

Adaptive Methods Exploring Intrinsic Sparse Structures of Stochastic Partial Differential Equations

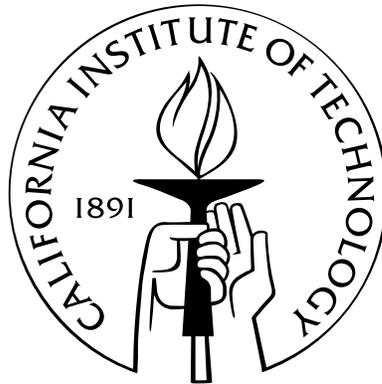
Thesis by

Mulin Cheng

In Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy



California Institute of Technology

Pasadena, California

2013

(Defended September 12, 2012)

© 2013

Mulin Cheng

All Rights Reserved

To

my wife

Lu Chi,

and my parents

Xiancai Cheng & Guihua Zhang

Acknowledgements

I owe my deepest gratitude to my adviser Prof. Thomas Y. Hou. Without his constant inspirations, encouragements, and challenges, I would not have had the chance to complete this thesis in the present form. A discussion from 10 a.m. to 2 a.m. not only marked the turning point in my research, but also enlightened and will continue to enlighten me throughout my future career and life.

Besides my adviser, I would like to thank the rest of thesis committee — Professor Houman Owahdi, Daniel I. Meiron, and James L. Beck — for taking time to review and comment on my thesis. I would also like to thank the staff of ACM department — Sydney Gastang, Carmen Nemer-Sirois, William Yardley, and Sheila Shull — for their help over the past years.

In addition, I would like to thank Prof. Yalchin Efendiev and Dr. Xiaohui Wu for their hospitality and many inspiring discussions when I visited them. My sincere thanks also go to Prof. Lexing Ying, for help and endless suggestions on living and research when he was a postdoc here.

I am grateful to Dr. Pengchong Yan, Dr. Zuoqiang Shi, and Dr. Zhiwen Zhang for many stimulating discussions over the past three years. I thank my friends at Caltech — Maolin Ci, Xin Hu, Peyman Tavallali, Wuan Luo, Lei Zhang, Guo Luo, Chia-Chieh Chu, Zhen Li, Yuebin Liu, Yao Sha, Yan Xia, Stephen Becker, Catherine Beni, Yaniv Plan and Svitlana Vyetenko — for the help, the fun, and the good time we had together.

Most importantly, I would like to thank my beloved parents: Mr. Xiancai Cheng and Mrs. Guihua Zhang, for their endless love and support. No words can express how grateful I am for you

and how much I love you.

Last but not the least, my thanks to my lovely wife, Lu Chi. You have supported me in the darkest times and believed in me even when I did not believe in myself. Without you, I could not have survived this strenuous Ph.D. journey, a journey you have just started and I will be there with you all the time. This thesis is dedicated to you and my parents with all my heart.

Abstract

Many physical and engineering problems involving uncertainty enjoy certain low-dimensional structures, e.g., in the sense of Karhunen-Loeve expansions (KLEs), which in turn indicate the existence of reduced-order models and better formulations for efficient numerical simulations. In this thesis, we target a class of time-dependent stochastic partial differential equations whose solutions enjoy such structures at any time and propose a new methodology (DyBO) to derive equivalent systems whose solutions closely follow KL expansions of the original stochastic solutions. KL expansions are known to be the most compact representations of stochastic processes in an L^2 sense. Our methods explore such sparsity and offer great computational benefits compared to other popular generic methods, such as traditional Monte Carlo (MC), generalized Polynomial Chaos (gPC) method, and generalized Stochastic Collocation (gSC) method. Such benefits are demonstrated through various numerical examples ranging from spatially one-dimensional examples, such as stochastic Burgers' equations and stochastic transport equations to spatially two-dimensional examples, such as stochastic flows in 2D unit square. Parallelization is also discussed, aiming toward future industrial-scale applications. In addition to numerical examples, theoretical aspects of DyBO are also carefully analyzed, such as preservation of bi-orthogonality, error propagation, and computational complexity. Based on theoretical analysis, strategies are proposed to overcome difficulties in numerical implementations, such as eigenvalue crossing and adaptively adding or removing mode pairs. The effectiveness of the proposed strategies is numerically verified. Generalization to a system of SPDEs

is considered as well in the thesis, and its success is demonstrated by applications to stochastic Boussinesq convection problems. Other generalizations, such as generalized stochastic collocation formulation of DyBO method, are also discussed.

Contents

Acknowledgements	iv
Abstract	vi
1 Introduction	1
1.1 Uncertainty Quantification and Stochastic Partial Differential Equations	2
1.2 Existing Numerical Methods	4
1.2.1 Non-Sampling Methods	4
1.2.2 Non-Intrusive Sampling Methods	5
1.2.3 Intrusive Methods	7
1.2.4 Post-Processing	9
1.3 Exploring Sparsity	10
1.3.1 Karhunen-Loeve Expansions	11
1.3.2 Overview of Dynamically Bi-Orthogonal Methods	15
1.4 Summary of the Thesis	17
1.4.1 Summary of Main Contributions	18
1.4.2 Roadmap	20
2 Dynamically Bi-Orthogonal (DyBO) Method	21
2.1 Derivations of DyBO Method	24

2.1.1	Derivations of Reduced-Order Model for Truncated KLE	25
2.1.2	Enforcing Bi-Orthogonality Condition	27
2.1.3	Eliminating Time Derivatives from RHS	29
2.1.4	DyBO Formulation for Time-Evolutionary SPDE	33
2.2	Bi-Orthogonality Preservation	35
2.3	Representation of Stochastic Modes	40
2.3.1	Ensemble Representation	40
2.3.2	Spectral Representation	41
2.3.2.1	Some Preliminaries of gPC Methods	41
2.3.2.2	gPC Version of DyBO	44
2.3.3	gSC Version of DyBO	46
2.4	Eigenvalue Crossing	47
2.4.1	Detection of Eigenvalue Crossing	48
2.4.2	Freeze \mathbf{Y} or \mathbf{Y} -Stage	48
2.4.3	Freeze \mathbf{U} or U -Stage	52
2.5	Error Analysis	56
2.5.1	Type-0 Errors	56
2.5.2	Type- \mathfrak{J} Error	60
2.5.3	Type-KL Error	62
2.6	Adding or Removing Mode Pairs	64
2.6.1	Removing Modes	64
2.6.2	Adding Modes	65
2.7	Overall DyBO-gPC Algorithm	71
2.7.1	Initializations	71

2.7.2	DyBO-gPC Algorithm	72
3	Applications to Spatially One-dimensional SPDEs	73
3.1	SPDE Purely Driven by Stochastic Force	74
3.1.1	Eigenvalue Crossing	78
3.1.2	Adding and Removing Modes	79
3.2	Linear Deterministic Differential Operators with Random Initial Conditions	82
3.3	Burgers' Equation Driven by Stochastic Forces	85
3.3.1	gPC Formulation for Stochastic Burgers' Equation	86
3.3.2	DyBO-gPC Formulation for Stochastic Burgers' Equation	89
3.3.3	Numerical Examples	93
3.3.3.1	Hierarchy of Errors	95
3.3.3.2	Numerical Results	97
4	Applications to Spatially Two-Dimensional SPDE	102
4.1	Computational Complexity Analysis	102
4.1.1	Storage Complexity	103
4.1.2	Computational Cost	104
4.1.2.1	Linear PDE Driven by Stochastic Forces	105
4.1.2.2	Second-Order Nonlinear PDE Driven by Stochastic Forces	107
4.2	Stochastic Navier-Stokes Equations	110
4.2.1	gPC and DyBO-gPC Formulations of SNSE	113
4.2.2	Numerical Results	114
5	Generalization to Stochastic Processes $\mathbb{L}^2(\Omega \rightarrow \mathcal{H}^k(\mathcal{D}))$	126
5.1	Necessary Condition for Spatial Modes in $\mathbb{L}^2(\Omega \rightarrow \mathcal{V}(\mathcal{D}))$	127

5.2	Generalized KLE for $\mathbb{L}^2(\Omega \rightarrow \mathcal{H}^k(\mathcal{D}))$	129
5.3	Generalized DyBO for $\mathbb{L}^2(\Omega \rightarrow \mathcal{H}^k(\mathcal{D}))$	134
6	Generalizations of DyBO for a System of Time-Dependent SPDEs	136
6.1	DyBO for a System of SPDEs	137
6.2	Stochastic Navier-Stokes Equations with Boussinesq Approximation	140
6.2.1	gPC Formulation of SNSE	143
6.2.2	DyBO Formulation of SNSE	144
6.3	Numerical Results	145
6.4	Parallelization	147
6.4.1	Parallelization Strategy	151
6.4.2	Implementation and Speedup	153
7	Generalized Stochastic Collocation Formulation of DyBO Method (DyBO-gSC)	155
7.1	Formulation and Algorithm	155
7.1.1	Sparse Grid	156
7.1.2	Generalized Stochastic Collocation Method (gSC)	157
7.1.3	gSC version of DyBO Method (DyBO-gSC)	159
7.2	Numerical Example — Stochastic Burgers Equation	161
8	Conclusions and Future Work	164
A	Derivation of the gPC formulation of Time-Evolutionary SPDE	170
B	Proof of Corollary 2.8 and 2.9	172
C	Derivation of DyBO Methods for a System of Time-Dependent SPDEs	174

D Derivations of gPC Formulation of SNSE	181
E Derivations of the DyBO Formulation of SNSE	186
Bibliography	193

List of Figures

1.1	Conceptual illustration of efficiency of gPC, gSC, MC, and CM with respect to dimensionality of random space	9
1.2	The numerical simulation process of SPDEs	10
2.1	Illustration of eigenvalue crossing and two strategies, Y -Stage and U -Stage	47
2.2	Illustration of strategies of adding and removing mode pairs	69
3.1	Mean and STD computed by DyBO at time $t = 1.2$	79
3.2	Spatial modes in KL expansion of SPDE solution computed by DyBO at time $t = 1.2$	79
3.3	Eigenvalues computed by DyBO. Two zooming-outs are provided at the time when two eigenvalues cross each other to show the invocation of U -stage Algorithm 2.2	80
3.4	Bi-orthogonality of spatial and stochastic modes. Orthogonality of U is preserved throughout the computation while stochastic modes deviate from orthogonality only at eigenvalue crossing	80
3.5	Eigenvalues. Eigenvalues are plotted as function of time. λ_3 becomes small near $t = 0.25$	81
3.6	Change rate of the largest unresolved eigenvalue $\frac{d\sqrt{\lambda_3}}{dt}$. Solid line is given by the exact solution, while the dotted line are computed as described in Sec. 2.6.2	82
3.7	L^2 relative errors of STD given by DyBO for different invoking frequency of Algorithm 2.4 for adding or removing mode pairs	82

3.8	Spatial modes in KL expansion of SPDE solution computed by DyBO at time $t = 0.4$	85
3.9	\mathbb{L}^2 relative errors of STD computed by DyBO with different spatial and temporal grid sizes. The horizontal axis is time t	85
3.10	Hierarchy of solutions	97
3.11	\mathbb{L}^2 relative errors of mean and STD as functions of time	99
3.12	The first nine spatial modes computed by DyBO at time $t = 1.0$	100
3.13	Stochastic modes HA computed by DyBO at time $t = 1.0$	101
4.1	Stochastic flows driven by stochastic force f in 2D unit square	113
4.2	Wall time of a single RK step of the DyBO-gPC system as a function of m	117
4.3	Mean and STD of vorticity field at time $t = 1.0$. The left column is computed by DyBO-gPC, while the right column is computed by gPC. They are essentially the same.	118
4.4	The first four spatial modes at time $t = 1.0$. The left column is computed by DyBO-gPC, while the right column is computed by gPC. They are essentially the same.	119
4.5	The \mathbb{L}^2 relative errors of mean and STD of vorticity field computed by DyBO. The errors are plotted as functions of time in the top two figures, while the numbers of mode pairs used in the adaptive strategy are given in the last figure.	120
4.6	Comparison of energy spectrum of gPC and DyBO solutions at time $t = 1.0$	122
4.7	Comparisons of gPC and DyBO methods	124
4.8	Stochastic modes computed by DyBO and gPC	125

6.1	Stochastic flow driven by stochastic force and buoyancy force due to Boussinesq approximation. On the left: Diagram of the stochastic flow in an unit square. The gravity is downward parallel to y -axis and periodic boundary conditions are assumed on both x and y directions. On the right, the initial temperature field is plotted, while the initial vorticity is uniformly zero.	140
6.2	STD of vorticity and temperature fields at time $t = 1.0$. Left column by DyBO and right column by gPC	148
6.3	Vorticity spatial modes at time $t = 1.0$. Left column by DyBO and right column by gPC	149
6.4	Temperature spatial modes at time $t = 1.0$. Left column by DyBO and right column by gPC	150
6.5	\mathbb{L}^2 relative errors of vorticity and temperature STDs as functions of time. The evolutions of the numbers of vorticity and temperature mode pairs are also given in the last figure.	151
6.6	Illustration of spatial domain decomposition	152
7.1	Sparse grid vs. tensor grid	158

List of Tables

1.1	Illustration of the exponential growth of the number of polynomial chaos basis functions, N_P .	8
3.1	Relative errors of statistical quantities computed by DyBO, Adaptive-DyBO, and gPC at time $t = 1.0$	100
4.1	Storage complexity comparison between gPC and DyBO-gPC methods	104
4.2	The computational time of some typical terms on the right-hand side of the DyBO-gPC formulation	106
4.3	The computational time of terms on the right-hand sides in the gPC formulation for second-order nonlinear PDE driven by stochastic forces	108
4.4	The computational costs of matrix-tensor \mathbf{A} and $\mathfrak{T}(\mathbf{H})$ products	109
4.5	The computational costs of terms on the right-hand sides in the DyBO-gPC formulation for second-order nonlinear PDE driven by stochastic forces	110

- 4.6 Comparison of wall times of a single RK step of gPC and DyBO-gPC systems. Depending on whether the sparsity of tensor $\mathfrak{T}^{(H)}$ is explored or not, the wall times are reported in the second and third columns for gPC method, respectively. The wall times of DyBO-gPC are reported in columns 4–7 for $m = 4, 8, 12, 16$. The exponents α and β in eqn. (4.9) are estimated by linear regression. The last column in red uses wall times corresponding to $m = 8, 12, 16$ to compute the exponent α , while the second to last column in gray uses all four values of m to estimate the exponent β . The values of β are reported in the last two rows. 116
- 6.1 Speedups by proposed parallelization strategy for different spatial grids and mode number vectors. Wall times of a single RK time step for different parameters are given in the second, fourth, sixth, and eighth columns. All times are in seconds. . . . 154
- 7.1 Relative errors of statistical quantities computed by DyBO-gSC and gSC at time $t = 1.0162$

List of Algorithms

2.1 Strategy in Y -Stage	51
2.2 Strategy in U -Stage	55
2.3 Removing Mode Pairs	65
2.4 Adding Mode Pairs	70
2.5 Overall DyBO-gPC Algorithm	72

Chapter 1

Introduction

Uncertainty arises in many complex real-world problems of physical and engineering interests, such as wave, heat, and pollution propagation through random media [75], and flow driven by stochastic forces [109, 85, 73, 108, 55, 53]. Additional examples can be found extensively in other branches of science and engineering, such as geosciences [90, 32], statistical mechanics, meteorology [28], biology [11], finance [42, 43], and social science [64]. Meanwhile, Ordinary Differential Equations (ODEs) and Partial Differential Equations (PDEs) are often, if not all the time, adopted to describe and represent physical, engineering, economic or even social models. Therefore, Stochastic Ordinary Differential Equations (SODEs) and Stochastic Partial Differential Equations (SPDEs), which contain randomnesses in terms of random variables or stochastic processes, are natural extensions of ODEs and PDEs to investigate Uncertainty Quantification (UQ).

Numerical simulations of PDEs by themselves can be very challenging, e.g., direct numerical simulation of 3D Navier-Stokes equations [82, 51, 58], and flows in porous medium with high contrast [8, 35, 36, 25]. With the introduction of random variables and/or stochastic processes, numerical simulations are greatly complicated. In this chapter, we first briefly review SPDEs and existing numerical methods to put our discussions in appropriate settings. After introducing Karhunen-Loeve expansions, we point out that many stochastic problems have an inherent low-dimensional structure which has not been fully taken advantage of in existing numerical methods.

1.1 Uncertainty Quantification and Stochastic Partial Differential Equations

There are a number of different sources of uncertainty in SPDEs. Below we will discuss a few of them which have been used widely in the literature [62, 85, 27].

Parametric Uncertainty which comes from some model parameters whose exact values are infeasible to obtain in practice. A non-trivial example comes from predictions of flow and transport in natural porous media such as water aquifer and oil reservoirs [75], which essentially requires numerical simulations of an elliptic PDE for the steady state flow u ,

$$-\frac{\partial}{\partial x_p} \left(a_{pq}(x, \omega) \frac{\partial}{\partial x_q} u(x, \omega) \right) = f(x),$$

where the permeability field $a(x, \omega)$ contains random model parameters in the sense that their exact values are infeasible to be obtained in practice primarily due to low resolution of seismic data.

Model deficiency/inadequacy which comes from discrepancy between mathematical models and physical processes. Such discrepancy may be due to the lack of full understanding of true physical processes, or the lack of sophisticated mathematical tools available to describe adequately the underlying complex physical processes, or simplifications of complicated mathematical models for further theoretical or numerical analysis. A concrete example, which we will discuss extensively in the thesis, is randomly forced Burgers or Navier-Stokes equations. The large-scale structures of flows are primarily governed by deterministic PDEs and random forcing $f(x, t, \omega)$ in terms of random variables or Brownian motions are introduced to model unresolved small-scale effects from turbulence.

Algorithmic uncertainty which comes from numerical round-off errors due to finite digits

of modern computer architectures and numerical approximation errors arising from discretizations, numerical integrations, and truncation of infinite summations. Generally, such numerical algorithms solve exactly some modified ODEs/PDEs, which are similar to the original ODEs/PDEs with extra terms. However, the exact forms of such terms are only known in very limited applications, so it is convenient to introduce noise terms to replace them and to study the effect of numerical errors on complicated systems.

Analytical solutions of SODEs/SPDEs are only available in very special cases. For most of practical problems, numerical schemes are used to approximate stochastic solutions and quantify uncertainty. However, due to the complex nature of SODEs and especially SPDEs, numerical simulations pose a great challenge to the applied mathematical community. In this thesis, we are primarily concerned with SPDEs and their efficient numerical schemes. Methods developed in this thesis are still applicable to certain types of SODEs.

Throughout the thesis, we consider the following time-evolutionary/time-dependent stochastic partial differential equation

$$\frac{\partial u}{\partial t}(x, t, \omega) = \mathcal{L}u(x, t, \omega), \quad x \in \mathcal{D} \subset \mathbb{R}^d, t \in [0, T], \omega \in \Omega, \quad (1.1a)$$

$$u(x, 0, \omega) = u_0(x, \omega), \quad (1.1b)$$

$$\mathcal{B}(u(x, t, \omega), \frac{\partial u}{\partial x}(x, t, \omega)) = 0, \quad x \in \partial\mathcal{D}, \quad (1.1c)$$

where \mathcal{L} is a differential operator and may contain random coefficients and/or stochastic forces. The randomnesses may also enter the system through initial condition u_0 and/or boundary condition \mathcal{B} .

1.2 Existing Numerical Methods

Before embracing the data-driven philosophy underlying in this thesis, it is worth reviewing briefly existing numerical methods for SPDEs along with their applicable regime and limitations.

1.2.1 Non-Sampling Methods

Perturbation methods (PM). When both input and output uncertainties are small compared to mean values, perturbation methods are shown to be effective in many engineering problems [75]. These methods start with expanding stochastic solutions via Taylor expansion and result in a system of deterministic equations by truncating after certain order. Typically, at most second-order expansions are used in practice as the system of equations become very cumbersome if higher-order terms are included. One limitation of perturbation methods is that the magnitude of the uncertainty must be small. Moreover, they do not reveal faithfully probability distributions of stochastic solutions.

Probability Density Function Methods (PDF) and Moment Equation (ME) Methods. When statistical information is of main interests, instead of tackling directly SPDEs, effective equations for moments and PDFs can be derived and solved numerically, such as Fokker-Planck equations [68]. For nonlinear problems, such methods suffer from closure issues, i.e., equations of lower moments depend on higher-order moments or certain terms in PDF equations take complicated integral forms and are hardly computable. Such difficulty can be alleviated by using some heuristic arguments to achieve closure, but it also introduces modeling errors which are hard to control.

Concentration of Measure (CM). Modern engineering or biological systems tend to have more than hundreds of inputs. Examples of such kind include neural networks, the Internet, and smart grids of next-generation electricity supply networks and so on. Solutions or response functions of such systems are actually functions over high-dimensional input spaces where oscillation on a

single input can only cause small oscillation on outputs, i.e., response functions are well centered around their expectations. Such a phenomenon is called concentration of measure. Study of this phenomenon can be tracked back to Levy and has been studied systematically by Milman, Ledoux [74], Talagand [105], etc. A bunch of powerful concentration-of-measure inequalities, such as McDiarmid's inequality, have been developed in the past years, which can be used to provide tight bounds on response fluctuations over means. Owhadi and others have demonstrated the success of this method in the context of parameter sensitivity and certification [2, 3, 107, 104, 76], i.e., showing the failure probability of complex system is within a certain tolerance, and have proposed the Optimal Uncertainty Quantification (OUQ) framework [103, 92] to achieve optimal bounds on uncertainty for a given set of assumptions and information. However, these methods do not reveal the KL expansions of the stochastic solutions, which is important for understanding physical systems.

1.2.2 Non-Intrusive Sampling Methods

Modern software systems for numerical simulations of deterministic models of engineering or physical systems can be very complex and may contain more than thousands of lines of codes and its validations against experimental results may take years. For example, sophisticated Computational Fluid Dynamic (CFD) code can easily exceed ten thousands of line of codes [89]. Changes in a small segment of code may introduce unforeseeable bugs and require another round of validations. Therefore, sampling methods are preferred in some of industrial applications for their non-intrusive nature, i.e., requiring no direct modifications on legacy codes.

Monte Carlo Method (MC). Because of stability and non-intrusive nature of MC, it remains a popular numerical method for SPDEs especially when the dimensionality of input uncertainty, p , is very large. According to the law of large numbers, the convergence rate of MC is merely

$O\left(M^{-\frac{1}{2}}\right)$, where M is the number of realizations, so MC methods are “famous” for their slow convergence.

Quasi-Monte Carlo Method (qMC). The slow convergence of MC is partially due to the low uniformity or high discrepancy of pseudo-random number sequences generated by computer algorithms [60, 59, 83, 17]. qMC methods [45, 84] use instead a deterministic sequence of numbers of low discrepancy to achieve almost $O\left(M^{-1}\right)$ for small to moderate dimensionality of input uncertainty. However, such convergence rate is still considered to be slow for industrial-scale applications where numerical simulation of a single realization may take hours or even days. Furthermore, the above convergence rate degrades to $O\left(M^{-\frac{1}{2}}\right)$ for large p .

Generalized Stochastic Collocation Method (gSC). gSC [78, 87, 33, 7, 106, 116, 7] methods explore the smoothness of stochastic solutions with respect to random variables, and use instead a deterministic sequence of sampling points $\{\omega_i\}_{i=1}^M$ resulting from tensor products of one-dimensional quadrature points. The exact locations of such points and weights w_i associated with them depend on underlying probability distributions. Because of tensor products, the number of sampling points increases exponentially fast as the number of random variables increases, which is also known as *curse of dimensionality*. Sparse grid [18] is introduced and further exploits smoothness to reduce the number of sampling points to some extent.

The sampling methods are after an ensemble of solutions $\{u(x, \omega_i)\}_{i=1}^M$ along with weights $\{w_i\}_{i=1}^M$ associated with each realization. The above mentioned three non-intrusive methods only differ in the selection of sampling points and associated weights.

Multilevel Monte Carlo Methods (MLMC). Recently, there is a surge of interests in Multilevel Monte Carlo methods, which was pioneered by Giles [43, 42] for SODEs in areas of stock pricing models and later extended to stochastic elliptic problems by Cliffe et al. [26] and Barth et al. [8], and Navier-Stokes equations with random initial data by Mishra et al. [80]. MLMC explores two

aspects of numerical simulations of SODEs and SPDEs to achieve accelerations. One is that, besides sampling errors, numerical errors due to spatial and temporal discretizations also contribute to final results and thus the effective number of sampling points are constrained by grid resolutions. The other, which may be more important, is that the variances of solution differences on two adjacent discretization grids in a geometrical grid hierarchy may decay exponentially fast and the number of sampling points on each level decay exponentially. An intrinsic limitation associated with MLMC methods comes from that the fore-mentioned assumption about variance decay is true only when the grid size reaches some asymptotic regime. In this case, each realization on the “coarse level” is already quite expensive and the computational saving is not as much as one would like to get.

1.2.3 Intrusive Methods

Generalized Polynomial Chaos Methods (gPC). While non-intrusive sampling methods represent the solution of SPDEs as an ensemble of realizations, another way to represent stochastic solutions is to employ truncated expansion on a set of polynomial chaos basis and drive a system of coupled deterministic PDEs. The stochastic solution generally takes the form of $u(x, \xi) = \sum_{\alpha \in \mathfrak{J}} u_{\alpha}(x) \mathbf{H}_{\alpha}(\xi)$ where \mathfrak{J} is some multi-index set and will be explained in detail in Sec. 2.3.2.1. The original idea can be traced back to Wiener [113] in 1938. At the time, the method did not receive much attention from the applied mathematics community until Cameron and Martin [19] developed an explicit and intuitive formulation for Wiener Chaos Expansion(WCE) in their elegant work in 1947. In the past decade or so, we have witnessed huge interests and research efforts on developing efficient methods based on generalized Polynomial Chaos Expansions (gPCE), see, e.g., Ghanem and Spanos [41], Xiu [115], Xiu and Shen [119], Xiu and Karniadakis [118, 117], Wan and Karniadakis [111, 110], Hou et al. [55, 77], and Matthies and Keese [79], which demonstrate its effectiveness and potential applicability to broad classes of SPDE problems. However, gPC meth-

ods also suffer from the curse of dimensionality. For example, if a SPDE involve p independent random variables and only polynomials of total order at most P are chosen, then the total number of polynomial chaos basis functions is

$$N_P = \binom{P+p}{p} = \frac{(P+p)!}{P!p!},$$

which increases exponentially and can easily exceed available computational resource even for small p and P as demonstrated in Table 1.1.

p	P	N_P
4	3	35
4	5	126
6	3	84
6	5	462
8	3	162
8	5	1287
10	5	3003

Table 1.1: Illustration of the exponential growth of the number of polynomial chaos basis functions, N_P .

The situation gets worse for spatially three-dimensional SPDEs since each gPC coefficient is a function over a three-dimensional domain. Updating or solving a system of deterministic PDEs with hundreds of such components may not be a realistic goal in a reasonable time frame. Based partially on heuristic arguments, several strategies to select adaptively the most important polynomial chaos basis functions have been proposed and shown to be effective in some numerical tests. The situation gets even worse for time-dependent SPDEs since the set of most important polynomial chaos basis functions may change in time and is hard to predict. Other variants of gPC methods have also been developed, such as Wiener-Haar Chaos Expansion [71, 72, 70, 69], using wavelet bases, and multi-element gPC [110, 33, 111] based on domain decomposition of random spaces.

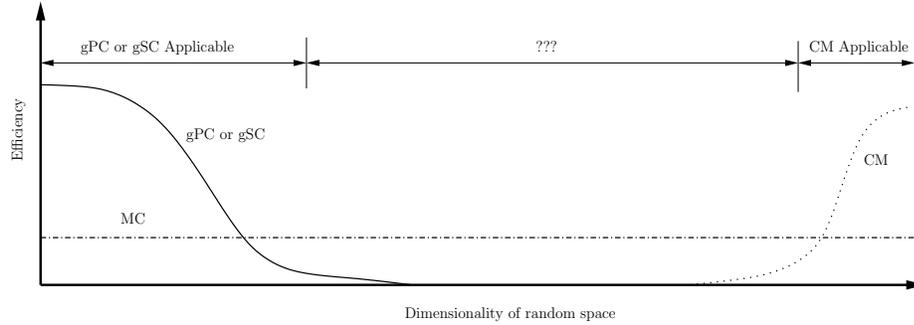


Figure 1.1: Conceptual illustration of efficiency of gPC, gSC, MC, and CM with respect to dimensionality of random space

1.2.4 Post-Processing

The efficiency of MC, gPC, gSC, and CM with increasing number of random variables are illustrated in Fig. 1.1. Both gPC and gSC suffer from the curse of dimensionality, so their efficiencies degrade very fast as the number of random variables increases. On the other hand, concentration-of-measure inequalities, although powerful, inherently assume the presence of a large number of uncertainty sources and therefore are only effective at the high end of random dimensions. The convergence rate of MC methods depends largely on the number of realizations, so its efficiency remains almost constant, regardless of the dimensionality of the space of random variables. It is clear that other structures of stochastic solutions of SPDEs should be further explored to design new and more efficient numerical algorithms.

One of potential candidates reveals itself after we pay a close examination to the post-processing of intrusive and non-intrusive methods, an aspect which receives a little attention in the literature. The numerical simulation process of SPDEs never stops at just getting a set of realizations $\{u(x, \omega_i)\}_{i=1}^M$ for non-intrusive methods or a set of gPC coefficients $\{u_\alpha(x)\}_{\alpha \in \mathfrak{J}}$ for intrusive methods. To obtain useful information and grasp physical insights of the system involving randomness, some post-processing steps of the large number of realizations or gPC coefficients are necessary, for example, to obtain expectation and variances of stochastic solutions.

Dynamically Bi-Orthogonal Method (DyBO)

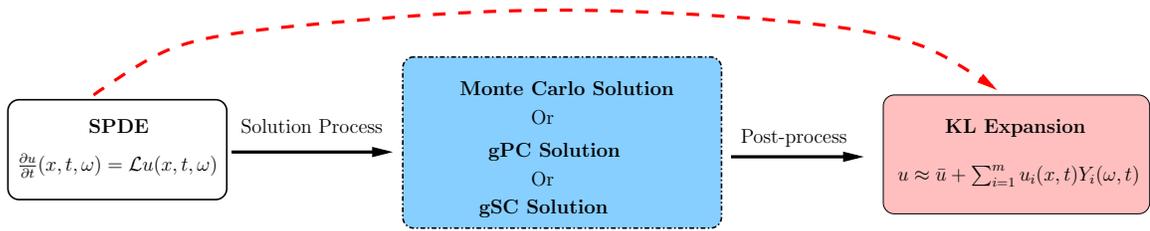


Figure 1.2: The numerical simulation process of SPDEs

The Karhunen-Loeve Expansion (KLE) has received increasing attention in many fields, such as statistics [1], image processing [65], and UQ [100, 22]. Among many reasons for its popularity, one is that KLE reveals not only mean and variance but also the most important spatial and stochastic modes of the stochastic solution, allowing sparse representation of stochastic solutions and extractions of physical understandings. However, the computation of KLE is not trivial at all. For example, the computation of KLE from an ensemble of realizations generally involves solving a large-scale SVD problem. Certain methods have been developed to make such computation more efficient, e.g., randomized algorithms [86]. If we look at the entire numerical simulation process of SPDEs in Fig. 1.2, we find that a large number of realizations or gPC coefficients are *first* obtained after a lot of computational resources and time are spent, and then “*condensed*” to KLE. Therefore, a question arises naturally,

Can we skip the step of solving a large number of realizations or gPC coefficients and target directly KL expansions of SPDE solutions?

1.3 Exploring Sparsity

Sparse structures prevail in many physical and engineering problems in one form or another and have been explored in several methods, such as signal reconstruction [20], hierarchical matrix [48, 47], fast multipole methods [100], and adaptive data analysis [56]. In this thesis, we propose a

method for the numerical simulation of SPDEs, which fully explores the sparsity in the sense of KL expansions. To make the following discussion more precise, we first review briefly KL expansions in this section and then outline the philosophy of proposed data-driven algorithms for SPDEs. Before that, the following two remarks are worth mentioning.

Remark 1.1. The sparsity which we explore here is not the usual *entry-wise sparsity*, i.e., only a few non-zero entries or coefficients, but *data-sparsity*, i.e., few data are required to provide an accurate description of the stochastic solutions. Such a difference has been pointed out by Hackbusch [47], and Hackbusch and Khoromskij [48], in the context of matrices. For numerical SPDEs, the former sparsity has been explored in elliptic PDEs with stochastic coefficients by Todor and Schwab [106], Bieri and Schwab [14], Schwab and Todor [99], and Doostan and Owhadi [29]. However, the set of sparse coefficients are unknown *a priori*, which is necessary for effective computation reduction. A heuristic algorithm based on purely-stochastic problems is suggested by Schwab to select such a set prior to computations. Doostan formulates l_0 - and l_1 - optimization problems and uses Orthogonal Marching Pursuit (OMP) algorithm and Spectral Projected Gradient (SPGL) algorithm to demonstrate its effectiveness.

Remark 1.2. The combined sparsity, i.e., data sparsity and entry-wise sparsity, has been explored by Chandrasekaran et al. [24, 23] in the context of matrix decomposition. However, it is not clear at the moment how such combined structures should be explored in numerical SPDEs.

1.3.1 Karhunen-Loeve Expansions

Consider a probability space $(\Omega, \mathcal{F}, \mathbb{P})$, where Ω is called the event space and is equipped with a σ -algebra \mathcal{F} and a probability measure \mathbb{P} . Suppose a stochastic process $u(x, \omega)$ defined on a compact

spatial domain $\mathcal{D} \subset \mathbb{R}^d$ is a second-order stochastic process, i.e.,

$$u \in \mathbb{L}^2(\mathcal{D} \times \Omega) = \left\{ v(x, \omega) \mid \int_{\mathcal{D}} \int_{\Omega} v(x, \omega)^2 \mathbb{P}(d\omega) dx \right\}.$$

Sometimes, we also write $\mathbb{L}^2(\mathcal{D} \times \Omega) = \mathbb{L}^2(\Omega \rightarrow \mathbb{L}^2(\mathcal{D}))$ to emphasize that u is a function-valued random variable taking values in $\mathbb{L}^2(\mathcal{D})$. Here $\mathbb{L}^2(\mathcal{D})$ is the Hilbert space equipped with the usual inner product

$$\langle f(x), g(x) \rangle_{\mathbb{L}^2(\mathcal{D})} = \int_{\mathcal{D}} f(x)g(x) dx,$$

which induces the usual \mathbb{L}^2 -norm

$$\|f(x)\|_{\mathbb{L}^2(\mathcal{D})}^2 = \langle f(x), f(x) \rangle$$

on $\mathbb{L}^2(\mathcal{D})$ and a norm

$$\|v(x, \omega)\|_{\mathbb{L}^2(\mathcal{D} \times \Omega)}^2 = \int_{\mathcal{D}} \int_{\Omega} v(x, \omega)^2 \mathbb{P}(d\omega) dx = \mathbb{E} \left[\|v(x, \omega)\|_{\mathbb{L}^2(\mathcal{D})}^2 \right]$$

on $\mathbb{L}^2(\mathcal{D} \times \Omega)$. When no ambiguity arises, we may drop subscripts. Denote $\bar{u}(x) = \mathbb{E}[u(x, \omega)]$, the expectation of u , and the covariance function of stochastic process u is defined as

$$\text{Cov}_u(x, y) = \mathbb{E}[(u(x, \omega) - \bar{u}(x))(u(y, \omega) - \bar{u}(y))], \quad \forall x, y \in \mathcal{D}.$$

When no ambiguity arises, we simply write $\text{Cov}(x, y)$. The covariance function naturally defines a linear operator on $\mathbb{L}^2(\mathcal{D})$, i.e.,

$$g(x) = \int_{\mathcal{D}} \text{Cov}(x, y)f(y) dy : \mathbb{L}^2(\mathcal{D}) \rightarrow \mathbb{L}^2(\mathcal{D}).$$

It is easy to show that this linear operator is compact, self-adjoint and positive definite and, by Mercer's theorem, admits a spectral decomposition [41] on $\mathbb{L}^2(\mathcal{D})$,

$$\text{Cov}(x, y) = \sum_{i=1}^{\infty} \lambda_i f_i(x) f_i(y),$$

where λ_i 's and f_i 's are the eigenvalues and eigenfunctions of the covariance kernel and $\{f_i(x)\}_i^\infty \subset \mathbb{L}^2(\mathcal{D})$ form a complete orthonormal basis, i.e., $\langle f_i, f_j \rangle = \delta_{ij}$. The stochastic process $u(x, \omega)$ can be expanded in a Fourier-type series as

$$u(x, \omega) = \bar{u}(x) + \sum_{i=1}^{\infty} \sqrt{\lambda_i} f_i(x) Y_i(\omega), \quad (1.2)$$

where random variables $Y_i(\omega)$'s are zero mean and uncorrelated, i.e., $\mathbb{E}[Y_i] = 0$, $\mathbb{E}[Y_i Y_j] = \delta_{ij}$, and are given by

$$Y_i(\omega) = \frac{1}{\sqrt{\lambda_i}} \int \tilde{u}(x, \omega) f_i(x) dx, \quad \text{for } i = 1, 2, 3, \dots,$$

with $\tilde{u}(x, \omega) = u(x, \omega) - \bar{u}(x)$. $f_i(x)$'s or $\sqrt{\lambda_i} f_i(x)$'s are referred to as spatial modes and Y_i 's are referred to as stochastic modes in this thesis.

The KL expansion (1.2) is named after Kari Karhunen and Michel Loeve who derived it independently around 1947 [41]. Generally, the eigenvalues λ_i 's are sorted in descending order and cluster at zero, whose decay rate depends only on the regularity of the covariance kernel Cov_u . It has been shown that algebraic decay rate, i.e., $\lambda_i = O(i^{-\gamma})$, is achieved asymptotically if the covariance kernel is of finite Sobolev regularity, and exponential decay rate, i.e., $\lambda_i = O(e^{-\gamma i})$, if the covariance kernel is piecewisely analytical [100]. In either case, an m -term truncated KLE converges to the original stochastic process $u(x, \omega)$ in $\|\cdot\|_{\mathbb{L}^2(\mathcal{D} \times \Omega)}$ as $m \rightarrow \infty$. This can be seen by

computing the truncation error

$$\|\epsilon_m\|_{\mathbb{L}^2(\mathcal{D} \times \Omega)}^2 = \left\| \sum_{i=m+1}^{\infty} \sqrt{\lambda_i} f_i(x) Y_i(\omega) \right\|_{\mathbb{L}^2(\mathcal{D} \times \Omega)}^2 = \sum_{i=m+1}^{\infty} \lambda_i \rightarrow 0, \quad \text{as } m \rightarrow \infty,$$

where we have used the bi-orthogonality of KLE.

The most important property characterizing KLE is the so-called error-minimizing property, which can be summarized in the following theorem,

Theorem 1.1 (Error Minimizing Property of KLE). *Among all complete orthonormal bases of $\mathbb{L}^2(\mathcal{D})$, KLE of second-order stochastic process $u(x, \omega) \in \mathbb{L}^2(\mathcal{D} \times \Omega)$ minimizes the m -term truncation errors for $m = 1, 2, 3, \dots$, i.e.,*

$$\{f_i(x)\}_{i=1}^{\infty} = \arg \min_{\langle h_i, h_j \rangle = \delta_{ij}} \|\epsilon_m(x, \omega)\|_{\mathbb{L}^2(\mathcal{D} \times \Omega)}, \quad (1.3)$$

where the m -term truncation error

$$\epsilon_m(x, \omega) = u(x, \omega) - \bar{u}(x) - \sum_{i=1}^m \sqrt{\theta_i} h_i(x) Z_i(\omega),$$

and $Z_i(\omega) = \frac{1}{\sqrt{\theta_i}} \int \tilde{u}(x, \omega) h_i(x) dx$.

Proof. See [41] or Sec. 5.2. □

While the error-minimizing property implies KLE is the sparsest expansion of a second-order stochastic process in the energy norm $\|\cdot\|_{\mathbb{L}^2(\mathcal{D} \times \Omega)}$, bi-orthogonality of KLE characterizes it completely from another aspect, i.e.,

Theorem 1.2 (Bi-Orthogonality of KLE). *The second-order stochastic process $u(x, \omega) \in \mathbb{L}^2(\mathcal{D} \times \Omega)$*

admits a unique bi-orthogonal expansion (1.2) where

$$\langle f_i(x), f_j(x) \rangle = \delta_{ij},$$

$$\mathbb{E}[Y_i(\omega)Y_j(\omega)] = \delta_{ij}.$$

Proof. See [41] or Sec. 5.2. □

1.3.2 Overview of Dynamically Bi-Orthogonal Methods

Many physical and engineering problems involving uncertainty enjoy certain low-dimensional structures, e.g., in the sense of KL expansions, which in turn indicates the existence of reduced-order models and better formulations for efficient numerical simulations. Clearly, such basis do not come for free and should be constructed by using available information, i.e., data-driven algorithms. In this thesis, we consider the numerical simulation of SPDEs and will derive an equivalent system whose solution closely follows the KL expansion of the stochastic solution. Theorem 1.1 implies that the KL expansion is the most compact representation of a stochastic process in the \mathbb{L}^2 sense. Therefore, such a system offers great computational benefits compared to other popular generic methods, such as traditional MC, gSC, and gPC.

Such low-dimensional structures of SPDE solutions have been explored *partially* in some methods, such as classic Proper Orthogonal Decomposition (POD) methods [95, 12], certified reduced basis (RB) methods [68], and Dynamically Orthogonal (DO) methods [97, 96]. Most of these methods try to construct a spatial basis, either offline (e.g., POD or RB) or on-the-fly (e.g., DO), to replace the standard spatial discretizations. Thus, they reduce the number of equations to solve and improve computational efficiency. In POD methods, such spatial bases are constructed from Singular Value Decomposition (SVD) of a cluster of snapshot solutions taken at different times. In RB

methods, the spatial bases are constructed in a similar way from a cluster of solutions corresponding to different parameters/realizations. The selection of the set of parameter values are guided by some posterior error estimates. While both POD and RB methods construct a spatial basis in an offline manner, the DO method tracks such a spatial basis on-the-fly by imposing DO conditions. Although the DO method tries to achieve a similar objective as ours, it requires computation of the covariance function frequently in time to take advantage of the low-dimensional structure, which could be computationally expensive.

All these methods emphasize spatial bases and ignore stochastic bases, so they only partially explore the low-dimensional structures provided by KL expansions. The method proposed in this thesis essentially tracks the KL expansions of SPDE solutions by deriving and solving a system of equations for the spatial and stochastic modes in KL expansions. More specifically, our method dynamically constructs the optimal basis on-the-fly by exploiting the sparse structure provided by the KLE of the stochastic solution without the need of computing the covariance functions. This unique feature is what distinguishes our method from other methods.

We consider the following KLE-type expansion of the stochastic solution at time t of the time-dependent SPDE (1.1),

$$u(x, t, \omega) \approx \bar{u}(x, t) + \sum_{i=1}^m u_i(x, t) Y_i(\omega, t) = \bar{u}(x, t) + \mathbf{U}(x, t) \mathbf{Y}^T(\omega, t),$$

where $\mathbf{U} = (u_1, u_2, \dots, u_m)$ and $\mathbf{Y} = (Y_1, Y_2, \dots, Y_m)$. Later, we will see how vector notation significantly simplifies the formulations.

Unlike POD or gPC methods, we allow both spatial basis \mathbf{U} and stochastic basis \mathbf{Y} to change dynamically in the above representation. Clearly, without additional requirements on \mathbf{U} and \mathbf{Y} , the decomposition is not unique and we cannot derive a closed system of evolutionary equations for

\bar{u} , \mathbf{U} and \mathbf{Y} . In terms of computational costs, the natural requirement is to minimize the number of terms, m , for a given error threshold. According to Theorem 1.2 and 1.1, this is equivalent to preserving the bi-orthogonality of spatial modes \mathbf{U} and stochastic modes \mathbf{Y} all the time, which turns out to be sufficient for deriving a closed system of equations for \bar{u} , \mathbf{U} and \mathbf{Y} (see Sec. 2.1 for details).

By enforcing the bi-orthogonal condition, we obtain

$$\begin{aligned}\frac{\partial \bar{u}}{\partial t} &= \mathbb{E}[\mathcal{L}u], \\ \frac{\partial \mathbf{U}}{\partial t} &= -\mathbf{U}\mathbf{D}^T + \mathbb{E}[\tilde{\mathcal{L}}u\mathbf{Y}], \\ \frac{d\mathbf{Y}}{dt} &= -\mathbf{Y}\mathbf{C}^T + \langle \tilde{\mathcal{L}}u, \mathbf{U} \rangle \mathbf{\Lambda}_{\mathbf{U}}^{-1},\end{aligned}$$

where $\tilde{\mathcal{L}}u = \mathcal{L}u - \mathbb{E}[\mathcal{L}u]$ and m -by- m matrices \mathbf{C} and \mathbf{D} can be solved from a linear system involving only matrix $G_*(u, \mathbf{U}, \mathbf{Y}) = \mathbf{\Lambda}_{\mathbf{U}}^{-1} \langle \mathbf{U}^T, \mathbb{E}[\tilde{\mathcal{L}}u\mathbf{Y}] \rangle \in \mathbb{R}^{m \times m}$. The first two equations are time-dependent *deterministic* PDEs and are coupled to the third equation, a system of stochastic ODEs. Because the method we have developed preserves bi-orthogonality of spatial and stochastic modes on the fly, it is called the Dynamically Bi-Orthogonal (DyBO) method.

1.4 Summary of the Thesis

This thesis focuses on the innovative DyBO methods and their efficient numerical implementations for a class of time-dependent SPDEs or a system of time-dependent SPDEs, whose solutions enjoy low-dimensional structures in the sense of KL expansions. We have given careful considerations to various aspects of the proposed methods, such as formulations, analysis, computational complexities, and numerical implementations. Several challenging problems involving randomnesses have been carefully chosen and used to demonstrate the applicability and the efficiency of DyBO meth-

ods, and to verify the theoretical results. The contributions of this thesis is summarized in the next section. A road map is also provided in Sec. 1.4.2 for the ease of readers.

1.4.1 Summary of Main Contributions

Direct tracking of KL expansions. In the popular numerical methods for SPDEs, such as MC, qMC, gPC, and gSC, additional post-processing steps are necessary to get the KL expansions of stochastic solutions. Without any additional post-processing steps, DyBO methods explore the inherent low-dimensional structures and give *directly* the stochastic solutions in bi-orthogonal form that essentially tracks the KL expansion of the stochastic solutions. The ability to preserve bi-orthogonal forms in DyBO methods has been proved rigorously and verified numerically in various challenging problems, such as stochastic Burgers equations, and stochastic flows in 2D unit square.

Reduced-order models and computation reductions. An important benefit associated with DyBO formulations is the significant savings of computational costs both in memory consumptions and computational times. Detailed complexity analysis has been conducted for DyBO-gPC methods for certain classes of time-dependent SPDEs and verified numerically in Chapter 4.2. In practice, we have observed speedup up to 200 times compared to gPC methods in 2D stochastic flow problems.

Adoption of vector/matrix/tensor notations in formulations. With the introduction of multiple spatial modes and stochastic modes, the final formulations of DyBO methods can be very messy and error-prone in numerical implementations. To avoid such technical difficulties, we have carefully designed the formulations and extensively adopted vector/matrix/tensor notations. For examples, we write the spatial modes as a row vector of functions of space and time, $\mathbf{U}(x, t) = (u_1(x, t), u_2(x, t), \dots, u_m(x, t))$. In this way, the DyBO formations are presented in very concise forms and also reveal physical insights and interesting structures for some numerical examples considered in this thesis. For examples, see the formulation (4.17) or (4.18) for 2D stochastic flows.

Adoptions of vector/matrix/tensor notations also enable programming to proceed in an intuitive way, especially for object-oriented programming languages.

Adaptivity and other implementation issues. Based on error analysis, sophisticated strategies have been proposed in this thesis to adaptively and dynamically add and remove spatial and stochastic mode pairs to achieve both accuracy and efficiency. The effectiveness of such strategies have been confirmed in numerical examples in this thesis. See Sec. 3.3.3.2, Sec. 4.2.2, and Sec. 6.3, for example. Other difficulties in numerical implementations, e.g., eigenvalue crossing, are also considered and strategies are proposed.

Generalizations to a system of SPDEs. Multiple physical fields are generally involved in practical applications along with randomneses, which result in a coupled system of time-dependent SPDEs. For example, the standard three-dimensional incompressible Navier-Stokes equations involve four physical fields, velocity components along x -, y -, z -axis and pressure. The DyBO method is also generalized for a system of SPDEs and demonstrated in 2D stochastic flow driven by both stochastic external forces and buoyancy forces in Chapter 6. Such generalization, certainly interesting by itself, potentially provides a way to tackle problems involving both multiscale phenomena and randomneses, which will be explained in detail under Future Work in Chapter 8.

Parallelization Nowadays, parallelization almost becomes an indispensable ingredient for successful numerical simulations of PDEs in industrial applications [80, 102, 44]. Although the proposed DyBO methods have explored the inherent sparsity within SPDEs themselves, further computational reductions by parallelization are still necessary, especially for spatially three-dimensional SPDEs with multiple physical components. Based on the computational complexity analysis in Sec. 4.1.2.2, we propose a simple, yet powerful, parallelization strategy based on domain decomposition for stochastic flows in Sec. 6.4. The effectiveness of the parallelization strategy is confirmed on multi-core computing nodes at the Caltech Center for Advanced Computing Research (CACR)

and about 5 times speedup is observed with 8 computing processes.

The proposed method still has some limitations. For example, the decay rate of eigenvalues in the KL expansion of stochastic solutions can be very slow in stochastic multiscale problems with small correlations [100]. For this type of problem, our method may lead to expensive computational costs. Potential fixes along with other issues are discussed under Future Work in Chapter. 8.

1.4.2 Roadmap

The thesis is organized as follows.

- Chapter 2 focuses on the theoretical aspects of the DyBO method, such as bi-orthogonality preserving, and error analysis, along with numerical algorithms and other numerical implementation details.
- Chapter 3 and Chapter 4 present various numerical examples ranging from spatially-one-dimensional stochastic Burgers' equation to spatially two-dimensional stochastic Navier-Stokes equation. Each numerical example emphasizes different aspects of our analysis in Chapter 2. Computational complexities in terms of memory and computational time are carefully analyzed in Chapter 4.
- Chapter 5, Chapter 6 and Chapter 7 include generalizations to other stochastic processes, generalizations to a system of time-dependent SPDEs and DyBO formulations on sparse grids. Parallelization is also considered in stochastic flow problems to demonstrate the applicability of DyBO methods for industrial-scale problems.
- Chapter 8 summarizes the thesis and points out several future directions.

Chapter 2

Dynamically Bi-Orthogonal (DyBO) Method

For most time-dependent types of physical and engineering problems involving randomness, we can assume the stochastic process or SPDE solution $u(x, t, \omega)$ given by the system (1.1) is a second-order stochastic process at each fixed time $t > 0$, i.e., $u(\cdot, t, \cdot) \in \mathbb{L}^2(\mathcal{D} \times \Omega)$. Therefore, u admits the KL expansion (1.2),

$$u(x, t, \omega) = \bar{u}(x, t) + \sum_{i=1}^{\infty} \sqrt{\lambda_i(t)} \tilde{u}_i(x, t) Y_i(\omega, t), \quad (2.1)$$

where $\bar{u}(x, t) = \mathbb{E}[u(x, t, \omega)]$ is the expectation of u and Y_i 's are zero-mean random variables, i.e., $\mathbb{E}[Y_i] \equiv 0$, given by

$$Y_i(\omega, t) = \frac{1}{\sqrt{\lambda_i(t)}} \int_{\mathcal{D}} \tilde{u}(x, t, \omega) \tilde{u}_i(x, t) dx, \quad \text{for } i = 1, 2, 3, \dots$$

According to Theorem 1.2, both \tilde{u}_i 's and Y_i 's satisfy bi-orthogonality conditions, i.e.,

$$\langle \tilde{u}_i, \tilde{u}_j \rangle = \int_{\mathcal{D}} \tilde{u}_i(x) \tilde{u}_j(x) dx = \delta_{ij},$$

$$\mathbb{E}[Y_i Y_j] = \delta_{ij},$$

and $\{\check{u}_i\}_{i=1}^{\infty}$ and $\{Y_i\}_{i=1}^{\infty}$ are orthogonal basis for $\mathbb{L}^2(\mathcal{D})$ and $\mathbb{L}^2(\Omega)$, respectively.

If the eigenvalue λ_i 's are distinct, *such decomposition is unique up to a sign*. For practical problems, such requirement is almost always true. Thus, we can assimilate the factor $\sqrt{\lambda_i(t)}$ into \check{u}_i and arrive at

$$u(x, t, \omega) = \bar{u}(x, t) + \sum_{i=1}^{\infty} u_i(x, t) Y_i(\omega, t), \quad (2.2)$$

where $u_i = \sqrt{\lambda_i} \check{u}_i$ and $\langle u_i, u_j \rangle = \lambda_i \delta_{ij}$. For the ease of future derivations, we use, extensively throughout this thesis, vector and matrix notations. For example, we group the first m spatial and stochastic modes into *row* vectors, respectively, i.e.,

$$\begin{aligned} \mathbf{U}(x, t) &= (u_1(x, t), u_2(x, t), \dots, u_m(x, t)) = (u_i(x, t))_{i=1,2,\dots,m} \in \mathbb{R}^{1 \times m}, \\ \mathbf{Y}(\omega, t) &= (Y_1(\omega, t), Y_2(\omega, t), \dots, Y_m(\omega, t)) = (Y_i(\omega, t))_{i=1,2,\dots,m} \in \mathbb{R}^{1 \times m}. \end{aligned}$$

When the ranges of indices are clear from the context, we drop them and simply write $\mathbf{U} = (u_i)_i$ and $\mathbf{Y} = (Y_i)_i$. Sometimes, we also write $\mathbf{U} = (u_i)_{1 \times i}$ to emphasize \mathbf{U} is a row vector. Similarly for the rest of the spatial and stochastic modes,

$$\begin{aligned} \tilde{\mathbf{U}}(x, t) &= (u_{m+1}(x, t), u_{m+2}(x, t), \dots) \in \mathbb{R}^{1 \times \infty}, \\ \tilde{\mathbf{Y}}(\omega, t) &= (Y_{m+1}(\omega, t), Y_{m+2}(\omega, t), \dots) \in \mathbb{R}^{1 \times \infty}. \end{aligned}$$

Correspondingly, $\mathbb{E}[\mathbf{Y}^T \mathbf{Y}]$ and $\langle \mathbf{U}^T, \mathbf{U} \rangle$ are m -by- m matrices and the bi-orthogonality condition can be rewritten compactly as

$$\mathbb{E}[\mathbf{Y}^T \mathbf{Y}](t) = (\mathbb{E}[Y_i Y_j])_{ij} = \mathbf{I} \in \mathbb{R}^{m \times m}, \quad (2.3a)$$

$$\langle \mathbf{U}^T, \mathbf{U} \rangle(t) = (\langle u_i, u_j \rangle)_{ij} = \mathbf{\Lambda}_{\mathbf{U}} \in \mathbb{R}^{m \times m}, \quad (2.3b)$$

where $\Lambda_{\mathbf{U}} = \text{diag}(\langle \mathbf{U}^T, \mathbf{U} \rangle) = (\langle u_i, u_j \rangle \delta_{ij})_{ij} \in \mathbb{R}^{m \times m}$. The KL expansion of u at some fixed time t reads

$$u = \bar{u} + \mathbf{U}\mathbf{Y}^T + \tilde{\mathbf{U}}\tilde{\mathbf{Y}}^T. \quad (2.4)$$

As stated in the previous chapter, the class of time-dependent SPDEs we consider in this thesis enjoy a low-dimensional structure in the sense of KLE, or precisely, the eigenvalue spectrum $\{\lambda_i(t)\}_{i=1}^{\infty}$ decays fast enough, allowing for a good approximation with a truncated KL expansion (2.4). Therefore, we have the following approximation of u ,

$$u \approx \bar{u} + \mathbf{U}\mathbf{Y}^T = \bar{u} + \mathbf{Y}^T\mathbf{U}. \quad (2.5)$$

The last equality is due to the fact that both \mathbf{U} and \mathbf{Y} are simply row vectors. This chapter is devoted to the derivation of the reduced-order model of the original system (1.1) stemming from the above truncated KLE (2.5), i.e., a system of equations for \bar{u} , \mathbf{U} and \mathbf{Y} , and efficient numerical algorithms based on the reduced-order model. The next section focuses on the derivation of the reduced-order formulation and its bi-orthogonality-preserving property is proved in Sec. 2.2. Because error analysis also hints on how spatial and stochastic modes can be adaptively and dynamically added into and removed in computations (Sec. 2.6), it is deferred to Sec. 2.5 after two important aspects of numerical algorithms, representations of stochastic modes and eigenvalue crossing, have been discussed (Sec. 2.3 and Sec. 2.4). In Sec. 2.7, the overall numerical algorithm is presented along with other implementation details.

Before jumping to the derivation, two remarks are worthy of mentioning,

Remark 2.1. Later in the derivation, we will find many terms emerging as orthogonal complements with respect to the orthogonal basis \mathbf{U} in $\mathbb{L}^2(\mathcal{D})$ or \mathbf{Y} in $\mathbb{L}^2(\Omega)$. Thus, we define the following

orthogonal complementary operators

$$\begin{aligned}\Pi_{\mathbf{U}}(v) &= v - \mathbf{U}\mathbf{\Lambda}_{\mathbf{U}}^{-1}\langle\mathbf{U}^T, v\rangle && \text{for } v(x) \in \mathbb{L}^2(\mathcal{D}), \\ \Pi_{\mathbf{Y}}(Z) &= Z - \mathbf{Y}\mathbb{E}[\mathbf{Y}^T Z] && \text{for } Z(\omega) \in \mathbb{L}^2(\Omega).\end{aligned}$$

Remark 2.2. Anti-symmetrization operator $\mathcal{Q} : \mathbb{R}^{k \times k} \rightarrow \mathbb{R}^{k \times k}$ and partial anti-symmetrization operator $\tilde{\mathcal{Q}} : \mathbb{R}^{k \times k} \rightarrow \mathbb{R}^{k \times k}$ play an important role in our derivations, which are defined, respectively, as follows. For any real matrix $\mathbf{A} \in \mathbb{R}^{k \times k}$,

$$\begin{aligned}\mathcal{Q}(\mathbf{A}) &= \frac{1}{2}(\mathbf{A} - \mathbf{A}^T), \\ \tilde{\mathcal{Q}}(\mathbf{A}) &= \frac{1}{2}(\mathbf{A} - \mathbf{A}^T) + \text{diag}(\mathbf{A}),\end{aligned}$$

where $\text{diag}(\mathbf{A})$ denotes a diagonal matrix constructed from a square matrix \mathbf{A} by removing the off-diagonal entries. In the partial “anti-symmetrization” operation, the diagonal entries remain untouched while the off-diagonal entries are anti-symmetrized. Both \mathcal{Q} and $\tilde{\mathcal{Q}}$ are used primarily to enforce bi-orthogonality, which characterizes KLE according to Theorem 1.2.

2.1 Derivations of DyBO Method

For the ease of readers, the derivations are broken down into three steps.

2.1.1 Derivations of Reduced-Order Model for Truncated KLE

We begin our derivation by plugging eqn. (2.5) into the first equation of the original stochastic partial equation eqn. (1.1a) and get

$$\frac{\partial \bar{u}}{\partial t} + \frac{\partial \mathbf{U}}{\partial t} \mathbf{Y}^T + \mathbf{U} \frac{d\mathbf{Y}^T}{dt} = \mathcal{L}u, \quad (2.7)$$

or equivalently,

$$\frac{\partial \bar{u}}{\partial t} + \mathbf{Y} \frac{\partial \mathbf{U}^T}{\partial t} + \frac{d\mathbf{Y}}{dt} \mathbf{U}^T = \mathcal{L}u. \quad (2.8)$$

Taking expectations on both sides of eqn. (2.7) and taking into account the fact that Y_i 's are zero-mean random variables, we have

$$\frac{\partial \bar{u}}{\partial t} = \mathbb{E}[\mathcal{L}u],$$

which gives the evolutionary equation for the expectation of the solution u . Multiplying both sides of eqn. (2.7) by \mathbf{Y} from the right and taking expectations, we obtain

$$\frac{\partial \mathbf{U}}{\partial t} \mathbb{E}[\mathbf{Y}^T \mathbf{Y}] + \mathbf{U} \mathbb{E}\left[\frac{d\mathbf{Y}^T}{dt} \mathbf{Y}\right] = \mathbb{E}\left[\left(\mathcal{L}u - \frac{\partial \bar{u}}{\partial t}\right) \mathbf{Y}\right].$$

After using the orthogonality of stochastic modes \mathbf{Y} , we have

$$\frac{\partial \mathbf{U}}{\partial t} = \mathbb{E}\left[\tilde{\mathcal{L}}u \mathbf{Y}\right] - \mathbf{U} \mathbb{E}\left[\frac{d\mathbf{Y}^T}{dt} \mathbf{Y}\right],$$

where $\tilde{\mathcal{L}}u = \mathcal{L}u - \mathbb{E}[\mathcal{L}u]$. Similarly steps can be repeated for \mathbf{Y} , i.e., multiplying both sides of eqn. (2.8) by \mathbf{U} from the right and taking inner products give

$$\mathbf{Y} \left\langle \frac{\partial \mathbf{U}^T}{\partial t}, \mathbf{U} \right\rangle + \frac{d\mathbf{Y}}{dt} \langle \mathbf{U}^T, \mathbf{U} \rangle = \left\langle \mathcal{L}u - \frac{\partial \bar{u}}{\partial t}, \mathbf{U} \right\rangle,$$

or after the orthogonality of spatial modes \mathbf{U} is used,

$$\frac{d\mathbf{Y}}{dt}\mathbf{\Lambda}_{\mathbf{U}} = \langle \tilde{\mathcal{L}}u, \mathbf{U} \rangle - \mathbf{Y} \left\langle \frac{\partial \mathbf{U}^T}{\partial t}, \mathbf{U} \right\rangle.$$

Therefore, we have the following time-evolutionary system

$$\frac{\partial \bar{u}}{\partial t} = \mathbb{E}[\mathcal{L}u], \quad (2.9a)$$

$$\frac{\partial \mathbf{U}}{\partial t} = \mathbb{E}[\tilde{\mathcal{L}}u\mathbf{Y}] - \mathbf{U}\mathbb{E}\left[\frac{d\mathbf{Y}^T}{dt}\mathbf{Y}\right], \quad (2.9b)$$

$$\frac{d\mathbf{Y}}{dt}\mathbf{\Lambda}_{\mathbf{U}} = \langle \tilde{\mathcal{L}}u, \mathbf{U} \rangle - \mathbf{Y} \left\langle \frac{\partial \mathbf{U}^T}{\partial t}, \mathbf{U} \right\rangle, \quad (2.9c)$$

which is apparently not a truly time-evolutionary system since the right sides still involve time derivatives such as $\frac{d\mathbf{Y}}{dt}$ and $\frac{\partial \mathbf{U}}{\partial t}$. To transform it into a truly time-evolutionary system, we continue by substituting eqn. (2.9c) into eqn. (2.9b) and get,

$$\begin{aligned} \frac{\partial \mathbf{U}}{\partial t} &= \mathbb{E}[\tilde{\mathcal{L}}u\mathbf{Y}] - \mathbf{U}\mathbb{E}\left[\mathbf{\Lambda}_{\mathbf{U}}^{-1}\left(\langle \mathbf{U}^T, \tilde{\mathcal{L}}u \rangle - \left\langle \mathbf{U}^T, \frac{\partial \mathbf{U}}{\partial t} \right\rangle \mathbf{Y}^T\right)\mathbf{Y}\right] \\ &= \mathbb{E}[\tilde{\mathcal{L}}u\mathbf{Y}] - \mathbf{U}\mathbf{\Lambda}_{\mathbf{U}}^{-1}\langle \mathbf{U}^T, \mathbb{E}[\tilde{\mathcal{L}}u\mathbf{Y}] \rangle + \mathbf{U}\mathbf{\Lambda}_{\mathbf{U}}^{-1}\left\langle \mathbf{U}^T, \frac{\partial \mathbf{U}}{\partial t} \right\rangle, \end{aligned}$$

where we have used the orthogonality of the stochastic modes, i.e., $\mathbb{E}[\mathbf{Y}^T\mathbf{Y}] = \mathbf{I}$. Similarly, substituting eqn. (2.9b) to eqn. (2.9c) yields,

$$\begin{aligned} \frac{d\mathbf{Y}}{dt}\mathbf{\Lambda}_{\mathbf{U}} &= \langle \tilde{\mathcal{L}}u, \mathbf{U} \rangle - \mathbf{Y} \left\langle \mathbb{E}[\tilde{\mathcal{L}}u\mathbf{Y}^T] - \mathbb{E}\left[\mathbf{Y}^T \frac{d\mathbf{Y}}{dt}\right] \mathbf{U}^T, \mathbf{U} \right\rangle \\ &= \langle \tilde{\mathcal{L}}u, \mathbf{U} \rangle - \mathbf{Y} \left\langle \mathbb{E}[\tilde{\mathcal{L}}u\mathbf{Y}^T], \mathbf{U} \right\rangle + \mathbf{Y}\mathbb{E}\left[\mathbf{Y}^T \frac{d\mathbf{Y}}{dt}\right] \mathbf{\Lambda}_{\mathbf{U}}. \end{aligned}$$

So the system (2.9) can be re-written as

$$\frac{\partial \bar{u}}{\partial t} = \mathbb{E}[\mathcal{L}u], \quad (2.10a)$$

$$\frac{\partial \mathbf{U}}{\partial t} = \mathbf{U} \mathbf{\Lambda}_{\mathbf{U}}^{-1} \left\langle \mathbf{U}^T, \frac{\partial \mathbf{U}}{\partial t} \right\rangle + G_{\mathbf{U}}(u, \mathbf{U}, \mathbf{Y}), \quad (2.10b)$$

$$\frac{d\mathbf{Y}}{dt} = \mathbf{Y} \mathbb{E} \left[\mathbf{Y}^T \frac{d\mathbf{Y}}{dt} \right] + G_{\mathbf{Y}}(u, \mathbf{U}, \mathbf{Y}), \quad (2.10c)$$

where

$$G_{\mathbf{U}}(u, \mathbf{U}, \mathbf{Y}) = \Pi_{\mathbf{U}} \left(\mathbb{E} [\tilde{\mathcal{L}}u\mathbf{Y}] \right) = \mathbb{E} [\tilde{\mathcal{L}}u\mathbf{Y}] - \mathbf{U} \mathbf{\Lambda}_{\mathbf{U}}^{-1} \left\langle \mathbf{U}^T, \mathbb{E} [\tilde{\mathcal{L}}u\mathbf{Y}] \right\rangle, \quad (2.11a)$$

$$G_{\mathbf{Y}}(u, \mathbf{U}, \mathbf{Y}) = \Pi_{\mathbf{Y}} \left(\left\langle \tilde{\mathcal{L}}u, \mathbf{U} \right\rangle \right) \mathbf{\Lambda}_{\mathbf{U}}^{-1} = \left\langle \tilde{\mathcal{L}}u, \mathbf{U} \right\rangle \mathbf{\Lambda}_{\mathbf{U}}^{-1} - \mathbf{Y} \left\langle \mathbb{E} [\tilde{\mathcal{L}}u\mathbf{Y}^T], \mathbf{U} \right\rangle \mathbf{\Lambda}_{\mathbf{U}}^{-1}, \quad (2.11b)$$

i.e., $G_{\mathbf{U}}(u, \mathbf{U}, \mathbf{Y})$ is the orthogonal compliment of $\mathbb{E} [\tilde{\mathcal{L}}u\mathbf{Y}]$ with respect to $\text{span}(\mathbf{U})$ in $\mathbb{L}^2(\mathcal{D})$

and $G_{\mathbf{Y}}(u, \mathbf{U}, \mathbf{Y})$ is the orthogonal compliment of $\left\langle \tilde{\mathcal{L}}u, \mathbf{U} \right\rangle$ with respect to $\text{span}(\mathbf{Y})$ in $\mathbb{L}^2(\Omega)$.

