and dual scaling functions and equal number of wavelet and dual wavelet functions. Therefore, we conclude that with respect to every scaling and wavelet function, we have dual scaling and wavelet function.

3.2 Wavelet approximation of a 1D stochastic process

We now look for a wavelet representation of $X(\xi(\theta))$, a second-order random process with ξ being the RV. Also, $\xi(\theta)$ represents the stochastic nature of ξ. Now, let us consider the expansion of the form

$$X(\xi(\theta)) = \sum_{k=-n_B+1}^{2^{J_0}-1} \mathfrak{c}_{J_0,k} \phi^s_{J_0,k} + \sum_{j=J_0}^{\infty} \sum_{k=-1}^{2^j-2} \mathfrak{d}_{j,k} \psi^w_{j,k}. \tag{3.2.1}$$

By truncating the expression in Eq. (3.2.1) to J and also by taking the value of $J_0 = 2$ from Eq. (3.1.9), we get the expression as

$$X(\xi(\theta)) = \sum_{k=-n_B+1}^{2^{J_0}-1} \mathfrak{c}_{J_0,k} \phi^s_{J_0,k} + \sum_{j=J_0}^{J} \sum_{k=-1}^{2^j-2} \mathfrak{d}_{j,k} \psi^w_{j,k}. \tag{3.2.2}$$

Moreover,

$$X(\xi(\theta)) = \sum_{k=-n_B+1}^{2^{J_0}-1} X^s_{J_0,k} S_{J_0,k}(\xi(\theta)) + \sum_{j=J_0}^{J} \sum_{k=-1}^{2^j-2} X^w_{j,k} \mathscr{W}_{j,k}(\xi(\theta)), \tag{3.2.3}$$

where $X^s_{J_0,k}$ are the coefficients of scaling approximation of $X(\xi)$ and $X^w_{j,k}$ are the coefficients of wavelet approximation of $X(\xi)$ such that

$$\begin{aligned} S_{J_0,k}(\xi(\theta) \in [a,b]) &= \phi^s_{J_0,k}(\wp(\xi)), \\ \mathscr{W}_{j,k}(\xi(\theta) \in [a,b]) &= \psi^w_{j,k}(\wp(\xi)). \end{aligned} \tag{3.2.4}$$

We can rewrite Eq. (3.2.3) as

$$X(\xi(\theta) \in [a,b]) = \sum_{k=-n_B+1}^{2^{J_0}-1} X^s_{J_0,k}\phi^s_{J_0,k}(\wp(\xi)) + \sum_{j=J_0}^{J}\sum_{k=-1}^{2^j-2} X^w_{j,k}\psi^w_{j,k}(\wp(\xi)). \tag{3.2.5}$$

Let us now denote by $\triangledown$ the set of index integers λ concatenating the scale index j and space index k such that $\triangledown \equiv \{\lambda : (\lambda_s = k+2, J_0 = 2, k = -1, 0, 1, 2, \ldots, 2^{J_0}-1) \cup (\lambda_w = 2^j + (k+3), j = J_0, J_0+1 \ldots, \infty, k = -1, \ldots, 2^j - 2)\}$. The resolution level will be denoted by $|\lambda|$. Therefore, the 1D wavelet expansion of $X(\xi(\theta))$ can be expressed as

$$X(\xi(\theta) \in [a,b]) = \sum_{\lambda \in \triangledown} X_\lambda W_\lambda(\xi(\theta)), \tag{3.2.6}$$

where

$$\begin{aligned} X_\lambda &= \int_0^1 X(\wp^{-1}(y))\phi^s_\lambda(y)dy \quad \text{when} \quad \lambda = 1, 2, \ldots, 2^{J_0}+1, \\ X_\lambda &= \int_0^1 X(\wp^{-1}(y))\psi^w_\lambda(y)dy \quad \text{when} \quad \lambda > 2^{J_0}. \end{aligned}$$

Also, we define the inner products of $\{W_\lambda(\xi(\theta)\}$ with respect to another set of dual scaling and wavelet functions $\{\tilde{W}_\lambda(\xi(\theta))\}$ such that when $\lambda_1, \lambda_2 = 1, 2, \ldots, 2^{J_0}+1$, we have

$$\begin{aligned} \int_a^b W_{\lambda_1}(\xi)\tilde{W}_{\lambda_2}(\xi)pdf(\xi)d\xi &= \int_a^b \phi_{j,k}(\wp(\xi))\tilde{\phi}_{j,l}(\wp(\xi))pdf(\xi)d\xi, \\ &= \int_a^b \phi_{j,k}(y)\tilde{\phi}_{j,l}(y)dy \quad (\because pdf(\xi)d\xi = d\wp(\xi) = dy, \text{ as } \wp(\xi) = y), \\ &= \delta_{kl}. \end{aligned} \tag{3.2.7}$$

Similarly, when $\lambda_1, \lambda_2 > 2^{J_0}$, we have

$$\begin{aligned} \int_a^b W_{\lambda_1}(\xi)\tilde{W}_{\lambda_2}(\xi)pdf(\xi)d\xi &= \int_a^b \psi_{j,k}(\wp(\xi))\tilde{\psi}_{l,m}(\wp(\xi))pdf(\xi)d\xi, \\ &= \int_a^b \psi_{j,k}(y)\tilde{\psi}_{l,m}(y)dy, \\ &= \delta_{jl}\delta_{km}. \end{aligned} \tag{3.2.8}$$

So, we have the semi-orthogonal WBe approximation of the stochastic process as

$$X(\xi) \approx \sum_{k=1}^{P} X_k W_k(\xi(\theta)). \tag{3.2.9}$$

After that, we move forward to the post processing step, i.e., we aim at extracting the statistical descriptors namely mean and variance using the WBe expansion (3.2.9). The mean is given by

$$\begin{aligned} \langle X \rangle &= \left\langle \sum_{i=1}^{P} X_i W_i(\xi), 1 \right\rangle, \\ &= \sum_{i=1}^{P} X_i \langle W_i(\xi), 1 \rangle. \end{aligned}$$

After defining the expectation, we can define another widely-used concept derived from moments, which are the centered moments. Variance is a measure of how much we should expect the output X to deviate from mean and it is given by

$$\begin{aligned} \sigma^2(X) &= \langle X^2 \rangle - (\langle X \rangle)^2, \\ &= \left\langle \sum_{i=1}^{P} \sum_{j=1}^{P} X_i X_j W_i(\xi) W_j(\xi), 1 \right\rangle - \left(\sum_{i=1}^{P} X_i \langle W_i(\xi), 1 \rangle \right)^2, \\ &= \sum_{i=1}^{P} \sum_{j=1}^{P} X_i X_j \langle W_i(\xi) W_j(\xi), 1 \rangle - \big(\langle X \rangle\big)^2. \end{aligned}$$

Therefore, S.D. is given by

$$S.D. = \sigma(X) = \sqrt{\sum_{i=1}^{P} \sum_{j=1}^{P} X_i X_j \langle W_i(\xi) W_j(\xi), 1 \rangle - \big(\langle X \rangle\big)^2}.$$

We now apply the procedure defined in section 3.1 and section 3.2 on three test problems in order to check the efficiency of our method.

3.3 Numerical Results

Firstly, we apply the WBe decomposition to a simple model problem with non-linear term that involves discontinuous dependence of the process on the random data. Then, we apply the same decomposition on stochastic linear oscillator and stochastic Kraichan-Orszag (K-

O) problem which consists of system of stochastic ODEs.

3.3.1 Second order stochastic differential equation

We consider a deterministic second order ODE:

$$\frac{d^2x}{dt^2} + f\frac{dx}{dt} = -\frac{dH}{dx}, \tag{3.3.1}$$

with parameters $\frac{dH}{dx}$ and $f > 0$. As it is a second order ODE, therefore we need two ICs as follows:

$$\begin{aligned} x(0) &= x_0, \\ v(t=0) \equiv \frac{dx}{dt}(t=0) &= v_0. \end{aligned} \tag{3.3.2}$$

The system can be interpreted as the governing equation for a particle moving under the influence of a friction force and of a potential field. We have,

$$H(x) = \frac{35}{8}x^4 - \frac{15}{4}x^2, \tag{3.3.3}$$

such that the differential equation has two stable fixed points

$$x = \sqrt{\frac{15}{35}}, \quad x = -\sqrt{\frac{15}{35}}, \tag{3.3.4}$$

and an unstable fixed point at $x = 0$.

We construct the stochastic variant of the aforementioned ODE by considering an uncertain initial position x_0. While performing the computations, the initial position is supposed to be uniformly distributed over the interval $[x_1, x_2]$, i.e.,

$$\begin{aligned} pdf(x_0 \in [x_1, x_2]) &= \frac{1}{|x_2 - x_1|}, \\ \text{and} \quad \int_{x_1}^{x_2} pdf(x)dx &= 1. \end{aligned} \tag{3.3.5}$$

Therefore, the stochastic ICs are as follows:

$$X(t=0,\xi) = X_0 + \Delta X\xi, \tag{3.3.6}$$

$$\left.\frac{dX}{dt}\right|_{t=0} = 0. \tag{3.3.7}$$

where

$$X_0 = \frac{x_1 + x_2}{2}, \Delta X = \frac{\mid x_1 - x_2 \mid}{2},$$

and $X(t, \xi)$ describes the stochastic response of the system. Also, ξ is uniformly distributed over $[-1, 1]$ along with $pdf(\xi) = 1/2$. Moreover, we can formulate the stochastic system as

$$\begin{aligned} \frac{d^2X}{dt^2} + f\frac{dX}{dt} &= -\frac{dH}{dx}, \\ \frac{d^2X}{dt^2} + f\frac{dX}{dt} &= -\frac{35}{2}X^3 + \frac{15}{2}X. \end{aligned} \tag{3.3.8}$$

3.3.1.1 Solution Method

Here, we apply the simulation method, i.e., wavelet based spline PCE for integrating the stochastic formulation. The truncated WBe expansion of the solution process is given as

$$X(t, \xi(\theta)) \approx \tilde{X}(t, \xi) = \sum_{k=1}^{P} X_k(t) W_k(\xi). \tag{3.3.9}$$

We now derive the governing equations for the wavelet coefficients X_k by inserting the truncated wavelet expansion into Eq. (3.3.8). Also, we keep the cubic term untouched and will explain it separately so as to get the clarity about the implementation of the method.

$$\frac{d^2}{dt^2}\left(\sum_{i=1}^{P} X_i(t) W_i(\xi)\right) + f\frac{d}{dt}\left(\sum_{i=1}^{P} X_i(t) W_i(\xi)\right) = -\frac{35}{2}X^3 + \frac{15}{2}\sum_{i=1}^{P} X_i(t) W_i(\xi) \tag{3.3.10}$$

Further, we use stochastic Galerkin method to perform projections onto the wavelet basis w.r.t dual $\tilde{W}_l(\xi)$ as follows:

$$\frac{d^2}{dt^2}\sum_{i=1}^{P} X_i(t)\langle W_i(\xi)\tilde{W}_l(\xi)\rangle + f\frac{d}{dt}\sum_{i=1}^{P} X_i(t)\langle W_i(\xi)\tilde{W}_l(\xi)\rangle = -\frac{35}{2}X^3 + \frac{15}{2}\sum_{i=1}^{P} X_i(t)\langle W_i(\xi)\tilde{W}_l(\xi)\rangle$$

$$\sum_{i=1}^{P}\frac{d^2}{dt^2} X_i(t)\langle W_i(\xi)\tilde{W}_l(\xi)\rangle + \sum_{i=1}^{P} f\frac{d}{dt} X_i(t)\langle W_i(\xi)\tilde{W}_l(\xi)\rangle = -\frac{35}{2}X^3 + \frac{15}{2}\sum_{i=1}^{P} X_i(t)\langle W_i(\xi)\tilde{W}_l(\xi)\rangle, \tag{3.3.11}$$

Similarly, we derive the ICs by applying Galerkin approach for the individual modes:

$$X_l(t=0) = \langle (X_0 + \Delta X\xi)\ \tilde{W}_l(\xi)\rangle \quad \text{for} \quad l = 1, \ldots, P. \tag{3.3.12}$$

We derive the second ICs in the same way

$$\begin{aligned} \frac{d}{dt}\sum_{i=1}^{P}\left(X_i(t=0)\langle W_i(\xi)\tilde{W}_l(\xi)\rangle\right) = 0, \\ \left.\frac{dX_l}{dt}\right|_{t=0} = 0 \quad \text{for} \quad l = 1, 2, \ldots, P. \end{aligned} \tag{3.3.13}$$

We can easily integrate Eq. (3.3.11) once the cubic term is determined. To do the same, there are two approaches available in literature. These are

1) Galerkin projection,

2) Pseudo-spectral approximation.

For explaining the two approaches, we need to understand "multiplicative tensor ($\mathscr{C}_{ijk}$)" and the "triple product tensor (T_{ijkl})"

$$\mathscr{C}_{ijk} = \langle W_i(\xi)W_j(\xi)\tilde{W}_k(\xi)\rangle, \tag{3.3.14}$$

$$T_{ijkl} = \langle W_i(\xi)W_j(\xi)W_k(\xi)\tilde{W}_l(\xi)\rangle. \tag{3.3.15}$$

The first approach i.e., Galerkin approach involves the triple product tensor for computing the cubic term X^3 through convolution.

$$X^3 = \sum_{i=1}^{P}\sum_{j=1}^{P}\sum_{k=1}^{P} X_i(t)X_j(t)X_k(t)T_{ijkl}, \tag{3.3.16}$$

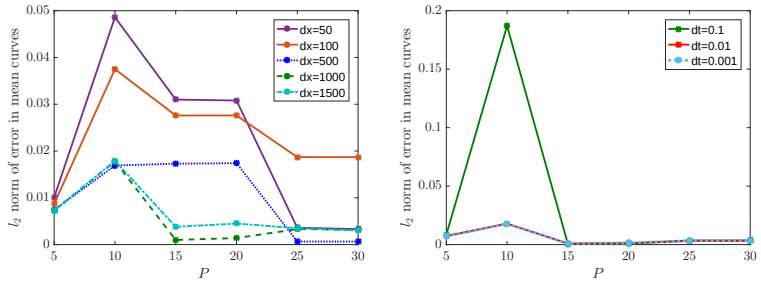
where $l = 1, 2, \ldots, P$.

The second approach i.e., pseudo-spectral approach involves the multiplicative tensor for computing the cubic term X^3 by repeated application of binary multiplication operator as follows:

$$X^2 = \sum_{i=1}^{P}\sum_{j=1}^{P} X_i(t)X_j(t)\mathscr{C}_{ijk}, \tag{3.3.17}$$

$$X^3 = (X)(X^2) = \sum_{i=1}^{P}\sum_{j=1}^{P} X_i(t)X_j^2(t)\mathscr{C}_{ijk}. \tag{3.3.18}$$

We evaluate the tensor $\mathscr{C}_{ijk}$ and T_{ijkl} in the preprocessing step so that we can later use them in the simulations. The time integration of Eq. (3.3.11) is done using RK4 method. To select the parameters such as time step dt and the number of grid points dx used to discretize the domain, we have performed the SA of the method with respect to changes in dt and dx. Fig. 3.3a shows the effect of dx on the error in means of the numerical and analytical predictions with change in value of P. It can be observed that as the value of dx changes from 1000 to 1500, there is an insignificant change in the mean curve whereas before $dx = 1000$, there are substantial changes with change in the value of dx. Therefore, we have chosen $dx = 1000$ for all the further calculations. Fig. 3.3b displays the effect of dt on the mean of the solution with respect to P. From this figure, it can be seen that a further decrement in dt from 0.001 has negligible effect on the mean curves. Therefore, we have chosen $dt = .001$.



(a) (b)

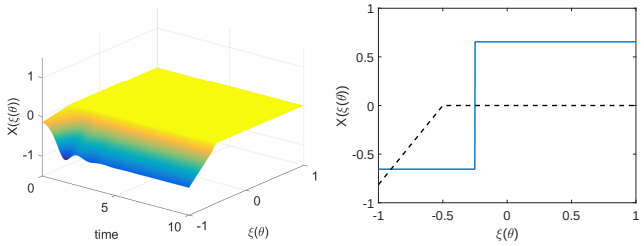
Figure 3.3: l_2 norm of error in mean curves of first model problem (Galerkin Approach). a) The time step is fixed $dt = 0.001$ and number of grid points (dx) are changed b) dx is fixed as $dx = 1000$ and the time step is changed.

3.3.1.2 Results and Discussion

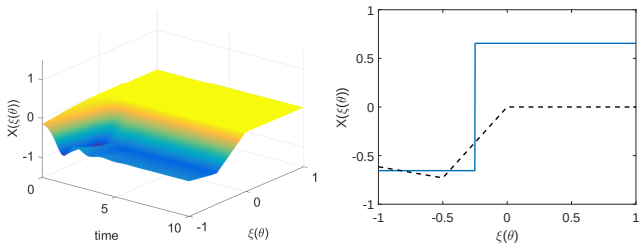
The WBe scheme is now applied on the above mentioned problem with initial stochastic conditions defined by $X_0 = 0.05$ and $\Delta X = 0.2$. To achieve the steady solution in a very short time period, we assign a comparatively bigger value to friction coefficient i.e., $f = 2$. For the present conditions, the analytical prediction of the steady-state is given by

$$\begin{cases} X(t \to \infty, \xi) = -\sqrt{\frac{15}{35}}, & \xi < -0.25, \\ X(t \to \infty, \xi) = \sqrt{\frac{15}{35}}, & \xi > 0.25. \end{cases} \tag{3.3.19}$$

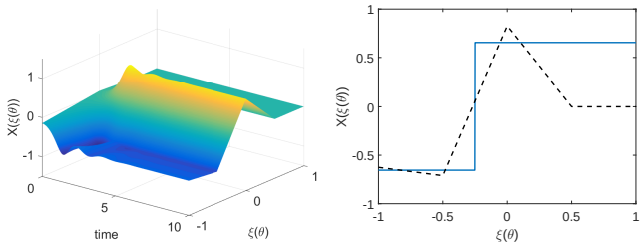
Also, we have the analytical predictions [55] of mean and S.D. which comes out to be $\langle X \rangle = 0.16366$ and $\sigma(X) = 0.63418$. We obtain the results for increasing number P of



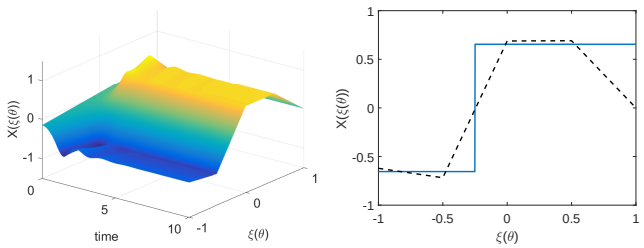
(a) $P = 1$ (b) $P = 1$



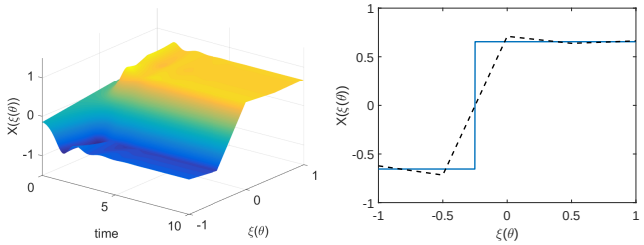
(c) $P = 2$ (d) $P = 2$



(e) $P = 3$ (f) $P = 3$



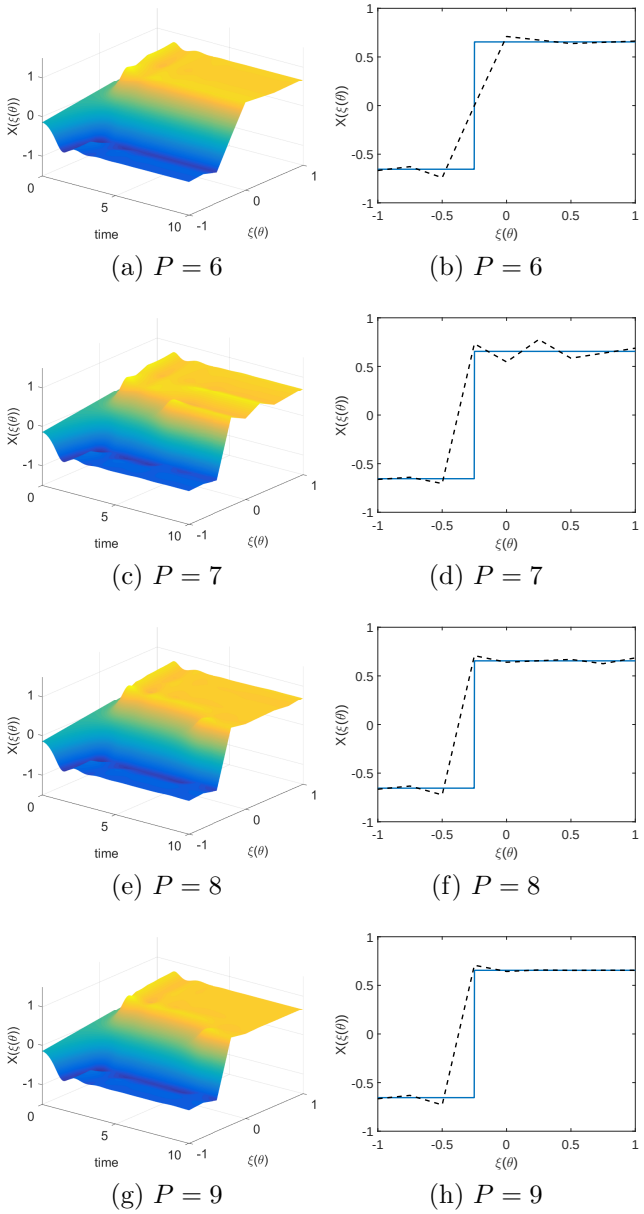
(g) $P = 4$ (h) $P = 4$



(i) $P = 5$ (j) $P = 5$

Figure 3.4: Galerkin linear WBe solutions for the test problem 1 at $t = 10$ with 1000 spatial grid points and $dt = 0.001$. The results are acquired for $P = 1, 2, 3, 4, 5$ arranged from top to bottom. The first column shows the 3D plots of solution $X(t, \xi)$ and the second column plots the solution $X(t = 10)$.

resolution levels. Fig. 3.4 shows the 3D plots of evolution of $X(t,\xi)$ and 2D plots of $X(t)$ with $0 \leq t \leq 10$ by using the first five linear scaling B-spline basis functions at level $\jmath = 2$ for $P = 1,2,3,4,5$. Fig. 3.5 displays the $3D$ plots and 2D plots of solution $X(t,\xi)$ at $t = 10$ for $P = 6,7,8,9$, i.e., using the first linear B-spline wavelet functions at $\jmath = 2$. Fig. 3.6 presents the next linear B-spline wavelet functions with increasing resolution level, i.e., $P = 32,40,45,50,55$. For $P < 4$, the WBe scheme doesn't correctly



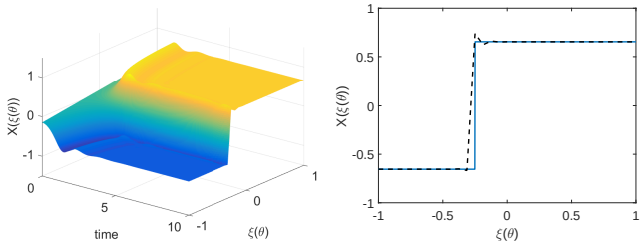
(a) $P = 6$ (b) $P = 6$

(c) $P = 7$ (d) $P = 7$

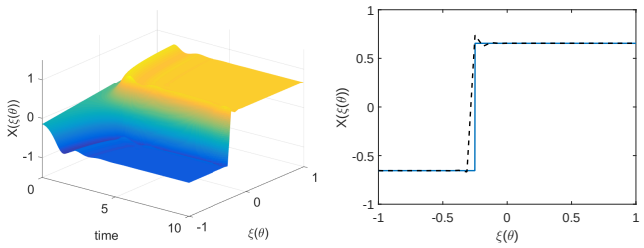
(e) $P = 8$ (f) $P = 8$

(g) $P = 9$ (h) $P = 9$

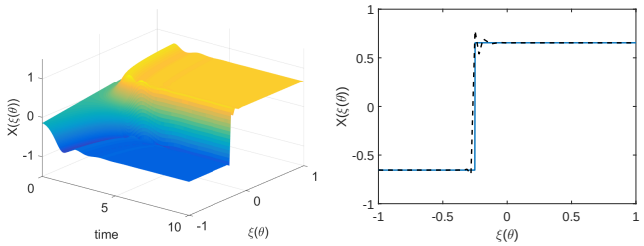
Figure 3.5: Galerkin linear WBe solutions for the test problem 1 at $t = 10$ with 1000 spatial grid points and $dt = 0.001$. The results are obtained for $P = 6,7,8,9$ arranged from top to bottom. The first column shows the 3D plots of solution $X(t,\xi)$ and the second column plots the solution $X(t = 10)$.



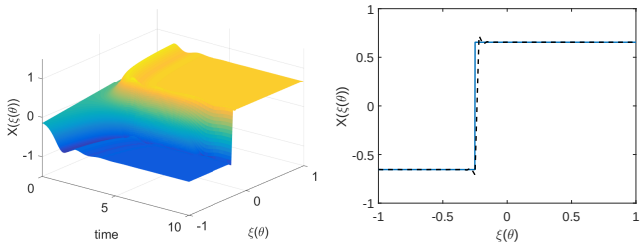
(a) $P = 32$ (b) $P = 32$



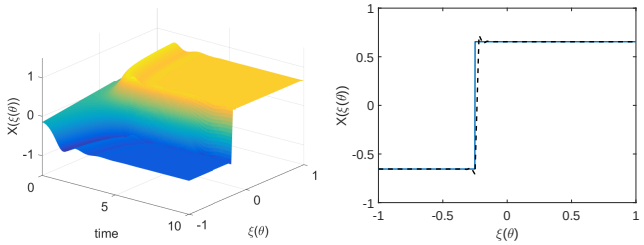
(c) $P = 40$ (d) $P = 40$



(e) $P = 45$ (f) $P = 45$

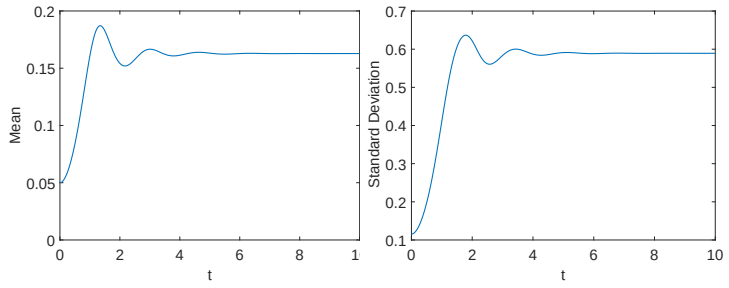


(g) $P = 50$ (h) $P = 50$

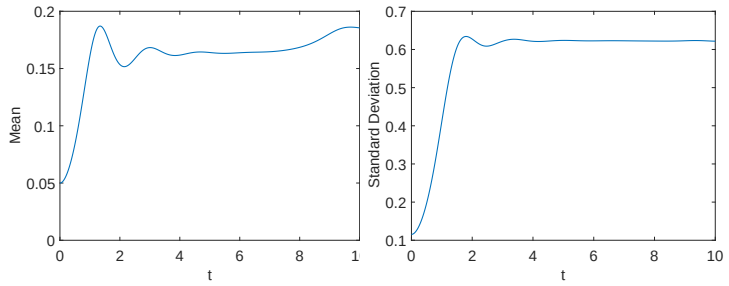


(i) $P = 55$ (j) $P = 55$

Figure 3.6: Galerkin linear WBe solutions for the model problem at $t = 10$ with 1000 spatial grid points and $dt = 0.001$. The results are shown for PCE orders $P = 32, 40, 45, 50, 55$ arranged from top to bottom. The first column shows the 3D plots of solution $X(t, \xi)$ and the second column plots the solution $X(t = 10)$.



(a) Mean at $P = 5$ (b) S.D. at $P = 5$



(c) Mean at $P = 30$ (d) S.D. at $P = 30$

Figure 3.7: Mean and S.D. for the first test problem at $t = 10$ with 1000 spatial grid points and $dt = 0.001$ by using Galerkin approximation. The expansion orders are taken as $P = 5$ and $P = 30$.

captures the bifurcation separating the trajectories but for $P > 4$ WBe scheme captures it correctly. Also, we see the transition for the first time at $P = 5$. Figs. 3.4, 3.5, 3.6 shows the Galerkin WBe approximate solution of our problem at $t = 10$ whereas the Figs. 3.8, 3.9, 3.10 illustrates WBe approximate solution of our problem using the pseudo-spectral approximation at $t = 10$. In these figures, the analytical solution is shown by solid line whereas the linear B-spline numerical solution is shown by dotted line. We can mark from Fig. 3.4 that as P increases the approximate solution is moving close to the analytical predicted solution. Furthermore, as we include the wavelet functions, the approximation is quite satisfactory as clear from Figs. 3.5 and 3.6.

Figs. 3.8, 3.9, 3.10 are then compared with Figs. 3.4, 3.5, 3.6 and we notice sharp peaks near the discontinuities in case of pseudo-spectral approach, i.e., it highlights Gibbs phenomenon. In Fig. 3.11, we can see that pseudo-spectral approach shows variations near the discontinuities for $P = 120$ whereas Galerkin approach approximates the analytical solution exactly which is numerically expected. So, we conclude that the Galerkin approach is more efficient as compared to pseudo-spectral approach. Also, from these figures, we analyze that the wavelet based expansions show more accurate results as compared to scaling functions. The results become more and more accurate as we increase the order of PCE. The same test problem has been solved using Wiener-Legendre scheme in paper entitled

"Uncertainty propagation using Wiener-Haar expansions [55]" and we have compared our results. It is found that WBe performs better than Wiener-Legendre scheme. In Fig. 3.7 and Fig. 3.12, we have plotted mean and S.D. for $P = 5$ and $P = 30$ by using Galerkin approximation and pseudo-spectral approximation respectively. We have also shown the convergence of mean and S.D. with respect to the order of PCE for Galerkin (first row) and pseudo-spectral approach (second row) in Fig. 3.13. Hence, from here also, we observe that it is convenient to use Galerkin approximation than pseudo-spectral approach.

Fig. 3.14b shows variation of the error in means of the numerical and analytical predictions with respect to P. It can be observed from the graph that after $P = 25$, there is negligible change in error. We have also calculated the computational complexity and time by evaluating the CPU time for increasing values of P. As P is increasing, the CPU time is getting directly proportional to the square of P as shown in Figure 3.14b. Thus, the computational complexity is $O(P^2)$.

3.3.2 Linear oscillator Problem-System of differential equations

Consider the equations of motion of linear oscillator defined in two dimensions with random frequency [2]. Basically, we have the following equations

$$\begin{aligned} \frac{du(t,\xi)}{dt} &= \quad v(t,\xi), \\ \frac{dv(t,\xi)}{dt} &= -qu(t,\xi). \end{aligned} \tag{3.3.20}$$

such that u is the position, $\dot{v} = u$ is the impulse and $q > 0$. Also, the deterministic ICs are given by

$$u(t=0,\xi) = \quad 1, \quad v(t=0,\xi) = \quad 0, \tag{3.3.21}$$

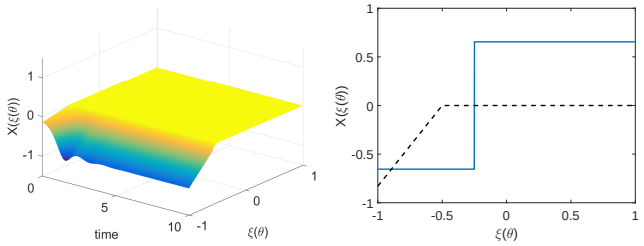
and random frequency

$$q(\xi) = q_0 + q_1\xi, \tag{3.3.22}$$

where

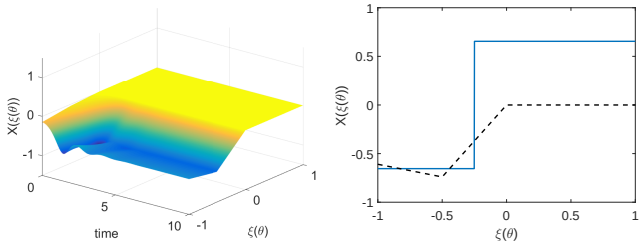
$$q_0 = (2\pi)^2, q_1 = 0.2(2\pi)^2 \tag{3.3.23}$$

and ξ is uniformly distributed within the real interval $(-1, 1)$, i.e., $\xi \sim U(-1, 1)$.



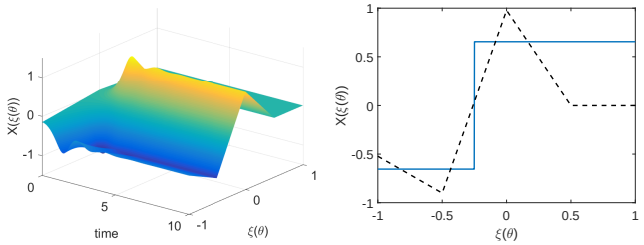
(a) $P = 1$

(b) $P = 1$



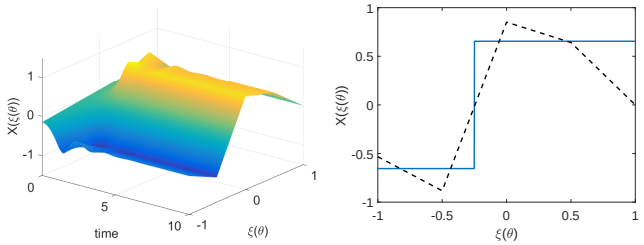
(c) $P = 2$

(d) $P = 2$



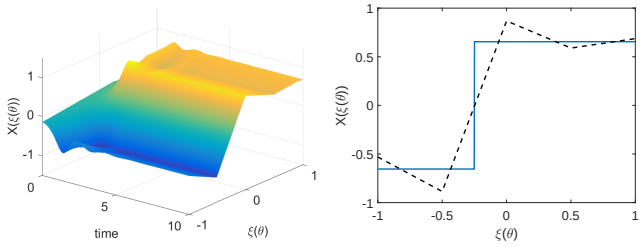
(e) $P = 3$

(f) $P = 3$



(g) $P = 4$

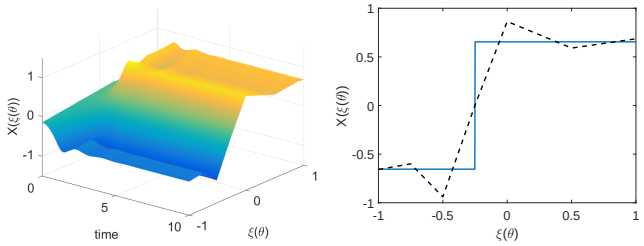
(h) $P = 4$



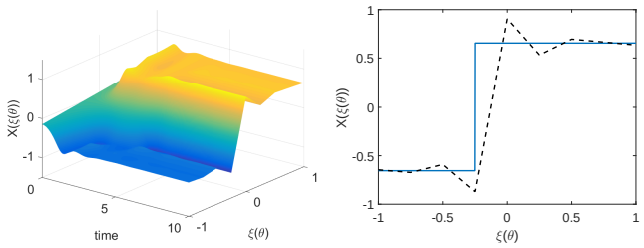
(i) $P = 5$

(j) $P = 5$

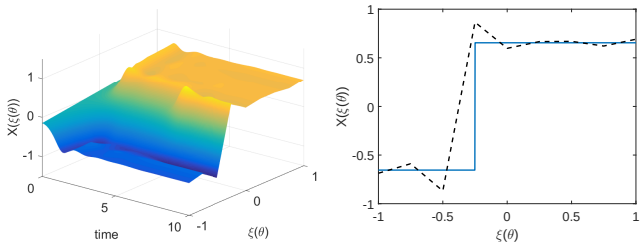
Figure 3.8: Pseudo-spectral linear WBe solutions for the model problem at $t = 10$ with 1000 spatial grid points and $dt = 0.001$. Results are captured for $P = 1, 2, 3, 4, 5$ arranged from top to bottom. The first column shows the 3D plots of solution $X(t, \xi)$ and the second column plots the solution $X(t = 10)$.



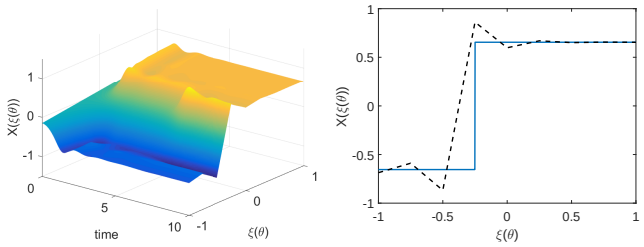
(a) $P = 6$ (b) $P = 6$



(c) $P = 7$ (d) $P = 7$



(e) $P = 8$ (f) $P = 8$



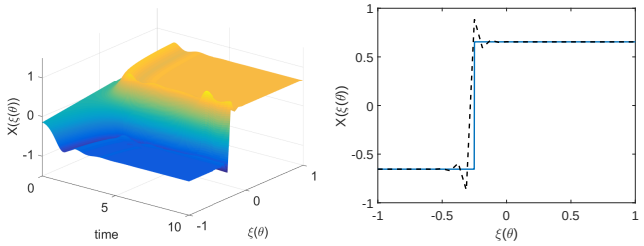
(g) $P = 9$ (h) $P = 9$

Figure 3.9: Pseudo-spectral linear WBe solutions for the model problem at $t = 10$ with 1000 spatial grid points and $dt = 0.001$. Results are captured for PCE orders $P = 6, 7, 8, 9$ arranged from top to bottom. The first column shows the 3D plots of $X(t, \xi)$ and the second column plots the solution $X(t = 10)$.

3.3.2.1 Solution Method

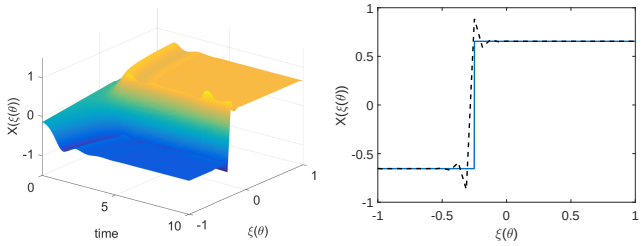
We write down the truncated WBe expansions of u, v, q as follows

$$u(t,\xi) = \sum_{i=1}^{P} u_i(t) W_i(\xi), \quad v(t,\xi) = \sum_{i=1}^{P} v_i(t) W_i(\xi), \quad q(\xi) = \sum_{i=1}^{P} q_i(t) W_i(\xi). \tag{3.3.24}$$



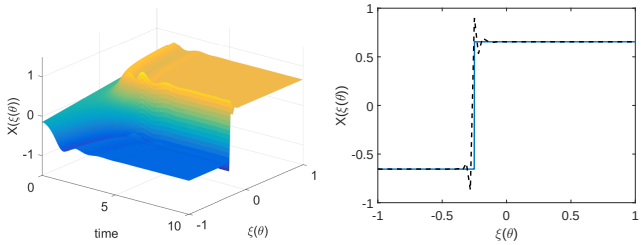
(a) $P = 32$

(b) $P = 32$



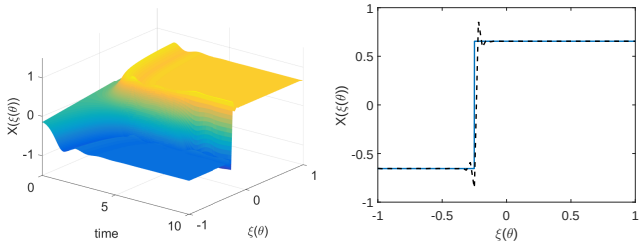
(c) $P = 40$

(d) $P = 40$



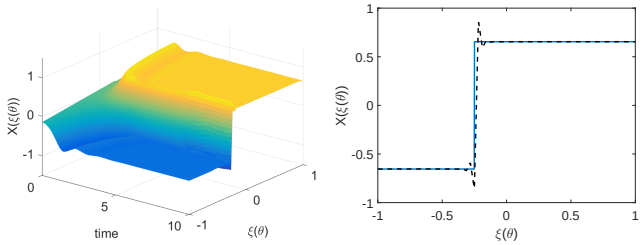
(e) $P = 45$

(f) $P = 45$



(g) $P = 50$

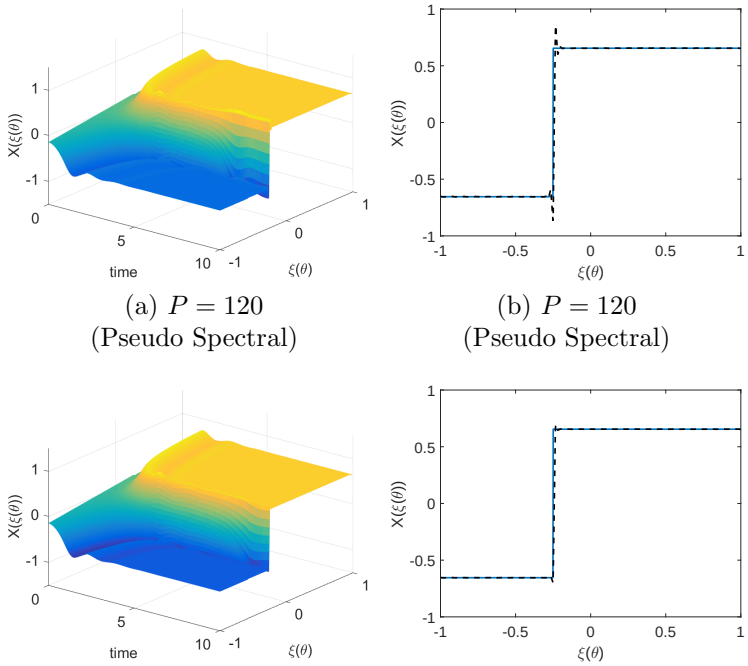
(h) $P = 50$



(i) $P = 55$

(j) $P = 55$

Figure 3.10: Pseudo-Spectral linear WBe solutions for the model problem at $t = 10$ with 1000 spatial grid points and $dt = 0.001$. Results are obtained for $P = 32, 40, 45, 50, 55$ arranged from top to bottom. The first column highlights the evolution of $X(t, \xi)$ and the second column plots the solution $X(t = 10)$.



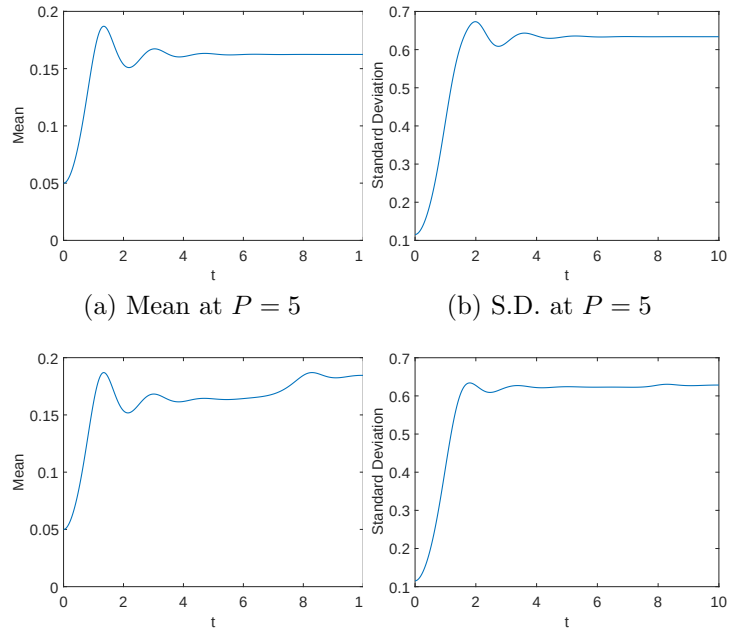
(a) $P = 120$ (Pseudo Spectral)

(b) $P = 120$ (Pseudo Spectral)

(c) $P = 120$ (Galerkin)

(d) $P = 120$ (Galerkin)

Figure 3.11: Pseudo-Spectral and Galerkin linear WBe solutions for the model problem at $t = 10$ with 1000 spatial grid points and $dt = 0.001$ for expansion order $P = 120$.



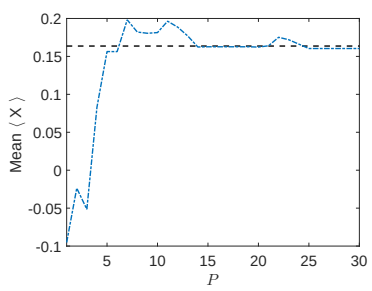
(a) Mean at $P = 5$

(b) S.D. at $P = 5$

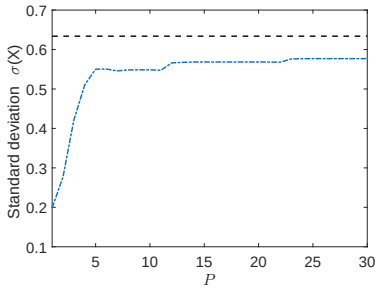
(c) Mean at $P = 30$

(d) S.D. at $P = 30$

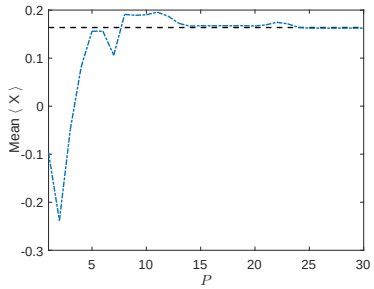
Figure 3.12: Mean and S.D. for the first test problem at $t = 10$ with 1000 spatial grid points and $dt = 0.001$ by using pseudo-spectral approximation. The expansion orders are taken as $P = 5$ and $P = 30$.



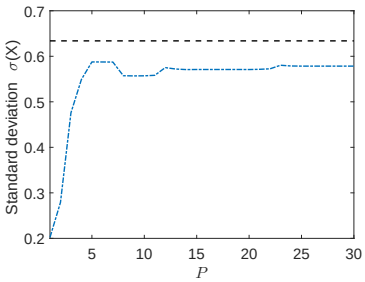
(a) P vs Mean (Galerkin)



(b) P vs S.D. (Galerkin)

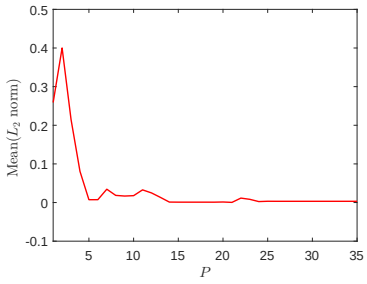


(c) P vs Mean (Pseudo-Spectral)

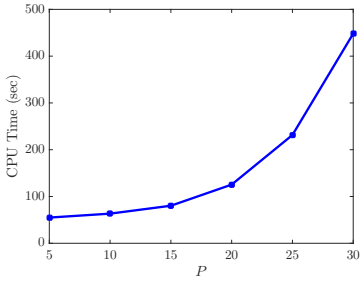


(d) P vs S.D. (Pseudo-spectral)

Figure 3.13: Convergence of mean and S.D. with respect to order of PCE for the first test problem using linear WBe expansion. Here we have plotted results for Galerkin approximation in part a) and b) whereas the pseudo-spectral approximation results are shown in parts c) and d). The analytical predictions of mean and S.D. comes out to be $\langle X \rangle = 0.16366$ and $\sigma(X) = 0.63418$.



(a)



(b)

Figure 3.14: a) l_2 norm of mean of first model problem (Galerkin Approach) b)P vs CPU time for the first test problem (Galerkin).

We now derive the governing equations by inserting the truncated wavelet expansions (3.3.24) into Eq. (3.3.20) and then performing projections w.r.t $\tilde{W}_k(\xi)$ as follows

$$\frac{d}{dt}\left(\sum_{i=1}^{P} u_i(t)W_i(\xi)\right) \;=\; \sum_{i=1}^{P} v_i(t)W_i(\xi), \tag{3.3.25}$$

$$\begin{aligned}\sum_{i=1}^{P}\frac{du_i(t)}{dt}\langle W_i(\xi)\tilde{W}_k(\xi)\rangle &= \sum_{i=1}^{P}v_i(t)\langle W_i(\xi)\tilde{W}_k(\xi)\rangle,\\ \frac{du_k(t)}{dt} &= v_k(t), \quad \text{where} \quad k=1,2,\ldots,P. \end{aligned} \tag{3.3.26}$$

The second equation is solved in the same way as follows:

$$\begin{aligned}\frac{d}{dt}\Big(\sum_{i=1}^{P}v_i(t)W_i(\xi)\Big) &= -\sum_{i=1}^{P}q_i(t)W_i(\xi)\sum_{j=1}^{P}u_j(t)W_j(\xi),\\ \sum_{i=1}^{P}\frac{d}{dt}\big(v_i(t)W_i(\xi)\big) &= -\sum_{i=1}^{P}\sum_{j=1}^{P}q_i(t)u_j(t)W_i(\xi)W_j(\xi).\end{aligned}$$

Taking inner product w.r.t $\tilde{W}_k(\xi)$, we get

$$\begin{aligned}\sum_{i=1}^{P}\frac{d}{dt}v_i(t)\langle W_i(\xi)\tilde{W}_k(\xi)\rangle &= -\sum_{i=1}^{P}\sum_{j=1}^{P}q_i(t)u_j(t)\langle W_i(\xi)W_j(\xi)\tilde{W}_k(\xi)\rangle,\\ \frac{d}{dt}v_k(t) &= -\sum_{i=1}^{P}\sum_{j=1}^{P}q_i(t)u_j(t)\mathscr{C}_{ijk} \text{ where } k=1,2,\ldots,P.\end{aligned}$$

Similarly, we have ICs as

$$\begin{aligned}\sum_{i=1}^{P}u_i(t=0)W_i(\xi)=1,\\ \sum_{i=1}^{P}v_i(t=0)W_i(\xi)=0.\end{aligned}$$

Taking inner product w.r.t $\tilde{W}_k(\xi)$, we get

$$\begin{aligned}\sum_{i=1}^{P}u_i(t=0)\langle W_i(\xi)\tilde{W}_k(\xi)\rangle &= \langle 1,\tilde{W}_k(\xi)\rangle,\\ u_k(t=0) &= \langle 1,\tilde{W}_k(\xi)\rangle \quad \text{where} \quad k=1,2,\ldots,P. \end{aligned} \tag{3.3.27}$$

$$\begin{aligned}\sum_{i=1}^{P}v_i(t=0)\langle W_i(\xi)\tilde{W}_k(\xi)\rangle &= \langle 0,\tilde{W}_k(\xi)\rangle,\\ v_k(t=0) &= \langle 0,\tilde{W}_k(\xi)\rangle \quad \text{where} \quad k=1,2,\ldots,P. \end{aligned} \tag{3.3.28}$$

3.3.2.2 Results and Discussion

In simulations, RK4 method is used for time integration by taking small $dt = 0.001$. In Fig. 3.15, we have plotted mean of state variables u and v with respect to time $t \in [0, 50]$ for different expansion orders $P = 5, 10, 15, 20, 25$. As we increase the order of PCE, the oscillations of state variables u and v move faster towards zero, i.e., they tend to diminish faster. In Fig. 3.16, we have plotted the trajectories by taking mean of u and v in x-axis and y-axis respectively on the left side and plotted the variance of u and v by taking it on x-axis and y-axis respectively on the right side. We observe that as we increase P, the trajectories are more seen at position zero whereas for lower order of PCE such as $P = 5$, we can see that the mean of u and v is scattered almost everywhere. We also notice that the variance is scattered almost everywhere for lower order $P = 5$ whereas it is scattered in a proper shape as we increase the value of P.

3.3.3 Stochastic Kraichan-Orszag Problem

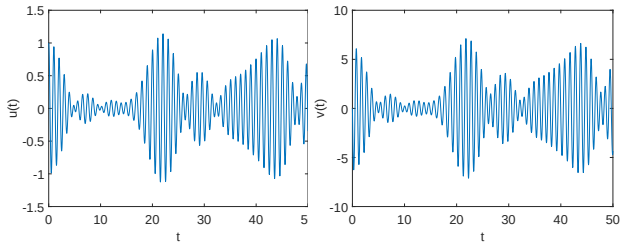
We now study the well-known K-O three mode problem and investigate the long time numerical behavior of UQ schemes. It was first introduced by R. H. Kraichnan [217] and later used by S. A. Orszag [77] for studying uncertainties of Gaussian nature in turbulence. It is basically derived from simplified inviscid Navier-Stokes equations and consist of three nonlinear coupled ODEs, given as

$$\begin{aligned} \frac{du(t,\xi)}{dt} &= \quad v(t,\xi)w(t,\xi), \\ \frac{dv(t,\xi)}{dt} &= \quad u(t,\xi)w(t,\xi), \\ \frac{dw(t,\xi)}{dt} &= -2u(t,\xi)v(t,\xi). \end{aligned} \tag{3.3.29}$$

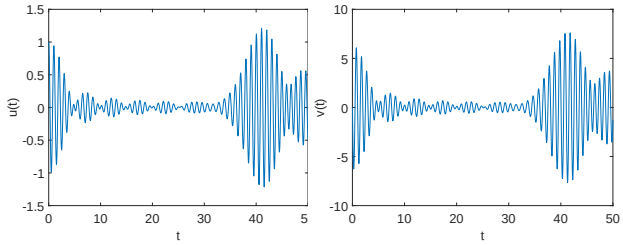
Uncertainty in the system is introduced through ICs. Each of the three state variables $u(t,\xi), v(t,\xi)$ and $w(t,\xi)$, may have an uncertain IC. The number of state variables having initial uncertainty determines the stochastic dimension of the system. The critical range of α is known [113] to be $(0.9, 1)$ as it is highly dependent on the ICs. We only consider the case with initial uncertainty in just one state variable represented as

$$u(0) = \alpha + 0.01\xi, \quad v(0) = 1, \quad w(0) = 1. \tag{3.3.30}$$

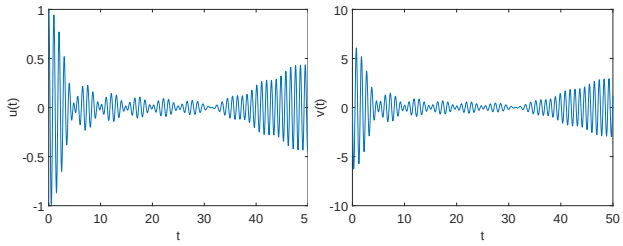
where ξ is uniformly distributed RV $\xi \sim U(-1, 1)$.



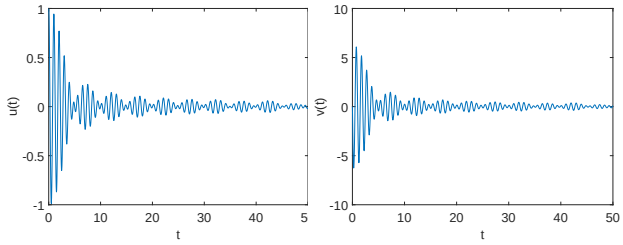
(a) $P = 5$ (b) $P = 5$



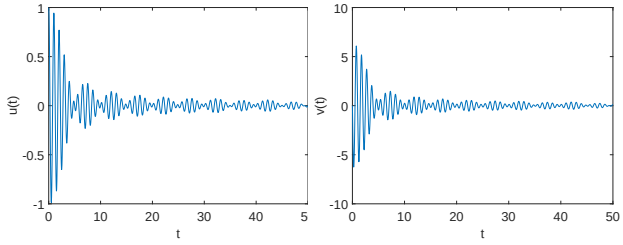
(c) $P = 10$ (d) $P = 10$



(e) $P = 15$ (f) $P = 15$

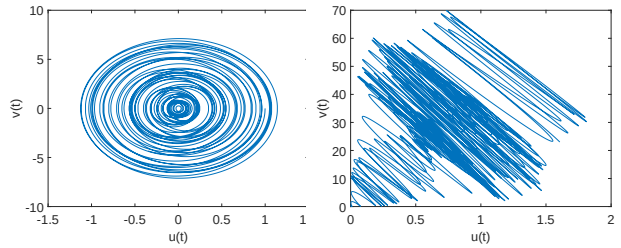


(g) $P = 20$ (h) $P = 20$

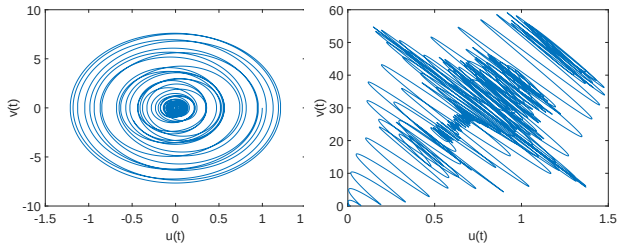


(i) $P = 25$ (j) $P = 25$

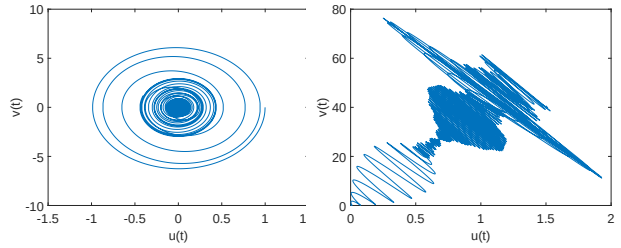
Figure 3.15: Mean of u and v with respect to time are plotted for the linear oscillator problem at $t = [0, 50]$ with 1000 spatial grid points and $dt = 0.001$. Results are obtained for expansion orders $P = 5, 10, 15, 20, 25$ arranged from top to bottom.



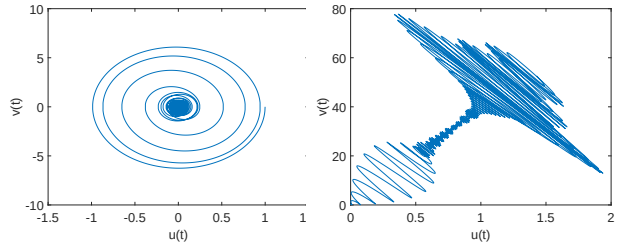
(a) $P = 5$ (b) $P = 5$



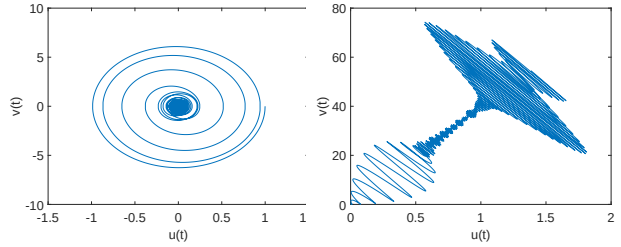
(c) $P = 10$ (d) $P = 10$



(e) $P = 15$ (f) $P = 15$



(g) $P = 20$ (h) $P = 20$



(i) $P = 25$ (j) $P = 25$

Figure 3.16: (First Column) Evolution of trajectories corresponding to mean (Left) and variance (Right) in the state space of the Galerkin approximation of solution $u(t)$ for different expansion orders $P = 5, 10, 15, 20, 25$ arranged from top to bottom for the linear oscillator problem at $t = [0, 50]$ with 1000 spatial grid points and $dt = 0.001$.

3.3.3.1 Solution method

Firstly, we write down the truncated linear WBe expansion of u, v, w as follows

$$u(t,\xi) = \sum_{i=1}^{P} u_i(t)W_i(\xi), \quad v(t,\xi) = \sum_{i=1}^{P} v_i(t)W_i(\xi), \quad w(t,\xi) = \sum_{i=1}^{P} w_i(t)W_i(\xi). \tag{3.3.31}$$

and then apply them in Eq. (3.3.29)

$$\begin{aligned}
\frac{d}{dt}\sum_{i=1}^{P} u_i(t)W_i(\xi) &= \sum_{i=1}^{P} v_i(t)W_i(\xi)\sum_{j=1}^{P} w_j(t)W_j(\xi), \\
\sum_{i=1}^{P}\frac{d}{dt}u_i(t)W_i(\xi) &= \sum_{i=1}^{P}\sum_{j=1}^{P} v_i(t)w_j(t)W_i(\xi)W_j(\xi).
\end{aligned}$$

Now, we use the stochastic Galerkin method to perform projections w.r.t $\tilde{W}_k(\xi)$ in the following way

$$\begin{aligned}
\sum_{i=1}^{P}\frac{d}{dt}u_i(t)\langle W_i(\xi)\tilde{W}_k(\xi)\rangle &= \sum_{i=1}^{P}\sum_{j=1}^{P} v_i(t)w_j(t)\langle W_i(\xi)W_j(\xi)\tilde{W}_k(\xi)\rangle, \\
\sum_{i=1}^{P}\frac{d}{dt}u_i(t)\langle W_i(\xi)\tilde{W}_k(\xi)\rangle &= \sum_{i=1}^{P}\sum_{j=1}^{P} v_i(t)w_j(t)\mathscr{C}_{ijk}, \\
\frac{d}{dt}u_k(t) = \sum_{i=1}^{P}\sum_{j=1}^{P} v_i(t)w_j(t)\mathscr{C}_{ijk}, &\quad \text{where} \quad k = 1, 2, \ldots, P.
\end{aligned} \tag{3.3.32}$$

Similarly, we have second and the third equation

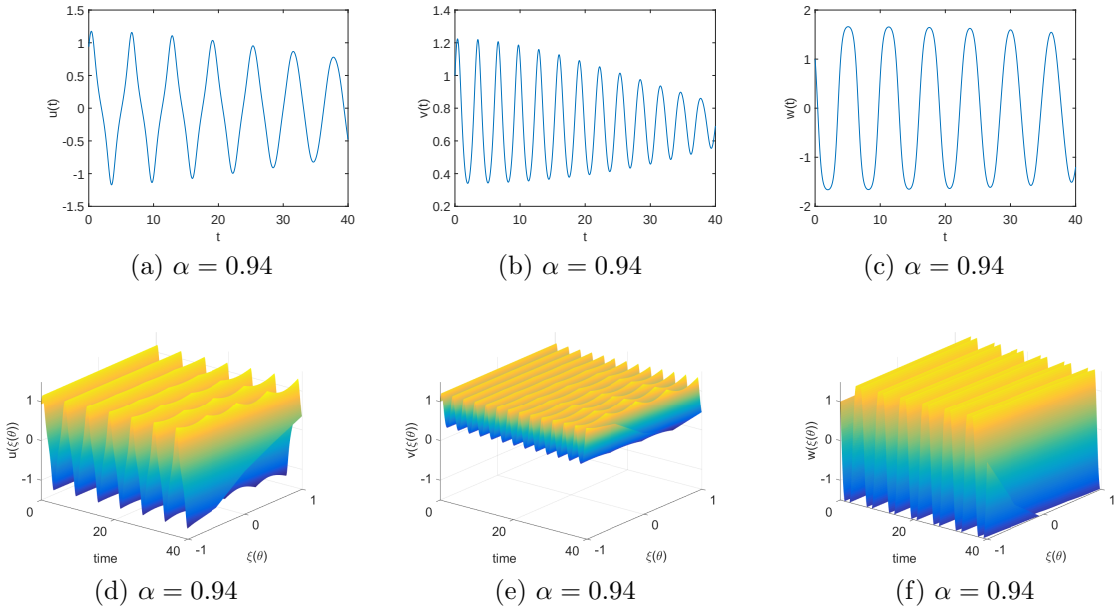
$$\frac{d}{dt}v_k(t) = \sum_{i=1}^{P}\sum_{j=1}^{P} u_i(t)w_j(t)\mathscr{C}_{ijk}, \quad \text{where} \quad k = 1, 2, \ldots, P. \tag{3.3.33}$$

$$\frac{d}{dt}w_k(t) = -2\sum_{i=1}^{P}\sum_{j=1}^{P} u_i(t)v_j(t)\mathscr{C}_{ijk}, \quad \text{where} \quad k = 1, 2, \ldots, P. \tag{3.3.34}$$

and the ICs after stochastic Galerkin projection turns out to be

$$\begin{aligned}
u_k(t=0) &= \langle(\alpha + 0.01\xi), \tilde{W}_k(\xi)\rangle, \\
v_k(t=0) &= \langle 1,\ \tilde{W}_k(\xi)\rangle, \\
w_k(t=0) &= \langle 1,\ \tilde{W}_k(\xi)\rangle,
\end{aligned} \tag{3.3.35}$$

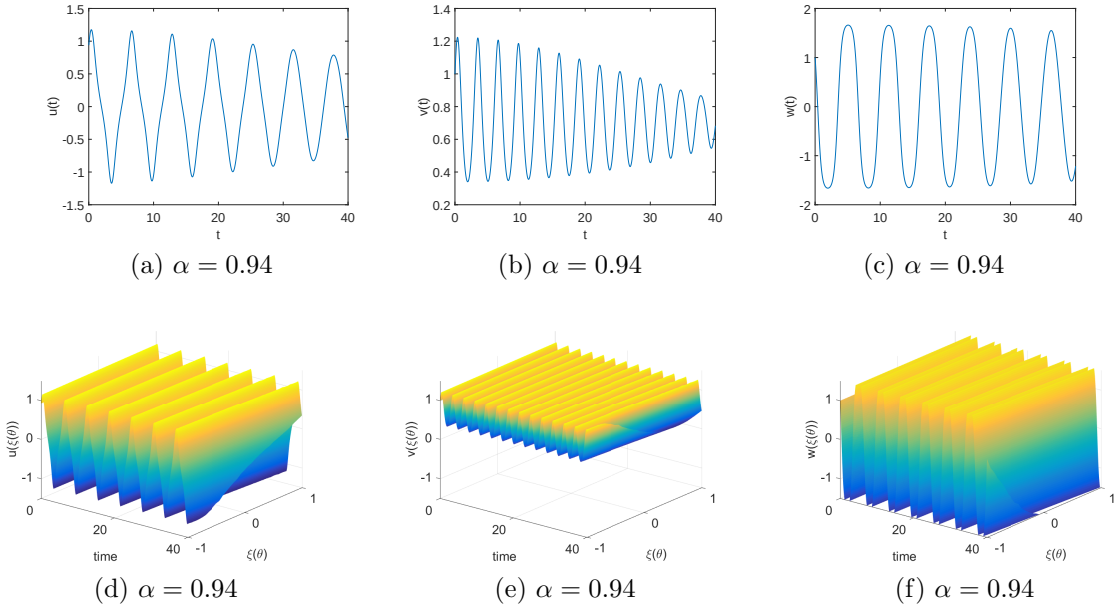
for $k = 1, 2, \ldots, P$.



(a) $\alpha = 0.94$ (b) $\alpha = 0.94$ (c) $\alpha = 0.94$

(d) $\alpha = 0.94$ (e) $\alpha = 0.94$ (f) $\alpha = 0.94$

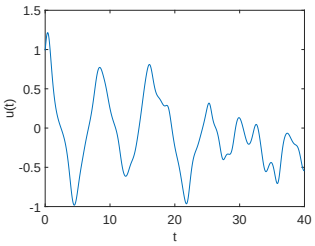
Figure 3.17: First row shows the mean plots of state variables u, v, w using linear WBe expansions for the K-O problem at $t = [0, 40]$ with 1000 spatial grid points and $dt = 0.01$. Second row shows the 3D plots of solution u, v, w in (d), (e), (f) respectively. Results are obtained for expansion orders $P = 5$.



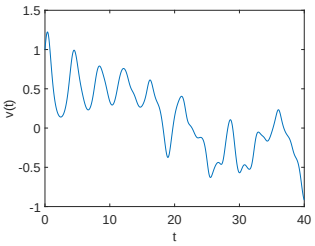
(a) $\alpha = 0.94$ (b) $\alpha = 0.94$ (c) $\alpha = 0.94$

(d) $\alpha = 0.94$ (e) $\alpha = 0.94$ (f) $\alpha = 0.94$

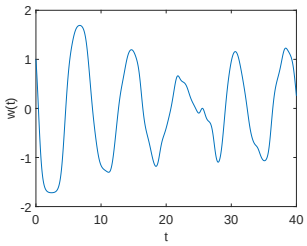
Figure 3.18: First row shows the mean plots of state variables u, v, w using linear WBe expansions for the K-O problem at $t = [0, 40]$ with 1000 spatial grid points and $dt = 0.01$. Second row represents the 3-D plots of the solution of state variables u, v, w. Results are obtained for expansion orders $P = 30$.



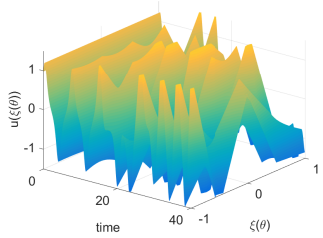
(a) $\alpha = 0.99$



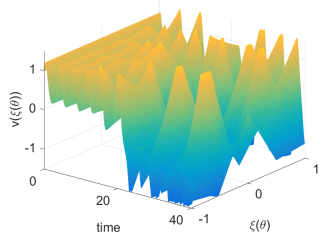
(b) $\alpha = 0.99$



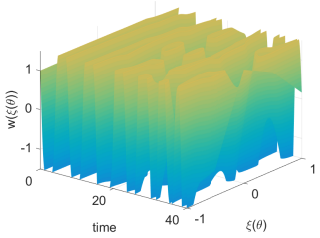
(c) $\alpha = 0.99$



(d) $\alpha = 0.99$

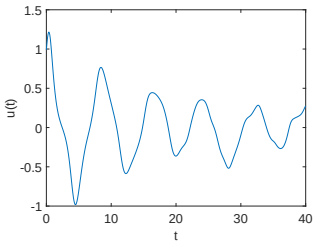


(e) $\alpha = 0.99$

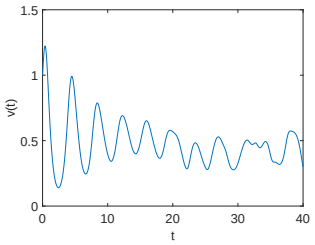


(f) $\alpha = 0.99$

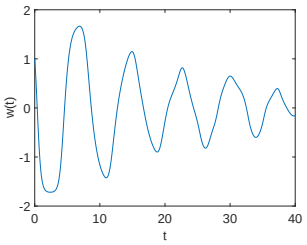
Figure 3.19: First row shows the mean plots of state variables u, v, w using linear WBe expansions for the K-O problem at $t = [0, 40]$ with 1000 spatial grid points and $dt = 0.01$. Second row represents the 3-D plots of the solution of state variables u, v, w. Results are obtained for expansion orders $P = 5$.



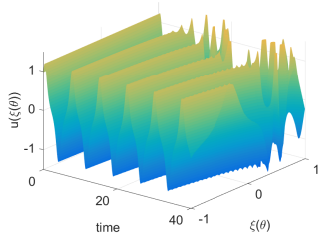
(a) $\alpha = 0.99$



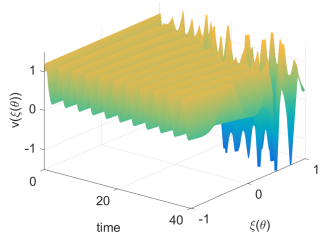
(b) $\alpha = 0.99$



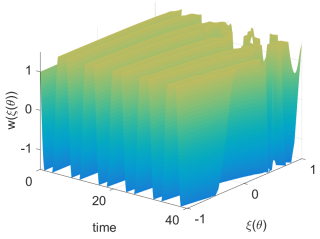
(c) $\alpha = 0.99$



(d) $\alpha = 0.99$



(e) $\alpha = 0.99$



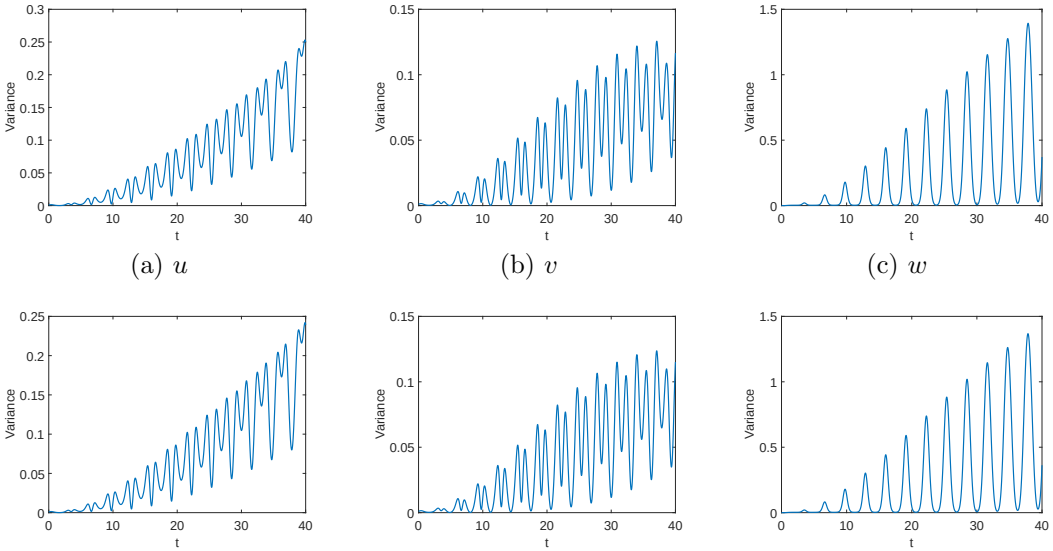
(f) $\alpha = 0.99$

Figure 3.20: First row shows the mean plots of state variables u, v, w using linear WBe expansions for the K-O problem at $t = [0, 40]$ with 1000 spatial grid points and $dt = 0.01$. Second row represents the 3-D plots of the solution of state variables u, v, w. Results are obtained for expansion orders $P = 30$.

3.3.3.2 Results and Discussion

We have used the RK4 solver for the time integration of the deterministic part by taking $dt = 0.01$. In the figures, the value of α is considered as 0.94 and 0.99. By taking $P = 5$, the results for K-O problem are shown in Fig. 3.17 for $\alpha = 0.94$ and for $\alpha = 0.99$, the results are depicted in Fig. 3.19 at $t \in [0, 40]$. Similarly, by taking $P = 30$, the results of this problem are shown in Fig. 3.18 for $\alpha = 0.94$ and for $\alpha = 0.99$, the results are depicted in Fig. 3.20 at $t \in [0, 40]$. The first row plots the mean of the three state variables u, v, w in part a, b, c whereas parts d, e, f represents the 3D solution plots of the state variables u, v, w in all the above mentioned figures. One can note in Figs. 3.17 and 3.18 for $\alpha = 0.94$ that mean plots of u, v, w at $P = 5$ are almost resembling with the mean plots of u, v, w at $P = 30$ whereas in Figs. 3.19 and 3.20, for $\alpha = 0.99$, there is a strong variation in the mean plots of u, v, w and these variations are also effected by the PCE order. This shows that $\alpha = 0.94$ doesn't affect the solution much whereas $\alpha = 0.99$ is the influencing factor.

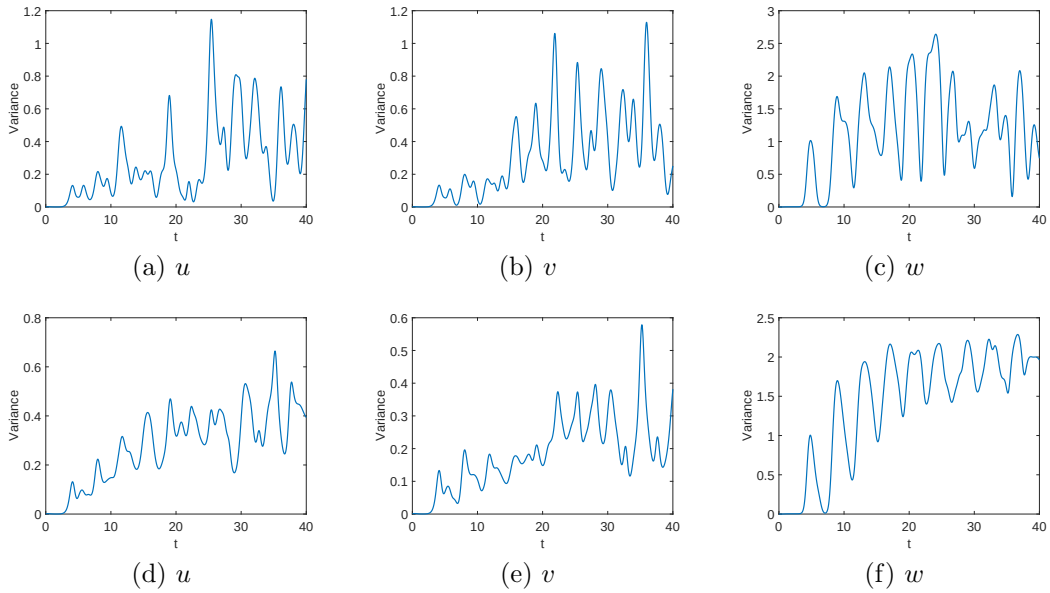
The variance plots for $\alpha = 0.94$ and $\alpha = 0.99$ are shown in Fig 3.21 and Fig 3.22 respectively for different PCE orders ($P = 5$ (First row) and $P = 30$ (Second Row)). It has been observed that in Fig. 3.21 for $\alpha = 0.94$ that the variance plots of the three state variables for $P = 5$ are not affected much even if we plot the same plots for higher PCE order $P = 30$. When $\alpha = 0.99$, the variance plots displays that the variance of u, v, w decreases as we increase the P.



(a) u (b) v (c) w

(d) u (e) v (f) w

Figure 3.21: Variance for the K-O problem at $t = 40$ with 1000 spatial grid points and $dt = 0.01$, $\alpha = 0.94$. Results are obtained for expansion orders $P = 5$ (First Row) and $P = 30$ (Second Row).



(a) u (b) v (c) w

(d) u (e) v (f) w

Figure 3.22: Variance for the K-O problem at $t = 40$ with 1000 spatial grid points and $dt = 0.01$, $\alpha = 0.99$. Results are obtained for expansion orders $P = 5$ (First Row) and $P = 30$ (Second Row).

3.3.4 Conclusion

In this chapter, we have constructed a scheme by using B-spline wavelets which is regarded as basis in the gPC representation. The expressions of scaling and wavelet functions of semi-orthogonal compactly supported linear B-spline wavelets for the bounded interval $[0, 1]$ act as the basis in the PCE for possible stochastic processes. Further, we have applied both semi-orthogonal Galerkin projection and pseudo-spectral projection on the solution variables and the stochastic data in the first test problem for the non-linear cubic term so as to assess the deterministic coefficients of PCE. Then, the behaviour of the stochastic process is directed by integrating the system of equations obtained from both the projection using RK4 method. We came to the conclusion that Galerkin projection is better in comparison to pseudo-spectral approximation which is numerically expected. We have demonstrated the scheme through engineering problems of real life importance- ODE with non-linear term, stochastic linear oscillator and stochastic K-O problem. In all the three test problems, it has been concluded that the wavelet function based expansion shows superior results as compared to scaling function based expansion. In the third problem, the effect of parameter α on the state variables has been analyzed.

Chapter 4

Uncertainty Quantification of Stochastic Epidemic SIR models[1]

Mathematical models are indispensable tools to understand and make predictions regarding the processes involved in epidemiology. These models help in analyzing the behavior of the spread and control of infectious diseases. SIR models are amongst the most significant epidemic models in disease control. Earlier, only specific infectious diseases like smallpox, malaria, rubella, measles, etc., were modeled using simple ODEs. However, emerging new diseases have motivated scientists and researchers to study infectious diseases. Nowadays, mathematical models of many new diseases are coming into existence such as Ebola, influenza, AIDS, hantavirus, and many more. In 1927, Kermack and McKendrick [218] gave the first detailed study for deterministic epidemic models. Later on, Bartlett made important contributions in 1949 [219] and Kendall in 1956 [220], by considering both deterministic and stochastic models. Stochasticity is included in these models to make them more realistic.

In literature, there are many epidemic models such as MSEIRS, MSEIR, SEIRS, SEIR, SIR, SIRS, SEI, SEIS, SI, and SIS which are made according to the flow patterns [221] available among different compartments (M-Mother, E-exposed, S-Susceptibles, I-Infectives, R-Recovered/Removed). Throughout this chapter, we focus only on the SIR model in different diseases and scenarios. SIR model simply divides the population into three classes - Susceptibles, Infected and Removed, or Recovered. As the rate of transmission and recovery parameters are not exactly known or one may say they are uncertain in nature, so we will consider them as RVs with some PDF. Moreover, we will examine how small changes in these parameters are going to effect the quantities of interest of the model. However, better knowledge of the transmission features of infectious diseases will lead to the approaches which will diminish the transmission of these diseases. In a simple SIR epidemic, the transitions happen from susceptible to infected and from infected to recovered. Mathematical models are employed for designing, planning, performing, assessing,

[1]The content of this chapter is published as "Uncertainty Quantification of Stochastic Epidemic SIR Models Using B-spline Polynomial Chaos", *Regular and Chaotic Dynamics*, 26(1), 22-38, 2021, (**SCI: Impact Factor: 1.421**).

and optimizing various identification, preventive, and control measures.

Since wavelet-based PCEs have never been utilized in quantifying the uncertainties in an epidemic model, therefore, in this chapter, we intend to enhance the capabilities of PCEs by introducing wavelet-based PCEs [55, 88, 222] into the field of epidemiology. As wavelet theory is itself a very vast field, so we aim to employ B-spline wavelet-based gPC in the medical area by solving the simple SIR model. Therefore, the main contribution of this chapter is to implement the linear and cubic Wiener B-spline chaos method for understanding the behavior of infectious diseases.

Also, in literature, there is substantial work on gPC but only a few papers have been applied to epidemic models. B. Kegan and R. W. West [223] examined the deterministic epidemic model under the influence of uncertain ICs. Chen-Charpentier and Stanescu [224] applied gPC to SIR as well as SIRS models by engaging random coefficients. Santonja and Chen-Charpentier [118] explained how PCEs can turn out to be a wonderful tool for random obesity epidemics along with SA. Roberts [225] employed gPC by introducing randomness in the reproduction number. Omar and Hasan [226] investigated the dynamic behavior of the epidemic under the influence of random beta distributed initial state. Harman and Johnston [227] compared the stochastic Galerkin and MC method by applying it on a simple SIR epidemic model with some level of randomness.

4.1 Mathematical Model- Stochastic SIR model

Mathematical models have been significantly used in analyzing how infectious diseases spread and how they can be controlled. SIR model is among the most simple and easiest models for expressing the epidemics. For communicable diseases like chickenpox, influenza, H1N1, smallpox, etc., we have three different classes of individuals. These are as follows:

a) **Susceptibles $\mathcal{S}(t)$** are the healthy people who have not contracted the disease and are prone to get infected,

b) **Infectives $\mathbf{I}(t)$** who contracted the disease and are going to transmit it to healthy people, and

c) **Removed or Recovered $\mathcal{R}(t)$** are those who are immune from the disease, so they can no longer be infected with the disease.

Together these three classes make the SIR model. In such models, individuals can only follow two moves, i.e., move from susceptibles $\mathcal{S}(t)$ to infectives $\mathrm{I}(t)$ and from infectives

I(t) to Recovered $\mathcal{R}(t)$. Specifically, without considering the demographic factors (births and natural deaths), one can model these three categories into a system of coupled ODEs in the following way:

$$\begin{aligned} \frac{d\mathcal{S}}{dt} &= -\beta \mathcal{S} I, & \mathcal{S}(t=0) \geq 0, \\ \frac{d\mathrm{I}}{dt} &= \beta \mathcal{S} I - \gamma I, & \mathrm{I}(t=0) > 0, \\ \frac{d\mathcal{R}}{dt} &= \gamma I, & \mathcal{R}(t=0) \geq 0. \end{aligned} \tag{4.1.1}$$

The first equation states that to get infected, the susceptibles must get in contact with the infectives, i.e., infections occur at the rate proportional to $\mathcal{S}(t)\mathrm{I}(t)$ with β being the contact/transmission rate i.e., adequate number of contacts with infectives. Also, the parameter $1/\gamma$ is the average infection rate and $R_0 = \beta/\gamma$ denotes the reproduction number which tells us the average number of secondary infections that take place when a single infective enters a completely susceptible host population.

The total population is denoted by $\mathcal{T}(t)$ such that $\mathcal{T}(t)$=$\mathcal{S}(t)$+$\mathrm{I}(t)$+$\mathcal{R}(t)$. The infectives must have "close contact" with other individuals randomly which means a contact must be capable of resulting in an infection if the individual on the other side is susceptible otherwise the contact will not affect. As soon as the susceptible comes in close contact with the infectives, it immediately becomes infected and then following the same rules, it starts spreading the disease. There is a random time (infectious period) when infected individuals remain infectious and after that, they end up being infectious, heal up or recover, and become immune to the disease.

Since in reality, the contact rate and recovery rate parameters, β and γ, are neither fixed nor exactly known, we will consider them as RVs with some PDFs. Also, if we are not having the exact information regarding the ICs such as the number of susceptibles at the starting, then we can also introduce uncertainty into the ICs and accordingly analyze the behavior of the model.

4.2 Expressions for B-spline Wavelets

Here, we will discuss only linear ($n_B = 2$) and cubic ($n_B = 2$) B-spline wavelets. These wavelets consist of inner and boundary scaling and wavelet functions. For the standard case, we have $n_B - 1$ boundary scaling functions at each boundary and $2^j - n_B + 1$ inner scaling functions. Further, we have $n_B - 1$ boundary wavelets at each boundary and

$2^j - 2n_B + 2$ inner wavelets [213]. The scale $\jmath$ of linear B-spline wavelets starts with $\jmath = \jmath_0 = 2$ whereas for cubic B-spline wavelets it starts with $\jmath = \jmath_0 = 3$.

The expressions for scaling and wavelet functions for linear B-spline wavelets which have been clearly described in chapter 3. On the other hand, we will focus only on the scaling functions of cubic B-spline wavelets in this chapter. The inner scaling functions of cubic B-spline wavelets [228, 229] with $x_\jmath = 2^\jmath x$ for $k = 0,\ 1,\ \ldots,\ 2^\jmath - 4$ are as follows:

$$\phi_{\jmath,k}(x) = \begin{cases} \frac{1}{6}(x_\jmath - k)^3, & k \le x_\jmath < k+1, \\ \frac{1}{6}(-3(x_\jmath - k)^3 + 12(x_\jmath - k)^2 - 12(x_\jmath - k) + 4), & k+1 \le x_\jmath < k+2, \\ \frac{1}{6}(3(x_\jmath - k)^3 - 24(x_\jmath - k)^2 + 60(x_\jmath - k) - 44), & k+2 \le x_\jmath \le k+3, \\ \frac{1}{6}(4 - (x_\jmath - k)^3), & k+3 \le x_\jmath \le k+4, \\ 0, & \text{otherwise.} \end{cases} \tag{4.2.1}$$

The cubic left boundary B-spline scaling functions for $k = -3, -2, -1$ are given as follows: For $k = -3$

$$\phi_{\jmath,k}(x) = \begin{cases} (1-8x)^3, & 0 \le x < \frac{1}{8}, \\ 0, & \text{otherwise.} \end{cases} \tag{4.2.2}$$

For $k = -2$

$$\phi_{\jmath,k}(x) = \begin{cases} 896x^3 - 288x^2 + 24x, & 0 \le x < \frac{1}{8}, \\ 2(1-4x)^3, & \frac{1}{8} \le x < \frac{2}{8}, \\ 0, & \text{otherwise.} \end{cases} \tag{4.2.3}$$

For $k = -1$

$$\phi_{\jmath,k}(x) = \begin{cases} -\frac{1408}{3}x^3 + 96x^2, & 0 \le x < \frac{1}{8}, \\ \frac{896}{3}x^3 - \frac{576}{3}x^2 + 36x - \frac{3}{2} & \frac{1}{8} \le x < \frac{2}{8}, \\ -\frac{1}{6}(8x-3)^3 & \frac{2}{8} \le x < \frac{3}{8}, \\ 0, & \text{otherwise.} \end{cases} \tag{4.2.4}$$

The cubic right boundary B-spline scaling functions for $k = 2^j - 3,\ 2^j - 2,\ 2^j - 1$ are given by $\phi_{3,k}(x) = \phi_{3,4-k}(1-x)$ such that for $k = 2^j - 3$, we have,

$$\phi_{j,k}(x) = \begin{cases} -\frac{1}{6}(8(1-x)-3)^3 & \frac{5}{8} \le x < \frac{6}{8}, \\ \frac{896}{3}(1-x)^3 - \frac{576}{3}(1-x)^2 + 36(1-x) - \frac{3}{2} & \frac{6}{8} \le x < \frac{7}{8}, \\ -\frac{1408}{3}(1-x)^3 + 96(1-x)^2, & \frac{7}{8} \le x < 1, \\ 0, & \text{otherwise.} \end{cases} \tag{4.2.5}$$

For $k = 2^J - 2$, we have,

$$\phi_{J,k}(x) = \begin{cases} 2(1-4(1-x))^3, & \frac{6}{8} \leq x < \frac{7}{8}, \\ 896(1-x)^3 - 288(1-x)^2 + 24(1-x), & \frac{7}{8} \leq x < 1, \\ 0, & \text{otherwise.} \end{cases} \tag{4.2.6}$$

For $k = 2^J - 1$, we have,

$$\phi_{J,k}(x) = \begin{cases} (1-8(1-x))^3, & \frac{7}{8} \leq x < 1, \\ 0, & \text{otherwise.} \end{cases} \tag{4.2.7}$$

If we look into the formation of linear and cubic B-spline scaling functions, it is clearly seen that more computational effort is required in case of cubic B-spline wavelets. We can obtain dual scaling and wavelet functions for both linear and cubic B-spline wavelets as discussed in Chapter 3.