In other words,

$$G_{\mathbf{U}}(u, \mathbf{U}, \mathbf{Y}) \perp \text{span} \mathbf{U} \quad \text{in } \mathbb{L}^2(\mathcal{D}),$$

$$G_{\mathbf{Y}}(u, \mathbf{U}, \mathbf{Y}) \perp \text{span} \mathbf{Y} \quad \text{in } \mathbb{L}^2(\Omega).$$

2.1.2 Enforcing Bi-Orthogonality Condition

The following crucial observation makes further derivation possible and its meaning will become clear soon. The partial anti-symmetrization $\tilde{\mathcal{Q}}$ of $\left\langle \mathbf{U}^T, \frac{\partial \mathbf{U}}{\partial t} \right\rangle$ in the first term on the right side of eqn. (2.10b) and the anti-symmetrization \mathcal{Q} of $\mathbb{E} \left[\mathbf{Y}^T \frac{d\mathbf{Y}}{dt} \right]$ in the first term on the right side of eqn. (2.10c) do produce an equivalent system to the system (2.10) as long as \mathbf{U} and \mathbf{Y} are bi-orthogonal. This can be seen by taking the temporal derivative of orthogonality conditions (2.3a)

and (2.3b),

$$\begin{aligned}\frac{d}{dt}\langle \mathbf{U}^T, \mathbf{U} \rangle &= \left\langle \frac{d\mathbf{U}^T}{dt}, \mathbf{U} \right\rangle + \left\langle \mathbf{U}^T, \frac{d\mathbf{U}}{dt} \right\rangle, \\ \frac{d}{dt}\mathbb{E}[\mathbf{Y}^T \mathbf{Y}] &= \mathbb{E}\left[\frac{d\mathbf{Y}^T}{dt} \mathbf{Y}\right] + \mathbb{E}\left[\mathbf{Y}^T \frac{d\mathbf{Y}}{dt}\right].\end{aligned}$$

Obviously, the off-diagonal elements of both matrices $\frac{d}{dt}\langle \mathbf{U}^T, \mathbf{U} \rangle$ and $\frac{d}{dt}\mathbb{E}[\mathbf{Y}^T \mathbf{Y}]$ are zeros due to the bi-orthogonality condition (2.3), which in turn implies both matrices $\left\langle \frac{d\mathbf{U}^T}{dt}, \mathbf{U} \right\rangle$ and $\mathbb{E}\left[\frac{d\mathbf{Y}^T}{dt} \mathbf{Y}\right]$ are invariant under partial anti-symmetrization $\tilde{\mathcal{Q}}$. Furthermore, \mathbf{Y} being orthonormal implies $\mathbb{E}\left[\frac{d\mathbf{Y}^T}{dt} \mathbf{Y}\right]$ is anti-symmetric. On the other hand, bi-orthogonality (2.3) is preserved if it is satisfied initially at $t = 0$ and such invariance is satisfied at any later time $t > 0$, which is summarized in the following theorem.

Theorem 2.1 (Equivalence of bi-orthogonality and anti-symmetry). *Bi-orthogonality condition (2.3) is preserved all the time if and only if it is true initially and, for $\forall t > 0$,*

$$\tilde{\mathcal{Q}}\left(\left\langle \mathbf{U}^T, \frac{\partial \mathbf{U}}{\partial t} \right\rangle\right) = \left\langle \mathbf{U}^T, \frac{\partial \mathbf{U}}{\partial t} \right\rangle, \quad (2.14a)$$

$$\mathcal{Q}\left(\mathbb{E}\left[\mathbf{Y}^T \frac{d\mathbf{Y}}{dt}\right]\right) = \mathbb{E}\left[\mathbf{Y}^T \frac{d\mathbf{Y}}{dt}\right]. \quad (2.14b)$$

In other words, anti-symmetrization enforces the bi-orthogonality condition, which essentially characterizes KL expansions. After applying the above theorem to the system (2.10), we arrive at

$$\frac{\partial \bar{u}}{\partial t} = \mathbb{E}[\mathcal{L}u], \quad (2.15a)$$

$$\frac{\partial \mathbf{U}}{\partial t} = \mathbf{U} \boldsymbol{\Lambda}_{\mathbf{U}}^{-1} \tilde{\mathcal{Q}}\left(\left\langle \mathbf{U}^T, \frac{\partial \mathbf{U}}{\partial t} \right\rangle\right) + G_{\mathbf{U}}(u, \mathbf{U}, \mathbf{Y}), \quad (2.15b)$$

$$\frac{d\mathbf{Y}}{dt} = \mathbf{Y} \mathcal{Q}\left(\mathbb{E}\left[\mathbf{Y}^T \frac{d\mathbf{Y}}{dt}\right]\right) + G_{\mathbf{Y}}(u, \mathbf{U}, \mathbf{Y}). \quad (2.15c)$$

2.1.3 Eliminating Time Derivatives from RHS

We continue with our derivation by noticing that $(\mathbf{U}, \tilde{\mathbf{U}})$ forms a complete orthogonal basis of $\mathbb{L}^2(\mathcal{D})$. Thus, the change of spatial modes, $\frac{\partial \mathbf{U}}{\partial t}$, can be written in the form of

$$\frac{\partial \mathbf{U}}{\partial t} = \mathbf{U}\mathbf{C} + \tilde{\mathbf{U}}\tilde{\mathbf{C}}, \quad (2.16)$$

where $\mathbf{C}(t) \in \mathbb{R}^{m \times m}$ and $\tilde{\mathbf{C}}(t) \in \mathbb{R}^{\infty \times m}$. Note that

$$\left\langle \mathbf{U}^T, \frac{\partial \mathbf{U}}{\partial t} \right\rangle = \left\langle \mathbf{U}^T, \mathbf{U}\mathbf{C} + \tilde{\mathbf{U}}\tilde{\mathbf{C}} \right\rangle = \langle \mathbf{U}^T, \mathbf{U} \rangle \mathbf{C} + \langle \mathbf{U}^T, \tilde{\mathbf{U}} \rangle \tilde{\mathbf{C}} = \langle \mathbf{U}^T, \mathbf{U} \rangle \mathbf{C}.$$

Substituting this and eqn. (2.16) into the second equation of the system (2.15) gives

$$\mathbf{U}\mathbf{C} + \tilde{\mathbf{U}}\tilde{\mathbf{C}} = \mathbf{U}\mathbf{\Lambda}_{\mathbf{U}}^{-1} \tilde{\mathcal{Q}}(\mathbf{\Lambda}_{\mathbf{U}}\mathbf{C}) + G_{\mathbf{U}}(u, \mathbf{U}, \mathbf{Y}),$$

or

$$\mathbf{U} \left(\mathbf{C} - \mathbf{\Lambda}_{\mathbf{U}}^{-1} \tilde{\mathcal{Q}}(\mathbf{\Lambda}_{\mathbf{U}}\mathbf{C}) \right) = G_{\mathbf{U}}(u, \mathbf{U}, \mathbf{Y}) - \tilde{\mathbf{U}}\tilde{\mathbf{C}},$$

where the left side is in $\text{span}(\mathbf{U}) \subset \mathbb{L}^2(\mathcal{D})$ while the right side is in its orthogonal complement, so

$$\mathbf{C} - \mathbf{\Lambda}_{\mathbf{U}}^{-1} \tilde{\mathcal{Q}}(\mathbf{\Lambda}_{\mathbf{U}}\mathbf{C}) = 0, \quad (2.17a)$$

$$\tilde{\mathbf{U}}\tilde{\mathbf{C}} = G_{\mathbf{U}}(u, \mathbf{U}, \mathbf{Y}). \quad (2.17b)$$

Similarly, the change of stochastic modes, $\frac{d\mathbf{Y}}{dt}$, can be written in the form of

$$\frac{d\mathbf{Y}}{dt} = \mathbf{Y}\mathbf{D} + \tilde{\mathbf{Y}}\tilde{\mathbf{D}}, \quad (2.18)$$

where $\mathbf{D}(t) \in \mathbb{R}^{m \times m}$ and $\tilde{\mathbf{D}}(t) \in \mathbb{R}^{\infty \times m}$. Immediately, we see

$$\mathbb{E} \left[\mathbf{Y}^T \frac{d\mathbf{Y}}{dt} \right] = \mathbb{E} [\mathbf{Y}^T \mathbf{Y}] \mathbf{D} + \mathbb{E} [\mathbf{Y}^T \tilde{\mathbf{Y}}] \tilde{\mathbf{D}} = \mathbf{D}.$$

substituting this and eqn. (2.18) into the third equation of the system (2.15), we get

$$\mathbf{Y}\mathbf{D} + \tilde{\mathbf{Y}}\tilde{\mathbf{D}} = \mathbf{Y}\mathcal{Q}(\mathbf{D}) + G_{\mathbf{Y}}(u, \mathbf{U}, \mathbf{Y}),$$

or

$$\mathbf{Y}(\mathbf{D} - \mathcal{Q}(\mathbf{D})) = G_{\mathbf{Y}}(u, \mathbf{U}, \mathbf{Y}) - \tilde{\mathbf{Y}}\tilde{\mathbf{D}},$$

where the left side is in $\text{span}(\mathbf{Y}) \subset \mathbb{L}^2(\Omega)$ and the right side is in its orthogonal complement, so

$$\mathbf{D} - \mathcal{Q}(\mathbf{D}) = 0, \tag{2.19a}$$

$$\tilde{\mathbf{Y}}\tilde{\mathbf{D}} = G_{\mathbf{Y}}(u, \mathbf{U}, \mathbf{Y}). \tag{2.19b}$$

However, eqn. (2.17a) and eqn. (2.19a) are not sufficient to determine matrices \mathbf{C} and \mathbf{D} . To find additional equations for \mathbf{C} and \mathbf{D} , we substitute eqn. (2.16) and eqn. (2.18) back into the original stochastic partial differential equation eqn. (1.1a) and get

$$\frac{\partial \mathbf{U}}{\partial t} \mathbf{Y}^T + \mathbf{U} \frac{d\mathbf{Y}^T}{dt} = \mathbf{Y}\mathbf{C}^T \mathbf{U}^T + \mathbf{Y}\tilde{\mathbf{C}}^T \tilde{\mathbf{U}}^T + \mathbf{U}\mathbf{D}^T \mathbf{Y}^T + \mathbf{U}\tilde{\mathbf{D}}^T \tilde{\mathbf{Y}}^T = \tilde{\mathcal{L}}u,$$

or

$$\mathbf{U}(\mathbf{D}^T + \mathbf{C}) \mathbf{Y}^T + \mathbf{U}\tilde{\mathbf{D}}^T \tilde{\mathbf{Y}}^T + \tilde{\mathbf{U}}\tilde{\mathbf{C}}\mathbf{Y}^T = \tilde{\mathcal{L}}u.$$

Multiplying \mathbf{U}^T from the left and \mathbf{Y} from the right on both sides of the above equality and taking

inner products $\langle \cdot, \cdot \rangle$ and expectations $\mathbb{E}[\cdot]$, we obtain

$$\begin{aligned} \langle \mathbf{U}^T, \mathbf{U} \rangle (\mathbf{D}^T + \mathbf{C}) \mathbb{E}[\mathbf{Y}^T \mathbf{Y}] + \langle \mathbf{U}^T, \mathbf{U} \rangle \tilde{\mathbf{D}}^T \mathbb{E}[\tilde{\mathbf{Y}}^T \mathbf{Y}] \\ + \langle \mathbf{U}^T, \tilde{\mathbf{U}} \rangle \tilde{\mathbf{C}} \mathbb{E}[\mathbf{Y}^T \mathbf{Y}] = \langle \mathbf{U}^T, \mathbb{E}[\tilde{\mathcal{L}}u\mathbf{Y}] \rangle, \end{aligned}$$

or after applying bi-orthogonality conditions,

$$\mathbf{D}^T + \mathbf{C} = G_*(u, \mathbf{U}, \mathbf{Y}), \quad (2.20)$$

where $G_*(u, \mathbf{U}, \mathbf{Y}) = \Lambda_{\mathbf{U}}^{-1} \langle \mathbf{U}^T, \mathbb{E}[\tilde{\mathcal{L}}u\mathbf{Y}] \rangle \in \mathbb{R}^{m \times m}$. Matrices \mathbf{C} and \mathbf{D} can then be solved uniquely assuming $\langle u_i, u_i \rangle \neq \langle u_j, u_j \rangle$ for $i \neq j$ as the following theorem shows,

Theorem 2.2. *If $\langle u_i, u_i \rangle \neq \langle u_j, u_j \rangle$ for $i \neq j$, $i = 1, 2, \dots, m$, $j = 1, 2, \dots, m$, m -by- m matrices \mathbf{C} and \mathbf{D} can be solved uniquely from the following linear system*

$$\mathbf{C} - \Lambda_{\mathbf{U}}^{-1} \tilde{\mathcal{Q}}(\Lambda_{\mathbf{U}} \mathbf{C}) = 0, \quad (2.21a)$$

$$\mathbf{D} - \mathcal{Q}(\mathbf{D}) = 0, \quad (2.21b)$$

$$\mathbf{D}^T + \mathbf{C} = G_*(u, \mathbf{U}, \mathbf{Y}). \quad (2.21c)$$

The solutions are given entry-wisely

$$C_{ii} = G_{*ii}, \quad (2.22a)$$

$$C_{ij} = \frac{\|u_j\|_{\mathbb{L}^2(\mathcal{D})}^2}{\|u_j\|_{\mathbb{L}^2(\mathcal{D})}^2 - \|u_i\|_{\mathbb{L}^2(\mathcal{D})}^2} (G_{*ij} + G_{*ji}) \quad \text{for } i \neq j, \quad (2.22b)$$

$$D_{ii} = 0, \quad (2.22c)$$

$$D_{ij} = \frac{1}{\|u_j\|_{\mathbb{L}^2(\mathcal{D})}^2 - \|u_i\|_{\mathbb{L}^2(\mathcal{D})}^2} \left(\|u_j\|_{\mathbb{L}^2(\mathcal{D})}^2 G_{*ji} + \|u_i\|_{\mathbb{L}^2(\mathcal{D})}^2 G_{*ij} \right) \quad \text{for } i \neq j. \quad (2.22d)$$

Proof of solvability of matrices C and D. From the second equation (2.21b), we have $\mathbf{D} - \mathcal{Q}(\mathbf{D}) = \mathbf{D} - \frac{1}{2}(\mathbf{D} - \mathbf{D}^T) = \frac{1}{2}(\mathbf{D} + \mathbf{D}^T) = 0$, i.e., \mathbf{D} must be anti-symmetric matrix, which gives eqn. (2.22c). Taking transposes on both sides of eqn. (2.21c) and adding to itself, we arrive at

$$\mathbf{C} + \mathbf{C}^T = G_*(u, \mathbf{U}, \mathbf{Y}) + G_*(u, \mathbf{U}, \mathbf{Y})^T,$$

which gives eqn. (2.22a) for the diagonal entries of matrix \mathbf{C} . For the off-diagonal entries $i \neq j$,

$$C_{ij} + C_{ji} = G_{*ij}(u, \mathbf{U}, \mathbf{Y}) + G_{*ji}(u, \mathbf{U}, \mathbf{Y}). \quad (2.23)$$

On the other hand, by the definition of anti-symmetrization operator $\tilde{\mathcal{Q}}$, eqn. (2.21a) can be simplified to

$$\frac{1}{2}(\mathbf{C} + \mathbf{\Lambda}_{\mathbf{U}}^{-1} \mathbf{C}^T \mathbf{\Lambda}_{\mathbf{U}}) - \text{diag}(\mathbf{C}) = 0, \quad (2.24)$$

which gives, for the diagonal entries $i = j$, a trivial equality,

$$C_{ii} - C_{ii} = 0,$$

and for the off-diagonal entries $i \neq j$,

$$C_{ij} + \frac{\|u_j\|_{\mathbb{L}^2(\mathcal{D})}^2}{\|u_i\|_{\mathbb{L}^2(\mathcal{D})}^2} C_{ji} = 0. \quad (2.25)$$

Since $\|u_i\|_{\mathbb{L}^2(\mathcal{D})} \neq \|u_j\|_{\mathbb{L}^2(\mathcal{D})}$ for $i \neq j$, solving eqn. (2.23) and eqn. (2.25) gives eqn. (2.22b).

Substituting in eqn. (2.21c), we arrive at eqn. (2.22d). \square

Remark 2.3. In the event that two eigenvalues approach each other and then separate as time increases, which we refer to as the eigenvalue-crossing case, we can detect such an event and temporarily freeze the spatial modes \mathbf{U} or the stochastic modes \mathbf{Y} and continue to evolve the system for a short duration. Once the two eigenvalues separate, the solution can be recast into the bi-orthogonal form via KL expansion. However, we can take advantage of the fact that one of \mathbf{U} and \mathbf{Y} is always kept orthogonal even during the “freezing” stage and devise a KL expansion algorithm which avoids the need to form the covariance function explicitly. The details will be discussed in Sec. 2.4.

2.1.4 DyBO Formulation for Time-Evolutionary SPDE

Combining the above discussion, we arrive at the following reduced-order system

$$\frac{\partial \bar{u}}{\partial t} = \mathbb{E}[\mathcal{L}u], \quad (2.26a)$$

$$\frac{\partial \mathbf{U}}{\partial t} = \mathbf{U}\mathbf{C} + G_{\mathbf{U}}(u, \mathbf{U}, \mathbf{Y}), \quad (2.26b)$$

$$\frac{d\mathbf{Y}}{dt} = \mathbf{Y}\mathbf{D} + G_{\mathbf{Y}}(u, \mathbf{U}, \mathbf{Y}), \quad (2.26c)$$

where \mathbf{C} and \mathbf{D} are given in (2.22), $G_{\mathbf{U}}$ and $G_{\mathbf{Y}}$ are given in (2.11) and the solution of SPDE is approximated by a bi-orthogonal form

$$u \approx \bar{u} + \mathbf{U}^T \mathbf{Y} = \bar{u} + \sum_{i=1}^m u_i(x, t) Y_i(\omega, t).$$

From the definition of G_* , note that $G_{\mathbf{U}}$ and $G_{\mathbf{Y}}$ can be rewritten as

$$\begin{aligned} G_{\mathbf{U}} &= \mathbb{E} \left[\tilde{\mathcal{L}}u \mathbf{Y} \right] - \mathbf{U} G_*, \\ G_{\mathbf{Y}} &= \left\langle \tilde{\mathcal{L}}u, \mathbf{U} \right\rangle \mathbf{\Lambda}_{\mathbf{U}}^{-1} - \mathbf{Y} G_*^T, \end{aligned}$$

so the system (2.26) can be rewritten as

$$\frac{\partial \bar{u}}{\partial t} = \mathbb{E} [\mathcal{L}u], \quad (2.27a)$$

$$\frac{\partial \mathbf{U}}{\partial t} = \mathbf{U} (\mathbf{C} - G_*) + \mathbb{E} \left[\tilde{\mathcal{L}}u \mathbf{Y} \right], \quad (2.27b)$$

$$\frac{d\mathbf{Y}}{dt} = \mathbf{Y} (\mathbf{D} - G_*^T) + \left\langle \tilde{\mathcal{L}}u, \mathbf{U} \right\rangle \mathbf{\Lambda}_{\mathbf{U}}^{-1}. \quad (2.27c)$$

Furthermore, after plugging in eqn. (2.21c), we finally arrive at the reduced-order system for the original SPDE (1.1),

$$\frac{\partial \bar{u}}{\partial t} = \mathbb{E} [\mathcal{L}u], \quad (2.28a)$$

$$\frac{\partial \mathbf{U}}{\partial t} = -\mathbf{U} \mathbf{D}^T + \mathbb{E} \left[\tilde{\mathcal{L}}u \mathbf{Y} \right], \quad (2.28b)$$

$$\frac{d\mathbf{Y}}{dt} = -\mathbf{Y} \mathbf{C}^T + \left\langle \tilde{\mathcal{L}}u, \mathbf{U} \right\rangle \mathbf{\Lambda}_{\mathbf{U}}^{-1}, \quad (2.28c)$$

which will be the main focus throughout the thesis. We also call the above system the DyBO formulation of SPDE (1.1a).

Remark 2.4. In some applications, such as predictions of flow and transport in porous media, convergence of such expansion in stronger norms, e.g., $\|\cdot\|_{\mathbb{L}^2(\Omega \rightarrow \mathcal{H}^k(\mathcal{D}))}$, is preferred. Generalization of KLE and then our DyBO method in this sense will be discussed in Chapter 5, which in turn enables the method proposed in this thesis to be applicable for a broader class of problems.

2.2 Bi-Orthogonality Preservation

In this section, we are going to show in the following theorem that the DyBO formulation preserves the bi-orthogonality of the spatial modes \mathbf{U} and the stochastic modes \mathbf{Y} for $t \in [0, T]$ if the bi-orthogonality condition is satisfied initially. The case that \mathbf{U} and \mathbf{Y} initially are not perfectly bi-orthogonal is considered in Theorem 2.4 and Remark 2.5.

Theorem 2.3 (Preservation of Bi-Orthogonality in DyBO). *The solutions \mathbf{U} and \mathbf{Y} of system (2.28) satisfy the bi-orthogonality condition (2.3) exactly as long as the initial conditions $\mathbf{U}(x, 0)$ and $\mathbf{Y}(\omega, 0)$ satisfy the bi-orthogonality condition. Moreover, \mathbf{Y} remains normalized exactly, i.e., $\|Y_i\|_{\mathbb{L}^2(\Omega)}(t) = 1$ for $i = 1, 2, \dots, m$ and $t \in [0, T]$.*

To motivate the proof, we first study the semi-discretization of system (2.28) at $t = n \, dt$, $n = 0, 1, 3, \dots$, if we are set out to use the forward Euler scheme with time step size dt . Assume the bi-orthogonality is preserved at time $t = t^n$ and let us evaluate at time $t = t^n$,

$$\begin{aligned}
\left\langle \frac{\partial \mathbf{U}^T}{\partial t}, \mathbf{U} \right\rangle \Big|_{t=t^n} &= \langle \mathbf{C}^T \mathbf{U}^T + G_{\mathbf{U}}(u, \mathbf{U}, \mathbf{Y})^T, \mathbf{U} \rangle \Big|_{t=t^n} \\
&= \langle \mathbf{C}^T \mathbf{U}^T + \mathbb{E} [\tilde{\mathcal{L}} u \mathbf{Y}^T] - \langle \mathbb{E} [\tilde{\mathcal{L}} u \mathbf{Y}^T], \mathbf{U} \rangle \mathbf{\Lambda}_{\mathbf{U}}^{-1} \mathbf{U}^T, \mathbf{U} \rangle \Big|_{t=t^n} \\
&= \mathbf{C}^T \langle \mathbf{U}^T, \mathbf{U} \rangle \Big|_{t=t^n} + \langle \mathbb{E} [\tilde{\mathcal{L}} u \mathbf{Y}^T], \mathbf{U} \rangle \Big|_{t=t^n} \\
&\quad - \langle \mathbb{E} [\tilde{\mathcal{L}} u \mathbf{Y}^T], \mathbf{U} \rangle \mathbf{\Lambda}_{\mathbf{U}}^{-1} \langle \mathbf{U}^T, \mathbf{U} \rangle \Big|_{t=t^n} \\
&= \mathbf{C}^T \mathbf{\Lambda}_{\mathbf{U}} \Big|_{t=t^n},
\end{aligned}$$

where we have used the orthogonality of \mathbf{U} at $t = t^n$ in the second to last line. Similarly,

$$\left\langle \mathbf{U}^T, \frac{\partial \mathbf{U}}{\partial t} \right\rangle \Big|_{t=t^n} = \boldsymbol{\Lambda}_{\mathbf{U}} \mathbf{C} \Big|_{t=t^n}.$$

Combining the above two equations, we have

$$\begin{aligned} \frac{d}{dt} \langle \mathbf{U}^T, \mathbf{U} \rangle \Big|_{t=t^n} &\approx \left\langle \frac{\partial \mathbf{U}^T}{\partial t}, \mathbf{U} \right\rangle \Big|_{t=t^n} + \left\langle \mathbf{U}^T, \frac{\partial \mathbf{U}}{\partial t} \right\rangle \Big|_{t=t^n} \\ &= \mathbf{C}^T \boldsymbol{\Lambda}_{\mathbf{U}} + \boldsymbol{\Lambda}_{\mathbf{U}} \mathbf{C} \Big|_{t=t^n} \\ &= \boldsymbol{\Lambda}_{\mathbf{U}} (\mathbf{C} + \boldsymbol{\Lambda}_{\mathbf{U}}^{-1} \mathbf{C}^T \boldsymbol{\Lambda}_{\mathbf{U}}) \Big|_{t=t^n}. \end{aligned}$$

We know immediately from eqn. (2.24) that the off-diagonal entry

$$\frac{d}{dt} \langle u_i, u_j \rangle \Big|_{t=t^n} = 0, \quad \text{for } i \neq j,$$

implying that \mathbf{U} remains orthogonal at time $t = t^{n+1}$ since $\langle u_i, u_j \rangle \Big|_{t=t^{n+1}} = \langle u_i, u_j \rangle \Big|_{t=t^n} + \frac{d}{dt} \langle u_i, u_j \rangle \Big|_{t=t^n} dt$. Similarly for \mathbf{Y} , we have

$$\begin{aligned} \mathbb{E} \left[\frac{d\mathbf{Y}^T}{dt} \mathbf{Y} \right] \Big|_{t=t^n} &= \mathbb{E} [(\mathbf{D}^T \mathbf{Y}^T + G_{\mathbf{Y}}(u, \mathbf{U}, \mathbf{Y})^T) \mathbf{Y}] \Big|_{t=t^n} \\ &= \mathbb{E} \left[\left(\mathbf{D}^T \mathbf{Y}^T + \boldsymbol{\Lambda}_{\mathbf{U}}^{-1} \langle \mathbf{U}^T, \tilde{\mathcal{L}}u \rangle - \boldsymbol{\Lambda}_{\mathbf{U}}^{-1} \langle \mathbf{U}^T, \mathbb{E} [\tilde{\mathcal{L}}u \mathbf{Y}] \rangle \right) \mathbf{Y}^T \right] \Big|_{t=t^n} \\ &= \mathbf{D}^T \mathbb{E} [\mathbf{Y}^T \mathbf{Y}] \Big|_{t=t^n} + \boldsymbol{\Lambda}_{\mathbf{U}}^{-1} \langle \mathbf{U}^T, \mathbb{E} [\tilde{\mathcal{L}}u \mathbf{Y}^T] \rangle \Big|_{t=t^n} \\ &\quad - \boldsymbol{\Lambda}_{\mathbf{U}}^{-1} \langle \mathbf{U}^T, \mathbb{E} [\tilde{\mathcal{L}}u \mathbf{Y}] \rangle \mathbb{E} [\mathbf{Y}^T \mathbf{Y}] \Big|_{t=t^n} \\ &= \mathbf{D}^T \Big|_{t=t^n}, \end{aligned}$$

and

$$\mathbb{E} \left[\mathbf{Y}^T \frac{d\mathbf{Y}}{dt} \right] \Big|_{t=t^n} = \mathbf{D} \Big|_{t=t^n}.$$

Thus, \mathbf{Y} remains orthonormal at time $t = t^{n+1}$ because, at time $t = t^n$,

$$\frac{d}{dt} \mathbb{E} [\mathbf{Y}^T \mathbf{Y}] \Big|_{t=t^n} \approx \mathbb{E} \left[\frac{d\mathbf{Y}^T}{dt} \mathbf{Y} \right] \Big|_{t=t^n} + \mathbb{E} \left[\mathbf{Y}^T \frac{d\mathbf{Y}}{dt} \right] \Big|_{t=t^n} = \mathbf{D}^T + \mathbf{D} \Big|_{t=t^n} = 0.$$

The above ‘‘discretized’’ version of proof can be turned into a rigorous and ‘‘continuous’’ one

Proof of Theorem 2.3. Like in the ‘‘discretized’’ version, we evaluate directly

$$\begin{aligned} \frac{d}{dt} \langle \mathbf{U}^T, \mathbf{U} \rangle &= \left\langle \frac{\partial \mathbf{U}^T}{\partial t}, \mathbf{U} \right\rangle + \left\langle \mathbf{U}^T, \frac{\partial \mathbf{U}}{\partial t} \right\rangle \\ &= -\mathbf{D} \langle \mathbf{U}^T, \mathbf{U} \rangle + \left\langle \mathbb{E} [\tilde{\mathcal{L}}u \mathbf{Y}^T], \mathbf{U} \right\rangle - \langle \mathbf{U}^T, \mathbf{U} \rangle \mathbf{D} + \left\langle \mathbf{U}^T, \mathbb{E} [\tilde{\mathcal{L}}u \mathbf{Y}] \right\rangle \\ &= -\mathbf{D} \langle \mathbf{U}^T, \mathbf{U} \rangle - \langle \mathbf{U}^T, \mathbf{U} \rangle \mathbf{D}^T + G_*^T \Lambda_{\mathbf{U}} + \Lambda_{\mathbf{U}} G_*, \end{aligned}$$

where we have used the definition of G_* in the second to last equality. For $i \neq j$,

$$\begin{aligned} \frac{d}{dt} \langle u_i, u_j \rangle &= - \sum_{k=1}^m D_{ik} \langle u_k, u_j \rangle - \sum_{l=1}^m \langle u_i, u_l \rangle D_{jl} + G_{*ji} \|u_j\|_{\mathbb{L}^2(\mathcal{D})}^2 + G_{*ij} \|u_i\|_{\mathbb{L}^2(\mathcal{D})}^2 \\ &= - \sum_{k=1, k \neq j}^m D_{ik} \langle u_k, u_j \rangle - \sum_{l=1, l \neq i}^m \langle u_i, u_l \rangle D_{jl} \\ &\quad - D_{ij} \|u_j\|_{\mathbb{L}^2(\mathcal{D})}^2 - D_{ji} \|u_i\|_{\mathbb{L}^2(\mathcal{D})}^2 + \left(\|u_j\|_{\mathbb{L}^2(\mathcal{D})}^2 - \|u_i\|_{\mathbb{L}^2(\mathcal{D})}^2 \right) D_{ij} \\ &= - \sum_{k=1, k \neq i, j}^m D_{ik} \langle u_k, u_j \rangle - \sum_{l=1, l \neq i, j}^m D_{jl} \langle u_i, u_l \rangle, \end{aligned} \tag{2.29}$$

where eqn. (2.22d) is used in the first equality and the last equality is due to the fact that matrix \mathbf{D} is anti-symmetric. Write $\chi_{(i,j)}(t) = \langle u_i, u_j \rangle(t)$ for $i > j$ and column vector $\boldsymbol{\chi}(t) = (\chi_{(i,j)})_{i>j} \in \mathbb{R}^{\frac{m(m-1)}{2} \times 1}$ where the parenthesis enclosing the pair i and j is used to emphasize the pair (i, j) is

actually a linear index. From eqn. (2.29), we see that $\chi(t)$ satisfies a linear ODE system

$$\frac{d\chi(t)}{dt} = \mathcal{W}(t)\chi(t), \quad (2.30a)$$

$$\chi(0) = 0, \quad (2.30b)$$

where matrix $\mathcal{W}(t) \in \mathbb{R}^{\frac{m(m-1)}{2} \times \frac{m(m-1)}{2}}$ is given in terms of D_{ij} and the initial condition comes from the fact that $\mathbf{U}(x, 0)$ are a set of orthogonal functions. It is easy to see that, as long as the solutions of system (2.28) exist, $\mathcal{W}(t)$ is well defined and the ODE system admits only zero solution, i.e., orthogonality of \mathbf{U} is preserved for $t \in [0, T]$. Similar arguments can be used to show \mathbf{Y} remains orthonormal for $t \in [0, T]$, which completes the proof. \square

Furthermore, we have the following theorem regarding the structure of matrix \mathcal{W} .

Theorem 2.4. *Matrix \mathcal{W} in ODE system (2.30) is anti-symmetric.*

Proof of Theorem 2.4. Consider two pairs of indices (i, j) and (p, q) with $i > j$ and $p > q$. The corresponding equations for $\chi_{(i,j)}$ and $\chi_{(p,q)}$ are

$$\frac{d\chi_{(i,j)}}{dt} = - \sum_{k=1, k \neq i, j}^m D_{ik} \chi_{(k,j)} - \sum_{l=1, l \neq i, j}^m D_{jl} \chi_{(i,l)}, \quad (2.31a)$$

$$\frac{d\chi_{(p,q)}}{dt} = - \sum_{k=1, k \neq p, q}^m D_{pk} \chi_{(k,q)} - \sum_{l=1, l \neq p, q}^m D_{ql} \chi_{(p,l)}, \quad (2.31b)$$

where we have identified $\chi_{(i,j)}$ with $\chi_{(j,i)}$. First, we consider the case two index pairs are identical. We see from eqn. (2.31a) that there is no $\chi_{(i,j)}$ on the right side, so $\mathcal{W}_{(i,j),(i,j)} = 0$. Next, we consider the case none of indices in one index pair is equal to any in the other, i.e., $i \neq p, q$ and $j \neq p, q$. It is obvious that no $\chi_{(p,q)}$ appears in the summations on the right side of eqn. (2.31a), which implies $\mathcal{W}_{(i,j),(p,q)} = 0$. Similarly, we have $\mathcal{W}_{(p,q),(i,j)} = 0$. Last, we consider the case in which one index in one index pair is equal to one index in another index pair but the remaining two

indices from two index pair are not equal. Without loss of generality, we assume $i = p$ and $j \neq p, q$.

From eqn. (2.31a), we have

$$\frac{d\chi_{(i,j)}}{dt} = \cdots - D_{jq}\chi_{(i,q)} - \cdots = \cdots - D_{jq}\chi_{(p,q)} - \cdots ,$$

where we have intentionally isolated the relevant term from the second summation and the last equality is due to $i = p$. Similarly, from eqn. (2.31b), we have

$$\frac{d\chi_{(p,q)}}{dt} = \cdots - D_{qj}\chi_{(p,j)} - \cdots = \cdots - D_{qj}\chi_{(i,j)} - \cdots .$$

The above two equations imply that $\mathcal{W}_{(i,j),(p,q)} = \mathcal{W}_{(p,q),(i,j)} = -D_{jq}$. Similar arguments apply to other cases. □

Similar results hold for the stochastic modes \mathbf{Y} . Due to numerical errors, such as round-off errors, and discretization errors, \mathbf{U} and \mathbf{Y} may not be perfectly bi-orthogonal at the beginning or become so later in the computation. Theorem 2.4 sheds lights on the numerical stability of DyBO formulation for this scenario. As we know, the eigenvalues of an anti-symmetric matrix are purely imaginary or zero and the geometric multiplicities are 1 (see page 115, [34]), so any deviation from the bi-orthogonal form will not be amplified.

Remark 2.5. However, we have observed much better results through all of numerical examples, i.e., that the bi-orthogonality condition is preserved almost all the time. Even if spatial and stochastic modes become imperfectly bi-orthogonal, e.g., due to introducing new mode pairs as discussed in Sec. 2.6, the bi-orthogonality is restored after several time iterations. This may be due to the non-symplecticness and artificial numerical dissipation of numerical integrators (see page 33, [49]).

2.3 Representation of Stochastic Modes

Before presenting error analysis of DyBO, we devote this and the next sections to some numerical aspects of DyBO formulation. One of conceptual difficulties, from the viewpoint of the classical numerical PDEs, involves representations of random variables or functions defined on abstract probability space Ω . Essentially, there are two ways to represent numerically stochastic modes $\mathbf{Y}(\omega, t)$: ensemble representations in sampling methods, e.g., MC, and spectral representations, e.g., gPC.

2.3.1 Ensemble Representation

In MC, the stochastic modes $\mathbf{Y}(\omega, t)$ are represented by an ensemble of realizations, i.e.,

$$\mathbf{Y}(\omega, t) \approx \{\mathbf{Y}(\omega_1, t), \mathbf{Y}(\omega_2, t), \mathbf{Y}(\omega_3, t), \dots, \mathbf{Y}(\omega_{N_r}, t)\},$$

where N_r is the total number of realizations. Then the expectations in the DyBO formulation (2.28) can be replaced by ensemble averages, i.e.,

$$\frac{\partial \bar{u}}{\partial t}(x, t) = \frac{1}{N_r} \sum_{i=1}^{N_r} \mathcal{L}u(x, t, \omega_i), \quad (2.32a)$$

$$\frac{\partial \mathbf{U}}{\partial t}(x, t) = -\mathbf{U}(x, t)\mathbf{D}(t)^T + \frac{1}{N_r} \sum_{i=1}^{N_r} \tilde{\mathcal{L}}u(x, t, \omega_i)\mathbf{Y}(\omega_i, t), \quad (2.32b)$$

$$\frac{d\mathbf{Y}}{dt}(\omega_i, t) = -\mathbf{Y}(\omega_i, t)\mathbf{C}(t)^T + \left\langle \tilde{\mathcal{L}}u(x, t, \omega_i), \mathbf{U}(x, t) \right\rangle \mathbf{\Lambda}_{\mathbf{U}}(t)^{-1}, \quad (2.32c)$$

where $\mathbf{C}(t)$ and $\mathbf{D}(t)$ can be solved from (2.22) with

$$G_*(u, \mathbf{U}, \mathbf{Y}) = \mathbf{\Lambda}_{\mathbf{U}}^{-1} \left\langle \mathbf{U}^T, \frac{1}{N_r} \sum_{i=1}^{N_r} \tilde{\mathcal{L}}u(x, t, \omega_i)\mathbf{Y}(\omega_i, t) \right\rangle. \quad (2.33)$$

The above system is the MC version of DyBO method and we call it DyBO-MC. Close examinations reveal that eqn. (2.32a) for expectation and eqn. (2.32b) for spatial modes are only solved once for all the realizations at each time iteration while eqn. (2.32c) for stochastic modes is decoupled from realization to realization and can be solved simultaneously across all realizations.

Clearly, not only the number of realizations, but also how these realizations $\{\omega_i\}_{i=1}^{N_r}$ are chosen in Ω affects the numerical accuracy and the efficiency of the DyBO approximation to the solution of SPDE (1.1). qMC, variance reduction, and other techniques may be combined into DyBO-MC to further improve its efficiency and accuracy. Since such choice of realizations are generally problem-dependent and our main interests here are the discussion of a general methodology, we will make no further discussion on this topic.

2.3.2 Spectral Representation

2.3.2.1 Some Preliminaries of gPC Methods

In many physical and engineering problems, randomness generally comes from various independent sources, so randomness in SPDE (1.1) is often given in terms of independent random variables $\xi_i(\omega)$. Throughout this thesis, we assume only a finite number of independent random variables are involved, i.e., $\boldsymbol{\xi}(\omega) = (\xi_1(\omega), \xi_2(\omega), \dots, \xi_{N_p}(\omega))$, where N_p is the number of such random variables. Without loss of generality, we can further assume they all have identical distribution $\rho(\cdot)$. Thus, the solution of SPDE (1.1) is a functional of these random variables, $u(x, t, \omega) = u(x, t, \boldsymbol{\xi}(\omega))$. When SPDE is driven by some known stochastic process, such as Brownian Motion B_t , by Cameron-Martin theorem [19], the stochastic solution can be approximated by a functional of identical independent standard Gaussian variables, i.e., $u(x, t, \omega) \approx u(x, t, \boldsymbol{\xi}(\omega))$ with ξ_i 's being standard normal random variables.

We write $\{H_i(\xi)\}_{i=1}^{\infty}$, the one-dimensional polynomials orthogonal to each other with respect

to the common distribution $\rho(z)$, i.e.,

$$\int_{-\infty}^{\infty} H_i(\xi)H_j(\xi)\rho(\xi) d\xi = \delta_{ij}.$$

For some common distributions, such as Gaussian, uniform, and Poisson, such polynomial sets are well-known and well-studied, many of which fall in the Ashley scheme (see [117, 98, 5, 38] and references therein). For general distributions, such polynomial sets can be obtained by numerical methods (see [111] and references therein). Furthermore, by a tensor product representation, we can use the one-dimensional polynomial $H_i(\xi)$ to construct a complete orthonormal basis $\mathbf{H}_\alpha(\boldsymbol{\xi})$'s of $\mathbb{L}^2(\Omega)$ as follows

$$\mathbf{H}_\alpha(\boldsymbol{\xi}) = \prod_{i=1}^{N_p} H_{\alpha_i}(\xi_i), \quad \alpha \in \mathfrak{J}_{N_p}^\infty,$$

where α is a multi-index, i.e., a row vector of non-negative integers,

$$\alpha = (\alpha_1, \alpha_2, \dots, \alpha_{N_p}) \quad \alpha_i \in \mathbb{N}, \alpha_i \geq 0, i = 1, 2, \dots, N_p,$$

and $\mathfrak{J}_{N_p}^\infty$ is a multi-index set of countable cardinality,

$$\mathfrak{J}_{N_p}^\infty = \{\alpha = (\alpha_1, \alpha_2, \dots, \alpha_{N_p}) \mid \alpha_i \geq 0, \alpha_i \in \mathbb{N}\} \setminus \{\mathbf{0}\},$$

We have intentionally removed the zero multi-index corresponding to $\mathbf{H}_\mathbf{0}(\boldsymbol{\xi}) = 1$ since it is associated with the expectation of the stochastic solution and it is better to deal with it separately according to our experience. When no ambiguity arises, we simply write the multi-index set $\mathfrak{J}_{N_p}^\infty$ as \mathfrak{J}^∞ . Clearly, the cardinality of $\mathfrak{J}_{N_p}^\infty$ is infinite. For the purpose of numerical computations, we prefer a polynomial set of finite size. There are many choices of truncations, such as the set of

polynomials whose total orders are at most P , i.e.,

$$\mathfrak{J}_{N_p}^P = \left\{ \boldsymbol{\alpha} \mid \boldsymbol{\alpha} = (\alpha_1, \alpha_2, \dots, \alpha_{N_p}), \alpha_i \geq 0, \alpha_i \in \mathbb{N}, |\boldsymbol{\alpha}| = \sum_{i=1}^{N_p} \alpha_i \leq P \right\} \setminus \{\mathbf{0}\},$$

and sparse truncations proposed in Luo's thesis [77]. Again, we may simply write such truncated set as \mathfrak{J} when no ambiguity arises. The cardinality of \mathfrak{J} , or the number of polynomial chaos basis functions, is denoted as $N_P = |\mathfrak{J}| < \infty$.

By Cameron-Martin theorem [19], we know that the solution of SPDE (1.1) admits a generalized Polynomial Chaos expansion (gPCE)

$$u(x, t, \omega) = \bar{v}(x, t) + \sum_{\boldsymbol{\alpha} \in \mathfrak{J}^\infty} v_{\boldsymbol{\alpha}}(x, t) \mathbf{H}_{\boldsymbol{\alpha}}(\boldsymbol{\xi}(\omega)) \approx \bar{v}(x, t) + \sum_{\boldsymbol{\alpha} \in \mathfrak{J}} v_{\boldsymbol{\alpha}}(x, t) \mathbf{H}_{\boldsymbol{\alpha}}(\boldsymbol{\xi}(\omega)). \quad (2.34)$$

If we write

$$\begin{aligned} \mathbf{H}_{\mathfrak{J}}(\boldsymbol{\xi}) &= \left(\mathbf{H}_{\boldsymbol{\alpha}_1}(\boldsymbol{\xi}), \mathbf{H}_{\boldsymbol{\alpha}_2}(\boldsymbol{\xi}), \dots, \mathbf{H}_{\boldsymbol{\alpha}_{N_P}}(\boldsymbol{\xi}) \right)_{\boldsymbol{\alpha}_i \in \mathfrak{J}}, \\ \mathbf{V}(x, t) &= \left(v_{\boldsymbol{\alpha}_1}(x, t), v_{\boldsymbol{\alpha}_2}(x, t), \dots, v_{\boldsymbol{\alpha}_{N_P}}(x, t) \right)_{\boldsymbol{\alpha}_i \in \mathfrak{J}}, \end{aligned}$$

both of which are row vectors, the above gPCE can be compactly written in a vector form

$$u^{\text{gPC}}(x, t, \omega) = v(x, t, \omega) = \bar{v}(x, t) + \mathbf{V}(x, t) \mathbf{H}(\boldsymbol{\xi})^T. \quad (2.35)$$

By substituting the above gPCE into eqn. (1.1a), it is easy to get the gPC formulation for SPDE

(1.1) (see Appendix A for details),

$$\frac{\partial \bar{v}}{\partial t} = \mathbb{E}[\mathcal{L}v], \quad (2.36a)$$

$$\frac{\partial \mathbf{V}}{\partial t} = \mathbb{E}[\tilde{\mathcal{L}}v\mathbf{H}]. \quad (2.36b)$$

Remark 2.6. Throughout the thesis, we only consider SPDEs where randomness is given in terms of independent standard Gaussian random variables, i.e., $\xi_i \sim \mathcal{N}(0, 1)$ and \mathbf{H} are a set of Hermite polynomials. When the distribution of ξ_i is not Gaussian, the discussions remain almost identical and the major difference is the set of orthogonal polynomials.

2.3.2.2 gPC Version of DyBO

The Cameron-Martin theorem also implies the stochastic modes $Y_i(\omega, t)$'s in the KL expansion (2.5) can be approximated by the linear combination of polynomials chaos, i.e.,

$$Y_i(\omega, t) = \sum_{\alpha \in \mathfrak{J}} \mathbf{H}_\alpha(\boldsymbol{\xi}(\omega)) A_{\alpha i}(t), \quad i = 1, 2, \dots, m, \quad (2.37)$$

or in a matrix form,

$$\mathbf{Y}(\omega, t) = \mathbf{H}(\boldsymbol{\xi}(\omega)) \mathbf{A}, \quad (2.38)$$

where $\mathbf{A} \in \mathbb{R}^{N_p \times m}$. The expansion (2.5) now reads

$$u \approx \bar{u} + \mathbf{U} \mathbf{A}^T \mathbf{H}^T.$$

We can derive equations for \bar{u} , \mathbf{U} and \mathbf{A} , instead of \bar{u} , \mathbf{U} and \mathbf{Y} . In other words, the stochastic modes \mathbf{Y} are identified with a matrix $\mathbf{A} \in \mathbb{R}^{m \times m}$. Substituting eqn. (2.38) into eqn. (2.28c), we

have

$$\frac{d\mathbf{Y}}{dt} = \mathbf{H} \frac{d\mathbf{A}}{dt} = -\mathbf{H}\mathbf{A}\mathbf{C}^T + \langle \tilde{\mathcal{L}}u, \mathbf{U} \rangle \Lambda_{\mathbf{U}}^{-1}.$$

Multiply both sides by column vector \mathbf{H}^T from the left and take expectations,

$$\mathbb{E} [\mathbf{H}^T \mathbf{H}] \frac{d\mathbf{A}}{dt} = -\mathbb{E} [\mathbf{H}^T \mathbf{H}] \mathbf{A}\mathbf{C}^T + \langle \mathbb{E} [\mathbf{H}^T \tilde{\mathcal{L}}u], \mathbf{U} \rangle \Lambda_{\mathbf{U}}^{-1}.$$

After using the orthogonality of the polynomial chaos basis, i.e., $\mathbb{E} [\mathbf{H}^T \mathbf{H}] = \mathbf{I}$, we get

$$\frac{d\mathbf{A}}{dt} = -\mathbf{A}\mathbf{C}^T + \langle \mathbb{E} [\mathbf{H}^T \tilde{\mathcal{L}}u], \mathbf{U} \rangle \Lambda_{\mathbf{U}}^{-1}.$$

The gPC version of DyBO formulation for SPDE (1.1), which we call DyBO-gPC, is

$$\frac{\partial \bar{u}}{\partial t} = \mathbb{E} [\mathcal{L}u], \quad (2.39a)$$

$$\frac{\partial \mathbf{U}}{\partial t} = -\mathbf{U}\mathbf{D}^T + \mathbb{E} [\tilde{\mathcal{L}}u\mathbf{H}] \mathbf{A}, \quad (2.39b)$$

$$\frac{d\mathbf{A}}{dt} = -\mathbf{A}\mathbf{C}^T + \langle \mathbb{E} [\mathbf{H}^T \tilde{\mathcal{L}}u], \mathbf{U} \rangle \Lambda_{\mathbf{U}}^{-1}, \quad (2.39c)$$

where $\mathbf{C}(t)$ and $\mathbf{D}(t)$ can be solved from (2.22) with

$$G_*(u, \mathbf{U}, \mathbf{Y}) = \Lambda_{\mathbf{U}}^{-1} \langle \mathbf{U}^T, \mathbb{E} [\tilde{\mathcal{L}}u\mathbf{Y}] \rangle = \Lambda_{\mathbf{U}}^{-1} \langle \mathbf{U}^T, \mathbb{E} [\tilde{\mathcal{L}}u\mathbf{H}] \rangle \mathbf{A}. \quad (2.40)$$

By solving the system (2.39), we have an approximate solution to SPDE (1.1)

$$u^{\text{DyBO-gPC}} = \bar{u} + \mathbf{U}\mathbf{A}^T \mathbf{H}^T,$$

or simply u^{DyBO} or u when no ambiguity arises.

Remark 2.7. In DyBO-gPC formulation, $\mathbf{Y} \in \mathbb{R}^m$ is replaced by an N_P -by- m matrix. The stochastic modes \mathbf{Y} being orthonormal are translated to the columns of \mathbf{A} being orthonormal vectors, i.e., $\mathbf{A}^T \mathbf{A} = \mathbf{I} \in \mathbb{R}^{m \times m}$. Note that $\mathbf{A} \mathbf{A}^T \in \mathbb{R}^{N_P \times N_P}$ in general is *not* an identity matrix. When $m = N_P$, $\mathbf{A} \mathbf{A}^T = \mathbf{I}$. However, $m \ll N_P$ throughout the thesis since the SPDE solution is assumed to have a low-dimensional structure.

2.3.3 gSC Version of DyBO

In our DyBO-gPC method, the stochastic process $Y_i(\boldsymbol{\xi}(\omega), t)$ is projected on to the gPC basis \mathbf{H} and replaced by gPC coefficients, i.e., $A_{\alpha i}$, which can be computed by

$$A_{\alpha i} = \mathbb{E} [Y_i(\omega) \mathbf{H}_{\alpha}(\boldsymbol{\xi}(\omega))].$$

Numerically, the above integral can be evaluated with high accuracy on some sparse grid. In other words, the stochastic modes \mathbf{Y} can also be represented as an ensemble of realizations $\mathbf{Y}(\omega_i)$ where $\boldsymbol{\xi}(\omega_i)$'s are nodes from some sparse grid and associated with certain weight w_i . The gSC version of DyBO formulation, which we call DyBO-gSC, is

$$\begin{aligned} \frac{\partial \bar{u}}{\partial t}(x, t) &= \sum_{i=1}^{N_r} w_i \mathcal{L}u(x, t, \omega_i), \\ \frac{\partial \mathbf{U}}{\partial t}(x, t) &= -\mathbf{U}(x, t) \mathbf{D}(t)^T + \sum_{i=1}^{N_r} w_i \tilde{\mathcal{L}}u(x, t, \omega_i) \mathbf{Y}(\omega_i, t), \\ \frac{d\mathbf{Y}}{dt}(\omega_i, t) &= -\mathbf{Y}(\omega_i, t) \mathbf{C}(t)^T + \left\langle \tilde{\mathcal{L}}u(x, t, \omega_i), \mathbf{U}(x, t) \right\rangle \boldsymbol{\Lambda}_{\mathbf{U}}(t)^{-1}, \quad i = 1, 2, \dots, N_r \end{aligned}$$

where $\mathbf{C}(t)$ and $\mathbf{D}(t)$ can be solved from (2.22) with

$$G_*(u, \mathbf{U}, \mathbf{Y}) = \boldsymbol{\Lambda}_{\mathbf{U}}^{-1} \left\langle \mathbf{U}^T, \sum_{i=1}^{N_r} w_i \tilde{\mathcal{L}}u(x, t, \omega_i) \mathbf{Y}(\omega_i, t) \right\rangle. \quad (2.42)$$

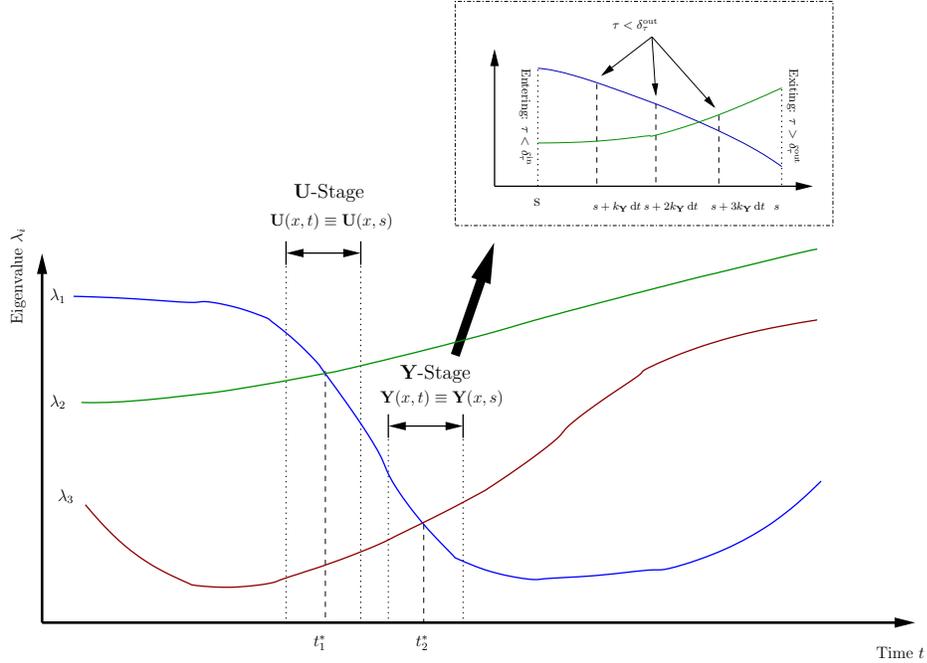


Figure 2.1: Illustration of eigenvalue crossing and two strategies, **Y-Stage** and **U-Stage**

We defer further discussion and numerical examples until Chapter 7.

2.4 Eigenvalue Crossing

As the system evolves, eigenvalues of different modes in the KL expansion of the SPDE solution may increase or decrease. Some of them may approach each other at some time, cross and then separate as illustrated in Fig. 2.1. In the figure, λ_1 and λ_2 cross each other at t_1^* and λ_1 and λ_3 cross each other at t_2^* . In this case, if **C** and **D** continue to be solved from the linear system (2.21) via (2.22), numerical errors will render the results useless. Here, we propose to freeze **U** or **Y** temporarily for a short time and continue to evolve the system using different equations as derived below. At the end of this short duration, the solution is recast into the bi-orthogonal form via KL expansion, which can be achieved efficiently since one of **U** and **Y** is still kept orthogonal in this short duration.

2.4.1 Detection of Eigenvalue Crossing

In order to apply the above strategy, we have to be able to detect that two eigenvalues may potentially cross each other in the near future. There are multiple ways to detect such crossing. Here we propose to monitor the following quantity

$$\tau = \min_{i \neq j} \frac{|\lambda_i - \lambda_j|}{\max(\lambda_i, \lambda_j)}. \quad (2.43)$$

Once this quantity drops below certain threshold $\delta_\tau^{\text{in}} \in (0, 1)$, the algorithm freezes \mathbf{U} or \mathbf{Y} , continues to evolve the system and monitor this quantity τ . When τ exceeds some pre-specified threshold $\delta_\tau^{\text{out}} \in (0, 1)$, the algorithm recasts the solution in the bi-orthogonal form via efficient algorithms detailed in the next two sub-sections and continues to evolve DyBO system. Since the computation of eigenvalues involves some non-trivial numerical operations, the quantity τ can be monitored periodically, i.e., every several time iterations, instead of at each time iteration, in favor of the balance of computational efficiency and accuracy. Threshold δ_τ^{in} and δ_τ^{out} may be problem-dependent. In our numerical experiments, we found $\delta_\tau^{\text{in}} = 1\%$ and $\delta_\tau^{\text{out}} = 1\%$ gave accurate results.

2.4.2 Freeze \mathbf{Y} or \mathbf{Y} -Stage

Suppose at some time $t = s$, potential eigenvalue crossing is detected and the stochastic modes \mathbf{Y} are frozen for a short duration Δs , i.e., $\mathbf{Y}(\omega, t) \equiv \mathbf{Y}(\omega, s)$ for $t \in [s, s + \Delta s]$, which we call \mathbf{Y} -stage. Because \mathbf{Y} are orthonormal, the solution of SPDE (1.1) admits the following approximation of truncated expansions

$$u(x, t, \omega) \approx \bar{u}(x, t) + \sum_{i=1}^m u_i(x, t) Y_i(\omega, s) = \bar{u}(x, t) + \mathbf{U}(x, t) \mathbf{Y}(\omega, s)^T. \quad (2.44)$$

It is easy to derive the new system for this stage,

$$\frac{\partial \bar{u}}{\partial t} = \mathbb{E}[\mathcal{L}u], \quad (2.45a)$$

$$\frac{\partial \mathbf{U}}{\partial t} = \mathbb{E}[\tilde{\mathcal{L}}u\mathbf{Y}]. \quad (2.45b)$$

During this stage, \mathbf{Y} is unchanged and orthogonal, but not \mathbf{U} . The solution is not bi-orthogonal any more, so eigenvalues cannot be computed via $\lambda_i = \|u_i\|_{\mathbb{L}^2(\mathcal{D})}^2$ any more. Next, we derive formula for eigenvalues in this stage and then show how the solutions are recast into the bi-orthogonal form at the end of the stage, i.e., $t = s + \Delta s$.

The covariance function can be computed as

$$\begin{aligned} \text{Cov}_u(x, y) &= \mathbb{E}[(u(x) - \bar{u}(x))(u(y) - \bar{u}(y))] \\ &= \mathbb{E}[\mathbf{U}(x)\mathbf{Y}^T\mathbf{Y}\mathbf{U}^T(y)] \\ &= \mathbf{U}(x)\mathbb{E}[\mathbf{Y}^T\mathbf{Y}]\mathbf{U}^T(y) \\ &= \mathbf{U}(x)\mathbf{U}^T(y), \end{aligned}$$

where t is omitted for simplicity and $\mathbb{E}[\mathbf{Y}^T\mathbf{Y}] = \mathbf{I}$ is used in the second to last line. By some stable orthogonalization procedures, such as a modified Gram-Schmidt process [44], $\mathbf{U}(x)$ can be written as

$$\mathbf{U}(x) = \mathbf{Q}(x)\mathbf{R}, \quad (2.46)$$

where $\mathbf{Q}(x) = (q_1(x), q_2(x), \dots, q_m(x))$, $q_i(x) \in \mathbb{L}^2(\mathcal{D})$ for $i = 1, 2, \dots, m$, $\langle \mathbf{Q}^T(x), \mathbf{Q}(x) \rangle = \mathbf{I}$ and $\mathbf{R} \in \mathbb{R}^{m \times m}$. $\mathbf{R}\mathbf{R}^T \in \mathbb{R}^{m \times m}$ is a positive definite symmetric matrix and its SVD decomposition

$$\mathbf{R}\mathbf{R}^T = \mathbf{W}\mathbf{\Lambda}_R\mathbf{W}^T, \quad (2.47)$$

where $\mathbf{W} \in \mathbb{R}^{m \times m}$ is an orthonormal matrix, i.e., $\mathbf{W}\mathbf{W}^T = \mathbf{W}^T\mathbf{W} = \mathbf{I}$, and $\Lambda_{\mathbf{R}}$ is a diagonal matrix with positive diagonal entries. The computational cost of \mathbf{W} and $\Lambda_{\mathbf{R}}$ is negligible compared to other parts of the algorithm. The covariance function can be rewritten as

$$\text{Cov}_u(x, y) = \mathbf{Q}(x)\mathbf{R}\mathbf{R}^T\mathbf{Q}^T(y) = \mathbf{Q}(x)\mathbf{W}\Lambda_{\mathbf{R}}\mathbf{W}^T\mathbf{Q}^T(y).$$

Now, it is easy to see that the eigenfunctions of covariance function $\text{Cov}_u(x, y)$ are

$$\tilde{\mathbf{U}}(x) = \mathbf{Q}(x)\mathbf{W}\Lambda_{\mathbf{R}}^{\frac{1}{2}}, \quad (2.48)$$

and eigenvalues are $\text{diag}(\Lambda_{\mathbf{R}})$ because

$$\begin{aligned} \langle \tilde{\mathbf{U}}^T, \tilde{\mathbf{U}} \rangle &= \left\langle \Lambda_{\mathbf{R}}^{\frac{1}{2}}\mathbf{W}^T\mathbf{Q}^T, \mathbf{Q}\mathbf{W}\Lambda_{\mathbf{R}}^{\frac{1}{2}} \right\rangle \\ &= \Lambda_{\mathbf{R}}^{\frac{1}{2}}\mathbf{W}^T \langle \mathbf{Q}^T, \mathbf{Q} \rangle \mathbf{W}\Lambda_{\mathbf{R}}^{\frac{1}{2}} \\ &= \Lambda_{\mathbf{R}}, \end{aligned}$$

where we have use the orthogonality of \mathbf{Q} and \mathbf{W} . To compute $\tilde{\mathbf{Y}}$, we start with the identity

$$\tilde{\mathbf{Y}}\tilde{\mathbf{U}}^T = \mathbf{Y}\mathbf{U}^T,$$

and multiply row vector $\tilde{\mathbf{U}}$ on both sides from the right and take inner product $\langle \cdot, \cdot \rangle$,

$$\tilde{\mathbf{Y}} \langle \tilde{\mathbf{U}}^T, \tilde{\mathbf{U}} \rangle = \mathbf{Y} \langle \mathbf{U}^T, \tilde{\mathbf{U}} \rangle,$$

where

$$\begin{aligned}\langle \tilde{\mathbf{U}}^T, \tilde{\mathbf{U}} \rangle &= \mathbf{\Lambda}_{\mathbf{R}}, \\ \langle \mathbf{U}^T, \tilde{\mathbf{U}} \rangle &= \left\langle \mathbf{R}^T \mathbf{Q}^T(x), \mathbf{Q}(x) \mathbf{W} \mathbf{\Lambda}_{\mathbf{R}}^{\frac{1}{2}} \right\rangle = \mathbf{R}^T \mathbf{W} \mathbf{\Lambda}_{\mathbf{R}}^{\frac{1}{2}}.\end{aligned}$$

So we have the equation for $\tilde{\mathbf{Y}}$

$$\tilde{\mathbf{Y}} = \mathbf{Y} \mathbf{R}^T \mathbf{W} \mathbf{\Lambda}_{\mathbf{R}}^{-\frac{1}{2}}. \quad (2.49)$$

Algorithm 2.1 Strategy in \mathbf{Y} -Stage

- 1: Compute τ via eqn. (2.43) at time $t = s$
 - 2: **if** $\tau \geq \delta_{\tau}^{\text{in}}$ **then**
 - 3: Continue to evolve the system according to (2.28).
 - 4: **else**
 - 5: Enter \mathbf{Y} -Stage by $\mathbf{Y}(x, t) \equiv \mathbf{Y}(x, s)$
 - 6: $i \leftarrow 1$
 - 7: **repeat**
 - 8: Continue to evolve the system from $t = s + (i - 1)k_{\mathbf{Y}}\delta t$ to $t = s + ik_{\mathbf{Y}}\delta t$ according to (2.45).
 - 9: Compute eigenvalues via eqn. (2.47) and τ
 - 10: **until** $\tau > \delta_{\tau}^{\text{out}}$
 - 11: Recast the solution to bi-orthogonal form via eqn. (2.48) and eqn. (2.49).
 - 12: **end if**
-

If a generalized polynomial basis \mathbf{H} is adopted to represent \mathbf{Y} , we freeze \mathbf{A} instead, i.e., $\mathbf{A}(t) \equiv \mathbf{A}(s)$ and the system (2.45) is replaced by

$$\frac{\partial \bar{u}}{\partial t} = \mathbb{E}[\mathcal{L}u], \quad (2.50a)$$

$$\frac{\partial \mathbf{U}}{\partial t} = \mathbb{E}[\tilde{\mathcal{L}}u\mathbf{H}] \mathbf{A}, \quad (2.50b)$$

where $\mathbf{A}(t) \equiv \mathbf{A}(s)$. At the exiting point, eqn. (2.49) is replaced by

$$\tilde{\mathbf{A}} = \mathbf{A}\mathbf{R}^T\mathbf{W}\mathbf{\Lambda}_{\mathbf{R}}^{-\frac{1}{2}}. \quad (2.51)$$

As we can see, the computation of eigenvalues in this stage is not trivial, so we do not want to compute τ every time iteration. Instead, τ is only evaluated every $k_{\mathbf{Y}}$ time steps to achieve a balance between computational efficiency and accuracy. The strategy is summarized in Algorithm 2.1. See the zoom-out figure in Fig. 2.1 for illustration.

2.4.3 Freeze \mathbf{U} or \mathbf{U} -Stage

Similarly, we consider the strategy of freezing \mathbf{U} or \mathbf{U} -stage. Suppose at some time $t = s$, the spatial modes \mathbf{U} are frozen, i.e., $\mathbf{U}(x, t) \equiv \mathbf{U}(x, s)$ for $t \in [s, s + \Delta s]$. Instead of approximation (2.44), we have

$$u(x, t, \omega) \approx \bar{u}(x, t) + \sum_{i=1}^m u_i(x, s) Y_i(\omega, t) = \bar{u}(x, t) + \mathbf{U}(x, s) \mathbf{Y}(\omega, t)^T, \quad (2.52)$$

which gives the new system for this stage,

$$\frac{\partial \bar{u}}{\partial t} = \mathbb{E}[\mathcal{L}u], \quad (2.53a)$$

$$\frac{d\mathbf{Y}}{dt} = \langle \tilde{\mathcal{L}}u, \mathbf{U} \rangle \mathbf{\Lambda}_{\mathbf{U}}^{-1}. \quad (2.53b)$$

During this stage, \mathbf{U} is orthogonal, but not \mathbf{Y} , so eigenvalues cannot be computed via $\lambda_i = \|u_i\|_{\mathbb{L}^2(\mathcal{D})}^2$ any more. We follow a similar procedure to derive eigenvalues and the new bi-orthogonal

form at the end of the stage. Again, start from the covariance function

$$\begin{aligned}
\text{Cov}_u(x, y) &= \mathbb{E} [(u(x) - \bar{u}(x)) (u(y) - \bar{u}(y))] \\
&= \mathbb{E} [\mathbf{U}(x) \mathbf{Y}^T \mathbf{Y} \mathbf{U}^T(y)] \\
&= \mathbf{U}(x) \mathbb{E} [\mathbf{Y}^T \mathbf{Y}] \mathbf{U}^T(y) \\
&= \mathbf{U}(x) \Lambda_{\mathbf{U}}^{-\frac{1}{2}} \underbrace{\Lambda_{\mathbf{U}}^{+\frac{1}{2}} \mathbb{E} [\mathbf{Y}^T \mathbf{Y}] \Lambda_{\mathbf{U}}^{+\frac{1}{2}}}_{\mathbf{R}} \Lambda_{\mathbf{U}}^{-\frac{1}{2}} \mathbf{U}^T(y) \\
&= \mathbf{U}(x) \Lambda_{\mathbf{U}}^{-\frac{1}{2}} \mathbf{W} \Lambda_{\mathbf{R}} \mathbf{W}^T \Lambda_{\mathbf{U}}^{-\frac{1}{2}} \mathbf{U}^T(y),
\end{aligned}$$

where

$$\mathbf{R} = \Lambda_{\mathbf{U}}^{+\frac{1}{2}} \mathbb{E} [\mathbf{Y}^T \mathbf{Y}] \Lambda_{\mathbf{U}}^{+\frac{1}{2}},$$

and its SVD decomposition

$$\mathbf{R} = \mathbf{W} \Lambda_{\mathbf{R}} \mathbf{W}^T. \quad (2.54)$$

It is easy to see that the eigenfunctions of covariance function $\text{Cov}_u(x, y)$ are

$$\tilde{\mathbf{U}}(x) = \mathbf{U}(x) \Lambda_{\mathbf{U}}^{-\frac{1}{2}} \mathbf{W} \Lambda_{\mathbf{R}}^{+\frac{1}{2}}, \quad (2.55)$$

and eigenvalues are $\text{diag}(\Lambda_{\mathbf{R}})$ because

$$\begin{aligned}
\langle \tilde{\mathbf{U}}^T, \tilde{\mathbf{U}} \rangle &= \left\langle \Lambda_{\mathbf{R}}^{+\frac{1}{2}} \mathbf{W}^T \Lambda_{\mathbf{U}}^{-\frac{1}{2}} \mathbf{U}^T(x), \mathbf{U}(x) \Lambda_{\mathbf{U}}^{-\frac{1}{2}} \mathbf{W} \Lambda_{\mathbf{R}}^{+\frac{1}{2}} \right\rangle \\
&= \Lambda_{\mathbf{R}}^{+\frac{1}{2}} \mathbf{W}^T \Lambda_{\mathbf{U}}^{-\frac{1}{2}} \langle \mathbf{U}^T(x), \mathbf{U}(x) \rangle \Lambda_{\mathbf{U}}^{-\frac{1}{2}} \mathbf{W} \Lambda_{\mathbf{R}}^{+\frac{1}{2}} \\
&= \Lambda_{\mathbf{R}}^{+\frac{1}{2}} \mathbf{W}^T \Lambda_{\mathbf{U}}^{-\frac{1}{2}} \Lambda_{\mathbf{U}} \Lambda_{\mathbf{U}}^{-\frac{1}{2}} \mathbf{W} \Lambda_{\mathbf{R}}^{+\frac{1}{2}} \\
&= \Lambda_{\mathbf{R}}^{+\frac{1}{2}} \mathbf{W}^T \mathbf{W} \Lambda_{\mathbf{R}}^{+\frac{1}{2}} \\
&= \Lambda_{\mathbf{R}}.
\end{aligned}$$

To compute $\tilde{\mathbf{Y}}$, we start with the identity

$$\tilde{\mathbf{Y}} \tilde{\mathbf{U}}^T = \mathbf{Y} \mathbf{U}^T,$$

and multiply row vector $\tilde{\mathbf{U}}$ on both sides from the right and take inner product $\langle \cdot, \cdot \rangle$,

$$\tilde{\mathbf{Y}} \langle \tilde{\mathbf{U}}^T, \tilde{\mathbf{U}} \rangle = \mathbf{Y} \langle \mathbf{U}^T, \tilde{\mathbf{U}} \rangle,$$

where

$$\begin{aligned}
\langle \tilde{\mathbf{U}}^T, \tilde{\mathbf{U}} \rangle &= \Lambda_{\mathbf{R}}, \\
\langle \mathbf{U}^T, \tilde{\mathbf{U}} \rangle &= \left\langle \mathbf{U}^T, \mathbf{U} \Lambda_{\mathbf{U}}^{-\frac{1}{2}} \mathbf{W} \Lambda_{\mathbf{R}}^{+\frac{1}{2}} \right\rangle = \Lambda_{\mathbf{U}}^{+\frac{1}{2}} \mathbf{W} \Lambda_{\mathbf{R}}^{+\frac{1}{2}}.
\end{aligned}$$

We have the equation for $\tilde{\mathbf{Y}}$

$$\tilde{\mathbf{Y}} = \mathbf{Y} \Lambda_{\mathbf{U}}^{+\frac{1}{2}} \mathbf{W} \Lambda_{\mathbf{R}}^{-\frac{1}{2}}. \quad (2.56)$$

The algorithm in U-Stage is summarized in Algorithm 2.2. See the zoom-out figure in Fig. 2.1 for illustration.

Algorithm 2.2 Strategy in U-Stage

- 1: Compute τ via eqn. (2.43) at time $t = s$
 - 2: **if** $\tau \geq \delta_\tau^{\text{in}}$ **then**
 - 3: Continue to evolve the system according to (2.28).
 - 4: **else**
 - 5: Enter U-Stage by $\mathbf{U}(x, t) \equiv \mathbf{U}(x, s)$
 - 6: $i \leftarrow 1$
 - 7: **repeat**
 - 8: Continue to evolve the system from $t = s + (i - 1)k_{\mathbf{U}}\delta t$ to $t = s + ik_{\mathbf{U}}\delta t$ according to (2.53).
 - 9: Compute eigenvalues via eqn. (2.54) and τ
 - 10: **until** $\tau > \delta_\tau^{\text{out}}$
 - 11: Recast the solution to bi-orthogonal form via eqn. (2.55) and eqn. (2.56).
 - 12: **end if**
-

If a generalized polynomial basis \mathbf{H} is adopted to represent \mathbf{Y} , we can write system (2.55) as

$$\frac{\partial \bar{u}}{\partial t} = \mathbb{E}[\mathcal{L}u], \quad (2.57a)$$

$$\frac{d\mathbf{A}}{dt} = \left\langle \mathbb{E}[\tilde{\mathcal{L}}u\mathbf{H}^T], \mathbf{U} \right\rangle \mathbf{\Lambda}_{\mathbf{U}}^{-1}, \quad (2.57b)$$

where the last equation comes from substituting $\mathbf{Y} = \mathbf{H}\mathbf{A}$ into the last equation of (2.53), multiplying \mathbf{H}^T on both sides from the left and taking expectations. The definition of matrix \mathbf{R} is also replaced by

$$\mathbf{R} = \mathbf{\Lambda}_{\mathbf{U}}^{+\frac{1}{2}} \mathbb{E}[\mathbf{A}^T \mathbf{H}^T \mathbf{H} \mathbf{A}] \mathbf{\Lambda}_{\mathbf{U}}^{+\frac{1}{2}} = \mathbf{\Lambda}_{\mathbf{U}}^{+\frac{1}{2}} \mathbf{A}^T \mathbf{A} \mathbf{\Lambda}_{\mathbf{U}}^{+\frac{1}{2}},$$

and new $\tilde{\mathbf{A}}$ at the exiting point is

$$\tilde{\mathbf{A}} = \mathbf{A} \mathbf{\Lambda}_{\mathbf{U}}^{+\frac{1}{2}} \mathbf{W} \mathbf{\Lambda}_{\mathbf{R}}^{-\frac{1}{2}}. \quad (2.58)$$

2.5 Error Analysis

In this section, we focus on error analysis of DyBO formulation with respect to the original system (1.1) and consider the evolution of three types of errors,

Type-0 Errors between the true solution $u(x, t, \omega)$ and DyBO solution $u^{\text{DyBO}} = \bar{u} + \mathbf{U}\mathbf{Y}^T$, which is the error in the usual sense.

Type- \mathfrak{J} Errors between true gPC coefficients v_α and ones computed from DyBO-gPC solution, i.e., $\sum_{i=1}^m u_i A_{\alpha i}$.

Type-KLE Errors between true spatial modes $\sum_{\alpha \in \mathfrak{J}} v_\alpha A_{\alpha i}$'s and ones u_i 's computed in DyBO-gPC

When randomness comes into the system through initial conditions only and operator \mathcal{L} is linear and deterministic, the next theorem shows the error of DyBO is associated with the well-posedness of the original system. Interestingly, the last type of errors shed some lights on the proper strategy to add new spatial and stochastic modes when necessary, which will be discussed in detail in Sec. 2.6.

2.5.1 Type-0 Errors

Theorem 2.5 shows DyBO formulation is consistent with the original SPDE. For the sake of notation simplicity, we write u for u^{DyBO} here.

Theorem 2.5. *The stochastic solution of system (2.28) satisfies the following modified SPDE*

$$\frac{\partial u}{\partial t} = \mathcal{L}u + e_m, \quad (2.59)$$

where e_m is the error due to m -term truncation and

$$\begin{aligned} e_m &= -\Pi_{\mathbf{Y}}(\Pi_{\mathbf{U}}(\tilde{\mathcal{L}}u)) \\ &= -\left(\tilde{\mathcal{L}}u - \mathbf{U}\Lambda_{\mathbf{U}}^{-1}\langle\mathbf{U}^T, \tilde{\mathcal{L}}u\rangle\right) + \mathbb{E}\left[\left(\tilde{\mathcal{L}}u - \mathbf{U}\Lambda_{\mathbf{U}}^{-1}\langle\mathbf{U}^T, \tilde{\mathcal{L}}u\rangle\right)\mathbf{Y}\right]\mathbf{Y}^T. \end{aligned} \quad (2.60)$$

Proof of Consistency. We show consistency by computing directly from eqn. (2.28b) and eqn. (2.28c)

$$\begin{aligned} \mathbf{U}\frac{d\mathbf{Y}^T}{dt} &= \mathbf{U}\mathbf{D}^T\mathbf{Y}^T + \mathbf{U}G_{\mathbf{Y}}(u, \mathbf{U}, \mathbf{Y})^T \\ &= \mathbf{U}\mathbf{D}^T\mathbf{Y}^T + \mathbf{U}\Lambda_{\mathbf{U}}^{-1}\langle\mathbf{U}^T, \tilde{\mathcal{L}}u\rangle - \mathbf{U}\Lambda_{\mathbf{U}}^{-1}\langle\mathbf{U}^T, \mathbb{E}[\tilde{\mathcal{L}}u\mathbf{Y}]\rangle\mathbf{Y}^T \\ &= \tilde{\mathcal{L}}u + \mathbf{U}\mathbf{D}^T\mathbf{Y}^T - \left(\tilde{\mathcal{L}}u - \mathbf{U}\Lambda_{\mathbf{U}}^{-1}\langle\mathbf{U}^T, \tilde{\mathcal{L}}u\rangle\right) - \mathbf{U}\Lambda_{\mathbf{U}}^{-1}\langle\mathbf{U}^T, \mathbb{E}[\tilde{\mathcal{L}}u\mathbf{Y}]\rangle\mathbf{Y}^T, \\ \mathbf{Y}\frac{\partial\mathbf{U}^T}{\partial t} &= \mathbf{Y}\mathbf{C}^T\mathbf{U}^T + \mathbf{Y}G_{\mathbf{U}}(u, \mathbf{U}, \mathbf{Y})^T \\ &= \mathbf{U}\mathbf{C}\mathbf{Y}^T + \mathbf{Y}\mathbb{E}[\tilde{\mathcal{L}}u\mathbf{Y}^T] - \mathbf{Y}\langle\mathbb{E}[\tilde{\mathcal{L}}u\mathbf{Y}^T], \mathbf{U}\rangle\Lambda_{\mathbf{U}}^{-1}\mathbf{U}^T. \end{aligned}$$

Combining the above two, we have

$$\mathbf{U}\frac{d\mathbf{Y}^T}{dt} + \mathbf{Y}\frac{\partial\mathbf{U}^T}{\partial t} = \tilde{\mathcal{L}}u + e_m,$$

where

$$\begin{aligned} e_m &= -\left(\tilde{\mathcal{L}}u - \mathbf{U}\Lambda_{\mathbf{U}}^{-1}\langle\mathbf{U}^T, \tilde{\mathcal{L}}u\rangle\right) + \mathbf{U}(\mathbf{D}^T + \mathbf{C})\mathbf{Y}^T - v\mathbf{U}\Lambda_{\mathbf{U}}^{-1}\langle\mathbf{U}^T, \mathbb{E}[\tilde{\mathcal{L}}u\mathbf{Y}]\rangle\mathbf{Y}^T \\ &\quad + \mathbf{Y}\mathbb{E}[\tilde{\mathcal{L}}u\mathbf{Y}^T] - \mathbf{Y}\langle\mathbb{E}[\tilde{\mathcal{L}}u\mathbf{Y}^T], \mathbf{U}\rangle\Lambda_{\mathbf{U}}^{-1}\mathbf{U}^T \\ &= -\left(\tilde{\mathcal{L}}u - \mathbf{U}\Lambda_{\mathbf{U}}^{-1}\langle\mathbf{U}^T, \tilde{\mathcal{L}}u\rangle\right) + \mathbf{Y}\mathbb{E}[\tilde{\mathcal{L}}u\mathbf{Y}^T] - \mathbf{U}\Lambda_{\mathbf{U}}^{-1}\langle\mathbf{U}^T, \mathbb{E}[\tilde{\mathcal{L}}u\mathbf{Y}]\rangle\mathbf{Y}^T \\ &= -\left(\tilde{\mathcal{L}}u - \mathbf{U}\Lambda_{\mathbf{U}}^{-1}\langle\mathbf{U}^T, \tilde{\mathcal{L}}u\rangle\right) + \mathbb{E}[\tilde{\mathcal{L}}u\mathbf{Y}]\mathbf{Y}^T - \mathbb{E}[\mathbf{U}\Lambda_{\mathbf{U}}^{-1}\langle\mathbf{U}^T, \tilde{\mathcal{L}}u\rangle\mathbf{Y}]\mathbf{Y}^T \\ &= -\left(\tilde{\mathcal{L}}u - \mathbf{U}\Lambda_{\mathbf{U}}^{-1}\langle\mathbf{U}^T, \tilde{\mathcal{L}}u\rangle\right) + \mathbb{E}\left[\left(\tilde{\mathcal{L}}u - \mathbf{U}\Lambda_{\mathbf{U}}^{-1}\langle\mathbf{U}^T, \tilde{\mathcal{L}}u\rangle\right)\mathbf{Y}\right]\mathbf{Y}^T, \end{aligned}$$

where eqn. (2.21c) is used in the first equality. By noticing $\tilde{\mathcal{L}}u = \mathcal{L}u - \mathbb{E}[\mathcal{L}u]$, eqn. (2.59) is recovered from the last line. \square

Remark 2.8. According to Theorem 2.3, the bi-orthogonality of \mathbf{U} and \mathbf{Y} are preserved all the time. Because both Hilbert space $\mathbb{L}^2(\mathcal{D})$ and $\mathbb{L}^2(\Omega)$ are separable, the spatial modes \mathbf{U} and the stochastic modes \mathbf{Y} become a complete set of basis for $\mathbb{L}^2(\mathcal{D})$ and $\mathbb{L}^2(\Omega)$ as $m \rightarrow +\infty$, respectively, which implies $\lim_{m \rightarrow +\infty} e_m = 0$.

Next we consider a special case where the differential operator \mathcal{L} is deterministic and linear, such as $\mathcal{L}v = c(x)\frac{\partial v}{\partial x}$, $\mathcal{L}v = -\frac{\partial}{\partial x_i} \left(a_{ij}(x) \frac{\partial v}{\partial x_j} \right)$ where $c(x)$ and $a(x)$ are deterministic and only functions of spatial coordinates x . Only initial conditions are assumed to be random, i.e., the randomness propagates into the system only through the initial conditions.

Corollary 2.6. *Let the differential operator \mathcal{L} be deterministic and linear. The residual in eqn. (2.59) is zero all the time, i.e.,*

$$e_m \equiv 0.$$

Proof. This can be seen by directly computing the truncation error e_m . First substitute $\tilde{\mathcal{L}}u = \mathcal{L}u - \mathbb{E}[\mathcal{L}u]$ into eqn. (2.60),

$$e_m = -\Pi_{\mathbf{Y}} \left(\mathcal{L}u - \mathbb{E}[\mathcal{L}u] - \mathbf{U}\mathbf{\Lambda}_{\mathbf{U}}^{-1} \langle \mathbf{U}^T, \mathcal{L}u - \mathbb{E}[\mathcal{L}u] \rangle \right).$$

Because of the linearity of \mathcal{L} , we have $\mathcal{L}u = \mathcal{L}\bar{u} + \mathcal{L}\mathbf{U}\mathbf{Y}^T$ where $\mathcal{L}\mathbf{U} = (\mathcal{L}u_1, \mathcal{L}u_2, \dots, \mathcal{L}u_m)$.

which gives

$$\begin{aligned}
e_m &= -\Pi_{\mathbf{Y}} (\mathcal{L}\bar{u} + \mathcal{L}\mathbf{U}\mathbf{Y}^T - \mathbb{E} [\mathcal{L}\bar{u} + \mathcal{L}\mathbf{U}\mathbf{Y}^T] - \mathbf{U}\mathbf{\Lambda}_{\mathbf{U}}^{-1} \langle \mathbf{U}^T, \mathcal{L}u - \mathbb{E} [\mathcal{L}u] \rangle) \\
&= -\Pi_{\mathbf{Y}} (\mathcal{L}\mathbf{U}\mathbf{Y}^T - \mathbf{U}\mathbf{\Lambda}_{\mathbf{U}}^{-1} \langle \mathbf{U}^T, \mathcal{L}\mathbf{U} \rangle \mathbf{Y}^T) \\
&= -\Pi_{\mathbf{Y}} ((\mathcal{L}\mathbf{U} - \mathbf{U}\mathbf{\Lambda}_{\mathbf{U}}^{-1} \langle \mathbf{U}^T, \mathcal{L}\mathbf{U} \rangle) \mathbf{Y}^T).
\end{aligned}$$

The last line is the orthogonal complementary project, so we see immediately that $e_m = 0$. \square

The above corollary implies that DyBO is exact if the randomness can be expressed in finite-term KL expansion. The next corollary concerns a slightly different case where the differential operator \mathcal{L} is affine in the sense that $\mathcal{L}u = \mathring{\mathcal{L}}u + f(x, t, \omega)$ and the differential operator $\mathring{\mathcal{L}}$ is linear and deterministic and $f(\cdot, t, \cdot) \in \mathbb{L}^2(\mathcal{D} \times \Omega)$ for all t .

Corollary 2.7. *If the differential operator \mathcal{L} is affine, i.e., $\mathcal{L}u = \mathring{\mathcal{L}}u + f(x, t, \omega)$, and f is a second-order stochastic process at each fixed time t , the residual in eqn. (2.60) is given below*

$$e_m = -\Pi_{\mathbf{Y}} \Pi_{\mathbf{U}}(f).$$

Proof. Again by directly computing, we have by the linearity of the differential operator $\mathring{\mathcal{L}}$

$$\mathcal{L}u = \mathring{\mathcal{L}}\bar{u} + \mathring{\mathcal{L}}\mathbf{U}\mathbf{Y}^T + f.$$

Simple calculation shows that

$$\begin{aligned}
 \tilde{\mathcal{L}}u &= \mathcal{L}u - \mathbb{E}[\mathcal{L}u] \\
 &= \mathring{\mathcal{L}}\bar{u} + \mathring{\mathcal{L}}\mathbf{U}\mathbf{Y}^T + f - \mathring{\mathcal{L}}\bar{u} - \mathbb{E}[f] \\
 &= \mathring{\mathcal{L}}\mathbf{U}\mathbf{Y}^T + f - \mathbb{E}[f].
 \end{aligned}$$

Substituting into eqn. (2.60), we complete the proof. \square

Remark 2.9. For this special case, Corollary 2.7 implies that numerical solutions are accurate as long as the spatial basis \mathbf{U} and stochastic basis \mathbf{Y} provide good approximations to the external forcing term f , which is not a surprising result at all.

2.5.2 Type- \mathfrak{J} Error

Next, we focus on the gPC and DyBO-gPC formulations and consider two types of errors. First, we consider the errors between true gPC coefficients v_α and ones computed from the DyBO-gPC solution, i.e.,

$$\bar{\epsilon} = \bar{u} - \bar{v}, \tag{2.61a}$$

$$\epsilon^{\mathfrak{J}} = \mathbf{U}\mathbf{A}^T - \mathbf{V}. \tag{2.61b}$$

When no ambiguity arises, we simply write $\epsilon = \epsilon^{\mathfrak{J}}$. To derive the error evolution equations, we start with

$$\begin{aligned}
\frac{\partial(\mathbf{U}\mathbf{A}^T)}{\partial t} &= \frac{\partial\mathbf{U}}{\partial t}\mathbf{A}^T + \mathbf{U}\frac{d\mathbf{A}^T}{dt} \\
&= -\mathbf{U}\mathbf{D}^T\mathbf{A}^T + \mathbb{E}\left[\tilde{\mathcal{L}}u\mathbf{H}\right]\mathbf{A}\mathbf{A}^T - \mathbf{U}\mathbf{C}\mathbf{A}^T + \mathbf{U}\Lambda_{\mathbf{U}}^{-1}\left\langle\mathbf{U}^T, \mathbb{E}\left[\tilde{\mathcal{L}}u\mathbf{H}\right]\right\rangle \\
&= \mathbb{E}\left[\tilde{\mathcal{L}}u\mathbf{H}\right]\mathbf{A}\mathbf{A}^T - \mathbf{U}\left(\mathbf{D}^T + \mathbf{C}\right)\mathbf{A}^T + \mathbf{U}\Lambda_{\mathbf{U}}^{-1}\left\langle\mathbf{U}^T, \mathbb{E}\left[\tilde{\mathcal{L}}u\mathbf{H}\right]\right\rangle \\
&= \mathbb{E}\left[\tilde{\mathcal{L}}u\mathbf{H}\right]\mathbf{A}\mathbf{A}^T - \mathbf{U}G_*\mathbf{A}^T + \mathbf{U}\Lambda_{\mathbf{U}}^{-1}\left\langle\mathbf{U}^T, \mathbb{E}\left[\tilde{\mathcal{L}}u\mathbf{H}\right]\right\rangle \\
&= \mathbb{E}\left[\tilde{\mathcal{L}}u\mathbf{H}\right]\mathbf{A}\mathbf{A}^T + \mathbf{U}\Lambda_{\mathbf{U}}^{-1}\left\langle\mathbf{U}^T, \mathbb{E}\left[\tilde{\mathcal{L}}u\mathbf{H}\right]\right\rangle\left(\mathbf{I} - \mathbf{A}\mathbf{A}^T\right),
\end{aligned}$$

where eqn. (2.39b) is plugged in the first equality, eqn. (2.21c) and G_* definition are used in the third equality. Now, from the gPC formulation (2.36) and the DyBO-gPC formulation (2.39), we arrive at the evolution equations for Type- \mathfrak{J} errors,

$$\frac{\partial\bar{\epsilon}}{\partial t} = \mathbb{E}\left[\mathcal{L}u - \mathcal{L}v\right], \quad (2.62a)$$

$$\frac{\partial\epsilon}{\partial t} = \mathbb{E}\left[\tilde{\mathcal{L}}u\mathbf{H}\right]\mathbf{A}\mathbf{A}^T - \mathbb{E}\left[\tilde{\mathcal{L}}v\mathbf{H}\right] + \mathbf{U}\Lambda_{\mathbf{U}}^{-1}\left\langle\mathbf{U}^T, \mathbb{E}\left[\tilde{\mathcal{L}}u\mathbf{H}\right]\right\rangle\left(\mathbf{I} - \mathbf{A}\mathbf{A}^T\right). \quad (2.62b)$$

An interesting observation regarding the affine operator \mathcal{L} as defined in Corollary 2.7 is summarized into the following corollary.

Corollary 2.8. *If the operator \mathcal{L} is affine as defined in Corollary 2.7, i.e., $\mathcal{L} = \mathring{\mathcal{L}} + f$, where $f = \mathbb{E}[f] + \mathbf{F}\mathbf{H}^T$. The evolution equations for type- \mathfrak{J} errors are*

$$\frac{\partial\bar{\epsilon}}{\partial t} = \mathring{\mathcal{L}}\bar{\epsilon}, \quad (2.63a)$$

$$\frac{\partial\epsilon}{\partial t} = \mathring{\mathcal{L}}\epsilon + \left(\mathbf{F} - \mathbf{U}\Lambda_{\mathbf{U}}^{-1}\left\langle\mathbf{U}^T, \mathbf{F}\right\rangle\right)\left(\mathbf{A}\mathbf{A}^T - \mathbf{I}\right). \quad (2.63b)$$

Proof. The corollary can be proved by directly substituting $\mathcal{L} = \mathring{\mathcal{L}} + f$ into eqn. (2.62b) and using the linearity of $\mathring{\mathcal{L}}$. \square

Corollary 2.8 implies that, for an affine operator \mathcal{L} , type- \mathfrak{J} errors are introduced when the spatial modes \mathbf{U} fail to represent the external force f accurately and subsequent amplification or diminishing of errors depends on the property of the deterministic linear operator $\mathring{\mathcal{L}}$.

2.5.3 Type-KL Error

Before ending this section, we consider another type of errors between the gPC and DyBO-gPC formulations. Define

$$\bar{\epsilon} = \bar{u} - \bar{v}, \quad (2.64a)$$

$$\epsilon^{\text{KL}} = \mathbf{U} - \mathbf{V}\mathbf{A}. \quad (2.64b)$$

When no ambiguity arises, we simply write $\epsilon = \epsilon^{\text{KL}}$. Rough speaking, these are errors between true spatial modes in KL expansions and ones computed from the DyBO-gPC solution.

Remark 2.10. As pointed out in Remark 2.7, $\mathbf{A}\mathbf{A}^T \neq \mathbf{I}$ in general when $m \ll N_P$. Type- \mathfrak{J} errors are related to type-KL errors in the following way

$$\begin{aligned} \epsilon^{\mathfrak{J}}\mathbf{A} &= (\mathbf{U}\mathbf{A}^T - \mathbf{V})\mathbf{A} = \mathbf{U}\mathbf{A}^T\mathbf{A} - \mathbf{V}\mathbf{A} = \epsilon^{\text{KL}}, \\ \epsilon^{\text{KL}}\mathbf{A}^T &= (\mathbf{U} - \mathbf{V}\mathbf{A})\mathbf{A}^T = \epsilon^{\mathfrak{J}} + \mathbf{V}(\mathbf{I} - \mathbf{A}\mathbf{A}^T). \end{aligned}$$

Clearly, $\epsilon^{\text{KL}}\mathbf{A}^T = \epsilon^{\mathfrak{J}}$, when $m = N_P$.

To derive the evolution equations for type-KL errors, we calculate

$$\begin{aligned}
\frac{\partial \epsilon}{\partial t} &= \frac{\partial \mathbf{U}}{\partial t} - \frac{\partial \mathbf{V}}{\partial t} \mathbf{A} - \mathbf{V} \frac{d\mathbf{A}}{dt} \\
&= -\mathbf{U}\mathbf{D}^T + \mathbb{E} \left[\tilde{\mathcal{L}}u\mathbf{H} \right] \mathbf{A} - \mathbb{E} \left[\tilde{\mathcal{L}}v\mathbf{H} \right] \mathbf{A} + \mathbf{V}\mathbf{A}\mathbf{C}^T - \mathbf{V} \left\langle \mathbb{E} \left[\tilde{\mathcal{L}}u\mathbf{H}^T \right], \mathbf{U} \right\rangle \Lambda_{\mathbf{U}}^{-1} \\
&= \left(\mathbb{E} \left[\left(\tilde{\mathcal{L}}u - \tilde{\mathcal{L}}v \right) \mathbf{H} \right] \right) \mathbf{A} + \mathbf{U}\mathbf{D} + \mathbf{V}\mathbf{A}\mathbf{C}^T - \mathbf{V} \left\langle \mathbb{E} \left[\tilde{\mathcal{L}}u\mathbf{H}^T \right], \mathbf{U} \right\rangle \Lambda_{\mathbf{U}}^{-1} \\
&= \left(\mathbb{E} \left[\left(\tilde{\mathcal{L}}u - \tilde{\mathcal{L}}v \right) \mathbf{H} \right] \right) \mathbf{A} + \epsilon\mathbf{D} + \mathbf{V}\mathbf{A} \left(\mathbf{D} + \mathbf{C}^T \right) - \mathbf{V} \left\langle \mathbb{E} \left[\tilde{\mathcal{L}}u\mathbf{H}^T \right], \mathbf{U} \right\rangle \Lambda_{\mathbf{U}}^{-1} \\
&= \left(\mathbb{E} \left[\left(\tilde{\mathcal{L}}u - \tilde{\mathcal{L}}v \right) \mathbf{H} \right] \right) \mathbf{A} + \epsilon\mathbf{D} + \mathbf{V} \left(\mathbf{A}\mathbf{A}^T - \mathbf{I} \right) \left\langle \mathbb{E} \left[\tilde{\mathcal{L}}u\mathbf{H}^T \right], \mathbf{U} \right\rangle \Lambda_{\mathbf{U}}^{-1},
\end{aligned}$$

where eqn. (2.39b) is plugged in the first equality and eqn. (2.21c) is used in the fourth equality.

Therefore, the evolution equations for type-KL errors are

$$\frac{\partial \bar{\epsilon}}{\partial t} = \mathbb{E} [\mathcal{L}u - \mathcal{L}v], \quad (2.65a)$$

$$\frac{\partial \epsilon}{\partial t} = \mathbb{E} \left[\left(\tilde{\mathcal{L}}u - \tilde{\mathcal{L}}v \right) \mathbf{H} \right] \mathbf{A} + \epsilon\mathbf{D} + \mathbf{V} \left(\mathbf{A}\mathbf{A}^T - \mathbf{I} \right) \left\langle \mathbb{E} \left[\tilde{\mathcal{L}}u\mathbf{H}^T \right], \mathbf{U} \right\rangle \Lambda_{\mathbf{U}}^{-1}. \quad (2.65b)$$

Similar to Corollary 2.8, we have

Corollary 2.9. *If the differential operator \mathcal{L} is affine as defined in Corollary 2.7, i.e., $\mathcal{L} = \mathring{\mathcal{L}} + f$, where $f = \mathbb{E} [f] + \mathbf{F}\mathbf{H}^T$. The evolution equations for type-KL errors are reduced to*

$$\frac{\partial \bar{\epsilon}}{\partial t} = \mathring{\mathcal{L}}\bar{\epsilon}, \quad (2.66a)$$

$$\frac{\partial \epsilon}{\partial t} = \mathring{\mathcal{L}}\epsilon + \epsilon\mathbf{D} + \mathbf{V} \left(\mathbf{A}\mathbf{A}^T - \mathbf{I} \right) \left\langle \mathbf{F}^T, \mathbf{U} \right\rangle \Lambda_{\mathbf{U}}^{-1}. \quad (2.66b)$$

Proof. The corollary can be proved by directly substituting $\mathcal{L} = \mathring{\mathcal{L}} + f$ into eqn. (2.65b) and using the linearity of $\mathring{\mathcal{L}}$. □

2.6 Adding or Removing Mode Pairs

So far, we have been assuming the number of spatial and stochastic mode pairs, $\{u_l, Y_l\}$'s, in the DyBO-gPC formulation is fixed to some integer m , which determines the number of functions \mathbf{U} and the size of matrix \mathbf{A} to update at each time iteration. *The detailed computational complexity analysis is deferred to Chapter 4 along with numerical verifications.* Certainly, fixing m is not ideal for practical applications for the consideration of both computational cost and accuracy. For example, some spatial and stochastic mode pairs may become negligibly small as the system evolves. Keeping such pairs in computation not only wastes computational resource and increase computational time, but may also bring in unexpected numerical instability since small spatial modes may not be well-resolved on the current spatial and temporal grids. On the other hand, some small pairs may become important later on. Ignoring them certainly introduces numerical errors. Therefore, strategies for changing m adaptively and on-the-fly are necessary.

In this section, we propose strategies to remove and add mode pairs on-the-fly, i.e., insert new equations or drop some equations in the DyBO-gPC formulation (2.39).

2.6.1 Removing Modes

Since $\mathbb{E}[Y_i^2] = 1$, we only need to check the size of $u_i(x, t)$ to evaluate the importance of mode pair (u_i, Y_i) . The strategy to remove modes is relatively straightforward. At the end of each time iteration, we compute $\|u_i\|$ and drop the i -th pair $\{u_i, Y_i\}$ if it is below certain threshold value. This simple strategy is summarized into Algorithm 2.3, where $\eta \in (0, 1)$ is a pre-selected threshold and

$$\lambda_{\max} = \max_{i=1,2,\dots,m} \lambda_i$$

Algorithm 2.3 Removing Mode Pairs

```

1: for  $i = 1, 2, \dots, m$  do
2:   Compute  $\lambda_i = \|u_i\|^2$ 
3:   if  $\frac{\lambda_i}{\lambda_{\max}} < \eta$  then
4:     Drop mode pair  $u_i$  and  $Y_i$  from (2.39).
5:      $m \leftarrow m - 1$ 
6:   end if
7: end for

```

2.6.2 Adding Modes

The situation for adding mode pairs becomes much more complicated. Essentially, we want an algorithm to know *when* and *what* to add without sacrificing too much computational efficiency. A naive approach would be adding some spatial and stochastic mode pair if the smallest eigenvalue rises above some threshold, i.e., $\lambda_{\min} > \eta\lambda_{\max}$. An immediate question is what spatial function and random variable should be used as the initial conditions for the new spatial mode $u_{m+1}(x, t)$ and the stochastic $a_{m+1}(t) \in \mathbb{R}^{N_p \times 1}$ at some time $t = s$.

What's more, the newly added mode pair may remain small and be removed later, which may happen repeatedly and should be avoided. In other words, we should estimate the growth rate of the largest unresolved eigenvalue, i.e., $\frac{d\lambda_{m+1}}{dt}$ or $\frac{d\sqrt{\lambda_{m+1}}}{dt}$ and check if it may potentially grow above the threshold, i.e., $\frac{d\sqrt{\lambda_{m+1}}}{dt} \Delta T \geq \sqrt{\eta\lambda_{\max}}$ after some finite time interval ΔT . It turns out that these two questions are related.

The basic idea of the strategy proposed later for adding mode pairs is to start from the same initial condition, evolve the system by gPC and DyBO-gPC methods, respectively, for a short duration Δs and use the solution discrepancy at a later time to estimate the growth rate of unresolved eigenvalues, and also initial conditions for the new spatial and stochastic modes if additions of new mode pairs are demanded. This heuristic conjecture can be made more rigorous by looking at the type-KL errors we have discussed in the previous section.

Suppose at time $t = s$, the DyBO-gPC solution

$$u(x, s, \boldsymbol{\xi}) = \bar{u}(x, s) + \mathbf{U}(x, s) \mathbf{A}(s)^T \mathbf{H}(\boldsymbol{\xi})^T \quad (2.67)$$

remains a good approximation to the gPC solution, i.e.,

$$v(x, s, \boldsymbol{\xi}) \approx u(x, s, \boldsymbol{\xi}). \quad (2.68)$$

However, as the system continues to evolve for a short time, discrepancy between these two solutions arises and can not be ignored any more. Otherwise, errors will accumulate significantly and render the DyBO-gPC solution at later time useless. Alternatively, we can add one pair of spatial mode $\sqrt{\lambda_{m+1}(t)} \check{u}_{m+1}(x, t)$ and stochastic modes $\mathbf{H}(\boldsymbol{\xi}(\omega)) a_{m+1}(t)$ to compensate such discrepancy, i.e., at time $t = s$,

$$\begin{aligned} u(x, s, \boldsymbol{\xi}) &= \bar{u}(x, s) + (\mathbf{U}(x, s), u_{m+1}(x, s)) (\mathbf{A}(s), a_{m+1}(s))^T \mathbf{H}(\boldsymbol{\xi})^T \\ &= \bar{u}(x, s) + \left(\mathbf{U}(x, s), \sqrt{\lambda_{m+1}(s)} \check{u}_{m+1}(x, s) \right) (\mathbf{A}(s), a_{m+1}(s))^T \mathbf{H}(\boldsymbol{\xi})^T, \end{aligned} \quad (2.69)$$

where $\lambda_{m+1} \approx 0$, $\langle \mathbf{U}(x, s), \check{u}_{m+1}(x, s) \rangle = \mathbf{0}$, $\|\check{u}(x, s)\|_{\mathbb{L}^2(\mathcal{D})} = 1$, $\mathbf{A}(s)^T a_{m+1}(s) = \mathbf{0}$. Both $\sqrt{\lambda_{m+1}(t)} \check{u}_{m+1}(x, t)$ and $\mathbf{H}(\boldsymbol{\xi}(\omega)) a_{m+1}(t)$ are unknown at this moment $t = s$ and will be derived later. After including the unresolved $(m + 1)$ th mode pair, the type-KL error is given by eqn. (2.65b). Now let's estimate both sides at time $t = s$. From eqn. (2.68), we know that

$\mathbf{V}(x, s) = \mathbf{U}(x, s)\mathbf{A}(s)^T$, so

$$\begin{aligned}\epsilon(x, s) &= \left(\mathbf{U}(x, s), \sqrt{\lambda_{m+1}(s)}\check{u}_{m+1}(x, s) \right) - \mathbf{U}(x, s)\mathbf{A}(s)^T (\mathbf{A}(s), a_{m+1}(s)) \\ &= (\mathbf{0}, \sqrt{\lambda_{m+1}(s)}\check{u}_{m+1}(x, s)) \\ &= \mathbf{0}, \quad \text{as } \lambda_{m+1} \rightarrow 0,\end{aligned}$$

where we have used orthogonality of $\mathbf{A}(s)$ and $a_{m+1}(s)$ in the first equality. This simply implies that the second term $\epsilon\mathbf{D}$ on the right-hand side of eqn. (2.65b) is zero at time $t = s$. Similar calculations reveal the third term on the right-hand side is also zero as $\lambda_{m+1} \rightarrow 0$, i.e.,

$$\begin{aligned}\mathbf{V}(x, s) &\left((\mathbf{A}(s), a_{m+1}(s)) (\mathbf{A}(s), a_{m+1}(s))^T - \mathbf{I} \right) \\ &= \mathbf{U}(x, s)\mathbf{A}(s)^T \left((\mathbf{A}(s), a_{m+1}(s)) (\mathbf{A}(s), a_{m+1}(s))^T - \mathbf{I} \right) \\ &= \mathbf{U}(x, s)\mathbf{A}(s)^T (\mathbf{A}(s)\mathbf{A}(s)^T + a_{m+1}(s)a_{m+1}(s)^T - \mathbf{I}) \\ &= \mathbf{U}(x, s) (\mathbf{A}(s)^T + \mathbf{0}a_{m+1}(s)^T - \mathbf{A}(s)^T) \\ &= \mathbf{0}.\end{aligned}$$

Therefore, only the first term on the right-hand side of eqn. (2.65b) really contributes, which can be approximated to the first order accuracy $O(\Delta s)$ as follows,

$$\begin{aligned}\mathbb{E} \left[\tilde{\mathcal{L}}v(x, s, \boldsymbol{\xi})\mathbf{H} \right] &= \mathbb{E} \left[\frac{v(x, s + \Delta s, \boldsymbol{\xi}) - v(x, s, \boldsymbol{\xi})}{\Delta s} \mathbf{H} \right], \\ \mathbb{E} \left[\tilde{\mathcal{L}}u(x, s, \boldsymbol{\xi})\mathbf{H} \right] &= \mathbb{E} \left[\frac{u(x, s + \Delta s, \boldsymbol{\xi}) - u(x, s, \boldsymbol{\xi})}{\Delta s} \mathbf{H} \right].\end{aligned}$$

Because $u|_{t=s} \rightarrow v|_{t=s}$ as $\lambda_{m+1} \rightarrow 0$,

$$\mathbb{E} \left[\left(\tilde{\mathcal{L}}u - \tilde{\mathcal{L}}v \right) \mathbf{H} \right] (\mathbf{A}, a_{m+1}) \Big|_{t=s} \approx \mathbb{E} \left[\frac{u(x, s + \Delta s, \boldsymbol{\xi}) - v(x, s + \Delta s, \boldsymbol{\xi})}{\Delta s} \mathbf{H} \right] (\mathbf{A}(s), a_{m+1}(s)).$$

The last component of the above equality is

$$\mathbb{E} \left[\left(\tilde{\mathcal{L}}u - \tilde{\mathcal{L}}v \right) \mathbf{H} \right] a_{m+1} \Big|_{t=s} = \mathbb{E} \left[\frac{u(x, s + \Delta s, \boldsymbol{\xi}) - v(x, s + \Delta s, \boldsymbol{\xi})}{\Delta s} \mathbf{H} \right] a_{m+1}(s) + \mathcal{O}(\Delta s). \quad (2.70)$$

Now we calculate the last component on the left hand side of eqn. (2.65b), i.e., $\frac{\partial \epsilon_{m+1}}{\partial t}(x, s, \boldsymbol{\xi})$. As

$\lambda_{m+1} \rightarrow 0$, we have

$$\begin{aligned} \frac{\partial \epsilon_{m+1}}{\partial t} \Big|_{t=s} &= \frac{\partial \left(\sqrt{\lambda_{m+1}} \check{u}_{m+1} \right)}{\partial t} \Big|_{t=s} \\ &= \frac{d\sqrt{\lambda_{m+1}}}{dt} \check{u}_{m+1} \Big|_{t=s} + \sqrt{\lambda_{m+1}} \frac{\partial \check{u}_{m+1}}{\partial t} \Big|_{t=s} \\ &= \frac{d\sqrt{\lambda_{m+1}}}{dt}(s) \check{u}_{m+1}(x, s). \end{aligned}$$

Combining the above discussion, we have the following equality from eqn. (2.65),

$$\frac{d\sqrt{\lambda_{m+1}}}{dt}(s) \check{u}_{m+1}(x, s) = \mathbb{E} \left[\frac{u(x, s + \Delta s, \boldsymbol{\xi}) - v(x, s + \Delta s, \boldsymbol{\xi})}{\Delta s} \mathbf{H} \right] a_{m+1}(s). \quad (2.71)$$

Now consider the KL expansion of the solution discrepancy at time $t = s + \Delta s$, i.e.,

$$\Delta u(x, s + \Delta s, \boldsymbol{\xi}) = u(x, s + \Delta s, \boldsymbol{\xi}) - v(x, s + \Delta s, \boldsymbol{\xi}) = \sqrt{\theta_1} w_1(x) b_1^T \mathbf{H} + \dots, \quad (2.72)$$

where θ_1 , $w_1(x)$ and $\mathbf{H}b_1$ are the largest eigenvalue, normalized spatial and stochastic modes, respectively. The above equality implies that the growth rate of the largest unresolved eigenvalue

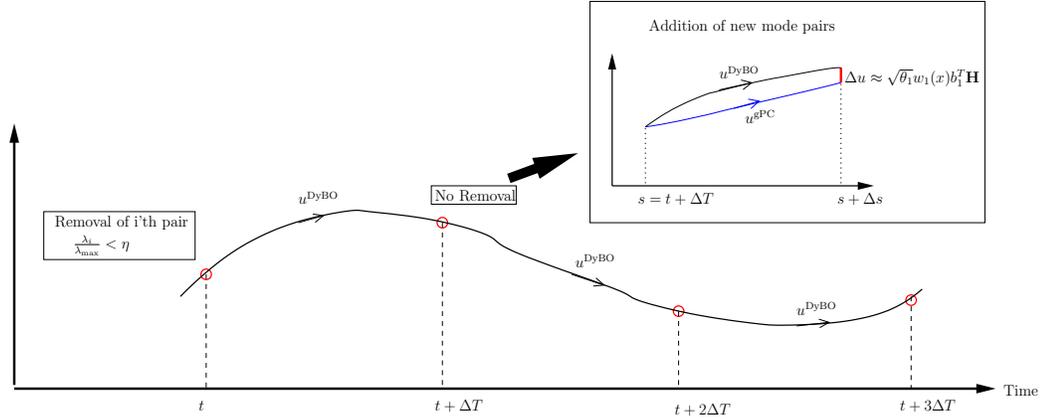


Figure 2.2: Illustration of strategies of adding and removing mode pairs

λ_{m+1} can be estimated from the largest eigenvalue of Δu . What's more, a sensible choice of initial conditions for the newly added mode pair $u_{m+1}(x, s)$ and $a_{m+1}(s)$ would be the largest spatial and stochastic mode of $\Delta u(x, s + \Delta s, \xi)$, i.e., $\sqrt{\theta_1} w_1(x)$ and $\mathbf{H}b_1$. This strategy involves computation of gPC solutions for a short time Δs , which can be expensive. Instead of invoking such strategy every time step, we can instead only invoke such procedure every duration ΔT , $\Delta T \gg \Delta s$.

The above discussion is summarized into Algorithm 2.4. See Fig. 2.2 for illustrations.

Remark 2.11. Similar strategies of removing and adding spatial and stochastic mode pairs may be developed for DyBO-gSC and DyBO-MC.

Remark 2.12. Algorithm 2.4 can be generalized to add more than one pair of spatial and stochastic modes at a time.

Remark 2.13. The new spatial mode u_{m+1} and new stochastic mode $\mathbf{H}a_{m+1}(t)$ may not be perfectly orthogonal to other modes $\mathbf{U}(x, t)$ and $\mathbf{H}\mathbf{A}$ at time $t = s$. Due to Remark 2.5 in Sec. 2.2, it will become so after several time iterations.

Algorithm 2.4 Adding Mode Pairs

- 1: **if** $\lambda_{\min} < \eta\lambda_{\max}$ **then**
- 2: Continue without adding mode pairs
- 3: **end if**
- 4: Evolve DyBO system (2.39) from $t = s$ to $t = s + \Delta s$ with the initial condition

$$u(x, s, \boldsymbol{\xi}) = \bar{u}(s) + \mathbf{U}(x, s)\mathbf{A}(s)^T\mathbf{H}(\boldsymbol{\xi})$$

- 5: Evolve gPC system (2.36) from $t = s$ to $t = s + \Delta s$ with the same initial condition

$$v(x, s, \boldsymbol{\xi}) = \bar{u}(s) + \mathbf{U}(x, s)\mathbf{A}(s)^T\mathbf{H}(\boldsymbol{\xi})$$

- 6: Compute the largest spatial and stochastic mode pair $\{\sqrt{\theta_1}w_1(x), b_1\}$ in the KLE of solution difference $\Delta u(x, t, \boldsymbol{\xi})$ at time $t = s + \Delta s$, i.e., eqn. (2.72).
- 7: Estimate the growth rate of the largest unresolved eigenvalue

$$\frac{d\sqrt{\lambda_{m+1}}}{dt}(s) \approx \frac{\sqrt{\theta_1}}{\Delta s}$$

- 8: **if** $\frac{d\sqrt{\lambda_{m+1}}}{dt} < \frac{\sqrt{\eta\lambda_{\max}}}{\Delta T}$ **then**
 - 9: Continue without adding mode pairs
 - 10: **else**
 - 11: $m \leftarrow m + 1$
 - 12: Compute λ_{m+1} by extrapolation from several smallest eigenvalues, i.e., $\lambda_{m-q}, \lambda_{m-q+1}, \dots, \lambda_m$.
 - 13: Add a new spatial mode $u_{m+1}(x, s) = \sqrt{\lambda_{m+1}}w_1(x)$.
 - 14: Add a new stochastic mode $a_{m+1}(s) = b_1$.
 - 15: **end if**
 - 16: Continue to evolve DyBO system (2.28).
-

2.7 Overall DyBO-gPC Algorithm

2.7.1 Initializations

Discussions in the previous sections mainly concern updates of spatial and stochastic modes as the system evolves. However, we still need to provide initial conditions for such updates to start. When randomness only comes into the system through the stochastic differential operator \mathcal{L} , e.g., stochastic forcing, the initial condition $u(x, 0, \omega)$ is deterministic and the spatial and stochastic modes, $\mathbf{U}(x, 0)$ and $\mathbf{HA}(0)$ in the DyBO-gPC formulation are not well defined. A similar situation arises when the randomness is only present along the spatial domain boundary $\partial\mathcal{D}$ and propagates into the interior.

To rectify such situations and improve numerical stability, we solve SPDE (1.1) by the gPC formulation (2.36) for a short time duration ΔT_0 , compute the KL expansion of $v(x, \Delta T_0, \omega)$ as described in Sec. 2.4.2 and truncate it after certain threshold, i.e.,

$$v(x, \Delta T_0, \boldsymbol{\xi}) \approx \bar{v}(x, \Delta T_0) + \sum_{i=1}^m \sqrt{\lambda_i} \check{v}_i(x, \Delta T_0) b_i(\Delta T_0)^T \mathbf{H}(\boldsymbol{\xi})^T,$$

where λ_i 's are arranged in descending order and m is chosen such that $\frac{\lambda_m}{\lambda_1} < \eta$. The initial conditions for DyBO-gPC formulation are

$$\bar{u}(x, \Delta T_0) = \bar{v}(x, \Delta T_0), \tag{2.73a}$$

$$\mathbf{U}(x, \Delta T_0) = \left(\sqrt{\lambda_1} \check{v}_1(x, \Delta T_0), \sqrt{\lambda_2} \check{v}_2(x, \Delta T_0), \dots, \sqrt{\lambda_m} \check{v}_m(x, \Delta T_0) \right), \tag{2.73b}$$

$$\mathbf{A}(\Delta T_0) = (b_1(\Delta T_0), b_2(\Delta T_0), \dots, b_m(\Delta T_0)). \tag{2.73c}$$

That is to say, we start the numerical simulation of the DyBO-gPC system (2.39) at time $t = \Delta T_0$

instead $t = 0$.

2.7.2 DyBO-gPC Algorithm

In addition to the finite truncation of KL expansion, spatial and temporal discretizations also contribute to numerical errors. Since the primary interests of this thesis are to investigate the possibility of utilizing the low-dimensional structure of the system and solving the SPDE efficiently, we assume periodic boundary conditions throughout the thesis, and apply a pseudo-spectral method [16] for spatial discretization and fourth-order Runge-Kutta method for temporal integration. In this way, errors due to spatial and temporal discretizations are considered to be much smaller than those due to finite truncation of expansions. An additional benefit of using a spectral method is that all the numerical computations of inner products $\langle \cdot, \cdot \rangle$ have spectral accuracy [94].

Now we are ready to present the overall algorithm in the following pseudo code.

Algorithm 2.5 Overall DyBO-gPC Algorithm

- 1: Solve gPC formulation (2.36) up to time $t = \Delta T_0$.
 - 2: Compute initial conditions (2.73) for DyBO-gPC method as described in Sec. 2.7.1.
 - 3: $t \leftarrow \Delta T_0$
 - 4: **while** $t < T$ **do**
 - 5: Evolve DyBO-gPC system (2.39) for duration ΔT_0 , where Algorithm 2.1 of Y-Stage or Algorithm 2.2 U-Stage is invoked every time step to check eigenvalue crossing.
 - 6: Invoke Algorithm 2.3 to remove small mode pairs if any.
 - 7: **if** No mode pair are removed **then**
 - 8: Invoke Algorithm 2.4 to add unresolved mode pairs if necessary.
 - 9: **end if**
 - 10: $t \leftarrow t + \Delta T$
 - 11: **end while**
 - 12: Compute various statistical quantities, such as mean, variance and high-order moments.
-

Chapter 3

Applications to Spatially One-dimensional SPDEs

While Chapter 2 highlights the theoretical aspects of DyBO method, this chapter demonstrates its success by several numerical examples of increasing difficulties, each of which emphasizes and verifies some of analytical results in the previous chapter. All of the numerical examples in this chapter are spatially one-dimensional, i.e., $d = 1$. In Sec. 3.1, SPDE driven purely by stochastic forces are considered, which can be considered the simplest situation. Corollary 2.6 from error analysis of DyBO is numerically verified in Sec. 3.2 by considering a transport equation with deterministic velocity and random initial conditions. In the last section, we consider Burgers' equation driven by stochastic force as an example of a nonlinear PDE driven by stochastic forces, where the convergence of the DyBO method with respect to the number of mode pairs, m , and the effectiveness of proposed strategies for adding and removing mode pairs are our primary focus. More involved numerical examples, such as spatially two-dimensional SPDE and/or a system of SPDEs, will be considered in Chapter 4 and Chapter 6.

Remark 3.1. The arrangement of sections in this chapter is not by accident, but actually reflects how our DyBO method along with its numerical algorithm was developed in the course of PhD study. At the very beginning, we did not consider eigenvalues crossing and were immediately forced to do so once we applied it to the first numerical example in Sec. 3.1. When we considered Burgers' equation

driven by stochastic forces, we were once again forced to consider some strategies for adding and removing mode pairs from the DyBO system. Surprisingly, the simplest numerical example Sec. 3.1 again guides our development of such strategies.

3.1 SPDE Purely Driven by Stochastic Force

To investigate numerically the proposed DyBO-gPC algorithms, it is ideal to have some SPDE problems whose solutions and their KL expansions are known exactly and can be controlled easily, so critical phenomena which may arise in more involved numerical simulations, such as eigenvalues crossing, can be thoroughly examined in such controlled environments first. However, it is not a trivial task to obtain such SPDE problems which satisfy conditions mentioned above. In this section, we consider a simple case, SPDE driven purely by stochastic force f , i.e., $\frac{\partial u}{\partial t} = \mathcal{L}u = f(x, t, \omega)$, whose solution can be obtained by direct integration $u(x, t, \omega) = u(x, 0, \omega) + \int_0^t f(x, s, \omega) ds$ or f can be obtained by differentiating directly the exact solution u . Specifically, in this section, we consider SPDE

$$\frac{\partial u}{\partial t} = \mathcal{L}u = f(x, t, \boldsymbol{\xi}(\omega)), \quad x \in \mathcal{D} = [0, 1], \quad t \in [0, T], \quad (3.1)$$

where $\boldsymbol{\xi} = (\xi_1, \xi_2, \dots, \xi_{N_p})$ are independent standard Gaussian random variables, i.e., $\xi_i \sim \mathcal{N}(0, 1)$ and the stochastic force f will be given by differentiating the exact solution. The exact solution is constructed as follows

$$u(x, t, \boldsymbol{\xi}) = \bar{v}(x, t) + \mathbf{V}(x, t)\mathbf{Z}^T(\boldsymbol{\xi}, t), \quad (3.2)$$

$$\mathbf{V}(x, t) = \mathring{\mathbf{V}}(x)\mathbf{W}_{\mathbf{V}}(t)\boldsymbol{\Lambda}_{\mathbf{V}}^{\frac{1}{2}}(t),$$

$$\mathbf{Z}(\boldsymbol{\xi}, t) = \mathring{\mathbf{Z}}(\boldsymbol{\xi})\mathbf{W}_{\mathbf{Z}}(t),$$

where $\mathring{\mathbf{V}}(x) = (\mathring{v}_1(x), \dots, \mathring{v}_m(x))$ with $\langle \mathring{v}_i(x), \mathring{v}_j(x) \rangle = \delta_{ij}$ and $\mathring{\mathbf{Z}}(\boldsymbol{\xi}) = (\mathring{Z}_1(\boldsymbol{\xi}), \dots, \mathring{Z}_m(\boldsymbol{\xi}))$ with $\mathbb{E} [\mathring{Z}_i \mathring{Z}_j] = \delta_{ij}$ for $i, j = 1, 2, \dots, m$. $\mathbf{W}_\mathbf{V}(t)$ and $\mathbf{W}_\mathbf{Z}(t)$ are m -by- m orthonormal matrices, and $\Lambda_{\mathring{\mathbf{V}}}^{\frac{1}{2}}(t)$ is a diagonal matrix.