4.3 Wavelet approximation of a 1D random process

Let ξ be a RV and $X(\xi(\theta))$ be the second-order random process. Then, we have generalized WBe chaos expansion of $X(\xi(\theta))$ as

$$X(\xi(\theta)) = \sum_{k=-n_B+1}^{2^{J_0}-1} \mathfrak{c}_{J_0,k} \phi^s_{J_0,k} + \sum_{j=J_0}^{\infty} \sum_{k=-n_B+1}^{2^j-n_B} \mathfrak{d}_{j,k} \psi^w_{j,k}. \tag{4.3.1}$$

After that, we truncate the expression in Eq. (4.3.1) to J and also by taking the value of $J_0 = 2$ (linear case) and $J_0 = 3$ (cubic case), we have

$$X(\xi(\theta)) = \sum_{k=-n_B+1}^{2^{J_0}-1} \mathfrak{c}_{J_0,k} \phi^s_{J_0,k} + \sum_{j=J_0}^{J} \sum_{k=-n_B+1}^{2^j-n_B} \mathfrak{d}_{j,k} \psi^w_{j,k}. \tag{4.3.2}$$

Moreover,

$$X(\xi(\theta)) = \sum_{k=-n_B+1}^{2^{J_0}-1} X^s_{J_0,k} S_{J_0,k}(\xi(\theta)) + \sum_{j=J_0}^{J} \sum_{k=-n_B+1}^{2^j-n_B} X^w_{j,k} \mathscr{W}_{j,k}(\xi(\theta)), \tag{4.3.3}$$

where $X^s_{J_0,k}$ and $X^w_{j,k}$ are respectively the coefficients of scaling and wavelet approximation of $X(\xi)$ such that

$$S_{J_0,k}(\xi(\theta) \in [a,b]) = \phi^s_{J_0,k}(\wp(\xi)),$$

$$\mathscr{W}_{j,k}(\xi(\theta) \in [a,b]) = \psi^w_{j,k}(\wp(\xi)). \tag{4.3.4}$$

We can rewrite Eq. (4.3.3) as

$$X(\xi(\theta) \in [a,b]) = \sum_{k=-n_B+1}^{2^{J_0}-1} X^s_{J_0,k}\phi^s_{J_0,k}(\wp(\xi)) + \sum_{j=J_0}^{J}\sum_{k=-n_B+1}^{2^j-n_B} X^w_{j,k}\psi^w_{j,k}(\wp(\xi)). \tag{4.3.5}$$

Let us now concatenate the scale index j and space index k and define the generalized set of index integers Υ by $\mathcal{Z}$ such that $\mathcal{Z} \equiv \{\Upsilon : (\Upsilon_s = k + n_B,\ J_0 = 2 \text{ (linear) and } J_0 = 3 \text{ (cubic)},\ k = -n_B+1,\ \ldots,\ 2^{J_0}-1) \cup (\Upsilon_w = 2^j + (k+2n_B-1),\ j = J_0,\ J_0+1,\ \ldots,\ J,\ k = -n_B+1,\ \ldots,\ 2^j - n_B)\}$. The resolution level will be denoted by $|\Upsilon|$. Therefore, the one-dimensional wavelet expansion of $X(\xi(\theta))$ can be expressed as

$$X(\xi(\theta) \in [a,b]) = \sum_{\Upsilon \in \mathcal{Z}} X_\Upsilon W_\Upsilon(\xi(\theta)), \tag{4.3.6}$$

where

$$\begin{aligned} X_\Upsilon &= \int_0^1 X(\wp^{-1}(y))\phi^s_\Upsilon(y)dy \quad \text{when} \quad \Upsilon = 1,\ 2,\ \ldots,\ 2^{J_0} + n_B - 1, \\ X_\Upsilon &= \int_0^1 X(\wp^{-1}(y))\psi^w_\Upsilon(y)dy \quad \text{when} \quad \Upsilon > 2^{J_0} + n_B - 1. \end{aligned}$$

Further, we execute the inner products of $\{W_\Upsilon(\xi(\theta)\}$ w.r.t dual scaling and wavelet functions $\{\tilde{W}_\Upsilon(\xi(\theta))\}$ such that when $\Upsilon_1, \Upsilon_2 = 1,\ 2,\ \ldots,\ 2^{J_0} + n_B - 1$, we have

$$\int_a^b W_{\Upsilon_1}(\xi)\tilde{W}_{\Upsilon_2}(\xi)pdf(\xi)d\xi = \int_a^b \phi_{j,k}(\wp(\xi))\tilde{\phi}_{j,l}(\wp(\xi))pdf(\xi)d\xi = \delta_{kl}. \tag{4.3.7}$$

In the similar manner, when $\Upsilon_1, \Upsilon_2 > 2^{J_0} + n_B - 1$, we have

$$\int_a^b W_{\Upsilon_1}(\xi)\tilde{W}_{\Upsilon_2}(\xi)pdf(\xi)d\xi = \int_a^b \psi_{j,k}(\wp(\xi))\tilde{\psi}_{l,m}(\wp(\xi))pdf(\xi)d\xi = \delta_{jl}\delta_{km}. \tag{4.3.8}$$

Therefore, the stochastic process is approximated by generalized WBe gPC as follows:

$$X(\xi) \approx \sum_{k=1}^{P} X_k W_k(\xi). \tag{4.3.9}$$

Now, we will extract the expectation

$$\langle X \rangle = \sum_{i=1}^{P} X_i \langle W_i(\xi), 1 \rangle.$$

and S.D. as follows:

$$\text{S.D.} = \sigma(X) \quad = \quad \sqrt{\sum_{i=1}^{P}\sum_{j=1}^{P} X_i X_j \langle W_i(\xi) W_j(\xi), 1\rangle - \big(\langle X\rangle\big)^2}.$$

After introducing the basic mathematical epidemic model, we will demonstrate it using real outbreaks in next section.

4.4 Numerical Results

4.4.1 Influenza Epidemic in British boys boarding school, 1978

The epidemic took place in north England in 1978 in a British boys boarding school. Murray [230] gave this epidemic on the basis of data organized by the British Communicable Disease Surveillance Centre (British Medical Journal, p. 587, March 4 1978 [231]). It was basically a flu epidemic that affected 763 resident boys. On 10 January, boys were returning for the Easter term from all over Britain and some from the Far East and Europe. At that time, one boy from Hong Kong had a temporary febrile infection from 15 to 18 January. Then, three boys were sent to the college medical center on 22 January, and later on, the number of infected boys started increasing. The symptoms which were seen among most of the infected boys were that they were feeling exhausted along with headache as fever developed. Apart from that, they suffered from sore throat and tracheitis. The temperature in the mornings stayed a little higher whereas it was usually 100 ° -102 ° F in the daytime. Fig. 4.1 highlights the daily infectives who were restricted to bed in the boarding school.

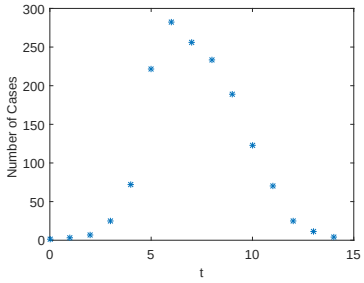


Figure 4.1: The graph published by British Medical Journal which displays data of boys who were confined to bed.

Also, the data for the influenza epidemic for two-week is constructed in Table 4.1 by reading values from the graph reported in the British Medical Journal 1978. Since the

values were read from the graph published, so the values given in Table 4.1 may consist of some degree of randomness. This whole influenza is modeled using the coupled ODEs involved in the SIR model as in Eq. (4.1.1). As per the data given, we have deterministic ICs as susceptibles are 762 in number and initially there was only one infective from Hong Kong, therefore, we have

$$\mathcal{S}(t=0) = 762, \quad I(t=0) = 1, \quad \mathcal{R}(t=0) = 0. \tag{4.4.1}$$

No. of days	Date	Infected	Recovering	Recovered
1	$1978-01-22$	3	0	1
2	$1978-01-23$	8	0	5
3	$1978-01-24$	26	0	17
4	$1978-01-25$	76	0	51
5	$1978-01-26$	225	9	152
6	$1978-01-27$	298	17	286
7	$1978-01-28$	258	105	402
8	$1978-01-29$	233	162	507
9	$1978-01-30$	189	176	592
10	$1978-01-31$	128	166	650
11	$1978-02-01$	68	150	681
12	$1978-02-02$	29	85	694
13	$1978-02-03$	14	47	700
14	$1978-02-04$	4	20	702

Table 4.1: School data

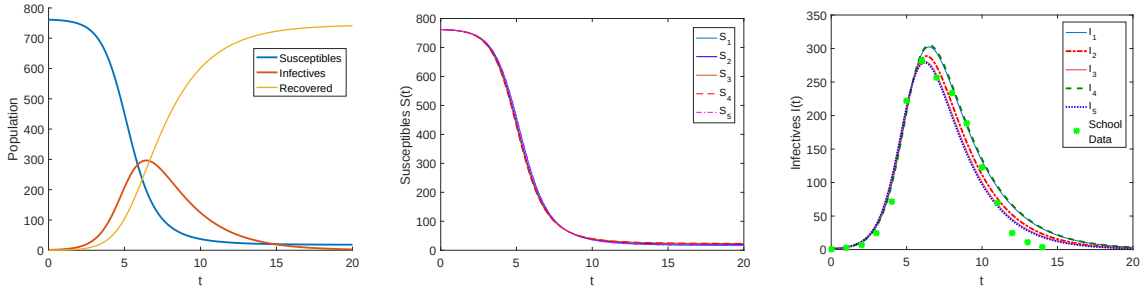
The transmission and recovery rate are uncertain in nature, therefore, we can write it as

$$\begin{aligned} \beta =& \beta_0 + \beta_1 \xi, \\ \gamma =& \gamma_0 + \gamma_1 \xi^*. \end{aligned} \tag{4.4.2}$$

where ξ and ξ^* are RVs with uniform distribution over the interval [-1,1].

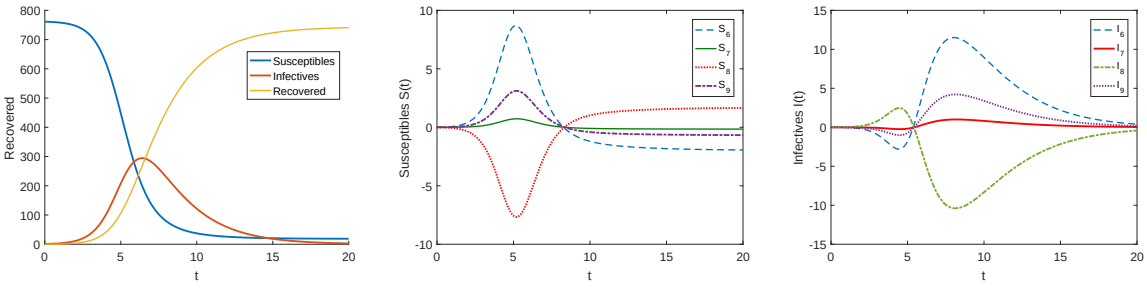
4.4.1.1 Solution Method

Let $\boldsymbol{\xi}$ be a vector with two components where each component is uniformly distributed over $[-1, 1]$. Then, the system in Eq. (4.1.1) is approximated by generalized WBe as



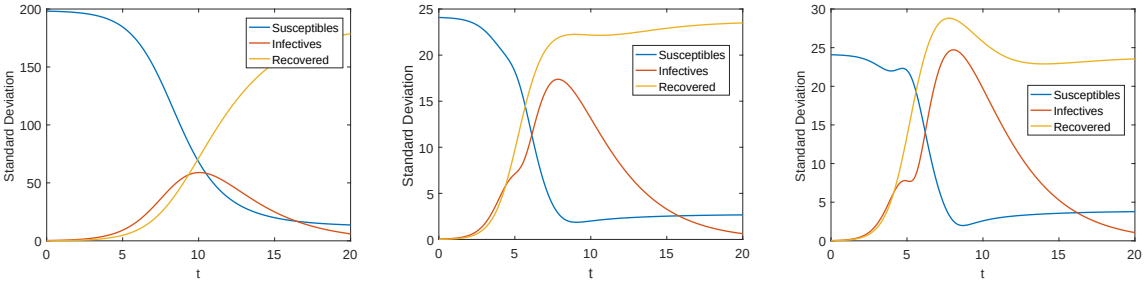
(a) (b) (c)

Figure 4.2: a) Mean of Susceptibles, Infectives, Recovered using linear B-spline wavelet gPC with $P = 5$. The first five coefficients of gPC for b) Susceptibles c) Infectives have been presented using linear B-spline scaling functions.



(a) (b) (c)

Figure 4.3: a) Mean of Susceptibles, Infectives, Recovered using linear B-spline wavelet gPC with $P = 9$. Next four coefficients of gPC for b) Susceptibles c) Infectives have been displayed using first four linear B-spline wavelets.



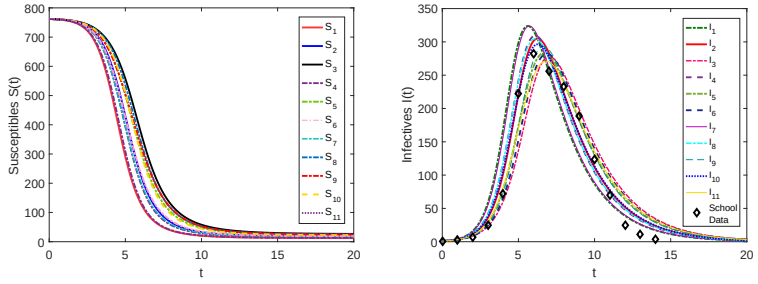
(a) (b) (c)

Figure 4.4: S.D. using linear B-spline wavelet for a) $P = 1$ b) $P = 5$ c) $P = 9$.

follows:

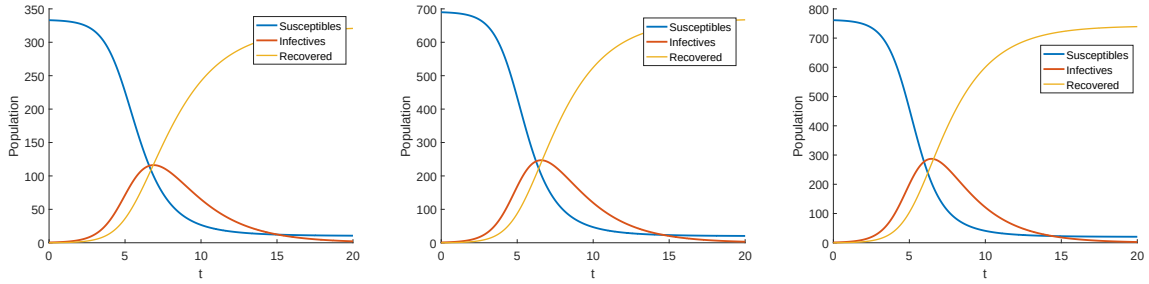
$$\mathcal{S}(t,\boldsymbol{\xi}) = \sum_{i=1}^{P} \mathcal{S}_i(t) W_i(\boldsymbol{\xi}), \quad \mathrm{I}(t,\boldsymbol{\xi}) = \sum_{i=1}^{P} \mathrm{I}_i(t) W_i(\boldsymbol{\xi}), \quad \mathcal{R}(t,\boldsymbol{\xi}) = \sum_{i=1}^{P} \mathcal{R}_i(t) W_i(\boldsymbol{\xi}),$$
$$\beta(\boldsymbol{\xi}) = \sum_{i=1}^{P} \beta_i W_i(\boldsymbol{\xi}), \quad \gamma(\boldsymbol{\xi}) = \sum_{i=1}^{P} \gamma_i W_i(\boldsymbol{\xi}). \tag{4.4.3}$$

After inserting above WBe approximations in Eq.(4.1.1), we get



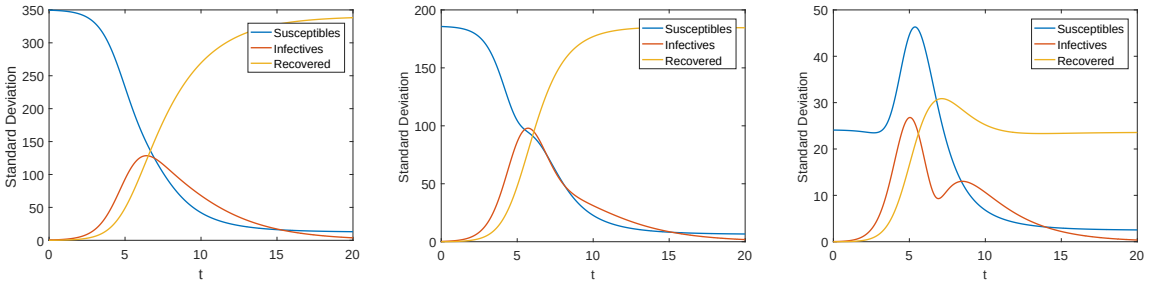
(a) (b)

Figure 4.5: First 11 gPC coefficients for a) Susceptibles b) Infectives using 11 cubic B-spline scaling functions. The school data of Infectives is shown by diamond in part b.



(a) (b) (c)

Figure 4.6: Mean of $\mathcal{S}$, I, $\mathcal{R}$ are plotted for a) $P = 5$ b) $P = 9$ c) $P = 11$ using scaling functions of cubic B-spline wavelets.



(a) (b) (c)

Figure 4.7: S.D. of state variables $\mathcal{S}$, I, $\mathcal{R}$ for a) $P = 5$ b) $P = 9$ c) $P = 11$ are plotted using scaling functions of cubic B-spline wavelets.

$$
\begin{aligned}
\frac{d}{dt}\Big(\sum_{i=1}^{P}\mathcal{S}_i(t)W_i\Big) &= -\sum_{i=1}^{P}\sum_{j=1}^{P}\sum_{k=1}^{P}\beta_i\mathcal{S}_j(t)\mathrm{I}_k(t)W_iW_jW_k,\\
\frac{d}{dt}\Big(\sum_{i=1}^{P}\mathrm{I}_i(t)W_i\Big) &= \sum_{i=1}^{P}\sum_{j=1}^{P}\sum_{k=1}^{P}\beta_i\mathcal{S}_j(t)\mathrm{I}_k(t)W_iW_jW_k - \sum_{i=1}^{P}\sum_{j=1}^{P}\gamma_i\mathrm{I}_j(t)W_iW_j, \qquad (4.4.4)\\
\frac{d}{dt}\Big(\sum_{i=1}^{P}\mathcal{R}_i(t)W_i\Big) &= \sum_{i=1}^{P}\sum_{j=1}^{P}\gamma_i\mathrm{I}_j(t)W_iW_j
\end{aligned}
$$

Taking inner product of Eq. (4.4.4) w.r.t $\tilde{W}_l$, we get

$$
\begin{aligned}
\frac{d}{dt}\sum_{i=1}^{P}\mathcal{S}_i(t)\langle W_i\tilde{W}_l\rangle &= -\sum_{i=1}^{P}\sum_{j=1}^{P}\sum_{k=1}^{P}\beta_i\mathcal{S}_j(t)\mathrm{I}_k(t)\langle W_iW_jW_k\tilde{W}_l\rangle,\\
\frac{d}{dt}\sum_{i=1}^{P}\mathrm{I}_i(t)\langle W_i\tilde{W}_l\rangle &= \sum_{i=1}^{P}\sum_{j=1}^{P}\sum_{k=1}^{P}\beta_i\mathcal{S}_j(t)\mathrm{I}_k(t)\langle W_iW_jW_k\tilde{W}_l\rangle\\
&\quad - \sum_{i=1}^{P}\sum_{j=1}^{P}\gamma_i\mathrm{I}_j(t)\langle W_iW_j\tilde{W}_l\rangle, \qquad (4.4.5)\\
\frac{d}{dt}\sum_{i=1}^{P}\mathcal{R}_i(t)\langle W_i\tilde{W}_l\rangle &= \sum_{i=1}^{P}\sum_{j=1}^{P}\gamma_i\mathrm{I}_j(t)\langle W_iW_j\tilde{W}_l\rangle
\end{aligned}
$$

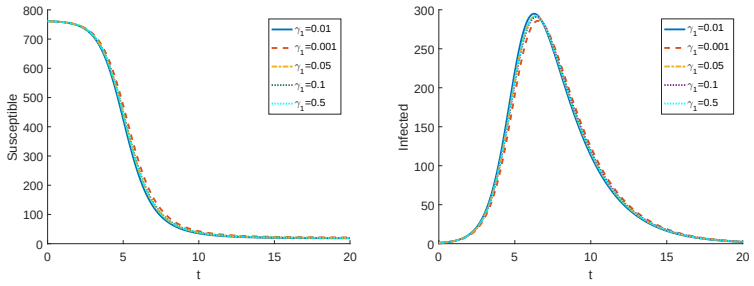
where $l = 1,\ 2,\ \ldots,\ P$. From the above equations, we see that we need to compute triple as well as fourth product so as to obtain the solutions of $\mathcal{S}(t)$, $\mathrm{I}(t)$ and $\mathcal{R}(t)$. Similarly, we can apply the WBe on the ICs if they are given to be uncertain.

4.4.1.2 Sensitivity analysis

SA [232–234] is used to assess the influence of variations in parameters on the performance or quantities of interest of the mathematical model. To put it simply, one parameter of the model is changed by a certain amount by keeping all the other parameters fixed. After that, simulations are done by running the model and we observe the changes in the output of the system. To achieve more credible solutions, SA should be carried out in sequence with uncertainty analysis, which consists of propagation and quantification of uncertainties. So, it contributes an essential prerequisite in designing and the validation of models in the presence of uncertainties. We will discuss four cases using linear WBe as follows:

I) Keep $\beta = 0.00218 + 0.001\xi,\ \gamma_0 = 0.441$ fixed and take different values of γ_1 such as γ_1=0.001, 0.05, 0.1, 0.01, 0.5.

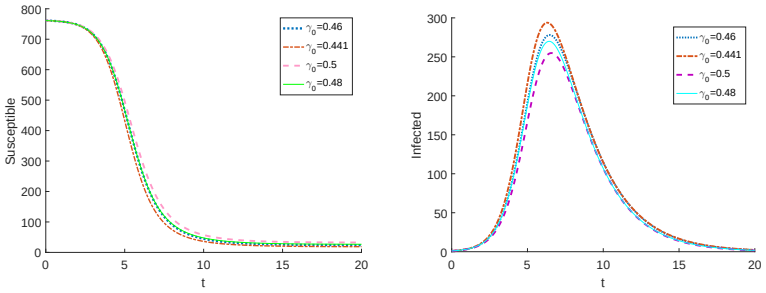
II) Fix $\beta = 0.00218 + 0.001\xi,\ \gamma_1 = 0.001$ and change γ_0 by taking different values such



(a) (b)

Figure 4.8: Mean of a) Susceptibles b) Infectives according to case 4.4.1.2 with $P = 5$.

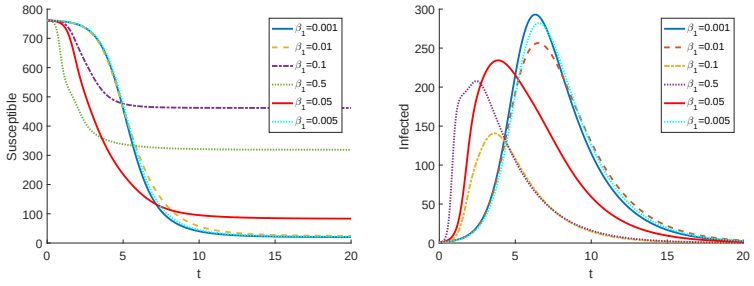
as γ_0=0.5, 0.441, 0.46, 0.48.



(a) (b)

Figure 4.9: Mean of a) Susceptibles b) Infectives according to case 4.4.1.2 with $P = 5$.

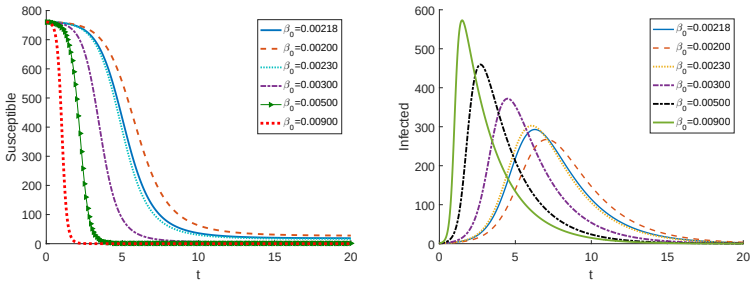
III) Keep $\beta_0 = 0.00218$, $\gamma = 0.441 + 0.001\xi^*$ fixed and change β_1 by taking β_1=0.001, 0.01, 0.1, 0.5, 0.05, 0.005.



(a) (b)

Figure 4.10: Mean of a) Susceptibles b) Infectives according to case 4.4.1.2 with $P = 5$.

IV) Keep $\beta_1 = 0.001$, $\gamma = 0.441 + 0.001\xi^*$ fixed and change β_0 as β_0=0.00218, 0.00200, 0.00230, 0.003, 0.005, 0.009.



(a) (b)

Figure 4.11: Mean of a) Susceptibles b) Infectives according to case 4.4.1.2 with $P = 5$.

4.4.1.3 Results and Discussion

We have applied RK4 method for time integration. The most important conclusions we get from B-spline gPC for the influenza outbreak are as follows:

1) From Fig. 4.2c, we can analyze that the coefficients of infectives are appropriately approximating the school data given in the British Medical Journal.

2) From Figs. 4.2a and 4.3a, we conclude that wavelet functions have negligible effect on the SIR model as scaling functions of linear B-spline wavelets are approximating the SIR model exactly in the same way as wavelet functions of linear B-spline wavelets do.

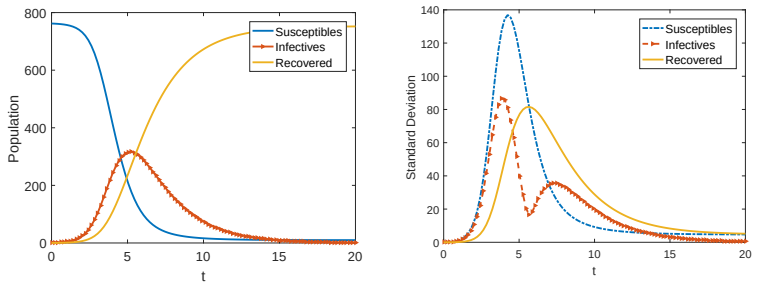
3) We analyze from Figs. 4.6 and 4.7 that we require atleast $dim(V_j) = 2^j + n_B - 1$ number of scaling functions to capture the local weighted average of the solution accurately.

4) It has been concluded that linear B-spline wavelets are showing accurate results at a lower PC order than the cubic B-spline wavelets. The reason being that linear B-spline wavelets involve only 5 scaling functions whereas cubic B-spline wavelets involve 11 scaling functions.

	MC method	B-spline gPC
N/P	100	5
CPU time (sec)	1.23	0.456

Table 4.2: Comparison of MC and linear B-spline gPC

5) From Figs. 4.8 and 4.9, we conclude that γ is an insensitive parameter which does not need as much effort to estimate as a small variation in this parameter is not producing large changes in quantity of interest. However, the solution is very sensitive towards



(a) (b)

Figure 4.12: a) Mean and b) S.D. of $\mathcal{S}$, I, $\mathcal{R}$ using MC method with $N = 100$ samples

the changes made in the parameter β. So, β deserves the most numerical attention and it should be carefully estimated.

6) Also, if we take higher values of β_0, the solution is coming out to be unstable.

7) From Fig. 4.12 and Table 4.2, we can notice that MC method takes $N = 100$ samples to approximate the SIR model. After comparing the S.D. plots and CPU time, we conclude that linear B-spline gPC is better than MC method.

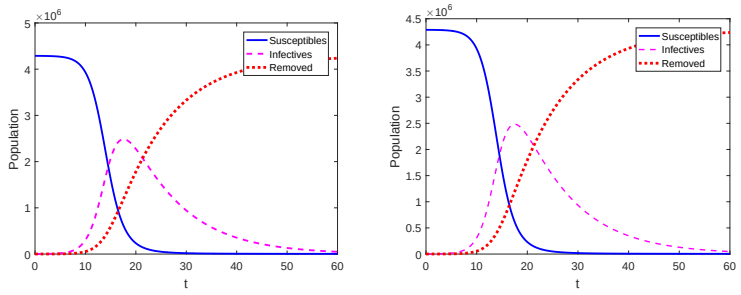
4.4.2 Ebola in Liberia 2014

Ebola virus was first located in the present Democratic Republic of Congo in 1976. Subsequently, many outbreaks came into picture, but the 2014 outbreak was the greatest and has spread through many countries, namely Sierra Leone, Guinea and Liberia. Africa suffered from around 20 outbreaks of Ebola with fatality rates of 25% to 90%. The sources of Ebola are human contacts with the secretions, blood, or other bodily fluids of infected animals found dead or sick in the rainforest. It is commonly assumed that the fruit bats of the Pteropodidae family are the natural Ebola virus hosts. According to the data given by WHO for Ebola that happened in Liberia in 2014, the total population $\mathcal{T}(t)$ estimated was 4294000 [235], and infected $\mathrm{I}(t = 0) = 846$ [236] and the people who died were 481 [236]. $\mathcal{R}(t)$ consists of people who have received permanent immunity, which includes people who have died as well as people who have recovered. Therefore, number of people recovered

$$\mathcal{R} = 481 + (0.3 \times 846) = 735.$$

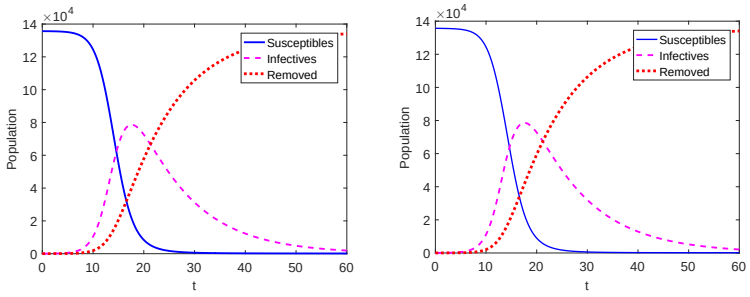
Hence [237],

$$\mathcal{T}(t=0) = 4294000, \quad I(t=0) = 846, \quad \mathcal{R}(t=0) = 735, \quad \mathcal{S}(t=0) = 4292419 \tag{4.4.6}$$



(a) (b)

Figure 4.13: Mean of $\mathcal{S}$, I, $\mathcal{R}$ of Ebola virus are shown for a) $P = 5$ and b) $P = 9$ using linear B-spline gPC.



(a) (b)

Figure 4.14: S.D. of $\mathcal{S}$, I, $\mathcal{R}$ of Ebola virus are shown for a) $P = 5$ and b) $P = 9$ using linear B-spline gPC.

We assume the total population, susceptible, infective and removed to be deterministic. But actually, for a large population like this, there may occur some kind of uncertainty or this knowledge about the population may not be quite accurate. Also, as reported by WHO, the mortality rate of Ebola was estimated to be 0.7 [**?**]. As the disease stays from 2 to 18 days, so if we take the midpoint approximate of the time period of the disease, i.e. 10 days, we get,

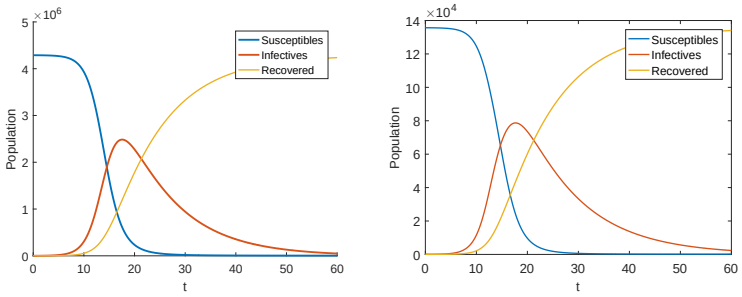
$$\gamma = \frac{1}{10} = 0.1 \tag{4.4.7}$$

But we assume it to be a bit uncertain as the duration of disease can be any number from

2 to 18 days. Similarly, we obtain β [238] as follows:

$$\beta = \frac{0.7}{4292419} = 1.63 \times 10^{-7} \tag{4.4.8}$$

Even β can be uncertain with ξ being uniformly distributed RV over [-1,1], i.e., if the mortality rate is 0.5, then $\beta = 1.16 \times 10^{-7}$.



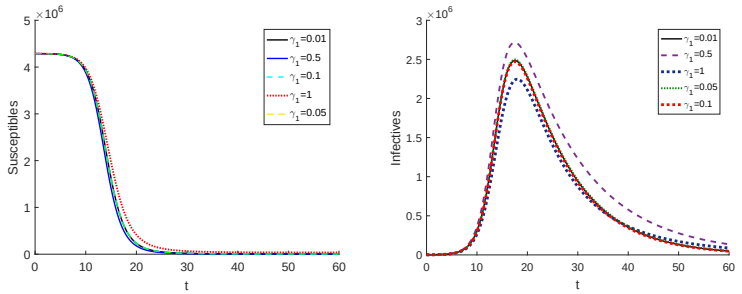
(a) (b)

Figure 4.15: Mean and S.D. of $\mathcal{S}$, I, $\mathcal{R}$ state variables is displayed for $P = 11$ using cubic B-spline gPC.

4.4.2.1 Sensitivity Analysis

Similarly, we will perform SA for the Ebola outbreak by taking four cases as earlier. As we are dealing with a large population, so even small change in output will cover upto a thousand of population. We will study the impact of the below mentioned four cases on the Ebola outbreak.

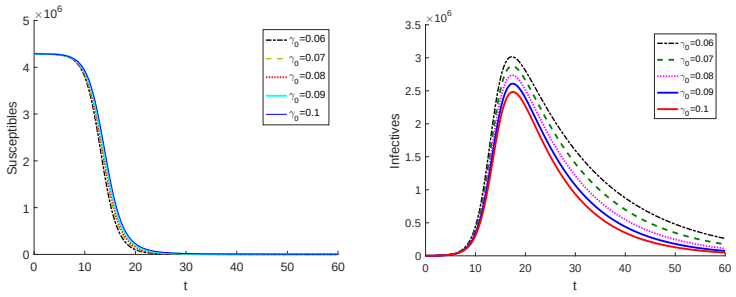
I) Keep $\beta = 1.63 \times 10^{-7} + 0.1 \times 10^{-7}\xi$, $\gamma_0 = 0.1$ fixed and change γ_1 by taking γ_1=0.1, 0.5, 0.01, 0.05, 1.



(a) (b)

Figure 4.16: Mean of a) Susceptibles $\mathcal{S}(t)$ and b)Infectives I(t) is shown for different values of γ_1 using linear B-spline gPC.

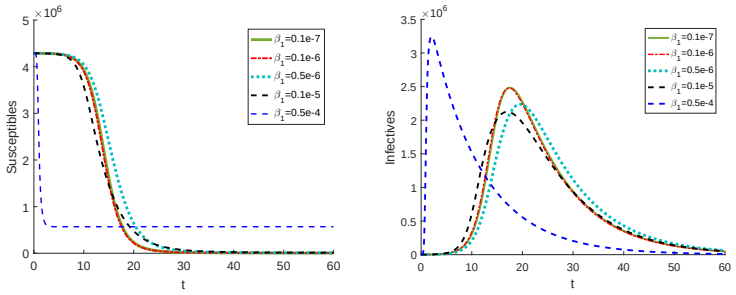
II) Keep $\beta = 1.63\times10^{-7}+0.1\times10^{-7}\xi$, $\gamma_1 = 0.01$ fixed and change γ_0 by taking γ_0=0.06, 0.07, 0.08, 0.09, 0.1.



(a) (b)

Figure 4.17: Mean of a) Susceptibles $\mathcal{S}(t)$ and b) Infectives I(t) is shown for different values of γ_0 using linear B-spline gPC.

III) Keep $\beta_0 = 1.63\times10^{-7}$, $\gamma = 0.1 + 0.01\xi^*$ fixed and take different values of β_1 as $\beta_1 = 0.1\times10^{-7},\ 0.1\times10^{-6},\ 0.5\times10^{-6},\ 0.1\times10^{-5},\ 0.5\times10^{-4}$.



(a) (b)

Figure 4.18: Mean of a) Susceptibles $\mathcal{S}(t)$ and b)Infectives I(t) is shown for different values of β_1 using linear B-spline gPC.

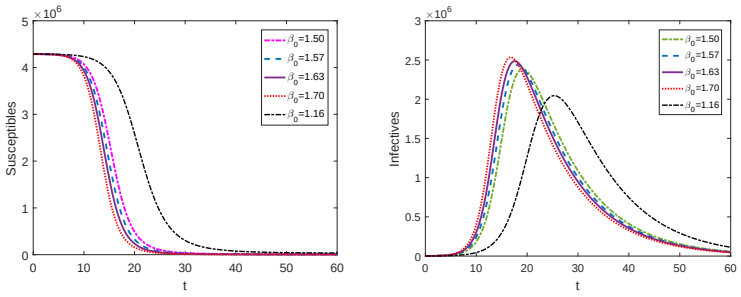
IV) Keep $\beta_1 = 0.1\times10^{-7}$, $\gamma = 0.1+0.01\xi^*$ fixed and change β_0 as β_0=0.163×10^{-7}, 0.150×10^{-7}, 0.157×10^{-7}, 0.17×10^{-7}.

4.4.2.2 Results and Discussion

The main findings for the Ebola outbreak are as follows:

1) We conclude that linear B-spline wavelets (Fig. 4.13) require fewer PCE order (P) than cubic B-spline wavelets (Fig. 4.15) to compute the solution.

2) As Ebola virus has effected a very large population, so, a minute variation in any of the parameters will lead to surprising results.



(a) (b)

Figure 4.19: Mean of a) Susceptibles $\mathcal{S}(t)$ and b) Infectives $\mathrm{I}(t)$ is shown for different values of β_0 using linear B-spline gPC.

3) In Fig. 4.16, we analyze that the solution is sensitive to the values of $\gamma_1 = 0.5, 1$. So, these two values should not be considered.

4) As we increase the value of β_1 (shown in Fig. 4.18), results seem to be less accurate whereas $\beta_1 = 0.5 \times 10^{-4}$ shows extremely unacceptable results. Therefore, we can say that the solution is highly sensitive to the changes in β than the changes in γ.

4.5 Conclusion

In this chapter, we have studied real-life epidemic outbreaks by utilizing B-spline wavelet-based gPC. It has been analyzed that wavelet functions of linear B-spline wavelets are showing negligible variations, so we have employed only scaling functions of cubic B-spline wavelets for comparison. Real data from British Medical Journal (Influenza) and WHO (Ebola) has been incorporated into the model and it has been concluded that the proposed method is approximating the school data given in the British Medical Journal appropriately. The number of scaling functions of B-spline wavelets increases as we increase the order of B-spline wavelets. Due to this fact, higher-order B-spline wavelets require more number of PCE to compute the solution. Thus, a linear B-spline wavelet basis shows faster accurate results as compared to cubic B-spline wavelets. With the help of SA, we found the most accurate values of the random parameters which produced the expected results. The solution is more sensitive to the changes made in the parameter β than the changes in γ. It has been analyzed that parameters $\beta = 0.00218 + 0.001\xi$, $\gamma = 0.441 + 0.001\xi^*$ approximate the influenza outbreak perfectly whereas for Ebola virus outbreak, the parameters $\beta = 1.63 \times 10^{-7} + 0.1 \times 10^{-7}\xi$, $\gamma = 0.1 + 0.01\xi^*$ gives the best results.

Chapter 5

An adaptive wavelet optimized finite difference B-spline polynomial chaos method for Burger's Equation

Wavelets, initiated by Grossman and Morlet in 1984, are known for their wonderful properties such as vanishing moments, wavelet decomposition, multiscale analysis, localization, fast wavelet transform, and data compression. These attractive properties are useful in numerically solving PDEs [239–244]. Further, there are two ways to solve a PDE numerically- one way is static node arrangement [245] which is created at the beginning of the computation and the other way is adaptive node arrangement which keeps on modifying itself in accordance with the numerical solution of PDE at various times [246–249]. The need of adaptive method arises when we wish to reduce the CPU cost by adjusting the quality of representation by catching those features which consists of the most important parts of the solution.

Adaptive schemes perform refinements where most needed in order to reduce the computational effort. Such local refinements plays a vital role when the system dynamics indicate steep dependencies on the random parameters. In fact, instead of using unnecessarily high order global PC expansions, a series of local low order expansions at controlled resolution levels are employed. There are also few other techniques available in literature for adaptivity [2] such as adaptive partitioning of random parameter space, generalized spectral decomposition method, a posterior error estimation for the variational problem. Maître et al. [88] applied a multi-resolution analysis approach to Galerkin projection schemes for treating discontinuous response surfaces by adaptively refining the multi-wavelet basis. Witteveen and Iaccarino [250] solved unsteady stochastic problems by means of multi-elements techniques and by employing the collocation simplex method. Adaptivity is implemented in the stochastic space in accordance with the regularity of the solution for all these approaches.

The motivation behind this method was to bring the best out of the wavelet methods in the field of UQ. Wavelet methods are generally known for their self adaptive nature which makes it a good choice for the numerical solution of a PDE. The self adaptivity property

comes from the good localization properties of wavelets which are seen both in space and frequency. Also, wavelet optimized finite difference method (WOFD) uses wavelets on irregular grids which is then utilized for the finite difference method [251, 252]. Therefore, it comes under the family of fitted mesh methods. In this chapter, we will execute adaptive wavelet optimized finite difference B-spline gPC to solve the PDEs with randomness.