Clearly, the exact solution u is intentionally given in the form of KL expansion which has only finite terms and the diagonal entries of matrix $\Lambda_{\mathbf{V}}(t)$ consist of eigenvalues. By carefully choosing the values of eigenvalues, we can mimic various difficulties which may arise in more involved situations and devise corresponding strategies. Thus, the stochastic forcing term on the right-hand side of eqn. (3.1) can be obtained by differentiating the exact solution, i.e.,

$$\mathcal{L}u = f = \frac{\partial \bar{v}}{\partial t} + \frac{\partial \mathbf{V}}{\partial t} \mathbf{Z}^T + \mathbf{V} \frac{d\mathbf{Z}^T}{dt}, \quad (3.3)$$

$$\begin{aligned} \frac{\partial \mathbf{V}}{\partial t} &= \mathring{\mathbf{V}} \left(\frac{d\mathbf{W}_\mathbf{V}}{dt} \Lambda_{\mathring{\mathbf{V}}}^{\frac{1}{2}} + \mathbf{W}_\mathbf{V} \frac{d\Lambda_{\mathring{\mathbf{V}}}^{\frac{1}{2}}}{dt} \right), \\ \frac{d\mathbf{Z}}{dt} &= \mathring{\mathbf{Z}} \frac{d\mathbf{W}_\mathbf{Z}}{dt}. \end{aligned}$$

The initial condition of SPDE (3.1) can be obtained by simply setting $t = 0$ in eqn. (3.2). Now we are ready to derive the DyBO-gPC formulation for SPDE (3.1). Simple calculations give

$$\begin{aligned} \mathbb{E} [\tilde{\mathcal{L}}u \mathbf{H}] &= \frac{\partial \mathbf{V}}{\partial t} \mathbb{E} [\mathbf{Z}^T \mathbf{H}] + \mathbf{V} \mathbb{E} \left[\frac{d\mathbf{Z}^T}{dt} \mathbf{H} \right] \\ &= \frac{\partial \mathbf{V}}{\partial t} \mathbf{W}_\mathbf{Z}^T \mathbb{E} [\mathring{\mathbf{Z}}^T \mathbf{H}] + \mathbf{V} \frac{d\mathbf{W}_\mathbf{Z}^T}{dt} \mathbb{E} [\mathring{\mathbf{Z}}^T \mathbf{H}] \\ &= \left(\frac{\partial \mathbf{V}}{\partial t} \mathbf{W}_\mathbf{Z}^T + \mathbf{V} \frac{d\mathbf{W}_\mathbf{Z}^T}{dt} \right) \mathbb{E} [\mathring{\mathbf{Z}}^T \mathbf{H}], \\ \langle \mathbb{E} [\mathbf{H}^T \tilde{\mathcal{L}}u], \mathbf{U} \rangle &= \mathbb{E} [\mathbf{H}^T \mathring{\mathbf{Z}}] \left\langle \mathbf{W}_\mathbf{Z} \frac{\partial \mathbf{V}^T}{\partial t} + \frac{d\mathbf{W}_\mathbf{Z}}{dt} \mathbf{V}^T, \mathbf{U} \right\rangle. \end{aligned}$$

By plugging the above equalities into DyBO-gPC system (2.39), we arrive at the DyBO-gPC for-

mulation for SPDE (3.1)

$$\frac{\partial \bar{u}}{\partial t} = \frac{\partial \bar{v}}{\partial t}, \quad (3.4a)$$

$$\frac{\partial \mathbf{U}}{\partial t} = -\mathbf{U}\mathbf{D}^T + \left(\frac{\partial \mathbf{V}}{\partial t} \mathbf{W}_{\mathbf{Z}}^T + \mathbf{V} \frac{d\mathbf{W}_{\mathbf{Z}}^T}{dt} \right) \mathbb{E} \left[\dot{\mathbf{Z}}^T \mathbf{H} \right] \mathbf{A}, \quad (3.4b)$$

$$\frac{d\mathbf{A}}{dt} = -\mathbf{A}\mathbf{C}^T + \mathbb{E} \left[\mathbf{H}^T \dot{\mathbf{Z}} \right] \left\langle \mathbf{W}_{\mathbf{Z}} \frac{\partial \mathbf{V}^T}{\partial t} + \frac{d\mathbf{W}_{\mathbf{Z}}}{dt} \mathbf{V}^T, \mathbf{U} \right\rangle \Lambda_{\mathbf{U}}^{-1}, \quad (3.4c)$$

and

$$G_*(u, \mathbf{U}, \mathbf{A}) = \Lambda_{\mathbf{U}}^{-1} \left\langle \mathbf{U}, \frac{\partial \mathbf{V}}{\partial t} \mathbf{W}_{\mathbf{Z}}^T + \mathbf{V} \frac{d\mathbf{W}_{\mathbf{Z}}^T}{dt} \right\rangle \mathbb{E} \left[\dot{\mathbf{Z}}^T \mathbf{H} \right] \mathbf{A}.$$

The initial conditions are simply

$$\bar{u}(x, 0) = \bar{v}(x, 0),$$

$$\mathbf{U}(x, 0) = \mathbf{V}(x, 0),$$

$$\mathbf{A}(0) = \mathbb{E} \left[\mathbf{H}^T \mathbf{Z}(\boldsymbol{\xi}, 0) \right] = \mathbb{E} \left[\mathbf{H}^T \dot{\mathbf{Z}} \right] \mathbf{W}_{\mathbf{Z}}(0).$$

In the event of eigenvalue crossing, we have the \mathbf{Y} -stage system from (2.50),

$$\frac{\partial \bar{u}}{\partial t} = \frac{\partial \bar{v}}{\partial t}, \quad (3.5a)$$

$$\frac{\partial \mathbf{U}}{\partial t} = \left(\frac{\partial \mathbf{V}}{\partial t} \mathbf{W}_{\mathbf{Z}}^T + \mathbf{V} \frac{d\mathbf{W}_{\mathbf{Z}}^T}{dt} \right) \mathbb{E} \left[\dot{\mathbf{Z}}^T \mathbf{H} \right] \mathbf{A}, \quad (3.5b)$$

or the \mathbf{U} -stage from system (2.57),

$$\frac{\partial \bar{u}}{\partial t} = \frac{\partial \bar{v}}{\partial t}, \quad (3.6a)$$

$$\frac{d\mathbf{A}}{dt} = \mathbb{E} \left[\mathbf{H}^T \dot{\mathbf{Z}} \right] \left\langle \mathbf{W}_{\mathbf{Z}} \frac{\partial \mathbf{V}^T}{\partial t} + \frac{d\mathbf{W}_{\mathbf{Z}}}{dt} \mathbf{V}^T, \mathbf{U} \right\rangle \Lambda_{\mathbf{U}}^{-1}. \quad (3.6b)$$

In the numerical examples presented in this section, we consider a small system $m = 3$ and use the following settings,

$$\mathring{\mathbf{V}}(x) = \left(\sqrt{2} \sin(\pi x), \sqrt{2} \sin(5\pi x), \sqrt{2} \sin(9\pi x) \right),$$

$$\mathring{\mathbf{Z}}(x) = (\mathbf{H}_1(\xi_1), \mathbf{H}_2(\xi_1), \mathbf{H}_3(\xi_1)),$$

$$\mathbf{W}_{\mathbf{V}}(t) = \mathbf{P}_{\mathbf{V}} \mathbf{O}_{\mathbf{V}}(t) \mathbf{P}_{\mathbf{V}}^T,$$

$$\mathbf{W}_{\mathbf{Z}}(t) = \mathbf{P}_{\mathbf{Z}} \mathbf{O}_{\mathbf{Z}}(t) \mathbf{P}_{\mathbf{Z}}^T,$$

$$\mathbf{O}_{\mathbf{V}}(t) = \begin{pmatrix} \cos b_{\mathbf{V}}t & -\sin b_{\mathbf{V}}t & 0 \\ \sin b_{\mathbf{V}}t & \cos b_{\mathbf{V}}t & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad b_{\mathbf{V}} = 2.0,$$

$$\mathbf{O}_{\mathbf{Z}}(t) = \begin{pmatrix} \cos b_{\mathbf{Z}}t & -\sin b_{\mathbf{Z}}t & 0 \\ \sin b_{\mathbf{Z}}t & \cos b_{\mathbf{Z}}t & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad b_{\mathbf{Z}} = 2.0,$$

$$\mathbf{H}(\boldsymbol{\xi}) = (\mathbf{H}_1(\xi_1), \mathbf{H}_2(\xi_1), \dots, \mathbf{H}_5(\xi_1)),$$

where $\mathbf{P}_{\mathbf{V}}$ and $\mathbf{P}_{\mathbf{Z}}$ are two orthonormal matrices generated randomly,

$$\mathbf{P}_{\mathbf{V}} = \begin{pmatrix} 1.49785196602 \times 10^{-02} & 9.75211144424 \times 10^{-01} & 2.20768810614 \times 10^{-01} \\ 9.97706302488 \times 10^{-01} & 0 & -6.76914616037 \times 10^{-02} \\ 6.60134677383 \times 10^{-02} & -2.21276351631 \times 10^{-01} & 9.72974305049 \times 10^{-01} \end{pmatrix},$$

$$\mathbf{P}_{\mathbf{Z}} = \begin{pmatrix} -9.53187324001 \times 10^{-01} & -3.02188804978 \times 10^{-01} & 1.07634338851 \times 10^{-02} \\ 1.12913248587 \times 10^{-02} & 0 & 9.99936250959 \times 10^{-01} \\ 3.02169540731 \times 10^{-01} & -9.53248092652 \times 10^{-01} & -3.41211196568 \times 10^{-03} \end{pmatrix}.$$

3.1.1 Eigenvalue Crossing

Eigenvalues in a KL expansion of an SPDE solution may increase or decrease as time goes. Some of them may approach each other at some time, cross and then separate later. When two eigenvalues are close or equal to each other, numerical instability may arise in solving matrices \mathbf{C} and \mathbf{D} via (2.22). In Sec. 2.4.1, we have proposed a \mathbf{U} -stage by freezing the spatial modes \mathbf{U} or a \mathbf{Y} -stage by freezing the stochastic modes \mathbf{Y} temporarily to solve this issue. Here, we demonstrate the success of incorporation of such strategies into the overall DyBO algorithm. To this end, we choose $T = 1.2$ and eigenvalues

$$\Lambda_{\mathbf{V}}^{\frac{1}{2}} = \text{diag}(\sin 2\pi t + 2, \cos 2\pi t + 1.5, 1.8),$$

where eigenvalues cross each other at $t \approx 0.0675, 0.2015, 0.5320, 0.7985, 0.9650, 1.0675$ for $t \in [0, T]$. See Fig. 3.3.

In this numerical example, the time step $\delta t = 1.0 \times 10^{-3}$ and $k_{\mathbf{U}} = 20$ in the \mathbf{U} -stage, i.e., the exiting condition is checked every 20 time iterations when the system is in the \mathbf{U} -stage. Mean $\mathbb{E}[u]$, Standard Deviation (STD) $\sqrt{\text{Var}(u)}$ and the three spatial modes in KLE at time $t = T$ are shown in Fig. 3.1 and Fig. 3.2, respectively. Clearly, the results given by DyBO almost perfectly match the exact ones with \mathbb{L}^2 relative errors of mean and STD below 10^{-10} since only spatial and temporal discretizations contribute to numerical errors in this example.

In Fig. 3.3, eigenvalues are plotted as functions of time and zooming-outs are given at $t = 0.0675$ and $t = 0.7985$ to show eigenvalue crossings and invoking of the \mathbf{U} -stage Algorithm 2.2.

We also check the deviation of computed spatial and stochastic modes from bi-orthogonality by monitoring $\frac{\langle u_i, u_j \rangle}{\|u_i\|_{\mathbb{L}^2(\mathcal{D})} \|u_j\|_{\mathbb{L}^2(\mathcal{D})}}$ for spatial modes and $\mathbb{E}[Y_i Y_j]$ for stochastic modes, which are shown in Fig. 3.4. We observe that the deviation of spatial modes from orthogonality is very small ($< 10^{-10}$) throughout the computation. Large deviations of stochastic modes from orthogonality

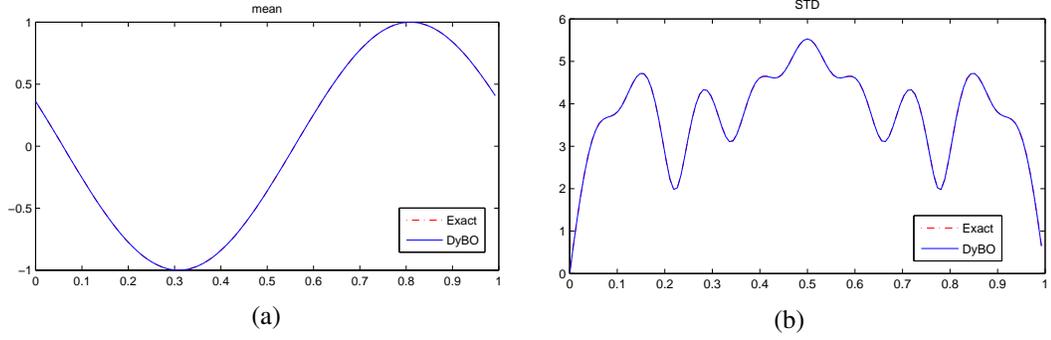


Figure 3.1: Mean and STD computed by DyBO at time $t = 1.2$

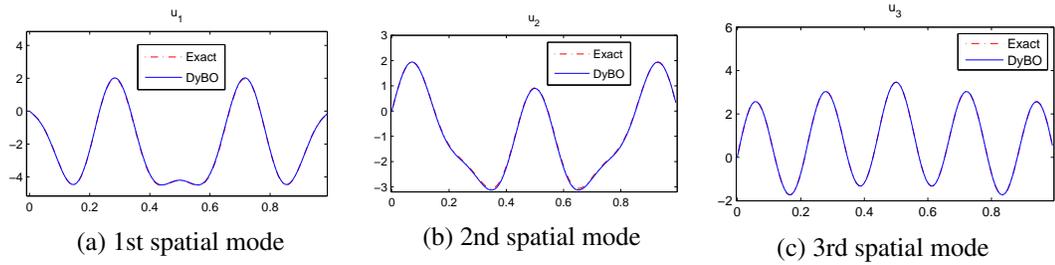


Figure 3.2: Spatial modes in KL expansion of SPDE solution computed by DyBO at time $t = 1.2$

only occur when eigenvalues cross each other since the DyBO algorithm is not designed to preserve the orthogonality of \mathbf{Y} during the \mathbf{U} -stage and orthogonality is only restored once the algorithm exits from the \mathbf{U} -stage. Instead, we also apply the \mathbf{Y} -stage to overcome eigenvalue-crossing issues in this numerical examples and get similar results.

3.1.2 Adding and Removing Modes

Now we consider a slightly different numerical example for adding or removing mode pairs from the DyBO system (3.4).

$$\Lambda_{\mathbf{V}}^{\frac{1}{2}} = \text{diag} (3.0001 + \sin(2\pi t), 2.0001 + \sin(2\pi t), 1.0001 + \sin(2\pi t)),$$

where λ_3 becomes very small $\sim 10^{-8}$ near $t = 0.25$. See Fig. 3.5. When Algorithm 2.4 for adding mode pairs is invoked, it is crucial to know the change rate of the largest unresolved eigenvalue and

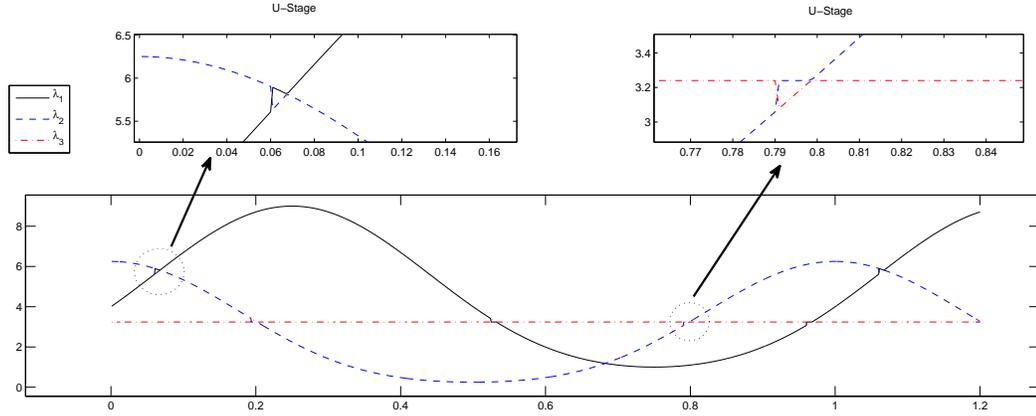


Figure 3.3: Eigenvalues computed by DyBO. Two zooming-outs are provided at the time when two eigenvalues cross each other to show the invocation of **U-stage Algorithm 2.2**

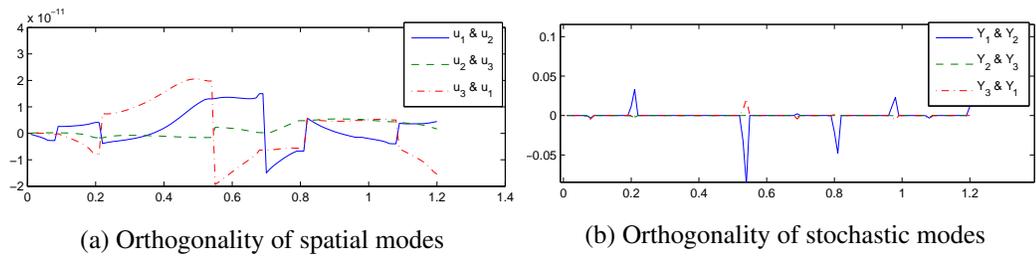


Figure 3.4: Bi-orthogonality of spatial and stochastic modes. Orthogonality of \mathbf{U} is preserved throughout the computation while stochastic modes deviate from orthogonality only at eigenvalue crossing

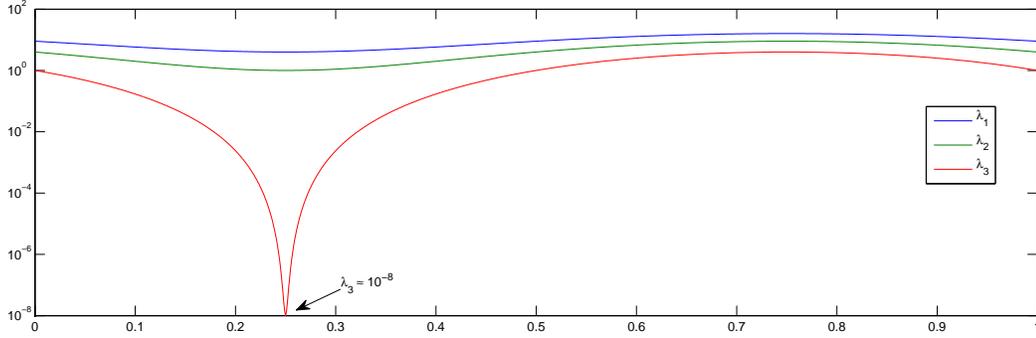


Figure 3.5: Eigenvalues. Eigenvalues are plotted as function of time. λ_3 becomes small near $t = 0.25$

avoid adding such mode pair if it continues to be small in the near future ΔT . This is accomplished, as described in Sec. 2.6.2, by computing solutions by DyBO and gPC for a short time Δs and estimating the change rate $\frac{d\sqrt{\lambda_{m+1}}}{dt}$ from the difference.

In Fig. 3.6, we verify the accuracy of such estimates, where the third mode pair is intentionally dropped at $t = 0.2$ when it becomes small ($\sim 10^{-3}$) and never put back in the remaining computation. The solid line is the exact change rate of the largest unresolved eigenvalue, i.e., $\frac{d\sqrt{\lambda_3}}{dt}$, while the dotted line is the estimate. In computations, we actually use different short time duration $\Delta s = 8\delta t, 4\delta t, 2\delta t, \delta t$ to verify the convergence of such estimate. However, all of these estimates cluster together and cannot be distinguished from the figure. As we can see from Fig. 3.6, such estimates are very accurate when the largest unresolved eigenvalue is indeed small and become less accurate when the largest unresolved eigenvalue is not so small compared to resolved ones.

In Fig. 3.7, we consider the effect of invoking frequency of Algorithm 2.4, i.e., $\frac{1}{\Delta T}$. If no mode pair is added, the relative error of STD at $t = 1.0$ is about 26%. When Algorithm 2.4 is incorporated, the error can be brought down to $\leq 1.5\%$ depending on the invoking frequency. The threshold η in Algorithm 2.4 is taken to be 10^{-4} and $\sqrt{\eta} = 10^{-2}$, so we see such difference is relatively marginal. We will continue to demonstrate the effectiveness of Algorithm 2.4 in more involved numerical examples in the rest of the thesis.

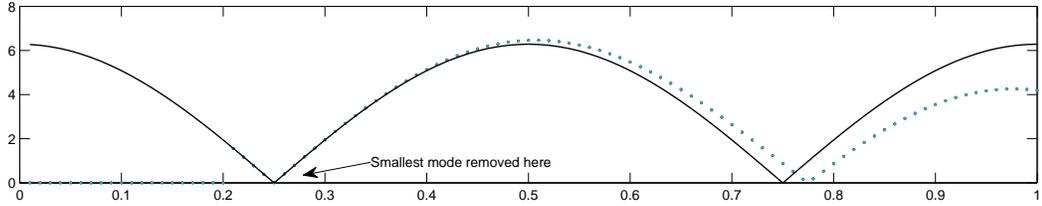


Figure 3.6: Change rate of the largest unresolved eigenvalue $\frac{d\sqrt{\lambda_3}}{dt}$. Solid line is given by the exact solution, while the dotted line are computed as described in Sec. 2.6.2

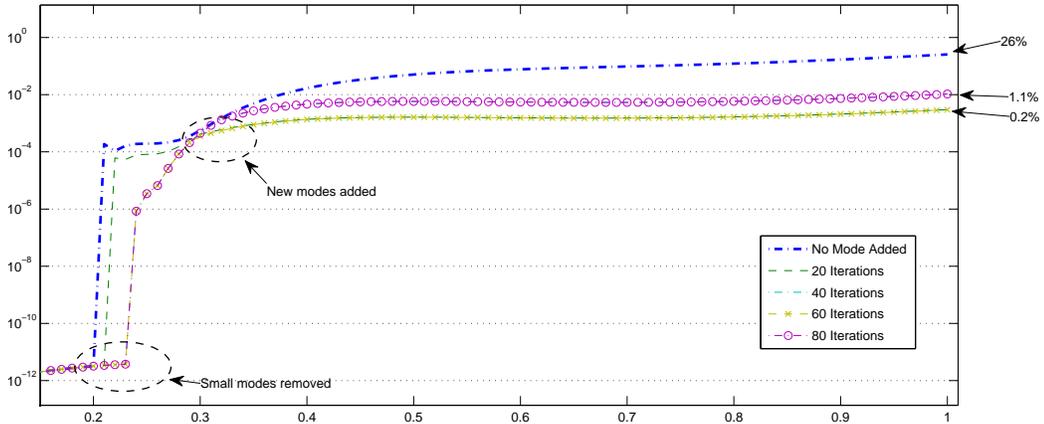


Figure 3.7: \mathbb{L}^2 relative errors of STD given by DyBO for different invoking frequency of Algorithm 2.4 for adding or removing mode pairs

3.2 Linear Deterministic Differential Operators with Random Initial Conditions

In this section, we consider a linear PDE with random initial conditions, i.e., differential operator \mathcal{L} is deterministic and linear. For simplicity, we assume periodic boundary conditions.

$$\frac{\partial u}{\partial t} = \mathcal{L}u, \quad x \in [0, 1], \quad t \in [0, 0.4], \quad (3.7a)$$

$$u(0, t, \boldsymbol{\xi}) = u(1, t, \boldsymbol{\xi}), \quad (3.7b)$$

$$u(x, 0, \boldsymbol{\xi}) = \hat{u}(x, \boldsymbol{\xi}), \quad (3.7c)$$

where $\boldsymbol{\xi} = (\xi_1, \xi_2, \dots, \xi_{N_P})$ are independent standard Gaussian random variables.

Consider the gPC expansion of solution $u = \bar{v} + \mathbf{V}\mathbf{H}^T$. From eqn. (2.36), it is easy to get the gPC formulation of this SPDE,

$$\frac{\partial \bar{v}}{\partial t} = \mathcal{L}\bar{v}, \quad (3.8a)$$

$$\frac{\partial \mathbf{V}}{\partial t} = \mathcal{L}\mathbf{V}, \quad (3.8b)$$

where initial conditions $\bar{v}(x, 0) = \mathbb{E}[\hat{u}]$ and $\mathbf{V}(x, 0) = \mathbb{E}[\hat{u}(x)\mathbf{H}]$. Clearly, gPC formulation (3.8) is linear and provides the exact solution to the original SPDE (3.7) if the gPC expansion of initial condition is exact, i.e., $\hat{u} = \mathbb{E}[\hat{u}] + \mathbb{E}[\hat{u}\mathbf{H}]\mathbf{H}^T$.

Now consider the KL expansion of solution $u = \bar{u} + \mathbf{U}\mathbf{Y}^T = \bar{u} + \mathbf{U}\mathbf{A}^T\mathbf{H}^T$. Simple calculations give

$$\mathcal{L}u = \mathcal{L}\bar{u} + \mathcal{L}\mathbf{U}\mathbf{A}^T\mathbf{H}^T,$$

$$\mathbb{E}[\mathcal{L}u] = \mathcal{L}\bar{u},$$

$$\tilde{\mathcal{L}}u = \mathcal{L}\mathbf{U}\mathbf{A}^T\mathbf{H}^T,$$

$$\mathbb{E}[\tilde{\mathcal{L}}u\mathbf{H}] = \mathcal{L}\mathbf{U}\mathbf{A}^T.$$

From eqn. (2.39), we have the DyBO-gPC formulation

$$\frac{\partial \bar{u}}{\partial t} = \mathcal{L}\bar{u}, \quad (3.9a)$$

$$\frac{\partial \mathbf{U}}{\partial t} = -\mathbf{U}\mathbf{D}^T + \mathcal{L}\mathbf{U}, \quad (3.9b)$$

$$\frac{d\mathbf{A}}{dt} = -\mathbf{A}\mathbf{C}^T + \mathbf{A}\langle \mathcal{L}\mathbf{U}^T, \mathbf{U} \rangle \Lambda_{\mathbf{U}}^{-1}, \quad (3.9c)$$

where \mathbf{C} and \mathbf{D} can be computed via eqn. (2.22) from

$$G_* = \Lambda_{\mathbf{U}}^{-1} \langle \mathbf{U}^T, \mathcal{L}\mathbf{U} \rangle.$$

If we compare the DyBO formulation (3.9) and the gPC formulation (3.8) for the linear SPDE (3.7), an interesting observation arises immediately. Although the original SPDE is linear and the gPC formulation remains linear, the DyBO formulation is clearly *not* linear. However, this is not a surprise because KL expansion is not a linear procedure, i.e., we cannot simply add spatial and stochastic modes of two stochastic processes to get the KL expansion of the sum of these two stochastic processes. On the other hand, the gPC expansion is indeed a linear procedure. As we argue in Sec. 2.5, our DyBO formulation essentially tracks the KL expansion of the exact solution. Therefore, the nonlinearity of KLE must be naturally built into the DyBO formulation to enable such tracking.

Corollary 2.6 implies that the solution given by the DyBO formulation is exact if the initial condition \hat{u} can be expressed exactly by a finite-term KL expansion. To verify this numerically, we consider a non-trivial transport equation, i.e., $\mathcal{L} = -\sin(2\pi(x+2t))\frac{\partial}{\partial x}$ and the initial condition is a functional of three standard Gaussian random variables, i.e., $N_p = 3$

$$\hat{u}(x, \boldsymbol{\xi}) = \cos(2\pi x) + \left(\sin(2\pi x), \frac{1}{2} \sin(4\pi x), \frac{1}{3} \sin(6\pi x) \right) \mathring{\mathbf{A}}^T \mathbf{H}_{\mathfrak{J}}^T,$$

where $\mathfrak{J} = \{\boldsymbol{\alpha} \mid \boldsymbol{\alpha} \in \mathfrak{J}_3^4, \alpha_3 \leq 3\} \setminus \{\mathbf{0}\}$. Matrix $\mathring{\mathbf{A}}$ is generated randomly and its columns a_i 's are orthogonal to each other and have unit length.

With time step $\delta t = 1.0 \times 10^{-3}$ and spatial grid size $\delta x = 1/128$, the spatial modes computed by DyBO match perfectly the exact ones given by gPC as shown in Fig. 3.8. In Fig. 3.9, the \mathbb{L}^2 relative error of STD is plotted as a function of time t and remains very small ($\leq 10^{-8}$). To confirm

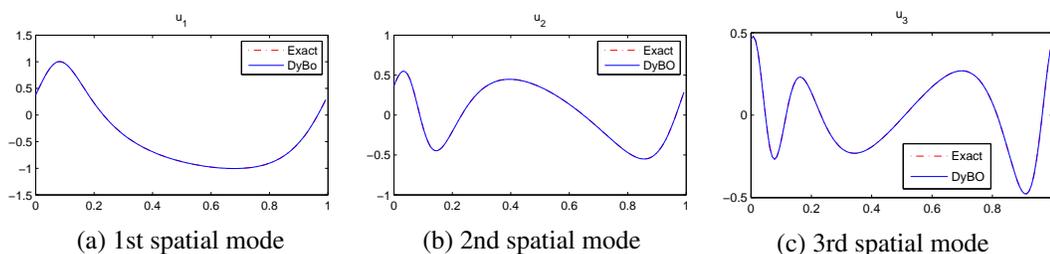


Figure 3.8: Spatial modes in KL expansion of SPDE solution computed by DyBO at time $t = 0.4$

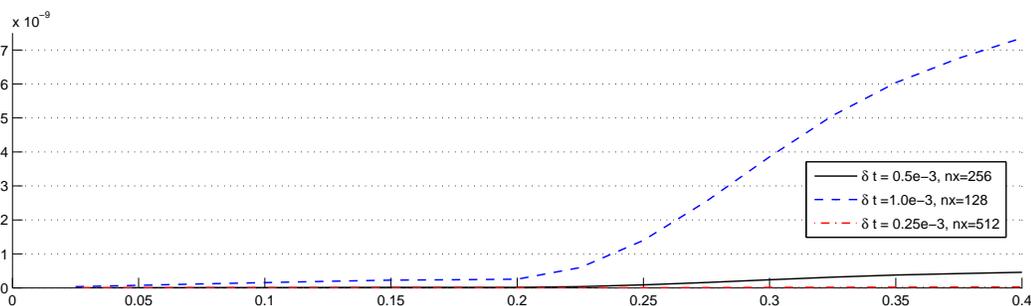


Figure 3.9: \mathbb{L}^2 relative errors of STD computed by DyBO with different spatial and temporal grid sizes. The horizontal axis is time t

such errors are introduced mainly by spatial and temporal discretizations, we use another two sets of finer grid sizes, $\delta t = 0.5 \times 10^{-3}$, $\delta x = 1/256$, and, $\delta t = 0.25 \times 10^{-3}$, $\delta x = 1/512$ and repeat the computations. The STD error drops significantly since we use spectral methods and fourth order RK method, which essentially verifies numerically Corollary 2.6.

3.3 Burgers' Equation Driven by Stochastic Forces

Having demonstrated the success of DyBO-gPC method for relatively simple SPDEs, we should move on to nonlinear SPDEs. Stochastic Burgers equations immediately catch our attention not only because many successful numerical schemes have used deterministic Burgers equations as a benchmark, but also because of the rich structures stemming out of the interaction between nonlinearity and randomness in stochastic Burger equations and their applications in statistical mechanics, such as interfacial dynamics and directed polymers in random media, and in the context of turbulence. For more discussion regarding the stochastic Burgers equations and their applications, please

see [13, 81, 55, 77, 63, 9, 31, 30, 22, 61, 91, 15] and the reference therein.

Consider a one-dimensional Burgers' equation driven by zero-mean stochastic force $f(x, t, \xi(\omega))$,

$$\frac{\partial u}{\partial t} = \mathcal{L}u = \mathring{\mathcal{L}}u + f = -u \frac{\partial u}{\partial x} + \nu \frac{\partial^2 u}{\partial x^2} + f, \quad x \in [0, 1], t \in [0, T], \quad (3.10a)$$

$$u(x, 0, \xi) = \mathring{u}(x), \quad (3.10b)$$

$$u(0, t, \xi) = u(1, t, \xi), \quad (3.10c)$$

where the initial condition is deterministic and the boundary condition is periodic. Here, we call $\mathring{\mathcal{L}}u = -u \frac{\partial u}{\partial x} + \nu \frac{\partial^2 u}{\partial x^2}$ the deterministic Burgers' differential operator or Burgers' operator in short. To ensure the stochastic solution does not blow up in a finite time, we assume that $f(x, t, \xi) \in \mathbb{L}^2(\mathcal{D} \times \Omega)$ at any fixed time t and the stochastic force admits a finite gPC expansion, i.e., $f(x, t, \xi) = \mathbf{F}(x, t) \mathbf{H}(\xi)^T$. where row vector $\mathbf{F} = (F_\alpha)_{\alpha \in \mathfrak{J}}$ for some multi-index set \mathfrak{J} and $|\mathfrak{J}| = N_P$.

Remark 3.2. When Burgers' equation is driven by stochastic processes, e.g., Brownian motions, the above formulation still provides a good approximate model. The exact form of stochastic force f can be obtained by choosing a certain orthonormal basis on $\mathbb{L}^2([0, T])$ and projecting the Brownian path onto such a basis. See [55, 77, 32] or the below numerical examples for details.

3.3.1 gPC Formulation for Stochastic Burgers' Equation

We first derive the gPC formulation of (3.10). In the thesis by Dr. Wuan Luo [77], the gPC formulation of Burgers' equation driven by Brownian motion has been developed. He did not use vector/matrix notation and gave the gPC formulation in terms of u_α . Here, we repeat the derivation, but use the vector/matrix/tensor notation introduced earlier or in the course of the derivation. Com-

paring the two approaches, ours not only simplifies the derivations, but also more clearly reveals the physical structure of the problem. Similar benefits carry over to numerical implementations.

Consider the gPC expansion of solution $v = \bar{v} + \mathbf{V}\mathbf{H}^T$. Simple calculations give

$$\begin{aligned}\mathcal{L}v &= -(\bar{v} + \mathbf{V}\mathbf{H}^T) \left(\frac{\partial \bar{v}}{\partial x} + \frac{\partial \mathbf{V}}{\partial x} \mathbf{H}^T \right) + \nu \left(\frac{\partial^2 \bar{v}}{\partial x^2} + \frac{\partial^2 \mathbf{V}}{\partial x^2} \mathbf{H}^T \right) + \mathbf{F}\mathbf{H}^T \\ &= -\bar{v} \frac{\partial \bar{v}}{\partial x} - \bar{v} \frac{\partial \mathbf{V}}{\partial x} \mathbf{H}^T - \frac{\partial \bar{v}}{\partial x} \mathbf{V}\mathbf{H}^T - \mathbf{V}\mathbf{H}^T \mathbf{H} \frac{\partial \mathbf{V}^T}{\partial x} + \nu \frac{\partial^2 \bar{v}}{\partial x^2} + \nu \frac{\partial^2 \mathbf{V}}{\partial x^2} \mathbf{H}^T + \mathbf{F}\mathbf{H}^T \\ &= \mathring{\mathcal{L}}\bar{v} + \left(\nu \frac{\partial^2 \mathbf{V}}{\partial x^2} - \frac{\partial (\bar{v}\mathbf{V})}{\partial x} \right) \mathbf{H}^T + \mathbf{F}\mathbf{H}^T - \mathbf{V}\mathbf{H}^T \mathbf{H} \frac{\partial \mathbf{V}^T}{\partial x}.\end{aligned}$$

where $\frac{\partial \mathbf{V}}{\partial x}$ and $\frac{\partial^2 \mathbf{V}}{\partial x^2}$ are understood as row vectors $\left(\frac{\partial v_\alpha}{\partial x} \right)_{\alpha \in \mathfrak{J}}$ and $\left(\frac{\partial^2 v_\alpha}{\partial x^2} \right)_{\alpha \in \mathfrak{J}}$, respectively. By using

$\mathbb{E}[\mathbf{H}] = \mathbf{0}$ and $\mathbb{E}[\mathbf{H}^T \mathbf{H}] = \mathbf{I}$, we have

$$\begin{aligned}\mathbb{E}[\mathcal{L}v] &= \mathring{\mathcal{L}}\bar{v} - \mathbf{V} \frac{\partial \mathbf{V}^T}{\partial x}, \\ \tilde{\mathcal{L}}v &= \left(\nu \frac{\partial^2 \mathbf{V}}{\partial x^2} - \frac{\partial (\bar{v}\mathbf{V})}{\partial x} \right) \mathbf{H}^T + \mathbf{F}\mathbf{H}^T + \mathbf{V} \frac{\partial \mathbf{V}^T}{\partial x} - \mathbf{V}\mathbf{H}^T \mathbf{H} \frac{\partial \mathbf{V}^T}{\partial x},\end{aligned}$$

and

$$\begin{aligned}\mathbb{E}[\tilde{\mathcal{L}}v\mathbf{H}] &= \left(\nu \frac{\partial^2 \mathbf{V}}{\partial x^2} - \frac{\partial (\bar{v}\mathbf{V})}{\partial x} \right) - \mathbb{E} \left[\mathbf{V}\mathbf{H}^T \mathbf{H} \frac{\partial \mathbf{V}^T}{\partial x} \mathbf{H} \right] + \mathbf{F} \\ &= \left(\nu \frac{\partial^2 \mathbf{V}}{\partial x^2} - \frac{\partial (\bar{v}\mathbf{V})}{\partial x} \right) - \left(v_\alpha \frac{\partial v_\beta}{\partial x} \mathbb{E}[\mathbf{H}_\alpha \mathbf{H}_\beta \mathbf{H}_\gamma] \right)_{1\gamma} + \mathbf{F} \\ &= \left(\nu \frac{\partial^2 \mathbf{V}}{\partial x^2} - \frac{\partial (\bar{v}\mathbf{V})}{\partial x} \right) - \left(v_\alpha \frac{\partial v_\beta}{\partial x} \mathfrak{T}_{\alpha\beta\gamma}^{(\mathbf{H})} \right)_{1\gamma} + \mathbf{F},\end{aligned}$$

where we have used the Einstein summation convention and tensor $\mathfrak{T}^{(\mathbf{H})}$ to simplify notations.

Here $(\cdot)_{1\gamma}$ is used to emphasize that it is a row vector. In the last equality, $\mathfrak{T}^{(\mathbf{H})}$ is a third-order

N_P -by- N_P -by- N_P tensor

$$\mathfrak{T}_{\alpha\beta\gamma}^{(\mathbf{H})} = \mathbb{E} [\mathbf{H}_\alpha \mathbf{H}_\beta \mathbf{H}_\gamma], \quad \alpha, \beta, \gamma \in \mathfrak{J},$$

which only depends on the set of orthonormal polynomials and the multi-index set \mathfrak{J} , and can be computed ahead. Clearly, $\mathfrak{T}^{(\mathbf{H})}$ is symmetric with respect to all of three indices, i.e., $\mathfrak{T}_{\alpha\beta\gamma}^{(\mathbf{H})} = \mathfrak{T}_{\text{perm}(\alpha\beta\gamma)}^{(\mathbf{H})}$ for $\forall \alpha, \beta, \gamma \in \mathfrak{J}$, where $\text{perm}(\alpha\beta\gamma)$ is any permutation of multi-indices α, β, γ .

From eqn. (2.36) and the above calculations, we arrive at the gPC formulation for stochastic Burgers equation (3.10a),

$$\frac{\partial \bar{v}}{\partial t} = \mathring{\mathcal{L}} \bar{v} - \mathbf{V} \frac{\partial \mathbf{V}^T}{\partial x}, \quad (3.11a)$$

$$\frac{\partial \mathbf{V}}{\partial t} = \left(\nu \frac{\partial^2 \mathbf{V}}{\partial x^2} - \frac{\partial (\bar{v} \mathbf{V})}{\partial x} \right) - \left(v_\alpha \frac{\partial v_\beta}{\partial x} \mathfrak{T}_{\alpha\beta\gamma}^{(\mathbf{H})} \right)_{1\gamma} + \mathbf{F}. \quad (3.11b)$$

From the above derivation of the gPC formulation, it is easy to see that the usage of vector and tensor notations not only greatly reduces the complexity of the derivation, but also clearly reveals the structure of the gPC formulation and its relation to the deterministic Burgers equation. For example, the mean flow \bar{v} is still driven by the deterministic Burgers' differential operator $\mathring{\mathcal{L}}$ and exchanges energy with stochastic flows \mathbf{V} through the term $-\mathbf{V} \frac{\partial \mathbf{V}^T}{\partial x}$. The stochastic flows \mathbf{V} are convected by the mean flow through term $-\frac{\partial (\bar{v} \mathbf{V})}{\partial x}$, dissipate through term $\nu \frac{\partial^2 \mathbf{V}}{\partial x^2}$ and interact with each others through $-\left(v_\alpha \frac{\partial v_\beta}{\partial x} \mathfrak{T}_{\alpha\beta\gamma}^{(\mathbf{H})} \right)_{1\gamma}$.

What's more, the usage of vector notation also enables fast, reliable and efficient numerical implementations. Nowadays, there are plenty of reliable and efficient software packages or libraries available in various computer languages for all kinds of numerical tasks ranging from basic vector-matrix multiplications to sophisticated Finite Element Methods (FEM). Instead of writing numerical programs from scratch, developing new algorithms based on the existing reliable packages has

proven to be an efficient and cost-effective way. In our case, the usage of vector notation enables us to write codes in matlab in a way that is very close to native mathematical expressions. For object-orientated programming languages, such as C++, Java, and C#, similar effects can be achieved if good packages for basic numerical tasks are adopted, which in turn reduces developing time and maintenance burdens.

Similar benefits for DyBO methods also stem out of the usage of the vector and tensor notations as we can see from the next section.

3.3.2 DyBO-gPC Formulation for Stochastic Burgers' Equation

Now consider the m -term truncated KL expansion of the stochastic Burgers equation solution $u = \bar{u} + \mathbf{U}\mathbf{Y}^T = \bar{u} + \mathbf{U}\mathbf{A}^T\mathbf{H}^T$. Again, simple calculations give

$$\begin{aligned}
\mathcal{L}u &= -(\bar{u} + \mathbf{U}\mathbf{Y}^T) \left(\frac{\partial \bar{u}}{\partial x} + \frac{\partial \mathbf{U}}{\partial x} \mathbf{Y}^T \right) + \nu \left(\frac{\partial^2 \bar{u}}{\partial x^2} + \frac{\partial^2 \mathbf{U}}{\partial x^2} \mathbf{Y}^T \right) + f \\
&= -\bar{u} \frac{\partial \bar{u}}{\partial x} - \bar{u} \frac{\partial \mathbf{U}}{\partial x} \mathbf{Y}^T - \frac{\partial \bar{u}}{\partial x} \mathbf{U} \mathbf{Y}^T - \mathbf{U} \mathbf{Y}^T \mathbf{Y} \frac{\partial \mathbf{U}^T}{\partial x} + \nu \frac{\partial^2 \bar{u}}{\partial x^2} + \nu \frac{\partial^2 \mathbf{U}}{\partial x^2} \mathbf{Y}^T + f \\
&= -\bar{u} \frac{\partial \bar{u}}{\partial x} + \nu \frac{\partial^2 \bar{u}}{\partial x^2} + \left(\nu \frac{\partial^2 \mathbf{U}}{\partial x^2} - \bar{u} \frac{\partial \mathbf{U}}{\partial x} - \frac{\partial \bar{u}}{\partial x} \mathbf{U} \right) \mathbf{Y}^T - \mathbf{U} \mathbf{Y}^T \mathbf{Y} \frac{\partial \mathbf{U}^T}{\partial x} + f \\
&= \mathring{\mathcal{L}}\bar{u} + \left(\nu \frac{\partial^2 \mathbf{U}}{\partial x^2} - \frac{\partial (\bar{u}\mathbf{U})}{\partial x} \right) \mathbf{Y}^T - \mathbf{U} \mathbf{Y}^T \mathbf{Y} \frac{\partial \mathbf{U}^T}{\partial x} + f.
\end{aligned}$$

So

$$\mathbb{E}[\mathcal{L}u] = \mathring{\mathcal{L}}\bar{u} - \mathbf{U} \frac{\partial \mathbf{U}^T}{\partial x},$$

where we have used the properties of zero-mean stochastic processes \mathbf{Y} , i.e., $\mathbb{E}[\mathbf{Y}^T \mathbf{Y}] = \mathbf{I}$ and $\mathbb{E}[\mathbf{Y}] = 0$, and also $\mathbb{E}[f] = 0$. Similar calculations give

$$\begin{aligned}\tilde{\mathcal{L}}u &= \left(\nu \frac{\partial^2 \mathbf{U}}{\partial x^2} - \frac{\partial(\bar{u}\mathbf{U})}{\partial x} \right) \mathbf{Y}^T - \mathbf{U} \mathbf{Y}^T \mathbf{Y} \frac{\partial \mathbf{U}^T}{\partial x} + \mathbf{U} \frac{\partial \mathbf{U}^T}{\partial x} + f, \\ \mathbb{E}[\tilde{\mathcal{L}}u \mathbf{Y}] &= \left(\nu \frac{\partial^2 \mathbf{U}}{\partial x^2} - \frac{\partial(\bar{u}\mathbf{U})}{\partial x} \right) + \mathbb{E}[f \mathbf{Y}] - \mathbb{E}\left[\mathbf{U} \mathbf{Y}^T \mathbf{Y} \frac{\partial \mathbf{U}^T}{\partial x} \mathbf{Y} \right], \\ \langle \tilde{\mathcal{L}}u, \mathbf{U} \rangle &= \mathbf{Y} \left\langle \nu \frac{\partial^2 \mathbf{U}^T}{\partial x^2} - \frac{\partial(\bar{u}\mathbf{U}^T)}{\partial x}, \mathbf{U} \right\rangle + \langle f, \mathbf{U} \rangle \\ &\quad + \left\langle \mathbf{U} \frac{\partial \mathbf{U}^T}{\partial x}, \mathbf{U} \right\rangle - \left\langle \mathbf{U} \mathbf{Y}^T \mathbf{Y} \frac{\partial \mathbf{U}^T}{\partial x}, \mathbf{U} \right\rangle, \\ \Lambda_{\mathbf{U}} G_*(u, \mathbf{U}, \mathbf{Y}) &= \left\langle \mathbf{U}^T, \nu \frac{\partial^2 \mathbf{U}}{\partial x^2} - \frac{\partial(\bar{u}\mathbf{U})}{\partial x} \right\rangle + \langle \mathbf{U}^T, \mathbb{E}[f \mathbf{Y}] \rangle - \left\langle \mathbf{U}^T, \mathbb{E}\left[\mathbf{U} \mathbf{Y}^T \mathbf{Y} \frac{\partial \mathbf{U}^T}{\partial x} \mathbf{Y} \right] \right\rangle,\end{aligned}$$

where the last terms on the right hand sides of the last three equations can be written in component forms, for $i, j, k = 1, 2, \dots, m$,

$$\begin{aligned}\mathbb{E}\left[\mathbf{U} \mathbf{Y}^T \mathbf{Y} \frac{\partial \mathbf{U}^T}{\partial x} \mathbf{Y} \right] &= \left(\mathbb{E}\left[u_i Y_i Y_j \frac{\partial u_j}{\partial x} Y_k \right] \right)_{1k} = \left(u_i \frac{\partial u_j}{\partial x} \mathbb{E}[Y_i Y_j Y_k] \right)_{1k}, \\ \left\langle \mathbf{U} \mathbf{Y}^T \mathbf{Y} \frac{\partial \mathbf{U}^T}{\partial x}, \mathbf{U} \right\rangle &= \left(\left\langle u_i Y_i Y_j \frac{\partial u_j}{\partial x}, u_k \right\rangle \right)_{1k} = \left(Y_i Y_j \left\langle u_i \frac{\partial u_j}{\partial x}, u_k \right\rangle \right)_{1k}, \\ \left\langle \mathbf{U}^T, \mathbb{E}\left[\mathbf{U} \mathbf{Y}^T \mathbf{Y} \frac{\partial \mathbf{U}^T}{\partial x} \mathbf{Y} \right] \right\rangle &= \left(\left\langle u_i \frac{\partial u_j}{\partial x}, u_l \right\rangle \mathbb{E}[Y_i Y_j Y_k] \right)_{lk},\end{aligned}$$

where $(\cdot)_{1k}$ and $(\cdot)_{lk}$ emphasize row vector and m -by- m matrix, respectively. We write third-order m -by- m -by- m tensors

$$\begin{aligned}\mathfrak{F}_{ijk}^{(\mathbf{Y})} &= \mathbb{E}[Y_i Y_j Y_k], \\ \mathfrak{F}_{ijl}^{(\mathbf{U})} &= \left\langle u_i \frac{\partial u_j}{\partial x}, u_l \right\rangle \quad i, j, k, l = 1, 2, \dots, m.\end{aligned}$$

Clearly, $\mathfrak{T}^{(\mathbf{Y})}$ is symmetric with respect to all three indices, i.e., $\mathfrak{T}_{ijk}^{(\mathbf{Y})} = \mathfrak{T}_{\text{perm}(ijk)}^{(\mathbf{Y})}$ for $\forall i, j, k \in \{1, 2, \dots, m\}$, where $\text{perm}(ijk)$ is any permutation of indices i, j, k , and $\mathfrak{T}^{(\mathbf{U})}$ is symmetric with respect to the first and the third indices, i.e., $\mathfrak{T}_{ijk}^{(\mathbf{U})} = \mathfrak{T}_{kji}^{(\mathbf{U})}$ for $\forall i, k \in \{1, 2, \dots, m\}$. Such symmetries can be explored in numerical implementations to achieve further computational reductions.

Using eqn. (2.28), we obtain the DyBO formulation,

$$\frac{\partial \bar{u}}{\partial t} = \mathring{\mathcal{L}}\bar{u} - \mathbf{U} \frac{\partial \mathbf{U}^T}{\partial x}, \quad (3.12a)$$

$$\frac{\partial \mathbf{U}}{\partial t} = -\mathbf{U} \mathbf{D}^T + \left(\nu \frac{\partial^2 \mathbf{U}}{\partial x^2} - \frac{\partial (\bar{u} \mathbf{U})}{\partial x} \right) - \left(u_i \frac{\partial u_j}{\partial x} \mathfrak{T}_{ijk}^{(\mathbf{Y})} \right)_{1k} + \mathbb{E}[f \mathbf{Y}], \quad (3.12b)$$

$$\begin{aligned} \frac{d\mathbf{Y}}{dt} = \mathbf{Y} \left(-\mathbf{C}^T + \left\langle \nu \frac{\partial^2 \mathbf{U}^T}{\partial x^2} - \frac{\partial (\bar{u} \mathbf{U}^T)}{\partial x}, \mathbf{U} \right\rangle \mathbf{\Lambda}_{\mathbf{U}}^{-1} \right) + \left(\mathfrak{T}_{iik}^{(\mathbf{U})} \right)_{1k} \mathbf{\Lambda}_{\mathbf{U}}^{-1} \\ - \left(Y_i Y_j \mathfrak{T}_{ijk}^{(\mathbf{U})} \right)_{1k} \mathbf{\Lambda}_{\mathbf{U}}^{-1} + \langle f, \mathbf{U} \rangle \mathbf{\Lambda}_{\mathbf{U}}^{-1}, \end{aligned} \quad (3.12c)$$

where

$$\mathbf{\Lambda}_{\mathbf{U}} G_*(u, \mathbf{U}, \mathbf{Y}) = \left\langle \mathbf{U}^T, \nu \frac{\partial^2 \mathbf{U}}{\partial x^2} - \frac{\partial (\bar{u} \mathbf{U})}{\partial x} \right\rangle - \left(\mathfrak{T}_{ijl}^{(\mathbf{U})} \mathfrak{T}_{ijk}^{(\mathbf{Y})} \right)_{lk} + \langle \mathbf{U}^T, \mathbb{E}[f \mathbf{Y}] \rangle.$$

Similarly, we can derive the DyBO-gPC formulation,

$$\begin{aligned}
\tilde{\mathcal{L}}u &= \left(\nu \frac{\partial^2 \mathbf{U}}{\partial x^2} - \frac{\partial(\bar{u}\mathbf{U})}{\partial x} \right) \mathbf{A}^T \mathbf{H}^T \\
&\quad - \mathbf{U} \left(\mathbf{A}^T \mathbf{H}^T \mathbf{H} \mathbf{A} - \mathbb{E} [\mathbf{A}^T \mathbf{H}^T \mathbf{H} \mathbf{A}] \right) \frac{\partial \mathbf{U}^T}{\partial x} + \mathbf{F} \mathbf{H}^T \\
&= \left(\nu \frac{\partial^2 \mathbf{U}}{\partial x^2} - \frac{\partial(\bar{u}\mathbf{U})}{\partial x} \right) \mathbf{A}^T \mathbf{H}^T - \mathbf{U} \mathbf{A}^T \mathbf{H}^T \mathbf{H} \mathbf{A} \frac{\partial \mathbf{U}^T}{\partial x} + \mathbf{U} \frac{\partial \mathbf{U}^T}{\partial x} + \mathbf{F} \mathbf{H}^T, \\
\mathbb{E} [\tilde{\mathcal{L}}u \mathbf{H}] &= \left(\nu \frac{\partial^2 \mathbf{U}}{\partial x^2} - \frac{\partial(\bar{u}\mathbf{U})}{\partial x} \right) \mathbf{A}^T - \mathbb{E} \left[\mathbf{U} \mathbf{A}^T \mathbf{H}^T \mathbf{H} \mathbf{A} \frac{\partial \mathbf{U}^T}{\partial x} \mathbf{H} \right] \\
&= \left(\nu \frac{\partial^2 \mathbf{U}}{\partial x^2} - \frac{\partial(\bar{u}\mathbf{U})}{\partial x} \right) \mathbf{A}^T - \left(u_i \frac{\partial u_j}{\partial x} A_{\alpha i} A_{\beta j} \mathfrak{T}_{\alpha\beta\gamma}^{(\mathbf{H})} \right)_{1\gamma} + \mathbf{F}, \\
\langle \mathbb{E} [\tilde{\mathcal{L}}u \mathbf{H}^T], \mathbf{U} \rangle &= \mathbf{A} \left\langle \nu \frac{\partial^2 \mathbf{U}^T}{\partial x^2} - \frac{\partial(\bar{u}\mathbf{U}^T)}{\partial x}, \mathbf{U} \right\rangle - \left(A_{\alpha i} A_{\beta j} \mathfrak{T}_{ijk}^{(\mathbf{U})} \mathfrak{T}_{\alpha\beta\gamma}^{(\mathbf{H})} \right)_{\gamma k} + \langle \mathbf{F}^T, \mathbf{U} \rangle,
\end{aligned}$$

where $\mathbb{E} [\mathbf{Y}^T \mathbf{Y}] = \mathbf{A}^T \mathbf{A} = \mathbf{I}$. From eqn. (2.39), we arrive at the DyBO-gPC formulation of the stochastic Burgers equation (3.10),

$$\frac{\partial \bar{u}}{\partial t} = \mathring{\mathcal{L}}\bar{u} - \mathbf{U} \frac{\partial \mathbf{U}^T}{\partial x}, \quad (3.13a)$$

$$\frac{\partial \mathbf{U}}{\partial t} = -\mathbf{U} \mathbf{D}^T + \left(\nu \frac{\partial^2 \mathbf{U}}{\partial x^2} - \frac{\partial(\bar{u}\mathbf{U})}{\partial x} \right) - \left(u_i \frac{\partial u_j}{\partial x} A_{\alpha i} A_{\beta j} A_{\gamma k} \mathfrak{T}_{\alpha\beta\gamma}^{(\mathbf{H})} \right)_{1k} + \mathbf{F} \mathbf{A}, \quad (3.13b)$$

$$\begin{aligned}
\frac{d\mathbf{A}}{dt} &= \mathbf{A} \left(-\mathbf{C}^T + \left\langle \nu \frac{\partial^2 \mathbf{U}^T}{\partial x^2} - \frac{\partial(\bar{u}\mathbf{U}^T)}{\partial x}, \mathbf{U} \right\rangle \mathbf{\Lambda}_{\mathbf{U}}^{-1} \right) \\
&\quad - \left(A_{\alpha i} A_{\beta j} \mathfrak{T}_{ijk}^{(\mathbf{U})} \mathfrak{T}_{\alpha\beta\gamma}^{(\mathbf{H})} \right)_{\gamma k} \mathbf{\Lambda}_{\mathbf{U}}^{-1} + \langle \mathbf{F}^T, \mathbf{U} \rangle \mathbf{\Lambda}_{\mathbf{U}}^{-1}, \quad (3.13c)
\end{aligned}$$

where matrices \mathbf{C} and \mathbf{D} can be solved from the linear system (2.21) with

$$\mathbf{\Lambda}_{\mathbf{U}} G_*(u, \mathbf{U}, \mathbf{Y}) = \left\langle \mathbf{U}^T, \nu \frac{\partial^2 \mathbf{U}}{\partial x^2} - \frac{\partial(\bar{u}\mathbf{U})}{\partial x} \right\rangle - \left(\mathfrak{T}_{ijl}^{(\mathbf{U})} A_{\alpha i} A_{\beta j} A_{\gamma k} \mathfrak{T}_{\alpha\beta\gamma}^{(\mathbf{H})} \right)_{lk} + \langle \mathbf{U}^T, \mathbf{F} \rangle \mathbf{A}.$$

In the following numerical examples, the fourth-order RK method is used as the numerical integrator and a pseudo-spectral method is applied to the first two equations, i.e., eqn. (3.13a) and

eqn. (3.13b), for spatial discretizations. See [94], [44] and [16] for details of these standard numerical methods.

3.3.3 Numerical Examples

By using the Cole-Hopf transformation, one can show that the deterministic Burgers equation can be transformed to a heat equation. Thus, this model cannot be used to study Navier-Stokes (NS) turbulence since it lacks an essential property: sensitivity to small perturbations [10, 9]. However, recent years have witnessed a renewed interests in Burgers turbulence resulting from Burgers models driven by stochastic forces, especially Brownian motions [77, 55, 112, 52], i.e.,

$$\frac{\partial u}{\partial t} = \mathcal{L}u = \mathring{\mathcal{L}}u + \frac{dB_t}{dt}, \quad x \in [0, 1], \quad t \in [0, T], \quad (3.14a)$$

$$u(x, 0, \xi) = \mathring{u}(x), \quad (3.14b)$$

$$u(0, t, \xi) = u(1, t, \xi). \quad (3.14c)$$

where the first SPDE is understood in the Itô sense. See [67, 66] and references therein for details of Itô calculus.

To demonstrate the success of DyBO and Adaptive-DyBO, we consider here a stochastic force f approximating the Brownian force $\frac{dB_t}{dt}$. The construction of such stochastic force f follows that in Luo's paper [55]. We start by selecting a complete orthonormal basis $\{M_i(t)\}_{i=1}^{\infty} \subset \mathbb{L}^2([0, 1])$, which induces an orthonormal basis $\{\tilde{M}_i(t) = \frac{1}{\sqrt{T}}M_i(\frac{t}{T})\}_{i=1}^{\infty} \subset \mathbb{L}^2([0, T])$. To facilitate further discussion, we write the characteristic function of time interval $[0, t]$, $t \in [0, 1]$,

$$\mathbb{1}_{[0,t]}(s) = \begin{cases} 1, & s < t, \\ 0, & s \geq t, \end{cases}$$

and its Fourier series under the orthonormal basis M_i ,

$$\mathbb{1}_{[0,t)}(s) = \sum_{i=1}^{\infty} c_i(t) M_i(s),$$

where $c_i(t) = \int_0^t M_i(s) ds$. It is easy to show that the characteristic function of time interval $[0, t)$ for $t \in [0, T]$ has a Fourier-type expansion

$$\mathbb{1}_{[0,t)}(s) = \sum_{i=1}^{\infty} \tilde{c}_i(t) \tilde{M}_i(s),$$

where $\tilde{c}_i(t) = \sqrt{T} c_i(\frac{t}{T})$. Substituting the above expansion into the obvious Ito integral

$$B_t = \int_0^T \mathbb{1}_{[0,t)}(s) dB_s,$$

we have

$$B_t = \sum_{i=1}^{\infty} \tilde{c}_i(t) \int_0^T \tilde{M}_i(s) dB_s = \sum_{i=1}^{\infty} \tilde{c}_i(t) \xi_i. \quad (3.15)$$

where random variables ξ_i 's are independent standard Gaussian random variables because of isometry. For further details, see the thesis by Dr. Luo [77], lecture notes by Evans ¹, the book by Holden et al. [52] and the reference therein for details. Truncating the above series after N_p terms, we get the stochastic force f used in the following numerical examples,

$$\begin{aligned} \frac{dB_t}{dt} &\approx f = \sum_{i=1}^{N_p} \frac{1}{\sqrt{T}} M_i\left(\frac{t}{T}\right) \xi_i \\ &= \sum_{i=1}^{N_p} \frac{1}{\sqrt{T}} M_i\left(\frac{t}{T}\right) \mathbf{H} \mathbf{e}_i \\ &= \left(\frac{1}{\sqrt{T}} M_1\left(\frac{t}{T}\right), \frac{1}{\sqrt{T}} M_2\left(\frac{t}{T}\right), \dots, \frac{1}{\sqrt{T}} M_{N_p}\left(\frac{t}{T}\right), 0, 0, \dots, 0 \right) \mathbf{H}^T, \end{aligned} \quad (3.16)$$

¹available at <http://math.berkeley.edu/~evans/SDE.course.pdf>.

where \mathbf{e}_i is a multi-index of length N_p whose i th entry is 1 and others are zeros.

Remark 3.3. This form of stochastic force f induces an interesting phenomena in the gPC formulation of stochastic Burgers equations: the randomness is first “injected” into the system through low-order gPC coefficients $v_{\mathbf{e}_i}$ and then spreads into the mean \bar{v} and other high-order gPC coefficients.

3.3.3.1 Hierarchy of Errors

The Burgers equation driven by Brownian motions serves as a perfect example to explain the relations among the exact solution $u^{(\text{Exact})}$ of eqn. (3.14), the gPC solutions $u^{(\text{gPC}, \mathfrak{J}^\infty)}$, $u^{(\text{gPC}, \mathfrak{J}_{N_p}^\infty)}$, $u^{(\text{gPC}, \mathfrak{J}_{N_p}^P)}$ given by eqn. (3.11) for different multi-index set \mathfrak{J} 's and the DyBO solutions $u^{(\text{DyBO}, \mathfrak{J}_{N_p}^P, m)}$, $u^{(\text{DyBO}, \mathfrak{J}_{N_p}^P, \tilde{M})}$ from eqn. (3.13), i.e., error hierarchy, see Fig. 3.10. When no truncation is made in the expansion of Brownian path (3.15) and Hermite polynomials of all orders are used in the gPC formulation (3.13), the solution $u^{(\text{gPC}, \mathfrak{J}^\infty)}$ is equal to the exact solution $u^{(\text{Exact})}$. When only finite number of random variables are used to approximate Brownian path, i.e., $N_p < \infty$, the solution $u^{(\text{gPC}, \mathfrak{J}_{N_p}^\infty)}$ approximates $u^{(\text{gPC}, \mathfrak{J}^\infty)}$.

Clearly, two aforementioned solutions are computationally intractable because an infinite number of terms are involved. To get a computational feasible gPC formulation, we can further trim down the multi-index set $\mathfrak{J}_{N_p}^\infty$. There are multiple ways of selecting such multi-index sets of finite size. Here, we choose polynomials of total orders at most P , i.e., $\mathfrak{J}_{N_p}^P$. For other selections of multi-index sets, the discussion stays the same. $u^{(\text{gPC}, \mathfrak{J}_{N_p}^P)}$ is a good approximation to $u^{(\text{gPC}, \mathfrak{J}_{N_p}^\infty)}$ when P is large enough [115, 117, 106]. Our DyBO solution $u^{(\text{DyBO}, \mathfrak{J}_{N_p}^P, m)}$ computed from eqn. (3.13) then approximates $u^{(\text{gPC}, \mathfrak{J}_{N_p}^P)}$ when m is large enough.

This error hierarchy shown in Fig. 3.10 has two profound implications regarding the numerical computation of SPDE solutions.

- We ultimately want to reduce the error between the exact solution $u^{(\text{Exact})}$ and the DyBO solution $u^{(\text{DyBO}, \mathfrak{J}_{N_P}^P, m)}$, which can be decomposed into two parts,

$$\begin{aligned} \mathcal{E} = u^{(\text{DyBO}, \mathfrak{J}_{N_P}^P, m)} - u^{(\text{Exact})} &= \left\{ u^{(\text{DyBO}, \mathfrak{J}_{N_P}^P, m)} - u^{(\text{gPC}, \mathfrak{J}_{N_P}^P)} \right\} + \left\{ u^{(\text{gPC}, \mathfrak{J}_{N_P}^P)} - u^{(\text{Exact})} \right\} \\ &= \mathcal{E}_m + \mathcal{E}_{\mathfrak{J}}. \end{aligned}$$

The first part of the error \mathcal{E}_m diminishes as $m \rightarrow \infty$, while the second part $\mathcal{E}_{\mathfrak{J}}$ is controlled by the multi-index set \mathfrak{J} and goes to 0 as $\mathfrak{J} \rightarrow \mathfrak{J}_{\infty}^{\infty}$. Thus, there is no need to increase m any further once $\mathcal{E}_m \ll \mathcal{E}_{\mathfrak{J}}$. We write m^* for such m and will verify the convergence of \mathcal{E}_m and calculate m^* later.

- Numerical integration of the DyBO system (3.13) involves considerably less computational work than that of the gPC system (3.11). Roughly speaking, for the gPC method, N_P gPC coefficients, functions of spatial variable x , are updated at each time iteration. However, for the DyBO method, only m spatial modes, still functions of spatial variable x , and one N_P -by- m matrix are updated at each time iteration. Generally, m is much smaller than N_P and spatial grid number, which leads to sizable computational saving. Computational complexity analysis for linear and second-order nonlinear PDEs driven by stochastic forces is discussed in detail in Chapter 4 accompanied by numerical examples of spatially two-dimensional stochastic Navier-Stokes equations. Such reduction in computational cost enables DyBO methods to achieve small error \mathcal{E} by using more random variables and higher-order polynomials, i.e., $\tilde{N}_{\tilde{P}} > N_P$ and $\tilde{P} > P$, without exceeding available computational resources, as illustrated at the top of Fig. 3.10.

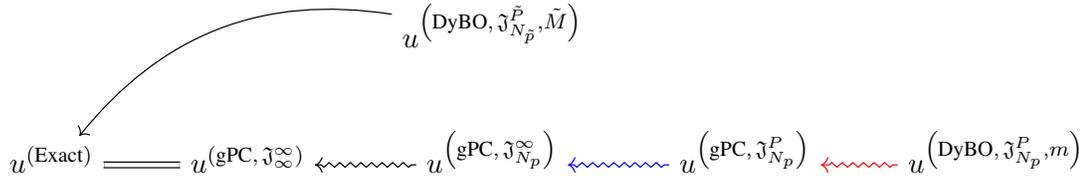


Figure 3.10: Hierarchy of solutions

3.3.3.2 Numerical Results

We have chosen the following orthonormal basis for $\mathbb{L}^2([0, T])$,

$$M_1(t) = 1,$$

$$M_i(t) = \sqrt{2} \cos((i-1)\pi t), \quad i = 2, 3, \dots$$

$N_p = 6$ and multi-index set $\mathfrak{J} = \mathfrak{J}_6^3 \setminus \{\mathbf{0}\}$, which results in totally 83 terms in gPC expansion, i.e., $|\mathfrak{J}| = 83$. Spatial grid size δx is set to $1/128$ and time step δt is set to 0.5×10^{-3} for both gPC and DyBO, both of which are numerically integrated to time $t = T = 1.0$. 10^6 realizations are computed in the MC method to approximate $u^{(\text{gPC}, \mathfrak{J}_{N_p}^\infty)}$ with error of order less than 10^{-3} according to the Central Limit theorem [66, 114].

Convergence to $u^{(\text{gPC}, \mathfrak{J}_{N_p}^P)}$. In Fig. 3.11, we plot the \mathbb{L}^2 relative errors of mean and STD computed by DyBO with $m = 4, 6, 8, 10, 12$ with respect to the gPC solution $u^{(\text{gPC}, \mathfrak{J}_{N_p}^P)}$. Indeed, as the number of mode pairs in DyBO increases, the \mathbb{L}^2 errors of mean and STD decreases, indicating the convergence of $u^{(\text{DyBO}, \mathfrak{J}_{N_p}^P, m)}$ to $u^{(\text{gPC}, \mathfrak{J}_{N_p}^P)}$. The relative errors of mean and STD at time $t = 1.0$ are also tabulated in the second and third columns of Table 3.1. With $m = 12$, both errors are below 0.3%.

Adaptive-DyBO. In both Fig. 3.11 and Table 3.1, we also include results by using adaptive

strategies of Alg. 2.3 and Alg. 2.4 to remove and add spatial and stochastic mode pairs when necessary. In Fig. 3.11c, the number of mode pairs in DyBO system is plotted as a function of time, showing that proposed adaptive strategies use a small number of mode pairs at the beginning and automatically add new spatial and stochastic mode pairs when higher-order modes are excited later. In Fig. 3.12, the first nine spatial modes at time $t = T$, i.e., $u_i(x, T)$, $i = 1, 2, \dots, 9$, are plotted and compared with ones computed from $u^{(\text{gPC}, \mathfrak{J}_{N_p}^P)}$. Very good matches are achieved for the first six eigenfunctions (the first two rows in Fig. 3.12). Although the 7th, 8th, and 9th spatial modes are under more influences of unresolved mode pairs, i.e., the $(m + 1)$ th, $(m + 2)$ th, ..., good matches are still obtained for these spatial modes (the last row in Fig. 3.12).

In Fig. 3.13, we also plot the stochastic modes $\mathbf{Y} = \mathbf{H}\mathbf{A}$. Since we never use directly the polynomial chaos set \mathbf{H} in our computation, we plot in Fig. 3.13b the N_P -by- m matrix \mathbf{A} computed in our DyBO method and in Fig. 3.13c the one computed from the gPC solution, respectively. The stochastic modes are plotted for time $t = T$. The meaning of both figures may deserve some further explanations. In Fig. 3.13a, the vertical axis from the top to the bottom is the multi-index $\alpha \in \mathfrak{J}$, while the horizontal axis from the left to the right is the index of stochastic mode $i = 1, 2, \dots, m$. In this setting, each column of such a plot represents a single column a_j of matrix \mathbf{A} , i.e., a stochastic mode $Y_j = \mathbf{H}a_j$. Similarly, each row of such plot represent the projections of all stochastic modes \mathbf{Y} on a certain polynomial basis \mathbf{H}_β , i.e., $\mathbb{E}[\mathbf{Y}\mathbf{H}_\beta]$.

From Fig. 3.12 and Fig. 3.13, we see that the solution of DyBO can effectively track the truncated KL expansion of the SPDE solution even if the differential operator is nonlinear.

Computational Speedup compared to gPC. Next we consider the efficiency of DyBO compared to gPC. In the fourth and fifth columns of Table 3.1, we also tabulate the relative errors of mean and STD computed by DyBO, with $m = 4, 6, 8, 10, 12$ and adaptive strategy, with respect to the exact solution $u^{(\text{gPC}, \mathfrak{J}_{N_p}^\infty)}$ obtained by MC method with 10^6 realizations. The relative errors of

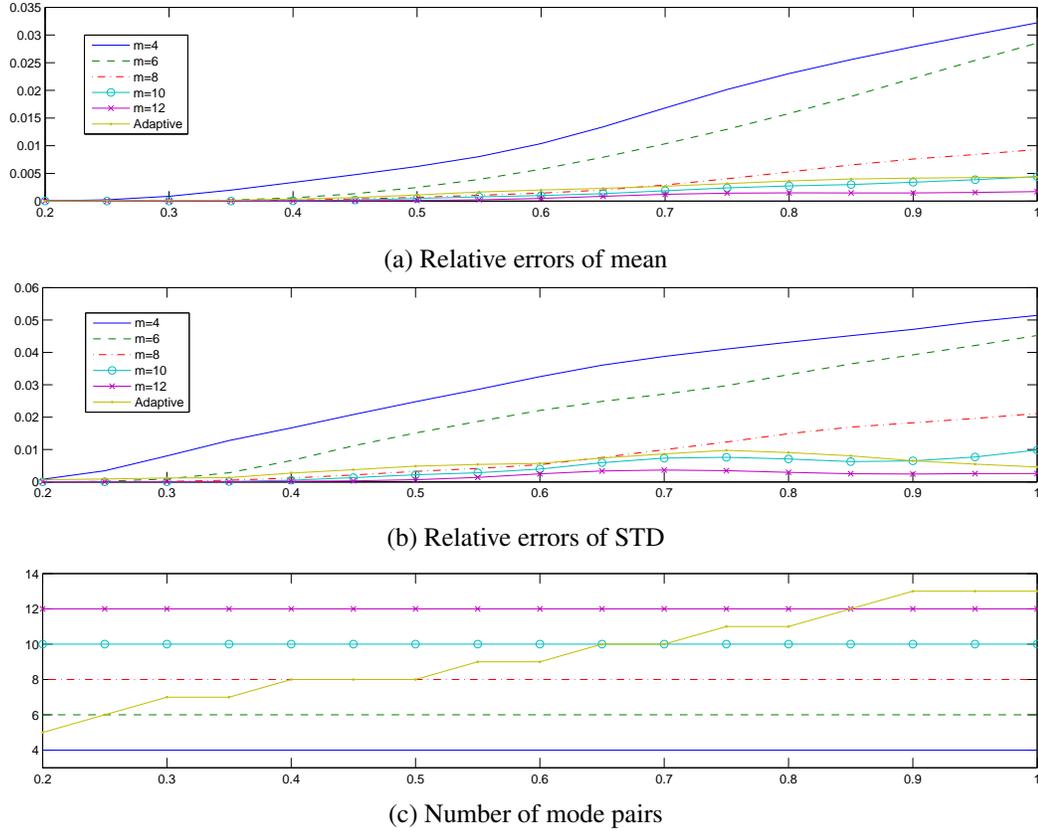
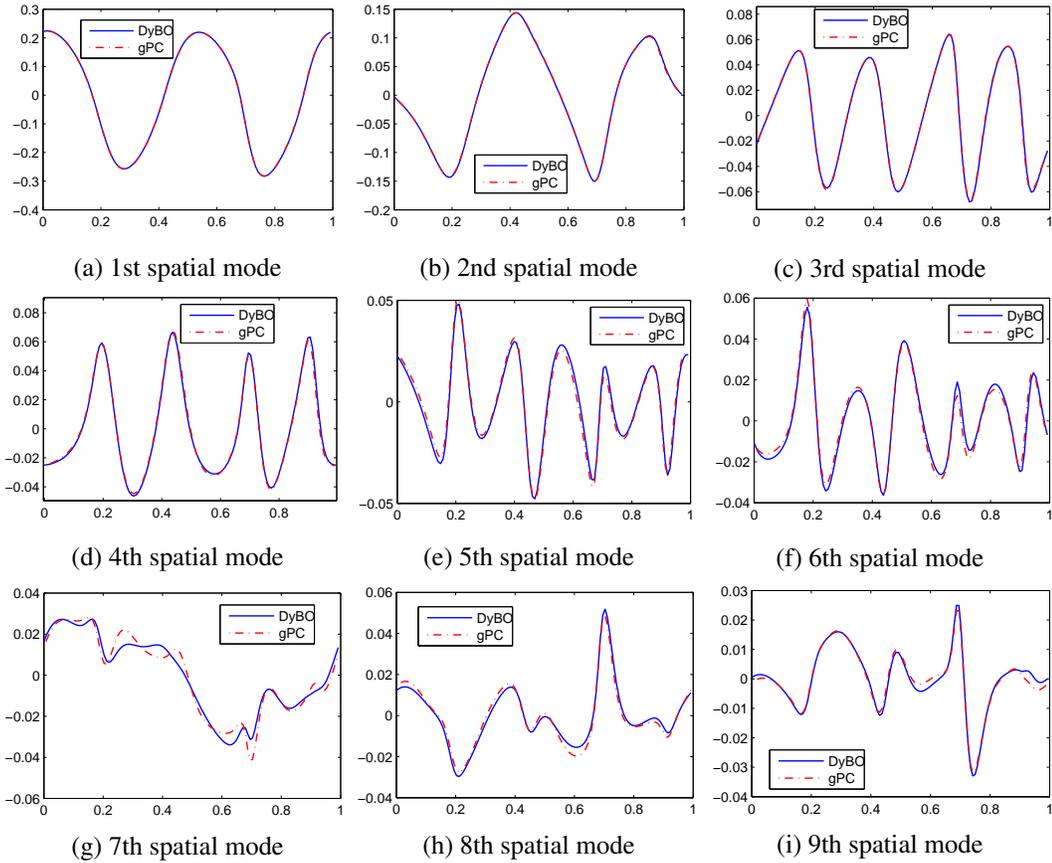


Figure 3.11: \mathbb{L}^2 relative errors of mean and STD as functions of time

mean and STD computed by gPC method with respect to $u^{(\text{gPC}, \mathfrak{J}_{N_p}^\infty)}$ are also given in the first row. Clearly, when $m = 10$ or $m = 12$, the errors of DyBO are comparable to those of gPC listed in the first row. That's to say, with $m = 10$ or $m = 12$, $\mathcal{E}_M \ll \mathcal{E}_{\mathfrak{J}}$ and further increasing m will not decrease the overall error \mathcal{E} , i.e., $m^* = 10$ or $m^* = 12$ in this numerical example. In this case, we achieve $9.6X \sim 12.7X$ speedup.

Figure 3.12: The first nine spatial modes computed by DyBO at time $t = 1.0$

Methods	Compared to $u^{(gPC, \mathfrak{J}_{N_p}^P)}$		Compared to $u^{(gPC, \mathfrak{J}_{N_p}^\infty)}$		Time (min)
	Mean	STD	Mean	STD	
gPC	NA	NA	1.1619%	2.1398%	42.1
DyBO m=4	3.219%	5.144%	3.096%	5.556%	1.32
DyBO m=6	2.855%	4.522%	2.935%	4.512%	2.01
DyBO m=8	0.930%	2.112%	1.333%	2.890%	2.64
DyBO m=10	0.444%	0.983%	1.222%	2.375%	3.32
DyBO m=12	0.171%	0.259%	1.109%	2.122%	4.37
Adaptive-DyBO	0.432%	0.462%	1.116%	2.157%	3.13

Table 3.1: Relative errors of statistical quantities computed by DyBO, Adaptive-DyBO, and gPC at time $t = 1.0$

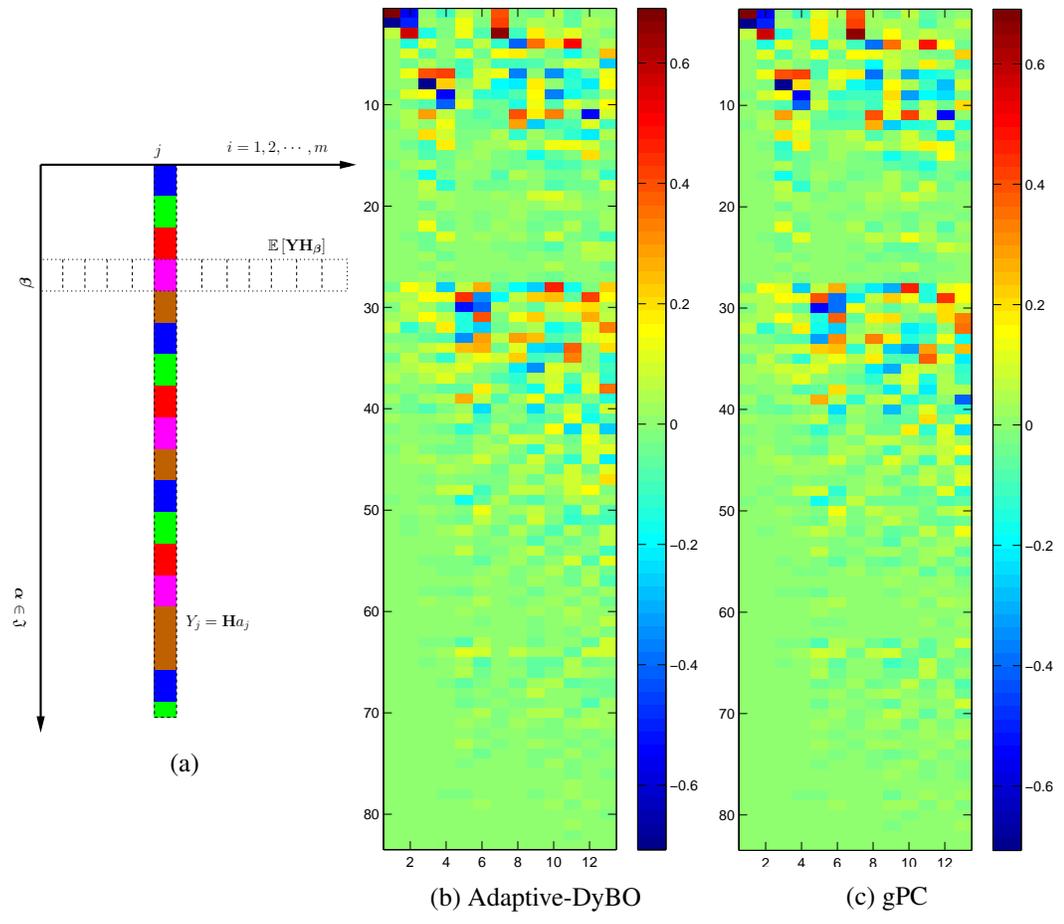


Figure 3.13: Stochastic modes $\mathbf{H}\mathbf{A}$ computed by DyBO at time $t = 1.0$

Chapter 4

Applications to Spatially Two-Dimensional SPDE

In this chapter, we continue to demonstrate the success of our DyBO-gPC method by applications to spatially two-dimensional Navier-Stokes (NS) equations driven by stochastic forces. In addition, we provide a detailed computational complexity analysis for linear PDE driven by stochastic forces and second-order nonlinear PDE driven by stochastic forces in Sec. 4.1. We also verify numerically the analysis in Sec. 4.2.

4.1 Computational Complexity Analysis

As we discussed in the previous chapters, the DyBO methods explore the low-dimensional structure of the stochastic solutions of time-dependent SPDE and represent the solution in the most compact form in the \mathbb{L}^2 sense. These changes in formulations not only bring in savings in memory consumption, i.e., storage of solutions in computer memory, but also reduce the total computational time since we have much fewer entries to update in each time iteration compared to gPC methods when both systems are numerically integrated. Storage complexity, i.e., memory consumption, is analyzed and compared with gPC method in Sec. 4.1.1. The saving in computational time depends on the specific form of the stochastic operator \mathcal{L} . Here, we provide analysis for two cases, linear

PDE driven by stochastic forces in Sec. 4.1.2.1 and second-order nonlinear PDE driven by stochastic forces in Sec. 4.1.2.2, respectively. As we will see later, these two cases already encompass a large class of problems.

To make the discussion concrete, we assume throughout this section that the randomness is given in terms of N_p independent random variables of the same distribution $\rho(\cdot)$ and a multi-index set \mathfrak{J} has been chosen of finite size $N_P = |\mathfrak{J}|$, i.e., the polynomial chaos basis $\mathbf{H}_{\mathfrak{J}}$ corresponding to the joint distribution $\prod_{l=1}^{N_p} \rho(z_l)$ has N_P elements. Furthermore, N_h grid nodes are used along each direction of the hyper-cube $\mathcal{D} \in \mathbb{R}^d$, which results in a spatial grid of total N_h^d nodes. Such discretizations generally lead to large systems for both gSC and DyBO-methods. In the numerical results presented in this section, we have used explicit types of numerical integrators, such as the forward Euler and the fourth-order RK methods. As a reminder, we have assumed throughout the thesis that the solutions of SPDEs under consideration enjoy low-dimensional structures, i.e., $m \ll N_P$.

4.1.1 Storage Complexity

From the gPC formulation (2.36), it is clear that $\bar{v}(x, t)$ and $\mathbf{V}(x, t) = (v_{\alpha}(x, t))_{\alpha \in \mathfrak{J}}$ have to be updated in each time iteration. Thus, the storage cost of the gPC solution is proportional to $O(N_h^d) + O(N_P N_h^d) = O(N_P N_h^d)$. On the other hand, the mean $\bar{u}(x, t)$, the spatial modes $\mathbf{U}(x, t) = (u_1(x, t), u_2(x, t), \dots, u_m(x, t))$ and the stochastic modes $\mathbf{A}(t) \in \mathbb{R}^{N_P \times m}$ from the DyBO-gPC formulation (2.39) are updated every time iteration, which implies the memory consumption is proportional to $O(N_h^d) + O(m N_h^d) + O(m N_P) = O(m N_h^d + m N_P)$. Here we have ignored the storage cost of auxiliary matrices in the DyBO formulation, i.e., matrices \mathbf{C} , \mathbf{D} and $G_* \in \mathbb{R}^{m \times m}$, which are $O(m^2)$. For the convenience of readers, the above discussion regarding the storage complexity is summarized in Table 4.1. Typically, the number of spatial grid nodes

is far more than the number of polynomial coefficients, i.e., $N_h^d \gg N_P$. Thus, the reduction of DyBO-gPC compared to gPC in terms of memory consumption is $O\left(\frac{m}{N_P}\right)$.

Method	Variables to update	Storage Complexity	Reduction
gPC (2.36)	\bar{v}, \mathbf{V}	$O(N_P N_h^d)$	NA
DyBO-gPC (2.39)	$\bar{u}, \mathbf{U}, \mathbf{A}$	$O(mN_h^d + mN_P)$	$O\left(\frac{m}{N_P}\right) + O\left(\frac{m}{N_h^d}\right) \approx O\left(\frac{m}{N_P}\right)$

Table 4.1: Storage complexity comparison between gPC and DyBO-gPC methods

4.1.2 Computational Cost

We continue to discuss the computational complexity in terms of computational time. Unlike the analysis of the storage complexity in the previous section, the analysis of computational time requires knowing the specific form of the stochastic differential operator \mathcal{L} . Here, we consider two important cases, linear or second-order nonlinear PDEs driven by stochastic forces, respectively, where second-order nonlinear PDEs are defined to be second-order polynomials of the solution u and its partial derivatives of any orders.

As analyzed in Sec. 2.5.2 and demonstrated numerically in Sec. 3.3, the DyBO-gPC formulation is a good approximation to the gPC formulation if m is taken properly, which in turn implies the computational time step sizes used to numerically integrate both systems should be comparable. Thus, to compare the total computational time, we only need to compare the time required to compute the right-hand sides if the same numerical integrating scheme is adopted for both gPC and DyBO-gPC.

Before we consider each case, we make the following two assumptions regarding the computational complexity of the spatial derivatives and the stochastic forces.

Assumption 4.1. *The spatial derivative can be computed in linear time. Let $\mathring{\mathcal{L}}$ be a deterministic linear differential operator. This assumption implies that the computational time of $\mathring{\mathcal{L}}u(x)$ on a*

spatial grid of N_h^d is $O(N_h^d)$. This assumption is certainly true for typical finite difference and finite volume schemes, since the evaluation of derivatives of any order generally involves only several adjacent layers of grid nodes. For pseudo-spectral methods, i.e., FFT is involved, this assumption is almost true since the computational time of FFT is $O(n \log n)$, where n is the number of grid nodes, and $\log n$ grows very slow as n increases.

Assumption 4.2. *The gPC expansion of stochastic force, \mathbf{F} , can be evaluated in linear time.*

We assume the stochastic force f is sparse on the selected polynomial basis \mathbf{H} . More precisely, the stochastic force has a finite gPC expansion, i.e., $f = \sum_{\alpha \in \mathfrak{J}} F_\alpha \mathbf{H}_\alpha = \mathbf{F}\mathbf{H}^T$. At most m polynomial coefficients F_α are non-zeros and the rest are identically zeros. This assumption implies that the computational cost of \mathbf{F} is bounded by mN_h^d .

4.1.2.1 Linear PDE Driven by Stochastic Forces

First, we consider the following SPDE

$$\frac{\partial u}{\partial t} = \mathcal{L}u = \mathring{\mathcal{L}}u + f = \mathring{\mathcal{L}}u + \mathbf{F}\mathbf{H}^T, \quad (4.1)$$

where the deterministic linear differential operator $\mathring{\mathcal{L}}$ is driven by a zero-mean stochastic force f .

Similar to Sec. 3.2, we have the gPC formulation of SPDE (4.1),

$$\frac{\partial \bar{v}}{\partial t} = \mathring{\mathcal{L}}\bar{v}, \quad (4.2a)$$

$$\frac{\partial \mathbf{V}}{\partial t} = \mathring{\mathcal{L}}\mathbf{V} + \mathbf{F}. \quad (4.2b)$$

Clearly, the evaluation of $\mathring{\mathcal{L}}v$ on the right-hand side costs $O(N_h^d)$, that of $\mathring{\mathcal{L}}\mathbf{V} = \left(\mathring{\mathcal{L}}v_\alpha\right)_{\alpha \in \mathfrak{J}}$ costs $O(N_P N_h^d)$ and that of \mathbf{F} costs $O(mN_h^d)$. Thus, the total computational cost of a single evaluation of the right hand sides in the gPC formulation (4.2) adds up to $O(N_P N_h^d)$.

From Sec. 3.2, it is easy to derive the DyBO-gPC formulation for SPDE (4.1),

$$\frac{\partial \bar{u}}{\partial t} = \mathring{\mathcal{L}}\bar{u}, \quad (4.3a)$$

$$\frac{\partial \mathbf{U}}{\partial t} = -\mathbf{U}\mathbf{D}^T + \mathring{\mathcal{L}}\mathbf{U} + \mathbf{F}\mathbf{A}, \quad (4.3b)$$

$$\frac{d\mathbf{A}}{dt} = -\mathbf{A}\mathbf{C}^T + \mathbf{A} \left\langle \mathring{\mathcal{L}}\mathbf{U}^T, \mathbf{U} \right\rangle \Lambda_{\mathbf{U}}^{-1} + \langle \mathbf{F}^T, \mathbf{U} \rangle \Lambda_{\mathbf{U}}^{-1}, \quad (4.3c)$$

where \mathbf{C} and \mathbf{D} can be computed via eqn. (2.22) from

$$G_* = \Lambda_{\mathbf{U}}^{-1} \left\langle \mathbf{U}^T, \mathring{\mathcal{L}}\mathbf{U} \right\rangle + \Lambda_{\mathbf{U}}^{-1} \left\langle \mathbf{U}^T, \mathbf{F}\mathbf{A} \right\rangle.$$

The computational costs of some typical terms on the right-hand side are given in Table 4.2. The second column is due to the fact that the computation of $-\mathbf{U}\mathbf{D}^T$ is essentially the multiplication of two matrices of size $N_n^d \times m$ and $m \times m$. Assumption 4.1 is used for the third column and Assumption 4.2 is used for the fourth column. The remaining columns can be deduced similarly.

Term	$-\mathbf{U}\mathbf{D}^T$	$\mathring{\mathcal{L}}\mathbf{U}$	$\mathbf{F}\mathbf{A}$	$-\mathbf{A}\mathbf{C}^T$	$\left\langle \mathbf{U}^T, \mathring{\mathcal{L}}\mathbf{U} \right\rangle$	Total
Time	$O(m^2 N_h^d)$	$O(m N_h^d)$	$O(m^2 N_h^d)$	$O(m^2 N_P)$	$O(m^2 N_h^d)$	$O(m^2 N_P + m^2 N_h^d)$

Table 4.2: The computational time of some typical terms on the right-hand side of the DyBO-gPC formulation

Therefore, the ratio between the computational cost of DyBO-gPC and that of gPC is

$$O\left(\frac{m^2}{N_P}\right) + O\left(\frac{m^2}{N_h^d}\right) \approx O\left(\frac{m^2}{N_P}\right). \quad (4.4)$$

4.1.2.2 Second-Order Nonlinear PDE Driven by Stochastic Forces

Under Assumptions 4.1 and 4.2, any second-order nonlinear PDE driven by stochastic forces is equivalent to the following SPDE in terms of computational cost

$$\frac{\partial u}{\partial t} = \mathcal{L}u = \left(\dot{\mathcal{L}}u\right)^2 + f = \left(\dot{\mathcal{L}}u\right)^2 + \mathbf{F}\mathbf{H}^T, \quad (4.5)$$

where $\dot{\mathcal{L}}$ is a deterministic linear differential operator.

Like what we have done in the linear case, we first consider the computational complexity of the gPC method. With the gPC expansion of the solution $v = \bar{v} + \mathbf{V}\mathbf{H}^T$, simple calculations give

$$\begin{aligned} \mathcal{L}v &= \left(\dot{\mathcal{L}}\bar{v}\right)^2 + 2\dot{\mathcal{L}}\bar{v}\dot{\mathcal{L}}\mathbf{V}\mathbf{H}^T + \dot{\mathcal{L}}\mathbf{V}\mathbf{H}^T\mathbf{H}\dot{\mathcal{L}}\mathbf{V}^T + \mathbf{F}\mathbf{H}^T, \\ \mathbb{E}[\mathcal{L}v] &= \left(\dot{\mathcal{L}}\bar{v}\right)^2 + \dot{\mathcal{L}}\mathbf{V}\dot{\mathcal{L}}\mathbf{V}^T, \\ \tilde{\mathcal{L}}v &= 2\dot{\mathcal{L}}\bar{v}\dot{\mathcal{L}}\mathbf{V}\mathbf{H}^T - \dot{\mathcal{L}}\mathbf{V}\dot{\mathcal{L}}\mathbf{V}^T + \dot{\mathcal{L}}\mathbf{V}\mathbf{H}^T\mathbf{H}\dot{\mathcal{L}}\mathbf{V}^T + \mathbf{F}\mathbf{H}^T, \end{aligned}$$

and

$$\begin{aligned} \mathbb{E}[\tilde{\mathcal{L}}v\mathbf{H}] &= 2\dot{\mathcal{L}}\bar{v}\dot{\mathcal{L}}\mathbf{V} + \mathbb{E}\left[\dot{\mathcal{L}}\mathbf{V}\mathbf{H}^T\mathbf{H}\dot{\mathcal{L}}\mathbf{V}^T\mathbf{H}\right] + \mathbf{F} \\ &= 2\dot{\mathcal{L}}\bar{v}\dot{\mathcal{L}}\mathbf{V} + \left(\dot{\mathcal{L}}v_\alpha\dot{\mathcal{L}}v_\beta\mathfrak{T}_{\alpha\beta\gamma}^{(\mathbf{H})}\right)_{1\times\gamma} + \mathbf{F}, \end{aligned}$$

where we have used the tensor $\mathfrak{T}^{\mathbf{H}}$ introduced in Sec. 3.3. Therefore, from eqn. (2.36), the gPC formulation for SPDE (4.5) is

$$\frac{\partial \bar{v}}{\partial t} = \left(\dot{\mathcal{L}}\bar{v}\right)^2 + \dot{\mathcal{L}}\mathbf{V}\dot{\mathcal{L}}\mathbf{V}^T, \quad (4.6a)$$

$$\frac{\partial \mathbf{V}}{\partial t} = 2\dot{\mathcal{L}}\bar{v}\dot{\mathcal{L}}\mathbf{V} + \left(\dot{\mathcal{L}}v_\alpha\dot{\mathcal{L}}v_\beta\mathfrak{T}_{\alpha\beta\gamma}^{(\mathbf{H})}\right)_{1\times\gamma} + \mathbf{F}. \quad (4.6b)$$

The computational cost of some typical terms on the right-hand sides is listed in Table 4.3.

Term	$(\mathring{\mathcal{L}}\bar{v})^2$	$\mathring{\mathcal{L}}\mathbf{V}\mathring{\mathcal{L}}\mathbf{V}^T$	$(\mathring{\mathcal{L}}v_\alpha\mathring{\mathcal{L}}v_\beta\mathfrak{T}_{\alpha\beta\gamma}^{(\mathbf{H})})_{1\times\gamma}$	\mathbf{F}	Total
Time	$O(N_h^d)$	$O(N_P N_h^d)$	$O(N_P^3 N_h^d)$	$O(m N_h^d)$	$O(N_P^3 N_h^d)$

Table 4.3: The computational time of terms on the right-hand sides in the gPC formulation for second-order nonlinear PDE driven by stochastic forces

Note that the third-order tensor $\mathfrak{T}^{(\mathbf{H})}$ only depends on the polynomial basis \mathbf{H} and can be pre-computed, so its computational cost is ignored in this analysis. To evaluate a single entry of row vector $(\mathring{\mathcal{L}}v_\alpha\mathring{\mathcal{L}}v_\beta\mathfrak{T}_{\alpha\beta\gamma}^{(\mathbf{H})})_{1\times\gamma}$, we have to compute the summation $\sum_{\alpha,\beta\in\mathfrak{J}}\mathring{\mathcal{L}}v_\alpha\mathring{\mathcal{L}}v_\beta\mathfrak{T}_{\alpha\beta\gamma}^{(\mathbf{H})}$, which costs $O(N_P^2 N_h^d)$, because a single evaluation of $\mathring{\mathcal{L}}v_\alpha$ costs $O(N_h^d)$ and a total of N_P^2 terms is summed up. Therefore, the total cost of the whole row vector is $O(N_P^3 N_h^d)$.

Next we consider the computational cost of DyBO-gPC. With the truncated KL expansion of the solution $u = \bar{u} + \mathbf{U}\mathbf{A}^T\mathbf{H}^T$, simple calculations give

$$\begin{aligned}\mathcal{L}u &= (\mathring{\mathcal{L}}\bar{u})^2 + 2\mathring{\mathcal{L}}\bar{u}\mathring{\mathcal{L}}\mathbf{U}\mathbf{A}^T\mathbf{H}^T + \mathring{\mathcal{L}}\mathbf{U}\mathbf{A}^T\mathbf{H}^T\mathbf{H}\mathbf{A}\mathring{\mathcal{L}}\mathbf{U}^T + \mathbf{F}\mathbf{H}^T, \\ \mathbb{E}[\mathcal{L}u] &= (\mathring{\mathcal{L}}\bar{u})^2 + \mathring{\mathcal{L}}\mathbf{U}\mathring{\mathcal{L}}\mathbf{U}^T, \\ \tilde{\mathcal{L}}u &= 2\mathring{\mathcal{L}}\bar{u}\mathring{\mathcal{L}}\mathbf{U}\mathbf{A}^T\mathbf{H}^T - \mathring{\mathcal{L}}\mathbf{U}\mathring{\mathcal{L}}\mathbf{U}^T + \mathring{\mathcal{L}}\mathbf{U}\mathbf{A}^T\mathbf{H}^T\mathbf{H}\mathbf{A}\mathring{\mathcal{L}}\mathbf{U}^T + \mathbf{F}\mathbf{H}^T,\end{aligned}$$

and

$$\begin{aligned}\mathbb{E}[\tilde{\mathcal{L}}u\mathbf{H}] &= 2\mathring{\mathcal{L}}\bar{u}\mathring{\mathcal{L}}\mathbf{U}\mathbf{A}^T + \mathbb{E}[\mathring{\mathcal{L}}\mathbf{U}\mathbf{A}^T\mathbf{H}^T\mathbf{H}\mathbf{A}\mathring{\mathcal{L}}\mathbf{U}^T\mathbf{H}] + \mathbf{F} \\ &= 2\mathring{\mathcal{L}}\bar{u}\mathring{\mathcal{L}}\mathbf{U}\mathbf{A}^T + (\mathring{\mathcal{L}}u_i\mathring{\mathcal{L}}u_j A_{\alpha i} A_{\beta j} \mathfrak{T}_{\alpha\beta\gamma}^{(\mathbf{H})})_{1\times\gamma} + \mathbf{F}.\end{aligned}$$

Therefore, from eqn. (2.39), we have the DyBO-gPC formulation for SPDE (4.5),

$$\frac{\partial \bar{u}}{\partial t} = \left(\dot{\mathcal{L}}\bar{u} \right)^2 + \dot{\mathcal{L}}\mathbf{U}\dot{\mathcal{L}}\mathbf{U}^T, \quad (4.7a)$$

$$\frac{\partial \mathbf{U}}{\partial t} = -\mathbf{U}\mathbf{D}^T + 2\dot{\mathcal{L}}\bar{u}\dot{\mathcal{L}}\mathbf{U} + \underbrace{\left(\dot{\mathcal{L}}u_i\dot{\mathcal{L}}u_j A_{\alpha i} A_{\beta j} A_{\gamma k} \mathfrak{T}_{\alpha\beta\gamma}^{(\mathbf{H})} \right)_{1 \times k}}_{\text{Term 1}} + \mathbf{F}\mathbf{A}, \quad (4.7b)$$

$$\frac{\partial \mathbf{A}}{\partial t} = -\mathbf{A}\mathbf{C}^T + 2\mathbf{A} \left\langle \dot{\mathcal{L}}\bar{u}\dot{\mathcal{L}}\mathbf{U}^T, \mathbf{U} \right\rangle \Lambda_{\mathbf{U}}^{-1} + \underbrace{\left(\mathfrak{T}_{ijk}^{(\mathbf{U})} A_{\alpha i} A_{\beta j} \mathfrak{T}_{\alpha\beta\gamma}^{(\mathbf{H})} \right)_{\gamma \times k}}_{\text{Term 2}} + \langle \mathbf{F}^T, \mathbf{U} \rangle \Lambda_{\mathbf{U}}^{-1}, \quad (4.7c)$$

where the third-order m -by- m -by- m tensor $\mathfrak{T}^{(\mathbf{U})} = \left(\left\langle \dot{\mathcal{L}}u_i\dot{\mathcal{L}}u_j, u_k \right\rangle \right)_{ijk}$, and matrices \mathbf{C} and \mathbf{D} can be solved via eqn. (2.22) from G_* ,

$$\Lambda_{\mathbf{U}} G_* = 2 \left\langle \mathbf{U}^T, \dot{\mathcal{L}}\bar{u}\dot{\mathcal{L}}\mathbf{U} \right\rangle + \underbrace{\left(\mathfrak{T}_{ijk}^{(\mathbf{U})} A_{\alpha i} A_{\beta j} A_{\gamma l} \mathfrak{T}_{\alpha\beta\gamma}^{(\mathbf{H})} \right)_{k \times l}}_{\text{Term 3}} + \langle \mathbf{U}^T, \mathbf{F}\mathbf{A} \rangle. \quad (4.8)$$

It is not difficult to derive the computational costs listed in Table 4.4 regarding matrix-tensor products in the above formulation. Please note that the Einstein summation convention is implicitly assumed and the product should be computed in a recursive way, i.e., $A_{\alpha i} A_{\beta j} A_{\gamma k} \mathfrak{T}_{\alpha\beta\gamma}^{(\mathbf{H})} = A_{\alpha i} \left(A_{\beta j} \left(A_{\gamma k} \mathfrak{T}_{\alpha\beta\gamma}^{(\mathbf{H})} \right) \right)$. For simplicity of notation, we also drop $\mathcal{O}(\cdot)$ from the second row.

Matrix-Tensor Product	$A_{\alpha i} \mathfrak{T}_{\alpha\beta\gamma}^{(\mathbf{H})}$	$A_{\alpha i} A_{\beta j} \mathfrak{T}_{\alpha\beta\gamma}^{(\mathbf{H})}$	$A_{\alpha i} A_{\beta j} A_{\gamma k} \mathfrak{T}_{\alpha\beta\gamma}^{(\mathbf{H})}$
Time	mN_P^3	$mN_P^3 + m^2N_P^2 \lesssim mN_P^3$	$mN_P^3 + m^2N_P^2 + m^3N_P \lesssim mN_P^3$

Table 4.4: The computational costs of matrix-tensor \mathbf{A} and $\mathfrak{T}^{(\mathbf{H})}$ products

Similar to the gPC, the computational costs of some typical terms on the right-hand side of the DyBO-gPC formulation (4.7) are given in Table 4.5, where the estimate of term 2 goes as follows. The computation of $\mathfrak{T}^{(\mathbf{U})}$ in term 2 costs $\mathcal{O}(m^3N_h^d)$, while the computation of matrix-tensor product $A_{\alpha i} A_{\beta j} \mathfrak{T}_{\alpha\beta\gamma}^{(\mathbf{H})}$ costs $\mathcal{O}(mN_P^3)$ as shown in Table 4.4. The last step of computing tensor-tensor product costs $\mathcal{O}(N_P m^3)$. Thus, the total computational cost of term 2 in eqn. (4.7c)

is $O(m^3 N_h^d) + O(m N_P^3) + O(N_P m^3) \leq O(m^3 N_h^d) + O(m N_P^3)$ since $m \ll N_P$.

Term	$\mathring{\mathcal{L}}\mathbf{U}\mathring{\mathcal{L}}\mathbf{U}^T$	$\langle \mathbf{U}^T, \mathring{\mathcal{L}}\bar{u}\mathring{\mathcal{L}}\mathbf{U} \rangle$	$\mathfrak{T}^{(\mathbf{U})}$	Term 1,2 or 3	Total
Time	$m N_h^d$	$m^2 N_h^d$	$m^3 N_h^d$	$m N_P^3 + m^3 N_h^d$	$m N_P^3 + m^3 N_h^d$

Table 4.5: The computational costs of terms on the right-hand sides in the DyBO-gPC formulation for second-order nonlinear PDE driven by stochastic forces

In the light of the above discussions, the ratio of the computational costs between DyBO-gPC and gPC for the second-order PDE driven by stochastic force is

$$O\left(\frac{m}{N_h^d}\right) + O\left(\left(\frac{m}{N_P}\right)^3\right) \approx O\left(\left(\frac{m}{N_P}\right)^3\right) = O\left(m^\alpha N_P^{-\beta}\right), \quad (4.9)$$

where the exponents $\alpha = 3$ and $\beta = 3$. In the next section, we will numerically verify these two exponents for Navier-Stokes equations driven by stochastic forces.

Remark 4.1. If the distribution of ξ_i 's is Gaussian, the tensor $\mathfrak{T}^{(\mathbf{H})}$ can be quite sparse, i.e., a few non-zero entries out of total N_P^3 entries. However, this may not be the case for general distributions, so we do not explore this sparsity in the above analysis. Later in numerical results, we will show that even if such sparsity is explored in numerical implementations of the gPC method, our DyBO-gPC is still superior.

4.2 Stochastic Navier-Stokes Equations

As a model to test numerically the proposed DyBO formulation for a spatially two-dimensional non-linear SPDE, we consider the incompressible Navier-Stokes equations driven by stochastic forces. Specifically, we consider the stochastic flow in a unit square, i.e., $\mathcal{D} = [0, 1] \times [0, 1]$, with periodic boundary conditions on both spatial directions x and y (see Fig. 4.1a). The governing SPDE of this

stochastic flow is the Stochastic Navier-Stokes equations (SNS)

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0, \quad (4.10a)$$

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = -\frac{\partial p}{\partial x} + \nu \Delta u + f_1, \quad (4.10b)$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} = -\frac{\partial p}{\partial y} + \nu \Delta v + f_2, \quad (4.10c)$$

where the meanings of symbols are tabulated below

$u(x, t, \omega), v(x, t, \omega)$	-	x and y components of velocity
$p(x, t, \omega)$	-	Pressure
$f_1(x, t, \omega), f_2(x, t, \omega)$	-	x and y components of zero-mean stochastic force f
deterministic constant ν	-	Dynamic viscosity

For spatially two-dimensional incompressible flow problems, it is convenient to consider the vorticity-stream function formulation instead of the pressure-velocity formulation. The vorticity-stream function formulation gives $w = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}$ and ψ , with $u = \frac{\partial \psi}{\partial y}$ and $v = -\frac{\partial \psi}{\partial x}$. Thus, the above pressure-velocity formulation of SNS can be transformed to the vorticity-stream formulation,

$$\frac{\partial w}{\partial t} + \left(u \frac{\partial}{\partial x} + v \frac{\partial}{\partial y} \right) w = \nu \Delta w + \left(\frac{\partial f_2}{\partial x} - \frac{\partial f_1}{\partial y} \right),$$

$$-\Delta \psi = w,$$

$$u = \frac{\partial \psi}{\partial y}, \quad v = -\frac{\partial \psi}{\partial x},$$

or cast in the standard form (1.1a),

$$\frac{\partial w}{\partial t} = \mathcal{L}w = - \left(u \frac{\partial}{\partial x} + v \frac{\partial}{\partial y} \right) w + \nu \Delta w + \left(\frac{\partial f_2}{\partial x} - \frac{\partial f_1}{\partial y} \right), \quad (4.12)$$

where

$$-\Delta\psi = w, \quad (4.13)$$

$$u = \frac{\partial\psi}{\partial y} = -\frac{\partial}{\partial y}\Delta^{-1}w, \quad (4.14)$$

$$v = -\frac{\partial\psi}{\partial x} = \frac{\partial}{\partial x}\Delta^{-1}w. \quad (4.15)$$

We assume the randomness is given in terms of N_p independent standard Gaussian random variables, $\boldsymbol{\xi} = (\xi_1, \xi_2, \dots, \xi_{N_p})$, and the initial vorticity is deterministic, i.e., $w(x, y, 0, \boldsymbol{\xi}) = \dot{w}(x, y)$. That is to say, the randomness is injected into the system only through the stochastic force $f = (f_1, f_2)$. Let \mathfrak{J} is a multi-index set of finite size. We assume that the stochastic force in eqn. (4.12) has a finite-term gPC expansion, i.e.,

$$\frac{\partial f_2}{\partial x} - \frac{\partial f_1}{\partial y} = \sum_{\alpha \in \mathfrak{J}} F_\alpha \mathbf{H}_\alpha = \mathbf{F} \mathbf{H}^T.$$

In the following numerical example, we adopt the initial vorticity field used in Luo's thesis [77],

$$\dot{w}(x, y) = \text{const} - \frac{1}{2\delta_1} \exp\left(-\frac{I(x)(y-0.5)^2}{2\delta_1^2}\right),$$

where $I(x) = 1 + \delta_2 (\cos(4\pi x) - 1)$ and the constant is taken such that $\int_{\mathcal{D}} \dot{w} \, dx \, dy = 0$. $\delta_1 = 0.025$ and $\delta_2 = 0.3$, so the initial vorticity concentrates in a narrow band along $y = 0.5$ as shown in Fig. 4.1b, which models a perturbed flat vortex sheet in the limit that $\delta_1 \rightarrow 0$. Like in the numerical examples of stochastic Burgers equation in Sec. 3.3, we adopt an approximated version of Brownian force $(\sigma_1(x, y)B_1(t), \sigma_2(x, y)B_2(t))$

$$\left(\frac{\partial f_2}{\partial x} - \frac{\partial f_1}{\partial y}\right) = -\frac{\partial\sigma_1}{\partial y} \sum_{i=1}^4 \frac{1}{\sqrt{T}} M_i\left(\frac{t}{T}\right) \xi_i + \frac{\partial\sigma_2}{\partial x} \sum_{i=5}^8 \frac{1}{\sqrt{T}} M_{i-4}\left(\frac{t}{T}\right) \xi_i. \quad (4.16)$$

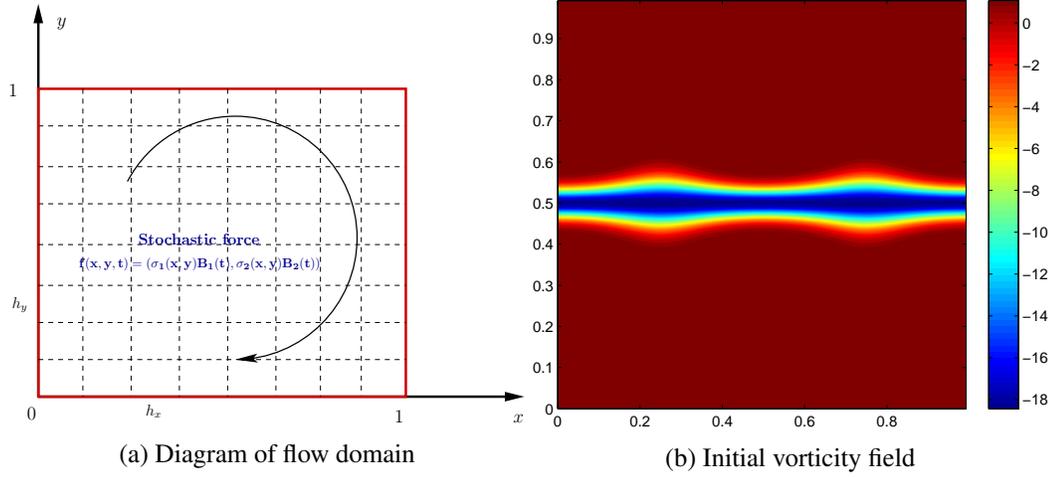


Figure 4.1: Stochastic flows driven by stochastic force f in 2D unit square

The functions σ_1 and σ_2 are chosen such that

$$\begin{aligned}\frac{\partial \sigma_1}{\partial y} &= \delta_3 \pi \cos(2\pi x) \cos(2\pi y), \\ \frac{\partial \sigma_2}{\partial x} &= \delta_3 \pi \sin(2\pi x) \sin(2\pi y),\end{aligned}$$

where $\delta_3 = 0.3$.

4.2.1 gPC and DyBO-gPC Formulations of SNSE

The derivations of gPC or DyBO-gPC formulations of SNSE, although quite technical, essentially follow the similar steps of stochastic Burgers equations in Sec. 3.3. Here, we just give the results.

For details, see Appendix D and E.

By considering the gPC expansion $w = \bar{w} + \mathbf{W}\mathbf{H}^T$, we can derive the gPC formulation of SNSE (4.12),

$$\frac{\partial \bar{w}}{\partial t} = \nu \Delta \bar{w} - \mathfrak{D}_{(\bar{u}, \bar{v})} \bar{w} - \mathfrak{D}_{(\mathbf{U}, \mathbf{V})} \mathbf{W}, \quad (4.17a)$$

$$\frac{\partial \mathbf{W}}{\partial t} = \nu \Delta \mathbf{W} - \mathfrak{D}_{(\bar{u}, \bar{v})} \mathbf{W}^T - \mathfrak{D}_{(\mathbf{U}, \mathbf{V})} \bar{w} - \left(\mathfrak{D}_{(u_\alpha, v_\alpha)} w_\beta \mathfrak{I}_{\alpha\beta\gamma}^{(\mathbf{H})} \right)_{1 \times \gamma} + \mathbf{F}, \quad (4.17b)$$

where $\mathfrak{D}_{(\cdot,\cdot)}(\cdot)$ is the generalized material derivative defined in eqn. (D.9). DyBO-gPC can be obtained by considering the m -term truncated KL expansion $w = \bar{w} + \mathbf{W}\mathbf{B}^T\mathbf{H}^T$,

$$\frac{\partial \bar{w}}{\partial t} = \nu \Delta \bar{w} - \mathfrak{D}_{(\bar{u}, \bar{v})} \bar{w} - \mathfrak{D}_{(\mathbf{U}, \mathbf{V})} \mathbf{W}, \quad (4.18a)$$

$$\begin{aligned} \frac{\partial \mathbf{W}}{\partial t} = & -\mathbf{W}\mathbf{D}_w^T + [\nu \Delta \mathbf{W} - \mathfrak{D}_{(\bar{u}, \bar{v})} \mathbf{W} - \mathfrak{D}_{(\mathbf{U}, \mathbf{V})} \bar{w}] \\ & - \left[\mathfrak{D}_{(u_i, v_i)} w_j B_{\alpha i} B_{\beta j} B_{\gamma k} \mathfrak{T}_{\alpha \beta \gamma}^{(\mathbf{H})} \right]_{1 \times k} + \mathbf{F}\mathbf{B}, \end{aligned} \quad (4.18b)$$

$$\begin{aligned} \frac{d\mathbf{B}}{dt} = & \mathbf{B} \left(-\mathbf{C}_w^T + \left\langle \nu \Delta \mathbf{W}^T - (\mathfrak{D}_{(\bar{u}, \bar{v})} \mathbf{W})^T - (\mathfrak{D}_{(\mathbf{U}, \mathbf{V})} \bar{w})^T, \mathbf{W} \right\rangle \Lambda_{\mathbf{W}}^{-1} \right) \\ & - \left[\mathfrak{T}_{ijk}^{(\mathbf{W})} B_{\alpha i} B_{\beta j} \mathfrak{T}_{\alpha \beta \gamma}^{(\mathbf{H})} \right]_{\gamma \times k} \Lambda_{\mathbf{W}}^{-1} + \langle \mathbf{F}^T, \mathbf{W} \rangle \Lambda_{\mathbf{W}}^{-1}, \end{aligned} \quad (4.18c)$$

where matrices \mathbf{C}_w and \mathbf{D}_w can be solved via eqn. (2.22) from G_{*w} ,

$$\Lambda_{\mathbf{W}} G_{*w} = \langle \mathbf{W}^T, \nu \Delta \mathbf{W} - \mathfrak{D}_{(\bar{u}, \bar{v})} \mathbf{W} - \mathfrak{D}_{(\mathbf{U}, \mathbf{V})} \bar{w} \rangle - \left[\mathfrak{T}_{ijk}^{(\mathbf{W})} B_{\alpha i} B_{\beta j} B_{\gamma l} \mathfrak{T}_{\alpha \beta \gamma}^{(\mathbf{H})} \right]_{k \times l} + \langle \mathbf{W}^T, \mathbf{F} \rangle \mathbf{B}.$$

Both the gPC system and DyBO-gPC systems are numerically integrated by fourth-order RK method with time step $\delta t = 10^{-3}$ on a 128×128 spatial grid. The pseudo-spectral method with the 36th-order Fourier smoothing [54] is used to compute spatial derivatives. Unless specified otherwise, all computations are performed on a desktop equipped with Intel Pentium D processor of 3.4 GHz and 4 GB memory. Various numerical results are presented in the next section.

4.2.2 Numerical Results

Verification of Complexity Analysis. Clearly, SNSE (4.12) is a second-order nonlinear PDE driven by stochastic forces, so the complexity analysis in Sec. 4.1.2.2 is applicable. Before presenting computational results, we first verify the complexity analysis, i.e., eqn. (4.9). To this end, we record the wall time of a single time iteration when the gPC system (4.17) or the DyBO-gPC system

(4.18) is numerically integrated by the fourth-order Runge-Kutta method. For $N_P = 80, 100, 120$ and $m = 4, 8, 12, 16$, the computational times are summarized into Table 4.6. To improve the accuracy of recorded wall times, we actually compute the average wall time of 10 time iterations.

In Table 4.6, the exponents α and β in eqn. (4.9) are estimated by linear regression. The last column uses wall times corresponding to $m = 8, 12, 16$, while the second to last column uses all four values of m . As we can see from the fourth column of the table, the computational time is relatively small when $m = 4$. In this case, the dominant terms in our previous analysis may not truly dominate other terms and some inevitable programming overheads, such as memory allocation and function calling overhead, may kick in. In Fig. 4.2 accompanying Table 4.6, the computational times corresponding to $m = 8, 12, 16$ align nicely into a straight line for each $N_P = 80, 100, 120$, respectively, but the computational times corresponding to $m = 4$ drift up. If we remove these points from our fitting, the linear regression estimate of the exponent α in eqn. (4.4) would be approximately equal to 2.73, close to the theoretically predicated value 3.

As we mentioned in Remark 4.1, $\mathfrak{T}^{(H)}$ is very sparse when Hermite polynomials are used for Gaussian random variables. In Table 4.6, we also report the wall times of gPC in the second and third columns, respectively, depending on whether such sparsity is explored or not in the numerical implementation of gPC methods. Clearly, the computational time is significantly smaller if such sparsity is considered. However, we may not have such luxury for non-Gaussian random variables, i.e., general distributions. In the last two row of Table 4.6, the exponent β is estimated by linear regression, respectively, when sparsity is explored or not. The last row gives ~ 2.9 for the exponent β confirming our analysis in eqn. (4.9).

Numerical Errors of DyBO-gPC. As mentioned in Chapter 1, the number of polynomial chaos basis functions \mathbf{H}_α grows exponentially fast as the number of random variables N_p and the total order P increase. The scheme of sparse truncation proposed in Luo's thesis [77] proves to be

N_P	gPC (<i>sec</i>)		DyBO-gPC (<i>sec</i>)				α	
	Sparse	Non-Sparse	$m = 4$	$m = 8$	$m = 12$	$m = 16$	α_1	α_2
80	17.242	772.10	0.3946	1.6238	4.8850	10.5483	2.3670	2.7006
100	26.302	1482.7	0.4221	1.5666	4.9577	10.7119	2.3334	2.7779
120	36.440	2558.3	0.4246	1.6567	5.0451	10.8200	2.3342	2.7099
β	β - Sparse		1.6621	1.8056	1.7683	1.7844		
	β - Non-Sparse		2.7683	2.9117	2.8744	2.8905		

Table 4.6: Comparison of wall times of a single RK step of gPC and DyBO-gPC systems. Depending on whether the sparsity of tensor $\mathfrak{T}^{(\mathbf{H})}$ is explored or not, the wall times are reported in the second and third columns for gPC method, respectively. The wall times of DyBO-gPC are reported in columns 4–7 for $m = 4, 8, 12, 16$. The exponents α and β in eqn. (4.9) are estimated by linear regression. The last column in red uses wall times corresponding to $m = 8, 12, 16$ to compute the exponent α , while the second to last column in gray uses all four values of m to estimate the exponent β . The values of β are reported in the last two rows.

a relatively effective method to alleviate the situation. In the following computation, we follow this scheme and choose a multi-index set \mathfrak{J} obtained from a sparse truncation of the multi-index set \mathfrak{J}_8^3 ,

$$\mathfrak{J} = \{ \boldsymbol{\alpha} \in \mathfrak{J}_8^3 \text{ and if } |\boldsymbol{\alpha}| = 3, \text{ then } \alpha_2 \leq 2, \alpha_3 \leq 1, \alpha_4 \leq 1, \alpha_6 \leq 2, \alpha_7 \leq 1, \alpha_8 \leq 1 \} \setminus \{ \mathbf{0} \},$$

which still results in total 130 multi-indices!

The mean and STD of the vorticity field and the first four spatial modes in the KL expansion of vorticity field at time $t = 1.0$ are given in Fig. 4.3 and Fig. 4.4, respectively. In both figures, the results by our DyBO-gPC method with $m = 8$ are given in the left column and compared to the results by the gPC method in the right column. The results are almost indistinguishable. We further confirm the convergence of DyBO-gPC to gPC by plotting the relative errors of mean and STD of vorticity field as functions of time in the top two subplots of Fig. 4.5.

In the same figure, we also report the relative errors as functions of time when the adaptive strategy of adding and/or removing mode pairs is enacted. Two numerical examples are provided: one starts with four mode pairs, i.e., $m_0 = 4$ and the other starts with six mode pairs, i.e., $m_0 = 6$. In Fig. 4.5c, the number of mode pairs in the DyBO-gPC method is plotted against time t .

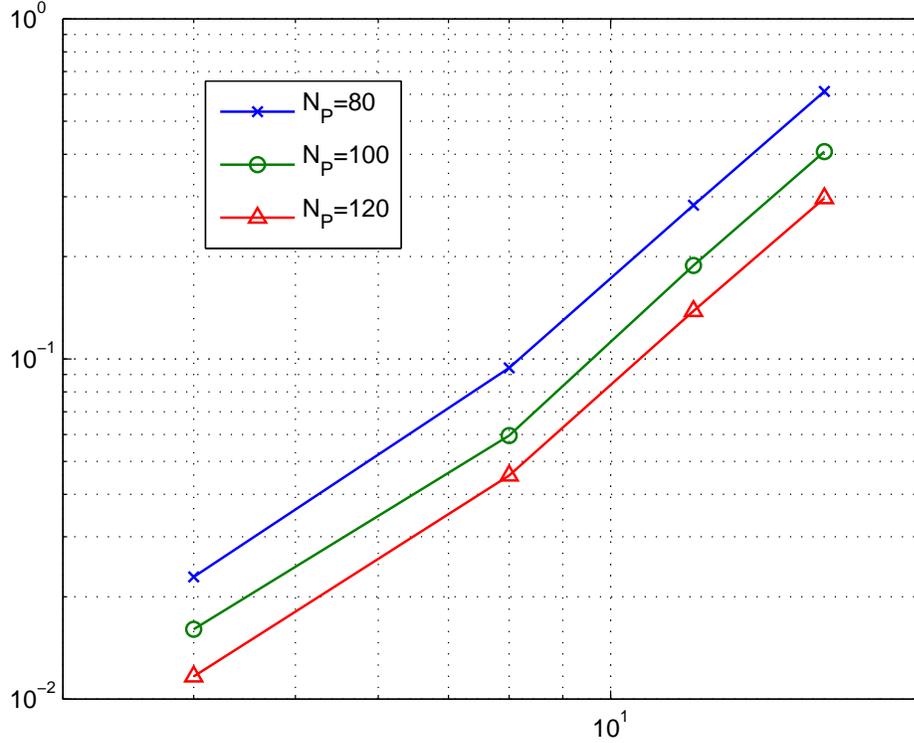


Figure 4.2: Wall time of a single RK step of the DyBO-gPC system as a function of m

Because of the special form of the stochastic force f considered in this numerical example, the randomness is only introduced through low-order gPC coefficients and then spread to the mean and other high-order gPC coefficients. See Remark 3.3 for details. At the beginning of the evolution of the stochastic flow, the randomness is not strong and the adaptive algorithm finds no need to add new mode pairs before time $t = 0.35$. As the system evolves, the randomnesses get strong through interactions among different mode pairs and the adaptive algorithm automatically adds more mode pairs when necessary.

Avoiding Selection of Multi-index Set \mathfrak{J} . As mentioned in Chapter 1, the gPC method suffers greatly from the curse of dimensionality. In the above numerical example, we use low-order (≤ 3) polynomials and also the sparse truncation technique to further reduce the size of multi-index set \mathfrak{J} , which still results in a set of 130 polynomials. It takes more than 8 hour of wall time to numerically integrate the gPC system from $t = 0.0$ to $t = 1.0$! Adaptive gPC methods try to include

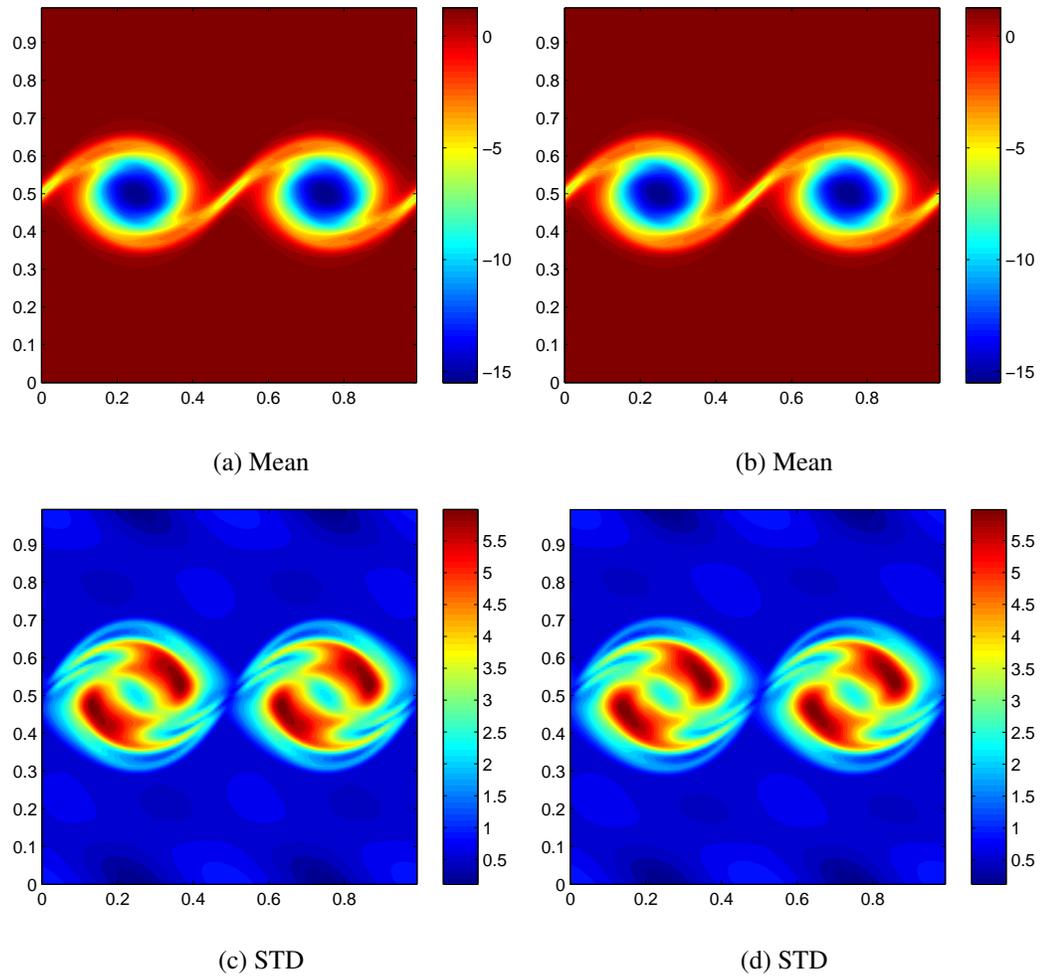


Figure 4.3: Mean and STD of vorticity field at time $t = 1.0$. The left column is computed by DyBO-gPC, while the right column is computed by gPC. They are essentially the same.