5.1 Finite difference on an irregular grid

Let grid points be defined as

$$a = x_1 < x_2 < \ldots < x_{\mathcal{M}-1} < x_{\mathcal{M}} = b \tag{5.1.1}$$

which are not necessarily at a equal distance with $[a, b]$ being the domain. For approximating derivatives of u, we construct a Lagrangian interpolating polynomial through p points and differentiate it. Let us take odd $p \geq 3$ for the simplicity of the algorithm. Take $w = \frac{p-1}{2}$ and then define

$$u_I(x) = \sum_{k=i-w}^{i+w} u(x_k) \frac{\mathcal{P}_{w,i,k}(x)}{\mathcal{P}_{w,i,k}(x_k)}, \quad \text{where} \quad \mathcal{P}_{w,i,k}(x) = \sum_{\substack{l=i-w, \\ l \neq k}}^{i+w} (x - x_l) \tag{5.1.2}$$

Because of periodicity, we have $u_I(x_i) = u(x_i)$ for $i = 1, \ldots, \mathcal{M}$, i.e., u_I interpolates u at the grid points. Differentiating it d times, we get

$$u_I^d(x) = \sum_{k=i-w}^{i+w} u(x_k) \frac{\mathcal{P}^{(d)}_{w,i,k}(x)}{\mathcal{P}_{w,i,k}(x_k)} \tag{5.1.3}$$

Replacing x by x_i in Eq. (5.1.3) gives p point difference approximation for $u^{(d)}(x)$ centered at x_i. Let $\mathbf{u} = [u(x_1), u(x_2), \ldots, u(x_{\mathcal{M}})]$. The derivatives $u^{(d)}(x)$ can be approximated as follows:

$$u^{(d)} = \mathfrak{D}_p^{(d)} \mathbf{u}, \quad \text{where} \quad [\mathfrak{D}_p^{(d)}]_{i,k} = \frac{\mathcal{P}^{(d)}_{w,i,k}(x)}{\mathcal{P}_{w,i,k}(x_k)} \tag{5.1.4}$$

Now, for the first and second derivatives, we get

$$\mathcal{P}^{(1)}_{w,i,k}(x) = \frac{d}{dx} \mathcal{P}_{w,i,k}(x) = \sum_{\substack{l=i-w, \\ l \neq k}}^{i+w} \prod_{\substack{m=i-w, \\ m \neq k,l}}^{i+w} (x - x_m) \tag{5.1.5}$$

and

$$\mathcal{P}^{(2)}_{w,i,k}(x) = \frac{d^2}{dx^2}\mathcal{P}_{w,i,k}(x) = \sum_{\substack{l=i-w,\\ l\neq k}}^{i+w} \sum_{\substack{m=i-w,\\ m\neq k,l}}^{i+w} \prod_{\substack{n=i-w,\\ n\neq k,l,m}}^{i+w} (x - x_n) \tag{5.1.6}$$

After simplifying the above equations, we have the first derivatives as follows:

$$\frac{du(x_j)}{dx} = \frac{u(x_{j+1}) - u(x_{j-1})}{x_{j+1} - x_{j-1}},\ j = 1, 2, \ldots, \mathcal{M} \tag{5.1.7}$$

and second derivative as follows:

$$\begin{aligned}\frac{d^2u(x_j)}{dx^2} = &\frac{2u(x_{j-1})}{(x_{j-1} - x_j)(x_{j-1} - x_{j+1})} + \frac{2u(x_j)}{(x_j - x_{j-1})(x_j - x_{j+1})} \\ &+ \frac{2u(x_{j+1})}{(x_{j+1} - x_{j-1})(x_{j+1} - x_j)}, j = 1, 2, \ldots, \mathcal{M}\end{aligned}$$

Also, one should take care that the points x_0 and $x_{\mathcal{M}+1}$ are not included in the domain [253]. Throughout this paper, we will be dealing with periodic boundary conditions, so we will assume all the indices to be computed modulo $\mathcal{M} - 1$ ($u(a) = u(b)$), i.e., x_0 will be treated as $x_{\mathcal{M}-1}$ and $x_{\mathcal{M}+1}$ will be treated as x_2. Also suppose $x_1 - x_0 = x_2 - x_1$ and $x_{\mathcal{M}+1} - x_{\mathcal{M}} = x_{\mathcal{M}} - x_{\mathcal{M}-1}$ and first and second order FD matrices as $\mathfrak{D}^{(1)}$ and $\mathfrak{D}^{(2)}$. In simple words,

$$\begin{pmatrix} u'(x_1) \\ u'(x_2) \\ \vdots \\ \vdots \\ u'(x_{\mathcal{M}}) \end{pmatrix} = \mathfrak{D}^{(1)}_{\mathcal{M}\times\mathcal{M}} \begin{pmatrix} u(x_1) \\ u(x_2) \\ \vdots \\ \vdots \\ u(x_{\mathcal{M}}) \end{pmatrix} \tag{5.1.8}$$

where

$$\begin{aligned}\mathfrak{D}^{(1)}(1,2) = -\mathfrak{D}^{(1)}(1,\mathcal{M}-1) &= \frac{1}{2(x_2 - x_1)}, \\ \mathfrak{D}^{(1)}(\mathcal{M},2) = -\mathfrak{D}^{(1)}(\mathcal{M},\mathcal{M}-1) &= \frac{1}{2(x_{\mathcal{M}} - x_{\mathcal{M}-1})}, \\ \mathfrak{D}^{(1)}(\imath,\imath-1) = \mathfrak{D}^{(1)}(\imath,\imath+1) &= -\frac{1}{(x_{\imath+1} - x_{\imath-1})}, \\ \mathfrak{D}^{(1)}(\imath,j) &= 0,\ \forall \text{ other } \imath \text{ and } j\end{aligned} \tag{5.1.9}$$

On the other side, we have

$$\begin{pmatrix} u''(x_1) \\ u''(x_2) \\ \vdots \\ u''(x_{\mathcal{M}}) \end{pmatrix} = \mathfrak{D}^{(2)}_{\mathcal{M}\times\mathcal{M}} \begin{pmatrix} u(x_1) \\ u(x_2) \\ \vdots \\ u(x_{\mathcal{M}}) \end{pmatrix} \tag{5.1.10}$$

$$\begin{aligned} -\frac{1}{2}\mathfrak{D}^{(2)}(1,1) = \mathfrak{D}^{(2)}(1,2) = \mathfrak{D}^{(2)}(1,\mathcal{M}-1) &= \frac{1}{(x_2-x_1)^2}, \\ \mathfrak{D}^{(2)}(\mathcal{M},2) = \mathfrak{D}^{(2)}(\mathcal{M},\mathcal{M}-1) = -\frac{1}{2}\mathfrak{D}^{(2)}(\mathcal{M},\mathcal{M}) &= \frac{1}{(x_{\mathcal{M}}-x_{\mathcal{M}-1})^2}, \\ \mathfrak{D}^{(2)}(\imath,\imath-1) &= \frac{2}{(x_{\imath-1}-x_\imath)(x_{\imath-1}-x_{\imath+1})}, \quad 2 \le \imath \le \mathcal{M}-1 \\ \mathfrak{D}^{(2)}(\imath,\imath) &= \frac{2}{(x_\imath-x_{\imath-1})(x_\imath-x_{\imath+1})}, \quad 2 \le \imath \le \mathcal{M}-1 \\ \mathfrak{D}^{(2)}(\imath,\imath+1) &= \frac{2}{(x_{\imath+1}-x_{\imath-1})(x_{\imath+1}-x_\imath)}, \quad 2 \le \imath \le \mathcal{M}-1 \\ \mathfrak{D}^{(2)}(\imath,j) &= 0, \forall \text{ other } \imath \text{ and } j \end{aligned} \tag{5.1.11}$$

5.2 Grid Adaptation

While representing the solution using wavelet gPC, we generally have scaling and wavelet coefficients represented by $U^s_{J_0,k}$ and $U^w_{\jmath,k}$ respectively. The beauty of wavelets lies in the value of wavelet coefficients as these coefficients decrease rapidly when the function is smooth. Moreover, if the function experiences discontinuity in one of its derivatives, then these coefficients tend to decrease gradually in the proximity of discontinuity and exhibit rapid decay away from it. Therefore, they are known for identifying shocks in the numerical solution of a PDE. This means that the magnitude of wavelet coefficient plays a significant role in capturing the essential features of a solution. As we know, the wavelet representation of a stochastic process is given as

$$U(\xi \in [a,b]) = \sum_{k=-1}^{3} U^s_{J_0,k}\phi^s_{J_0,k}(\wp(\xi)) + \sum_{\jmath=J_0}^{J}\sum_{k=-1}^{2^\jmath-2} U^w_{\jmath,k}\psi^w_{\jmath,k}(\wp(\xi)).$$

We can discard wavelets with small coefficients, thus, retaining a good approximation. The wavelet coefficients $U^w_{\jmath,k}$ will be small unless the $U(\xi)$ has some fluctuation on the scale of $\jmath$ in the instant proximity of wavelets $\psi^w_{\jmath,k}(\wp(\xi))$. Further, we divide the above equation

according to the magnitude

$$U(\xi \in [a,b]) = U_{\geq\epsilon}(\xi) + U_{<\epsilon}(\xi) \tag{5.2.1}$$

where

$$U_{\geq\epsilon}(\xi \in [a,b]) = \sum_{k=-1}^{3} U^s_{J_0,k}\phi^s_{J_0,k}(\wp(\xi)) + \sum_{\jmath=J_0}^{J} \sum_{\substack{k=-1 \\ |U^w_{\jmath,k}|\geq\epsilon}}^{2^\jmath-2} U^w_{\jmath,k}\psi^w_{\jmath,k}(\wp(\xi)). \tag{5.2.2}$$

$$U_{<\epsilon}(\xi \in [a,b]) = \sum_{\substack{k=-1 \\ |U^w_{\jmath,k}|<\epsilon}}^{2^\jmath-2} U^w_{\jmath,k}\psi^w_{\jmath,k}(\wp(\xi)). \tag{5.2.3}$$

There are basically two techniques available in literature for adaptivity. One of them is standard adaptive technique which works as follows:

1. We need to initiate from finest level of resolution.
2. Then, on the basis of magnitude of the wavelet coefficients, we remove the grid points which are less than ϵ.

and the other one is modified adaptation technique

1. We initiate from the coarsest level of resolution.
2. Then, on the basis of magnitude of the wavelet coefficients, we add the grid points.

The algorithm of the grid generation is given below:

Algorithm 5.1 Adaptive grid generation using wavelet optimized B-spline gPC

1: Suppose $\mathcal{X}^c_k$ are the current coarse node arrangements and $\{u_k(x_j)\}_{j\in\mathcal{X}^c}$ is known where u_k are the coefficients with $k = 1, 2, \ldots, P$.
2: Use interpolation to compute $\{u_k(x_j)\}_{j\in\mathcal{X}^J}$ from $\{u_k(x_j)\}_{j\in\mathcal{X}^c}$ where $k = 0, 1, 2, \ldots, P$.
3: Obtain scaling and wavelet functions by applying wavelet transform on the expanded $u_k(x)$.
4: Start with finest grid $\mathcal{X}^J_k$ where $k = 0, 1, \ldots, P$.
5: Keep all the grid points corresponding to B-spline scaling functions intact.
6: Keep all the grid points $|U^w_{\jmath,k}| > \epsilon$ intact for every equation.
7: Delete all the other points.

Numerically, it is proven that adaptive node arrangements from both the methods are identically same [254]. Wavelet optimized methods can be collaborated with FD, finite

element and finite volume. Jameson [255, 256] introduced WOFD which uses wavelets to obtain the adaptive grid [257, 258]. Rather than expanding the solution through scaling or wavelet expansion, the wavelet transform is employed to identify where the FD grid must be coarsened or refined [259, 260]. Wavelets offers excellent grid selection mechanism such that sparse grids are arranged in smooth regions and fine grids are arranged in chaotic areas of the domain.

5.3 Numerical Results

In this section, we will apply the proposed method on three numerical problems- linear heat equation with uncertain ICs, non-linear Burger equation with uncertain ICs and non-linear Burger's equation with uncertain viscosity and different uncertain ICs. For the three test problems, we have considered periodic boundary conditions.

5.3.1 Stochastic Heat Equation

Let us first consider a simple linear example i.e., heat equation on a unit interval

$$\frac{\partial u}{\partial t} = \nu \frac{\partial^2 u}{\partial x^2}, \quad 0 \leq x \leq 1, \quad 0 \leq t \leq 0.5 \tag{5.3.1}$$

$$u(0,t) = u(1,t) \tag{5.3.2}$$

with random IC $u(x,0,\xi) = u_{init}(x,\xi)$ where

$$u_{init}(x,\xi) = \begin{cases} x + 0.05\xi, & 0 \leq x \leq 0.5 \\ 1 - x, & 0.5 \leq x \leq 1 \end{cases} \tag{5.3.3}$$

which has a discontinuous derivative at $x = 0.5$ and ξ is uniformly distributed over the interval $[-1, 1]$.

5.3.1.1 Method

The PCE representation of a solution is given by

$$u(x,t,\xi) = \sum_{i=1}^{P} u_i(x,t) W_i(\xi) \tag{5.3.4}$$

So, we will insert B-spline gPC i.e., Eq. (5.3.4) in Eq. (5.3.1),

$$\frac{\partial}{\partial t}\Big(\sum_{i=1}^{P} u_i(x,t)W_i(\xi)\Big) = \nu\frac{\partial^2}{\partial x^2}\Big(\sum_{i=1}^{P} u_i(x,t)W_i(\xi)\Big)$$

Now, applying Galerkin projection w.r.t $\tilde{W}_k(\xi)$

$$\begin{aligned}
\frac{\partial}{\partial t}\Big(\sum_{i=1}^{P} u_i(x,t)\langle W_i(\xi)\tilde{W}_k(\xi)\rangle\Big) &= \nu\frac{\partial^2}{\partial x^2}\Big(\sum_{i=1}^{P} u_i(x,t)\langle W_i(\xi)\tilde{W}_k(\xi)\rangle\Big) \\
\sum_{i=1}^{P}\frac{\partial u_i(x,t)}{\partial t}\langle W_i(\xi)\tilde{W}_k(\xi)\rangle &= \nu\sum_{i=1}^{P}\frac{\partial^2 u_i(x,t)}{\partial x^2}\langle W_i(\xi)\tilde{W}_k(\xi)\rangle \\
\frac{\partial u_k(x,t)}{\partial t}\langle W_i(\xi)\tilde{W}_k(\xi)\rangle &= \nu\frac{\partial^2 u_k(x,t)}{\partial x^2}\langle W_i(\xi)\tilde{W}_k(\xi)\rangle \quad \text{where} \quad k=1,\ldots,P \\
\frac{\partial u_k(x,t)}{\partial t} &= \underbrace{\nu\frac{\partial^2 u_k(x,t)}{\partial x^2}}_{Linear Part \quad \mathbf{L}}
\end{aligned} \tag{5.3.5}$$

For better understanding, let us take $P = 3$. So, discretizing Eq. (5.3.5) for $k = 1$, we get

$$\begin{aligned}
\frac{d\mathbf{u_1}(t)}{dt} &= \nu\mathbf{L}(\mathbf{u_1}(t)) \\
\frac{d\mathbf{u_2}(t)}{dt} &= \nu\mathbf{L}(\mathbf{u_2}(t)) \\
\frac{d\mathbf{u_3}(t)}{dt} &= \nu\mathbf{L}(\mathbf{u_3}(t))
\end{aligned} \tag{5.3.6}$$

where

$$\begin{aligned}
\mathbf{L}(\mathbf{u_1}(t)) &= diag(\mathfrak{D}^{(2)}\mathbf{u_1}(t)), \\
\mathbf{L}(\mathbf{u_2}(t)) &= diag(\mathfrak{D}^{(2)}\mathbf{u_2}(t)), \\
\mathbf{L}(\mathbf{u_3}(t)) &= diag(\mathfrak{D}^{(2)}\mathbf{u_3}(t)).
\end{aligned}$$

Applying CN scheme

$$\begin{aligned}
\frac{u_1^n - u_1^{n-1}}{\Delta t} &= \nu\mathbf{L}\left(\frac{u_1^n + u_1^{n-1}}{2}\right) \\
\frac{u_2^n - u_2^{n-1}}{\Delta t} &= \nu\mathbf{L}\left(\frac{u_2^n + u_2^{n-1}}{2}\right) \\
\frac{u_3^n - u_3^{n-1}}{\Delta t} &= \nu\mathbf{L}\left(\frac{u_3^n + u_3^{n-1}}{2}\right)
\end{aligned}$$

Simplifying the equations, we get

$$\left(I - \frac{\nu \mathbf{L} \Delta t}{2}\right) u_1^n = \left(I + \frac{\nu \mathbf{L} \Delta t}{2}\right) u_1^{n-1}$$
$$\left(I - \frac{\nu \mathbf{L} \Delta t}{2}\right) u_2^n = \left(I + \frac{\nu \mathbf{L} \Delta t}{2}\right) u_2^{n-1}$$
$$\left(I - \frac{\nu \mathbf{L} \Delta t}{2}\right) u_3^n = \left(I + \frac{\nu \mathbf{L} \Delta t}{2}\right) u_3^{n-1}$$

$$\mathcal{A} u_1^n = \mathcal{B} u_1^{n-1}$$
$$\mathcal{A} u_2^n = \mathcal{B} u_2^{n-1}$$
$$\mathcal{A} u_3^n = \mathcal{B} u_3^{n-1}$$

where

$$\mathcal{A} = I - \frac{\nu \mathbf{L} \Delta t}{2}$$
$$\mathcal{B} = I + \frac{\nu \mathbf{L} \Delta t}{2}$$

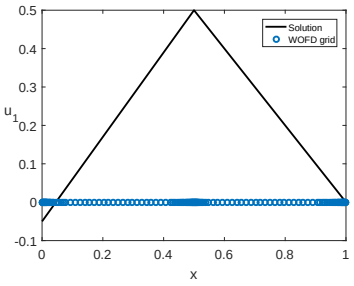
$u(x,0) = u_0(x)$, $\nu = 1 \quad \epsilon = 10^{-5}$						
Time	u_1	u_2	u_3	u_4	u_5	u_6
0	1024	1024	1024	1024	1024	1024
0.0625	136	136	136	136	136	136
0.1250	156	156	156	156	156	156
0.1875	126	126	126	126	126	126
0.25	101	101	101	101	101	101
0.3125	95	95	95	95	95	95
0.3750	92	92	92	92	92	92
0.4375	79	79	79	79	79	79
0.5	74	74	74	74	74	74

Table 5.1: Grid modifications of Test Problem 1 at different times

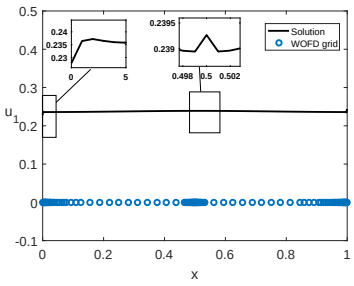
Therefore, we have

$$u_1^n = \mathcal{A}^{-1} \mathcal{B} u_1^{n-1}$$
$$u_2^n = \mathcal{A}^{-1} \mathcal{B} u_2^{n-1}$$
$$u_3^n = \mathcal{A}^{-1} \mathcal{B} u_3^{n-1}$$

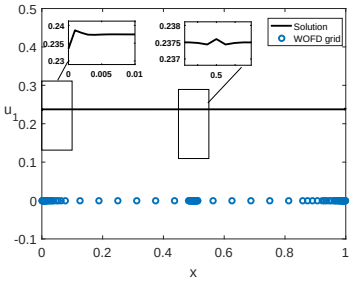
Similarly, we can apply the same for higher order P of PCE.



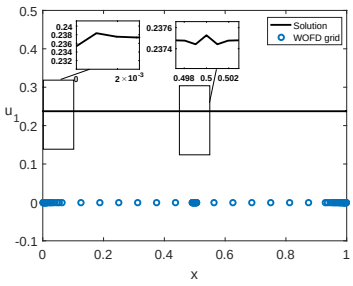
(a) $t = 0$



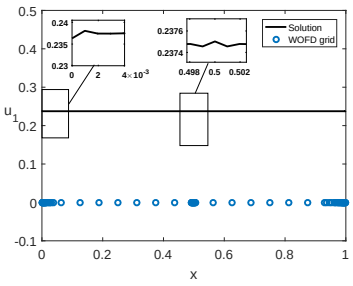
(b) $t = 0.125$



(c) $t = 0.25$

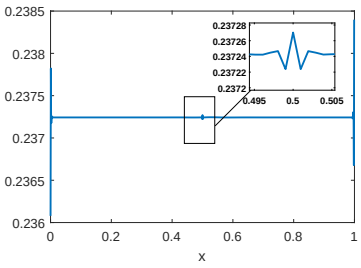


(d) $t = 0.375$

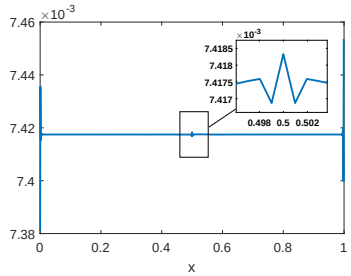


(e) $t = 0.5$

Figure 5.1: Solution of stochastic heat equation (coeff u_1) at $t = 0, 0.125, 0.25, 0.375, 0.5$ with $\nu = 1$ with $P = 6$, $m = 2^{10}$.



(a) Mean



(b) S.D.

Figure 5.2: Mean and S.D. of stochastic heat equation at $t = 0.5$ with $\nu = 1$, $P = 6$.

Fig. 5.1 displays the solution of the first coefficient and the associated adaptive grid at various time steps. We have magnified the solution as there were variations at the left

side, center and right side of the solution. We can notice from here that more points are being added near the variations. In Fig 5.2, we have demonstrated mean and S.D. plots of the solution of stochastic Burger's equation with uncertain ICs. Table 5.1 presents the grid modifications at different times for all the coefficients $u_1, u_2, u_3, u_4, u_5, u_6$.

5.3.2 Stochastic Burger's Equation

Consider non-linear Burger's equation

$$\frac{\partial u}{\partial t} = \nu \frac{\partial^2 u}{\partial x^2} - u \frac{\partial u}{\partial x} \tag{5.3.7}$$

with uncertain ICs

$$u(x, t = 0, \xi) = sin(4\pi x) + 0.05\xi \tag{5.3.8}$$

and periodic BCs

$$u(x = 0, t) = u(x = 1, t)$$

5.3.2.1 Method

Applying PCE in Eq. (5.3.8), we get

$$\sum_{i=1}^{P} u_i(x, 0) W_i(\xi) = sin(4\pi x) + 0.05\xi$$

After that, we employ Galerkin projection w.r.t $\tilde{W}_k$, we get

$$\sum_{i=1}^{P} u_i(x, 0) \langle W_i(\xi) \tilde{W}_k(\xi) \rangle = \left\langle \big(sin(4\pi x) + 0.05\xi\big) \tilde{W}_k(\xi) \right\rangle$$

Next, we insert Eq. (5.3.4) in Eq. (5.3.7), we get

$$\frac{\partial}{\partial t}\Big(\sum_{i=1}^{P} u_i(x, t) W_i(\xi) \Big) = \nu \frac{\partial^2}{\partial x^2}\Big(\sum_{i=1}^{P} u_i(x, t) W_i(\xi) \Big) - \sum_{i=1}^{P} u_i(x, t) W_i(\xi) \frac{\partial}{\partial x}\Big(\sum_{j=1}^{P} u_j(x, t) W_j(\xi) \Big)$$

Taking inner product

$$\begin{aligned}
\frac{\partial}{\partial t}\Big(\sum_{i=1}^{P} u_i(x,t)\langle W_i(\xi)\tilde{W}_k(\xi)\rangle\Big) &= \nu\frac{\partial^2}{\partial x^2}\Big(\sum_{i=1}^{P} u_i(x,t)\langle W_i(\xi)\tilde{W}_k(\xi)\rangle\Big) \\
&- \sum_{i=1}^{P}\sum_{j=1}^{P} u_i(x,t)\frac{\partial u_j(x,t)}{\partial x}\langle W_i(\xi)W_j(\xi)\tilde{W}_k(\xi)\rangle \\
\sum_{i=1}^{P}\frac{\partial u_i(x,t)}{\partial t}\langle W_i(\xi)\tilde{W}_k(\xi)\rangle &= \nu\sum_{i=1}^{P}\frac{\partial^2 u_i(x,t)}{\partial x^2}\langle W_i(\xi)\tilde{W}_k(\xi)\rangle \\
&- \sum_{i=1}^{P}\sum_{j=1}^{P} u_i(x,t)\frac{\partial u_j(x,t)}{\partial x}\langle W_i(\xi)W_j(\xi)\tilde{W}_k(\xi)\rangle \\
\frac{\partial u_k(x,t)}{\partial t} &= \underbrace{\nu\frac{\partial^2 u_k(x,t)}{\partial x^2}}_{Linear\ \ Part\ \mathbf{L}} - \underbrace{\sum_{i=1}^{P}\sum_{j=1}^{P} u_i(x,t)\frac{\partial u_j(x,t)}{\partial x}\mathscr{C}_{ijk}}_{Non-linear\ \ Part\ \mathbf{N}},\ k=1,\ldots,P
\end{aligned}$$

Suppose $P=3$ be the chaos order, then we have

$$\frac{\partial u_1(x,t)}{\partial t} = \nu\frac{\partial^2 u_1(x,t)}{\partial x^2}$$

$$-\begin{pmatrix} u_1 & u_2 & u_3 \end{pmatrix}\begin{pmatrix} \langle W_1(\xi)W_1(\xi)\tilde{W}_1(\xi)\rangle & \langle W_1(\xi)W_2(\xi)\tilde{W}_1(\xi)\rangle & \langle W_1(\xi)W_3(\xi)\tilde{W}_1(\xi)\rangle \\ \langle W_2(\xi)W_1(\xi)\tilde{W}_1(\xi)\rangle & \langle W_2(\xi)W_2(\xi)\tilde{W}_1(\xi)\rangle & \langle W_2(\xi)W_3(\xi)\tilde{W}_1(\xi)\rangle \\ \langle W_3(\xi)W_1(\xi)\tilde{W}_1(\xi)\rangle & \langle W_3(\xi)W_2(\xi)\tilde{W}_1(\xi)\rangle & \langle W_3(\xi)W_3(\xi)\tilde{W}_1(\xi)\rangle \end{pmatrix}\begin{pmatrix} \frac{\partial u_1(x,t)}{\partial x} \\ \frac{\partial u_2(x,t)}{\partial x} \\ \frac{\partial u_3(x,t)}{\partial x} \end{pmatrix} \tag{5.3.9}$$

$$\frac{\partial u_2(x,t)}{\partial t} = \nu\frac{\partial^2 u_2(x,t)}{\partial x^2}$$

$$-\begin{pmatrix} u_1 & u_2 & u_3 \end{pmatrix}\begin{pmatrix} \langle W_1(\xi)W_1(\xi)\tilde{W}_2(\xi)\rangle & \langle W_1(\xi)W_2(\xi)\tilde{W}_2(\xi)\rangle & \langle W_1(\xi)W_3(\xi)\tilde{W}_2(\xi)\rangle \\ \langle W_2(\xi)W_1(\xi)\tilde{W}_2(\xi)\rangle & \langle W_2(\xi)W_2(\xi)\tilde{W}_2(\xi)\rangle & \langle W_2(\xi)W_3(\xi)\tilde{W}_2(\xi)\rangle \\ \langle W_3(\xi)W_1(\xi)\tilde{W}_2(\xi)\rangle & \langle W_3(\xi)W_2(\xi)\tilde{W}_2(\xi)\rangle & \langle W_3(\xi)W_3(\xi)\tilde{W}_2(\xi)\rangle \end{pmatrix}\begin{pmatrix} \frac{\partial u_1(x,t)}{\partial x} \\ \frac{\partial u_2(x,t)}{\partial x} \\ \frac{\partial u_3(x,t)}{\partial x} \end{pmatrix} \tag{5.3.10}$$

$$\frac{\partial u_3(x,t)}{\partial t} = \nu\frac{\partial^2 u_3(x,t)}{\partial x^2}$$

$$-\begin{pmatrix} u_1 & u_2 & u_3 \end{pmatrix}\begin{pmatrix} \langle W_1(\xi)W_1(\xi)\tilde{W}_3(\xi)\rangle & \langle W_1(\xi)W_2(\xi)\tilde{W}_3(\xi)\rangle & \langle W_1(\xi)W_3(\xi)\tilde{W}_3(\xi)\rangle \\ \langle W_2(\xi)W_1(\xi)\tilde{W}_3(\xi)\rangle & \langle W_2(\xi)W_2(\xi)\tilde{W}_3(\xi)\rangle & \langle W_2(\xi)W_3(\xi)\tilde{W}_3(\xi)\rangle \\ \langle W_3(\xi)W_1(\xi)\tilde{W}_3(\xi)\rangle & \langle W_3(\xi)W_2(\xi)\tilde{W}_3(\xi)\rangle & \langle W_3(\xi)W_3(\xi)\tilde{W}_3(\xi)\rangle \end{pmatrix}\begin{pmatrix} \frac{\partial u_1(x,t)}{\partial x} \\ \frac{\partial u_2(x,t)}{\partial x} \\ \frac{\partial u_3(x,t)}{\partial x} \end{pmatrix} \tag{5.3.11}$$

Discretizing Eq. (5.3.9) to get

$$\begin{aligned}\frac{d\mathbf{u_1}(t)}{dt} = \mathbf{L}(\mathbf{u_1}(t)) \quad - \quad & \Big[\mathbf{N}(\mathbf{u_1}(t))\mathbf{u_1}(t)\langle W_1W_1\tilde{W}_1\rangle + \mathbf{N}(\mathbf{u_2}(t))\mathbf{u_1}(t)\langle W_1W_2\tilde{W}_1\rangle \\ + \quad & \mathbf{N}(\mathbf{u_3}(t))\mathbf{u_1}(t)\langle W_1W_3\tilde{W}_1\rangle + \mathbf{N}(\mathbf{u_1}(t))\mathbf{u_2}(t)\langle W_2W_1\tilde{W}_1\rangle \\ + \quad & \mathbf{N}(\mathbf{u_2}(t))\mathbf{u_2}(t)\langle W_2W_2\tilde{W}_1\rangle + \mathbf{N}(\mathbf{u_3}(t))\mathbf{u_2}(t)\langle W_2W_3\tilde{W}_1\rangle \\ + \quad & \mathbf{N}(\mathbf{u_1}(t))\mathbf{u_3}(t)\langle W_3W_1\tilde{W}_1\rangle + \mathbf{N}(\mathbf{u_2}(t))\mathbf{u_3}(t)\langle W_3W_2\tilde{W}_1\rangle \\ + \quad & \mathbf{N}(\mathbf{u_3}(t))\mathbf{u_3}(t)\langle W_3W_3\tilde{W}_1\rangle\Big]\end{aligned}$$

where

$$\begin{aligned}\mathbf{L}(\mathbf{u_1}(t)) &= \nu(\mathfrak{D}^{(2)}\mathbf{u_1}(t)) \\ \mathbf{N}(\mathbf{u_1}(t)) &= diag(\mathfrak{D}^{(1)}\mathbf{u_1}(t)) \\ \mathbf{N}(\mathbf{u_2}(t)) &= diag(\mathfrak{D}^{(1)}\mathbf{u_2}(t)) \\ \mathbf{N}(\mathbf{u_3}(t)) &= diag(\mathfrak{D}^{(1)}\mathbf{u_3}(t))\end{aligned}$$

$$\begin{aligned}\frac{d\mathbf{u_1}(t)}{dt} = \mathbf{L}\mathbf{u_1}(t) &- \Big[\mathbf{N}(\mathbf{u_1}(t))\langle W_1W_1\tilde{W}_1\rangle + \mathbf{N}(\mathbf{u_2}(t))\langle W_1W_2\tilde{W}_1\rangle + \mathbf{N}(\mathbf{u_3}(t))\langle W_1W_3\tilde{W}_1\rangle\Big]\mathbf{u_1}(t) \\ &- \Big[\mathbf{N}(\mathbf{u_1}(t))\langle W_2W_1\tilde{W}_1\rangle + \mathbf{N}(\mathbf{u_2}(t))\langle W_2W_2\tilde{W}_1\rangle + \mathbf{N}(\mathbf{u_3}(t))\langle W_2W_3\tilde{W}_1\rangle\Big]\mathbf{u_2}(t) \\ &- \Big[\mathbf{N}(\mathbf{u_1}(t))\langle W_3W_1\tilde{W}_1\rangle + \mathbf{N}(\mathbf{u_2}(t))\langle W_3W_2\tilde{W}_1\rangle + \mathbf{N}(\mathbf{u_3}(t))\langle W_3W_3\tilde{W}_1\rangle\Big]\mathbf{u_3}(t)\end{aligned}$$

Applying CN scheme

$$\begin{aligned}\frac{u_1^n - u_1^{n-1}}{\Delta t} = \mathbf{L}\left(\frac{u_1^n + u_1^{n-1}}{2}\right) &- \left[\mathbf{N}\left(\frac{u_1^n + u_1^{n-1}}{2}\right)\langle W_1W_1\tilde{W}_1\rangle + \mathbf{N}\left(\frac{u_2^n + u_2^{n-1}}{2}\right)\langle W_1W_2\tilde{W}_1\rangle\right. \\ &\left. + \mathbf{N}\left(\frac{u_3^n + u_3^{n-1}}{2}\right)\langle W_1W_3\tilde{W}_1\rangle\right]\left(\frac{u_1^n + u_1^{n-1}}{2}\right) - \left[\mathbf{N}\left(\frac{u_1^n + u_1^{n-1}}{2}\right)\langle W_2W_1\tilde{W}_1\rangle\right. \\ &\left. + \mathbf{N}\left(\frac{u_2^n + u_2^{n-1}}{2}\right)\langle W_2W_2\tilde{W}_1\rangle + \mathbf{N}\left(\frac{u_3^n + u_3^{n-1}}{2}\right)\langle W_2W_3\tilde{W}_1\rangle\right]\left(\frac{u_2^n + u_2^{n-1}}{2}\right) \\ &- \left[\mathbf{N}\left(\frac{u_1^n + u_1^{n-1}}{2}\right)\langle W_3W_1\tilde{W}_1\rangle + \mathbf{N}\left(\frac{u_2^n + u_2^{n-1}}{2}\right)\langle W_3W_2\tilde{W}_1\rangle\right. \\ &\left. + \mathbf{N}\left(\frac{u_3^n + u_3^{n-1}}{2}\right)\langle W_3W_3\tilde{W}_1\rangle\right]\left(\frac{u_3^n + u_3^{n-1}}{2}\right)\end{aligned}$$

$$
\begin{aligned}
\mathcal{A}u_1^n = \mathcal{B}u_1^{n-1} - \Delta t\Bigg\{ & \left[\mathbf{N}\left(\frac{u_1^n + u_1^{n-1}}{2}\right)\langle W_1 W_1 \tilde{W}_1\rangle + \mathbf{N}\left(\frac{u_2^n + u_2^{n-1}}{2}\right)\langle W_1 W_2 \tilde{W}_1\rangle \right. \\
& \left. + \mathbf{N}\left(\frac{u_3^n + u_3^{n-1}}{2}\right)\langle W_1 W_3 \tilde{W}_1\rangle\right]\left(\frac{u_1^n + u_1^{n-1}}{2}\right) + \left[\mathbf{N}\left(\frac{u_1^n + u_1^{n-1}}{2}\right)\langle W_2 W_1 \tilde{W}_1\rangle \right. \\
& \left. + \mathbf{N}\left(\frac{u_2^n + u_2^{n-1}}{2}\right)\langle W_2 W_2 \tilde{W}_1\rangle + \mathbf{N}\left(\frac{u_3^n + u_3^{n-1}}{2}\right)\langle W_2 W_3 \tilde{W}_1\rangle\right]\left(\frac{u_2^n + u_2^{n-1}}{2}\right) \\
& + \left[\mathbf{N}\left(\frac{u_1^n + u_1^{n-1}}{2}\right)\langle W_3 W_1 \tilde{W}_1\rangle + \mathbf{N}\left(\frac{u_2^n + u_2^{n-1}}{2}\right)\langle W_3 W_2 \tilde{W}_1\rangle \right. \\
& \left. + \mathbf{N}\left(\frac{u_3^n + u_3^{n-1}}{2}\right)\langle W_3 W_3 \tilde{W}_1\rangle\right]\left(\frac{u_3^n + u_3^{n-1}}{2}\right)\Bigg\}
\end{aligned}
$$

where

$$
\begin{aligned}
\mathcal{A} &= I - \frac{\mathbf{L}\Delta t}{2} \\
\mathcal{B} &= I + \frac{\mathbf{L}\Delta t}{2}
\end{aligned}
\tag{5.3.12}
$$

$$
\begin{aligned}
u_1^n \;=\; & \mathcal{A}^{-1}\Bigg[\mathcal{B}u_1^{n-1} - \Delta t\Bigg\{ \left[\mathbf{N}\left(\frac{u_1^n + u_1^{n-1}}{2}\right)\langle W_1 W_1 \tilde{W}_1\rangle + \mathbf{N}\left(\frac{u_2^n + u_2^{n-1}}{2}\right)\langle W_1 W_2 \tilde{W}_1\rangle \right. \\
& \left. + \mathbf{N}\left(\frac{u_3^n + u_3^{n-1}}{2}\right)\langle W_1 W_3 \tilde{W}_1\rangle\right]\left(\frac{u_1^n + u_1^{n-1}}{2}\right) + \left[\mathbf{N}\left(\frac{u_1^n + u_1^{n-1}}{2}\right)\langle W_2 W_1 \tilde{W}_1\rangle \right. \\
& \left. + \mathbf{N}\left(\frac{u_2^n + u_2^{n-1}}{2}\right)\langle W_2 W_2 \tilde{W}_1\rangle + \mathbf{N}\left(\frac{u_3^n + u_3^{n-1}}{2}\right)\langle W_2 W_3 \tilde{W}_1\rangle\right]\left(\frac{u_2^n + u_2^{n-1}}{2}\right) \\
& + \left[\mathbf{N}\left(\frac{u_1^n + u_1^{n-1}}{2}\right)\langle W_3 W_1 \tilde{W}_1\rangle + \mathbf{N}\left(\frac{u_2^n + u_2^{n-1}}{2}\right)\langle W_3 W_2 \tilde{W}_1\rangle \right. \\
& \left. + \mathbf{N}\left(\frac{u_3^n + u_3^{n-1}}{2}\right)\langle W_3 W_3 \tilde{W}_1\rangle\right]\left(\frac{u_3^n + u_3^{n-1}}{2}\right)\Bigg\}\Bigg]
\end{aligned}
\tag{5.3.13}
$$

Similarly, we have

$$
\begin{aligned}
u_2^n \;=\; & \mathcal{A}^{-1}\Bigg[\mathcal{B}u_2^{n-1} - \Delta t\Bigg\{ \left[\mathbf{N}\left(\frac{u_1^n + u_1^{n-1}}{2}\right)\langle W_1 W_1 \tilde{W}_2\rangle + \mathbf{N}\left(\frac{u_2^n + u_2^{n-1}}{2}\right)\langle W_1 W_2 \tilde{W}_2\rangle \right. \\
& \left. + \mathbf{N}\left(\frac{u_3^n + u_3^{n-1}}{2}\right)\langle W_1 W_3 \tilde{W}_2\rangle\right]\left(\frac{u_1^n + u_1^{n-1}}{2}\right) + \left[\mathbf{N}\left(\frac{u_1^n + u_1^{n-1}}{2}\right)\langle W_2 W_1 \tilde{W}_2\rangle \right. \\
& \left. + \mathbf{N}\left(\frac{u_2^n + u_2^{n-1}}{2}\right)\langle W_2 W_2 \tilde{W}_2\rangle + \mathbf{N}\left(\frac{u_3^n + u_3^{n-1}}{2}\right)\langle W_2 W_3 \tilde{W}_2\rangle\right]\left(\frac{u_2^n + u_2^{n-1}}{2}\right) \\
& + \left[\mathbf{N}\left(\frac{u_1^n + u_1^{n-1}}{2}\right)\langle W_3 W_1 \tilde{W}_2\rangle + \mathbf{N}\left(\frac{u_2^n + u_2^{n-1}}{2}\right)\langle W_3 W_2 \tilde{W}_2\rangle \right. \\
& \left. + \mathbf{N}\left(\frac{u_3^n + u_3^{n-1}}{2}\right)\langle W_3 W_3 \tilde{W}_2\rangle\right]\left(\frac{u_3^n + u_3^{n-1}}{2}\right)\Bigg\}\Bigg]
\end{aligned}
\tag{5.3.14}
$$

$$
\begin{aligned}
u_3^n &= \mathcal{A}^{-1}\Bigg[\mathcal{B}u_3^{n-1} - \Delta t\Bigg\{\Bigg[\mathbf{N}\Big(\frac{u_1^n+u_1^{n-1}}{2}\Big)\langle W_1W_1\tilde{W}_3\rangle + \mathbf{N}\Big(\frac{u_2^n+u_2^{n-1}}{2}\Big)\langle W_1W_2\tilde{W}_3\rangle \\
&+ \mathbf{N}\Big(\frac{u_3^n+u_3^{n-1}}{2}\Big)\langle W_1W_3\tilde{W}_3\rangle\Bigg]\Big(\frac{u_1^n+u_1^{n-1}}{2}\Big) + \Bigg[\mathbf{N}\Big(\frac{u_1^n+u_1^{n-1}}{2}\Big)\langle W_2W_1\tilde{W}_3\rangle \\
&+ \mathbf{N}\Big(\frac{u_2^n+u_2^{n-1}}{2}\Big)\langle W_2W_2\tilde{W}_3\rangle + \mathbf{N}\Big(\frac{u_3^n+u_3^{n-1}}{2}\Big)\langle W_2W_3\tilde{W}_3\rangle\Bigg]\Big(\frac{u_2^n+u_2^{n-1}}{2}\Big) \\
&+ \Bigg[\mathbf{N}\Big(\frac{u_1^n+u_1^{n-1}}{2}\Big)\langle W_3W_1\tilde{W}_3\rangle + \mathbf{N}\Big(\frac{u_2^n+u_2^{n-1}}{2}\Big)\langle W_3W_2\tilde{W}_3\rangle \\
&+ \mathbf{N}\Big(\frac{u_3^n+u_3^{n-1}}{2}\Big)\langle W_3W_3\tilde{W}_3\rangle\Bigg]\Big(\frac{u_3^n+u_3^{n-1}}{2}\Big)\Bigg\}\Bigg] \qquad (5.3.15)
\end{aligned}
$$

An iterative process is required to solve these Eqs. (5.3.13), (5.3.14), (5.3.15)

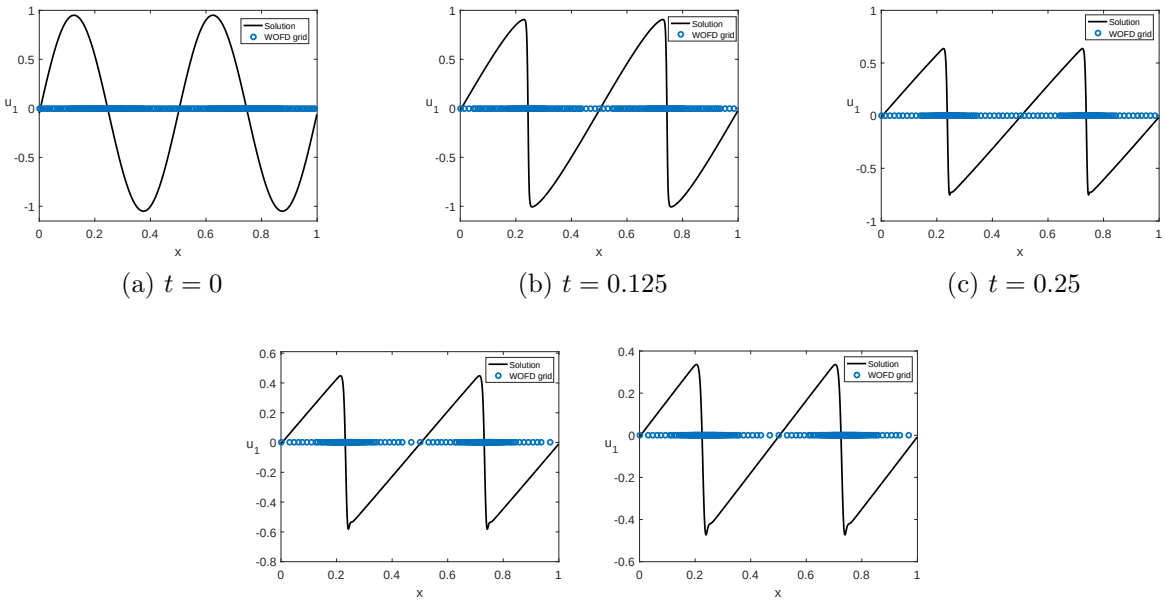
1) First of all, let $u_1^{n,(0)} = u_1^{n-1}$, $u_2^{n,(0)} = u_2^{n-1}$, $u_3^{n,(0)} = u_3^{n-1}$ and

2) Iterate until convergence is achieved.