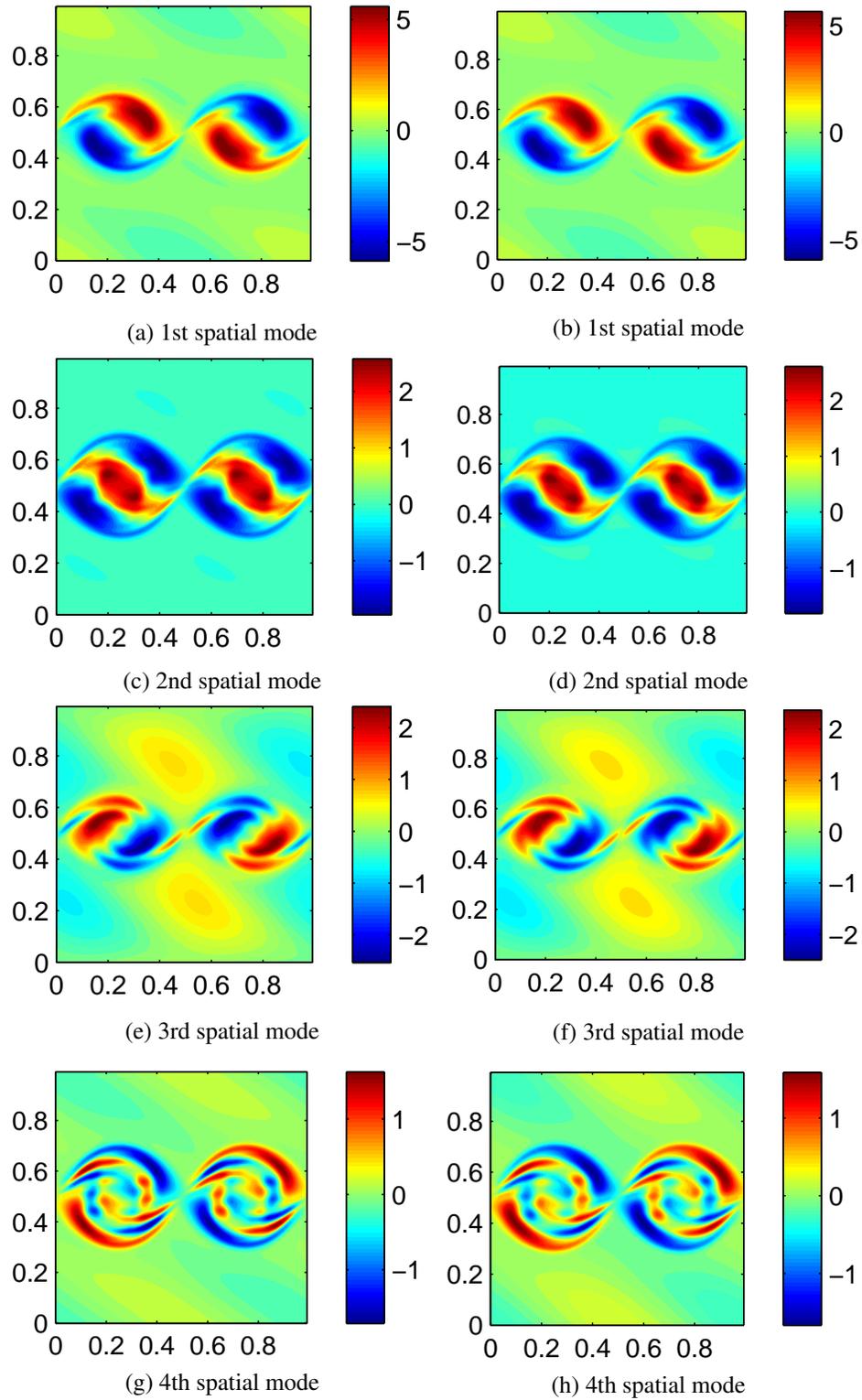


Figure 4.4: The first four spatial modes at time $t = 1.0$. The left column is computed by DyBO-gPC, while the right column is computed by gPC. They are essentially the same.

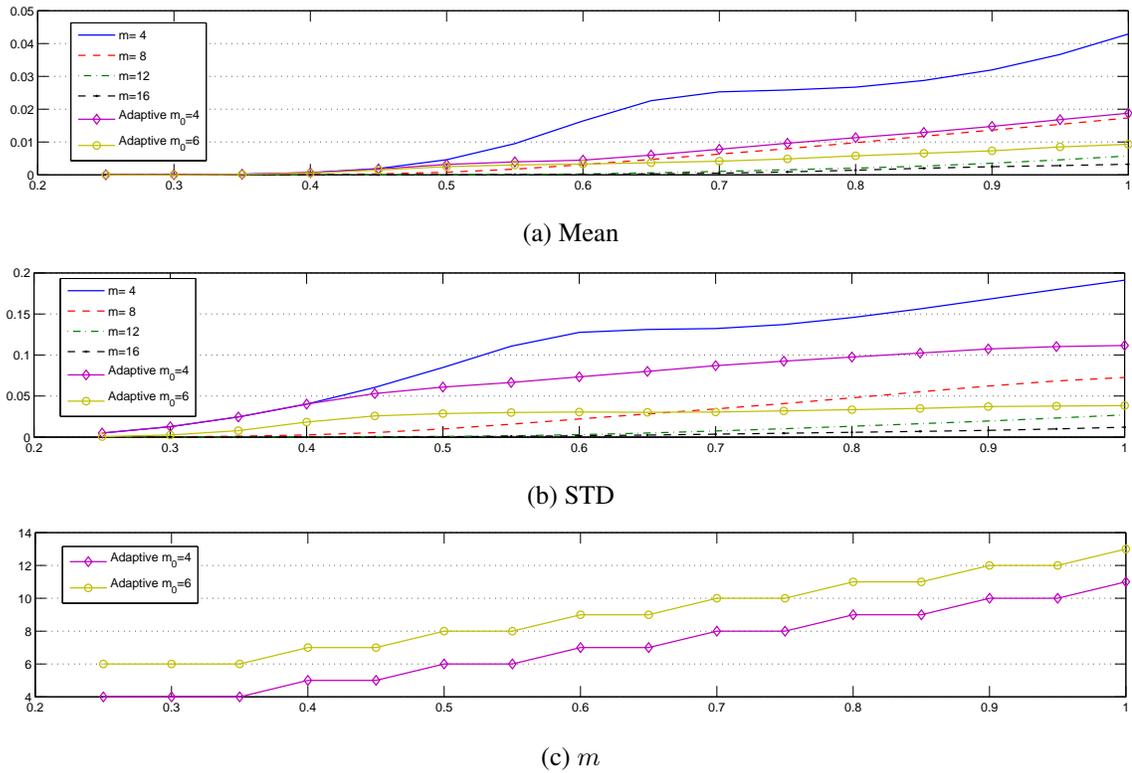


Figure 4.5: The \mathbb{L}^2 relative errors of mean and STD of vorticity field computed by DyBO. The errors are plotted as functions of time in the top two figures, while the numbers of mode pairs used in the adaptive strategy are given in the last figure.

only important gPC coefficients w_α in the computation, i.e., the selection of multi-index set \mathfrak{J} .

In Fig. 4.6a, we plot the energy spectrum of the gPC solution at $t = 1.0$, i.e., $\left\{ \|w_\alpha\|_{\mathbb{L}^2(\mathcal{D})} \right\}_{\alpha \in \mathfrak{J}}$, which clearly does not decay monotonically. Index \mathfrak{J} is a multi-index set, so we do not have sufficient information and a good strategy, prior to the computation, to sort \mathfrak{J} and select the most important ones. We can re-arrange the multi-indices α 's, such that $\|w_\alpha\|_{\mathbb{L}^2(\mathcal{D})}$ decays monotonically as is shown in Fig. 4.6b. We also call it the sorted energy spectrum of the gPC solution.

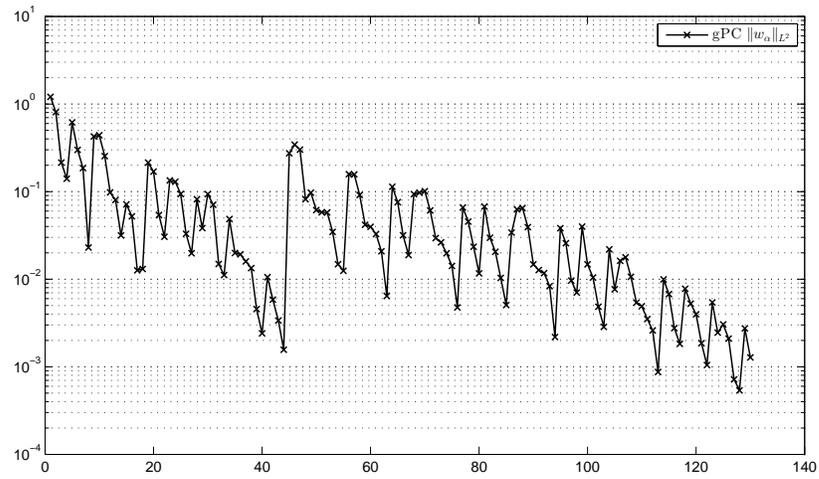
On the other hand, our DyBO method tracks the KL expansion of the true solution and *automatically* includes only the most important ones. Furthermore, the KL expansion is known to provide the most compact representation of second-order stochastic process, so the energy spectrum of the DyBO solution, i.e., $\left\{ \|w_i\|_{\mathbb{L}^2(\mathcal{D})} \right\}_{i=1,2,\dots,m}$, has a faster decay rate even compared to the sorted energy spectrum of the gPC solution (see Fig. 4.6b). This difference in decay rate implies that our method gives a smaller system to solve, leading to less computational cost.

To further illustrate and understand the benefits of the DyBO method, we consider a little “stronger” stochastic force

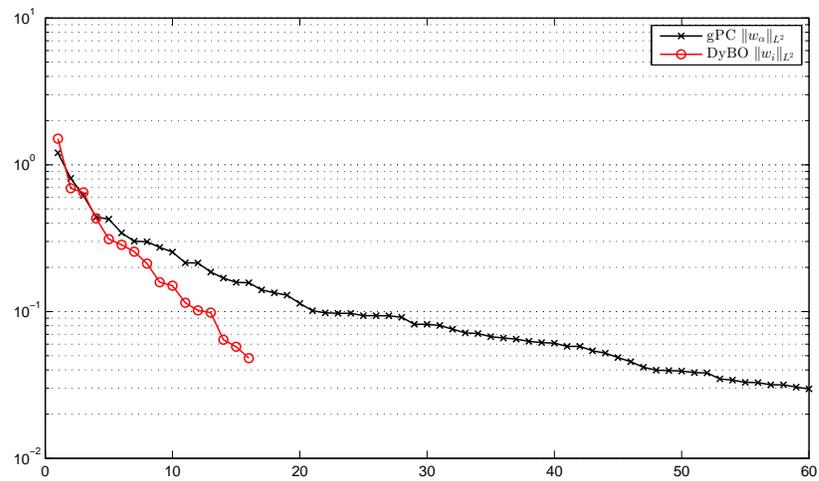
$$\left(\frac{\partial f_2}{\partial x} - \frac{\partial f_1}{\partial y} \right) = -\frac{\partial \sigma_1}{\partial y} \sum_{i=1}^4 \frac{it}{T} \xi_i + \frac{\partial \sigma_2}{\partial x} \sum_{i=5}^8 \frac{(i-4)t}{T} \left(\frac{t}{T} \right) \xi_i. \quad (4.19)$$

With this stochastic force, the (sorted) energy spectrum of both gPC and DyBO solutions along with the square roots of eigenvalues are plotted in Fig. 4.7a. Clearly, the energy spectrum of DyBO decays much faster than that of gPC.

Once the gPC coefficients w_α 's are sorted in the descending order of $\|w_\alpha\|_{\mathbb{L}^2(\mathcal{D})}$, we can use the first several gPC coefficients, i.e., the most important ones, to compute a solution and compare with the exact one. The relative errors of STD computed by this procedure are plotted in Fig. 4.7b against the number of gPC coefficients. However, from the perspective of practical numerical simulations,



(a) Unsorted energy spectrum of the gPC solution

 \Downarrow Sort

(b) (Sorted) energy spectrum of the gPC and DyBO solutions

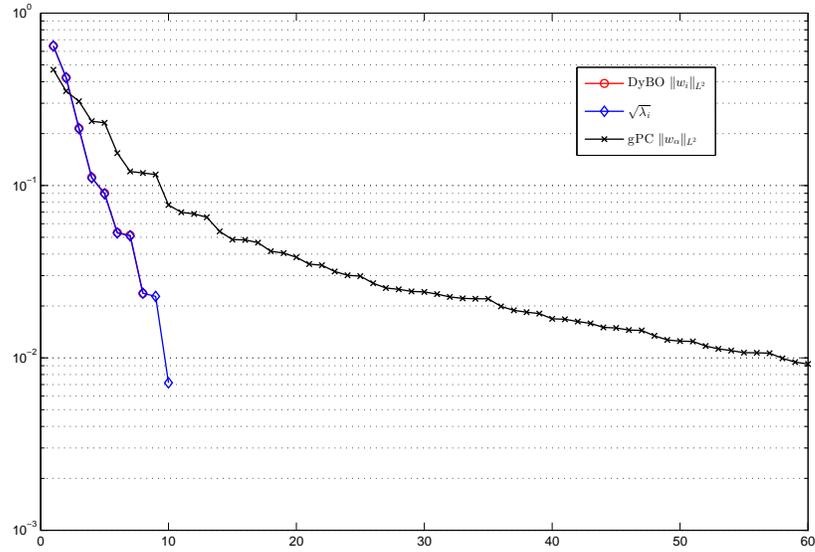
Figure 4.6: Comparison of energy spectrum of gPC and DyBO solutions at time $t = 1.0$

we are making two “crimes” here.

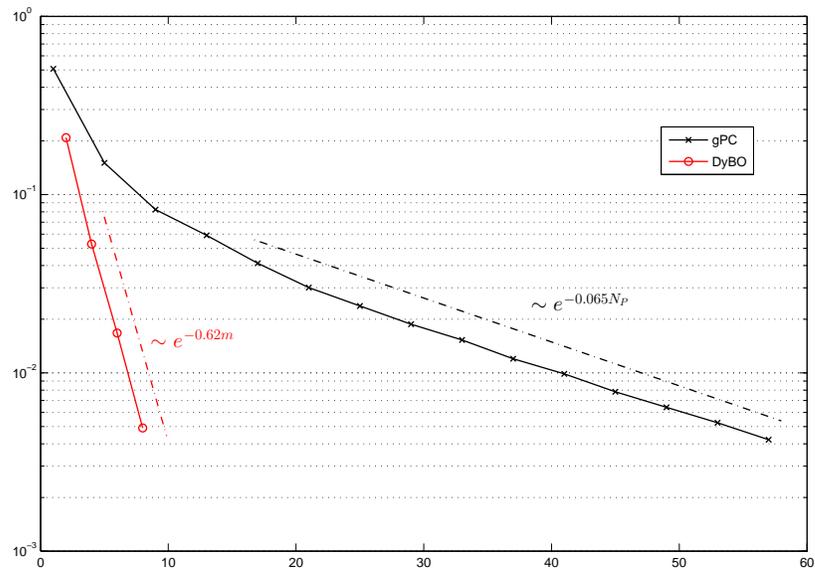
- In general, the multi-indices α 's corresponding to the most important gPC coefficients can not be known prior to the beginning of computations. What's more, such set of multi-indices may change with respect to time t , making the selection of an effective multi-index set \mathfrak{J} even harder.
- Less important gPC coefficients excluded from the gPC system may induce additional errors when we solve the system which only includes the most important ones.

Even if we are willing to make these two “crimes”, the solution obtained by this procedure is less accurate than that by our DyBO method with the same number of mode pairs, as shown in Fig. 4.7b. With only 8 mode pairs, our DyBO method achieves the same accuracy ($\sim 0.5\%$) as that by gPC method with 60 gPC coefficients. By using Table 4.6, we can estimate speedup in this case. When the sparsity of tensor $\mathfrak{T}^{\mathbf{H}}$ is not explored in the numerical implementation of gPC, the speedup is $\sim 200X$ (327.8 sec vs. 1.6567 sec per time iteration). When the sparsity is explored, the speedup is $\sim 6X$ (10.628 sec vs. 1.6567 sec per time iteration). Fig. 4.7a also confirms numerically that our DyBO method can accurately recover the eigenvalues in the KL expansion.

Looking into the stochastic modes $\mathbf{Y} = \mathbf{H}\mathbf{A}$ reveals the origin of fast error decay in our DyBO-gPC method. In Fig. 4.8, we plot the stochastic modes computed by DyBO in the second figure and ones recovered from the gPC solution in the third figure, respectively. For detailed explanations of the meaning of such plots, see Fig. 3.13 and explanation there. Clearly, each stochastic mode Y_i is a linear combination of several, possibly many, gPC bases. Therefore, unlike gPC methods, where the stochastic basis \mathbf{H} is fixed and does not change with time, our DyBO method “agglomerates” the polynomial basis and forms a more efficient stochastic basis \mathbf{Y} . Moreover, this set of stochastic bases is automatically adapted in time without introducing any heuristics for the selection of the



(a) (Sorted) energy spectrum of the gPC and DyBO solutions



(b) Relative errors of STD computed by gPC and DyBO methods

Figure 4.7: Comparisons of gPC and DyBO methods

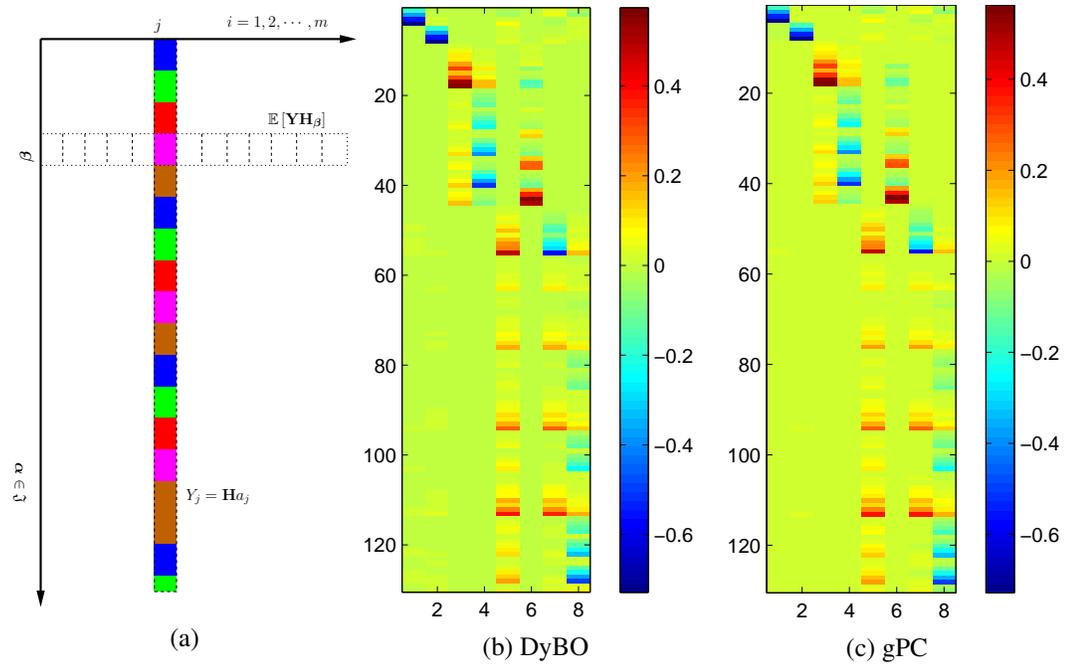


Figure 4.8: Stochastic modes computed by DyBO and gPC

multi-index set.

Chapter 5

Generalization to Stochastic Processes

$$\mathbb{L}^2 \left(\Omega \rightarrow \mathcal{H}^k(\mathcal{D}) \right)$$

As we see in the previous chapters, the DyBO solution u^{DyBO} converges to the true solution of SPDE in the $\|\cdot\|_{\mathbb{L}^2(\mathcal{D} \times \Omega)}$ sense, which turns out to be insufficient or too restrictive in some practical applications. For example, when predictions of deterministic flow and transport in porous media are under considerations, we generally expect numerical solutions to converge to true ones in a stronger norm $\|\cdot\|_{\mathcal{H}^1(\mathcal{D})}$, instead of merely $\|\cdot\|_{\mathbb{L}^2(\mathcal{D})}$, to ensure an accurate estimate of the flux, which is an important quantity in understanding underground flows [93, 57]. Correspondingly, we desire for convergence in the same strong norm $\|\cdot\|_{\mathbb{L}^2(\Omega \rightarrow \mathcal{H}^k(\mathcal{D}))}$, instead of $\|\cdot\|_{\mathbb{L}^2(\Omega \rightarrow \mathbb{L}^2(\mathcal{D}))}$, when porous media is modeled as some stochastic process to account for uncertainty. On the other hand, a weaker norm may be desired for the computational efficiency when uncertainty only concentrates over some isolated small regions within the whole domain or only a part of region are of primary interests.

In this chapter, we generalize the DyBO formulation (2.28) for a stochastic solution $u(\cdot, t, \cdot) \in \mathbb{L}^2(\Omega \rightarrow \mathcal{H}^k(\mathcal{D}))$ by first generalizing the KL expansion while preserving the error-minimizing (Theorem 1.1) and bi-orthogonal (Theorem 1.2) properties of KL in $\mathbb{L}^2(\mathcal{D} \times \Omega)$. The essence of the KL expansion is to find a complete orthonormal set of spatial modes $f_i(x)$'s to minimize the error of an m -term truncated expansion among all such orthonormal sets. To this end, we derive

first in the next section the necessary condition (5.4) for such spatial modes.

5.1 Necessary Condition for Spatial Modes in $\mathbb{L}^2(\Omega \rightarrow \mathcal{V}(\mathcal{D}))$

Assume $\mathcal{V}(\mathcal{D})$ is a separable Hilbert space of functions defined on a compact domain $\mathcal{D} \in \mathbb{R}^d$. $\mathcal{V}(\mathcal{D})$ is equipped with an inner product $\langle \cdot, \cdot \rangle_{\mathcal{V}(\mathcal{D})}$ and an induced norm $\|\cdot\|_{\mathcal{V}(\mathcal{D})}$. Consider a stochastic process of zero mean $u(x, \omega) \in \mathbb{L}^2(\Omega \rightarrow \mathcal{V}(\mathcal{D}))$ where

$$\mathbb{L}^2(\Omega \rightarrow \mathcal{V}(\mathcal{D})) = \left\{ u(x, \omega) \mid \mathbb{E}[u] = 0, \|u\|_{\mathbb{L}^2(\Omega \rightarrow \mathcal{V}(\mathcal{D}))}^2 = \mathbb{E} \left[\|u\|_{\mathcal{V}(\mathcal{D})}^2 \right] < \infty \right\}.$$

Part of the derivation is adapted from Ghanem and Spanos's book (see Chapter 2, [41]). Consider any complete orthonormal set of functions $\{h_i(x)\}_{i=1}^{\infty} \subset \mathcal{V}(\mathcal{D})$, i.e.,

$$\langle h_i, h_j \rangle = \delta_{ij}, \text{ for } i, j = 1, 2, \dots$$

The stochastic process $u(x, \omega)$ admits a Fourier-type series expansion

$$u(x, \omega) = \sum_{i=1}^{\infty} \xi_i(\omega) h_i(x), \quad (5.1)$$

where

$$\xi_i(\omega) = \langle u(x, \omega), h_i(x) \rangle_{\mathcal{V}(\mathcal{D})} \quad (5.2)$$

Then the m -term truncation error is

$$\epsilon_m(x, \omega) = \sum_{i=m+1}^{\infty} \xi_i(\omega) h_i(x),$$

whose norm is

$$\begin{aligned}
\|\epsilon_m\|_{\mathbb{L}^2(\Omega \rightarrow \mathcal{V}(\mathcal{D}))}^2 &= \mathbb{E} \left[\langle \epsilon_m, \epsilon_m \rangle_{\mathcal{V}(\mathcal{D})} \right] \\
&= \mathbb{E} \left[\left\langle \sum_{i=m+1}^{\infty} \xi_i h_i, \sum_{j=m+1}^{\infty} \xi_j h_j \right\rangle_{\mathcal{V}(\mathcal{D})} \right] \\
&= \sum_{i=m+1}^{\infty} \sum_{j=m+1}^{\infty} \mathbb{E} [\xi_i \xi_j] \langle h_i, h_j \rangle_{\mathcal{V}(\mathcal{D})} \\
&= \sum_{i=m+1}^{\infty} \sum_{j=m+1}^{\infty} \mathbb{E} [\xi_i \xi_j] \delta_{ij} \\
&= \sum_{i=m+1}^{\infty} \mathbb{E} \left[\langle u, h_i \rangle_{\mathcal{V}(\mathcal{D})}^2 \right],
\end{aligned}$$

where we have used the orthogonality of h_i 's in the second equality and eqn. (5.2) in the second to last equality. To minimize $\|\epsilon_m\|_{\mathbb{L}^2(\Omega \rightarrow \mathcal{V}(\mathcal{D}))}^2$ subject to the orthogonality of $\{h_i(x)\}_{i=1}^{\infty}$, we use a Lagrange multiplier λ_i and consider the following optimization problem of minimizing functional

$$\begin{aligned}
\mathbf{F} [\{h_i\}_{i=m+1}^{\infty}] &= \|\epsilon_m\|_{\mathbb{L}^2(\Omega \rightarrow \mathcal{V}(\mathcal{D}))}^2 - \sum_{i=m+1}^{\infty} \lambda_i \left(\langle h_i, h_i \rangle_{\mathcal{V}(\mathcal{D})} - 1 \right) \\
&= \sum_{i=m+1}^{\infty} \mathbb{E} \left[\langle u, h_i \rangle_{\mathcal{V}(\mathcal{D})}^2 \right] - \lambda_i \left(\langle h_i, h_i \rangle_{\mathcal{V}(\mathcal{D})} - 1 \right). \tag{5.3}
\end{aligned}$$

Differentiating with respect to h_j for $j = m + 1, m + 2, \dots$, we have

$$\begin{aligned}
\frac{1}{2} \frac{\partial \mathbf{F} (\{h_i\}_{i=1}^{\infty})}{\partial h_j} \delta h_j &= \mathbb{E} \left[\langle u, h_j \rangle_{\mathcal{V}(\mathcal{D})} \langle u, \delta h_j \rangle_{\mathcal{V}(\mathcal{D})} \right] - \lambda_j \langle h_j, \delta h_j \rangle_{\mathcal{V}(\mathcal{D})} \\
&= \mathbb{E} \left[\left\langle \langle u, h_j \rangle_{\mathcal{V}(\mathcal{D})} u, \delta h_j \right\rangle_{\mathcal{V}(\mathcal{D})} \right] - \lambda_j \langle h_j, \delta h_j \rangle_{\mathcal{V}(\mathcal{D})} \\
&= \left\langle \mathbb{E} \left[\langle u, h_j \rangle_{\mathcal{V}(\mathcal{D})} u \right], \delta h_j \right\rangle_{\mathcal{V}(\mathcal{D})} - \lambda_j \langle h_j, \delta h_j \rangle_{\mathcal{V}(\mathcal{D})} \\
&= \left\langle \mathbb{E} \left[\langle u, h_j \rangle_{\mathcal{V}(\mathcal{D})} u \right] - \lambda_j h_j, \delta h_j \right\rangle_{\mathcal{V}(\mathcal{D})}.
\end{aligned}$$

Since δh_j is arbitrary, we obtain the necessary conditions for a complete orthonormal set $\{h_i(x)\}_{i=1}^{\infty}$ being spatial modes in the KL expansion in the generalized sense,

$$\mathbb{E} \left[\langle u, h_j \rangle_{\mathcal{V}(\mathcal{D})} u \right] - \lambda_j h_j = 0, \quad \text{for } j = 1, 2, \dots,$$

which is an eigenvalue problem

$$\mathbb{E} \left[\langle u, h \rangle_{\mathcal{V}(\mathcal{D})} u \right] = \lambda h. \quad (5.4)$$

as explained in the next section.

5.2 Generalized KLE for $\mathbb{L}^2(\Omega \rightarrow \mathcal{H}^k(\mathcal{D}))$

Next we confine ourselves to the Sobolev space, i.e.,

$$\mathcal{V}(\mathcal{D}) = \mathcal{H}^k(\mathcal{D}) = \left\{ f(x) \mid \|f\|_{\mathcal{H}^k(\mathcal{D})}^2 = \sum_{|\gamma| \leq k} \|D^\gamma f\|_{\mathbb{L}^2(\mathcal{D})}^2 < \infty \right\},$$

where $\gamma = (\gamma_1, \gamma_2, \dots, \gamma_d)$ is a multi-index of length d and $D^\gamma f(x) = \frac{\partial^{|\gamma|} f(x_1, x_2, \dots, x_d)}{\partial^{\gamma_1} x_1 \partial^{\gamma_2} x_2 \dots \partial^{\gamma_d} x_d}$. This multi-index γ should *not* be confused with the multi-index α of length N_P used in gPC expansions.

In this section, we consider the stochastic process $u \in \mathbb{L}^2(\Omega \rightarrow \mathcal{H}^k(\mathcal{D}))$. Clearly, the generalized KL expansion of u can be obtained by solving the following eigenfunction problem,

$$\mathcal{A}h = \mathbb{E} \left[\langle u, h \rangle_{\mathcal{H}^k(\mathcal{D})} u \right] = \lambda h. \quad (5.5)$$

In the following two lemmas, we are going to show that \mathcal{A} is a linear, compact, non-negative definite and self-adjoint operator on $\mathcal{H}^k(\mathcal{D})$. We should note that \mathcal{A} is a deterministic operator.

Lemma 5.1. *\mathcal{A} is a linear and compact operator on $\mathcal{H}^k(\mathcal{D})$.*

Proof. The linearity of \mathcal{A} is obvious. For $\forall h \in \mathcal{H}^k(\mathcal{D})$, we will show that $\mathcal{A}h \in \mathcal{H}^k(\mathcal{D})$. By linearity of expectation $\mathbb{E}[\cdot]$ and inner product $\langle \cdot, \cdot \rangle$, for $\forall |\gamma| \leq k$

$$D^\gamma(\mathcal{A}h) = \mathbb{E} \left[\langle u, h \rangle_{\mathcal{H}^k(\mathcal{D})} D^\gamma u \right].$$

So we have

$$\begin{aligned} \|D^\gamma(\mathcal{A}h)\|_{\mathbb{L}^2(\mathcal{D})}^2 &= \left\| \mathbb{E} \left[\langle u, h \rangle_{\mathcal{H}^k(\mathcal{D})} D^\gamma u \right] \right\|_{\mathbb{L}^2(\mathcal{D})}^2 && \text{(Jensen's inequality)} \\ &\leq \left(\mathbb{E} \left[\left\| \langle u, h \rangle_{\mathcal{H}^k(\mathcal{D})} D^\gamma u \right\|_{\mathbb{L}^2(\mathcal{D})} \right] \right)^2 \\ &= \left(\mathbb{E} \left[\left| \langle u, h \rangle_{\mathcal{H}^k(\mathcal{D})} \right| \|D^\gamma u\|_{\mathbb{L}^2(\mathcal{D})} \right] \right)^2 && \text{(Cauchy-Schwarz inequality)} \\ &\leq \mathbb{E} \left[\langle u, h \rangle_{\mathcal{H}^k(\mathcal{D})}^2 \right] \mathbb{E} \left[\|D^\gamma u\|_{\mathbb{L}^2(\mathcal{D})}^2 \right] && \text{(Cauchy-Schwarz inequality)} \\ &\leq \mathbb{E} \left[\|u\|_{\mathcal{H}^k(\mathcal{D})}^2 \|h\|_{\mathcal{H}^k(\mathcal{D})}^2 \right] \mathbb{E} \left[\|D^\gamma u\|_{\mathbb{L}^2(\mathcal{D})}^2 \right] \\ &= \|u\|_{\mathbb{L}^2(\Omega \rightarrow \mathcal{H}^k(\mathcal{D}))}^2 \|h\|_{\mathcal{H}^k(\mathcal{D})}^2 \mathbb{E} \left[\|D^\gamma u\|_{\mathbb{L}^2(\mathcal{D})}^2 \right]. \end{aligned}$$

Summing over all $|\gamma| \leq k$, we have

$$\|\mathcal{A}h\|_{\mathcal{H}^k(\mathcal{D})}^2 \leq \|u\|_{\mathbb{L}^2(\Omega \rightarrow \mathcal{H}^k(\mathcal{D}))}^4 \|h\|_{\mathcal{H}^k(\mathcal{D})}^2,$$

or

$$\|\mathcal{A}h\|_{\mathcal{H}^k(\mathcal{D})} \leq \|u\|_{\mathbb{L}^2(\Omega \rightarrow \mathcal{H}^k(\mathcal{D}))}^2 \|h\|_{\mathcal{H}^k(\mathcal{D})}. \quad (5.6)$$

This implies that $\mathcal{A} : \mathcal{H}^k(\mathcal{D}) \rightarrow \mathcal{H}^k(\mathcal{D})$ is compact as long as $u \in \mathbb{L}^2(\Omega \rightarrow \mathcal{H}^k(\mathcal{D}))$. \square

Lemma 5.2. *\mathcal{A} is self-adjoint and non-negative definite..*

Proof. The self-adjointness of operator \mathcal{A} , i.e.,

$$\langle \mathcal{A}h, f \rangle_{\mathcal{H}^k(\mathcal{D})} = \langle h, \mathcal{A}f \rangle_{\mathcal{H}^k(\mathcal{D})}, \quad \forall f, h \in \mathcal{H}^k(\mathcal{D}),$$

follows from the following equality,

$$\begin{aligned} \langle \mathcal{A}h, f \rangle_{\mathcal{H}^k(\mathcal{D})} &= \left\langle \mathbb{E} \left[\langle u, h \rangle_{\mathcal{H}^k(\mathcal{D})} u \right], f \right\rangle_{\mathcal{H}^k(\mathcal{D})} && \text{(because } f \text{ is deterministic)} \\ &= \mathbb{E} \left[\left\langle \langle u, h \rangle_{\mathcal{H}^k(\mathcal{D})} u, f \right\rangle_{\mathcal{H}^k(\mathcal{D})} \right] \\ &= \mathbb{E} \left[\langle u, h \rangle_{\mathcal{H}^k(\mathcal{D})} \langle u, f \rangle_{\mathcal{H}^k(\mathcal{D})} \right]. \end{aligned}$$

Therefore, $\langle \mathcal{A}h, h \rangle_{\mathcal{H}^k(\mathcal{D})} = \mathbb{E} \left[\langle u, h \rangle_{\mathcal{H}^k(\mathcal{D})}^2 \right] \geq 0$. The equality implies that $\langle u, h \rangle_{\mathcal{H}^k(\mathcal{D})} = 0$ almost surely. Clearly, \mathcal{A} is a non-negative definite self-adjoint operator. \square

By applying Mercer's theorem, we have the following theorem concerning the solution of the eigenvalue problem (5.5).

Theorem 5.3. *The eigenvalues λ_i are non-negative and may cluster at 0 only. The eigenfunctions h_i 's form a complete orthogonal basis of $\mathcal{H}^k(\mathcal{D})$.*

We will assume the eigenfunctions are properly normalized and form an orthonormal set, i.e., $\langle h_i, h_j \rangle_{\mathcal{H}^k(\mathcal{D})} = \delta_{ij}$. The stochastic process $u \in \mathbb{L}^2(\Omega \rightarrow \mathcal{H}^k(\mathcal{D}))$ admits the following generalized Karhunen Loeve expansion (gKLE),

$$u(x, \omega) = \sum_{i=1}^{\infty} \xi_i(\omega) h_i(x), \tag{5.7}$$

with $\xi_i(\omega) = \langle u(x, \omega), h_i(x) \rangle_{\mathcal{H}^k(\mathcal{D})}$. Next lemma shows the bi-orthogonality of gKLE

Lemma 5.4. *ξ_i 's in gKLE (5.7) are orthogonal to each other and $\mathbb{E} [\xi_i^2] = \lambda_i$*

Proof. Substituting gKLE (5.7) into the eigenvalue problem (5.5) yields

$$\mathbb{E} \left[\xi_j \sum_{i=1}^{\infty} \xi_i h_i \right] = \lambda_j h_j,$$

or

$$\sum_{i=1}^{\infty} \mathbb{E} [\xi_j \xi_i] h_i = \lambda_j h_j.$$

Taking inner product with h_k , we have

$$\sum_{i=1}^{\infty} \mathbb{E} [\xi_i \xi_j] \langle h_i, h_k \rangle_{\mathcal{H}^k(\mathcal{D})} = \lambda_j \langle h_j, h_k \rangle_{\mathcal{H}^k(\mathcal{D})},$$

or

$$\mathbb{E} [\xi_k \xi_j] = \lambda_j \delta_{jk}.$$

□

In practice, we also normalize the stochastic modes and write the gKLE of u as

$$u(x, \omega) = \sum_{i=1}^{\infty} \sqrt{\lambda_i} \hat{\xi}_i(\omega) h_i(x), \quad (5.8)$$

where $\hat{\xi} = \frac{1}{\sqrt{\lambda_i}} \xi = \frac{1}{\sqrt{\lambda_i}} \langle u, h_i \rangle_{\mathcal{H}^k(\mathcal{D})}$.

Obviously, gKLE has the desired error-minimizing property in the sense of $\|\cdot\|_{\mathbb{L}^2(\Omega \rightarrow \mathcal{H}^k(\mathcal{D}))}$.

The bi-orthogonality of gKLE is shown in the following theorem.

Theorem 5.5. *If the eigenvalues of the covariance function of the stochastic process $u(x, \omega) \in \mathbb{L}^2(\Omega \rightarrow \mathcal{H}^k(\mathcal{D}))$ are distinct, it admits a unique bi-orthogonal expansion (5.8), and λ_i and h_i are eigenvalue and eigenfunction solving (5.5).*

Proof. We only need to show that the spatial modes h_i 's are eigenfunctions of \mathcal{A} , i.e.,

$$\mathcal{A}h_i = \mathbb{E} \left[\langle u, h_i \rangle_{\mathcal{H}^k(\mathcal{D})} u \right] = \mathbb{E} \left[\sqrt{\lambda_i} \hat{\xi}_i u \right] = \lambda_i h_i,$$

where we have used the bi-orthogonality property of the spatial and stochastic modes in the second and third equalities. Since we assume the eigenvalues are distinct, the above eigenfunction problem admits a unique set of solutions up to signs. This proves the theorem. \square

Remark 5.1. When $k = 0$, \mathcal{A} reduces to the covariance kernel. This can be seen by computing directly

$$\begin{aligned} \mathcal{A}h(x) &= \mathbb{E} \left[\langle u, h \rangle_{\mathbb{L}^2(\mathcal{D})} u(x, \omega) \right] \\ &= \mathbb{E} \left[\int_{\mathcal{D}} u(y, \omega) h(y) \, dy u(x, \omega) \right] \\ &= \mathbb{E} \left[\int_{\mathcal{D}} u(x, \omega) u(y, \omega) h(y) \, dy \right] \\ &= \int_{\mathcal{D}} \mathbb{E} [u(x, \omega) u(y, \omega)] h(y) \, dy \\ &= \int_{\mathcal{D}} \text{Cov}_u(x, y) h(y) \, dy. \end{aligned}$$

Remark 5.2. When $k \geq 1$, the corresponding eigenvalue problem (5.5) is no longer of Fredholm integral equation type as demonstrated below for a simple case where $k = 1$ and $d = 1$.

$$\begin{aligned} \mathcal{A}h(x) &= \mathbb{E} \left[\langle u, h \rangle_{\mathcal{H}^1(\mathcal{D})} u(x, \omega) \right] \\ &= \mathbb{E} \left[\langle u, h \rangle_{\mathbb{L}^2(\mathcal{D})} u(x, \omega) \right] + \mathbb{E} \left[\left\langle \frac{\partial u}{\partial x}, \frac{dh}{dx} \right\rangle_{\mathbb{L}^2(\mathcal{D})} u(x, \omega) \right] \\ &= \int_{\mathcal{D}} \mathbb{E} [u(x, \omega) u(y, \omega)] h(y) \, dy + \int_{\mathcal{D}} \mathbb{E} \left[u(x, \omega) \frac{\partial u}{\partial x}(y, \omega) \right] \frac{dh}{dx}(y) \, dy \\ &= \int_{\mathcal{D}} \text{Cov}_u(x, y) h(y) \, dy + \int_{\mathcal{D}} \mathbb{E} \left[u(x, \omega) \frac{\partial u}{\partial x}(y, \omega) \right] \frac{dh}{dx}(y) \, dy. \end{aligned}$$

5.3 Generalized DyBO for $\mathbb{L}^2(\Omega \rightarrow \mathcal{H}^k(\mathcal{D}))$

Now we are ready to present our generalized DyBO for a SPDE solution $u(\cdot, t, \cdot) \in \mathbb{L}^2(\Omega \rightarrow \mathcal{H}^k(\mathcal{D}))$.

$$\frac{\partial \bar{u}}{\partial t} = \mathbb{E}[\mathcal{L}u], \quad (5.9a)$$

$$\frac{\partial \mathbf{U}}{\partial t} = -\mathbf{U}\mathbf{D}^T + \mathbb{E}[\tilde{\mathcal{L}}u\mathbf{Y}], \quad (5.9b)$$

$$\frac{d\mathbf{Y}}{dt} = -\mathbf{Y}\mathbf{C}^T + \left\langle \tilde{\mathcal{L}}u, \mathbf{U} \right\rangle_{\mathcal{H}^k(\mathcal{D})} \mathbf{\Lambda}_{\mathbf{U}}^{-1}, \quad (5.9c)$$

where $\mathbf{C}(t)$ and $\mathbf{D}(t)$ can be solved from the linear system (2.21) via (2.22) with

$$G_*(u, \mathbf{U}, \mathbf{Y}) = \mathbf{\Lambda}_{\mathbf{U}}^{-1} \left\langle \mathbf{U}^T, \mathbb{E}[\tilde{\mathcal{L}}u\mathbf{Y}] \right\rangle_{\mathcal{H}^k(\mathcal{D})}. \quad (5.10)$$

When the stochastic modes \mathbf{Y} are represented by gPC expansions as described in Sec. 2.3.2, i.e.,

$\mathbf{Y} = \mathbf{H}\mathbf{A}$, the DyBO-gPC formulation is

$$\frac{\partial \bar{u}}{\partial t} = \mathbb{E}[\mathcal{L}u], \quad (5.11a)$$

$$\frac{\partial \mathbf{U}}{\partial t} = -\mathbf{U}\mathbf{D}^T + \mathbb{E}[\tilde{\mathcal{L}}u\mathbf{H}] \mathbf{A}, \quad (5.11b)$$

$$\frac{d\mathbf{A}}{dt} = -\mathbf{A}\mathbf{C}^T + \left\langle \mathbb{E}[\mathbf{H}^T \tilde{\mathcal{L}}u], \mathbf{U} \right\rangle_{\mathcal{H}^k(\mathcal{D})} \mathbf{\Lambda}_{\mathbf{U}}^{-1}, \quad (5.11c)$$

where $\mathbf{C}(t)$ and $\mathbf{D}(t)$ can be solved from (2.22) with

$$G_*(u, \mathbf{U}, \mathbf{Y}) = \mathbf{\Lambda}_{\mathbf{U}}^{-1} \left\langle \mathbf{U}^T, \mathbb{E}[\tilde{\mathcal{L}}u\mathbf{H}] \right\rangle_{\mathcal{H}^k(\mathcal{D})} \mathbf{A}. \quad (5.12)$$

Remark 5.3. Comparing the generalized DyBO-gPC formulation (5.11) and the DyBO-gPC for-

mulation (2.39) discussed in Chapter 2, we see that the essential difference is at the definition of the inner product $\langle \cdot, \cdot \rangle$. Therefore, the discussions regarding the bi-orthogonality preservation and error propagation in Chapter 2 can be trivially extended to the generalized DyBO formulations here. What's more, the strategies proposed in the previous chapters to overcome some issues in numerical implementations, such as eigenvalue crossing and removing/adding mode pairs, are also applicable to the generalized formulations.

Remark 5.4. The appropriate definition of the inner product $\langle \cdot, \cdot \rangle$ depends on the specific applications. Certainly, a different definition of the inner product gives a different convergence property of the DyBO solution. It will also affect the number of terms that we need to keep in the gKLE, which will in turn affect the computational cost. In general, if a stronger inner product is used in the DyBO formulation, we will need to take more terms, i.e., larger m , which gives rise to higher computational cost. From a numerical implementation perspective, such definition of inner products can be encapsulated in a single subroutine. Without changing the rest of codes, different definitions of inner products can be tested easily.

Chapter 6

Generalizations of DyBO for a System of Time-Dependent SPDEs

In the previous chapters, we have demonstrated both analytically and numerically the successes of DyBO methods for a single time-dependent SPDE. However, many applications involve multiple physical fields, or physical components, i.e., a system of SPDEs. For example, the standard three-dimensional incompressible Navier-Stokes equations involve four physical components, three velocity components along x -, y -, z -axis and pressure. When compressibility cannot be ignored, e.g., in aerodynamics [4], two additional components, typically density and temperature fields, get involved. For multiphase flow problems, combustion problems, chemical reaction flow problems and the like, more equations are needed for additional physical components.

To demonstrate that real-world numerical simulations may continue to benefit from DyBO methods, we generalize DyBO methods for a system of time-dependent SPDEs in this chapter. More precisely, we consider

$$\frac{\partial u_l}{\partial t}(x, t, \omega) = \mathcal{L}_l \{u_1, u_2, \dots, u_{N_s}\}, \quad l = 1, 2, \dots, N_s, \quad x \in \mathcal{D} \subset \mathbb{R}^d, \quad t \in [0, T], \quad (6.1)$$

where each \mathcal{L}_l is a stochastic differential operator acting on the physical components u_1, u_2, \dots, u_{N_s} .

When no ambiguity arises, we simply use shorthand notation

$$u = \{u_1, u_2, \dots, u_{N_s}\} \quad \text{and} \quad \mathcal{L}_l u = \mathcal{L}_l \{u_1, u_2, \dots, u_{N_s}\}.$$

For simplicity, we also assume periodic boundary conditions for all physical components and suitable initial conditions.

The development of our DyBO method for a system of SPDEs is similar to that for a single SPDE in Chapter 2. In the next section, we just outline the derivation and highlight some key steps and results. Appendix C provides more details. In Sec. 6.3, we consider 2D stochastic Rayleigh-Benard convection as a test model for the generalized DyBO method. When more physical components get involved or spatially three-dimensional simulations are under considerations, effective parallelization strategies are necessary to achieve numerical results in reasonable time. Based on the computational complexity analysis in Chapter 4.1, we propose a simple, yet powerful, parallelization strategy for this numerical example.

6.1 DyBO for a System of SPDEs

Unlike a single SPDE, randomnesses introduced through initial conditions, boundary conditions, stochastic forcing terms not only propagate in space and time, but also between different physical components. Randomness introduced by one physical component may affect other components. Generally, different physical components respond to randomness differently, so a common basis, such as the ones used in gPC methods, may still work in some numerical simulation, but not work most efficiently. The most compact representations in \mathbb{L}^2 sense are the KL expansions of each physical component, which is our starting point to derive the DyBO formulation for system (6.1).

Consider the m_l -term truncated KL expansion of the l th physical component $u_l(x, t, \omega)$,

$$u_l = \bar{u}_l + \sum_{i=1}^{m_l} u_{li} Y_{li} = \bar{u}_l + \mathbf{U}_l \mathbf{Y}_l^T, \quad (6.2)$$

where \mathbf{U}_l is a row vector of functions of spatial coordinate x and temporal coordinate t ,

$$\mathbf{U}_l(x, t) = (u_{l1}(x, t), u_{l2}(x, t), \dots, u_{lm_l}(x, t)) \in \mathbb{R}^{1 \times m_l},$$

and \mathbf{Y}_l is a row vector of random variables,

$$\mathbf{Y}_l(\omega, t) = (Y_{l1}(\omega, t), Y_{l2}(\omega, t), \dots, Y_{lm_l}(\omega, t)) \in \mathbb{R}^{1 \times m_l}.$$

Remark 6.1. The physical components should not be confused with spatial modes. The former refers to the stochastic solutions of a system of SPDEs and each component represents some physical quantity, e.g., x -velocity. The latter refers to the eigenfunctions in the KL expansion of some stochastic processes. One physical component may have several physical modes.

By following the steps in the DyBO derivation of a single SPDE, i.e., plugging the expansion (6.2) into the system (6.1), using anti-symmetrization \mathcal{Q} and $\tilde{\mathcal{Q}}$ operators to enforce the bi-orthogonality of spatial and stochastic modes \mathbf{U}_l and \mathbf{Y}_l of each physical component u_l , and projecting the growth rate of spatial and stochastic modes $\frac{\partial \mathbf{U}_l}{\partial t}$ and $\frac{d \mathbf{Y}_l}{dt}$ onto themselves, we arrive at the generalized DyBO formulation for (6.1), for $l = 1, 2, \dots, N_s$,

$$\frac{\partial \bar{u}_l}{\partial t} = \mathbb{E}[\mathcal{L}_l u], \quad (6.3a)$$

$$\frac{\partial \mathbf{U}_l}{\partial t} = -\mathbf{U}_l \mathbf{D}_l^T + \mathbb{E}[\tilde{\mathcal{L}}_l u \mathbf{Y}_l], \quad (6.3b)$$

$$\frac{d \mathbf{Y}_l}{dt} = -\mathbf{Y}_l \mathbf{C}_l^T + \langle \tilde{\mathcal{L}}_l u, \mathbf{U}_l \rangle \mathbf{\Lambda}_{\mathbf{U}_l}^{-1}, \quad (6.3c)$$

where matrices \mathbf{C}_l 's and \mathbf{D}_l 's can be solved from linear systems

$$\mathbf{C}_l - \mathbf{\Lambda}_{\mathbf{U}_l}^{-1} \tilde{\mathcal{Q}} (\mathbf{\Lambda}_{\mathbf{U}_l} \mathbf{C}_l) = 0, \quad (6.4a)$$

$$\mathbf{D}_l - \mathcal{Q} (\mathbf{D}_l) = 0, \quad (6.4b)$$

$$\mathbf{D}_l^T + \mathbf{C}_l = G_{*l}(u, \mathbf{U}_l, \mathbf{Y}_l), \quad (6.4c)$$

with $G_{*l}(u, \mathbf{U}_l, \mathbf{Y}_l) = \mathbf{\Lambda}_{\mathbf{U}_l}^{-1} \left\langle \mathbf{U}_l^T, \mathbb{E} \left[\tilde{\mathcal{L}}_l u \mathbf{Y}_l \right] \right\rangle \in \mathbb{R}^{m_l \times m_l}$.

For DyBO-gPC, the stochastic modes \mathbf{Y}_l are presented in the form of gPC expansions, i.e.,

$$\mathbf{Y}_l(\omega, t) = \mathbf{H}(\boldsymbol{\xi}(\omega)) \mathbf{A}_l,$$

where $\mathbf{A}_l \in \mathbb{R}^{N_p \times m_l}$. The DyBO-gPC formulation is, for $l = 1, 2, \dots, N_s$,

$$\frac{\partial \bar{u}_l}{\partial t} = \mathbb{E} [\mathcal{L}_l u], \quad (6.5a)$$

$$\frac{\partial \mathbf{U}_l}{\partial t} = -\mathbf{U}_l \mathbf{D}_l^T + \mathbb{E} \left[\tilde{\mathcal{L}}_l u \mathbf{H} \right] \mathbf{A}_l, \quad (6.5b)$$

$$\frac{d\mathbf{A}_l}{dt} = -\mathbf{A}_l \mathbf{C}_l^T + \left\langle \mathbb{E} \left[\mathbf{H}^T \tilde{\mathcal{L}}_l u \right], \mathbf{U}_l \right\rangle \mathbf{\Lambda}_{\mathbf{U}_l}^{-1}, \quad (6.5c)$$

where $\mathbf{C}_l(t)$ and $\mathbf{D}_l(t)$ can be solved from

$$G_{*l}(u, \mathbf{U}_l, \mathbf{Y}_l) = \mathbf{\Lambda}_{\mathbf{U}_l}^{-1} \left\langle \mathbf{U}_l^T, \mathbb{E} \left[\tilde{\mathcal{L}}_l u \mathbf{Y}_l \right] \right\rangle = \mathbf{\Lambda}_{\mathbf{U}_l}^{-1} \left\langle \mathbf{U}_l^T, \mathbb{E} \left[\tilde{\mathcal{L}}_l u \mathbf{H} \right] \right\rangle \mathbf{A}_l. \quad (6.6)$$

Please see Appendix C for details.

Various theoretical results in Chapter 2, such as preservation of bi-orthogonality and error analysis, can be trivially generalized to the generalized DyBO formulation for a SPDE system. Furthermore, strategies proposed in Chapter 2.2 for several numerical implementation issues, such as

eigenvalues crossing and adding new mode pairs, are also applicable here.

6.2 Stochastic Navier-Stokes Equations with Boussinesq Approximation

As a model to test numerically the proposed DyBO formulation for a SPDE system, we consider the Navier-Stokes equations whose velocity components are driven by *both* stochastic forces and buoyancy forces due to small density difference induced by temperature variations. Specifically, we consider the stochastic flow in a unit square, i.e., $\mathcal{D} = [0, 1] \times [0, 1]$, with periodic boundary conditions on both spatial directions. See Fig. 6.1a.

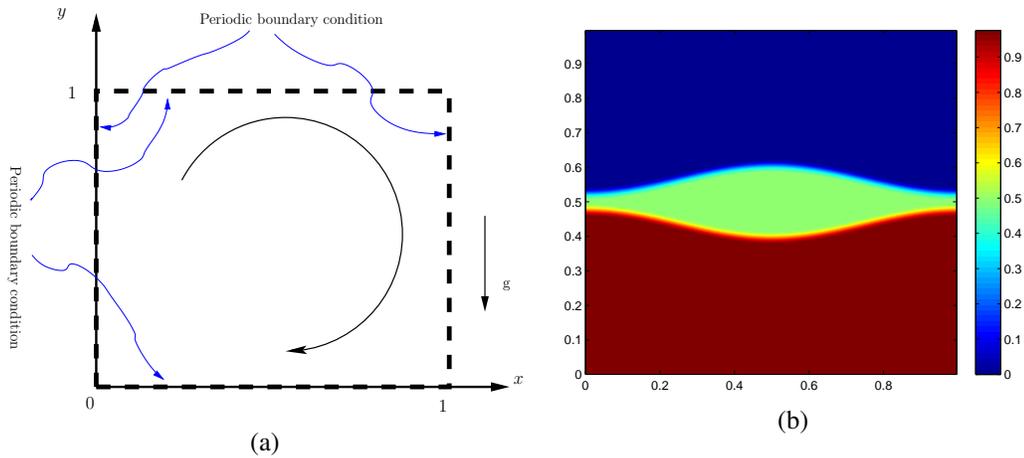


Figure 6.1: Stochastic flow driven by stochastic force and buoyancy force due to Boussinesq approximation. On the left: Diagram of the stochastic flow in an unit square. The gravity is downward parallel to y -axis and periodic boundary conditions are assumed on both x and y directions. On the right, the initial temperature field is plotted, while the initial vorticity is uniformly zero.

The temperature field is not spatially uniform and causes variations of the density field. Such variations are small because we assume the thermal expansion coefficient is very small. The induced buoyancy force may drive the flow motion in addition to external stochastic forces. Here, we adopt the Boussinesq approximation to model such buoyancy force. The governing SPDE of such

stochastic flow in Fig. 6.1a is the Stochastic Navier-Stokes equations (SNSE)

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0, \quad (6.7a)$$

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = -\frac{\partial p}{\partial x} + \nu \Delta u + f_1, \quad (6.7b)$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} = -\frac{\partial p}{\partial y} + \nu \Delta v + f_2 + \mu g \theta, \quad (6.7c)$$

$$\frac{\partial \theta}{\partial t} + u \frac{\partial \theta}{\partial x} + v \frac{\partial \theta}{\partial y} = \kappa \Delta \theta, \quad (6.7d)$$

where the meanings of symbols are tabulated below

$u(x, t, \omega), v(x, t, \omega)$	-	x and y components of velocity
$p(x, t, \omega)$	-	Pressure with static hydraulic part removed
$f_1(x, t, \omega), f_2(x, t, \omega)$	-	x and y components of zero-mean the stochastic force f
$\theta(x, t, \omega)$	-	temperature
deterministic constant ν	-	Dynamic viscosity
deterministic constant κ	-	Thermal diffusivity
deterministic constant μ	-	Thermal expansion coefficient
deterministic constant g	-	The gravity of Earth

For the constants, we have chosen $\nu = 2.0 \times 10^{-4}$, $\kappa = 2.0 \times 10^{-4}$ and scaled gravity $\mu g = 11.31$, partially following the paper by Cenicerros and Hou [21].

First glance, this numerical example is similar to the one in Sec. 4.2.2 except introducing temperature field and gravity effect. However, the stochastic flow is very different from the one considered previously for the following reason. We see that the buoyancy force depends on the gradient of temperature field and is actually a stochastic force. In other words, the stochastic flow considered in this section is driven by two kinds of stochastic forces: one “*external*” stochastic force injecting randomness from the ambient environment into the unit square and one “*internal*” stochastic force feeding randomness back to the system, from temperature component to vertical velocity component. Such stochastic flow provides a severe test model for our generalized DyBO methods.

As we did in Sec. 4.2.2, it is convenient to consider the vorticity-stream formulation instead of

the standard one for incompressible flows in 2D. By introducing the vorticity $w = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}$ and stream function ψ satisfying $u = \frac{\partial \psi}{\partial y}$ and $v = -\frac{\partial \psi}{\partial x}$, the above pressure-velocity formulation of SNSE can be transformed to the vorticity-stream formulation,

$$\begin{aligned} \frac{\partial \theta}{\partial t} + \left(u \frac{\partial}{\partial x} + v \frac{\partial}{\partial y} \right) \theta &= \kappa \Delta \theta, \\ \frac{\partial w}{\partial t} + \left(u \frac{\partial}{\partial x} + v \frac{\partial}{\partial y} \right) w &= \nu \Delta w + \left(\frac{\partial f_2}{\partial x} - \frac{\partial f_1}{\partial y} \right) + \mu g \frac{\partial \theta}{\partial x}, \\ -\Delta \psi &= w, \\ u = \frac{\partial \psi}{\partial y}, \quad v &= -\frac{\partial \psi}{\partial x}, \end{aligned}$$

or cast in the standard form defined in the system of SPDEs (6.1),

$$\frac{\partial \theta}{\partial t} = \mathcal{L}_\theta \{ \theta, w \} = - \left(u \frac{\partial}{\partial x} + v \frac{\partial}{\partial y} \right) \theta + \kappa \Delta \theta, \quad (6.9a)$$

$$\frac{\partial w}{\partial t} = \mathcal{L}_w \{ \theta, w \} = - \left(u \frac{\partial}{\partial x} + v \frac{\partial}{\partial y} \right) w + \nu \Delta w + \left(\frac{\partial f_2}{\partial x} - \frac{\partial f_1}{\partial y} \right) + \mu g \frac{\partial \theta}{\partial x}, \quad (6.9b)$$

where

$$-\Delta \psi = w, \quad (6.10)$$

$$u = \frac{\partial \psi}{\partial y} = -\frac{\partial}{\partial y} \Delta^{-1} w, \quad (6.11)$$

$$v = -\frac{\partial \psi}{\partial x} = \frac{\partial}{\partial x} \Delta^{-1} w. \quad (6.12)$$

We assume the randomness is given in terms of N_p independent standard Gaussian random variables, $\boldsymbol{\xi} = (\xi_1, \xi_2, \dots, \xi_{N_p})$, and the initial conditions are deterministic, i.e., $\theta(x, 0, \boldsymbol{\xi}) = \hat{\theta}(x)$ and $w(x, 0, \boldsymbol{\xi}) = \hat{w}(x)$. Let \mathfrak{J} be a multi-index set of finite size. We assume that the stochastic force has

a finite gPC expansion, i.e.,

$$\frac{\partial f_2}{\partial x} - \frac{\partial f_1}{\partial y} = \sum_{\alpha \in \mathfrak{J}} F_{\alpha} \mathbf{H}_{\alpha} = \mathbf{F} \mathbf{H}^T.$$

6.2.1 gPC Formulation of SNSE

Consider the finite-term gPC expansion of the stochastic solutions of SNSE (6.9),

$$\theta = \bar{\theta} + \boldsymbol{\theta} \mathbf{H}^T, \quad (6.13a)$$

$$w = \bar{w} + \mathbf{W} \mathbf{H}^T, \quad (6.13b)$$

where row vectors $\boldsymbol{\theta} = (\theta_{\alpha})_{\alpha \in \mathfrak{J}}$ and $\mathbf{W} = (w_{\alpha})_{\alpha \in \mathfrak{J}}$. By plugging the above expansion into SNSE (6.9), we can obtain the gPC formulation of SNSE. However, the derivation is technical. We leave the details in Appendix D and give the main result here.

$$\frac{\partial \bar{\theta}}{\partial t} = \kappa \Delta \bar{\theta} - \mathfrak{D}_{(\bar{u}, \bar{v})} \bar{\theta} - \mathfrak{D}_{(\mathbf{U}, \mathbf{V})} \boldsymbol{\theta}, \quad (6.14a)$$

$$\frac{\partial \bar{w}}{\partial t} = \nu \Delta \bar{w} - \mathfrak{D}_{(\bar{u}, \bar{v})} \bar{w} - \mathfrak{D}_{(\mathbf{U}, \mathbf{V})} \mathbf{W} + \mu g \frac{\partial \bar{\theta}}{\partial x}, \quad (6.14b)$$

$$\frac{\partial \boldsymbol{\theta}}{\partial t} = \kappa \Delta \boldsymbol{\theta} - \mathfrak{D}_{(\bar{u}, \bar{v})} \boldsymbol{\theta} - \mathfrak{D}_{(\mathbf{U}, \mathbf{V})} \bar{\theta} - \left(\mathfrak{D}_{(u_{\alpha}, v_{\alpha})} \theta_{\beta} \mathfrak{F}_{\alpha\beta\gamma}^{(\mathbf{H})} \right)_{1 \times \gamma}, \quad (6.14c)$$

$$\frac{\partial \mathbf{W}}{\partial t} = \nu \Delta \mathbf{W} - \mathfrak{D}_{(\bar{u}, \bar{v})} \mathbf{W} - \mathfrak{D}_{(\mathbf{U}, \mathbf{V})} \bar{w} + \mu g \frac{\partial \boldsymbol{\theta}}{\partial x} - \left(\mathfrak{D}_{(u_{\alpha}, v_{\alpha})} w_{\beta} \mathfrak{F}_{\alpha\beta\gamma}^{(\mathbf{H})} \right)_{1 \times \gamma} + \mathbf{F}, \quad (6.14d)$$

where $\mathfrak{D}_{(\cdot, \cdot)}(\cdot)$ is the generalized material derivative defined in eqn. (D.9) in Appendix D.