$u(x,0) = sin(4\pi x) + 0.05\xi$, $\nu = 0.002$ $\epsilon = 10^{-5}$						
Time	u_1	u_2	u_3	u_4	u_5	u_6
0	1024	1024	1024	1024	1024	1024
0.0625	282	278	280	278	280	370
0.1250	280	280	282	296	303	366
0.1875	230	228	230	248	238	220
0.25	202	196	210	242	216	228
0.3125	214	198	216	226	222	242
0.3750	224	200	218	230	236	250
0.4375	228	200	220	240	244	264

Table 5.2: Grid modifications of Test Problem 2 at different times

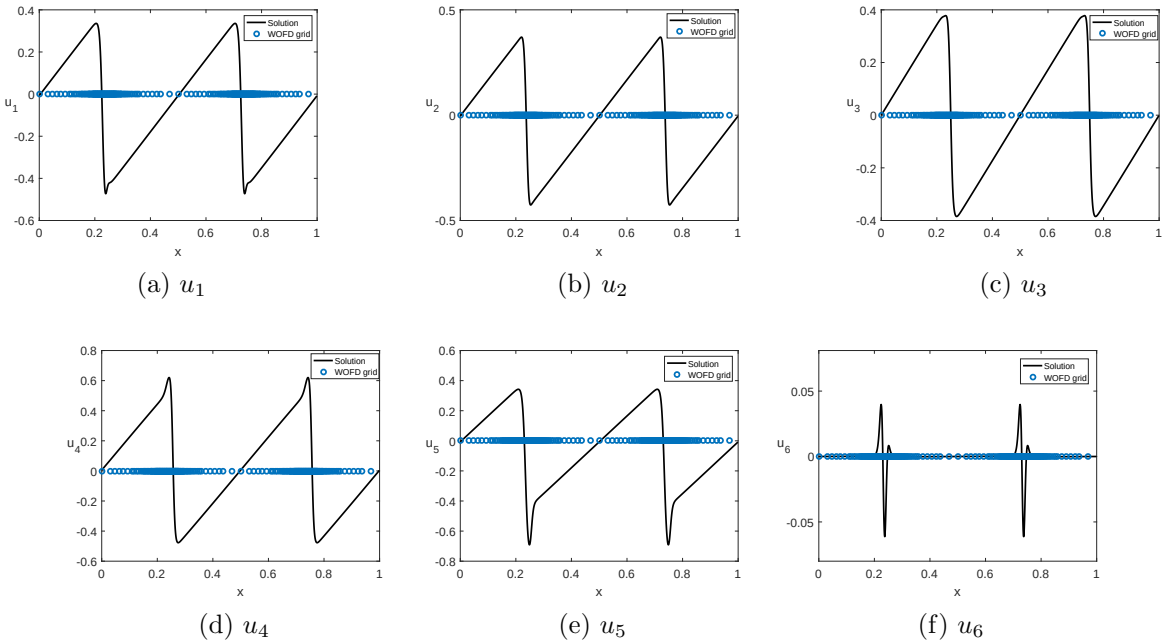
$$
\begin{aligned}
u_1^{n,(q+1)} &= \mathcal{A}^{-1}\Bigg[\mathcal{B}u_1^{n-1} - \Delta t\Bigg\{\Bigg[\mathbf{N}\Big(\frac{u_1^{n,(q)}+u_1^{n-1}}{2}\Big)\langle W_1W_1\tilde{W}_1\rangle + \mathbf{N}\Big(\frac{u_2^{n,(q)}+u_2^{n-1}}{2}\Big)\langle W_1W_2\tilde{W}_1\rangle \\
&+ \mathbf{N}\Big(\frac{u_3^{n,(q)}+u_3^{n-1}}{2}\Big)\langle W_1W_3\tilde{W}_1\rangle\Bigg]\Big(\frac{u_1^{n,(q)}+u_1^{n-1}}{2}\Big) + \Bigg[\mathbf{N}\Big(\frac{u_1^{n,(q)}+u_1^{n-1}}{2}\Big)\langle W_2W_1\tilde{W}_1\rangle \\
&+ \mathbf{N}\Big(\frac{u_2^{n,(q)}+u_2^{n-1}}{2}\Big)\langle W_2W_2\tilde{W}_1\rangle + \mathbf{N}\Big(\frac{u_3^{n,(q)}+u_3^{n-1}}{2}\Big)\langle W_2W_3\tilde{W}_1\rangle\Bigg]\Big(\frac{u_2^{n,(q)}+u_2^{n-1}}{2}\Big) \\
&+ \Bigg[\mathbf{N}\Big(\frac{u_1^{n,(q)}+u_1^{n-1}}{2}\Big)\langle W_3W_1\tilde{W}_1\rangle + \mathbf{N}\Big(\frac{u_2^{n,(q)}+u_2^{n-1}}{2}\Big)\langle W_3W_2\tilde{W}_1\rangle \\
&+ \mathbf{N}\Big(\frac{u_3^{n,(q)}+u_3^{n-1}}{2}\Big)\langle W_3W_3\tilde{W}_1\rangle\Bigg]\Big(\frac{u_3^{n,(q)}+u_3^{n-1}}{2}\Big)\Bigg\}\Bigg] \quad \text{where} \quad q = 0,1,2,\ldots
\end{aligned}
$$



(a) $t = 0$ (b) $t = 0.125$ (c) $t = 0.25$

(d) $t = 0.375$ (e) $t = 0.5$

Figure 5.3: (Test problem 2) Solution of stochastic Burgers' equation (coeff u_1) at $t = 0, 0.0625, 0.25, 0.5$ with uncertain ICs $u(x, 0) = sin(4\pi x) + 0.05\xi$ with $P = 6$, $m = 2^{10}$.



(a) u_1 (b) u_2 (c) u_3

(d) u_4 (e) u_5 (f) u_6

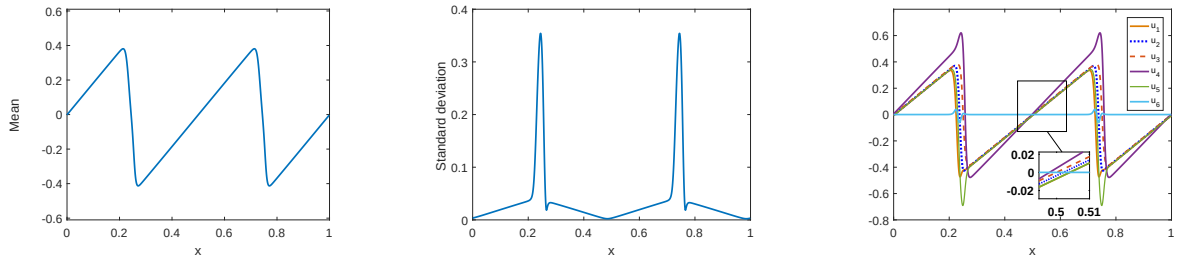
Figure 5.4: (Test problem 2) Solution of stochastic Burgers' equation at $t = 0.5$ (coeff $u_1, u_2, u_3, u_4, u_5, u_6$) with uncertain ICs $u(x, 0) = sin(4\pi x) + 0.05\xi$ with $P = 6$, $m = 2^{10}$.

So, we perform LU decomposition on matrix $\mathcal{A}$ and subsequent backward and forward substitutions in each step. Similarly, we can solve other two equations for coefficients u_2 and u_3 as

$$u_2^{n,(q+1)} = \mathcal{A}^{-1}\Bigg[\mathcal{B}u_2^{n-1} - \Delta t\Bigg\{\Bigg[\mathbf{N}\left(\frac{u_1^{n,(q)}+u_1^{n-1}}{2}\right)\langle W_1W_1\tilde{W}_2\rangle + \mathbf{N}\left(\frac{u_2^{n,(q)}+u_2^{n-1}}{2}\right)\langle W_1W_2\tilde{W}_2\rangle$$
$$+\mathbf{N}\left(\frac{u_3^{n,(q)}+u_3^{n-1}}{2}\right)\langle W_1W_3\tilde{W}_2\rangle\Bigg]\left(\frac{u_1^{n,(q)}+u_1^{n-1}}{2}\right) + \Bigg[\mathbf{N}\left(\frac{u_1^{n,(q)}+u_1^{n-1}}{2}\right)\langle W_2W_1\tilde{W}_2\rangle$$
$$+\mathbf{N}\left(\frac{u_2^{n,(q)}+u_2^{n-1}}{2}\right)\langle W_2W_2\tilde{W}_2\rangle + \mathbf{N}\left(\frac{u_3^{n,(q)}+u_3^{n-1}}{2}\right)\langle W_2W_3\tilde{W}_2\rangle\Bigg]\left(\frac{u_2^{n,(q)}+u_2^{n-1}}{2}\right)$$
$$+\Bigg[\mathbf{N}\left(\frac{u_1^{n,(q)}+u_1^{n-1}}{2}\right)\langle W_3W_1\tilde{W}_2\rangle + \mathbf{N}\left(\frac{u_2^{n,(q)}+u_2^{n-1}}{2}\right)\langle W_3W_2\tilde{W}_2\rangle$$
$$+\mathbf{N}\left(\frac{u_3^{n,(q)}+u_3^{n-1}}{2}\right)\langle W_3W_3\tilde{W}_2\rangle\Bigg]\left(\frac{u_3^{n,(q)}+u_3^{n-1}}{2}\right)\Bigg\}\Bigg]$$

$$u_3^{n,(q+1)} = \mathcal{A}^{-1}\Bigg[\mathcal{B}u_3^{n-1} - \Delta t\Bigg\{\Bigg[\mathbf{N}\left(\frac{u_1^{n,(q)}+u_1^{n-1}}{2}\right)\langle W_1W_1\tilde{W}_3\rangle + \mathbf{N}\left(\frac{u_2^{n,(q)}+u_2^{n-1}}{2}\right)\langle W_1W_2\tilde{W}_3\rangle$$
$$+\mathbf{N}\left(\frac{u_3^{n,(q)}+u_3^{n-1}}{2}\right)\langle W_1W_3\tilde{W}_3\rangle\Bigg]\left(\frac{u_1^{n,(q)}+u_1^{n-1}}{2}\right) + \Bigg[\mathbf{N}\left(\frac{u_1^{n,(q)}+u_1^{n-1}}{2}\right)\langle W_2W_1\tilde{W}_3\rangle$$
$$+\mathbf{N}\left(\frac{u_2^{n,(q)}+u_2^{n-1}}{2}\right)\langle W_2W_2\tilde{W}_3\rangle + \mathbf{N}\left(\frac{u_3^{n,(q)}+u_3^{n-1}}{2}\right)\langle W_2W_3\tilde{W}_3\rangle\Bigg]\left(\frac{u_2^{n,(q)}+u_2^{n-1}}{2}\right)$$
$$+\Bigg[\mathbf{N}\left(\frac{u_1^{n,(q)}+u_1^{n-1}}{2}\right)\langle W_3W_1\tilde{W}_3\rangle + \mathbf{N}\left(\frac{u_2^{n,(q)}+u_2^{n-1}}{2}\right)\langle W_3W_2\tilde{W}_3\rangle$$
$$+\mathbf{N}\left(\frac{u_3^{n,(q)}+u_3^{n-1}}{2}\right)\langle W_3W_3\tilde{W}_3\rangle\Bigg]\left(\frac{u_3^{n,(q)}+u_3^{n-1}}{2}\right)\Bigg\}\Bigg]$$

Fig. 5.3 demonstrates the solution of the first coefficient along with their associated adaptive grid



(a) $t = 0.5$ (b) $t = 0.5$ (c) $t = 0.5$

Figure 5.5: (Test problem 2) a)Mean and b) S.D. of stochastic Burgers' equation at $t = 0.5$ with uncertain ICs $u(x, 0) = sin(4\pi x) + 0.05\xi$ with $P = 6$, $m = 2^{10}$ c) All coefficients.

at various times. It has been noticed that more points are being added close to the variations. In Fig. 5.4, we have the solution of all coefficients with their adaptive grids at $t = 0.5$. In Fig 5.5a and b, we have respectively displayed mean and S.D. plots of the solution of stochastic heat equation with uncertain ICs. It has been observed that S.D is high where there are sharp

variations in the solution. Also, Table 5.2 illustrates the grid modifications at various times for all the coefficients $u_1, u_2, u_3, u_4, u_5, u_6$. It is clearly seen from this table that different coefficients require different number of grid points. The coefficient u_6 requires more number of grid points as this coefficient contains wavelet function which will capture the sharp variations.

5.3.3 Stochastic Burger's Equation with uncertain viscosity and initial conditions

Now, again considering the same Burger's equation as in Eq. (5.3.7) but now we will consider ν to be random which means

$$\nu = \nu_0 + \nu_1 \xi \tag{5.3.16}$$

where ξ is uniformly distributed over the interval $[-1, 1]$. Moreover, we consider randomness in ICs as well

$$u(x, t = 0, \xi) = sin(2\pi x) + 0.01\xi \tag{5.3.17}$$

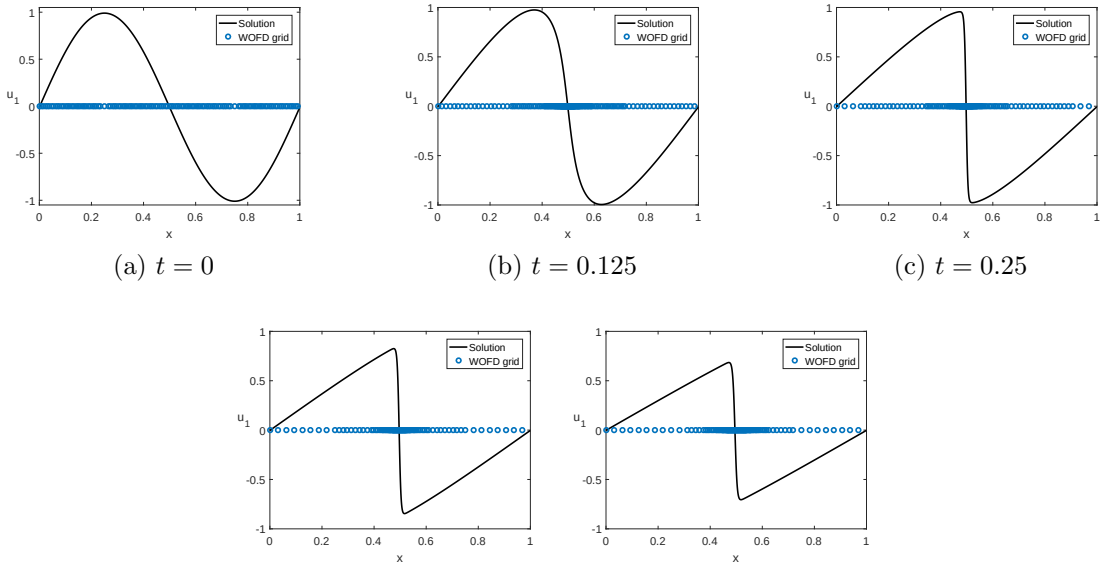
Since, ν is random, so the calculations will become a little hard in comparison to the previous examples. By taking uncertain ν, the representation of Eq. (5.3.7) becomes

$$\begin{aligned}
\frac{\partial}{\partial t}\Big(\sum_{i=1}^{P} u_i(x,t)W_i(\xi)\Big) &= \sum_{i=1}^{P} \nu_i W_i(\xi)\frac{\partial^2}{\partial x^2}\Big(\sum_{j=1}^{P} u_j(x,t)W_j(\xi)\Big) \\
&- \sum_{i=1}^{P} u_i(x,t)W_i(\xi)\frac{\partial}{\partial x}\Big(\sum_{j=1}^{P} u_j(x,t)W_j(\xi)\Big) \\
\frac{\partial}{\partial t}\Big(\sum_{i=1}^{P} u_i(x,t)W_i(\xi)\Big) &= \sum_{i=1}^{P}\sum_{j=1}^{P} \nu_i \frac{\partial^2}{\partial x^2}\Big(u_j(x,t)W_i(\xi)W_j(\xi)\Big) \\
&- \sum_{i=1}^{P}\sum_{j=1}^{P} u_i(x,t)\frac{\partial}{\partial x}\Big(u_j(x,t)W_i(\xi)W_j(\xi)\Big)
\end{aligned}$$

Taking Galerkin projection w.r.t $\tilde{W}_k(\xi)$, we get

$$\begin{aligned}
\sum_{i=1}^{P} \frac{\partial u_i(x,t)}{\partial t}\langle W_i(\xi)\tilde{W}_k(\xi)\rangle &= \sum_{i=1}^{P}\sum_{j=1}^{P} \nu_i \frac{\partial^2 u_j(x,t)}{\partial x^2}\langle W_i(\xi)W_j(\xi)\tilde{W}_k(\xi)\rangle \\
&- \sum_{i=1}^{P}\sum_{j=1}^{P} u_i(x,t)\frac{\partial u_j(x,t)}{\partial x}\langle W_i(\xi)W_j(\xi)\tilde{W}_k(\xi)\rangle
\end{aligned}$$

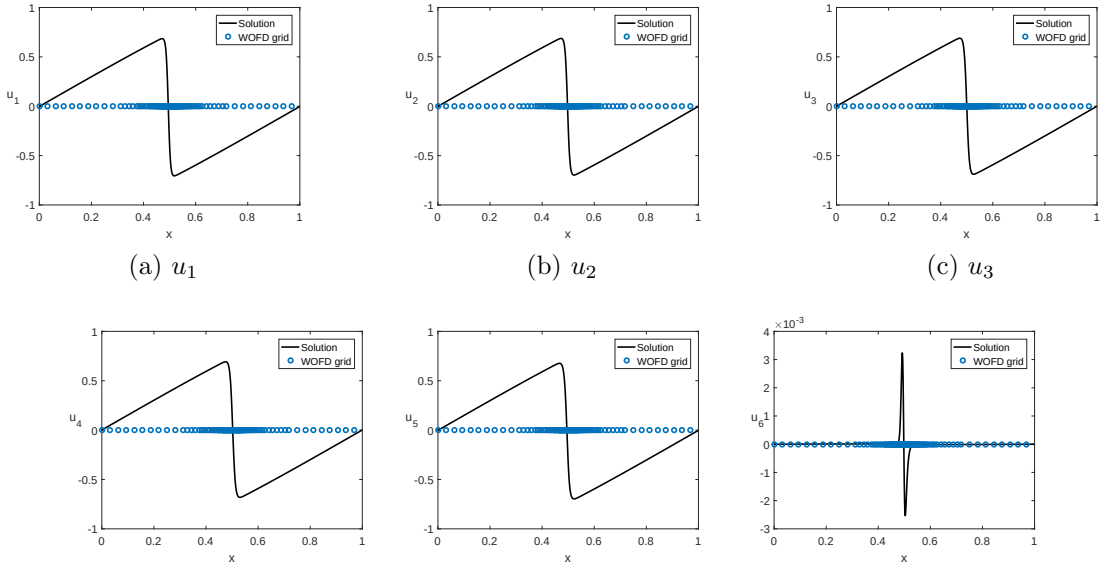
$$\frac{\partial u_k(x,t)}{\partial t} = \sum_{i=1}^{P}\sum_{j=1}^{P} \nu_i \frac{\partial^2 u_j(x,t)}{\partial x^2} \mathscr{C}_{ijk} - \sum_{i=1}^{P}\sum_{j=1}^{P} u_i(x,t) \frac{\partial u_j(x,t)}{\partial x} \mathscr{C}_{ijk}$$



(a) $t = 0$ (b) $t = 0.125$ (c) $t = 0.25$

(d) $t = 0.375$ (e) $t = 0.5$

Figure 5.6: Solution of stochastic Burgers' equation with $P = 6$ (coeff u_1) at $t = 0, 0.125, 0.25, 0.375, 0.5$.



(a) u_1 (b) u_2 (c) u_3

(d) u_4 (e) u_5 (f) u_6

Figure 5.7: Solution of stochastic Burgers' equation (all coeffs) at $t = 0.5$ with $P = 6$.

Suppose $P = 3$ be the chaos order, then for $k = 1$, we have

$$\frac{\partial u_1(x,t)}{\partial t} = \underbrace{\begin{pmatrix} \nu_1 & \nu_2 & \nu_3 \end{pmatrix} \begin{pmatrix} \langle W_1(\xi)W_1(\xi)\tilde{W}_1(\xi)\rangle & \langle W_1(\xi)W_2(\xi)\tilde{W}_1(\xi)\rangle & \langle W_1(\xi)W_3(\xi)\tilde{W}_1(\xi)\rangle \\ \langle W_2(\xi)W_1(\xi)\tilde{W}_1(\xi)\rangle & \langle W_2(\xi)W_2(\xi)\tilde{W}_1(\xi)\rangle & \langle W_2(\xi)W_3(\xi)\tilde{W}_1(\xi)\rangle \\ \langle W_3(\xi)W_1(\xi)\tilde{W}_1(\xi)\rangle & \langle W_3(\xi)W_2(\xi)\tilde{W}_1(\xi)\rangle & \langle W_3(\xi)W_3(\xi)\tilde{W}_1(\xi)\rangle \end{pmatrix} \begin{pmatrix} \frac{\partial^2 u_1(x,t)}{\partial x^2} \\ \frac{\partial^2 u_2(x,t)}{\partial x^2} \\ \frac{\partial^2 u_3(x,t)}{\partial x^2} \end{pmatrix}}_{Linear Part}$$
$$\underbrace{- \begin{pmatrix} u_1 & u_2 & u_3 \end{pmatrix} \begin{pmatrix} \langle W_1(\xi)W_1(\xi)\tilde{W}_1(\xi)\rangle & \langle W_1(\xi)W_2(\xi)\tilde{W}_1(\xi)\rangle & \langle W_1(\xi)W_3(\xi)\tilde{W}_1(\xi)\rangle \\ \langle W_2(\xi)W_1(\xi)\tilde{W}_1(\xi)\rangle & \langle W_2(\xi)W_2(\xi)\tilde{W}_1(\xi)\rangle & \langle W_2(\xi)W_3(\xi)\tilde{W}_1(\xi)\rangle \\ \langle W_3(\xi)W_1(\xi)\tilde{W}_1(\xi)\rangle & \langle W_3(\xi)W_2(\xi)\tilde{W}_1(\xi)\rangle & \langle W_3(\xi)W_3(\xi)\tilde{W}_1(\xi)\rangle \end{pmatrix} \begin{pmatrix} \frac{\partial u_1(x,t)}{\partial x} \\ \frac{\partial u_2(x,t)}{\partial x} \\ \frac{\partial u_3(x,t)}{\partial x} \end{pmatrix}}_{Non-Linear Part} \tag{5.3.18}$$

Similarly, we can obtain the equations for u_2 and u_3. Discretizing eq. (5.3.3) to get

$$\begin{aligned} \frac{d\mathbf{u_1}(t)}{dt} = \Big[& \mathbf{L}(\mathbf{u_1}(t))\nu_1\langle W_1W_1\tilde{W}_1\rangle + \mathbf{L}(\mathbf{u_2}(t))\nu_1\langle W_1W_2\tilde{W}_1\rangle + \mathbf{L}(\mathbf{u_3}(t))\nu_1\langle W_1W_3\tilde{W}_1\rangle \\ & + \mathbf{L}(\mathbf{u_1}(t))\nu_2\langle W_2W_1\tilde{W}_1\rangle + \mathbf{L}(\mathbf{u_2}(t))\nu_2\langle W_2W_2\tilde{W}_1\rangle + \mathbf{L}(\mathbf{u_3}(t))\nu_2\langle W_2W_3\tilde{W}_1\rangle \\ & + \mathbf{L}(\mathbf{u_1}(t))\nu_3\langle W_3W_1\tilde{W}_1\rangle + \mathbf{L}(\mathbf{u_2}(t))\nu_3\langle W_3W_2\tilde{W}_1\rangle + \mathbf{L}(\mathbf{u_3}(t))\nu_3\langle W_3W_3\tilde{W}_1\rangle \Big] \\ - \Big[& \mathbf{N}(\mathbf{u_1}(t))\mathbf{u_1}(t)\langle W_1W_1\tilde{W}_1\rangle + \mathbf{N}(\mathbf{u_2}(t))\mathbf{u_1}(t)\langle W_1W_2\tilde{W}_1\rangle + \mathbf{N}(\mathbf{u_3}(t))\mathbf{u_1}(t)\langle W_1W_3\tilde{W}_1\rangle \\ & + \mathbf{N}(\mathbf{u_1}(t))\mathbf{u_2}(t)\langle W_2W_1\tilde{W}_1\rangle + \mathbf{N}(\mathbf{u_2}(t))\mathbf{u_2}(t)\langle W_2W_2\tilde{W}_1\rangle + \mathbf{N}(\mathbf{u_3}(t))\mathbf{u_2}(t)\langle W_2W_3\tilde{W}_1\rangle \\ & + \mathbf{N}(\mathbf{u_1}(t))\mathbf{u_3}(t)\langle W_3W_1\tilde{W}_1\rangle + \mathbf{N}(\mathbf{u_2}(t))\mathbf{u_3}(t)\langle W_3W_2\tilde{W}_1\rangle + \mathbf{N}(\mathbf{u_3}(t))\mathbf{u_3}(t)\langle W_3W_3\tilde{W}_1\rangle \Big] \end{aligned}$$

where

$$\mathbf{L}(\mathbf{u_1}(t)) = diag(\mathfrak{D}^{(2)}\mathbf{u_1}(t)) \quad \mathbf{L}(\mathbf{u_2}(t)) = diag(\mathfrak{D}^{(2)}\mathbf{u_2}(t)) \quad \mathbf{L}(\mathbf{u_3}(t)) = diag(\mathfrak{D}^{(2)}\mathbf{u_3}(t))$$
$$\mathbf{N}(\mathbf{u_1}(t)) = diag(\mathfrak{D}^{(1)}\mathbf{u_1}(t)) \quad \mathbf{N}(\mathbf{u_2}(t)) = diag(\mathfrak{D}^{(1)}\mathbf{u_2}(t)) \quad \mathbf{N}(\mathbf{u_3}(t)) = diag(\mathfrak{D}^{(1)}\mathbf{u_3}(t))$$

$\nu = 0.002 + 0.0005\zeta \quad \epsilon = 10^{-5}$						
Time	u_1	u_2	u_3	u_4	u_5	u_6
0	1024	1024	1024	1024	1024	1024
0.0625	124	124	124	124	124	124
0.1250	160	160	161	161	161	195
0.1875	169	169	170	169	172	200
0.25	168	168	170	178	172	200
0.3125	142	146	148	153	153	163
0.3750	133	134	136	138	138	151
0.4375	128	131	133	135	133	150

Table 5.3: Grid modifications for Test Problem 3 at different times

Applying CN scheme as earlier, we get

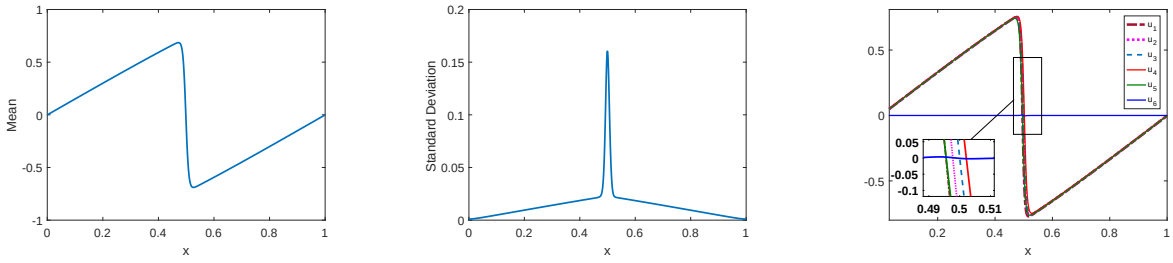
$$\frac{u_1^n - u_1^{n-1}}{\Delta t} = \Bigg[\mathbf{L}\Big(\frac{u_1^n + u_1^{n-1}}{2}\Big)\langle W_1 W_1 \tilde{W}_1\rangle + \mathbf{L}\Big(\frac{u_2^n + u_2^{n-1}}{2}\Big)\langle W_1 W_2 \tilde{W}_1\rangle$$
$$+ \mathbf{L}\Big(\frac{u_3^n + u_3^{n-1}}{2}\Big)\langle W_1 W_3 \tilde{W}_1\rangle\Bigg]\nu_1 + \Bigg[\mathbf{L}\Big(\frac{u_1^n + u_1^{n-1}}{2}\Big)\langle W_2 W_1 \tilde{W}_1\rangle + \mathbf{L}\Big(\frac{u_2^n + u_2^{n-1}}{2}\Big)\langle W_2 W_2 \tilde{W}_1\rangle$$
$$+ \mathbf{L}\Big(\frac{u_3^n + u_3^{n-1}}{2}\Big)\langle W_2 W_3 \tilde{W}_1\rangle\Bigg]\nu_2 + \Bigg[\mathbf{L}\Big(\frac{u_1^n + u_1^{n-1}}{2}\Big)\langle W_3 W_1 \tilde{W}_1\rangle + \mathbf{L}\Big(\frac{u_2^n + u_2^{n-1}}{2}\Big)\langle W_3 W_2 \tilde{W}_1\rangle$$
$$+ \mathbf{L}\Big(\frac{u_3^n + u_3^{n-1}}{2}\Big)\langle W_3 W_3 \tilde{W}_1\rangle\Bigg]\nu_3 - \Bigg[\mathbf{N}\Big(\frac{u_1^n + u_1^{n-1}}{2}\Big)\langle W_1 W_1 \tilde{W}_1\rangle + \mathbf{N}\Big(\frac{u_2^n + u_2^{n-1}}{2}\Big)\langle W_1 W_2 \tilde{W}_1\rangle$$
$$+ \mathbf{N}\Big(\frac{u_3^n + u_3^{n-1}}{2}\Big)\langle W_1 W_3 \tilde{W}_1\rangle\Bigg]\Big(\frac{u_1^n + u_1^{n-1}}{2}\Big) - \Bigg[\mathbf{N}\Big(\frac{u_1^n + u_1^{n-1}}{2}\Big)\langle W_2 W_1 \tilde{W}_1\rangle$$
$$+ \mathbf{N}\Big(\frac{u_2^n + u_2^{n-1}}{2}\Big)\langle W_2 W_2 \tilde{W}_1\rangle + \mathbf{N}\Big(\frac{u_3^n + u_3^{n-1}}{2}\Big)\langle W_2 W_3 \tilde{W}_1\rangle\Bigg]\Big(\frac{u_2^n + u_2^{n-1}}{2}\Big)$$
$$- \Bigg[\mathbf{N}\Big(\frac{u_1^n + u_1^{n-1}}{2}\Big)\langle W_3 W_1 \tilde{W}_1\rangle + \mathbf{N}\Big(\frac{u_2^n + u_2^{n-1}}{2}\Big)\langle W_3 W_2 \tilde{W}_1\rangle$$
$$+ \mathbf{N}\Big(\frac{u_3^n + u_3^{n-1}}{2}\Big)\langle W_3 W_3 \tilde{W}_1\rangle\Bigg]\Big(\frac{u_3^n + u_3^{n-1}}{2}\Big)$$

$$\begin{aligned}
\mathcal{A}_{u_1} u_1^n &= \mathfrak{B}_{u_1} u_1^{n-1} + \Delta t\Bigg[\mathbf{L}\Big(\frac{u_2^n + u_2^{n-1}}{2}\Big)\nu_1\langle W_1 W_2 \tilde{W}_1\rangle + \mathbf{L}\Big(\frac{u_3^n + u_3^{n-1}}{2}\Big)\nu_1\langle W_1 W_3 \tilde{W}_1\rangle \\
&+ \mathbf{L}\Big(\frac{u_2^n + u_2^{n-1}}{2}\Big)\nu_2\langle W_2 W_2 \tilde{W}_1\rangle + \mathbf{L}\Big(\frac{u_3^n + u_3^{n-1}}{2}\Big)\nu_2\langle W_2 W_3 \tilde{W}_1\rangle \\
&+ \mathbf{L}\Big(\frac{u_2^n + u_2^{n-1}}{2}\Big)\nu_3\langle W_3 W_2 \tilde{W}_1\rangle + \mathbf{L}\Big(\frac{u_3^n + u_3^{n-1}}{2}\Big)\nu_3\langle W_3 W_3 \tilde{W}_1\rangle\Bigg] \\
&- \Delta t\Bigg\{\Bigg[\mathbf{N}\Big(\frac{u_1^n + u_1^{n-1}}{2}\Big)\langle W_1 W_1 \tilde{W}_1\rangle + \mathbf{N}\Big(\frac{u_2^n + u_2^{n-1}}{2}\Big)\langle W_1 W_2 \tilde{W}_1\rangle \\
&+ \mathbf{N}\Big(\frac{u_3^n + u_3^{n-1}}{2}\Big)\langle W_1 W_3 \tilde{W}_1\rangle\Bigg]\Big(\frac{u_1^n + u_1^{n-1}}{2}\Big) \\
&+ \Bigg[\mathbf{N}\Big(\frac{u_1^n + u_1^{n-1}}{2}\Big)\langle W_2 W_1 \tilde{W}_1\rangle + \mathbf{N}\Big(\frac{u_2^n + u_2^{n-1}}{2}\Big)\langle W_2 W_2 \tilde{W}_1\rangle \\
&+ \mathbf{N}\Big(\frac{u_3^n + u_3^{n-1}}{2}\Big)\langle W_2 W_3 \tilde{W}_1\rangle\Bigg]\Big(\frac{u_2^n + u_2^{n-1}}{2}\Big) \\
&+ \Bigg[\mathbf{N}\Big(\frac{u_1^n + u_1^{n-1}}{2}\Big)\langle W_3 W_1 \tilde{W}_1\rangle + \mathbf{N}\Big(\frac{u_2^n + u_2^{n-1}}{2}\Big)\langle W_3 W_2 \tilde{W}_1\rangle \\
&+ \mathbf{N}\Big(\frac{u_3^n + u_3^{n-1}}{2}\Big)\langle W_3 W_3 \tilde{W}_1\rangle\Bigg]\Big(\frac{u_3^n + u_3^{n-1}}{2}\Big)\Bigg\}
\end{aligned}$$

where

$$\mathcal{A}_{u_1} = I - \frac{\mathbf{L}\Delta t\nu_1\langle W_1 W_1 \tilde{W}_1\rangle}{2} - \frac{\mathbf{L}\Delta t\nu_2\langle W_2 W_1 \tilde{W}_1\rangle}{2} - \frac{\mathbf{L}\Delta t\nu_3\langle W_3 W_1 \tilde{W}_1\rangle}{2}$$
$$\mathfrak{B}_{u_1} = I + \frac{\mathbf{L}\Delta t\nu_1\langle W_1 W_1 \tilde{W}_1\rangle}{2} + \frac{\mathbf{L}\Delta t\nu_2\langle W_2 W_1 \tilde{W}_1\rangle}{2} + \frac{\mathbf{L}\Delta t\nu_3\langle W_3 W_1 \tilde{W}_1\rangle}{2}$$

We can have equations for u_2 and u_3 in the similar manner such that $\mathcal{A}_{u_2}, \mathcal{A}_{u_3}, \mathfrak{B}_{u_2}, \mathfrak{B}_{u_3}$ will



(a) $t = 0.5$ (b) $t = 0.5$ (c) $t = 0.4375$

Figure 5.8: Mean and S.D. of stochastic Burgers' equation at $t = 0.5$ with $P = 6$.

ϵ	u_1	u_2	u_3	u_4	u_5	u_6
0	1024	1024	1024	1024	1024	1024
10^{-2}	54	54	54	54	54	85
10^{-3}	74	72	73	86	72	114
10^{-4}	134	137	136	136	136	150
10^{-5}	128	131	133	135	133	150

Table 5.4: (Test problem 3) Grid points with different values of ϵ

ϵ	10^{-2}	10^{-3}	10^{-4}	10^{-5}
CPU(ϵ)	4.937	6.015	8.234	10.125
Θ	56.32	46.23	33.77	27.464

Table 5.5: (Test Problem 3) Performance of the proposed method

look like

$$\mathcal{A}_{u_2} = I - \frac{\mathbf{L}\Delta t\nu_1\langle W_1W_2\tilde{W}_2\rangle}{2} - \frac{\mathbf{L}\Delta t\nu_2\langle W_2W_2\tilde{W}_2\rangle}{2} - \frac{\mathbf{L}\Delta t\nu_3\langle W_3W_2\tilde{W}_2\rangle}{2}$$
$$\mathcal{A}_{u_3} = I - \frac{\mathbf{L}\Delta t\nu_1\langle W_1W_3\tilde{W}_3\rangle}{2} - \frac{\mathbf{L}\Delta t\nu_2\langle W_2W_3\tilde{W}_3\rangle}{2} - \frac{\mathbf{L}\Delta t\nu_3\langle W_3W_3\tilde{W}_3\rangle}{2}$$
$$\mathfrak{B}_{u_2} = I + \frac{\mathbf{L}\Delta t\nu_1\langle W_1W_2\tilde{W}_2\rangle}{2} + \frac{\mathbf{L}\Delta t\nu_2\langle W_2W_2\tilde{W}_2\rangle}{2} + \frac{\mathbf{L}\Delta t\nu_3\langle W_3W_2\tilde{W}_2\rangle}{2}$$
$$\mathfrak{B}_{u_3} = I + \frac{\mathbf{L}\Delta t\nu_1\langle W_1W_3\tilde{W}_3\rangle}{2} + \frac{\mathbf{L}\Delta t\nu_2\langle W_2W_3\tilde{W}_3\rangle}{2} + \frac{\mathbf{L}\Delta t\nu_3\langle W_3W_3\tilde{W}_3\rangle}{2}$$

$$
\begin{aligned}
\mathcal{A}_{u_2}u_2^n &= \mathfrak{B}_{u_2}u_2^{n-1} + \Delta t\Bigg[\mathbf{L}\Big(\frac{u_1^n+u_1^{n-1}}{2}\Big)\nu_1\langle W_1W_1\tilde{W}_2\rangle + \mathbf{L}\Big(\frac{u_3^n+u_3^{n-1}}{2}\Big)\nu_1\langle W_1W_3\tilde{W}_2\rangle \\
&+ \mathbf{L}\Big(\frac{u_1^n+u_1^{n-1}}{2}\Big)\nu_2\langle W_2W_1\tilde{W}_2\rangle + \mathbf{L}\Big(\frac{u_3^n+u_3^{n-1}}{2}\Big)\nu_2\langle W_2W_3\tilde{W}_2\rangle \\
&+ \mathbf{L}\Big(\frac{u_1^n+u_1^{n-1}}{2}\Big)\nu_3\langle W_3W_1\tilde{W}_2\rangle + \mathbf{L}\Big(\frac{u_3^n+u_3^{n-1}}{2}\Big)\nu_3\langle W_3W_3\tilde{W}_2\rangle\Bigg] \\
&- \Delta t\Bigg\{\Bigg[\mathbf{N}\Big(\frac{u_1^n+u_1^{n-1}}{2}\Big)\langle W_1W_1\tilde{W}_2\rangle + \mathbf{N}\Big(\frac{u_2^n+u_2^{n-1}}{2}\Big)\langle W_1W_2\tilde{W}_2\rangle \\
&+ \mathbf{N}\Big(\frac{u_3^n+u_3^{n-1}}{2}\Big)\langle W_1W_3\tilde{W}_2\rangle\Bigg]\Big(\frac{u_1^n+u_1^{n-1}}{2}\Big) \\
&+ \Bigg[\mathbf{N}\Big(\frac{u_1^n+u_1^{n-1}}{2}\Big)\langle W_2W_1\tilde{W}_2\rangle + \mathbf{N}\Big(\frac{u_2^n+u_2^{n-1}}{2}\Big)\langle W_2W_2\tilde{W}_2\rangle \\
&+ \mathbf{N}\Big(\frac{u_3^n+u_3^{n-1}}{2}\Big)\langle W_2W_3\tilde{W}_2\rangle\Bigg]\Big(\frac{u_2^n+u_2^{n-1}}{2}\Big) \\
&+ \Bigg[\mathbf{N}\Big(\frac{u_1^n+u_1^{n-1}}{2}\Big)\langle W_3W_1\tilde{W}_2\rangle + \mathbf{N}\Big(\frac{u_2^n+u_2^{n-1}}{2}\Big)\langle W_3W_2\tilde{W}_2\rangle \\
&+ \mathbf{N}\Big(\frac{u_3^n+u_3^{n-1}}{2}\Big)\langle W_3W_3\tilde{W}_2\rangle\Bigg]\Big(\frac{u_3^n+u_3^{n-1}}{2}\Big)\Bigg\}
\end{aligned}
$$

$$
\begin{aligned}
\mathcal{A}_{u_3}u_3^n &= \mathfrak{B}_{u_3}{u_3}^{n-1} + \Delta t\Bigg[\mathbf{L}\Big(\frac{u_1^n+u_1^{n-1}}{2}\Big)\nu_1\langle W_1W_1\tilde{W}_3\rangle + \mathbf{L}\Big(\frac{u_2^n+u_2^{n-1}}{2}\Big)\nu_1\langle W_1W_2\tilde{W}_3\rangle \\
&+ \mathbf{L}\Big(\frac{u_1^n+u_1^{n-1}}{2}\Big)\nu_2\langle W_2W_1\tilde{W}_3\rangle + \mathbf{L}\Big(\frac{u_2^n+u_2^{n-1}}{2}\Big)\nu_2\langle W_2W_2\tilde{W}_3\rangle \\
&+ \mathbf{L}\Big(\frac{u_1^n+u_1^{n-1}}{2}\Big)\nu_3\langle W_3W_1\tilde{W}_3\rangle + \mathbf{L}\Big(\frac{u_2^n+u_2^{n-1}}{2}\Big)\nu_3\langle W_3W_2\tilde{W}_3\rangle\Bigg] \\
&- \Delta t\Bigg\{\Bigg[\mathbf{N}\Big(\frac{u_1^n+u_1^{n-1}}{2}\Big)\langle W_1W_1\tilde{W}_3\rangle + \mathbf{N}\Big(\frac{u_2^n+u_2^{n-1}}{2}\Big)\langle W_1W_2\tilde{W}_3\rangle \\
&+ \mathbf{N}\Big(\frac{u_3^n+u_3^{n-1}}{2}\Big)\langle W_1W_3\tilde{W}_3\rangle\Bigg]\Big(\frac{u_1^n+u_1^{n-1}}{2}\Big) + \Bigg[\mathbf{N}\Big(\frac{u_1^n+u_1^{n-1}}{2}\Big)\langle W_2W_1\tilde{W}_3\rangle \\
&+ \mathbf{N}\Big(\frac{u_2^n+u_2^{n-1}}{2}\Big)\langle W_2W_2\tilde{W}_3\rangle + \mathbf{N}\Big(\frac{u_3^n+u_3^{n-1}}{2}\Big)\langle W_2W_3\tilde{W}_3\rangle\Bigg]\Big(\frac{u_2^n+u_2^{n-1}}{2}\Big) \\
&+ \Bigg[\mathbf{N}\Big(\frac{u_1^n+u_1^{n-1}}{2}\Big)\langle W_3W_1\tilde{W}_3\rangle + \mathbf{N}\Big(\frac{u_2^n+u_2^{n-1}}{2}\Big)\langle W_3W_2\tilde{W}_3\rangle \\
&+ \mathbf{N}\Big(\frac{u_3^n+u_3^{n-1}}{2}\Big)\langle W_3W_3\tilde{W}_3\rangle\Bigg]\Big(\frac{u_3^n+u_3^{n-1}}{2}\Big)\Bigg\}
\end{aligned}
$$

Fig. 5.6 illustrates the solution of the first coefficient along with their associated adaptive grid at various times. Also, we can notice in this figure that more points are being added near the variations. In Fig. 5.7, we have the solution of all coefficients with their adaptive grids at $t = 0.5$. In Fig 5.8a and b, we have respectively given mean and S.D. plots of the solution of stochastic Burger equation. It has been observed that S.D is high near $x = 0.5$ i.e. near sharp variations in the solution. Also, Table 5.3 presents the grid modifications at various times for all the coefficients

$u_1, u_2, u_3, u_4, u_5, u_6$. It is clearly seen from this table that different coefficients require different number of grid points. The coefficient u_6 requires more number of grid points as this coefficient contains wavelet function which will capture the sharp variations. In table 5.4, we have discussed the number of grid points for different coefficients by taking different values of ϵ. As a means of depicting the efficiency of developed method, we compared the CPU time taken for ϵ and $\epsilon = 0$. Moreover, we have considered efficiency coefficient $\Theta = \frac{CPU(\epsilon=0)}{CPU(\epsilon)}$. The efficiency of the adaptive algorithm has been identified by the value of Θ which increases on increasing the ϵ. Higher the value of Θ, the more efficient is the adaptive algorithm.

5.4 Conclusion

In this chapter, adaptive B-spline optimized finite difference polynomial chaos is proposed to solve PDEs with uncertainty. B-spline polynomial chaos method is used to handle the randomness present in the PDEs and to obtain the adaptive grid. Central FD matrices are utilized to approximate the differential operators. For time integration, CN scheme is employed. The proposed method is applied on three test problems- heat equation with uncertain ICs, Burger's equation with uncertain ICs and Burger's equation with both uncertain ICs and uncertain viscosity. We have highlighted the grid modifications for every coefficient in each problem. The efficiency of the proposed method is checked for the third test problem by taking CPU time comparisons and the results are quite satisfactory. We have also plotted mean and S.D. for each test problem. It has been observed that S.D. is high near the discontinuities.

Chapter 6

An adaptive wavelet collocation B-spline generalized for Stochastic Partial Differential Equations

Wavelets consists of wonderful properties like fast wavelet transform, wavelet decomposition, localization, vanishing moments, and so on. These properties play a vital role while solving a PDE numerically. Collocation method involves numerical operators which acts on the collocation points in the physical space. However, in wavelet collocation method [261], we choose a wavelet and some kind of computationally adapted grid structure. In 1995, Vasilyev et al. [262] used collocation method on the basis of Daubechies autocorrelation scaling function [263] and the Mexican hat wavelet for the application of BCs. Further, Garba [264] used collocation scheme by involving Daubechies scaling functions.

In this chapter, we will make use of wavelet collocation method given by Bertoluzza and Naldi [265]. This method exercises Deslauriers-Dubuc interpolating basis in hierarchical form by utilizing autocorrelation function instead of Daubechies orthonormal scaling functions. A new multiresolution analysis is generated in which an interpolation operator is easily defined. In this method, we seek for an approximate solution by using the collocation technique. This collocation algorithm can be viewed as a very effective meshless technique. However, this technique is said to eliminate difficulties like the treatment of BCs and non-linear terms. The Dirichlet BCs are simply imposed by setting $u_j(0) = a$ and $u_j(1) = b$. On the other side, regarding the nonlinear operators, we remark that we are already in the physical space and so no extra computation is required for the passage between wavelet coefficients and physical space.

Wavelet methods are generally known for their self adaptive nature which makes it a good choice for the numerical solution of a PDE. The self adaptivity property comes from the good localization properties of wavelets which are seen both in space and frequency. We recall that the condition number controls the rate of convergence of a number of iterative algorithms for solving linear systems. However, it has been shown in [266] that by projecting the linear systems into the wavelet domain and rescaling it by a diagonal preconditioned matrix, we are guaranteed to obtain a small condition number, hence the solution is obtained in just a few iterations. Therefore, we use a diagonal preconditioning technique which makes the condition number of the preconditioned matrix, uniformly bounded. Moreover, to speed up the numerical scheme, we make use of

wavelet based fast algorithms for solving the stochastic PDEs. Such algorithms mainly consists of fast wavelet transform, for instance, algorithms for fast matrix-vector and fast matrix-matrix multiplication.

This chapter consists of four sections. In Section 6.1, wavelet collocation method has been discussed in detail. Moreover, it discusses the connection coefficients which are used to approximate the differential operators, fast wavelet transforms and preconditioning of matrices. Section 6.2 describes the numerical results of stochastic elliptic equation, stochastic advection diffusion and stochastic Burgers equation with uncertain initial data. Section 6.3 explains the standard adaptation technique for performing the adaptivity along with the results from the three stochastic PDEs. Section 6.4 concludes the current work.

6.1 Wavelet Collocation Approach

For wavelet collocation approach, first of all, we need a new MRA on the line by introducing autocorrelation function θ of the scaling function ϕ

$$\theta(x) = \phi(\cdot) * \phi(-\cdot)x = \int_{-\infty}^{\infty} \phi(y)\phi(y-x)dy \tag{6.1.1}$$

It consists of a wonderful "interpolation property" at integers which makes it a good choice for the design of collocation method i.e.,

$$\begin{aligned} &\theta(0) = 1 \\ &\theta(n) = 0, \quad n \neq 0 \end{aligned} \tag{6.1.2}$$

Next, we define the space $\tilde{V_j}$

$$\tilde{V_j} = span < \theta_{jk}, k \in \mathbb{Z} >$$

where θ_{jk} is defined as

$$\theta_{jk} = 2^{j/2}\theta(2^j x - k) \tag{6.1.3}$$

Let us suppose, the spatial grid is defined as

$$x_{jk} = \frac{k}{2^j}, \qquad k = 0, 1, \ldots, 2^j$$

Each function θ_{jk} is defined corresponding to point x_{jk}. Further, we define a uniform grid on (0,1) by taking x_{jk}, $k = 0, 1, \ldots, 2^j$. Corresponding to such points, we take functions θ_{jk} and

look for a solution of the form

$$u = \sum_{k=0}^{N+1} u_k \theta_{jk} \quad \text{where} \quad N = 2^j - 1 \tag{6.1.4}$$

Inserting grid points into it, we get

$$\begin{aligned}
u(x_{j0}) &= \sum_{k=0}^{N+1} u_k \theta_{jk}(x_{j0}) \\
u(0) &= u_0\theta_{j0}(x_{j0}) + u_1\theta_{j1}(x_{j0}) + u_2\theta_{j2}(x_{j0}) + \ldots + u_{N+1}\theta_{jN+1}(x_{j0}) \\
a &= u_0 2^{j/2} + u_1 2^{j/2}\theta(2^j x_{j0} - 1) + u_2 2^{j/2}\theta(2^j x_{j0} - 2) + \ldots (\because \text{Eq.(6.1.3)}) \\
&= u_0 2^{j/2} + u_1 2^{j/2}\theta(-1) + u_2 2^{j/2}\theta(-2) + \ldots \\
&= u_0 2^{j/2} + 0 + 0 + 0 \\
a &= 2^{j/2} u_0
\end{aligned} \tag{6.1.5}$$

which implies that $u_0 = 2^{-j/2}a$ and hence verifying the left BC. Similarly, we have

$$\begin{aligned}
u(x_{jN+1}) = \sum_{k=0}^{N+1} u_k \theta_{jk}(x_{jN+1}) & \\
u(1) =& u_0\theta_{j0}(x_{jN+1}) + u_1\theta_{j1}(x_{jN+1}) + u_2\theta_{j2}(x_{jN+1}) + \ldots + u_{N+1}\theta_{jN+1}(x_{jN+1}) \\
b =& u_0 2^{j/2}\theta(2^j x_{jN+1} - 0) + u_1 2^{j/2}\theta(2^j x_{jN+1} - 1) + \ldots + u_{N+1}2^{j/2}\theta(2^j x_{jN+1} - N - 1) \\
=& u_0 2^{j/2}\theta(2^j) + u_1 2^{j/2}\theta(2^j - 1) + \ldots + u_{N+1}2^{j/2}\theta(2^j - N - 1) \\
=& 0 + 0 + 0 + \ldots + u_{N+1}2^{j/2} \\
b =& 2^{j/2} u_{N+1}
\end{aligned} \tag{6.1.6}$$

which implies that $u_{N+1} = 2^{-j/2}b$ and hence verifying the right BC. For verifying the inner grid points, we have,

$$\begin{aligned}
u(x_{j1}) = \sum_{k=1}^{N+1} u_k \theta_{jk}(x_{j1}) & \\
u(1/2^j) =& u_1\theta_{j1}(x_{j1}) + u_2\theta_{j2}(x_{j1}) + \ldots + u_{N+1}\theta_{jN+1}(x_{j1}) \\
\\
\alpha_{j1} =& u_1 2^{j/2}\theta\left(2^j \times \frac{1}{2^j} - 1\right) + u_2 2^{j/2}\theta\left(2^j \times \frac{1}{2^j} - 2\right) + \ldots \\
=& u_1 2^{j/2}\theta(0) + u_2 2^{j/2}\theta(-1) + \ldots \\
=& u_1 2^{j/2} + 0 + 0 + 0 + \ldots \\
\alpha_{j1} =& 2^{j/2} u_1
\end{aligned} \tag{6.1.7}$$

i.e., $u_1 = 2^{-j/2}\alpha_{j1}$. Therefore, in similar way, we have $u_k = 2^{-j/2}\alpha_{jk}, \quad k = 1, 2, \ldots, N$.
Now, we modify the functions θ_{j0} and θ_{jN+1} in order to have the constants interpolated exactly. Let us introduce

$$\begin{aligned} \tilde{\theta}_{j0} &= \sum_{k\leq 0} \theta_{jk} \\ \tilde{\theta}_{jN+1} &= \sum_{k\geq N+1} \theta_{jk} \end{aligned}$$

Result 6.1.1 *We now look forward to some interesting properties of these functions*

$$\tilde{\theta}_{j0}(0) = 2^{j/2}, \quad \tilde{\theta}_{j0}(x_{jk}) = 0, \quad k > 0 \tag{6.1.8}$$

$$\tilde{\theta}_{jN+1}(1) = 2^{j/2}, \quad \tilde{\theta}_{jN+1}(x_{jk}) = 0, \quad k < N+1 \tag{6.1.9}$$

Proof - Eq. (6.1.8) is proved as below

$$\begin{aligned} &\tilde{\theta}_{j0}(x_{j0}) = \theta_{j0}(x_{j0}) + \theta_{j-1}(x_{j0}) + \theta_{j-2}(x_{j0}) + \ldots \\ &= 2^{j/2}\theta(2^j x_{j0} - 0) + 2^{j/2}\theta(2^j x_{j0} + 1) + 2^{j/2}\theta(2^j x_{j0} + 2) + \ldots \\ &= 2^{j/2}\theta(0) + 2^{j/2}\theta(1) + 2^{j/2}\theta(2) + \ldots \\ &= 2^{j/2}\theta(0) + 2^{j/2} \times 0 + 2^{j/2} \times 0 + \ldots \\ &= 2^{j/2} \end{aligned}$$

$$\begin{aligned} &\tilde{\theta}_{j0}(x_{j1}) = \theta_{j0}(x_{j1}) + \theta_{j-1}(x_{j1}) + \theta_{j-2}(x_{j1}) + \ldots \\ &= 2^{j/2}\theta(2^j \times \frac{1}{2^j} - 0) + 2^{j/2}\theta(2^j \times \frac{1}{2^j} + 1) + 2^{j/2}\theta(2^j \times \frac{1}{2^j} + 2) + \ldots \\ &= 2^{j/2}\theta(1) + 2^{j/2}\theta(2) + 2^{j/2}\theta(3) + \ldots \\ &= 2^{j/2} \times 0 + 2^{j/2} \times 0 + 2^{j/2} \times 0 + \ldots \\ &= 0 \end{aligned}$$

We can also prove Eq. (6.1.9) in the similar way.

According to wavelet collocation approach, we have

$$\begin{aligned} u =& u_0\tilde{\theta}_{j0} + \sum_{k=1}^{N} u_k\theta_{jk} + u_{N+1}\tilde{\theta}_{jN+1} \quad \text{where} \quad N = 2^j - 1 \\ =& 2^{-j/2}a\tilde{\theta}_{j0} + \sum_{k=1}^{N} u_k\theta_{jk} + 2^{-j/2}b\tilde{\theta}_{jN+1} \\ =& 2^{-j/2}a\tilde{\theta}_{j0} + 2^{-j/2}\sum_{k=1}^{N} \alpha_{jk}\theta_{jk} + 2^{-j/2}b\tilde{\theta}_{jN+1} \quad (\because \text{Eq.(6.1.5), (6.1.6), (6.1.7)}) \end{aligned}$$

Therefore, we have u_j with the form

$$u_j = 2^{-j/2}\left(a\tilde{\theta}_{j0} + \sum_{k=1}^{N} \alpha_{jk}\theta_{jk} + b\tilde{\theta}_{jN+1}\right) \tag{6.1.10}$$

Since we have BCs $u(0) = 0$ and $u(1) = 0$. From Eq. (6.1.10), the left BC is:

$$\begin{aligned}
u_j(0) &= 2^{-j/2}\left(a\tilde{\theta}_{j0}(0) + \sum_{k=1}^{N} \alpha_{jk}\theta_{jk}(0) + b\tilde{\theta}_{jN+1}(0)\right) \\
&= 2^{-j/2}\left(a2^{j/2} + \sum_{k=1}^{N} \alpha_{jk}2^{j/2}\theta(2^j \times 0 - k) + b \times 0\right) \quad [\because \tilde{\theta}_{j0}(0) = 2^{j/2} \; and \;\; \tilde{\theta}_{jN+1}(0) = 0] \\
&= a + \sum_{k=1}^{N} \alpha_{jk}\theta(-k) = a + \sum_{k=1}^{N} \alpha_{jk} \times 0 + 0 \;\; (\because \;\; \theta(n) = 0 \;\; n \neq 0) = a
\end{aligned}$$

Similarly, we have the right BC as follows:

$$\begin{aligned}
u_j(1) &= 2^{-j/2}\left(a\tilde{\theta}_{j0}(1) + \sum_{k=1}^{N} \alpha_{jk}\theta_{jk}(1) + b\tilde{\theta}_{jN+1}(1)\right) \\
&= 2^{-j/2}\left(0 + \sum_{k=1}^{N} \alpha_{jk}2^{j/2}\theta(2^j \times 1 - k) + b \times 2^{j/2}\right) \quad [\because \tilde{\theta}_{jN+1}(1) = 2^{j/2} \; and \;\; \tilde{\theta}_{j0}(1) = 0] \\
&= 0 + \sum_{k=1}^{N} \alpha_{jk}\theta(2^j - k) + b \\
&= \sum_{k=1}^{N} \alpha_{jk} \times 0 + b \;\; (\because \;\; \theta(n) = 0 \;\; n \neq 0) \\
&= b
\end{aligned} \tag{6.1.11}$$

and solution at the inner grid points is given as

$$\begin{aligned}
u_j(x_{jk}) &= 2^{-j/2}\left(a\tilde{\theta}_{j0}(x_{jk}) + \sum_{k=1}^{N} \alpha_{jk}\theta_{jk}(x_{jk}) + b\tilde{\theta}_{jN+1}(x_{jk})\right) \quad k = 1, 2, \ldots, 2^j - 1 \\
&= 2^{-j/2}\left(0 + \sum_{k=1}^{N} \alpha_{jk}2^{j/2}\theta(2^j x_{jk} - k) + b \times 0\right) \quad [\because \tilde{\theta}_{jN+1}(1) = 2^{j/2} \; and \;\; \tilde{\theta}_{j0}(x_{jk}) = 0, k > 0] \\
u_j(x_{jk}) &= \sum_{k=1}^{N} \alpha_{jk}\theta(2^j x_{jk} - k) \quad k = 1, 2, \ldots, 2^j - 1
\end{aligned} \tag{6.1.12}$$

6.1.1 Connection coefficients

While working with projection methods, one should understand the concept of two-term connection coefficients. Connection coefficients play a significant role in the numerical solution of PDEs.

Let us define the connection coefficients [267–269] as

$$\Gamma_{j,l,m}^{d_1,d_2} = \int_{-\infty}^{\infty} \phi_{j,l}^{(d_1)}(x)\phi_{j,m}^{(d_2)}(x)dx, \quad j,l,m \in Z \tag{6.1.13}$$

where d_1 and d_2 are orders of differentiation. Using change of variable $x \leftarrow (2^j x - l)$, we get

$$\Gamma_{j,l,m}^{d_1,d_2} = 2^{jd}\int_{-\infty}^{\infty} \phi^{(d_1)}(x)\phi^{(d_2)}(x-m+l)dx = 2^{jd}\Gamma_{0,0,m-l}^{d_1,d_2} \tag{6.1.14}$$

where $d = d_1 + d_2$. After repeated integration by parts, we get

$$\Gamma_{0,0,n}^{d_1,d_2} = (-1)^{d_1}\Gamma_{0,0,n}^{0,d} \tag{6.1.15}$$

Hence,

$$\Gamma_{j,l,m}^{d_1,d_2} = (-1)^{d_1}2^{jd}\Gamma_{0,0,m-l}^{0,d} \tag{6.1.16}$$

Hence, it is sufficient to take into consideration only one order of differentiation (d) and shift parameter ($m-l$). Therefore, we define

$$\Gamma_l^d = \int_{-\infty}^{\infty} \phi(x)\phi_l^{(d)}(x)dx \quad l \in \mathbb{Z} \tag{6.1.17}$$

Consequently

$$\Gamma_{j,l,m}^{d_1,d_2} = (-1)^{d_1}2^{jd}\Gamma_{m-l}^{d} \tag{6.1.18}$$

Also, we should note that

$$\Gamma_n^d = (-1)^d\Gamma_{-n}^d, \quad n \in [2-D, D-2] \tag{6.1.19}$$

which is obtained by using change of variable $x \leftarrow x-l$ followed by repeated integration by parts. Let D be the wavelet genus. Therefore, we will have $2D-3$ non-zero connection coefficients as the support of ϕ and $\phi_l^{(d)}$ overlap only for $-(D-2) \le l \le D-2$, i.e.,

$$\Gamma^d = \{\Gamma_l^d\}_{l=2-D}^{D-2} \tag{6.1.20}$$

Substituting the dilation equation, we get

$$\begin{aligned}\Gamma_l^d &= \int_{-\infty}^{\infty}\left[\sqrt{2}\sum_{r=0}^{D-1} a_r\phi_r(2x)\right]\left[2^d\sqrt{2}\sum_{s=0}^{D-1} a_s\phi_{2l+s}^{(d)}(2x)\right]dx \\ &= 2^{d+1}\sum_{r=0}^{D-1}\sum_{s=0}^{D-1} a_r a_s\int_{-\infty}^{\infty}\phi_r(2x)\phi_{2l+s}^{(d)}(2x)dx \quad x \leftarrow 2x\end{aligned}$$

$$= 2^d \sum_{r=0}^{D-1} \sum_{s=0}^{D-1} a_r a_s \int_{-\infty}^{\infty} \phi_r(x) \phi_{2l+s}^{(d)}(x) dx \quad x \leftarrow x - r$$

$$= 2^d \sum_{r=0}^{D-1} \sum_{s=0}^{D-1} a_r a_s \int_{-\infty}^{\infty} \phi(x) \phi_{2l+s-r}^{(d)}(x) dx$$

or
$$\sum_{r=0}^{D-1} \sum_{s=0}^{D-1} a_r a_s \Gamma_{2l+s-r}^d = \frac{1}{2^d} \Gamma_l^d \quad l \in [2-D, D-2]$$

It has been observed that it requires an additional inhomogeneous equation in order to find a unique solution and this additional equation is derived from the moment conditions of the scaling function and essentially imposes a normalization on Γ. If we consider wavelets with $\mathcal{M}$ vanishing moments, then

$$x^d = \sum_{l=-\infty}^{\infty} M_l^d \phi(x-l) \tag{6.1.21}$$

Now, differentiating both sides d times, we get

$$d! = \sum_{l=-\infty}^{\infty} M_l^d \phi^d(x-l) \tag{6.1.22}$$

Multiplying by $\phi(x)$ and integrating we then obtain

$$d! \int_{-\infty}^{\infty} \phi(x) dx = \sum_{l=2-D}^{D-2} M_d^l \int_{-\infty}^{\infty} \phi(x) \phi^{(d)}(x-l) dx \tag{6.1.23}$$

Hence, we get

$$\sum_{l=2-D}^{D-2} M_d^l \Gamma_l^d = d! \tag{6.1.24}$$

Connection coefficients for first and second derivative with $D = 6$ are given by

$$\Gamma^1 = \begin{pmatrix} \Gamma_{-4}^1 \\ \Gamma_{-3}^1 \\ \Gamma_{-2}^1 \\ \Gamma_{-1}^1 \\ \Gamma_0^1 \\ \Gamma_1^1 \\ \Gamma_2^1 \\ \Gamma_3^1 \\ \Gamma_4^1 \end{pmatrix} = \begin{pmatrix} 0.0003 \\ 0.0146 \\ -0.1452 \\ 0.7452 \\ -0.0000 \\ -0.7452 \\ 0.1452 \\ -0.0146 \\ -0.0003 \end{pmatrix} \Gamma^2 = \begin{pmatrix} \Gamma_{-4}^2 \\ \Gamma_{-3}^2 \\ \Gamma_{-2}^2 \\ \Gamma_{-1}^2 \\ \Gamma_0^2 \\ \Gamma_1^2 \\ \Gamma_2^2 \\ \Gamma_3^2 \\ \Gamma_4^2 \end{pmatrix} \begin{pmatrix} 0.0054 \\ 0.1143 \\ -0.8762 \\ 3.3905 \\ -5.2679 \\ 3.3905 \\ -0.8762 \\ 0.1143 \\ 0.0054 \end{pmatrix} \tag{6.1.25}$$

6.1.2 Fast wavelet transform

Beylkin et al. [266, 270, 271] has pioneered significant progress in the direction of matrix compression and fast processing of certain densely populated matrices. The primary cause for the restricted use of integral equations as a numerical tool in large-scale calculations is that they usually head to heavy systems of linear algebraic equations, and the latter have to be solved, either directly or iteratively. Therefore, in almost all areas of computational mathematics, dense matrices are just avoided whenever possible. For instance, FD and finite element methods can be observed as devices for minimizing a PDE to a sparse linear system. Therefore, the cost of sparsity in such cases is the naturally high condition number of the resulting matrices. Hence, Beylkin et al. designed a class of fast numerical algorithms for instant application of dense matrices to vectors.

Interpolating wavelet transform [263] is a non-orthogonal transform with similarity to orthogonal wavelet transforms, such that it describes the object on the basis of translations and dilations of a basic function but for which the coefficients are acquired from linear combinations of samples instead from integrals. This transform fundamentally relies on the interpolation scheme of Deslauriers-Dubuc. Moreover, the interpolating transform is optimal from the viewpoint of calculating individual coefficients in parallel.

On the other hand, Beylkin et al. developed two schemes- standard and non-standard form of matrices. The standard form is a simple $Nlog(N)$ order procedure with a direct realization of the matrix of the operator in the wavelet basis. While this is a effective numerical tool in itself, its scope of applicability is greatly extended by the N order non-standard scheme. In this chapter, we will obtain the standard form of the differential operators by the rowwise and columnwise application of the fast interpolating wavelet transform. For this, we can use the wavelet toolbox "WAVELAB850"[1]. From this toolbox, we will make use of "FWT_PO.m" to obtain the standard form of operators.

$$\mathcal{L}u = f$$
$$FWT\_PO(\mathcal{L})u = FWT\_PO(f)$$

Also, from the same toolbox, we will use "IWT_PO" for calculating the inverse wavelet transform.

6.1.3 Preconditioning

Preconditioning [272] is a technique in mathematics which makes the condition number of the problem more suitable for solving the PDE numerically. A condition number is the ratio between

[1]WAVELAB850 (`http://statweb.stanford.edu/wavelab/Wavelab_850/download.html`)

the largest singular value to the smallest singular value of a matrix. Preconditioning basically aims at reducing a condition number of the problem. In case of periodized derivative operators, the bound on the condition number relies on the specific choice of the wavelet basis. In this chapter, we will apply preconditioning to the standard form of derivative operators. After applying the diagonal pre-conditioner, the condition number of the operator becomes uniformly bounded in correspondence to the size of the matrix.

Let $n = 2^m$ and second derivative $\mathfrak{D}^{(2)}$ is of size $n \times n$, then, the derivative operator is preconditioned in the wavelet basis by the diagonal matrix $\mathscr{P}$ as follows:

$$\mathfrak{D}^{(2),p} = \mathscr{P}\mathfrak{D}^{(2),w}\mathscr{P} \tag{6.1.26}$$

where $\mathscr{P}_{il} = \delta_{il}2^j,\ 1 \leq j \leq n$ and j is chosen depending on i, l so that $n - \frac{n}{2^{j-1}} + 1 \leq i, l \leq n - \frac{n}{2^j}$ and $\mathscr{P}_{nn} = 2^n$. For example, for $j = 3$, we have

$$\mathscr{P} = \begin{pmatrix} 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 8 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 8 \end{pmatrix}$$

6.1.4 Algorithm for linear B-spline wavelet collocation method

Algorithm 6.1 Wavelet collocation based linear B-spline gPC

1: Linear B-spline wavelets are employed to approximate the unknown u in terms of unknown deterministic coefficients and stochastic wavelet basis.
2: Galerkin projection is applied to calculate the unknown deterministic coefficients.
3: After applying B-spline chaos and Galerkin projection, we get P number of deterministic equations.
4: Further, the connection coefficients are computed using Daubechies wavelets for approximating the differential operators in space dimension.
5: A class of fast algorithms like fast wavelet transform are used to speed up the numerical scheme.
6: Then, diagonal preconditioning is performed so that the condition number of the matrices becomes bounded.
7: After that, inverse fast wavelet transform is applied in order to get the solution.
8: Finally, we estimate the mean and variance.

6.2 Numerical Results

In this section, we will apply the proposed method on three test problems. We have considered stochastic elliptic equation with uncertain f, stochastic advection diffusion equation as well as stochastic Burger's equation with uncertain initial data.

6.2.1 Stochastic Elliptic equation

Let us consider stochastic elliptic equation with Dirichlet BCs and random f

$$\begin{aligned} -\epsilon\frac{\partial^2 u}{\partial x^2}+\frac{\partial u}{\partial x} &= f \quad \text{in} \quad (0,1) \\ u(0)=0, \quad u(1) &= 0 \end{aligned} \tag{6.2.1}$$

with right hand side is $f = 1 + 0.1\xi$ where ξ is uniformly distributed over $[-1, 1]$. Applying PCE on Eq.(6.2.1), we get

$$-\epsilon\frac{\partial^2}{\partial x^2}\Big(\sum_{i=1}^{P} u_i(t)W_i(\xi)\Big)+\frac{\partial}{\partial x}\Big(\sum_{i=1}^{P} u_i(t)W_i(\xi)\Big) = \sum_{i=1}^{P} f_i W_i(\xi), \tag{6.2.2}$$

Taking inner product w.r.t to dual $\tilde{W}_k(\xi)$, we get

$$\begin{aligned} -\epsilon\frac{\partial^2}{\partial x^2}\Big(\sum_{i=1}^{P} u_i(t)\langle W_i(\xi)\tilde{W}_k(\xi)\rangle\Big)+\frac{\partial}{\partial x}\Big(\sum_{i=1}^{P} u_i(t)\langle W_i(\xi)\tilde{W}_k(\xi)\rangle\Big) &= \sum_{i=1}^{P} f_i\langle W_i(\xi)\tilde{W}_k(\xi)\rangle, \\ -\epsilon\frac{\partial^2}{\partial x^2}u_k(t)+\frac{\partial}{\partial x}u_k(t) &= f_k, \quad k=1,2,\ldots,P \end{aligned}$$

After obtaining P number of deterministic equations, we approximate differential operators using connection coefficients as follows:

$$\begin{aligned} \frac{\partial^2}{\partial x^2} &= \mathfrak{D}^{(2)} = \Gamma_l^2 \\ \frac{\partial}{\partial x} &= \mathfrak{D}^{(1)} = \Gamma_l^1 \end{aligned} \tag{6.2.3}$$

Therefore, we have

$$\begin{aligned} \mathcal{A}_k u_k &= f_k, \quad k=1,2,\ldots,P \\ u_k &= \mathcal{A}_k^{-1} f_k \end{aligned} \tag{6.2.4}$$

where

$$\mathcal{A}_k = -\epsilon u_k'' + u_k' \tag{6.2.5}$$

Therefore, we will apply the wavelet collocation method i.e., Eq. (6.1.10), we get

$$u_j(0) = 0,$$

$$\begin{aligned} -\epsilon u_j''(x_{jk}) + u_j'(x_{jk}) &= -\epsilon \sum_{k=1}^{N} \alpha_{jk}\theta_{jk}''(x_{jk}) + \sum_{k=1}^{N} \alpha_{jk}\theta_{jk}'(x_{jk}) \\ u_j(1) &= 0 \end{aligned} \tag{6.2.6}$$

Now to compute the matrix entries, we use connection coefficients

$$\begin{aligned} -\theta_{jn}''(x_{jk}) = \int_{\mathbb{R}} \phi_{jk}'\phi_{jn}' = \int_{\mathbb{R}} \phi(x)\phi_{n-k}''(x)dx = \Gamma_{n-k}'' = \Gamma_{n-k}^2 \\ -\theta_{jn}'(x_{jk}) == \int_{\mathbb{R}} \phi(x)\phi_{n-k}'(x)dx = \Gamma_{n-k}' = \Gamma_{n-k}^1 \end{aligned} \tag{6.2.7}$$

Taking $N = 2^j$, Eq.(6.2.6) results into following system

$$-2^{2j}\epsilon \begin{pmatrix} \theta_{j1}''(x_{j1}) & \theta_{j2}''(x_{j1}) & \theta_{j3}''(x_{j1}) & \theta_{j4}''(x_{j1}) & \theta_{j5}''(x_{j1}) & \theta_{j6}''(x_{j1}) & \theta_{j7}''(x_{j1}) & \theta_{j8}''(x_{j1}) \\ \theta_{j1}''(x_{j2}) & \theta_{j2}''(x_{j2}) & \theta_{j3}''(x_{j2}) & \theta_{j4}''(x_{j2}) & \theta_{j5}''(x_{j2}) & \theta_{j6}''(x_{j2}) & \theta_{j7}''(x_{j2}) & \theta_{j8}''(x_{j2}) \\ \theta_{j1}''(x_{j3}) & \theta_{j2}''(x_{j3}) & \theta_{j3}''(x_{j3}) & \theta_{j4}''(x_{j3}) & \theta_{j5}''(x_{j3}) & \theta_{j6}''(x_{j3}) & \theta_{j7}''(x_{j3}) & \theta_{j8}''(x_{j3}) \\ \theta_{j1}''(x_{j4}) & \theta_{j2}''(x_{j4}) & \theta_{j3}''(x_{j4}) & \theta_{j4}''(x_{j4}) & \theta_{j5}''(x_{j4}) & \theta_{j6}''(x_{j4}) & \theta_{j7}''(x_{j4}) & \theta_{j8}''(x_{j4}) \\ \theta_{j1}''(x_{j5}) & \theta_{j2}''(x_{j5}) & \theta_{j3}''(x_{j5}) & \theta_{j4}''(x_{j5}) & \theta_{j5}''(x_{j5}) & \theta_{j6}''(x_{j5}) & \theta_{j7}''(x_{j5}) & \theta_{j8}''(x_{j5}) \\ \theta_{j1}''(x_{j6}) & \theta_{j2}''(x_{j6}) & \theta_{j3}''(x_{j6}) & \theta_{j4}''(x_{j6}) & \theta_{j5}''(x_{j6}) & \theta_{j6}''(x_{j6}) & \theta_{j7}''(x_{j6}) & \theta_{j8}''(x_{j6}) \\ \theta_{j1}''(x_{j7}) & \theta_{j2}''(x_{j7}) & \theta_{j3}''(x_{j7}) & \theta_{j4}''(x_{j7}) & \theta_{j5}''(x_{j7}) & \theta_{j6}''(x_{j7}) & \theta_{j7}''(x_{j7}) & \theta_{j8}''(x_{j7}) \\ \theta_{j1}''(x_{j8}) & \theta_{j2}''(x_{j8}) & \theta_{j3}''(x_{j8}) & \theta_{j4}''(x_{j8}) & \theta_{j5}''(x_{j8}) & \theta_{j6}''(x_{j8}) & \theta_{j7}''(x_{j8}) & \theta_{j8}''(x_{j8}) \end{pmatrix} \begin{pmatrix} \alpha_{j1} \\ \alpha_{j2} \\ \alpha_{j3} \\ \alpha_{j4} \\ \alpha_{j5} \\ \alpha_{j6} \\ \alpha_{j7} \\ \alpha_{j8} \end{pmatrix} +$$

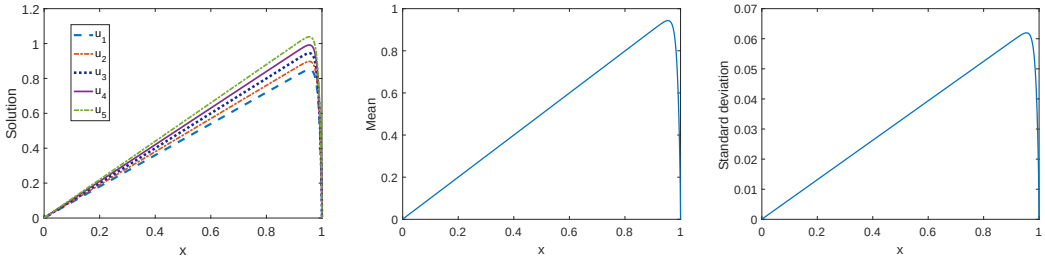
$$2^j \begin{pmatrix} \theta_{j1}'(x_{j1}) & \theta_{j2}'(x_{j1}) & \theta_{j3}'(x_{j1}) & \theta_{j4}'(x_{j1}) & \theta_{j5}'(x_{j1}) & \theta_{j6}'(x_{j1}) & \theta_{j7}'(x_{j1}) & \theta_{j8}'(x_{j1}) \\ \theta_{j1}'(x_{j2}) & \theta_{j2}'(x_{j2}) & \theta_{j3}'(x_{j2}) & \theta_{j4}'(x_{j2}) & \theta_{j5}'(x_{j2}) & \theta_{j6}'(x_{j2}) & \theta_{j7}'(x_{j2}) & \theta_{j8}'(x_{j2}) \\ \theta_{j1}'(x_{j3}) & \theta_{j2}'(x_{j3}) & \theta_{j3}'(x_{j3}) & \theta_{j4}'(x_{j3}) & \theta_{j5}'(x_{j3}) & \theta_{j6}'(x_{j3}) & \theta_{j7}'(x_{j3}) & \theta_{j8}'(x_{j3}) \\ \theta_{j1}'(x_{j4}) & \theta_{j2}'(x_{j4}) & \theta_{j3}'(x_{j4}) & \theta_{j4}'(x_{j4}) & \theta_{j5}'(x_{j4}) & \theta_{j6}'(x_{j4}) & \theta_{j7}'(x_{j4}) & \theta_{j8}'(x_{j4}) \\ \theta_{j1}'(x_{j5}) & \theta_{j2}'(x_{j5}) & \theta_{j3}'(x_{j5}) & \theta_{j4}'(x_{j5}) & \theta_{j5}'(x_{j5}) & \theta_{j6}'(x_{j5}) & \theta_{j7}'(x_{j5}) & \theta_{j8}'(x_{j5}) \\ \theta_{j1}'(x_{j6}) & \theta_{j2}'(x_{j6}) & \theta_{j3}'(x_{j6}) & \theta_{j4}'(x_{j6}) & \theta_{j5}'(x_{j6}) & \theta_{j6}'(x_{j6}) & \theta_{j7}'(x_{j6}) & \theta_{j8}'(x_{j6}) \\ \theta_{j1}'(x_{j7}) & \theta_{j2}'(x_{j7}) & \theta_{j3}'(x_{j7}) & \theta_{j4}'(x_{j7}) & \theta_{j5}'(x_{j7}) & \theta_{j6}'(x_{j7}) & \theta_{j7}'(x_{j7}) & \theta_{j8}'(x_{j7}) \\ \theta_{j1}'(x_{j8}) & \theta_{j2}'(x_{j8}) & \theta_{j3}'(x_{j8}) & \theta_{j4}'(x_{j8}) & \theta_{j5}'(x_{j8}) & \theta_{j6}'(x_{j8}) & \theta_{j7}'(x_{j8}) & \theta_{j8}'(x_{j8}) \end{pmatrix} \begin{pmatrix} \alpha_{j1} \\ \alpha_{j2} \\ \alpha_{j3} \\ \alpha_{j4} \\ \alpha_{j5} \\ \alpha_{j6} \\ \alpha_{j7} \\ \alpha_{j8} \end{pmatrix} = \begin{pmatrix} f(x_{j1}) \\ f(x_{j2}) \\ f(x_{j3}) \\ f(x_{j4}) \\ f(x_{j5}) \\ f(x_{j6}) \\ f(x_{j7}) \\ f(x_{j8}) \end{pmatrix}$$

Therefore, we get

$$-2^{2j_\epsilon} \begin{pmatrix} \Gamma_0^2 & \Gamma_1^2 & \Gamma_2^2 & \Gamma_3^2 & \Gamma_4^2 & \Gamma_5^2 & \Gamma_6^2 & \Gamma_7^2 \\ \Gamma_{-1}^2 & \Gamma_0^2 & \Gamma_1^2 & \Gamma_2^2 & \Gamma_3^2 & \Gamma_4^2 & \Gamma_5^2 & \Gamma_6^2 \\ \Gamma_{-2}^2 & \Gamma_{-1}^2 & \Gamma_0^2 & \Gamma_1^2 & \Gamma_2^2 & \Gamma_3^2 & \Gamma_4^2 & \Gamma_5^2 \\ \Gamma_{-3}^2 & \Gamma_{-2}^2 & \Gamma_{-1}^2 & \Gamma_0^2 & \Gamma_1^2 & \Gamma_2^2 & \Gamma_3^2 & \Gamma_4^2 \\ \Gamma_{-4}^2 & \Gamma_{-3}^2 & \Gamma_{-2}^2 & \Gamma_{-1}^2 & \Gamma_0^2 & \Gamma_1^2 & \Gamma_2^2 & \Gamma_3^2 \\ \Gamma_{-5}^2 & \Gamma_{-4}^2 & \Gamma_{-3}^2 & \Gamma_{-2}^2 & \Gamma_{-1}^2 & \Gamma_0^2 & \Gamma_1^2 & \Gamma_2^2 \\ \Gamma_{-6}^2 & \Gamma_{-5}^2 & \Gamma_{-4}^2 & \Gamma_{-3}^2 & \Gamma_{-2}^2 & \Gamma_{-1}^2 & \Gamma_0^2 & \Gamma_1^2 \\ \Gamma_{-7}^2 & \Gamma_{-6}^2 & \Gamma_{-5}^2 & \Gamma_{-4}^2 & \Gamma_{-3}^2 & \Gamma_{-2}^2 & \Gamma_{-1}^2 & \Gamma_0^2 \end{pmatrix} \begin{pmatrix} \alpha_{j1} \\ \alpha_{j2} \\ \alpha_{j3} \\ \alpha_{j4} \\ \alpha_{j5} \\ \alpha_{j6} \\ \alpha_{j7} \\ \alpha_{j8} \end{pmatrix} + 2^j \begin{pmatrix} \Gamma_0^1 & \Gamma_1^1 & \Gamma_2^1 & \Gamma_3^1 & \Gamma_4^1 & \Gamma_5^1 & \Gamma_6^1 & \Gamma_7^1 \\ \Gamma_{-1}^1 & \Gamma_0^1 & \Gamma_1^1 & \Gamma_2^1 & \Gamma_3^1 & \Gamma_4^1 & \Gamma_5^1 & \Gamma_6^1 \\ \Gamma_{-2}^1 & \Gamma_{-1}^1 & \Gamma_0^1 & \Gamma_1^1 & \Gamma_2^1 & \Gamma_3^1 & \Gamma_4^1 & \Gamma_5^1 \\ \Gamma_{-3}^1 & \Gamma_{-2}^1 & \Gamma_{-1}^1 & \Gamma_0^1 & \Gamma_1^1 & \Gamma_2^1 & \Gamma_3^1 & \Gamma_4^2 \\ \Gamma_{-4}^1 & \Gamma_{-3}^1 & \Gamma_{-2}^1 & \Gamma_{-1}^1 & \Gamma_0^1 & \Gamma_1^1 & \Gamma_2^1 & \Gamma_3^1 \\ \Gamma_{-5}^1 & \Gamma_{-4}^1 & \Gamma_{-3}^1 & \Gamma_{-2}^1 & \Gamma_{-1}^1 & \Gamma_0^1 & \Gamma_1^1 & \Gamma_2^1 \\ \Gamma_{-6}^1 & \Gamma_{-5}^1 & \Gamma_{-4}^1 & \Gamma_{-3}^1 & \Gamma_{-2}^1 & \Gamma_{-1}^1 & \Gamma_0^1 & \Gamma_1^1 \\ \Gamma_{-7}^1 & \Gamma_{-6}^1 & \Gamma_{-5}^1 & \Gamma_{-4}^1 & \Gamma_{-3}^1 & \Gamma_{-2}^1 & \Gamma_{-1}^1 & \Gamma_0^1 \end{pmatrix} \begin{pmatrix} \alpha_{j1} \\ \alpha_{j2} \\ \alpha_{j3} \\ \alpha_{j4} \\ \alpha_{j5} \\ \alpha_{j6} \\ \alpha_{j7} \\ \alpha_{j8} \end{pmatrix} = \begin{pmatrix} f(x_{j1}) \\ f(x_{j2}) \\ f(x_{j3}) \\ f(x_{j4}) \\ f(x_{j5}) \\ f(x_{j6}) \\ f(x_{j7}) \\ f(x_{j8}) \end{pmatrix}$$

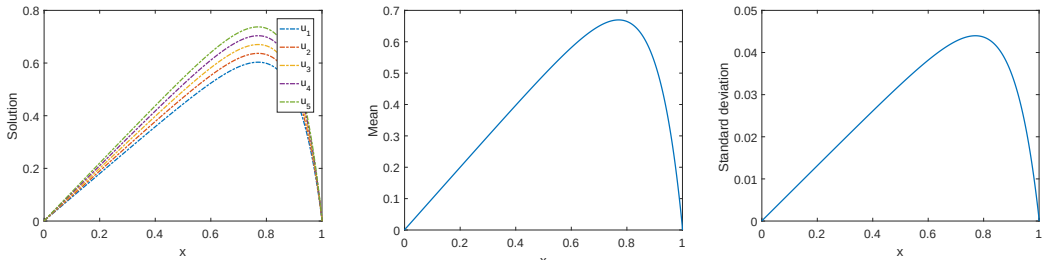
$$-2^{2j}\epsilon \begin{pmatrix} \Gamma_0^2 & \Gamma_1^2 & \Gamma_2^2 & \Gamma_3^2 & \Gamma_4^2 & 0 & 0 & 0 \\ \Gamma_{-1}^2 & \Gamma_0^2 & \Gamma_1^2 & \Gamma_2^2 & \Gamma_3^2 & \Gamma_4^2 & 0 & 0 \\ \Gamma_{-2}^2 & \Gamma_{-1}^2 & \Gamma_0^2 & \Gamma_1^2 & \Gamma_2^2 & \Gamma_3^2 & \Gamma_4^2 & 0 \\ \Gamma_{-3}^2 & \Gamma_{-2}^2 & \Gamma_{-1}^2 & \Gamma_0^2 & \Gamma_1^2 & \Gamma_2^2 & \Gamma_3^2 & \Gamma_4^2 \\ \Gamma_{-4}^2 & \Gamma_{-3}^2 & \Gamma_{-2}^2 & \Gamma_{-1}^2 & \Gamma_0^2 & \Gamma_1^2 & \Gamma_2^2 & \Gamma_3^2 \\ 0 & \Gamma_{-4}^2 & \Gamma_{-3}^2 & \Gamma_{-2}^2 & \Gamma_{-1}^2 & \Gamma_0^2 & \Gamma_1^2 & \Gamma_2^2 \\ 0 & 0 & \Gamma_{-4}^2 & \Gamma_{-3}^2 & \Gamma_{-2}^2 & \Gamma_{-1}^2 & \Gamma_0^2 & \Gamma_{-1}^2 \\ 0 & 0 & 0 & \Gamma_{-4}^2 & \Gamma_{-3}^2 & \Gamma_{-2}^2 & \Gamma_{-1}^2 & \Gamma_0^2 \end{pmatrix} \begin{pmatrix} \alpha_{j1} \\ \alpha_{j2} \\ \alpha_{j3} \\ \alpha_{j4} \\ \alpha_{j5} \\ \alpha_{j6} \\ \alpha_{j7} \\ \alpha_{j8} \end{pmatrix} + 2^j \begin{pmatrix} \Gamma_0^1 & \Gamma_1^1 & \Gamma_2^1 & \Gamma_3^1 & \Gamma_4^1 & 0 & 0 & 0 \\ \Gamma_{-1}^1 & \Gamma_0^1 & \Gamma_1^1 & \Gamma_2^1 & \Gamma_3^1 & \Gamma_4^1 & 0 & 0 \\ \Gamma_{-2}^1 & \Gamma_{-1}^1 & \Gamma_0^1 & \Gamma_1^1 & \Gamma_2^1 & \Gamma_3^1 & \Gamma_4^1 & 0 \\ \Gamma_{-3}^1 & \Gamma_{-2}^1 & \Gamma_{-1}^1 & \Gamma_0^1 & \Gamma_1^1 & \Gamma_2^1 & \Gamma_3^1 & \Gamma_4^2 \\ \Gamma_{-4}^1 & \Gamma_{-3}^1 & \Gamma_{-2}^1 & \Gamma_{-1}^1 & \Gamma_0^1 & \Gamma_1^1 & \Gamma_2^1 & \Gamma_3^1 \\ 0 & \Gamma_{-4}^1 & \Gamma_{-3}^1 & \Gamma_{-2}^1 & \Gamma_{-1}^1 & \Gamma_0^1 & \Gamma_1^1 & \Gamma_2^1 \\ 0 & 0 & \Gamma_{-4}^1 & \Gamma_{-3}^1 & \Gamma_{-2}^1 & \Gamma_{-1}^1 & \Gamma_0^1 & \Gamma_1^1 \\ 0 & 0 & 0 & \Gamma_{-4}^1 & \Gamma_{-3}^1 & \Gamma_{-2}^1 & \Gamma_{-1}^1 & \Gamma_0^1 \end{pmatrix} \begin{pmatrix} \alpha_{j1} \\ \alpha_{j2} \\ \alpha_{j3} \\ \alpha_{j4} \\ \alpha_{j5} \\ \alpha_{j6} \\ \alpha_{j7} \\ \alpha_{j8} \end{pmatrix} = \begin{pmatrix} f(x_{j1}) \\ f(x_{j2}) \\ f(x_{j3}) \\ f(x_{j4}) \\ f(x_{j5}) \\ f(x_{j6}) \\ f(x_{j7}) \\ f(x_{j8}) \end{pmatrix}$$

We have taken grid points $m = 2^{10}$ with Daubechies wavelet of genus 6, i.e., $D = 6$. Moreover,



(a) All coefficients (b) Mean (c) S.D.