6.2.2 DyBO Formulation of SNSE

Now consider the finite-term KL expansion of the solutions of SNSE (6.9),

$$\boldsymbol{\theta} = \bar{\boldsymbol{\theta}} + \boldsymbol{\theta} \mathbf{Y}^T = \bar{\boldsymbol{\theta}} + \boldsymbol{\theta} \mathbf{A}^T \mathbf{H}^T, \quad (6.15a)$$

$$w = \bar{w} + \mathbf{W} \mathbf{Z}^T = \bar{w} + \mathbf{W} \mathbf{B}^T \mathbf{H}^T, \quad (6.15b)$$

where row vectors $\boldsymbol{\theta} = (\theta_1, \theta_2, \dots, \theta_{m_\theta})$ and $\mathbf{W} = (w_1, w_2, \dots, w_{m_w})$, and matrices $\mathbf{A} \in \mathbb{R}^{N_P \times m_\theta}$ and $\mathbf{B} \in \mathbb{R}^{N_P \times m_w}$. We also write mode number vector $\mathbf{m} = (m_\theta, m_w)$. By plugging the above expansion into eqn. (6.5), we obtain the DyBO-gPC formulation for SNSE (6.9).

$$\frac{\partial \bar{\boldsymbol{\theta}}}{\partial t} = \kappa \Delta \bar{\boldsymbol{\theta}} - \mathfrak{D}_{(\bar{u}, \bar{v})} \bar{\boldsymbol{\theta}} - \mathfrak{D}_{(\mathbf{U}, \mathbf{V})} (\boldsymbol{\theta} \mathbf{A}^T \mathbf{B}), \quad (6.16a)$$

$$\begin{aligned} \frac{\partial \boldsymbol{\theta}}{\partial t} = & -\boldsymbol{\theta} \mathbf{D}_\theta^T + \kappa \Delta \boldsymbol{\theta} - \mathfrak{D}_{(\bar{u}, \bar{v})} \boldsymbol{\theta} - \mathfrak{D}_{(\mathbf{U}, \mathbf{V})} \bar{\boldsymbol{\theta}} \mathbf{B}^T \mathbf{A} \\ & - \left[\mathfrak{D}_{(u_i, v_i)} \theta_j B_{\alpha i} A_{\beta j} A_{\gamma k} \mathfrak{F}_{\alpha \beta \gamma}^{(\mathbf{H})} \right]_{1 \times k}, \end{aligned} \quad (6.16b)$$

$$\begin{aligned} \frac{d\mathbf{A}}{dt} = & \mathbf{A} \left(-\mathbf{C}_\theta^T + \kappa \langle \Delta \boldsymbol{\theta}^T, \boldsymbol{\theta} \rangle \boldsymbol{\Lambda}_\theta^{-1} - \langle (\mathfrak{D}_{(\bar{u}, \bar{v})} \boldsymbol{\theta})^T, \boldsymbol{\theta} \rangle \boldsymbol{\Lambda}_\theta^{-1} \right) \\ & - \mathbf{B} \langle (\mathfrak{D}_{(\mathbf{U}, \mathbf{V})} \bar{\boldsymbol{\theta}})^T, \boldsymbol{\theta} \rangle \boldsymbol{\Lambda}_\theta^{-1} - \left[\mathfrak{F}_{ijk}^{(\theta)} B_{\alpha i} A_{\beta j} \mathfrak{F}_{\alpha \beta \gamma}^{(\mathbf{H})} \right]_{\gamma \times k} \boldsymbol{\Lambda}_\theta^{-1}, \end{aligned} \quad (6.16c)$$

$$\frac{\partial \bar{w}}{\partial t} = \left(\nu \Delta \bar{w} - \mathfrak{D}_{(\bar{u}, \bar{v})} \bar{w} + \mu g \frac{\partial \bar{\boldsymbol{\theta}}}{\partial x} \right) - \mathfrak{D}_{(\mathbf{U}, \mathbf{V})} \mathbf{W}, \quad (6.16d)$$

$$\begin{aligned} \frac{\partial \mathbf{W}}{\partial t} = & -\mathbf{W} \mathbf{D}_w^T + [\nu \Delta \mathbf{W} - \mathfrak{D}_{(\bar{u}, \bar{v})} \mathbf{W} - \mathfrak{D}_{(\mathbf{U}, \mathbf{V})} \bar{w}] + \mu g \frac{\partial \boldsymbol{\theta}}{\partial x} \mathbf{A}^T \mathbf{B} \\ & - \left[\mathfrak{D}_{(u_i, v_i)} w_j B_{\alpha i} B_{\beta j} B_{\gamma k} \mathfrak{F}_{\alpha \beta \gamma}^{(\mathbf{H})} \right]_{1 \times k} + \mathbf{F} \mathbf{B}, \end{aligned} \quad (6.16e)$$

$$\begin{aligned} \frac{d\mathbf{B}}{dt} = & \mathbf{B} \left(-\mathbf{C}_w^T + \langle \nu \Delta \mathbf{W}^T - (\mathfrak{D}_{(\bar{u}, \bar{v})} \mathbf{W})^T - (\mathfrak{D}_{(\mathbf{U}, \mathbf{V})} \bar{w})^T, \mathbf{W} \rangle \boldsymbol{\Lambda}_\mathbf{W}^{-1} \right) \\ & + \mu g \mathbf{A} \left\langle \frac{\partial \boldsymbol{\theta}^T}{\partial x}, \mathbf{W} \right\rangle \boldsymbol{\Lambda}_\mathbf{W}^{-1} - \left[\mathfrak{F}_{ijk}^{(\mathbf{W})} B_{\alpha i} B_{\beta j} \mathfrak{F}_{\alpha \beta \gamma}^{(\mathbf{H})} \right]_{\gamma \times k} \boldsymbol{\Lambda}_\mathbf{W}^{-1} + \langle \mathbf{F}^T, \mathbf{W} \rangle \boldsymbol{\Lambda}_\mathbf{W}^{-1}, \end{aligned} \quad (6.16f)$$

where matrices \mathbf{C}_θ and \mathbf{D}_θ can be solved via eqn. (2.22) from $G_{*\theta}$,

$$\begin{aligned} \Lambda_\theta G_{*\theta} = & \kappa \langle \boldsymbol{\theta}^T, \Delta \boldsymbol{\theta} \rangle - \langle \boldsymbol{\theta}^T, \mathfrak{D}_{(\bar{u}, \bar{v})} \boldsymbol{\theta} \rangle - \langle \boldsymbol{\theta}^T, \mathfrak{D}_{(\mathbf{U}, \mathbf{V})} \bar{\boldsymbol{\theta}} \rangle \mathbf{B}^T \mathbf{A} \\ & - \left[\mathfrak{Z}_{ijk}^{(\boldsymbol{\theta})} B_{\alpha i} A_{\beta j} A_{\gamma l} \mathfrak{Z}_{\alpha\beta\gamma}^{(\mathbf{H})} \right]_{k \times l}, \end{aligned} \quad (6.17)$$

and matrices \mathbf{C}_w and \mathbf{D}_w can be solved via eqn. (2.22) from G_{*w} ,

$$\begin{aligned} \Lambda_{\mathbf{W}} G_{*w} = & \langle \mathbf{W}^T, \nu \Delta \mathbf{W} - \mathfrak{D}_{(\bar{u}, \bar{v})} \mathbf{W} - \mathfrak{D}_{(\mathbf{U}, \mathbf{V})} \bar{w} \rangle \\ & + \left\langle \mathbf{W}^T, \mu g \frac{\partial \boldsymbol{\theta}}{\partial x} \right\rangle \mathbf{A}^T \mathbf{B} - \left[\mathfrak{Z}_{ijk}^{(\mathbf{W})} B_{\alpha i} B_{\beta j} B_{\gamma l} \mathfrak{Z}_{\alpha\beta\gamma}^{(\mathbf{H})} \right]_{k \times l} + \langle \mathbf{W}^T, \mathbf{F} \rangle \mathbf{B}. \end{aligned} \quad (6.18)$$

See Appendix E for details.

6.3 Numerical Results

The stochastic force (4.16) in Sec. 4.2.2 is used in the following numerical example. Both the gPC system and the DyBO-gPC system are numerically integrated by fourth-order RK method with time step $\delta t = 10^{-3}$. Unlike the stochastic flow only driven by the stochastic force in Sec. 4.2.2, we found by numerical experiments a higher-resolution spatial grid is required to resolve some fine structures. Thus, we use 256×256 spatial grid in the numerical simulations. Computations on higher-resolution grid, 512×512 , are also performed for the gPC method to verify numerical convergence. The pseudo-spectral method with the 36th-order Fourier smoothing [54] is used to compute spatial derivatives. For the DyBO method, the gPC solution at $\Delta T_0 = 0.2$ are used as initial conditions. Different values of ΔT_0 , such as 0.1 and 0.15, have also been used and no significant differences have been found. Here, we use the sparse truncation technique and choose

the multi-index set

$$\mathfrak{J} = \{ \boldsymbol{\alpha} \in \mathfrak{J}_8^3 \text{ and if } |\boldsymbol{\alpha}| = 3, \text{ then } \alpha_2 \leq 2, \alpha_3 \leq 1, \alpha_4 \leq 1, \alpha_6 \leq 2, \alpha_7 \leq 1, \alpha_8 \leq 1 \} \setminus \{ \mathbf{0} \}.$$

Both initial vorticity and temperature fields are assumed to be deterministic. Unlike the numerical examples considered in Chapter 4, we are primarily interested in the combined effect of stochastic force f and the stochastic buoyancy force, so the vorticity is assumed to be zero initially. We adopt the initial temperature field from [21],

$$\theta(x, y, 0, \boldsymbol{\xi}) = \hat{\theta}(x, y) = \frac{1}{2} \mathbb{H}_{\delta_1}(y_{lb}(x) - y) + \frac{1}{2} \mathbb{H}_{\delta_1}(y_{ub}(x) - y), \quad (6.19)$$

where

$$\begin{aligned} y_{lb}(x) &= \frac{1}{2} - \delta_2 - \delta_3 y_0(x), \\ y_{ub}(x) &= \frac{1}{2} + \delta_2 + \delta_3 y_0(x), \\ y_0(x) &= 1 + \sin\left(2\pi\left(x + \frac{3}{4}\right)\right), \end{aligned}$$

and the mollified Heaviside step function,

$$\mathbb{H}_\epsilon(z) = \frac{x + \epsilon}{2\epsilon} + \frac{1}{2\pi} \sin\left(\frac{\pi x}{\epsilon}\right).$$

In Fig. 6.1b, the initial temperature field $\hat{\theta}$ is plotted.

In the first numerical example, we choose mode number vector $\mathbf{m} = (07, 08)$. In Fig. 6.2, STD fields of vorticity and temperature are plotted at time $t = 1.0$ with the results by DyBO in the left column and ones by gPC in the right column. In Fig. 6.3 and Fig. 6.4, we also compare

the spatial modes of vorticity and temperature, \mathbf{W} and $\boldsymbol{\theta}$, given by DyBO and gPC at time $t = 1.0$, respectively. All of these three figures confirm that the solutions given by DyBO are not only a good approximation to the solution given by gPC, but also track directly the KL expansion of the SPDE solution.

To further study the numerical convergence of our DyBO method, we choose another two mode number vectors, $\mathbf{m} = (03, 04)$ and $\mathbf{m} = (09, 10)$, and repeat the DyBO computation. The relative errors of vorticity and temperature STD fields as functions of time are plotted in Fig. 6.5a and Fig. 6.5b, respectively. When the mode number vector is increased from $\mathbf{m} = (03, 04)$ to $\mathbf{m} = (09, 10)$, the relative error of STD is brought down from 11.9% to 1.8% for vorticity and from 10.7% to 1.8% for temperature, respectively.

We also enact the adaptive strategy for adding and removing mode pairs described in Sec. 2.6. Initially, the mode number vector is chosen as $\mathbf{m}_0 = (05, 06)$. As we can see in the top two plots in Fig. 6.5, good accuracy is preserved as mode pairs are automatically added when necessary (see Fig. 6.5c for the evolution of numbers of vorticity and temperature mode pairs).

6.4 Parallelization

With the introduction of multiple physical components, the storage complexity analysis in Sec. 4.1.1 is modified to $O((mN_h^d + mN_P)N_s) \approx O(mN_sN_h^d)$, where we have assumed that m terms are retained in the KL expansion of each physical component, i.e., $m_l = m$ for $l = 1, 2, \dots, N_s$. Clearly, even for moderate values of m and N_s and spatial grids, the size of memory required in computations may exceed the in-core memory available to a single processor. Furthermore, the wall time of integrating numerically the DyBO system may not give a reasonable time span for practical applications. The situation gets much worse for spatially three-dimensional SPDEs, which makes distributive parallel computations paramount and indispensable. Based on the computational

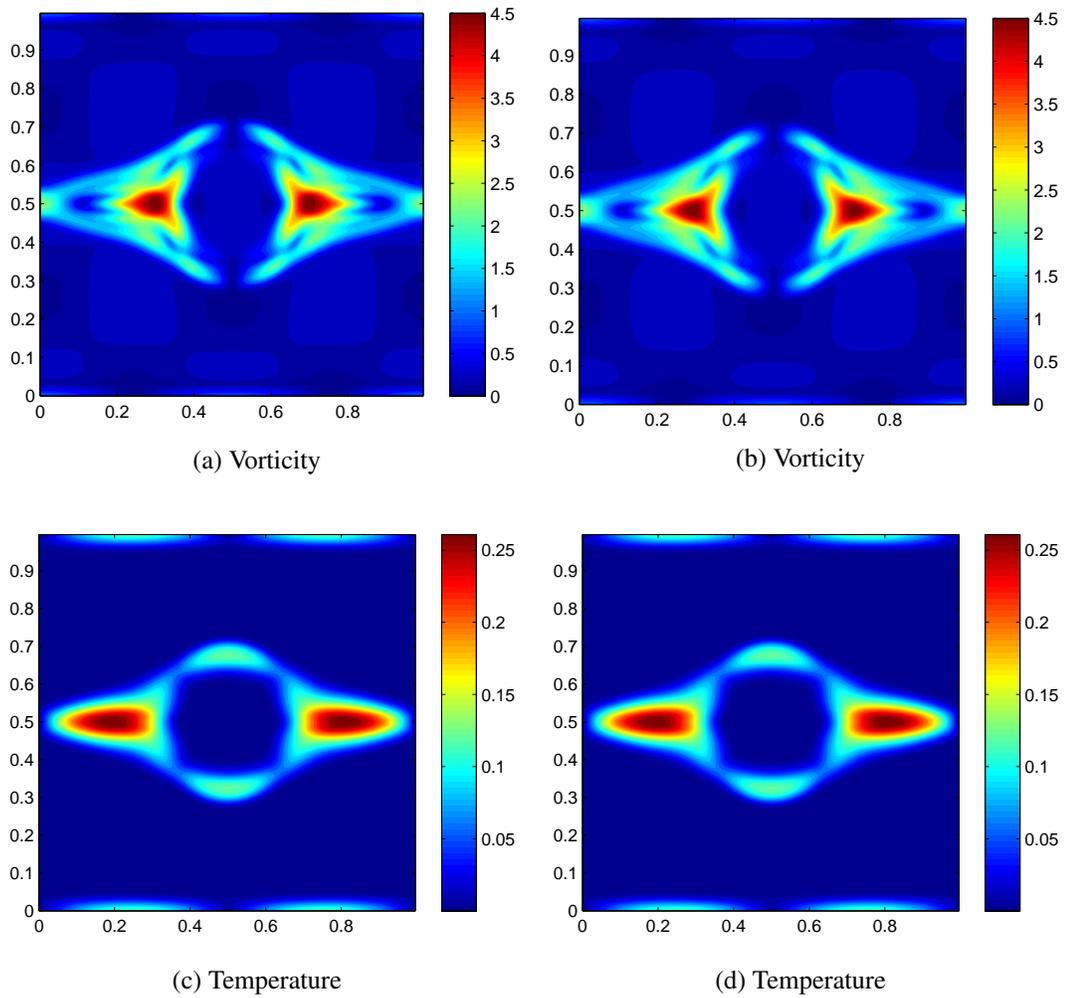


Figure 6.2: STD of vorticity and temperature fields at time $t = 1.0$. Left column by DyBO and right column by gPC

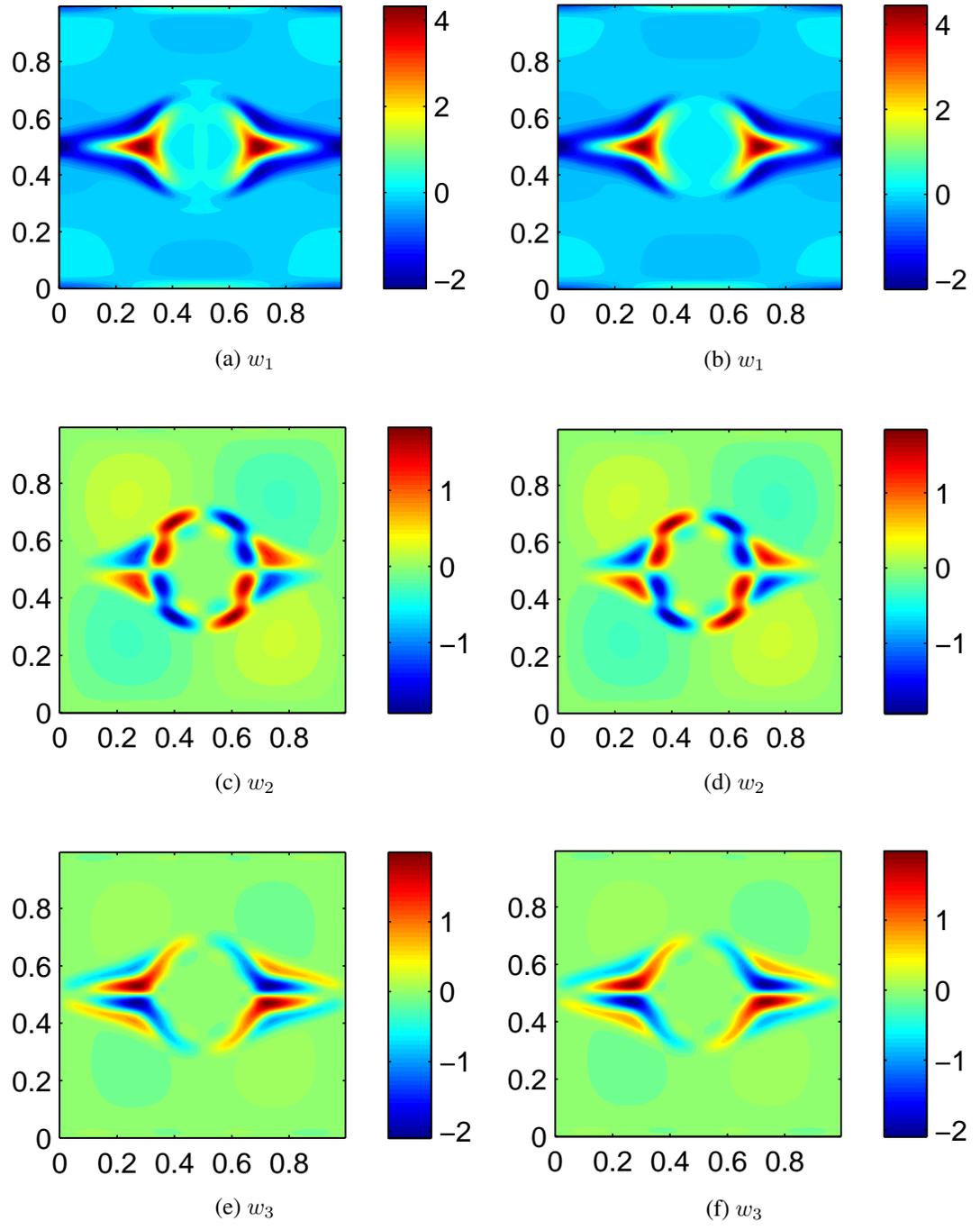


Figure 6.3: Vorticity spatial modes at time $t = 1.0$. Left column by DyBO and right column by gPC

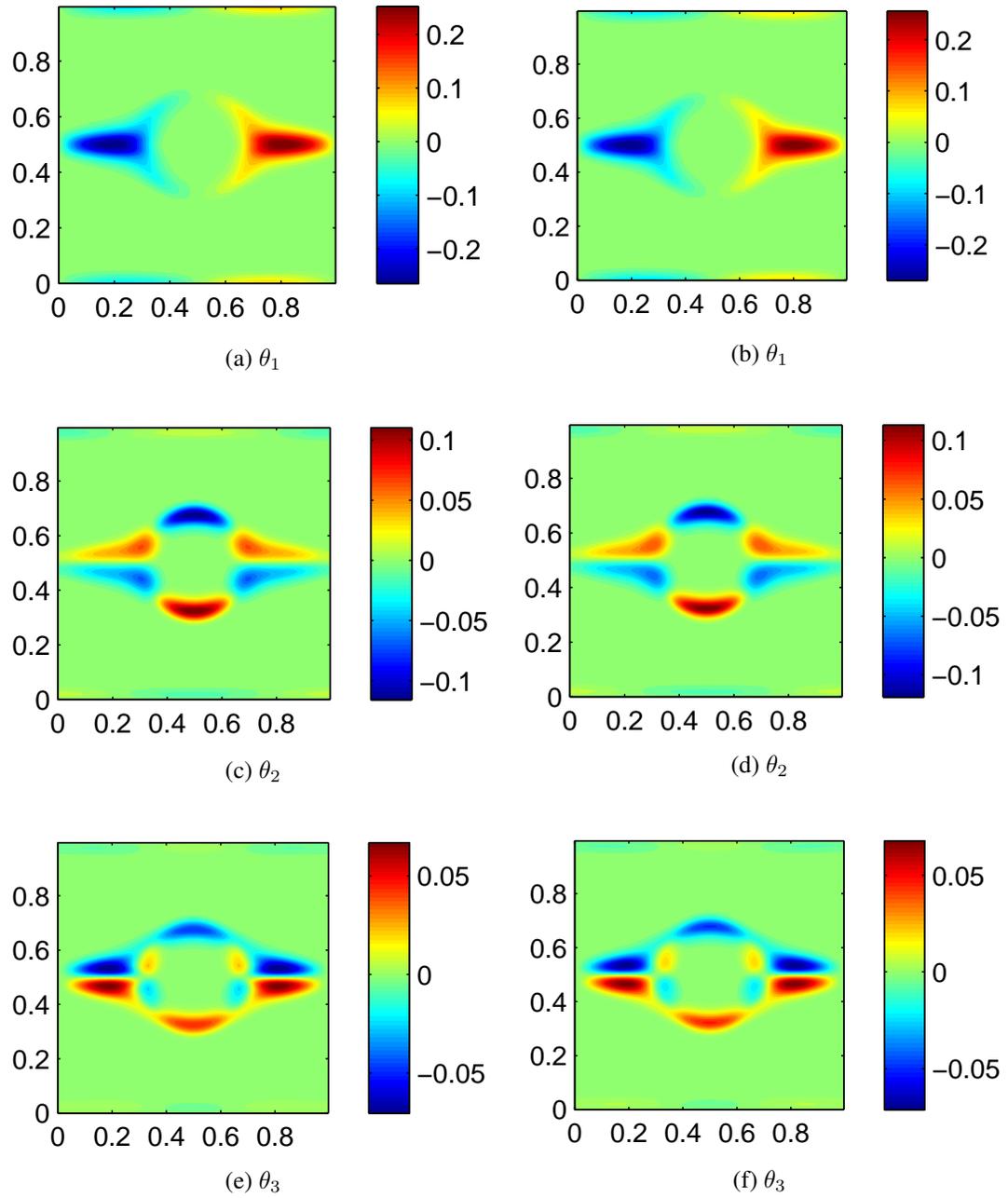


Figure 6.4: Temperature spatial modes at time $t = 1.0$. Left column by DyBO and right column by gPC

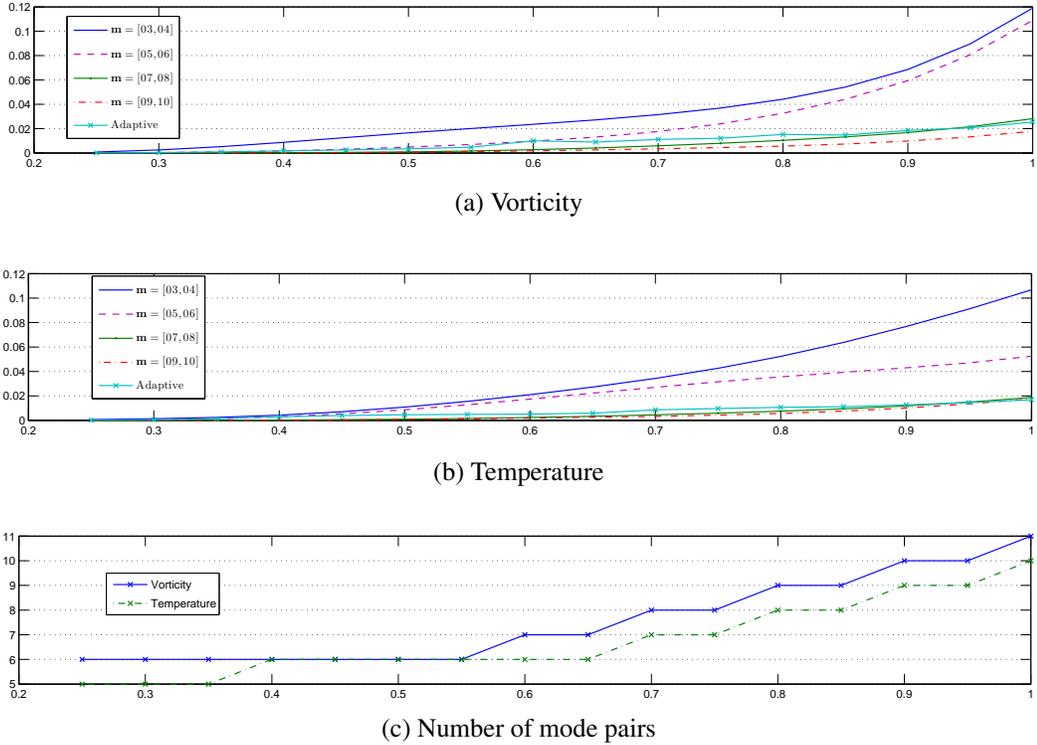


Figure 6.5: \mathbb{L}^2 relative errors of vorticity and temperature STDs as functions of time. The evolutions of the numbers of vorticity and temperature mode pairs are also given in the last figure.

complexity analysis in Sec. 4.1.2.2, we propose a simple, yet powerful, parallelization strategy based on domain decomposition for stochastic Navier-Stokes equations.

6.4.1 Parallelization Strategy

From the computational complexity analysis in Sec. 4.1.2.2, the computation costs of the following four terms dominate others,

$$\left[\mathfrak{D}_{(u_i, v_i)} \theta_j B_{\alpha i} A_{\beta j} A_{\gamma k} \mathfrak{I}_{\alpha \beta \gamma}^{(\mathbf{H})} \right]_{1 \times k}, \left[\mathfrak{D}_{(u_i, v_i)} w_j B_{\alpha i} B_{\beta j} B_{\gamma k} \mathfrak{I}_{\alpha \beta \gamma}^{(\mathbf{H})} \right]_{1 \times k}, \mathfrak{I}^{(\mathbf{W})}, \mathfrak{I}^{(\theta)},$$

and bear prohibitive costs of $O(m^3 N_h^d)$. Without resorting to other fancy parallelization techniques, the definitions of these terms actually suggest a simple strategy based on the domain decomposition method. In what follows, we explain this in detail for the computation of the third-order tensor

$\mathfrak{T}(\mathbf{W})$ while the same strategy applies to other three terms similarly.

Suppose the whole spatial domain \mathcal{D} is partitioned to Q disjoint subdomains \mathcal{D}_q 's, i.e., $\cup_{q=1}^Q \mathcal{D}_q = \mathcal{D}$ and $\mathcal{D}_{q_1} \cap \mathcal{D}_{q_2} = \emptyset$ for $q_1 \neq q_2$. From the definition of $\mathfrak{T}(\mathbf{W})$, each entry

$$\begin{aligned} \mathfrak{T}_{ijk}(\mathbf{W}) &= \int_{\mathcal{D}} w_k \left(u_i \frac{\partial w_j}{\partial x} + v_i \frac{\partial w_j}{\partial y} \right) dx dy \\ &= \sum_{q=1}^Q \int_{\mathcal{D}_q} w_k \left(u_i \frac{\partial w_j}{\partial x} + v_i \frac{\partial w_j}{\partial y} \right) dx dy = \sum_{q=1}^Q \mathfrak{T}_{ijk}(\mathbf{W}, q), \end{aligned} \quad (6.20)$$

where $\mathfrak{T}(\mathbf{W}, q)$ is the portion of $\mathfrak{T}(\mathbf{W})$ on the q th subdomain.

Assume Q processors or computational nodes are available and the q th processor is assigned to the subdomain \mathcal{D}_q . On the q th processor, only the solutions constrained to the subdomain \mathcal{D}_q , i.e., $\bar{\theta}|_{\mathcal{D}_i}$, $\theta|_{\mathcal{D}_q}$, $\bar{w}|_{\mathcal{D}_q}$, and $\mathbf{W}|_{\mathcal{D}_q}$ are stored in in-core memory. Thus, each process can compute its own portion of the third-order tensor $\mathfrak{T}(\mathbf{W}, q)$ via eqn. (6.20). The result on each subdomain will be combined at the end to get $\mathfrak{T}(\mathbf{W})$. The partition of domain may be problem-dependent. In Fig. 6.6, we illustrate a partition evenly along x -axis.

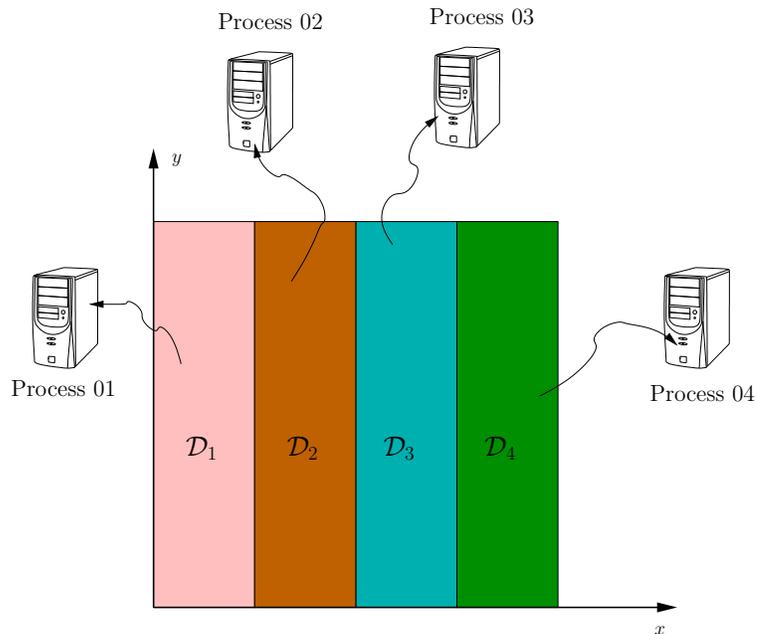


Figure 6.6: Illustration of spatial domain decomposition

6.4.2 Implementation and Speedup

The above parallelization strategy is implemented via POSIX multi-threaded programming in C++ and hooked to the main Matlab code via Matlab external APIs. When the computations of the above four terms are needed, the C++ codes are invoked by the main Matlab routine with necessary input arguments passed. The C++ codes in turn spawn Q threads corresponding to Q subdomains and collect the results from threads when computations on all threads finish. After that, the C++ codes return the results to the main Matlab routine and computations continue in Matlab.

Since we use FFT in the pseudo-spectral method, the domain partition scheme illustrated in Fig. 6.6 is adopted for the maximum performance of FFT. The simulations are conducted on the Shared Heterogeneous Cluster (SHC) at Caltech Center for Advanced Computing Research (CACR). Due to the limitations posted by available campus Matlab licenses, our simulation is constrained onto a single computing node where total 8 computing cores are available from two AMD Opteron 2390 of 2.5 GHz.

For two spatial grids, 256×256 and 512×512 , the wall times of a single RK iteration steps are recorded and reported in Table 6.1 for serial computation and parallel computation with 2, 4, 6 and 8 cores, respectively. Speedups of parallel strategies are also reported in the same table. Two mode number vectors $\mathbf{m} = (07, 08)$ and $\mathbf{m} = (19, 20)$ are used in the computation. Confirming our complexity analysis, the speedup is more significant for larger spatial grid and more mode pairs. For example, the computational time on 8 cores is reduced to 1/6 on 256×256 spatial grid compared to that on a single core.

Remark 6.2. In our implementation, the domain is only *virtually*, or conceptually, partitioned and each thread can still access the solution information on other subdomains. However, this can only be regarded as a special case from multi-threaded implementation. The proposed parallelization strat-

# proc	256×256				512×512			
	$\mathbf{m} = [7, 8]$		$\mathbf{m} = [19, 20]$		$\mathbf{m} = [7, 8]$		$\mathbf{m} = [19, 20]$	
	Time	Speedup	Time	Speedup	Time	Speedup	Time	Speedup
Serial	1.6930	NA	19.529	NA	10.658	NA	71.372	NA
2	1.2618	1.34	11.150	1.75	6.8218	1.56	34.471	2.07
4	1.0216	1.66	6.9873	2.79	4.4738	2.38	20.600	3.46
6	0.9787	1.73	4.4082	4.43	3.7902	2.81	16.940	4.21
8	0.9197	1.84	3.1262	6.25	3.5095	3.03	15.182	4.70

Table 6.1: Speedups by proposed parallelization strategy for different spatial grids and mode number vectors. Wall times of a single RK time step for different parameters are given in the second, fourth, sixth, and eighth columns. All times are in seconds.

egy applies to the true distributive computing environment and can use popular Message Passage Interface (MPI) protocol to collect results from computing nodes.

Chapter 7

Generalized Stochastic Collocation Formulation of DyBO Method (DyBO-gSC)

In this chapter, we explore another possibility proposed in Sec. 2.3.3 to numerically represent stochastic modes \mathbf{Y} , i.e., generalized Stochastic Collocation (gSC), which results in the gSC version of DyBO (DyBO-gSC).

7.1 Formulation and Algorithm

In previous chapters, we have discussed in length the DyBO-gPC method, where the stochastic process $Y_i(\omega, t)$ is projected onto a set of gPC basis functions \mathbf{H} and we only need to deal in numerical implementations with the gPC coefficients, $A_{\alpha i}$'s, $i = 1, 2, \dots, m$, $\alpha \in \mathfrak{J}$. That is

$$\mathbf{Y}(\boldsymbol{\xi}, t) = \mathbf{H}\mathbf{A}(t),$$

where the gPC coefficient is given by

$$A_{\alpha i}(t) = \mathbb{E}[Y_i(\boldsymbol{\xi}, t)\mathbf{H}_{\alpha}(\boldsymbol{\xi})].$$

Without loss of generality, we have been assuming that independent random variables ξ_l 's have identical distribution $\rho(z) : \mathbb{R} \rightarrow [0, 1]$. Therefore, the above computations of gPC coefficients are essentially multivariate integrals with a weight induced by the distribution $\rho(z)$, i.e.,

$$A_{\alpha i}(t) = \int_{\mathbb{R}^{N_p}} Y_i(\mathbf{z}, t) \mathbf{H}_{\alpha}(\mathbf{z}) \prod_{l=1}^{N_p} \rho(z_l) d\mathbf{z},$$

where $\mathbf{z} = (z_1, z_2, \dots, z_{N_p})^T \in \mathbb{R}^{N_p}$. Numerically, the above integral can be approximated with high accuracy by quadrature rules. A quadrature rule \mathfrak{K} for integrals in \mathbb{R}^{N_p} with weight function $\prod_{l=1}^{N_p} \rho(z_l)$ is just a combination of a cloud of points \mathbf{z}_r 's in \mathbb{R}^{N_p} and associated weights w_r 's, which we write $\mathfrak{K} = \{(\mathbf{z}_r, w_r) \mid \mathbf{z}_r \in \mathbb{R}^{N_p}, w_r \in \mathbb{R}, r = 1, 2, \dots, |\mathfrak{K}|\}$. Under this rule, the gPC coefficient

$$A_{\alpha i}(t) \approx \mathfrak{K} [Y_i \mathbf{H}_{\alpha}] = \sum_{r=1}^{|\mathfrak{K}|} w_r Y_i(\mathbf{z}_r, t) \mathbf{H}_{\alpha}(\mathbf{z}_r).$$

Therefore, instead of representing the stochastic modes \mathbf{Y} in DyBO-gPC method by their gPC coefficients \mathbf{A} , alternatively, we can also use their values at quadrature points \mathbf{z}_r 's from quadrature rule \mathfrak{K} , i.e., $\{\mathbf{Y}(\mathbf{z}_r, t)\}_{r=1}^{|\mathfrak{K}|}$. Before presenting our DyBO method in this form, we first elaborate a little on the construction of quadrature rules for multivariate integrals. For detailed discussions on sparse grid and multivariate integral, see [78, 87, 50, 33, 116, 88, 46, 40, 18].

7.1.1 Sparse Grid

Naive construction of quadrature rules for multivariate integrals will soon suffer from the curse of dimensionality as the number of random variables, N_p , increases. For example, consider the quadrature rule \mathfrak{G}_{N_p} constructed from the tensor product of one-dimensional quadrature rule \mathfrak{G}_1^q , which produces the integral exact value if the integrand is a polynomial of order at most $2q - 1$. The

total number of quadrature points in this tensor grid is

$$|\mathfrak{G}_{N_p}| = |\mathfrak{G}_1^q|^{N_p},$$

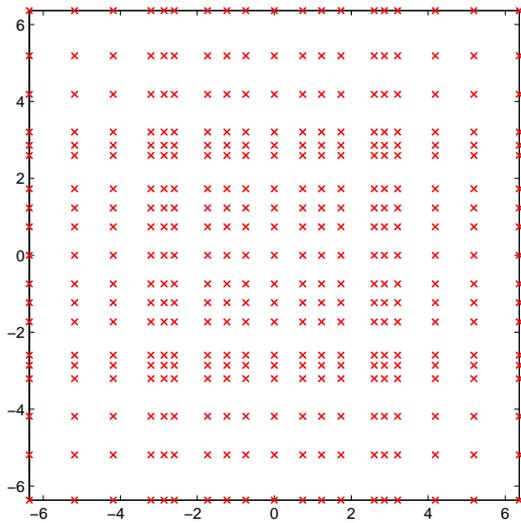
i.e., it explodes exponentially as the number of random variable, N_p , increases and incurs prohibitive computational costs.

Instead of direct tensor product of one-dimensional quadrature rule \mathfrak{G}_1^q , Smolyak algorithm [101] uses a linear combination of tensor products of one-dimensional quadrature rules with lower-order exactness, i.e., \mathfrak{G}_1^p , $p \leq q$, to construct the multivariate quadrature rule \mathfrak{R}_{N_p} . Since such quadrature rule has a significantly smaller number of quadrature points compared to \mathfrak{G}_{N_p} , it is called *sparse grid*.

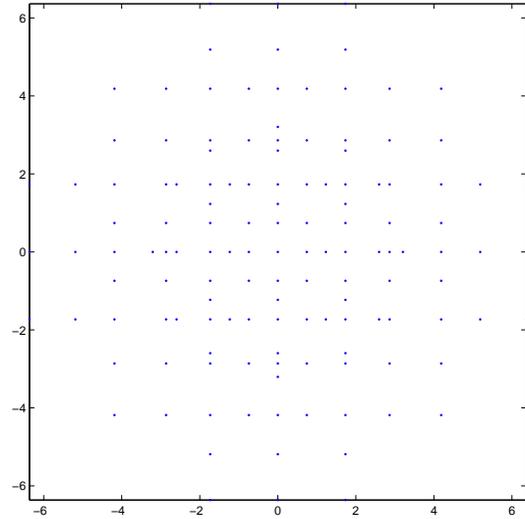
When the distribution $\rho(x)$ is symmetric, e.g., standard Gaussian distribution $\rho(x) = \frac{1}{\sqrt{2\pi}}e^{-x^2/2}$ considered in this thesis, such symmetry can be further explored to reduce the number of quadrature points [39, 88, 50]. In the numerical results presented in Sec. 7.2, we have used the sparse grid $\mathfrak{R}_{N_p}^q$ accompanying Heiss and Winschel's paper [50] and published at <http://www.sparse-grids.de/>. Parameter q indicates the accuracy level of the sparse grid, i.e., the corresponding quadrature rule integrates *exactly* polynomials of total order no more than $2q - 1$. In Fig. 7.1, we compare tensor grids $\mathfrak{G}_{N_p}^{10}$ and sparse grids $\mathfrak{R}_{N_p}^{10}$ for $N_p = 2, 3$, respectively. Even for $N_p = 3$, we see that sparse grid has significantly less number of quadrature points (703 vs. 6859).

7.1.2 Generalized Stochastic Collocation Method (gSC)

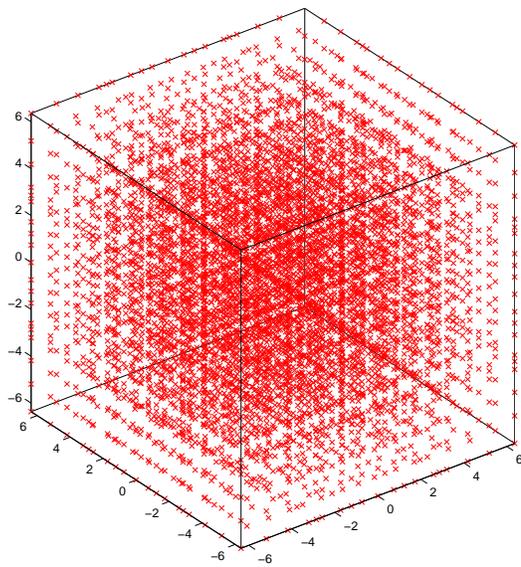
With the introduction of sparse grid, the application of gSC method to SPDE is very similar to that of MC method. Specifically, for each quadrature point \mathbf{z}_r in the sparse grid \mathfrak{R} , we solve numerically



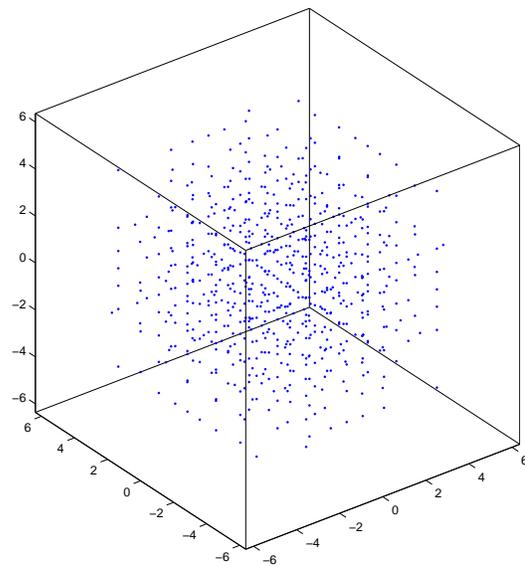
(a) 2d Tensor Grid. Total 361 nodes



(b) 2d Sparse Grid. Total 133 nodes



(c) 3d Tensor Grid. Total 6859



(d) 3d Sparse Grid. Total 703

Figure 7.1: Sparse grid vs. tensor grid

SPDE (1.1), i.e.,

$$\frac{\partial u}{\partial t}(x, t, \mathbf{z}_r) = \mathcal{L}u(x, t, \mathbf{z}_r), \quad x \in \mathcal{D} \in \mathbb{R}^d, t \in [0, T], \quad (7.1a)$$

$$u(x, 0, \mathbf{z}_r) = u_0(x, \mathbf{z}_r), \quad (7.1b)$$

$$\mathcal{B}(u(x, t, \mathbf{z}_r), \frac{\partial u}{\partial x}(x, t, \mathbf{z}_r)) = 0, \quad x \in \partial\mathcal{D}. \quad (7.1c)$$

After an collection of solutions $\{u(x, t, \mathbf{z}_r)\}_{r=1}^{|\mathfrak{R}|}$ has been obtained, various statistics of SPDE solution can be approximated by applying the quadrature rule \mathfrak{R} . For example, the mean and the second moment of the stochastic solution can be computed as

$$\begin{aligned} \mathbb{E}[u(x, t, \cdot)] &\approx \sum_{r=1}^{|\mathfrak{R}|} u(x, t, \mathbf{z}_r) w_r, \\ \mathbb{E}[u(x, t, \cdot)^2] &\approx \sum_{r=1}^{|\mathfrak{R}|} u(x, t, \mathbf{z}_r)^2 w_r. \end{aligned}$$

7.1.3 gSC version of DyBO Method (DyBO-gSC)

As proposed earlier, we can represent the stochastic modes \mathbf{Y} by their values on the sparse grid \mathfrak{R} and replace the expectations in the DyBO formulation (2.28) by the corresponding quadrature rule, which gives the gSC version of our DyBO method (DyBO-gSC),

$$\frac{\partial \bar{u}}{\partial t}(x, t) = \sum_{r=1}^{|\mathfrak{R}|} w_i \mathcal{L}u(x, t, \mathbf{z}_r), \quad (7.2a)$$

$$\frac{\partial \mathbf{U}}{\partial t}(x, t) = -\mathbf{U}(x, t) \mathbf{D}(t)^T + \sum_{r=1}^{|\mathfrak{R}|} w_r \tilde{\mathcal{L}}u(x, t, \mathbf{z}_r) \mathbf{Y}(\mathbf{z}_r, t), \quad (7.2b)$$

$$\begin{aligned} \frac{d\mathbf{Y}}{dt}(\mathbf{z}_r, t) &= -\mathbf{Y}(\mathbf{z}_r, t) \mathbf{C}(t)^T + \left\langle \tilde{\mathcal{L}}u(x, t, \mathbf{z}_r), \mathbf{U}(x, t) \right\rangle \mathbf{\Lambda}_{\mathbf{U}}(t)^{-1}, \\ &\text{for } r = 1, 2, \dots, |\mathfrak{R}|, \end{aligned} \quad (7.2c)$$

where $\mathbf{C}(t)$ and $\mathbf{D}(t)$ can be solved from (2.22) with

$$G_*(u, \mathbf{U}, \mathbf{Y}) = \mathbf{\Lambda}_{\mathbf{U}}^{-1} \left\langle \mathbf{U}^T, \sum_{r=1}^{|\mathfrak{R}|} w_r \tilde{\mathcal{L}}u(x, t, \mathbf{z}_r) \mathbf{Y}(\mathbf{z}_r, t) \right\rangle. \quad (7.3)$$

The DyBO-gSC formulation of SPDE is again a combination of time-dependent deterministic PDEs (7.2a) and (7.2b), and ODE system (7.2c). By using spatial and temporal discretization schemes of our choices, the DyBO-gSC system (7.2) can be numerically integrated and various statistics of the stochastic solution can be computed by applying the quadrature rule \mathfrak{R} . Discussions regarding numerical implementations of our DyBO-gPC method in Chapter 2 apply similarly to the DyBO-gSC method.

Comparing the DyBO-gSC method to the gSC method in Sec. 7.1.2, we identify the roots of potential speedups. In the gSC method, we have to update in each time iteration $|\mathfrak{R}|$ functions of spatial variable x , i.e., $\{u(x, t, \mathbf{z}_r)\}_{r=1}^{|\mathfrak{R}|}$, although we may not need to update them simultaneously. On the other hand, the DyBO-gSC method explores the inherent low-dimensional structure of the stochastic solution and only updates in each time iteration m spatial modes $\mathbf{U}(x, t)$ with $m \ll |\mathfrak{R}|$ plus $|\mathfrak{R}|$ row vectors $\{\mathbf{Y}(\mathbf{z}_l, t)\}_{l=1}^{|\mathfrak{R}|} \subset \mathbb{R}^{1 \times m}$. Thus, the speedups become more significant when larger spatial grids and sparse grids are used, which will be demonstrated numerically in the next section.

Comparing the DyBO-gSC formulation to the DyBO-gPC formulation in Sec. 2.3.2.2, we find the major difference between these two versions of DyBO methods are with the evolution equations of the stochastic modes. The ODEs of stochastic modes \mathbf{Y} in the DyBO-gSC formulation inherit certain properties of the gSC method and are *partially decoupled* in the following sense. When a numerical integrator is applied, the values of stochastic modes \mathbf{Y} at different quadrature points \mathbf{z}_l 's can be updated independently and/or simultaneously in each time iteration. That's to say, although

DyBO-gSC formulation does not preserve the non-intrusiveness of the gSC method, it inherits the easy parallelism of the gSC method.

7.2 Numerical Example — Stochastic Burgers Equation

In this section, we re-visit the numerical example of Burgers equation driven by stochastic force in Sec. 3.3, i.e., eqn. (3.10), and apply the gSC method and the DyBO-gSC method. From eqn. (3.12), we obtain the DyBO-gSC formulation of stochastic Burgers equation,

$$\frac{\partial \bar{u}}{\partial t} = \mathring{\mathcal{L}}\bar{u} - \mathbf{U} \frac{\partial \mathbf{U}^T}{\partial x}, \quad (7.4a)$$

$$\frac{\partial \mathbf{U}}{\partial t} = -\mathbf{U}\mathbf{D}^T + \left(\nu \frac{\partial^2 \mathbf{U}}{\partial x^2} - \frac{\partial (\bar{u}\mathbf{U})}{\partial x} \right) - \left(u_i \frac{\partial u_j}{\partial x} \mathfrak{F}_{ijk}^{(\mathbf{Y})} \right)_{1k} + \mathbf{F} \sum_{r=1}^{|\mathfrak{K}|} w_r \mathbf{H}^T(\mathbf{z}_r) \mathbf{Y}(\mathbf{z}_r, t), \quad (7.4b)$$

$$\begin{aligned} \frac{d\mathbf{Y}}{dt}(\mathbf{z}_r, t) = & \mathbf{Y}(\mathbf{z}_r, t) \left(-\mathbf{C}^T + \left\langle \nu \frac{\partial^2 \mathbf{U}^T}{\partial x^2} - \frac{\partial (\bar{u}\mathbf{U}^T)}{\partial x}, \mathbf{U} \right\rangle \mathbf{\Lambda}_{\mathbf{U}}^{-1} \right) \\ & + \left(\mathfrak{F}_{iik}^{(\mathbf{U})} \right)_{1k} \mathbf{\Lambda}_{\mathbf{U}}^{-1} - \left(Y_i(\mathbf{z}_r, t) Y_j(\mathbf{z}_r, t) \mathfrak{F}_{ijk}^{(\mathbf{U})} \right)_{1k} \mathbf{\Lambda}_{\mathbf{U}}^{-1} \\ & + \mathbf{H}(\mathbf{z}_r) \langle \mathbf{F}^T, \mathbf{U} \rangle \mathbf{\Lambda}_{\mathbf{U}}^{-1}, \quad \text{for } r = 1, 2, \dots, |\mathfrak{K}|, \quad (7.4c) \end{aligned}$$

where the third-order tensor

$$\mathfrak{F}_{ijk}^{(\mathbf{Y})}(t) = \sum_{r=1}^{|\mathfrak{K}|} w_r Y_i(\mathbf{z}_r, t) Y_j(\mathbf{z}_r, t) Y_k(\mathbf{z}_r, t), \quad \text{for } i, j, k = 1, 2, \dots, m,$$

and matrices $\mathbf{C}(t)$ and $\mathbf{D}(t)$ can be solved from (2.22) with

$$\mathbf{\Lambda}_{\mathbf{U}} G_*(u, \mathbf{U}, \mathbf{Y}) = \left\langle \mathbf{U}^T, \nu \frac{\partial^2 \mathbf{U}}{\partial x^2} - \frac{\partial (\bar{u}\mathbf{U})}{\partial x} \right\rangle - \left(\mathfrak{F}_{ijl}^{(\mathbf{U})} \mathfrak{F}_{ijk}^{(\mathbf{Y})} \right)_{lk} + \langle \mathbf{U}^T, \mathbf{F} \rangle \sum_{r=1}^{|\mathfrak{K}|} w_r \mathbf{H}^T(\mathbf{z}_r) \mathbf{Y}(\mathbf{z}_r, t).$$

In the following numerical experiment, we have adopted $\nu = 5.0 \times 10^{-3}$ and the same stochastic force defined in eqn. (3.16). By using the pseudo-spectral method ($\delta x = 1/256$) and the fourth-order Runge-Kutta integrator ($\delta t = 5 \times 10^{-4}$), the above system is integrated to time $t = T = 1.0$, where the initial conditions of the DyBO-gSC method are provided by the gSC method at time $t = \Delta T_0 = 0.15$. In Sec. 3.3.3, we have chosen a set of Hermit polynomials of total order at most 3. In the DyBO-gSC formulation, a triple product $\mathbb{E}[Y_i Y_j Y_k]$ is involved. To compare fairly the performances of the gPC and gSC versions of DyBO methods for the same problem, we have chosen the sparse grid \mathfrak{R}_6^6 which computes exactly the integral of polynomials of total order at most $2q - 1 = 11 > 9 = 3 + 3 + 3$. We also apply the gSC method on a much larger sparse grid \mathfrak{R}_6^{12} with 98837 quadrature points and consider the solution to be exact.

In Table 7.1, we report the relative errors of the mean and STD given by our DyBO-gSC method, with $m = 4, 8, 12, 16, 20$, with respect to the gSC solution on \mathfrak{R}_6^6 and the exact solution obtained on \mathfrak{R}_6^{12} , respectively. In the last columns, we also report the wall times for different methods. When $m = 20$, the DyBO-gSC method achieves a solution of comparable accuracy with respect to the exact solution as the gSC solution. See the third and the last rows in the fourth and the fifth columns. Therefore, our DyBO-gSC method achieves $\sim 2.45X$ speedup compared to gSC method.

Methods	gSC- \mathfrak{R}_6^6		gSC- \mathfrak{R}_6^{12}		Time (min)
	Mean (%)	STD (%)	Mean (%)	STD (%)	
gSC	NA	NA	0.5037	1.0600	15.2
DyBO-gSC m=4	2.9181	5.3131	2.9951	5.2513	1.0
DyBO-gSC m=8	2.3662	3.9506	2.4804	3.9780	1.7
DyBO-gSC m=12	0.8225	1.7178	0.9640	1.9889	2.9
DyBO-gSC m=16	0.5481	0.8811	0.7009	1.3024	4.2
DyBO-gSC m=20	0.3900	0.5723	0.6511	1.1613	6.2

Table 7.1: Relative errors of statistical quantities computed by DyBO-gSC and gSC at time $t = 1.0$

Remark 7.1. We also repeat the same numerical experiment on a smaller spatial grid of size $\delta x = 1/128$. Since the number of spatial nodes is far less than that of the sparse grid \mathfrak{R}_6^6 , the speedup

is nearly marginal, ~ 1.53 . We expect a much larger speedup for spatially two/three-dimensional SPDEs, which will be considered in future work.

Remark 7.2. Comparing Table 7.1 and Table 3.1, we see that DyBO-gSC achieves less speedup than DyBO-gPC. Since only limited numerical experiments are performed for the comparison between gPC and gSC versions of DyBO methods at this moment, we refrain from drawing a general conclusion here.

Chapter 8

Conclusions and Future Work

In this thesis we have proposed DyBO formulations and their numerical implementations for a class of time-dependent SPDEs or a system of time-dependent SPDEs, whose solutions enjoy low-dimensional structures in the sense of KL expansions. Unlike other traditional methods, such as MC, qMC, gPC, and gSC, our DyBO methods explore such inherent low-dimensional structures and give *directly* the stochastic solutions in bi-orthogonal forms, and essentially track the KL expansion. Thus, without additional post-processing steps, the DyBO methods reveal directly the most important and intrinsic low-dimensional structures/dynamics of related physical processes. We have proved rigorously the preservation of bi-orthogonality in DyBO methods and verify it numerically in several challenging problems, such as stochastic Burgers equations and stochastic flows in 2D unit square. Depending on the numerical representations of the stochastic modes, three versions of DyBO methods have been proposed in this thesis, i.e., DyBO-MC, DyBO-gSC, DyBO-gPC.

Another important benefit associated with the preservation of bi-orthogonality and the exploration of the low-dimensional structures in our DyBO methods is the significant savings of computational costs both in memory consumptions and computational times. Detailed complexity analysis has been conducted for DyBO-gPC methods for two classes of time-dependent SPDEs. In practice, we have observed speedups of up to 200 times compared to gPC methods. From the perspective of RB and POD methods, DyBO methods construct the best set of a spatial basis on the fly without

invoking any expensive offline computations in RB or pre-computation stages. From the perspective of adaptive gPC/gSC methods, our DyBO methods automatically use the linear combinations of polynomial chaos basis as the best set of a stochastic basis on the flight without introducing any heuristics to select multi-indices.

To the best of our knowledge, the DyBO methods presented in this thesis are the first systematic attempts to *directly* target KL expansions and *fully* explore the bi-orthogonality. One may wonder why such methodology has not been introduced earlier. This may be due to several challenges in both theory and numerical implementations. From the theoretical consideration, one of the main difficulties is how to eliminate the extra degrees of freedom induced by allowing both the spatial and the stochastic basis to change in time. From the numerical implementation view point, with the introduction of multiple spatial modes and stochastic modes and spatial and temporal discretization schemes, the final form of discretized system can be very messy, daunting and error-prone in numerical implementations. By extensively adopting vector/matrix/tensor notations, we can not only present the DyBO formations in a very concise form, but also enable codings to proceed in an intuitive way, especially for object-oriented programming languages. Temporarily freezing spatial modes \mathbf{U} or stochastic modes \mathbf{Y} has been proposed to deal with the special moments when eigenvalues cross each other. What's more, sophisticated strategies based on error analysis have been proposed to adaptively and dynamically add and remove spatial and stochastic mode pairs to achieve both accuracy and efficiency. The effectiveness of such strategies have been demonstrated in the various numerical examples in this thesis.

Certainly, other non-trivial issues may arise in applications of the DyBO methods to the industrial-scale problems involving uncertainty. Exhaustive discussions of such issues beyond the scope of this thesis. However, we have paid special attention to one of the important issues, i.e., parallelization of DyBO-gPC method, as the first step to fully unleash the power of our DyBO methods. Based on the

computational complexity analysis, we have proposed a parallelization scheme based on a domain decomposition technique for the stochastic flow in 2D unit square driven by stochastic and buoyancy forces. The effectiveness of such scheme has been verified on the Caltech Shared Heterogeneous Cluster.

This thesis has already covered many important aspects of the DyBO methods from formulations to numerical implementations. However, a number of issues associated with these innovative methods have been left for future work. We compile a list of selected topics in the following.

Intrinsic DyBO Formulations. Re-examining the KL expansion of stochastic solutions reveals that the spatial modes \mathbf{U} can be viewed as *one* kind of parametrization of some m -dimensional function space which is optimal in the sense of m -term truncation errors. Specifically, let's consider a second-order stochastic process of zero mean, $u \in \mathbb{L}^2(\mathcal{D} \times \Omega)$, and its m -term truncated KL expansion, i.e.,

$$u \approx \mathbf{U}\mathbf{Y}^T,$$

where we assume \mathbf{U} are orthonormal, i.e., $\langle \mathbf{U}^T, \mathbf{U} \rangle = \mathbf{I}$. Let $\mathbf{G} \in \mathbb{R}^{m \times m}$ be any normal matrix, i.e., $\mathbf{G}^T \mathbf{G} = \mathbf{G} \mathbf{G}^T = \mathbf{I}$. Consider another decomposition of the stochastic process u ,

$$u \approx \mathbf{V}\mathbf{Z}^T,$$

where $\mathbf{V} = \mathbf{U}\mathbf{G}$, $\mathbf{Z} = \mathbf{Y}\mathbf{G}$. Obviously, this “new” decomposition provides the same level of accuracy as $\mathbf{U}\mathbf{Y}^T$ does although the stochastic modes \mathbf{Z} may not be orthogonal to each other any more.

In other words, any basis set of the m -dimensional function subspace $\text{span}(\mathbf{U}) \subset \mathbb{L}^2(\mathcal{D})$ provides a decomposition of the same accuracy level. Therefore, it is this function subspace that is *intrinsically* associated with the stochastic process u . We write this function space as $\mathcal{V}_m(t) \subset \mathbb{L}^2(\mathcal{D})$, where t emphasizes the dependence on time for time-dependent SPDEs. From this perspective, the

spatial modes $\mathbf{U}(t)$ in the DyBO formulation are just one way to parametrize this intrinsic function space and our DyBO formulation essentially describes the evolution of $\mathcal{V}_m(t)$ in time. The DyBO formulation may also be given in terms of $\mathcal{V}_m(t)$ instead of any particular parameterization, which provides an *intrinsic formulation* of DyBO methods.

From the perspective of the intrinsic formulation discussed above, the DO method by Sapsis and Lermusiaux [97, 96] attempts to provide another parameterization of the function space $\mathcal{V}_m(t)$. At the moment, we have some preliminary understanding regarding the relationship between the DO method and our DyBO method. The intrinsic formulation in terms of the function space $\mathcal{V}(t)$ may provide an unified framework for both methods and better understanding about their connections. However, it is worth noticing that the DO method needs to compute covariance matrix from time to time, which is expensive, while our method does not need to do so.

Multiscale Stochastic Problems. Besides randomness, many physical and engineering problems also involve ubiquitously multiscale phenomena. For example, the underground water/pollution transport takes place simultaneously in many scales. Meanwhile, due to current technical limitations, the properties of the underground medium cannot be known precisely and have to be modeled by some stochastic processes to accommodate unavoidable uncertainty. The interaction of multiscale phenomena and randomnesses further complicates the problems under consideration. Previous attempts [37, 6] only explore the multiscale structures, but ignore the low-dimensional structures. By decomposing the stochastic multiscale solution u first in different spatial scales, i.e.,

$$u = u^{(c)} + u^{(f)},$$

where $u^{(c)}$ and $u^{(f)}$ are solutions on coarse and fine grids, respectively, we may apply the generalized DyBO methods for a system of SPDEs of $u^{(c)}$ and $u^{(f)}$. Such method may potentially explore

both the low-dimensional and the multiscale structures of the stochastic solutions on different spatial scales. Depending on problems under consideration, we may further restrict fine-scale solutions onto coarse-grid cells, i.e.,

$$u^{(f)} = \sum_{i=1}^{N_H} \phi_i \cdot u^{(f)}$$

where N_H is the number of grid points in the coarse grid and $\{\phi_i(x)\}_{i=1}^{N_H}$ are a partition of unity associated to the coarse grid on spatial domain \mathcal{D} , i.e., $\sum_{i=1}^{N_H} \phi_i(x) \equiv 1, \forall x \in \mathcal{D}$. The generalized DyBO methods can be applied to a system of SPDEs of $u^{(c)}$ and $\phi_1 u^f, \phi_2 u^f, \dots, \phi_{N_H} u^f$. The SPDE corresponding to $\phi_i u^f$ can be numerical simulated locally, which may further reduce computational costs.

Short-term Memory. Throughout the thesis, we have assumed the number of random variables involved in the time-dependent SPDEs does not change in time, which may not be true for certain class of SPDEs. For example, such number constantly increases in Burgers Equation driven by Brownian motion if the stochastic process is approximated like eqn. (3.16). Although our DyBO-gPC method already explores the inherent low-dimensional structure, its efficiency deteriorates very quickly due to the fast growth of the number of random variables, N_P . On the other hand, the Burgers equation is dissipative, so the stochastic solution of Burgers equation, as a functional of Brownian motion, does not depend uniformly on the *whole* Brownian trajectory, i.e., the segment of Brownian trajectory near the present time is more influential than the segments at early time. Such *short-term memory* property may be further explored to provide more efficient representations of stochastic solutions for this class of time-dependent SPDEs.