Figure 6.1: Test problem 1 with $P = 5$, $\epsilon = 0.01$



(a) All coefficients (b) Mean (c) S.D.

Figure 6.2: Test problem 1 with $P = 5$, $\epsilon = 0.1$

P	1	2	3	4	5	6	7	8	9
Mean	0.0551	0.1715	0.2940	0.4226	0.4900	0.4900	0.4900	0.4900	0.4900
S.D.	0.1148	0.2002	0.2053	0.1231	0.0322	0.0322	0.0322	0.0322	0.0322

Table 6.1: (Test Problem 1) Mean and S.D. $m = 2^{10}$, $D = 6$, $\epsilon = 0.01$

we have considered two values of $\epsilon = 0.1, 0.01$. From table 6.1, it has been noticed that after $P = 5$, the mean and S.D. are coming out to be same when $\epsilon = 0.01$. Also, from Fig. 6.1 and Fig. 6.2, it has been observed that as we decrease the value of ϵ, the problem is displaying a layer near $x = 1$. We analyse that the proposed method performs well and captures the uncertainty by requiring low polynomial chaos order P.

6.2.2 Stochastic Advection-Diffusion equation

One dimensional advection-diffusion equation is considered in $\mathcal{D} = [0, 2]$ and $T = (0, 0.3]$

$$\frac{\partial u}{\partial t} + v\frac{\partial u}{\partial x} - \mu\frac{\partial^2 u}{\partial x^2} = 0 \quad \text{in } \mathcal{D} \times T \times \Omega. \tag{6.2.8}$$

where μ is the diffusion coefficient and v denotes velocity field. The IC is

$$u(x, 0, \xi) = \begin{cases} 1 + 0.01\xi, & x \in [0.2, 0.7] \\ 0, & \text{otherwise} \end{cases}$$

and homogeneous Dirichlet BCs are given by

$$u(x = 0, t, \xi) = 0, \quad u(x = 2, t, \xi) = 0.$$

where ξ is uniformly distributed over $[-1, 1]$. Let us now apply the generalized WBe gPC on this problem as follows:

$$\frac{\partial}{\partial t}\Big(\sum_{i=1}^{P} u_i(x,t)W_i(\xi)\Big) + v\frac{\partial}{\partial x}\Big(\sum_{i=1}^{P} u_i(x,t)W_i(\xi)\Big) - \mu\frac{\partial^2}{\partial x^2}\Big(\sum_{i=1}^{P} u_i(x,t)W_i(\xi)\Big) = 0,$$

Applying Galerkin projection, we get

$$\begin{aligned} \frac{\partial}{\partial t}\Big(\sum_{i=1}^{P} u_i(x,t)\langle W_i(\xi)\tilde{W}_k(\xi)\rangle\Big) + v\frac{\partial}{\partial x}\Big(\sum_{i=1}^{P} u_i(x,t)\langle W_i(\xi)\tilde{W}_k(\xi)\rangle\Big) \\ - \mu\frac{\partial^2}{\partial x^2}\Big(\sum_{i=1}^{P} u_i(x,t)\langle W_i(\xi)\tilde{W}_k(\xi)\rangle\Big) = 0, \\ \frac{\partial u_k(x,t)}{\partial t} + v\frac{\partial u_k(x,t)}{\partial x} - \mu\frac{\partial^2 u_k(x,t)}{\partial x^2} = 0, \quad k = 1, 2, \ldots, P \\ \frac{\partial u_k(x,t)}{\partial t} = -v\frac{\partial u_k(x,t)}{\partial x} + \mu\frac{\partial^2 u_k}{\partial x^2}, \quad k = 1, 2, \ldots, P \end{aligned}$$

Applying CN scheme, we get

$$\begin{aligned} \frac{u_k^n - u_k^{n-1}}{\Delta t} &= -v\mathbf{L}_x\Big(\frac{u_k^n + u_k^{n-1}}{2}\Big) + \mu\mathbf{L}_{xx}\Big(\frac{u_k^n + u_k^{n-1}}{2}\Big) \\ \Big(I + \frac{v\Delta t\mathbf{L}_x}{2} - \frac{\mu\Delta t\mathbf{L}_{xx}}{2}\Big)u_k^n &= \Big(I - \frac{v\Delta t\mathbf{L}_x}{2} + \frac{\mu\Delta t\mathbf{L}_{xx}}{2}\Big)u_k^{n-1} \\ \mathcal{A}_k u_k^n &= \mathcal{B}_k u_k^{n-1} \\ u_k^n &= \mathcal{A}_k^{-1}\mathcal{B}_k u_k^{n-1}, \quad k = 1, 2, \ldots, P \end{aligned}$$

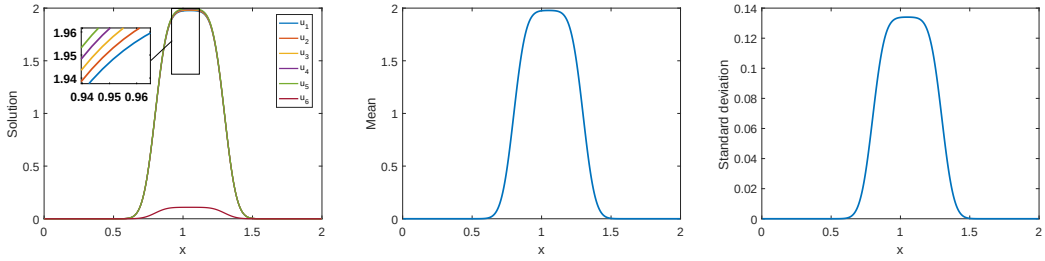
where

$$\mathcal{A}_k = I + \frac{v\Delta t\mathbf{L}_x}{2} - \frac{\mu\Delta t\mathbf{L}_{xx}}{2}$$
$$\mathcal{B}_k = I - \frac{v\Delta t\mathbf{L}_x}{2} + \frac{\mu\Delta t\mathbf{L}_{xx}}{2}$$
$$\mathbf{L}_x = \mathfrak{D}^{(1)}(u(t))$$
$$\mathbf{L}_{xx} = \mathfrak{D}^{(2)}(u(t))$$

It has been analysed from table 6.2 that mean and S.D. are same for $P = 5$ and $P > 5$. In this

P	1	2	3	4	5	6	7	8	9	10
Mean	0.0615	0.1849	0.3085	0.4325	0.4946	0.4946	0.4946	0.4946	0.4946	0.4946
S.D.	0.1250	0.2156	0.2170	0.1308	0.0311	0.0335	0.0335	0.0335	0.0335	0.0335

Table 6.2: (Test Problem 2)



(a) All coefficients (b) Mean (c) S.D.

Figure 6.3: Stochastic Advection diffusion equation with P=6

problem, we have considered $v = 1$, $\mu = 0.002$ and $t \in [0, 0.3]$. Moreover, Fig 6.3 a) presents the first 6 coefficients of the B-spline gPC. Fig 6.3 b and c) displays respectively mean and S.D. of the advection diffusion equation.

6.2.3 Stochastic Burger's equation

Consider non-linear Burger's equation

$$\frac{\partial u}{\partial t} = \nu\frac{\partial^2 u}{\partial x^2} - u\frac{\partial u}{\partial x} \tag{6.2.9}$$

with uncertain ICs

$$u(x, t = 0, \xi) = sin(2\pi x) + 0.01\xi \tag{6.2.10}$$

where ξ is uniformly distributed over $[-1,1]$ and Dirichlet BCs

$$u(x=0,t)=0, \quad u(x=1,t)=0$$

Applying B-spline gPC on Eq. (6.2.9), we get

$$\begin{aligned}
\frac{\partial}{\partial t}\Big(\sum_{i=1}^{P} u_i(x,t)W_i(\xi)\Big) - \nu\frac{\partial}{\partial x}\Big(\sum_{i=1}^{P} u_i(x,t)W_i(\xi)\Big) \\
+ \sum_{i=1}^{P} u_i(x,t)W_i(\xi)\frac{\partial^2}{\partial x^2}\Big(\sum_{j=1}^{P} u_j W_j(\xi)\Big) = 0, \\
\frac{\partial}{\partial t}\Big(\sum_{i=1}^{P} u_i\langle W_i(\xi)\tilde{W}_k(\xi)\rangle\Big) - \nu\frac{\partial}{\partial x}\Big(\sum_{i=1}^{P} u_i\langle W_i(\xi)\tilde{W}_k(\xi)\rangle\Big) \\
+ \sum_{i=1}^{P}\sum_{j=1}^{P} u_i\frac{\partial^2 u_j}{\partial x^2}\Big(\langle W_i(\xi)W_j(\xi)\tilde{W}_k(\xi)\rangle\Big) = 0, \\
\frac{\partial u_k}{\partial t} - \nu\frac{\partial u_k}{\partial x} + \sum_{i=1}^{P}\sum_{j=1}^{P} u_i\frac{\partial^2 u_j}{\partial x^2}\mathscr{C}_{ijk} = 0, \quad k=1,2,\ldots,P
\end{aligned}$$

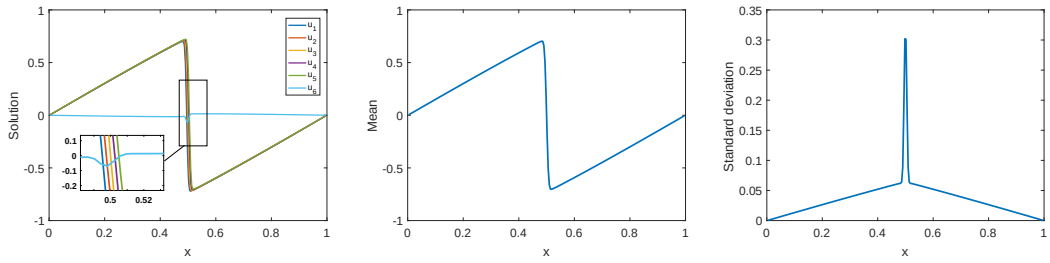
Applying CN scheme, we get

$$\begin{aligned}
\frac{u_k^n - u_k^{n-1}}{\Delta t} &= \nu\mathbf{L}_x\Big(\frac{u_k^n + u_k^{n-1}}{2}\Big) + \mathbf{N}\Big(\frac{u_k^n + u_k^{n-1}}{2}\Big) \\
\Big(I - \frac{\nu\Delta t\mathbf{L}_x}{2}\Big)u_k^n &= \Big(I + \frac{\nu\Delta t\mathbf{L}_x}{2}\Big)u_k^{n-1} + \mathbf{N}\Big(\frac{u_k^n + u_k^{n-1}}{2}\Big) \\
\mathcal{A}_k u_k^n &= \mathcal{B}_k u_k^{n-1} \\
u_k^n &= \mathcal{A}_k^{-1}\mathcal{B}_k u_k^{n-1}
\end{aligned}$$

In this test problem, we have taken $\nu = 0.002$. We have considered $P = 6$ and $t \in [0, 0.5]$. We have also taken Daubechies wavelets with $D = 6$. In fig 6.4 a), we have plotted the polynomial chaos coefficients $u_1, u_2, u_3, u_4, u_5, u_6$. In fig 6.4 b) and c), mean and S.D. are plotted. It has been noticed that more shocks are being located in the Burger's solution due to uncertainty present in initial data. Moreover, S.D. is large at $x = 0.5$. For non-linear Burger's equation, we have analysed from table 6.4 that after $P = 10$, the mean and S.D. are converging to -8.64×10^{-8} and 0.0340 respectively.

	0.1	0.01	0.001
Mean	-1.7908×10^{-7}	-1.9173×10^{-8}	-1.918×10^{-9}
S.D.	0.0829	0.0358	0.0324

Table 6.3: Stochastic Burger's equation with $P = 6$



(a) All coefficients (b) Mean (c) S.D.

Figure 6.4: Test problem 3 with P=6, $u(x, t = 0, \xi) = sin(2\pi x) + 0.01\xi$

P	Mean	S.D.
6	-1.9173×10^{-8}	0.0358
10	-3.7260×10^{-8}	0.0350
15	-3.6574×10^{-8}	0.0350
20	-8.6478×10^{-8}	0.0340
25	-8.6569×10^{-8}	0.0340

Table 6.4: Stochastic Burger's equation $u(x, t = 0, \xi) = sin(2\pi x) + 0.01\xi$

6.3 Grid Adaptation

Wavelets are well suited for the development of highly efficient adaptive procedures, since scaling functions capture the average information and wavelets are responsible for the representation of the details. Adaptivity aims at capturing only the essential features of the solution of a PDE. While representing the solution using wavelet gPC, we generally have scaling and wavelet coefficients represented by $u^s_{J_0,k}$ and $u^w_{j,k}$ respectively. Wavelet coefficients play a vital role in a solution of a PDE as the value of wavelet coefficients decrease rapidly when the function is smooth. Moreover, if the function experiences discontinuity in one of its derivatives, then these coefficients tend to decrease gradually in the vicinity of discontinuity and exhibit rapid decay away from it. Therefore, they are known for identifying shocks in the numerical solution of a PDE. This means that the magnitude of wavelet coefficient plays a significant role in capturing the essential features of a solution. As we can know the wavelet representation of a stochastic process is given as

$$u(\xi \in [a, b]) = \sum_{k=-1}^{2^{J_0}-1} u^s_{J_0,k}\phi^s_{J_0,k}(\wp(\xi)) + \sum_{j=J_0}^{J}\sum_{k=-1}^{2^j-2} u^w_{j,k}\psi^w_{j,k}(\wp(\xi)).$$

We can discard wavelets with small coefficients from all the coefficients, thus, retaining a good approximation. The wavelet coefficients $u^w_{j,k}$ will be small unless the $u(\xi)$ has some fluctuation on the scale of j in the instant proximity of wavelets $\psi^w_{j,k}(\wp(\xi))$. Further, we divide the above

equation according to the magnitude

$$u(\xi \in [a,b]) = u_{\geq\epsilon}(\xi) + u_{<\epsilon}(\xi)) \tag{6.3.1}$$

where

$$u_{\geq\epsilon}(\xi \in [a,b]) = \sum_{k=-1}^{2^{J_0}-1} u^s_{J_0,k}\phi^s_{J_0,k}(\wp(\xi)) + \sum_{j=J_0}^{J}\sum_{\substack{k=-1 \\ |u^w_{j,k}|\geq\epsilon}}^{2^j-2} u^w_{j,k}\psi^w_{j,k}(\wp(\xi)). \tag{6.3.2}$$

$$u_{<\epsilon}(\xi \in [a,b]) = \sum_{\substack{k=-1 \\ |u^w_{j,k}|<\epsilon}}^{2^j-2} u^w_{j,k}\psi^w_{j,k}(\wp(\xi)). \tag{6.3.3}$$

Standard and modified adaptation are the two techniques available in literature for adaptivity. In this chapter, we will execute standard adaptation technique which says

- We need to initiate from finest level of resolution, i.e., $\jmath$
- Then, on the basis of magnitude of the wavelet coefficients, we remove the grid points which are less than ϵ.

6.3.1 Numerical Results

This section displays the adaptive grid modifications for the stochastic elliptic equation, stochastic advection equation and stochastic Burger's equation with uncertain data.

6.3.1.1 Stochastic Elliptic problem

Let us consider same stochastic elliptic equation as in Eq. 6.2.1 with $f = 1 + 0.1\xi$ where ξ is uniformly distributed over $[-1, 1]$. In fig 6.5, we can notice that the grid points are dense near

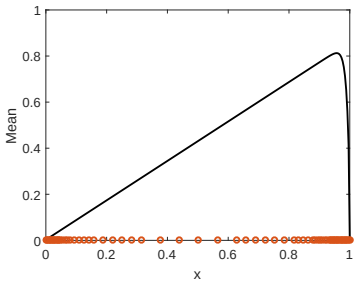


Figure 6.5: Stochastic elliptic problem with $P = 6,\ \epsilon = 10^{-4}$

the sharp variations. Out of 256 points, only 66 points were used to capture the essential features of the solution.

6.3.1.2 Stochastic Advection equation

Consider the stochastic advection equation

$$\frac{\partial u}{\partial t} + v\frac{\partial u}{\partial x} = 0, \qquad t \geq 0, \qquad x \in \mathbb{R} \tag{6.3.4}$$

with IC

$$u(x,0) = \exp(-200(x-0.25)^2) + 0.001\xi.$$

and Dirichlet BCs

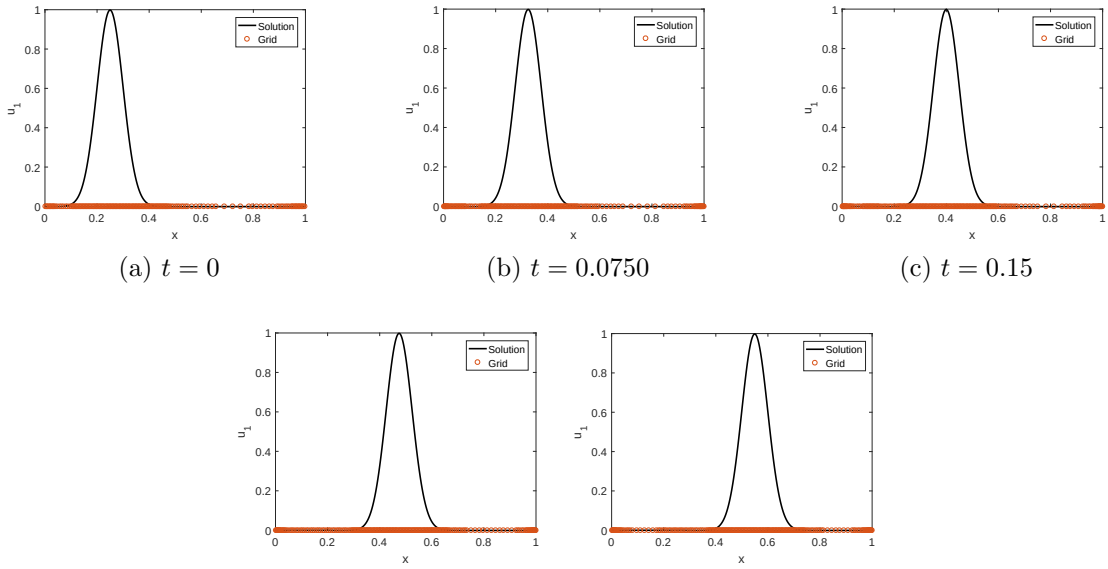
$$u(0,t) = 0,$$
$$u(1,t) = 0$$

We have considered $v = 1$ in this problem. In table 6.5, grid modifications at different time

Time	u_1	u_2	u_3	u_4	u_5	u_6
0	1024	1024	1024	1024	1024	1024
0.0375	162	148	136	148	162	162
0.0750	174	166	145	166	174	174
0.1125	175	169	135	169	175	175
0.15	179	174	136	174	179	179
0.1875	185	172	136	171	185	185
0.2250	179	172	136	172	179	179
0.2625	183	172	136	172	183	183
0.3	182	171	135	171	182	182

Table 6.5: Grid modifications for stochastic advection equation at different times by taking $\epsilon = 10^{-5}$

steps are presented and it is observed that at $t = 0.3$, u_1, u_5 and u_6 need same number of grid points. Fig. 6.6 shows how the solution and grid points are changing with the change in time. As a means of depicting the efficiency of developed method, we compared the CPU time taken for different values of ϵ and $\epsilon = 0$. Moreover, we have considered efficiency coefficient $\Theta = \frac{CPU(\epsilon=0)}{CPU(\epsilon)}$. From table 6.6, the proposed method is efficient as Θ increases with the increase in the value of ϵ.



(a) $t = 0$ (b) $t = 0.0750$ (c) $t = 0.15$

(d) $t = 0.2250$ (e) $t = 0.3$

Figure 6.6: Solution of stochastic advection equation (u_1) with P=6, $m = 2^{10}$.

ϵ	10^{-2}	10^{-3}	10^{-4}	10^{-5}
CPU(ϵ)	613.9	649.5	698.9	702.6
Θ	0.90	0.85	0.79	0.78

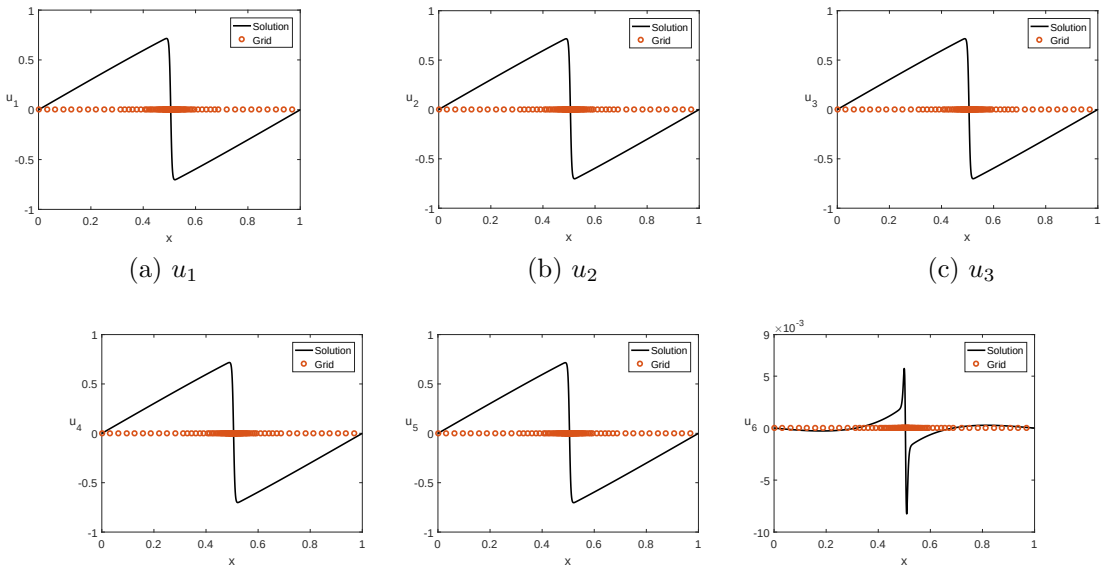
Table 6.6: Performance of the proposed method for stochastic advection equation

6.3.1.3 Stochastic Burger's equation

We have considered the same stochastic Burger's equation as in Eq. (6.2.9) with Dirichlet BCs. In Fig. 6.7, we have plotted the solution of Burger's equation at different times along with the grid modifications. It has been observed that the grid becomes dense when the solution shows sharp edges and it becomes sparse when the solution is smooth. Moreover, we can notice that the sixth coefficient which involves the wavelet function displays only the main details of the solution whereas all the other coefficients u_1, u_2, u_3, u_4, u_5 captures the average information. Table 6.7 displays the number of grid points of the solution at different time steps and it has been analyzed that the coefficient u_6 requires more number of grid points as it captures the details of the solution. The performance of the proposed method can be seen in table 6.8 which highlights that Θ increases on increasing the value of ϵ, thus, making it an efficient approach. Higher the value of Θ, more efficient is the algorithm.

ϵ	10^{-2}	10^{-3}	10^{-4}
CPU(ϵ)	2.34×10^3	2.66×10^3	3.75×10^3
Θ	0.9987	0.8721	0.265

Table 6.8: Performance of the proposed method for stochastic Burgers equation



(a) u_1 (b) u_2 (c) u_3

(d) u_4 (e) u_5 (f) u_6

Figure 6.7: Solution of stochastic Burgers' equation (all coeffs) at $t = 0.5$ with $P = 6$, $m = 2^{10}$.

$\nu = 0.002 + 0.001\xi \quad \epsilon = 10^{-3}$						
Time	u_1	u_2	u_3	u_4	u_5	u_6
0	1024	1024	1024	1024	1024	1024
0.0625	40	40	40	40	40	40
0.1250	42	42	42	42	42	50
0.1875	81	81	81	81	81	106
0.25	98	98	98	98	98	128
0.3125	107	107	107	107	107	129
0.3750	81	80	81	81	81	93
0.4375	80	80	83	83	80	94

Table 6.7: Grid modifications for stochastic Burger's equation at different times

6.4 Conclusion

We have proposed an adaptive wavelet collocation based gPC for numerically solving the stochastic PDEs. This wavelet collocation method uses the autocorrelation function of Daubechies compactly supported scaling function. Also, linear B-spline basis is implemented in gPC. The differential operators have been approximated using the connection coefficients. The fast interpolating wavelet transform and inverse wavelet transform are used to speed up the numerical scheme. Moreover, a good preconditioner is used for handling the condition number of matrices. The important features of the solution are captured using the adaptive scheme along with their grid modifications. The mean and S.D. are plotted for the three test problems. It has been concluded that the proposed method behaves well for the three test problems with satisfactory

results. Further, the proposed adaptive collocation scheme is an efficient algorithm as higher the value of Θ, more efficient is the algorithm.

Conclusion and Future Scope

Conclusion

In this thesis, B-spline wavelet based generalized polynomial chaos has been developed for numerically solving the stochastic differential equations. The most popular approach in uncertainty quantification is polynomial chaos which is a series like expansion that describes randomness in stochastic dynamical systems. First of all, we discussed a thorough literature review which is wrapped up in Table (6.9). Then, we have tried to extend the concept of polynomial chaos by introducing wavelets as the basis functions. A wavelet is a wave-like oscillation with an amplitude that begins at zero, increases, and then decreases back to zero. It can typically be visualized as a "brief oscillation" like one recorded by a seismograph or heart monitor. With wavelet analysis, we can use approximating functions that are contained neatly in finite domains.

First of all, we have united the traditional polynomial chaos method with the SBP-SAT operators for solving the stochastic partial differential equations. In this, we tried to find the stability conditions for partial differential equations with Dirichlet boundary conditions. Then, we have made an attempt to exploit useful properties of B-spline wavelets for stochastic solutions of PDEs, i.e., we shifted to developing wavelet based PCE in order to capture the sharp stochastic discontinuities which is a challenging task. As we know, wavelets are well-suited for approximating data with sharp discontinuities. So, we started with Linear B-spline wavelets and used their scaling and wavelet functions as the basis functions of polynomial chaos method. Then, we shifted to cubic B-spline wavelets by using their scaling functions only. A comparison analysis has been performed for linear and cubic B-spline wavelets.

Since, wavelet methods are known for their self adaptive nature which makes it a good choice for the numerical solution of a PDE. So, we worked on implementing adaptive schemes in the field of UQ. Wavelet optimized finite difference B-spline gPC scheme is employed in order to capture the essential features of the stochastic solution. Next, we investigated wavelet collocation B-spline gPC based on the concept of autocorrelation function of compactly supported Daubechies wavelets for the stochastic solutions of PDEs with Dirichlet boundary conditions. The postprocessing step which includes mean and variance are computed for each test problem.

UQ methods							
S.No.	**Methods**	**Recommended type of application**	**Suitability**	**Type of uncertainty for which it works best**	**Computational time comparison**	**Dimensionality**	**Convergence rate**
1.	Monte Carlo (MC) method	Any real life application	Any type of problem, linear, non-linear, as well as complex	Any distribution	Time consuming and computationally expensive	Last resort for high dimensional problems	slow convergence rate, independent of dimension of stochastic space
2.	KL expansion	problems where the random variable is correlated	problems with known covariance function	Gaussian distribution	computation of covariances eigen-pairs can be expensive	extensions of KL expansion handles multi-dimensionality	KL has an edge over the spectral method for highly correlated processes
3.	Polynomial Chaos Expansion (PCE)	any problem without discontinuity	non-linear problems	Gaussian distribution	Accurate results at lower cost than MC	curse of dimensionality	Exponential convergence
4.	generalized Polynomial Chaos (gPC)	any problem without discontinuity as it fails in discontinuities and long term integration	non-linear problems	Gaussian, Beta, Exponential, Uniform, Gamma	Accurate results at lower cost than MC	curse of dimensionality	Exponential convergence
5.	Wavelet based gPC	applications with discontinuities	handles discontinuities	Uniform distribution	computationally effective than gPC	curse of dimensionality	Fast convergence
6.	Multi-element gPC	handles long term integration	discontinuous problems	arbitrary distributions	high accuracy with negligible cost	Effective to employ low polynomial chaos order	more effective and efficient as compared to PCE
7.	Stochastic Collocation Method	handles complex non-linear problems	where solution is a smooth function of the random input variables and the dimension of the stochastic space is moderate	Uniform distribution	low computational cost and more efficient than PCE	Curse of dimensionality	faster convergence than MC

Table 6.9: Comparison of Uncertainty Quantification techniques

The proposed methods are tested on a number of numerical problems. Numerical results show that the developed methods are efficient in capturing the uncertain features of the stochastic differential equations. Moreover, the proposed method captures the adaptive solutions efficiently

along with grid refinements. More importantly, the strategy utilized in the manuscript appears to be a promising research path which differs significantly from existing approaches and shall attract more attention from the uncertainty quantification community.

Future Directions

- We have used B-spline wavelet in generalized polynomial chaos method for capturing the randomness of PDEs. However, the literature is extremely rich in various types of wavelets such as Morlet, Shannon Wavelets, coiflets, symlets, meyer, second generation wavelets, curvelets and so on. Morlet wavelet is closely related to human perception, both hearing and vision. Hence, this would be a great idea to explore different types of wavelets in the field of UQ.
- The computation of higher order moments like skewness and kurtosis is quite challenging as higher the moment, the harder it is to estimate. Therefore, one can work on computing the higher order moments with the help of wavelet based gPC.
- There are number of applications available in literature where we can explore our method by incorporating the real data. These applications can be in fluid dynamics, climate, brain tumour, cancer detection and so on.
- We can also look forward to higher dimensions of random variable using variants of polynomial chaos.

Bibliography

[1] M. Arnst, J.-P. Ponthot, An overview of nonintrusive characterization, propagation, and sensitivity analysis of uncertainties in computational mechanics, International Journal for Uncertainty Quantification 4 (2014) 387–421.

[2] O. P. Le Maître, L. Mathelin, O. Knio, M. Hussaini, Asynchronous time integration for polynomial chaos expansion of uncertain periodic dynamics, Discrete & Continuous Dynamical Systems 28 (2010) 199–226.

[3] T. Crestaux, O. P. Le Maître, J.-M. Martinez, Polynomial chaos expansion for sensitivity analysis, Reliability Engineering & System Safety 94 (7) (2009) 1161–1172.

[4] K. Alvin, W. Oberkampf, K. Diegert, B. Rutherford, Uncertainty quantification in computational structural dynamics: a new paradigm for model validation, in: Society for Experimental Mechanics, Inc, 16 th International Modal Analysis Conference., Vol. 2, 1998, pp. 1191–1198.

[5] C. J. Roy, W. L. Oberkampf, A comprehensive framework for verification, validation, and uncertainty quantification in scientific computing, Computer Methods in Applied Mechanics and Engineering 200 (25-28) (2011) 2131–2144.

[6] P. Renard, A. Alcolea, D. Gingsbourger, Stochastic versus deterministic approaches, in: Environmental Modelling: Finding Simplicity in Complexity, Second Edition (eds J. Wainwright and M. Mulligan), Wiley Online Library, 2013, pp. 133–149.

[7] W. L. Oberkampf, J. C. Helton, Evidence theory for engineering applications, in: Engineering Design Reliability Handbook, CRC Press, 2004, pp. 197–226.

[8] G. de Cooman, D. Ruan, E. E. Kerre, Foundations and Applications of Possibility Theory, World Scientific, 1995.

[9] E. Cox, O. Michael, The Fuzzy Systems Handbook: A Practitioner s Guide to Building, Using, and Maintaining Fuzzy Systems, Academic Press Professional Inc., 1994.

[10] C. Soize, Stochastic models of Uncertainties in Computational Mechanics, American Society of Civil Engineers Reston, 2012.

[11] C. Soize, A non-parametric model of random uncertainties for reduced matrix models in structural dynamics, Probabilistic Engineering Mechanics 15 (3) (2000) 277–294.

[12] O. Le Maître, O. M. Knio, Spectral methods for uncertainty quantification: with applications to computational fluid dynamics, Springer Science & Business Media, 2010.

[13] D. Xiu, Numerical methods for stochastic computations: a spectral method approach, Princeton University Press, 2010.

[14] M. Grigoriu, Stochastic systems: uncertainty quantification and propagation, Springer Science & Business Media, 2012.

[15] H. Bijl, D. Lucor, S. Mishra, C. Schwab, Uncertainty quantification in computational fluid dynamics, Vol. 92, Springer Science & Business Media, 2013.

[16] R. C. Smith, Uncertainty quantification: theory, implementation, and applications, Vol. 12, Siam, 2013.

[17] M. P. Pettersson, G. Iaccarino, J. Nordstrom, Polynomial Chaos Methods for Hyperbolic Partial Differential Equations: Numerical Techniques for Fluid Dynamics Problems in the Presence of Uncertainties., Springer, 2015.

[18] T. J. Sullivan, Introduction to uncertainty quantification, Vol. 63, Springer, 2015.

[19] R. W. Walters, L. Huyse, Uncertainty analysis for fluid mechanics with applications, Tech. rep., National Aeronautics and Space Administration Hampton va Langley Research Center (2002).

[20] N. Metropolis, S. Ulam, The Monte Carlo method, Journal of the American Statistical Association 44 (247) (1949) 335–341.

[21] J. M. Hammersley, D. Handscomb, Monte Carlo methods, Springer Science & Business Media, 2013.

[22] A. Cunha Jr, R. Nasser, R. Sampaio, H. Lopes, K. Breitman, Uncertainty quantification through the Monte Carlo method in a cloud computing setting, Computer Physics Communications 185 (5) (2014) 1355–1363.

[23] G. Fishman, Monte Carlo: concepts, algorithms, and applications, Springer Science & Business Media, 2013.

[24] C. Siva, M. S. Murugan, R. Ganguli, Uncertainty quantification in helicopter performance using Monte Carlo simulations, Journal of Aircraft 48 (5) (2011) 1503–1511.

[25] A. Dörr, M. Mögerle, M. Schneider, Monte Carlo methods in uncertainity quantification, Journal of Statistical Computation and Simulation 58 (2) (2014) 99–120.

[26] M. D. McKay, R. J. Beckman, W. J. Conover, Comparison of three methods for selecting values of input variables in the analysis of output from a computer code, Technometrics 21 (2) (1979) 239–245.

[27] W.-L. Loh, et al., On Latin hypercube sampling, The Annals of Statistics 24 (5) (1996) 2058–2080.

[28] H. Niederreiter, Random number generation and quasi-Monte Carlo methods, Vol. 63, Siam, 1992.

[29] M. B. Giles, Multilevel Monte Carlo path simulation, Operations Research 56 (3) (2008) 607–617.

[30] M. B. Giles, B. J. Waterhouse, Multilevel quasi-Monte Carlo path simulation, Advanced Financial Modelling, Radon Series on Computational and Applied Mathematics 8 (2009) 165–181.

[31] K. A. Cliffe, M. B. Giles, R. Scheichl, A. L. Teckentrup, Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients, Computing and Visualization in Science 14 (1) (2011) 1–3.

[32] R. E. Caflisch, Monte Carlo and quasi-Monte Carlo methods, Acta numerica 7 (1998) 1–49.

[33] D. Crevillen-Garcia, H. Power, Multilevel and quasi-Monte Carlo methods for uncertainty quantification in particle travel times through random heterogeneous porous media, Royal Society open science 4 (8) (2017) 170203(1–18).

[34] A. Barth, C. Schwab, N. Zollinger, Multi-level Monte Carlo finite element method for elliptic pdes with stochastic coefficients, Numerische Mathematik 119 (1) (2011) 123–161.

[35] S. Mishra, C. Schwab, J. Sukys, Multilevel Monte Carlo finite volume methods for shallow water equations with uncertain topography in multi-dimensions, SIAM Journal on Scientific Computing 34 (6) (2012) B761–B784.

[36] F. Müller, P. Jenny, D. W. Meyer, Multilevel Monte Carlo for two phase flow and Buckley–Leverett transport in random heterogeneous porous media, Journal of Computational Physics 250 (2013) 685–702.

[37] R. Scheichl, A. M. Stuart, A. L. Teckentrup, Quasi-Monte Carlo and multilevel Monte Carlo methods for computing posterior expectations in elliptic inverse problems, SIAM/ASA Journal on Uncertainty Quantification 5 (1) (2017) 493–518.

[38] M. Croci, V. Vinje, M. E. Rognes, Fast uncertainty quantification of tracer distribution in the brain interstitial fluid with multilevel and quasi Monte Carlo, arXiv preprint arXiv:2003.02311.

[39] K. Sepahvand, S. Marburg, H.-J. Hardtke, Uncertainty quantification in stochastic systems using polynomial chaos expansion, International Journal of Applied Mechanics 2 (02) (2010) 305–353.

[40] K. Karhunen, Zur spektraltheorie stochastischer prozesse, Annales Academiae Scientiarum Fennicae Mathematica 34.

[41] M. Kac, A. Siegert, An explicit representation of a stationary Gaussian process, The Annals of Mathematical Statistics 18 (3) (1947) 438–442.

[42] M. Loeve, Random functions of the second order, supplement to p, L 'evy, Stochastic Processes and Brownian Movement, Gauthier-Villars, Paris.

[43] A. Weinstein, et al., R. Courant and D. Hilbert, Methods of Mathematical Physics, Bulletin of the American Mathematical Society 60 (6) (1954) 578–579.

[44] I. Babuška, P. Chatzipantelidis, On solving elliptic stochastic partial differential equations, Computer Methods in Applied Mechanics and Engineering 191 (37–38) (2002) 4093–4122.

[45] K. Phoon, S. Huang, S. Quek, Simulation of second-order processes using Karhunen–Loeve expansion, Computers & Structures 80 (12) (2002) 1049–1060.

[46] H. G. Matthies, A. Keese, Galerkin methods for linear and nonlinear elliptic stochastic partial differential equations, Computer Methods in Applied Mechanics and Engineering 194 (12-16) (2005) 1295–1331.

[47] N. Wiener, The homogeneous chaos, American Journal of Mathematics 60 (4) (1938) 897–936.

[48] R. H. Cameron, W. T. Martin, The orthogonal development of non-linear functionals in series of Fourier-Hermite functionals, Annals of Mathematics 48 (2) (1947) 385–392.

[49] R. G. Ghanem, P. D. Spanos, Stochastic Finite Element Method: Response Statistics, in: Stochastic finite elements: A Spectral Approach, Springer, 1991, pp. 101–119.

[50] R. Ghanem, Stochastic finite elements with multiple random non-Gaussian properties, Journal of Engineering Mechanics 125 (1) (1999) 26–40.

[51] R. Ghanem, Ingredients for a general purpose stochastic finite elements implementation, Computer Methods in Applied Mechanics and Engineering 168 (1-4) (1999) 19–34.

[52] D. Xiu, G. E. Karniadakis, Modeling uncertainty in flow simulations via generalized polynomial chaos, Journal of Computational Physics 187 (1) (2003) 137–167.

[53] B. Puig, F. Poirion, C. Soize, Non-Gaussian simulation using hermite polynomial expansion: convergences and algorithms, Probabilistic Engineering Mechanics 17 (3) (2002) 253–264.

[54] D. Xiu, G. E. Karniadakis, The Wiener–Askey polynomial chaos for stochastic differential equations, SIAM Journal on Scientific Computing 24 (2) (2002) 619–644.

[55] O. Le Maître, O. Knio, H. Najm, R. Ghanem, Uncertainty propagation using Wiener-Haar expansions, Journal of Computational Physics 197 (1) (2004) 28–57.

[56] C. Pettit, P. Beran, Spectral and multiresolution Wiener expansions of oscillatory stochastic processes, Journal of Sound and Vibration 294 (4-5) (2006) 752–779.

[57] O. P. Le Maître, H. N. Najm, P. P. Pébay, R. G. Ghanem, O. M. Knio, Multi-resolution-analysis scheme for uncertainty quantification in chemical systems, SIAM Journal on Scientific Computing 29 (2) (2007) 864–889.

[58] H. N. Najm, B. J. Debusschere, Y. M. Marzouk, S. Widmer, O. Le Maître, Uncertainty quantification in chemical systems, International Journal for Numerical Methods in Engineering 80 (6-7) (2009) 789–814.

[59] L. Nechak, S. Berger, E. Aubry, Wiener–Haar expansion for the modeling and prediction of the dynamic behavior of self-excited nonlinear uncertain systems, Journal of Dynamic Systems, Measurement, and Control 134 (5) (2012) 01–11.

[60] T. Sahai, J. M. Pasini, Uncertainty quantification in hybrid dynamical systems, Journal of Computational Physics 237 (2013) 411–427.

[61] L. Wang, C. D. Sarris, A multi-resolution FDTD method for uncertainty quantification in the time-domain modeling of microwave structures, in: 2014 IEEE MTT-S International Microwave Symposium (IMS2014), IEEE, 2014, pp. 1–3.

[62] H. Shi, Q. Ma, N. Smith, F. Li, Data-driven uncertainty quantification and characterization for household energy demand across multiple time-scales, IEEE Transactions on Smart Grid 10 (3) (2018) 3092–3102.

[63] X. Wan, G. E. Karniadakis, Long-term behavior of polynomial chaos in stochastic flow simulations, Computer Methods in Applied Mechanics and Engineering 195 (41-43) (2006) 5582–5596.

[64] X. Wan, G. E. Karniadakis, Multi-element generalized polynomial chaos for arbitrary probability measures, SIAM Journal on Scientific Computing 28 (3) (2006) 901–928.

[65] X. Wan, G. E. Karniadakis, An adaptive multi-element generalized polynomial chaos method for stochastic differential equations, Journal of Computational Physics 209 (2) (2005) 617–642.

[66] X. Wan, G. E. Karniadakis, Beyond Wiener-Askey expansions: handling arbitrary pdfs, Journal of Scientific Computing 27 (1-3) (2006) 455–464.

[67] X. Wan, G. Karniadakis, Adaptive numerical solutions of stochastic differential equations, Computer Mathematics & its Applications (2006) 561–573.

[68] L. Bruno, C. Canuto, D. Fransos, Stochastic aerodynamics and aeroelasticity of a flat plate via generalised polynomial chaos, Journal of Fluids and Structures 25 (7) (2009) 1158–1176.

[69] P. Prempraneerach, F. S. Hover, M. S. Triantafyllou, G. E. Karniadakis, Uncertainty quantification in simulations of power systems: Multi-element polynomial chaos methods, Reliability Engineering & System Safety 95 (6) (2010) 632–646.

[70] L. Nechak, S. Berger, E. Aubry, A polynomial chaos approach to the robust analysis of the dynamic behaviour of friction systems, European Journal of Mechanics-A/Solids 30 (4) (2011) 594–607.

[71] L. Nechak, S. Berger, E. Aubry, Prediction of random self friction-induced vibrations in uncertain dry friction systems using a multi-element generalized polynomial chaos approach, Journal of Vibration and Acoustics 134 (4).

[72] M. H. Trinh, S. Berger, E. Aubry, Stability analysis of a clutch system with multi-element generalized polynomial chaos, Mechanics & Industry 17 (2) (2016) 205–218.

[73] C. Snoun, B. Bergeot, S. Berger, Prediction of the dynamic behavior of an uncertain friction system coupled to nonlinear energy sinks using a multi-element generalized polynomial chaos approach, European Journal of Mechanics-A/Solids 80 (2020) 103917(1–11).

[74] W. C. Meecham, A. Siegel, Wiener-Hermite expansion in model turbulence at large Reynolds numbers, The Physics of Fluids 7 (8) (1964) 1178–1190.

[75] A. Siegel, T. Imamura, W. C. Meecham, Wiener-Hermite expansion in model turbulence in the late decay stage, Journal of Mathematical Physics 6 (5) (1965) 707–721.

[76] W. C. Meecham, D.-T. Jeng, Use of the Wiener-Hermite expansion for nearly normal turbulence, Journal of Fluid Mechanics 32 (2) (1968) 225–249.

[77] S. A. Orszag, L. Bissonnette, Dynamical properties of truncated Wiener-Hermite expansions, The Physics of Fluids 10 (12) (1967) 2603–2613.

[78] S. Crow, G. Canavan, Relationship between a Wiener–Hermite expansion and an energy cascade, Journal of Fluid Mechanics 41 (2) (1970) 387–403.

[79] A. J. Chorin, Gaussian fields and random flow, Journal of Fluid Mechanics 63 (1) (1974) 21–32.

[80] O. Le Maître, O. M. KnIo, H. N. Najm, R. G. Ghanem, A stochastic projection method for fluid flow. I: Basic formulation, Journal of Computational Physics 173 (2) (2001) 481–511.

[81] O. Le Maître, M. T. Reagan, H. N. Najm, R. G. Ghanem, O. M. Knio, A stochastic projection method for fluid flow: II. Random process, Journal of Computational Physics 181 (1) (2002) 9–44.

[82] D. Xiu, D. Lucor, C.-H. Su, G. E. Karniadakis, Stochastic modeling of flow-structure interactions using generalized polynomial chaos, Journal of Fluids Engineering 124 (1) (2001) 51–59.

[83] D. Xiu, G. E. Karniadakis, A new stochastic approach to transient heat conduction modeling with uncertainty, International Journal of Heat and Mass Transfer 46 (24) (2003) 4681–4693.

[84] B. J. Debusschere, H. N. Najm, A. Matta, O. M. Knio, R. G. Ghanem, O. P. Le Maıtre, Protein labeling reactions in electrochemical microchannel flow: Numerical simulation and uncertainty propagation, Physics of Fluids 15 (8) (2003) 2238–2250.

[85] M. T. Reagana, H. N. Najm, R. G. Ghanem, O. M. Knio, Uncertainty quantification in reacting-flow simulations through non-intrusive spectral projection, Combustion and Flame 132 (3) (2003) 545–555.

[86] M. Reagan, H. Najm, B. Debusschere, O. Le Maître, O. Knio, R. Ghanem, Spectral stochastic uncertainty quantification in chemical systems, Combustion Theory and Modelling 8 (3) (2004) 607–632.

[87] M. T. Reagan, H. Najm, P. Pebay, O. Knio, R. Ghanem, Quantifying uncertainty in chemical systems modeling, International Journal of Chemical Kinetics 37 (6) (2005) 368–382.

[88] O. Le Maître, H. N. Najm, R. Ghanem, O. Knio, Multi-resolution analysis of Wiener-type uncertainty propagation schemes, Journal of Computational Physics 197 (2) (2004) 502–531.

[89] D. Lucor, C.-H. Su, G. E. Karniadakis, Generalized polynomial chaos and random oscillators, International Journal for Numerical Methods in Engineering 60 (3) (2004) 571–596.

[90] S. Masri, A. Smyth, M.-I. Traina, Probabilistic representation and transmission of nonstationary processes in multi-degree-of-freedom systems, Journal of Applied Mechanics (1998) 398–409.

[91] R. Ghanem, S. Dham, Stochastic finite element analysis for multiphase flow in heterogeneous porous media, Transport in Porous Media 32 (3) (1998) 239–262.

[92] R. Ghanem, Probabilistic characterization of transport in heterogeneous media, Computer Methods in Applied Mechanics and Engineering 158 (3-4) (1998) 199–220.

[93] Q.-Y. Chen, D. Gottlieb, J. S. Hesthaven, Uncertainty analysis for the steady-state flows in a dual throat nozzle, Journal of Computational Physics 204 (1) (2005) 378–398.

[94] T. Y. Hou, W. Luo, B. Rozovskii, H.-M. Zhou, Wiener chaos expansions and numerical solutions of randomly forced equations of fluid mechanics, Journal of Computational Physics 216 (2) (2006) 687–706.

[95] F. S. Hover, M. S. Triantafyllou, Application of polynomial chaos in stability and control, Automatica 42 (5) (2006) 789–795.

[96] R. Ghanem, P. D. Spanos, A stochastic Galerkin expansion for nonlinear random vibration analysis, Probabilistic Engineering Mechanics 8 (3-4) (1993) 255–264.

[97] R. Ghanem, Stochastic finite element analysis of randomly layered media, Journal of Engineering Mechanics 122 (4) (1996) 361–369.

[98] H. G. Matthies, C. E. Brenner, C. G. Bucher, C. G. Soares, Uncertainties in probabilistic numerical analysis of structures and solids-stochastic finite elements, Structural Safety 19 (3) (1997) 283–336.

[99] M. Anders, M. Hori, Stochastic finite element method for elasto-plastic body, International Journal for Numerical Methods in Engineering 46 (11) (1999) 1897–1916.

[100] B. Keshtegar, O. Kisi, M. Zounemat-Kermani, Polynomial chaos expansion and response surface method for nonlinear modelling of reference evapotranspiration, Hydrological Sciences Journal 64 (6) (2019) 720–730.

[101] A. C. Austin, N. Sood, J. Siu, C. D. Sarris, Application of polynomial chaos to quantify uncertainty in deterministic channel models, IEEE Transactions on Antennas and Propagation 61 (11) (2013) 5754–5761.

[102] O. Knio, O. Le Maître, Uncertainty propagation in CFD using polynomial chaos decomposition, Fluid Dynamics Research 38 (9) (2006) 616–640.

[103] H. Cheng, A. Sandu, Uncertainty quantification and apportionment in air quality models using the polynomial chaos method, Environmental Modelling & Software 24 (8) (2009) 917–925.

[104] A. Sandu, C. Sandu, M. Ahmadian, Modeling multibody systems with uncertainties. Part I: Theoretical and Computational aspects, Multibody System Dynamics 15 (4) (2006) 369–391.

[105] C. Sandu, A. Sandu, M. Ahmadian, Modeling multibody systems with uncertainties. Part II: Numerical applications, Multibody System Dynamics 15 (3) (2006) 241–262.

[106] G. Kewlani, J. Crawford, K. Iagnemma, A polynomial chaos approach to the analysis of vehicle dynamics under uncertainty, Vehicle System Dynamics 50 (5) (2012) 749–774.

[107] H. N. Najm, Uncertainty quantification and polynomial chaos techniques in computational fluid dynamics, Annual Review of Fluid Mechanics 41 (2009) 35–52.

[108] M. Villegas, F. Augustin, A. Gilg, A. Hmaidi, U. Wever, Application of the polynomial chaos expansion to the simulation of chemical reactors with uncertainties, Mathematics and Computers in Simulation 82 (5) (2012) 805–817.

[109] J. A. Witteveen, H. Bijl, Efficient quantification of the effect of uncertainties in advection-diffusion problems using polynomial chaos, Numerical Heat Transfer, Part B: Fundamentals 53 (5) (2008) 437–465.

[110] D. Gottlieb, D. Xiu, Galerkin method for wave equations with uncertain coefficients, Communications in Computational Physics 3 (2) (2008) 505–518.

[111] T. E. Lovett, A. Monti, F. Ponci, Automatic synthesis of uncertain models for linear circuit simulation: A polynomial chaos theory approach, Simulation Modelling Practice and Theory 16 (7) (2008) 796–816.

[112] C. Soize, C. Desceliers, Computational aspects for constructing realizations of polynomial chaos in high dimension, SIAM Journal on Scientific Computing 32 (5) (2010) 2820–2831.

[113] M. Gerritsma, J.-B. Van der Steen, P. Vos, G. Karniadakis, Time-dependent generalized polynomial chaos, Journal of Computational Physics 229 (22) (2010) 8333–8363.

[114] C. Y. Shen, T. E. Evans, S. Finette, Polynomial chaos quantification of the growth of uncertainty investigated with a Lorenz model, Journal of Atmospheric and Oceanic Technology 27 (6) (2010) 1059–1071.

[115] A. Prabhakar, J. Fisher, R. Bhattacharya, Polynomial chaos-based analysis of probabilistic uncertainty in hypersonic flight dynamics, Journal of Guidance, Control, and Dynamics 33 (1) (2010) 222–234.

[116] X. Jiang, Uncertainty quantification for mars atmospheric entry using polynomial chaos and spectral decomposition, in: 2018 AIAA Guidance, Navigation, and Control Conference, 2018, p. 1317.

[117] H. Panayirci, G.-I. Schuëller, On the capabilities of the polynomial chaos expansion method within SFE analysisan overview, Archives of Computational Methods in Engineering 18 (1) (2011) 43–55.

[118] F. Santonja, B. Chen-Charpentier, Uncertainty quantification in simulations of epidemics using polynomial chaos, Computational and Mathematical Methods in Medicine 2012.

[119] K. Sepahvand, S. Marburg, H.-J. Hardtke, Stochastic free vibration of orthotropic plates using generalized polynomial chaos expansion, Journal of Sound and Vibration 331 (1) (2012) 167–179.

[120] G. Perrin, C. Soize, D. Duhamel, C. Funfschilling, Identification of polynomial chaos representations in high dimension from a set of realizations, SIAM Journal on Scientific Computing 34 (6) (2012) A2917–A2945.

[121] W. Shi, C. Zhang, Error analysis of generalized polynomial chaos for nonlinear random ordinary differential equations, Applied Numerical Mathematics 62 (12) (2012) 1954–1964.

[122] R. Pulch, Polynomial chaos for boundary value problems of dynamical systems, Applied Numerical Mathematics 62 (10) (2012) 1477–1490.

[123] B. Pascual, S. Adhikari, Combined parametric–nonparametric uncertainty quantification using random matrix theory and polynomial chaos expansion, Computers & Structures 112 (2012) 364–379.

[124] A. O'Hagan, Polynomial chaos: A tutorial and critique from a statisticians perspective, SIAM/ASA J. Uncertainty Quantification 20 (2013) 1–20.

[125] K.-K. K. Kim, D. E. Shen, Z. K. Nagy, R. D. Braatz, Wiener's polynomial chaos for the analysis and control of nonlinear dynamical systems with probabilistic uncertainties [historical perspectives], IEEE Control Systems Magazine 33 (5) (2013) 58–67.

[126] A. Desai, J. A. Witteveen, S. Sarkar, Uncertainty quantification of a nonlinear aeroelastic system using polynomial chaos expansion with constant phase interpolation, Journal of Vibration and Acoustics 135 (5) (2013) 051034(1–13).

[127] K.-K. K. Kim, R. D. Braatz, Generalised polynomial chaos expansion approaches to approximate stochastic model predictive control, International Journal of Control 86 (8) (2013) 1324–1337.

[128] A. C. Austin, C. D. Sarris, Efficient analysis of geometrical uncertainty in the FDTD method using polynomial chaos with application to microwave circuits, IEEE Transactions on Microwave Theory and Techniques 61 (12) (2013) 4293–4301.

[129] Z. Zhang, T. A. El-Moselhy, I. M. Elfadel, L. Daniel, Calculation of generalized polynomial-chaos basis functions and Gauss quadrature rules in hierarchical uncertainty quantification, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 33 (5) (2014) 728–740.

[130] M. Navarro, J. Witteveen, J. Blom, Polynomial chaos expansion for general multivariate distributions with correlated variables, arXiv preprint arXiv:1406.5483.

[131] D. M. Luchtenburg, S. L. Brunton, C. W. Rowley, Long-time uncertainty propagation using generalized polynomial chaos and flow map composition, Journal of Computational Physics 274 (2014) 783–802.

[132] P. Delgado, V. Kumar, A stochastic Galerkin approach to uncertainty quantification in poroelastic media, Applied Mathematics and Computation 266 (2015) 328–338.

[133] V. Vittaldev, R. P. Russell, R. Linares, Spacecraft uncertainty propagation using gaussian mixture models and polynomial chaos expansions, Journal of Guidance, Control, and Dynamics 39 (12) (2016) 2615–2626.

[134] L. Daróczy, G. Janiga, D. Thévenin, Analysis of the performance of a H-Darrieus rotor under uncertainty using polynomial chaos expansion, Energy 113 (2016) 399–412.

[135] W. Jia, B. J. McPherson, F. Pan, T. Xiao, G. Bromhal, Probabilistic analysis of CO2 storage mechanisms in a CO2-EOR field using polynomial chaos expansion, International Journal of Greenhouse Gas Control 51 (2016) 218–229.

[136] J. Du, C. Roblin, Statistical modeling of disturbed antennas based on the polynomial chaos expansion, IEEE Antennas and Wireless Propagation Letters 16 (2016) 1843–1846.

[137] É. Savin, B. Faverjon, Computation of higher-order moments of generalized polynomial chaos expansions, International Journal for Numerical Methods in Engineering 111 (12) (2017) 1192–1200.

[138] J. Slim, F. Rathmann, A. Nass, H. Soltner, R. Gebel, J. Pretz, D. Heberling, Polynomial chaos expansion method as a tool to evaluate and quantify field homogeneities of a novel waveguide rf wien filter, Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment 859 (2017) 52–62.

[139] T. Mühlpfordt, R. Findeisen, V. Hagenmeyer, T. Faulwasser, Comments on truncation errors for polynomial chaos expansions, IEEE Control Systems Letters 2 (1) (2017) 169–174.

[140] Y. Xu, L. Mili, A. Sandu, M. R. von Spakovsky, J. Zhao, Propagating uncertainty in power system dynamic simulations using polynomial chaos, IEEE Transactions on Power Systems 34 (1) (2018) 338–348.

[141] Z. Hu, D. Du, Y. Du, Generalized polynomial chaos-based uncertainty quantification and propagation in multi-scale modeling of cardiac electrophysiology, Computers in Biology and Medicine 102 (2018) 57–74.

[142] B. J. Debusschere, H. N. Najm, P. P. Pébay, O. M. Knio, R. G. Ghanem, O. P. Le Maı tre, Numerical challenges in the use of polynomial chaos representations for stochastic processes, SIAM Journal on Scientific Computing 26 (2) (2004) 698–719.

[143] S. S. Isukapalli, Uncertainty analysis of transport-transformation models, Ph.D. thesis, New Brunswick Rutgers, The State University of New Jersey (1999).

[144] L. Mathelin, M. Y. Hussaini, T. A. Zang, Stochastic approaches to uncertainty quantification in CFD simulations, Numerical Algorithms 38 (1-3) (2005) 209–236.

[145] R. Walters, Stochastic fluid mechanics via polynomial chaos, in: 41st Aerospace Sciences Meeting and Exhibit, 2003, p. 413.

[146] S. Hosder, R. Walters, R. Perez, A non-intrusive polynomial chaos method for uncertainty propagation in cfd simulations, in: 44th AIAA aerospace sciences meeting and exhibit, 2006, p. 891.

[147] G. Loeven, J. Witteveen, H. Bijl, Probabilistic collocation: an efficient non-intrusive approach for arbitrarily distributed parametric uncertainties, in: 45th AIAA Aerospace Sciences Meeting and Exhibit, 2007, p. 317.

[148] X. Li, P. B. Nair, Z. Zhang, L. Gao, C. Gao, Aircraft robust trajectory optimization using non-intrusive polynomial chaos, Journal of Aircraft 51 (5) (2014) 1592–1603.

[149] K. Weise, L. Di Rienzo, H. Brauer, J. Haueisen, H. Toepfer, Uncertainty analysis in transcranial magnetic stimulation using nonintrusive polynomial chaos expansion, IEEE Transactions on Magnetics 51 (7) (2015) 1–8.

[150] M. Cooper, W. Wu, L. Mccue, Non-intrusive polynomial chaos for efficient uncertainty analysis in parametric roll simulations, Journal of Marine Science and Technology 21 (2) (2016) 282–296.

[151] K. Liang, Q. Sun, X. Liu, Investigation on imperfection sensitivity of composite cylindrical shells using the nonlinearity reduction technique and the polynomial chaos method, Acta Astronautica 146 (2018) 349–358.

[152] X. Liu, Q. Sun, Damage mechanics based probabilistic high-cycle fatigue life prediction for al 2024-t3 using non-intrusive polynomial chaos, Fatigue & Fracture of Engineering Materials & Structures.

[153] A. Alexanderian, J. Winokur, I. Sraj, A. Srinivasan, M. Iskandarani, W. C. Thacker, O. M. Knio, Global sensitivity analysis in an ocean general circulation model: a sparse spectral projection approach, Computational Geosciences 16 (3) (2012) 757–778.

[154] M. Mendes, S. Ray, J. Pereira, J. Pereira, D. Trimis, Quantification of uncertainty propagation due to input parameters for simple heat transfer problems, International Journal of Thermal Sciences 60 (2012) 94–105.

[155] L. Mathelin, M. Y. Hussaini, T. A. Zang, A stochastic collocation algorithm for uncertainty analysis, Tech. rep., National Aeronautics and Space Administration Hampton va Langley Research Center (2003).

[156] I. Babuška, F. Nobile, R. Tempone, A stochastic collocation method for elliptic partial differential equations with random input data, SIAM Journal on Numerical Analysis 45 (3) (2007) 1005–1034.

[157] F. Nobile, R. Tempone, C. G. Webster, A sparse grid stochastic collocation method for partial differential equations with random input data, SIAM Journal on Numerical Analysis 46 (5) (2008) 2309–2345.

[158] A. H. Stroud, Remarks on the disposition of points in numerical integration formulas, Mathematical Tables and Other Aids to Computation 11 (60) (1957) 257–261.

[159] Y. Ding, T. Li, D. Zhang, P. Zhang, Adaptive stroud stochastic collocation method for flow in random porous media via karhunen-loeve expansion, Communications in Computational Physics 4 (1) (2008) 102–123.

[160] D. Xiu, J. S. Hesthaven, High-order collocation methods for differential equations with random inputs, SIAM Journal on Scientific Computing 27 (3) (2005) 1118–1139.

[161] J. Baroth, P. Bressolette, C. Chauvière, M. Fogli, An efficient sfe method using lagrange polynomials: application to nonlinear mechanical problems with uncertain parameters, Computer Methods in Applied Mechanics and Engineering 196 (45-48) (2007) 4419–4429.

[162] P. Bonnet, F. Diouf, C. Chauvière, S. Lalléchère, M. Fogli, F. Paladian, Numerical simulation of a reverberation chamber with a stochastic collocation method, Comptes Rendus Physique 10 (1) (2009) 54–64.

[163] T. Tang, T. Zhou, Convergence analysis for stochastic collocation methods to scalar hyperbolic equations with a random wave speed, Communications in Computational Physics 8 (1) (2010) 226–248.

[164] H. Riahi, P. Bressolette, A. Chateauneuf, Random fatigue crack growth in mixed mode by stochastic collocation method, Engineering Fracture Mechanics 77 (16) (2010) 3292–3309.

[165] P. Bressolette, M. Fogli, C. Chauvière, A stochastic collocation method for large classes of mechanical problems with uncertain parameters, Probabilistic Engineering Mechanics 25 (2) (2010) 255–270.

[166] E. D. Fichtl, A. K. Prinja, The stochastic collocation method for radiation transport in random media, Journal of Quantitative Spectroscopy and Radiative Transfer 112 (4) (2011) 646–659.

[167] M. Motamed, F. Nobile, R. Tempone, A stochastic collocation method for the second order wave equation with a discontinuous random speed, Numerische Mathematik 123 (3) (2013) 493–536.

[168] A. M. DeGennaro, C. W. Rowley, L. Martinelli, Uncertainty quantification for airfoil icing using polynomial chaos expansions, Journal of Aircraft 52 (5) (2015) 1404–1411.

[169] S. A. Smolyak, Quadrature and interpolation formulas for tensor products of certain classes of functions, in: Doklady Akademii Nauk, Vol. 148, Russian Academy of Sciences, 1963, pp. 1042–1045.

[170] T. Gerstner, M. Griebel, Numerical integration using sparse grids, Numerical Algorithms 18 (3) (1998) 209–232.

[171] V. Barthelmann, E. Novak, K. Ritter, High dimensional polynomial interpolation on sparse grids, Advances in Computational Mathematics 12 (4) (2000) 273–288.

[172] F. Nobile, R. Tempone, C. G. Webster, An anisotropic sparse grid stochastic collocation method for partial differential equations with random input data, SIAM Journal on Numerical Analysis 46 (5) (2008) 2411–2442.

[173] S. Sankaran, A. L. Marsden, A stochastic collocation method for uncertainty quantification and propagation in cardiovascular simulations, Journal of Biomechanical Engineering 133 (3) (2011) 031001(1–12).

[174] B. Ganapathysubramanian, N. Zabaras, Sparse grid collocation schemes for stochastic natural convection problems, Journal of Computational Physics 225 (1) (2007) 652–685.

[175] J. He, S. Gao, J. Gong, A sparse grid stochastic collocation method for structural reliability analysis, Structural Safety 51 (2014) 29–34.

[176] W. Tian, W. Deng, Y. Wu, Polynomial spectral collocation method for space fractional advection–diffusion equation, Numerical Methods for Partial Differential Equations 30 (2) (2014) 514–535.

[177] A. W. Heemink, Stochastic modelling of dispersion in shallow water, Stochastic Hydrology and Hydraulics 4 (2) (1990) 161–174.

[178] L. Shi, L. Zeng, Y. Tang, C. Chen, J. Yang, Uncertainty quantification of contaminant transport and risk assessment with conditional stochastic collocation method, Stochastic Environmental Research and Risk Assessment 27 (6) (2013) 1453–1464.

[179] D. Datta, H. S. Kushwaha, Uncertainty quantification using stochastic response surface method case study-transport of chemical contaminants through groundwater, International Journal of Energy, Information and Communications 2 (3) (2011) 49–58.

[180] S. Berrone, S. Pieraccini, S. Scialò, Non-stationary transport phenomena in networks of fractures: Effective simulations and stochastic analysis, Computer Methods in Applied Mechanics and Engineering 315 (2017) 1098–1112.

[181] C. Ancey, P. Bohorquez, J. Heyman, Stochastic interpretation of the advection-diffusion equation and its relevance to bed load transport, Journal of Geophysical Research: Earth Surface 120 (12) (2015) 2529–2551.

[182] D. Galbally, K. Fidkowski, K. Willcox, O. Ghattas, Non-linear model reduction for uncertainty quantification in large-scale inverse problems, International Journal for Numerical methods in Engineering 81 (12) (2010) 1581–1608.

[183] I. Dranikov, P. Kondratenko, L. Matveev, Anomalous transport regimes in a stochastic advection-diffusion model, Journal of Experimental and Theoretical Physics 98 (5) (2004) 945–952.

[184] J. Erhel, Z. Mghazli, M. Oumouni, Numerical analysis of stochastic advection-diffusion equation via Karhunen-Loéve expansion, INRIA Research Report (2014) 1–26.

[185] K. Jinno, A. Kawamura, R. Berndtsson, M. Larson, J. Niemczynowicz, Real-time rainfall prediction at small space-time scales using a two-dimensional stochastic advection-diffusion model, Water Resources Research 29 (5) (1993) 1489–1504.

[186] V. A. B. Narayanan, N. Zabaras, Variational multiscale stabilized FEM formulations for transport equations: stochastic advection–diffusion and incompressible stochastic Navier–Stokes equations, Journal of Computational Physics 202 (1) (2005) 94–133.

[187] R. Mittal, S. Kumar, R. Jiwari, A cubic B-spline quasi-interpolation algorithm to capture the pattern formation of coupled reaction-diffusion models, Engineering with Computers (2021) 1–17.

[188] B. Strand, Summation by parts for finite difference approximations for d/dx, Journal of Computational Physics 110 (1994) 47–67.

[189] K. Mattsson, Summation by parts operators for finite difference approximations of second-derivatives with variable coefficients, Journal of Scientific Computing 51 (3) (2012) 650–682.

[190] K. Mattsson, J. Nordström, Summation by parts operators for finite difference approximations of second derivatives, Journal of Computational Physics 199 (2004) 503–540.

[191] M. Svärd, On coordinate transformations for summation-by-parts operators, Journal of Scientific Computing 20 (2004) 29–42.

[192] M. Svärd, K. Mattsson, J. Nordström, Steady-state computations using summation-by-parts operators, Journal of Scientific Computing 24 (2005) 79–95.

[193] J. Nordström, J. Gong, E. Van der Weide, M. Svärd, A stable and conservative high order multi-block method for the compressible Navier–Stokes equations, Journal of Computational Physics 228 (2009) 9020–9035.

[194] J. E. Hicken, D. W. Zingg, Parallel Newton-Krylov solver for the Euler equations discretized using simultaneous approximation terms, American Institute of Aeronautics and Astronautics (AIAA) Journal 46 (11) (2008) 2773–2786.

[195] D. C. D. R. Fernández, J. E. Hicken, D. W. Zingg, Review of summation-by-parts operators with simultaneous approximation terms for the numerical solution of partial differential equations, Computers & Fluids 95 (2014) 171–196.

[196] M. Jardak, C.-H. Su, G. E. Karniadakis, Spectral polynomial chaos solutions of the stochastic advection equation, Journal of Scientific Computing 17 (1) (2002) 319–338.

[197] N. Li, J. Zhao, X. Feng, D. Gui, Generalized polynomial chaos for the convection diffusion equation with uncertainty, International Journal of Heat and Mass Transfer 97 (2016) 289–300.

[198] M. El-Amrani, M. Seaid, N. L. Zaıdi, A new stochastic approach for advection-diffusion problems with uncertain parameters, Frontiers in Sciences and Technology International Journal, An International Journal Edited by Hassan II Academy of Science and Technology 2 (107).

[199] M. Zahri, K. Al Madinah, On numerical schemes for solving a stochastic advection-diffusion, International Journal of Pure and Applied Mathematics 77 (5) (2012) 681–694.

[200] R. Ghanem, Hybrid stochastic finite elements and generalized Monte Carlo simulation, Journal of Applied Mechanics 65 (4) (1998) 1004–1009.

[201] J. Witteveen, H. Bijl, Using polynomial chaos for uncertainty quantification in problems with nonlinearities, in: 47th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference 14th AIAA/ASME/AHS Adaptive Structures Conference 7th, 2006.

[202] M. P. Pettersson, G. Iaccarino, J. Nordström, Numerical analysis of the Burgers equation in the presence of uncertainty, Journal of Computational Physics 228 (22) (2009) 8394–8412.

[203] M. Wahlsten, J. Nordström, Stochastic Galerkin projection and numerical integration for stochastic investigations of the viscous Burger's equation (2018).

[204] M. Wahlsten, J. Nordström, On stochastic investigation of flow problems using the viscous Burger's equation as an example, Journal of Scientific Computing 81 (2) (2019) 1111–1117.

[205] E. Ullmann, Solution strategies for stochastic finite element discretizations, Ph.D. thesis (2008).

[206] M. T. Yassen, M. A. Sohaly, I. M. Elbaz, Stochastic solution for Cauchy one-dimensional advection model in mean square calculus, Journal of the Association of Arab Universities for Basic and Applied Sciences 24 (2017) 263–270.

[207] K. Kormann, M. Kronbichler, High order finite difference approximations for parabolic and hyperbolic-parabolic problems with variable coefficients, Project Report, Division of Scientific Computing, Department of Information Technology, Uppsala University.

[208] M. Mehra, K. Goyal, Algorithm 929: A suite on wavelet differentiation algorithms, ACM Transactions on Mathematical Software (TOMS) 39 (4) (2013) 27.

[209] M. Mehra, Ahmad, Wavelets Theory and Its Applications, Springer, 2018.

[210] C. K. Chui, E. Quak, Wavelets on a bounded interval, in: Numerical methods in approximation theory, Vol. 9, Springer, 1992, pp. 53–75.

[211] E. Quak, N. Weyrich, Decomposition and reconstruction algorithms for spline wavelets on a bounded interval, Applied and Computational Harmonic Analysis 1 (3) (1994) 217–231.

[212] L. Howle, A comparison of the reduced Galerkin and pseudo-spectral methods for simulation of steady Rayleigh-Bénard convection, International Journal of Heat and Mass Transfer 39 (12) (1996) 2401–2407.

[213] J. C. Goswami, A. K. Chan, C. K. Chui, On solving first-kind integral equations using wavelets on a bounded interval, IEEE Transactions on Antennas and Propagation 43 (6) (1995) 614–622.

[214] P. Sahu, S. S. Ray, A new approach based on semi-orthogonal B-spline wavelets for the numerical solutions of the system of nonlinear Fredholm integral equations of second kind, Computational and Applied Mathematics 33 (3) (2014) 859–872.

[215] M. Lakestani, M. Razzaghi, M. Dehghan, Semiorthogonal spline wavelets approximation for Fredholm integro-differential equations, Mathematical Problems in Engineering 2006.

[216] C. K. Chui, J.-Z. Wang, On compactly supported spline wavelets and a duality principle, Transactions of the American Mathematical Society 330 (2) (1992) 903–915.

[217] R. H. Kraichnan, Direct-interaction approximation for a system of several interacting simple shear waves, The Physics of Fluids 6 (11) (1963) 1603–1609.

[218] W. O. Kermack, A. G. McKendrick, A contribution to the mathematical theory of epidemics, Proceedings of the Royal society of London. Series A, containing papers of a mathematical and physical character 115 (772) (1927) 700–721.

[219] M. Bartlett, Some evolutionary stochastic processes, Journal of the Royal Statistical Society. Series B (Methodological) 11 (2) (1949) 211–229.

[220] D. G. Kendall, Deterministic and stochastic epidemics in closed populations, in: Proc. 3rd Berkeley Symp. Math. Statist. Prob, Vol. 4, 1956, pp. 149–165.

[221] H. W. Hethcote, The mathematics of infectious diseases, SIAM Review 42 (4) (2000) 599–653.

[222] M. Gunzburger, C. G. Webster, G. Zhang, An adaptive wavelet stochastic collocation method for irregular solutions of partial differential equations with random input data, in: Sparse Grids and Applications-Munich 2012, Springer, 2014, pp. 137–170.

[223] B. Kegan, R. W. West, Modeling the simple epidemic with deterministic differential equations and random initial conditions, Mathematical Biosciences 195 (2) (2005) 179–193.

[224] B. M. Chen-Charpentier, D. Stanescu, Epidemic models with random coefficients, Mathematical and Computer Modelling 52 (7-8) (2010) 1004–1010.

[225] M. Roberts, Epidemic models with uncertainty in the reproduction number, Journal of Mathematical Biology 66 (7) (2013) 1463–1474.

[226] A. H. A. Omar, Y. A. Hasan, Numerical simulations of an SIR epidemic model with random initial states, Science Asia S 39 (2013) 42–47.

[227] D. B. Harman, P. R. Johnston, Applying the stochastic Galerkin method to epidemic models with uncertainty in the parameters, Mathematical Biosciences 277 (2016) 25–37.

[228] K. Maleknejad, K. Nouri, M. N. Sahlan, Convergence of approximate solution of nonlinear fredholm–hammerstein integral equations, Communications in Nonlinear Science and Numerical Simulation 15 (6) (2010) 1432–1443.

[229] K. Maleknejad, R. Mollapourasl, M. Shahabi, On the solution of a nonlinear integral equation on the basis of a fixed point technique and cubic B-spline scaling functions, Journal of Computational and Applied Mathematics 239 (2013) 346–358.

[230] J. Murray, Mathematical Biology, 1993.

[231] C. D. S. C. 1978, News and Notes: Influenza in a Boarding School, British Medical Journal 1 (6112) 586–590.

[232] A. Saltelli, S. Tarantola, F. Campolongo, M. Ratto, Sensitivity Analysis in Practice: A Guide to Assessing Scientific Models, Vol. 1, Wiley Online Library, 2004.

[233] S. Y. Balaman, Decision-making for Biomass-based Production Chains: The Basic Concepts and Methodologies, Academic Press, 2018.

[234] S. C. Warder, Sensitivity analysis, uncertainty quantification and parameter estimation for a numerical tide and storm surge model, Ph.D. thesis, Imperial College London (2020).

[235] C. for Disease Control, Prevention, Appendix: Additional results and technical notes for the ebolaresponse modeling tool, Centers for Disease Control and Prevention.
URL https://www.cdc.gov/mmwr/preview/mmwrhtml/su63e0923a2.htm

[236] W. H. O., Ebola Virus Disease Update - West Africa (19 August 2014).
URL https://www.who.int/csr/don/2014_08_19_ebola/en/

[237] M. I. Meltzer, C. Y. Atkins, S. Santibanez, B. Knust, B. W. Petersen, E. D. Ervin, S. T. Nichol, I. K. Damon, M. L. Washington, Estimating the future number of cases in the ebola epidemic–Liberia and Sierra Leone, 2014–2015.

[238] M. Suthanthirakumaran, Modelling Ebola Using an SIR Model.

[239] R. Jiwari, A Haar wavelet quasilinearization approach for numerical simulation of Burger's equation, Computer Physics Communications 183 (11) (2012) 2413–2423.

[240] W. Dahmen, Wavelet and multiscale methods for operator equations, Citeseer, 1997.

[241] V. Kumar, K. K. Sharma, An optimized B-spline method for solving singularly perturbed differential difference equations with delay as well as advance, Neural, Parallel and Scientific Computations 16 (3) (2008) 371.

[242] J. Fröhlich, K. Schneider, An adaptive wavelet Galerkin algorithm for one and two dimensional flame computations, Journal of Computational Physics (1993) 439–471.

[243] L. Gagnon, J. Lina, Symmetric Daubechies' wavelets and numerical solution of NLS equations, Journal of Physics A: Mathematical and General 27 (24) (1994) 8207–8230.

[244] J. Liandrat, P. Tchamitchian, Resolution of the 1D regularized Burger's equation using a spatial wavelet approximation, Tech. rep., Institute for Computer Applications in Science and Engineering Hampton VA (1990).

[245] L. Zhang, J. Ouyang, X. Wang, X. Zhang, Variational multiscale element-free Galerkin method for 2D Burger's equation, Journal of Computational Physics 229 (19) (2010) 7147–7161.

[246] M. J. Berger, P. Colella, et al., Local adaptive mesh refinement for shock hydrodynamics, Journal of Computational Physics 82 (1) (1989) 64–84.

[247] L. Jameson, A wavelet-optimized, very high order adaptive grid and order numerical method, SIAM Journal on Scientific Computing 19 (6) (1998) 1980–2013.

[248] J. M. Alam, N. K.-R. Kevlahan, O. V. Vasilyev, Simultaneous space–time adaptive wavelet solution of nonlinear parabolic differential equations, Journal of Computational Physics 214 (2) (2006) 829–857.

[249] T. Plewa, T. Linde, V. G. Weirs, et al., Adaptive mesh refinement-theory and applications, Journal of Mechanical Design 134.

[250] J. A. Witteveen, G. Iaccarino, Simplex stochastic collocation with random sampling and extrapolation for nonhypercube probability spaces, SIAM Journal on Scientific Computing 34 (2) (2012) A814–A838.

[251] J. Mohapatra, S. Natesan, Uniform convergence analysis of finite difference scheme for singularly perturbed delay differential equation on an adaptively generated grid, Numerical Mathematics: Theory, Methods and Applications 3 (1) (2010) 1–22.

[252] W. Liao, J. Zhu, Efficient and accurate finite difference schemes for solving one-dimensional Burgers equation, International Journal of Computer Mathematics 88 (12) (2011) 2575–2590.

[253] D. Sharma, K. Goyal, Spectral graph wavelet optimized finite difference method for solution of Burger's equation with different boundary conditions, Journal of Difference Equations and Applications 25 (3) (2019) 373–395.

[254] N. A. Libre, A. Emdadi, E. J. Kansa, M. Shekarchi, M. Rahimian, A multiresolution prewavelet-based adaptive refinement scheme for RBF approximations of nearly singular problems, Engineering Analysis with Boundary Elements 33 (7) (2009) 901–914.

[255] L. Jameson, On the wavelet optimized finite difference method, Tech. rep., Institute for Computer Applications in Science and Engineering Hampton VA (1994).

[256] L. Jameson, A wavelet-optimized, very high order adaptive grid and order numerical method, SIAM Journal on Scientific Computing 19 (6) (1998) 1980–2013.

[257] K. Goyal, M. Mehra, A fast adaptive diffusion wavelet method for Burger's equation, Computers & Mathematics with Applications 68 (4) (2014) 568–577.

[258] K. Goyal, M. Mehra, Fast diffusion wavelet method for partial differential equations, Applied Mathematical Modelling 40 (7-8) (2016) 5000–5025.

[259] M. Mehra, N. K.-R. Kevlahan, An adaptive wavelet collocation method for the solution of partial differential equations on the sphere, Journal of Computational Physics 227 (11) (2008) 5610–5632.

[260] O. V. Vasilyev, C. Bowman, Second-generation wavelet collocation method for the solution of partial differential equations, Journal of Computational Physics 165 (2) (2000) 660–693.

[261] S. Singh, V. K. Patel, V. K. Singh, Application of wavelet collocation method for hyperbolic partial differential equations via matrices, Applied Mathematics and Computation 320 (2018) 407–424.

[262] O. V. Vasilyev, S. Paolucci, M. Sen, A multilevel wavelet collocation method for solving partial differential equations in a finite domain, Journal of Computational Physics 120 (1) (1995) 33–47.

[263] D. L. Donoho, Interpolating wavelet transforms, Preprint, Department of Statistics, Stanford University 2 (3) (1992) 1–54.

[264] A. Garba, A wavelet collocation method for the numerical solution of Burger's equation, Tech. rep., International Centre for Theoretical Physics (1996).

[265] S. Bertoluzza, G. Naldi, A wavelet collocation method for the numerical solution of partial differential equations, Applied and Computational Harmonic Analysis 3 (1) (1996) 1–9.

[266] G. Beylkin, Wavelets, multiresolution analysis and fast numerical algorithms, Wavelets Theory and Applications, Oxford University Press, New York, Oxford (1996) 182–262.

[267] A. Latto, The evaluation of connection coefficients of compactly supported wavelets, in: Proc. French-USA workshop on wavelets and turbulence, Springer, 1992.

[268] V. A. Barker, Computing connection coefficients.

[269] O. M. Nielsen, Wavelets in scientific computing, Ph.D. thesis, Technical University of Denmark (1998).

[270] C.-R. Beylkin, Gregory, V. Rokhlin, Fast wavelet transforms and numerical algorithms I, Communications on Pure and Applied Mathematics 44 (2) (1991) 141–183.

[271] G. Beylkin, Wavelets and fast numerical algorithms, Different perspectives on wavelets 47 (1993) 89–117.

[272] A. ORCK, J. Yuan, Preconditioners for least squares problems by LU factorization, Electronic Transactions on Numerical Analysis 8 (1999) 26–35.

Bio-Data

Full Name : Navjot Kaur

Education

- Ph.D.
  - Joined in July 2017.
  - Research Topic: Variants of Polynomial Chaos Methods in Uncertainty Quantification
  - Courses undertaken during course work:
    Numerical Analysis for Partial Differential Equations (PMC411)
    Research methodology (DMC007)
  - Passed the Pre-Ph.D. course work with C.G.P.A. of 9.63 on a 10 point scale.
- M.Sc. (Applied Mathematics and Computing)
  Passed in July, 2016 with 7.35 C.G.P.A. from Department of Mathematics, Punjabi University, Patiala, India.
- B.Sc. (Mathematics, Computer Applications and Management)
  Passed in July, 2014 with 77.7% marks from Department of Mathematics, Punjabi University, Patiala, India.
- Higher Secondary
  Passed in year 2011 with 79.4% marks from Budha Dal Public School, Patiala, India.
- High School
  Passed in year 2009 with 89.6% marks from Budha Dal Public School, Patiala, India.

Conferences and Workshops Attended

- Workshop on Advanced Computational Techniques for Differential Equations with MATLAB (ACTDEM 2018) organized by Department of Mathematics, IIT Roorkee, September, 2018.
- Workshop on Applications of Optimization Techniques organized by Thapar Mathematical Society (TMS) in collaboration with Scimatics, Thapar Institute of Engineering and Technology, Patiala, March, 2018.
- Workshop on MATLAB And Differential Equations (MATDIFF2020), TIET, Patiala, 2020.
- International Conference on Differential Equations and Control Problems: Modeling, Analysis and Computations held at IIT Mandi, India, June 17-19, 2019.

- National conference on Number Theory, Combinatorics, & Special Functions (NTCSF-2019) at Thapar Institute of Engineering and Technology, Patiala, October 11-12, 2019.
- Two-day workshop on "Mathematical Modeling and Computational Techniques (MMCT)", Thapar Institute of Engineering and Technology, Patiala, Punjab, India, Februray 24-25, 2020.
- Two-day workshop on "Recent Advances in the Numerical Approximation of Partial Differential Equations (RANAPDE2021)", Universit degli Studi di Milano, Italy, June 24 and 25, 2021.

Teaching Experience

- Teaching Assistant at TIET for the following courses:
  - UMA003 (Mathematics-I).
  - UMA004 (Mathematics-II).
  - UMA007 (Numerical Analysis).
  - UMA031 (Optimization Techniques).

Honours

- Qualified GATE-2018 with 84.81 percentile.