Refinements of Numerical Schemes for DyBO Formulations. The DyBO methods give a peculiar formulation which is a combination of coupled PDEs and ODEs. There are a couple of possible refinements which have not been considered in the numerical implementations in this the-

sis. For example, different time steps may be used for PDEs and ODEs to improve the overall computational efficiency. Furthermore, the lower-order spatial modes in KL expansions are generally smoother than the higher-order spatial modes. These can be explored by discretizing lower-order spatial modes on coarser spatial grids to achieve further computation reductions.

Appendix A

Derivation of the gPC formulation of Time-Evolutionary SPDE

In this appendix, we provide the detailed derivations of gPC formulation of time-evolutionary SPDEs, i.e., (2.36). We start by substituting gPCE expansion (2.35)

$$u(x, t, \omega) = \bar{v}(x, t) + \mathbf{V}(x, t)\mathbf{H}(\boldsymbol{\xi})^T,$$

into SPDE (1.1a) and get

$$\frac{\partial \bar{v}}{\partial t} + \frac{\partial \mathbf{V}}{\partial t} \mathbf{H}^T = \mathcal{L}u. \quad (\text{A.1})$$

Taking expectations on both sides, we get the first equation in (2.36a)

$$\frac{\partial \bar{v}}{\partial t} = \mathbb{E}[\mathcal{L}u],$$

since $\mathbb{E}[\mathbf{H}] = \mathbf{0}$. Substituting the above equality into eqn. (A.1), we have

$$\frac{\partial \mathbf{V}}{\partial t} \mathbf{H}^T = \mathcal{L}u - \mathbb{E}[\mathcal{L}u] = \tilde{\mathcal{L}}u, \quad (\text{A.2})$$

Multiplying both sides of the above equation by the row vector \mathbf{H} from the right and taking expectations, we have

$$\frac{\partial \mathbf{V}}{\partial t} = \mathbb{E} \left[\tilde{\mathcal{L}}u\mathbf{H} \right],$$

where we have used the orthogonality of the polynomial chaos basis \mathbf{H} , i.e., $\mathbb{E} [\mathbf{H}^T \mathbf{H}] = \mathbf{I}$. This completes the derivation of (2.36).

Appendix B

Proof of Corollary 2.8 and 2.9

In this appendix, we prove Corollary 2.8 and 2.9. Their proofs are quite similar.

Proof of Corollary 2.8. By the linearity of differential operator $\mathring{\mathcal{L}}$, we have for the gPC solution,

$$v = \bar{v} + \mathbf{V}\mathbf{H}^T,$$

$$\mathcal{L}v = \mathring{\mathcal{L}}v + f = \mathring{\mathcal{L}}\bar{v} + \mathring{\mathcal{L}}\mathbf{V}\mathbf{H}^T + f,$$

where $\mathring{\mathcal{L}}\mathbf{V} = (v_{\alpha_1}, v_{\alpha_2}, \dots, v_{\alpha_{N_P}})_{\alpha_i \in \mathfrak{J}}$ is a row vector. By the linearity of expectation, we have from the above equality

$$\mathbb{E}[\mathcal{L}v] = \mathring{\mathcal{L}}\bar{v} + \bar{f},$$

$$\tilde{\mathcal{L}}v = \mathring{\mathcal{L}}\mathbf{V}\mathbf{H}^T + f - \bar{f},$$

$$\mathbb{E}[\tilde{\mathcal{L}}v\mathbf{H}] = \mathring{\mathcal{L}}\mathbf{V} + \mathbf{F},$$

where $\bar{f} = \mathbb{E}[f]$ and $\mathbf{F} = \mathbb{E}[f\mathbf{H}]$, and we have used $\mathbb{E}[\mathbf{H}\mathbf{H}^T] = \mathbf{I}$ to get the last equation. We

have similar equations for the DyBO-gPC solution, $u = \bar{u} + \mathbf{U}\mathbf{A}^T\mathbf{H}^T$,

$$\mathbb{E}[\mathcal{L}u] = \mathring{\mathcal{L}}\bar{u} + \bar{f},$$

$$\tilde{\mathcal{L}}u = \mathring{\mathcal{L}}\mathbf{U}\mathbf{A}^T\mathbf{H}^T + f - \bar{f},$$

$$\mathbb{E}[\tilde{\mathcal{L}}u\mathbf{H}] = \mathring{\mathcal{L}}\mathbf{U}\mathbf{A}^T + \mathbf{F}.$$

Eqn. (2.63a) follows immediately from the above equations. Substituting the above equations into eqn. (2.62b), we obtain

$$\begin{aligned} \frac{\partial \epsilon}{\partial t} &= \left(\mathring{\mathcal{L}}\mathbf{U}\mathbf{A}^T + \mathbf{F} \right) \mathbf{A}\mathbf{A}^T - \left(\mathring{\mathcal{L}}\mathbf{V} + \mathbf{F} \right) + \mathbf{U}\Lambda_{\mathbf{U}}^{-1} \left\langle \mathbf{U}^T, \mathring{\mathcal{L}}\mathbf{U}\mathbf{A}^T + \mathbf{F} \right\rangle (\mathbf{I} - \mathbf{A}\mathbf{A}^T) \\ &= \mathring{\mathcal{L}}\epsilon + \mathbf{F} (\mathbf{A}\mathbf{A}^T - \mathbf{I}) + \mathbf{U}\Lambda_{\mathbf{U}}^{-1} \left\langle \mathbf{U}^T, \mathring{\mathcal{L}}\mathbf{U} \right\rangle (\mathbf{A}^T - \mathbf{A}^T\mathbf{A}\mathbf{A}^T) \\ &\quad + \mathbf{U}\Lambda_{\mathbf{U}}^{-1} \left\langle \mathbf{U}^T, \mathbf{F} \right\rangle (\mathbf{I} - \mathbf{A}\mathbf{A}^T) \\ &= \mathring{\mathcal{L}}\epsilon + \mathbf{F} (\mathbf{A}\mathbf{A}^T - \mathbf{I}) + \mathbf{U}\Lambda_{\mathbf{U}}^{-1} \left\langle \mathbf{U}^T, \mathbf{F} \right\rangle (\mathbf{I} - \mathbf{A}\mathbf{A}^T). \end{aligned}$$

This completes the proof. □

Proof of Corollary 2.9. Eqn. (2.66a) is obvious. To get eqn. (2.63b), we compute directly

$$\begin{aligned} \frac{\partial \epsilon}{\partial t} &= \left(\mathring{\mathcal{L}}\mathbf{U}\mathbf{A}^T - \mathring{\mathcal{L}}\mathbf{V} \right) \mathbf{A} + \epsilon\mathbf{D} + \mathbf{V} (\mathbf{A}\mathbf{A}^T - \mathbf{I}) \left\langle \mathbf{F}^T + \mathbf{A}\mathring{\mathcal{L}}\mathbf{U}^T, \mathbf{U} \right\rangle \Lambda_{\mathbf{U}}^{-1} \\ &= \mathring{\mathcal{L}}\epsilon + \epsilon\mathbf{D} + \mathbf{V} (\mathbf{A}\mathbf{A}^T - \mathbf{I}) \left\langle \mathbf{F}^T, \mathbf{U} \right\rangle \Lambda_{\mathbf{U}}^{-1} \\ &\quad + \mathbf{V} (\mathbf{A}\mathbf{A}^T - \mathbf{I}) \mathbf{A} \left\langle \mathring{\mathcal{L}}\mathbf{U}^T, \mathbf{U} \right\rangle \Lambda_{\mathbf{U}}^{-1}. \end{aligned}$$

This completes the proof. □

Appendix C

Derivation of DyBO Methods for a System of Time-Dependent SPDEs

In this appendix, we provide the details of the derivation of the DyBO formulation for the system of time-dependent SPDEs (6.1). The derivation is similar to that in Sec. 2.1. The major difference is that each physical component has its own KL expansion.

For $l = 1, 2, \dots, N_s$, consider the m_l -term truncated KL expansion of the l th physical component $u_l(x, t, \omega)$,

$$u_l = \bar{u}_l + \sum_{i=1}^{m_l} u_{li} Y_{li} = \bar{u}_l + \mathbf{U}_l \mathbf{Y}_l^T, \quad (\text{C.1})$$

where \mathbf{U}_l is a row vector consisting of functions of spatial coordinate x and temporal coordinate t

$$\mathbf{U}_l(x, t) = (u_{l1}(x, t), u_{l2}(x, t), \dots, u_{lm_l}(x, t)) \in \mathbb{R}^{1 \times m_l},$$

and \mathbf{Y}_l is a row vector of random variables

$$\mathbf{Y}_l(\omega, t) = (Y_{l1}(\omega, t), Y_{l2}(\omega, t), \dots, Y_{lm_l}(\omega, t)) \in \mathbb{R}^{1 \times m_l}.$$

We also write the mode numbers of different physical components as a row vector $\mathbf{m} = (m_1, \dots, m_{N_s})$.

We call \mathbf{m} a mode number vector. Substituting the expansion (C.1) into the l th equation of the SPDE

system yields

$$\frac{\partial \bar{u}_l}{\partial t} + \frac{\partial \mathbf{U}_l}{\partial t} \mathbf{Y}_l^T + \mathbf{U}_l \frac{d\mathbf{Y}_l^T}{dt} = \mathcal{L}_l u, \quad (\text{C.2})$$

or equivalently,

$$\frac{\partial \bar{u}_l}{\partial t} + \mathbf{Y}_l \frac{\partial \mathbf{U}_l^T}{\partial t} + \frac{d\mathbf{Y}_l}{dt} \mathbf{U}_l^T = \mathcal{L}_l u. \quad (\text{C.3})$$

Taking expectations on both sides gives

$$\frac{\partial \bar{u}_l}{\partial t} = \mathbb{E}[\mathcal{L}_l u].$$

We write $\tilde{\mathcal{L}}_l u = \mathcal{L}_l u - \mathbb{E}[\mathcal{L}_l u]$. Next, we multiply both sides of eqn. (C.2) by \mathbf{Y}_l from the right, take expectations and use the orthogonality of stochastic processes \mathbf{Y}_l . Similarly, multiply both sides of eqn. (C.3) by \mathbf{U}_l from the right, take inner products with \mathbf{U}_l and use the orthogonality of \mathbf{U}_l . At the end, we arrive at the following system, for $l = 1, 2, \dots, N_s$,

$$\frac{\partial \bar{u}_l}{\partial t} = \mathbb{E}[\mathcal{L}_l u], \quad (\text{C.4a})$$

$$\frac{\partial \mathbf{U}_l}{\partial t} = \mathbb{E}[\tilde{\mathcal{L}}_l u \mathbf{Y}_l] - \mathbf{U}_l \mathbb{E}\left[\frac{d\mathbf{Y}_l^T}{dt} \mathbf{Y}_l\right], \quad (\text{C.4b})$$

$$\frac{d\mathbf{Y}_l}{dt} \mathbf{\Lambda}_{\mathbf{U}_l} = \langle \tilde{\mathcal{L}}_l u, \mathbf{U}_l \rangle - \mathbf{Y}_l \left\langle \frac{\partial \mathbf{U}_l^T}{\partial t}, \mathbf{U}_l \right\rangle, \quad (\text{C.4c})$$

where $\mathbf{\Lambda}_{\mathbf{U}_l} = \text{diag}(\langle \mathbf{U}_l^T, \mathbf{U}_l \rangle) = (\langle u_{li}, u_{lj} \rangle \delta_{ij})_{ij} \in \mathbb{R}^{m_l \times m_l}$. By substituting eqn. (C.4c) into eqn. (C.4b), we can eliminate the time derivative $\frac{\partial \mathbf{U}_l^T}{\partial t}$ from eqn. (C.4b). Similarly, by substituting eqn. (C.4b) into eqn. (C.4c), we can eliminate the time derivative $\frac{\partial \mathbf{Y}_l^T}{\partial t}$ from eqn. (C.4c). After these

two steps, the system (C.4) becomes, for $l = 1, 2, \dots, N_s$,

$$\frac{\partial \bar{u}_l}{\partial t} = \mathbb{E}[\mathcal{L}_l u], \quad (\text{C.5a})$$

$$\frac{\partial \mathbf{U}_l}{\partial t} = \mathbf{U}_l \mathbf{\Lambda}_{\mathbf{U}_l}^{-1} \left\langle \mathbf{U}_l^T, \frac{\partial \mathbf{U}_l}{\partial t} \right\rangle + G_{\mathbf{U}_l}(u, \mathbf{U}_l, \mathbf{Y}_l), \quad (\text{C.5b})$$

$$\frac{d\mathbf{Y}_l}{dt} = \mathbf{Y}_l \mathbb{E} \left[\mathbf{Y}_l^T \frac{d\mathbf{Y}_l}{dt} \right] + G_{\mathbf{Y}_l}(u, \mathbf{U}_l, \mathbf{Y}_l), \quad (\text{C.5c})$$

where

$$\begin{aligned} G_{\mathbf{U}_l}(u, \mathbf{U}_l, \mathbf{Y}_l) &= \Pi_{\mathbf{U}_l} \left(\mathbb{E} \left[\tilde{\mathcal{L}}_l u \mathbf{Y}_l \right] \right) \\ &= \mathbb{E} \left[\tilde{\mathcal{L}}_l u \mathbf{Y}_l \right] - \mathbf{U}_l \mathbf{\Lambda}_{\mathbf{U}_l}^{-1} \left\langle \mathbf{U}_l^T, \mathbb{E} \left[\tilde{\mathcal{L}}_l u \mathbf{Y}_l \right] \right\rangle, \end{aligned} \quad (\text{C.6a})$$

$$\begin{aligned} G_{\mathbf{Y}_l}(u, \mathbf{U}_l, \mathbf{Y}_l) &= \Pi_{\mathbf{Y}_l} \left(\left\langle \tilde{\mathcal{L}}_l u, \mathbf{U}_l \right\rangle \right) \mathbf{\Lambda}_{\mathbf{U}_l}^{-1} \\ &= \left\langle \tilde{\mathcal{L}}_l u, \mathbf{U}_l \right\rangle \mathbf{\Lambda}_{\mathbf{U}_l}^{-1} - \mathbf{Y}_l \left\langle \mathbb{E} \left[\tilde{\mathcal{L}}_l u \mathbf{Y}_l^T \right], \mathbf{U}_l \right\rangle \mathbf{\Lambda}_{\mathbf{U}_l}^{-1}, \end{aligned} \quad (\text{C.6b})$$

i.e., $G_{\mathbf{U}_l}(u, \mathbf{U}_l, \mathbf{Y}_l)$ is the orthogonal compliment of $\mathbb{E} \left[\tilde{\mathcal{L}}_l u \mathbf{Y}_l \right]$ with respect to $\text{span}(\mathbf{U}_l)$ in $\mathbb{L}^2(\mathcal{D})$ and $G_{\mathbf{Y}_l}(u, \mathbf{U}_l, \mathbf{Y}_l)$ is the orthogonal compliment of $\left\langle \tilde{\mathcal{L}}_l u, \mathbf{U}_l \right\rangle$ with respect to $\text{span}(\mathbf{Y}_l)$ in $\mathbb{L}^2(\Omega)$.

By using the anti-symmetrization \mathcal{Q} and the partial anti-symmetrization $\tilde{\mathcal{Q}}$ operators, we can enforce the bi-orthogonality, which is essential to the KL expansions. Finally, we obtain an equivalent system from (C.5),

$$\frac{\partial \bar{u}_l}{\partial t} = \mathbb{E}[\mathcal{L}_l u], \quad (\text{C.7a})$$

$$\frac{\partial \mathbf{U}_l}{\partial t} = \mathbf{U}_l \mathbf{\Lambda}_{\mathbf{U}_l}^{-1} \tilde{\mathcal{Q}} \left(\left\langle \mathbf{U}_l^T, \frac{\partial \mathbf{U}_l}{\partial t} \right\rangle \right) + G_{\mathbf{U}_l}(u, \mathbf{U}_l, \mathbf{Y}_l), \quad (\text{C.7b})$$

$$\frac{d\mathbf{Y}_l}{dt} = \mathbf{Y}_l \mathcal{Q} \left(\mathbb{E} \left[\mathbf{Y}_l^T \frac{d\mathbf{Y}_l}{dt} \right] \right) + G_{\mathbf{Y}_l}(u, \mathbf{U}_l, \mathbf{Y}_l). \quad (\text{C.7c})$$

Let $\tilde{\mathbf{U}}_l$ be orthogonal to \mathbf{U}_l and $(\mathbf{U}_l, \tilde{\mathbf{U}}_l)$ form a complete set of orthogonal basis of $\mathbb{L}^2(\mathcal{D})$. To

remove the term involving time derivative of \mathbf{U}_l from the right hand side of eqn. (C.7b), we notice the growth of the spatial mode \mathbf{U}_l of the l th component can be written in the form of

$$\frac{\partial \mathbf{U}_l}{\partial t} = \mathbf{U}_l \mathbf{C}_l + \tilde{\mathbf{U}}_l \tilde{\mathbf{C}}_l, \quad (\text{C.8})$$

where $\mathbf{C}_l(t) \in \mathbb{R}^{m_l \times m_l}$ and $\tilde{\mathbf{C}}_l(t) \in \mathbb{R}^{\infty \times m_l}$. As a result, we get

$$\left\langle \mathbf{U}_l^T, \frac{\partial \mathbf{U}_l}{\partial t} \right\rangle = \left\langle \mathbf{U}_l^T, \mathbf{U}_l \mathbf{C}_l + \tilde{\mathbf{U}}_l \tilde{\mathbf{C}}_l \right\rangle = \left\langle \mathbf{U}_l^T, \mathbf{U}_l \right\rangle \mathbf{C}_l + \left\langle \mathbf{U}_l^T, \tilde{\mathbf{U}}_l \right\rangle \tilde{\mathbf{C}}_l = \left\langle \mathbf{U}_l^T, \mathbf{U}_l \right\rangle \mathbf{C}_l.$$

Substituting this and eqn. (C.8) into the second equation of the system (C.7) gives,

$$\mathbf{U}_l \mathbf{C}_l + \tilde{\mathbf{U}}_l \tilde{\mathbf{C}}_l = \mathbf{U}_l \mathbf{\Lambda}_{\mathbf{U}_l}^{-1} \tilde{\mathcal{Q}}(\mathbf{\Lambda}_{\mathbf{U}_l} \mathbf{C}_l) + G_{\mathbf{U}_l}(u, \mathbf{U}_l, \mathbf{Y}_l),$$

or

$$\mathbf{U}_l \left(\mathbf{C}_l - \mathbf{\Lambda}_{\mathbf{U}_l}^{-1} \tilde{\mathcal{Q}}(\mathbf{\Lambda}_{\mathbf{U}_l} \mathbf{C}_l) \right) = G_{\mathbf{U}_l}(u, \mathbf{U}_l, \mathbf{Y}_l) - \tilde{\mathbf{U}}_l \tilde{\mathbf{C}}_l,$$

where the left side is in $\text{span}(\mathbf{U}_l) \subset \mathbb{L}^2(\mathcal{D})$ while the right side is in its orthogonal complement.

Thus, we have

$$\mathbf{C}_l - \mathbf{\Lambda}_{\mathbf{U}_l}^{-1} \tilde{\mathcal{Q}}(\mathbf{\Lambda}_{\mathbf{U}_l} \mathbf{C}_l) = 0, \quad (\text{C.9a})$$

$$\tilde{\mathbf{U}}_l \tilde{\mathbf{C}}_l = G_{\mathbf{U}_l}(u, \mathbf{U}_l, \mathbf{Y}_l). \quad (\text{C.9b})$$

Similarly, the growth of stochastic modes $\frac{d\mathbf{Y}_l}{dt}$ can be written in the form of

$$\frac{d\mathbf{Y}_l}{dt} = \mathbf{Y}_l \mathbf{D}_l + \tilde{\mathbf{Y}}_l \tilde{\mathbf{D}}_l, \quad (\text{C.10})$$

where $\mathbf{D}_l(t) \in \mathbb{R}^{m_l \times m_l}$ and $\tilde{\mathbf{D}}_l(t) \in \mathbb{R}^{\infty \times m_l}$ and

$$\mathbf{D}_l - \mathcal{Q}(\mathbf{D}_l) = 0, \quad (\text{C.11a})$$

$$\tilde{\mathbf{Y}}_l \tilde{\mathbf{D}}_l = G_{\mathbf{Y}_l}(u, \mathbf{U}_l, \mathbf{Y}_l). \quad (\text{C.11b})$$

To find additional equations for \mathbf{C}_l and \mathbf{D}_l , we substitute eqn. (C.8) and eqn. (C.10) into the l th SPDE of system (6.1) to get

$$\mathbf{U}_l (\mathbf{D}_l^T + \mathbf{C}_l) \mathbf{Y}_l^T + \mathbf{U}_l \tilde{\mathbf{D}}_l^T \tilde{\mathbf{Y}}_l^T + \tilde{\mathbf{U}}_l \tilde{\mathbf{C}}_l \mathbf{Y}_l^T = \tilde{\mathcal{L}}_l u.$$

Multiplying \mathbf{U}_l^T from the left and \mathbf{Y}_l from the right on both sides of the above equality and taking both inner products $\langle \cdot, \cdot \rangle$ and expectations $\mathbb{E}[\cdot]$, we obtain

$$\begin{aligned} \langle \mathbf{U}_l^T, \mathbf{U}_l \rangle (\mathbf{D}_l^T + \mathbf{C}_l) \mathbb{E}[\mathbf{Y}_l^T \mathbf{Y}_l] + \langle \mathbf{U}_l^T, \mathbf{U}_l \rangle \tilde{\mathbf{D}}_l^T \mathbb{E}[\tilde{\mathbf{Y}}_l^T \mathbf{Y}_l] \\ + \langle \mathbf{U}_l^T, \tilde{\mathbf{U}}_l \rangle \tilde{\mathbf{C}}_l \mathbb{E}[\mathbf{Y}_l^T \mathbf{Y}_l] = \langle \mathbf{U}_l^T, \mathbb{E}[\tilde{\mathcal{L}}_l u \mathbf{Y}_l] \rangle. \end{aligned}$$

Again, by the bi-orthogonality condition, we can eliminate $\tilde{\mathbf{U}}_l$ and $\tilde{\mathbf{Y}}_l$ from the above and get

$$\mathbf{D}_l^T + \mathbf{C}_l = G_{*l}(u, \mathbf{U}_l, \mathbf{Y}_l), \quad (\text{C.12})$$

where $G_{*l}(u, \mathbf{U}_l, \mathbf{Y}_l) = \mathbf{\Lambda}_{\mathbf{U}_l}^{-1} \langle \mathbf{U}_l^T, \mathbb{E}[\tilde{\mathcal{L}}_l u \mathbf{Y}_l] \rangle \in \mathbb{R}^{m_l \times m_l}$. Thus, we have the following three

matrix equations for \mathbf{C}_l and \mathbf{D}_l ,

$$\mathbf{C}_l - \mathbf{\Lambda}_{\mathbf{U}_l}^{-1} \tilde{\mathcal{Q}} (\mathbf{\Lambda}_{\mathbf{U}_l} \mathbf{C}_l) = 0, \quad (\text{C.13a})$$

$$\mathbf{D}_l - \mathcal{Q} (\mathbf{D}_l) = 0, \quad (\text{C.13b})$$

$$\mathbf{D}_l^T + \mathbf{C}_l = G_{*l}(u, \mathbf{U}_l, \mathbf{Y}_l). \quad (\text{C.13c})$$

According to Theorem 2.2, matrices \mathbf{C}_l and \mathbf{D}_l can then be solved uniquely via eqn. (2.22) from eqn. (C.13) as long as $\|u_i\|_{\mathbb{L}^2(\mathcal{D})} \neq \|u_j\|_{\mathbb{L}^2(\mathcal{D})}$ for $i \neq j$.

Combining the above discussions, we arrive at the DyBO formulation for the time-evolutionary system of SPDEs (6.1), for $l = 1, 2, \dots, N_s$,

$$\frac{\partial \bar{u}_l}{\partial t} = \mathbb{E} [\mathcal{L}_l u], \quad (\text{C.14a})$$

$$\frac{\partial \mathbf{U}_l}{\partial t} = -\mathbf{U}_l \mathbf{D}_l^T + \mathbb{E} [\tilde{\mathcal{L}}_l u \mathbf{Y}_l], \quad (\text{C.14b})$$

$$\frac{d\mathbf{Y}_l}{dt} = -\mathbf{Y}_l \mathbf{C}_l^T + \langle \tilde{\mathcal{L}}_l u, \mathbf{U}_l \rangle \mathbf{\Lambda}_{\mathbf{U}_l}^{-1}, \quad (\text{C.14c})$$

where matrices \mathbf{C}_l 's and \mathbf{D}_l 's can be solved from linear systems (C.13).

If the stochastic modes \mathbf{Y}_l are presented in the form of the gPC expansion, i.e.,

$$\mathbf{Y}_l(\omega, t) = \mathbf{H}(\boldsymbol{\xi}(\omega)) \mathbf{A}_l, \quad (\text{C.15})$$

where $\mathbf{A}_l \in \mathbb{R}^{N_p \times m_l}$ for $l = 1, 2, \dots, N_s$. then the DyBO-gPC formulation becomes, ($l = 1, \dots, N_s$),

$$\frac{\partial \bar{u}_l}{\partial t} = \mathbb{E}[\mathcal{L}_l u], \quad (\text{C.16a})$$

$$\frac{\partial \mathbf{U}_l}{\partial t} = -\mathbf{U}_l \mathbf{D}_l^T + \mathbb{E}[\tilde{\mathcal{L}}_l u \mathbf{H}] \mathbf{A}_l, \quad (\text{C.16b})$$

$$\frac{d\mathbf{A}_l}{dt} = -\mathbf{A}_l \mathbf{C}_l^T + \langle \mathbb{E}[\mathbf{H}^T \tilde{\mathcal{L}}_l u], \mathbf{U}_l \rangle \mathbf{\Lambda}_{\mathbf{U}_l}^{-1}, \quad (\text{C.16c})$$

where $\mathbf{C}_l(t)$ and $\mathbf{D}_l(t)$ can be solved from

$$G_{*l}(u, \mathbf{U}_l, \mathbf{Y}_l) = \mathbf{\Lambda}_{\mathbf{U}_l}^{-1} \langle \mathbf{U}_l^T, \mathbb{E}[\tilde{\mathcal{L}}_l u \mathbf{Y}_l] \rangle = \mathbf{\Lambda}_{\mathbf{U}_l}^{-1} \langle \mathbf{U}_l^T, \mathbb{E}[\tilde{\mathcal{L}}_l u \mathbf{H}] \rangle \mathbf{A}_l. \quad (\text{C.17})$$

Appendix D

Derivations of gPC Formulation of SNSE

In this appendix, we provide the details of the derivation of the gPC formulation of SNSE (6.9) by using the vector/matrix/tensor notations introduced earlier. In addition to the gPC expansions (6.13) of the temperature and vorticity components, we also need the gPC expansions of the stream function and the velocity components in the x and y directions, i.e.,

$$\psi = \bar{\psi} + \boldsymbol{\psi}\mathbf{H}^T, \quad (\text{D.1a})$$

$$u = \bar{u} + \mathbf{U}\mathbf{H}^T, \quad (\text{D.1b})$$

$$v = \bar{v} + \mathbf{V}\mathbf{H}^T, \quad (\text{D.1c})$$

where $\boldsymbol{\psi} = (\psi_\alpha)_{\alpha \in \mathfrak{J}}$, $\mathbf{U} = (u_\alpha)_{\alpha \in \mathfrak{J}}$ and $\mathbf{V} = (v_\alpha)_{\alpha \in \mathfrak{J}}$ are row vectors. Plugging the above expansions into eqn. (6.10), we have

$$-\Delta\bar{\psi} - \Delta\boldsymbol{\psi}\mathbf{H}^T = \bar{w} + \mathbf{W}\mathbf{H}^T,$$

or

$$-\Delta\bar{\psi} = \bar{w}, \quad (\text{D.2})$$

$$-\Delta\psi = \mathbf{W}. \quad (\text{D.3})$$

Similarly, by plugging the expansions (D.1) into eqn. (6.11) and eqn. (6.12), we have

$$\bar{u} = \frac{\partial\bar{\psi}}{\partial y} = -\frac{\partial}{\partial y}\Delta^{-1}\bar{w}, \quad (\text{D.4})$$

$$\bar{v} = -\frac{\partial\bar{\psi}}{\partial x} = \frac{\partial}{\partial x}\Delta^{-1}\bar{w}, \quad (\text{D.5})$$

$$\mathbf{U} = \frac{\partial\psi}{\partial y} = -\frac{\partial}{\partial y}\Delta^{-1}\mathbf{W}, \quad (\text{D.6})$$

$$\mathbf{V} = -\frac{\partial\psi}{\partial x} = \frac{\partial}{\partial x}\Delta^{-1}\mathbf{W}. \quad (\text{D.7})$$

Substituting the gPC expansions (6.13) into the first equation of SNSE (6.9a), we get

$$\begin{aligned} \mathcal{L}_\theta\{w, \theta\} &= -(\bar{u} + \mathbf{U}\mathbf{H}^T) \left(\frac{\partial\bar{\theta}}{\partial x} + \frac{\partial\boldsymbol{\theta}}{\partial x}\mathbf{H}^T \right) - (\bar{v} + \mathbf{V}\mathbf{H}^T) \left(\frac{\partial\bar{\theta}}{\partial y} + \frac{\partial\boldsymbol{\theta}}{\partial y}\mathbf{H}^T \right) + \kappa\Delta\bar{\theta} + \kappa\Delta\boldsymbol{\theta}\mathbf{H}^T \\ &= \left(\kappa\Delta\bar{\theta} - \bar{u}\frac{\partial\bar{\theta}}{\partial x} - \bar{v}\frac{\partial\bar{\theta}}{\partial y} \right) \\ &\quad + \left(\kappa\Delta\boldsymbol{\theta} - \bar{u}\frac{\partial\boldsymbol{\theta}}{\partial x} - \bar{v}\frac{\partial\boldsymbol{\theta}}{\partial y} - \frac{\partial\bar{\theta}}{\partial x}\mathbf{U} - \frac{\partial\bar{\theta}}{\partial y}\mathbf{V} \right) \mathbf{H}^T - \mathbf{U}\mathbf{H}^T\mathbf{H}\frac{\partial\boldsymbol{\theta}^T}{\partial x} - \mathbf{V}\mathbf{H}^T\mathbf{H}\frac{\partial\boldsymbol{\theta}^T}{\partial y}. \end{aligned}$$

Taking expectations on both sides, we get

$$\begin{aligned}\mathbb{E}[\mathcal{L}_\theta\{w, \theta\}] &= \left(\kappa\Delta\bar{\theta} - \bar{u}\frac{\partial\bar{\theta}}{\partial x} - \bar{v}\frac{\partial\bar{\theta}}{\partial y}\right) - \mathbf{U}\frac{\partial\theta^T}{\partial x} - \mathbf{V}\frac{\partial\theta^T}{\partial y}, \\ \tilde{\mathcal{L}}_\theta\{w, \theta\} &= \left(\kappa\Delta\theta - \bar{u}\frac{\partial\theta}{\partial x} - \bar{v}\frac{\partial\theta}{\partial y} - \frac{\partial\bar{\theta}}{\partial x}\mathbf{U} - \frac{\partial\bar{\theta}}{\partial y}\mathbf{V}\right)\mathbf{H}^T \\ &\quad + \mathbf{U}\frac{\partial\theta^T}{\partial x} + \mathbf{V}\frac{\partial\theta^T}{\partial y} - \mathbf{U}\mathbf{H}^T\mathbf{H}\frac{\partial\theta^T}{\partial x} - \mathbf{V}\mathbf{H}^T\mathbf{H}\frac{\partial\theta^T}{\partial y},\end{aligned}$$

and

$$\begin{aligned}\mathbb{E}\left[\tilde{\mathcal{L}}_\theta\{w, \theta\}\mathbf{H}\right] &= \left(\kappa\Delta\theta - \bar{u}\frac{\partial\theta}{\partial x} - \bar{v}\frac{\partial\theta}{\partial y} - \frac{\partial\bar{\theta}}{\partial x}\mathbf{U} - \frac{\partial\bar{\theta}}{\partial y}\mathbf{V}\right) - \mathbb{E}\left[\mathbf{U}\mathbf{H}^T\mathbf{H}\frac{\partial\theta^T}{\partial x}\mathbf{H}\right] \\ &\quad - \mathbb{E}\left[\mathbf{V}\mathbf{H}^T\mathbf{H}\frac{\partial\theta^T}{\partial y}\mathbf{H}\right] \\ &= \left(\kappa\Delta\theta - \bar{u}\frac{\partial\theta}{\partial x} - \bar{v}\frac{\partial\theta}{\partial y} - \frac{\partial\bar{\theta}}{\partial x}\mathbf{U} - \frac{\partial\bar{\theta}}{\partial y}\mathbf{V}\right) - \left(\left(u_\alpha\frac{\partial\theta_\beta}{\partial x} + v_\alpha\frac{\partial\theta_\beta}{\partial y}\right)\mathfrak{I}_{\alpha\beta\gamma}^{(\mathbf{H})}\right)_{1\times\gamma}.\end{aligned}$$

Similarly, we have for the second equation of SNSE (6.9),

$$\begin{aligned}\mathcal{L}_w(w, \theta) &= -(\bar{u} + \mathbf{U}\mathbf{H}^T)\left(\frac{\partial\bar{w}}{\partial x} + \frac{\partial\mathbf{W}}{\partial x}\mathbf{H}^T\right) - (\bar{v} + \mathbf{V}\mathbf{H}^T)\left(\frac{\partial\bar{w}}{\partial y} + \frac{\partial\mathbf{W}}{\partial y}\mathbf{H}^T\right) \\ &\quad + (\nu\Delta\bar{w} + \nu\Delta\mathbf{W}\mathbf{H}^T) + \mathbf{F}\mathbf{H}^T + \left(\mu g\frac{\partial\bar{\theta}}{\partial x} + \mu g\frac{\partial\theta}{\partial x}\mathbf{H}^T\right) \\ &= \left(\nu\Delta\bar{w} - \bar{u}\frac{\partial\bar{w}}{\partial x} - \bar{v}\frac{\partial\bar{w}}{\partial y} + \mu g\frac{\partial\bar{\theta}}{\partial x}\right) \\ &\quad + \left(\nu\Delta\mathbf{W} - \bar{u}\frac{\partial\mathbf{W}}{\partial x} - \bar{v}\frac{\partial\mathbf{W}}{\partial y} - \frac{\partial\bar{w}}{\partial x}\mathbf{U} - \frac{\partial\bar{w}}{\partial y}\mathbf{V} + \mu g\frac{\partial\theta}{\partial x}\right)\mathbf{H}^T \\ &\quad - \mathbf{U}\mathbf{H}^T\mathbf{H}\frac{\partial\mathbf{W}^T}{\partial x} - \mathbf{V}\mathbf{H}^T\mathbf{H}\frac{\partial\mathbf{W}^T}{\partial y} + \mathbf{F}\mathbf{H}^T,\end{aligned}$$

where the second term on the second line is due to the stochastic force f and the third term is due to the buoyancy force. Thus,

$$\begin{aligned}\mathbb{E}[\mathcal{L}_w(w, \theta)] &= \left(\nu \Delta \bar{w} - \bar{u} \frac{\partial \bar{w}}{\partial x} - \bar{v} \frac{\partial \bar{w}}{\partial y} + \mu g \frac{\partial \bar{\theta}}{\partial x} \right) - \mathbf{U} \frac{\partial \mathbf{W}^T}{\partial x} - \mathbf{V} \frac{\partial \mathbf{W}^T}{\partial y}, \\ \tilde{\mathcal{L}}_w(w, \theta) &= \left(\nu \Delta \mathbf{W} - \bar{u} \frac{\partial \mathbf{W}}{\partial x} - \bar{v} \frac{\partial \mathbf{W}}{\partial y} - \frac{\partial \bar{w}}{\partial x} \mathbf{U} - \frac{\partial \bar{w}}{\partial y} \mathbf{V} + \mu g \frac{\partial \theta}{\partial x} \right) \mathbf{H}^T + \mathbf{U} \frac{\partial \mathbf{W}^T}{\partial x} + \mathbf{V} \frac{\partial \mathbf{W}^T}{\partial y} \\ &\quad - \mathbf{U} \mathbf{H}^T \mathbf{H} \frac{\partial \mathbf{W}^T}{\partial x} - \mathbf{V} \mathbf{H}^T \mathbf{H} \frac{\partial \mathbf{W}^T}{\partial y} + \mathbf{F} \mathbf{H}^T,\end{aligned}$$

and

$$\begin{aligned}\mathbb{E}[\tilde{\mathcal{L}}_w(w, \theta) \mathbf{H}] &= \left(\nu \Delta \mathbf{W} - \bar{u} \frac{\partial \mathbf{W}}{\partial x} - \bar{v} \frac{\partial \mathbf{W}}{\partial y} - \frac{\partial \bar{w}}{\partial x} \mathbf{U} - \frac{\partial \bar{w}}{\partial y} \mathbf{V} + \mu g \frac{\partial \theta}{\partial x} \right) \\ &\quad - \mathbb{E} \left[\mathbf{U} \mathbf{H}^T \mathbf{H} \frac{\partial \mathbf{W}^T}{\partial x} \mathbf{H} \right] - \mathbb{E} \left[\mathbf{V} \mathbf{H}^T \mathbf{H} \frac{\partial \mathbf{W}^T}{\partial y} \mathbf{H} \right] + \mathbf{F} \\ &= \left(\nu \Delta \mathbf{W} - \bar{u} \frac{\partial \mathbf{W}}{\partial x} - \bar{v} \frac{\partial \mathbf{W}}{\partial y} - \frac{\partial \bar{w}}{\partial x} \mathbf{U} - \frac{\partial \bar{w}}{\partial y} \mathbf{V} + \mu g \frac{\partial \theta}{\partial x} \right) \\ &\quad - \left(\left(u_\alpha \frac{\partial w_\beta}{\partial x} + v_\alpha \frac{\partial w_\beta}{\partial y} \right) \mathfrak{F}_{\alpha\beta\gamma}^{(\mathbf{H})} \right)_{1 \times \gamma} + \mathbf{F}.\end{aligned}$$

Combining the above discussion, we derive the gPC formulation for SNSE (6.9),

$$\frac{\partial \bar{\theta}}{\partial t} = \left(\kappa \Delta \bar{\theta} - \bar{u} \frac{\partial \bar{\theta}}{\partial x} - \bar{v} \frac{\partial \bar{\theta}}{\partial y} \right) - \mathbf{U} \frac{\partial \theta^T}{\partial x} - \mathbf{V} \frac{\partial \theta^T}{\partial y}, \quad (\text{D.8a})$$

$$\frac{\partial \bar{w}}{\partial t} = \left(\nu \Delta \bar{w} - \bar{u} \frac{\partial \bar{w}}{\partial x} - \bar{v} \frac{\partial \bar{w}}{\partial y} + \mu g \frac{\partial \bar{\theta}}{\partial x} \right) - \mathbf{U} \frac{\partial \mathbf{W}^T}{\partial x} - \mathbf{V} \frac{\partial \mathbf{W}^T}{\partial y}, \quad (\text{D.8b})$$

$$\frac{\partial \theta}{\partial t} = \left(\kappa \Delta \theta - \bar{u} \frac{\partial \theta}{\partial x} - \bar{v} \frac{\partial \theta}{\partial y} - \frac{\partial \bar{\theta}}{\partial x} \mathbf{U} - \frac{\partial \bar{\theta}}{\partial y} \mathbf{V} \right) - \left(\left(u_\alpha \frac{\partial \theta_\beta}{\partial x} + v_\alpha \frac{\partial \theta_\beta}{\partial y} \right) \mathfrak{F}_{\alpha\beta\gamma}^{(\mathbf{H})} \right)_{1 \times \gamma}, \quad (\text{D.8c})$$

$$\begin{aligned}\frac{\partial \mathbf{W}}{\partial t} &= \left(\nu \Delta \mathbf{W} - \bar{u} \frac{\partial \mathbf{W}}{\partial x} - \bar{v} \frac{\partial \mathbf{W}}{\partial y} - \frac{\partial \bar{w}}{\partial x} \mathbf{U} - \frac{\partial \bar{w}}{\partial y} \mathbf{V} + \mu g \frac{\partial \theta}{\partial x} \right) \\ &\quad - \left(\left(u_\alpha \frac{\partial w_\beta}{\partial x} + v_\alpha \frac{\partial w_\beta}{\partial y} \right) \mathfrak{F}_{\alpha\beta\gamma}^{(\mathbf{H})} \right)_{1 \times \gamma} + \mathbf{F}.\end{aligned} \quad (\text{D.8d})$$

To further simplify notations, we introduce a slightly generalized version of the material derivative of a scalar or row-vector field θ under scalar or row-vector velocity field u and v ,

$$\mathfrak{D}_{(u,v)}\theta = \begin{cases} u\frac{\partial\theta}{\partial x} + v\frac{\partial\theta}{\partial y}, & u, v, \theta \text{ are scalars,} \\ u\frac{\partial\theta}{\partial x} + v\frac{\partial\theta}{\partial y}, & u, v \text{ are scalars and } \theta \text{ is a row vector,} \\ \frac{\partial\theta}{\partial x}u + \frac{\partial\theta}{\partial y}v, & u, v \text{ are row vectors and } \theta \text{ is a scalar,} \\ u\frac{\partial\theta^T}{\partial x} + v\frac{\partial\theta^T}{\partial y}, & u, v, \theta \text{ are row vectors,} \end{cases} \quad (\text{D.9})$$

When u and v can be row vectors of the same length and the right-hand side is understood in the usual sense of vector-vector multiplications or scalar-vector multiplications. By using this notation, the above gPC formulation of SNSE can be compactly written as

$$\frac{\partial\bar{\theta}}{\partial t} = \kappa\Delta\bar{\theta} - \mathfrak{D}_{(\bar{u},\bar{v})}\bar{\theta} - \mathfrak{D}_{(\mathbf{U},\mathbf{V})}\theta, \quad (\text{D.10a})$$

$$\frac{\partial\bar{w}}{\partial t} = \nu\Delta\bar{w} - \mathfrak{D}_{(\bar{u},\bar{v})}\bar{w} - \mathfrak{D}_{(\mathbf{U},\mathbf{V})}\mathbf{W} + \mu g\frac{\partial\bar{\theta}}{\partial x}, \quad (\text{D.10b})$$

$$\frac{\partial\theta}{\partial t} = \kappa\Delta\theta - \mathfrak{D}_{(\bar{u},\bar{v})}\theta - \mathfrak{D}_{(\mathbf{U},\mathbf{V})}\bar{\theta} - \left(\mathfrak{D}_{(u_\alpha, v_\alpha)}\theta_\beta \mathfrak{T}_{\alpha\beta\gamma}^{(\mathbf{H})}\right)_{1\times\gamma}, \quad (\text{D.10c})$$

$$\frac{\partial\mathbf{W}}{\partial t} = \nu\Delta\mathbf{W} - \mathfrak{D}_{(\bar{u},\bar{v})}\mathbf{W} - \mathfrak{D}_{(\mathbf{U},\mathbf{V})}\bar{w} + \mu g\frac{\partial\theta}{\partial x} - \left(\mathfrak{D}_{(u_\alpha, v_\alpha)}w_\beta \mathfrak{T}_{\alpha\beta\gamma}^{(\mathbf{H})}\right)_{1\times\gamma} + \mathbf{F}. \quad (\text{D.10d})$$

Appendix E

Derivations of the DyBO Formulation of SNSE

In this appendix, we provide the details of the derivations of the DyBO-gPC formulation of SNSE

(6.9). Plugging the expansion (6.13b) into the incompressibility condition (6.10), we have

$$-\Delta\psi = \bar{w} + \mathbf{W}\mathbf{Z}^T, \quad (\text{E.1})$$

which implies an expansion $\psi = \bar{\psi} + \boldsymbol{\psi}\mathbf{Z}^T$ with

$$-\Delta\bar{\psi} = \bar{w},$$

$$-\Delta\boldsymbol{\psi} = \mathbf{W}.$$

From the above expansion of ψ , it is easy to get expansions of u and v from eqn. (6.11) and eqn. (6.12),

$$\begin{aligned} u &= \frac{\partial\bar{\psi}}{\partial y} + \frac{\partial\boldsymbol{\psi}}{\partial y}\mathbf{Z}^T = \bar{u} + \mathbf{U}\mathbf{Z}^T, & \text{i.e., } \bar{u} &= \frac{\partial\bar{\psi}}{\partial y} = -\frac{\partial}{\partial y}\Delta^{-1}\bar{w}, \quad \mathbf{U} = \frac{\partial\boldsymbol{\psi}}{\partial y} = -\frac{\partial}{\partial y}\Delta^{-1}\mathbf{W}, \\ v &= -\frac{\partial\bar{\psi}}{\partial x} - \frac{\partial\boldsymbol{\psi}}{\partial x}\mathbf{Z}^T = \bar{v} + \mathbf{V}\mathbf{Z}^T, & \text{i.e., } \bar{v} &= -\frac{\partial\bar{\psi}}{\partial x} = \frac{\partial}{\partial x}\Delta^{-1}\bar{w}, \quad \mathbf{V} = -\frac{\partial\boldsymbol{\psi}}{\partial x} = \frac{\partial}{\partial x}\Delta^{-1}\mathbf{W}. \end{aligned}$$

Note that all these expansions of ψ , u and v are not necessarily KL expansions.

First, we derive the equation for the temperature. Substituting the KL expansion into eqn. (6.9b),

we have

$$\begin{aligned}
\mathcal{L}_\theta \{\theta, w\} &= -(\bar{u} + \mathbf{U}\mathbf{B}^T\mathbf{H}^T) \left(\frac{\partial\bar{\theta}}{\partial x} + \frac{\partial\boldsymbol{\theta}}{\partial x}\mathbf{A}^T\mathbf{H}^T \right) - (\bar{v} + \mathbf{V}\mathbf{B}^T\mathbf{H}^T) \left(\frac{\partial\bar{\theta}}{\partial y} + \frac{\partial\boldsymbol{\theta}}{\partial y}\mathbf{A}^T\mathbf{H}^T \right) \\
&\quad + \kappa(\Delta\bar{\theta} + \Delta\boldsymbol{\theta}\mathbf{A}^T\mathbf{H}^T) \\
&= -\left(\bar{u}\frac{\partial\bar{\theta}}{\partial x} + \bar{v}\frac{\partial\bar{\theta}}{\partial y} \right) + \kappa\Delta\bar{\theta} + \left[\kappa\Delta\boldsymbol{\theta} - \left(\bar{u}\frac{\partial\boldsymbol{\theta}}{\partial x} + \bar{v}\frac{\partial\boldsymbol{\theta}}{\partial y} \right) \right] \mathbf{A}^T\mathbf{H}^T \\
&\quad - \left(\frac{\partial\bar{\theta}}{\partial x}\mathbf{U} + \frac{\partial\bar{\theta}}{\partial y}\mathbf{V} \right) \mathbf{B}^T\mathbf{H}^T - \mathbf{U}\mathbf{B}^T\mathbf{H}^T\mathbf{H}\mathbf{A}\frac{\partial\boldsymbol{\theta}^T}{\partial x} - \mathbf{V}\mathbf{B}^T\mathbf{H}^T\mathbf{H}\mathbf{A}\frac{\partial\boldsymbol{\theta}^T}{\partial y}.
\end{aligned}$$

Taking expectations on both sides yields

$$\begin{aligned}
\mathbb{E}[\mathcal{L}_\theta \{\theta, w\}] &= -\left(\bar{u}\frac{\partial\bar{\theta}}{\partial x} + \bar{v}\frac{\partial\bar{\theta}}{\partial y} \right) + \kappa\Delta\bar{\theta} - \mathbf{U}\mathbf{B}^T\mathbf{A}\frac{\partial\boldsymbol{\theta}^T}{\partial x} - \mathbf{V}\mathbf{B}^T\mathbf{A}\frac{\partial\boldsymbol{\theta}^T}{\partial y}, \\
\tilde{\mathcal{L}}_\theta \{\theta, w\} &= \left[\kappa\Delta\boldsymbol{\theta}\mathbf{A}^T - \left(\bar{u}\frac{\partial\boldsymbol{\theta}}{\partial x} + \bar{v}\frac{\partial\boldsymbol{\theta}}{\partial y} \right) \mathbf{A}^T - \left(\frac{\partial\bar{\theta}}{\partial x}\mathbf{U} + \frac{\partial\bar{\theta}}{\partial y}\mathbf{V} \right) \mathbf{B}^T \right] \mathbf{H}^T \\
&\quad + \mathbf{U}\mathbf{B}^T\mathbf{A}\frac{\partial\boldsymbol{\theta}^T}{\partial x} + \mathbf{V}\mathbf{B}^T\mathbf{A}\frac{\partial\boldsymbol{\theta}^T}{\partial y} - \mathbf{U}\mathbf{B}^T\mathbf{H}^T\mathbf{H}\mathbf{A}\frac{\partial\boldsymbol{\theta}^T}{\partial x} - \mathbf{V}\mathbf{B}^T\mathbf{H}^T\mathbf{H}\mathbf{A}\frac{\partial\boldsymbol{\theta}^T}{\partial y},
\end{aligned}$$

and

$$\begin{aligned}
\mathbb{E}[\tilde{\mathcal{L}}_\theta \{\theta, w\} \mathbf{H}] &= \kappa\Delta\boldsymbol{\theta}\mathbf{A}^T - \left(\bar{u}\frac{\partial\boldsymbol{\theta}}{\partial x} + \bar{v}\frac{\partial\boldsymbol{\theta}}{\partial y} \right) \mathbf{A}^T - \left(\frac{\partial\bar{\theta}}{\partial x}\mathbf{U} + \frac{\partial\bar{\theta}}{\partial y}\mathbf{V} \right) \mathbf{B}^T \\
&\quad - \mathbb{E}\left[\mathbf{U}\mathbf{B}^T\mathbf{H}^T\mathbf{H}\mathbf{A}\frac{\partial\boldsymbol{\theta}^T}{\partial x}\mathbf{H} \right] - \mathbb{E}\left[\mathbf{V}\mathbf{B}^T\mathbf{H}^T\mathbf{H}\mathbf{A}\frac{\partial\boldsymbol{\theta}^T}{\partial y}\mathbf{H} \right] \\
&= \kappa\Delta\boldsymbol{\theta}\mathbf{A}^T - \left(\bar{u}\frac{\partial\boldsymbol{\theta}}{\partial x} + \bar{v}\frac{\partial\boldsymbol{\theta}}{\partial y} \right) \mathbf{A}^T - \left(\frac{\partial\bar{\theta}}{\partial x}\mathbf{U} + \frac{\partial\bar{\theta}}{\partial y}\mathbf{V} \right) \mathbf{B}^T \\
&\quad - \left[\left(u_i\frac{\partial\theta_j}{\partial x} + v_i\frac{\partial\theta_j}{\partial y} \right) B_{\alpha i}A_{\beta j}\mathfrak{Z}_{\alpha\beta\gamma}^{(\mathbf{H})} \right]_{1\times\gamma}.
\end{aligned}$$

Further, we plug in the above equality into the inner product. We get

$$\begin{aligned}
\left\langle \boldsymbol{\theta}^T, \mathbb{E} \left[\tilde{\mathcal{L}}_\theta \{ \theta, w \} \mathbf{H} \right] \right\rangle &= \kappa \langle \boldsymbol{\theta}^T, \Delta \boldsymbol{\theta} \rangle \mathbf{A}^T - \left\langle \boldsymbol{\theta}^T, \bar{u} \frac{\partial \boldsymbol{\theta}}{\partial x} + \bar{v} \frac{\partial \boldsymbol{\theta}}{\partial y} \right\rangle \mathbf{A}^T - \left\langle \boldsymbol{\theta}^T, \frac{\partial \bar{\theta}}{\partial x} \mathbf{U} + \frac{\partial \bar{\theta}}{\partial y} \mathbf{V} \right\rangle \mathbf{B}^T \\
&\quad - \left[\left\langle \theta_k, u_i \frac{\partial \theta_j}{\partial x} + v_i \frac{\partial \theta_j}{\partial y} \right\rangle B_{\alpha i} A_{\beta j} \mathfrak{T}_{\alpha \beta \gamma}^{(\mathbf{H})} \right]_{k \times \gamma} \\
&= \kappa \langle \boldsymbol{\theta}^T, \Delta \boldsymbol{\theta} \rangle \mathbf{A}^T - \left\langle \boldsymbol{\theta}^T, \bar{u} \frac{\partial \boldsymbol{\theta}}{\partial x} + \bar{v} \frac{\partial \boldsymbol{\theta}}{\partial y} \right\rangle \mathbf{A}^T - \left\langle \boldsymbol{\theta}^T, \frac{\partial \bar{\theta}}{\partial x} \mathbf{U} + \frac{\partial \bar{\theta}}{\partial y} \mathbf{V} \right\rangle \mathbf{B}^T \\
&\quad - \left[\mathfrak{T}_{ijk}^{(\boldsymbol{\theta})} B_{\alpha i} A_{\beta j} \mathfrak{T}_{\alpha \beta \gamma}^{(\mathbf{H})} \right]_{k \times \gamma},
\end{aligned}$$

where the third-order m_w -by- m_θ -by- m_θ tensor

$$\mathfrak{T}^{(\boldsymbol{\theta})} = \left(\left\langle \theta_k, u_i \frac{\partial \theta_j}{\partial x} + v_i \frac{\partial \theta_j}{\partial y} \right\rangle \right)_{ijk}.$$

From eqn. (6.5), we have the DyBO-gPC formulation for the temperature component,

$$\begin{aligned}
\frac{\partial \bar{\theta}}{\partial t} &= \kappa \Delta \bar{\theta} - \left(\bar{u} \frac{\partial \bar{\theta}}{\partial x} + \bar{v} \frac{\partial \bar{\theta}}{\partial y} \right) - \mathbf{U} \mathbf{B}^T \mathbf{A} \frac{\partial \boldsymbol{\theta}^T}{\partial x} - \mathbf{V} \mathbf{B}^T \mathbf{A} \frac{\partial \boldsymbol{\theta}^T}{\partial y}, \\
\frac{\partial \boldsymbol{\theta}}{\partial t} &= -\boldsymbol{\theta} \mathbf{D}_\theta^T + \kappa \Delta \boldsymbol{\theta} - \left(\bar{u} \frac{\partial \boldsymbol{\theta}}{\partial x} + \bar{v} \frac{\partial \boldsymbol{\theta}}{\partial y} \right) - \left(\frac{\partial \bar{\theta}}{\partial x} \mathbf{U} + \frac{\partial \bar{\theta}}{\partial y} \mathbf{V} \right) \mathbf{B}^T \mathbf{A} \\
&\quad - \left[\left(u_i \frac{\partial \theta_j}{\partial x} + v_i \frac{\partial \theta_j}{\partial y} \right) B_{\alpha i} A_{\beta j} A_{\gamma k} \mathfrak{T}_{\alpha \beta \gamma}^{(\mathbf{H})} \right]_{1 \times k}, \\
\frac{d\mathbf{A}}{dt} &= \mathbf{A} \left(-\mathbf{C}_\theta^T + \kappa \langle \Delta \boldsymbol{\theta}^T, \boldsymbol{\theta} \rangle \boldsymbol{\Lambda}_\theta^{-1} - \left\langle \bar{u} \frac{\partial \boldsymbol{\theta}^T}{\partial x} + \bar{v} \frac{\partial \boldsymbol{\theta}^T}{\partial y}, \boldsymbol{\theta} \right\rangle \boldsymbol{\Lambda}_\theta^{-1} \right) \\
&\quad - \mathbf{B} \left\langle \frac{\partial \bar{\theta}}{\partial x} \mathbf{U}^T + \frac{\partial \bar{\theta}}{\partial y} \mathbf{V}^T, \boldsymbol{\theta} \right\rangle \boldsymbol{\Lambda}_\theta^{-1} - \left[\mathfrak{T}_{ijk}^{(\boldsymbol{\theta})} B_{\alpha i} A_{\beta j} \mathfrak{T}_{\alpha \beta \gamma}^{(\mathbf{H})} \right]_{\gamma \times k} \boldsymbol{\Lambda}_\theta^{-1},
\end{aligned}$$

where matrices \mathbf{C}_θ and \mathbf{D}_θ can be solved via eqn. (2.22) from $G_{*\theta}$,

$$\begin{aligned}
\boldsymbol{\Lambda}_\theta G_{*\theta} &= \kappa \langle \boldsymbol{\theta}^T, \Delta \boldsymbol{\theta} \rangle - \left\langle \boldsymbol{\theta}^T, \bar{u} \frac{\partial \boldsymbol{\theta}}{\partial x} + \bar{v} \frac{\partial \boldsymbol{\theta}}{\partial y} \right\rangle - \left\langle \boldsymbol{\theta}^T, \frac{\partial \bar{\theta}}{\partial x} \mathbf{U} + \frac{\partial \bar{\theta}}{\partial y} \mathbf{V} \right\rangle \mathbf{B}^T \mathbf{A} \\
&\quad - \left[\mathfrak{T}_{ijk}^{(\boldsymbol{\theta})} B_{\alpha i} A_{\beta j} A_{\gamma l} \mathfrak{T}_{\alpha \beta \gamma}^{(\mathbf{H})} \right]_{k \times l}.
\end{aligned}$$

Next, we derive the DyBO equations for the vorticity component. Substituting the KL expansion into eqn. (6.9a), we have

$$\begin{aligned}
\mathcal{L}_w \{\theta, w\} &= -(\bar{u} + \mathbf{U}\mathbf{B}^T\mathbf{H}^T) \left(\frac{\partial \bar{w}}{\partial x} + \frac{\partial \mathbf{W}}{\partial x} \mathbf{B}^T \mathbf{H}^T \right) - (\bar{v} + \mathbf{V}\mathbf{B}^T\mathbf{H}^T) \left(\frac{\partial \bar{w}}{\partial y} + \frac{\partial \mathbf{W}}{\partial y} \mathbf{B}^T \mathbf{H}^T \right) \\
&\quad + \nu (\Delta \bar{w} + \Delta \mathbf{W} \mathbf{B}^T \mathbf{H}^T) + \mathbf{F} \mathbf{H}^T + \mu g \frac{\partial \bar{\theta}}{\partial x} + \mu g \frac{\partial \boldsymbol{\theta}}{\partial x} \mathbf{A}^T \mathbf{H}^T \\
&= \left(\nu \Delta \bar{w} - \bar{u} \frac{\partial \bar{w}}{\partial x} - \bar{v} \frac{\partial \bar{w}}{\partial y} + \mu g \frac{\partial \bar{\theta}}{\partial x} \right) \\
&\quad + \left[\nu \Delta \mathbf{W} - \left(\bar{u} \frac{\partial \mathbf{W}}{\partial x} + \bar{v} \frac{\partial \mathbf{W}}{\partial y} \right) - \left(\frac{\partial \bar{w}}{\partial x} \mathbf{U} + \frac{\partial \bar{w}}{\partial y} \mathbf{V} \right) \right] \mathbf{B}^T \mathbf{H}^T + \mu g \frac{\partial \boldsymbol{\theta}}{\partial x} \mathbf{A}^T \mathbf{H}^T \\
&\quad - \mathbf{U} \mathbf{B}^T \mathbf{H}^T \mathbf{H} \mathbf{B} \frac{\partial \mathbf{W}^T}{\partial x} - \mathbf{V} \mathbf{B}^T \mathbf{H}^T \mathbf{H} \mathbf{B} \frac{\partial \mathbf{W}^T}{\partial y} + \mathbf{F} \mathbf{H}^T.
\end{aligned}$$

Taking expectations on both sides, we have

$$\begin{aligned}
\mathbb{E} [\mathcal{L}_w \{\theta, w\}] &= \left(\nu \Delta \bar{w} - \bar{u} \frac{\partial \bar{w}}{\partial x} - \bar{v} \frac{\partial \bar{w}}{\partial y} + \mu g \frac{\partial \bar{\theta}}{\partial x} \right) - \mathbf{U} \frac{\partial \mathbf{W}^T}{\partial x} - \mathbf{V} \frac{\partial \mathbf{W}^T}{\partial y}, \\
\tilde{\mathcal{L}}_w \{\theta, w\} &= \left[\nu \Delta \mathbf{W} - \left(\bar{u} \frac{\partial \mathbf{W}}{\partial x} + \bar{v} \frac{\partial \mathbf{W}}{\partial y} \right) - \left(\frac{\partial \bar{w}}{\partial x} \mathbf{U} + \frac{\partial \bar{w}}{\partial y} \mathbf{V} \right) \right] \mathbf{B}^T \mathbf{H}^T + \mu g \frac{\partial \boldsymbol{\theta}}{\partial x} \mathbf{A}^T \mathbf{H}^T \\
&\quad + \mathbf{U} \frac{\partial \mathbf{W}^T}{\partial x} + \mathbf{V} \frac{\partial \mathbf{W}^T}{\partial y} - \mathbf{U} \mathbf{B}^T \mathbf{H}^T \mathbf{H} \mathbf{B} \frac{\partial \mathbf{W}^T}{\partial x} - \mathbf{V} \mathbf{B}^T \mathbf{H}^T \mathbf{H} \mathbf{B} \frac{\partial \mathbf{W}^T}{\partial y} + \mathbf{F} \mathbf{H}^T,
\end{aligned}$$

and

$$\begin{aligned}
\mathbb{E} \left[\tilde{\mathcal{L}}_w \{\theta, w\} \mathbf{H} \right] &= \left[\nu \Delta \mathbf{W} - \left(\bar{u} \frac{\partial \mathbf{W}}{\partial x} + \bar{v} \frac{\partial \mathbf{W}}{\partial y} \right) - \left(\frac{\partial \bar{w}}{\partial x} \mathbf{U} + \frac{\partial \bar{w}}{\partial y} \mathbf{V} \right) \right] \mathbf{B}^T + \mu g \frac{\partial \boldsymbol{\theta}}{\partial x} \mathbf{A}^T \\
&\quad - \mathbb{E} \left[\mathbf{U} \mathbf{B}^T \mathbf{H}^T \mathbf{H} \mathbf{B} \frac{\partial \mathbf{W}^T}{\partial x} \mathbf{H} \right] - \mathbb{E} \left[\mathbf{V} \mathbf{B}^T \mathbf{H}^T \mathbf{H} \mathbf{B} \frac{\partial \mathbf{W}^T}{\partial y} \mathbf{H} \right] + \mathbf{F} \\
&= \left[\nu \Delta \mathbf{W} - \left(\bar{u} \frac{\partial \mathbf{W}}{\partial x} + \bar{v} \frac{\partial \mathbf{W}}{\partial y} \right) - \left(\frac{\partial \bar{w}}{\partial x} \mathbf{U} + \frac{\partial \bar{w}}{\partial y} \mathbf{V} \right) \right] \mathbf{B}^T + \mu g \frac{\partial \boldsymbol{\theta}}{\partial x} \mathbf{A}^T \\
&\quad - \left[\left(u_i \frac{\partial w_j}{\partial x} + v_i \frac{\partial w_j}{\partial y} \right) B_{\alpha i} B_{\beta j} \mathfrak{Z}_{\alpha \beta \gamma}^{(\mathbf{H})} \right]_{1 \times \gamma} + \mathbf{F}.
\end{aligned}$$

Plugging the above equality into the inner product gives

$$\begin{aligned}
\left\langle \mathbf{W}^T, \mathbb{E} \left[\tilde{\mathcal{L}}_w \{ \theta, w \} \mathbf{H} \right] \right\rangle &= \left\langle \mathbf{W}^T, \nu \Delta \mathbf{W} - \left(\bar{u} \frac{\partial \mathbf{W}}{\partial x} + \bar{v} \frac{\partial \mathbf{W}}{\partial y} \right) - \left(\frac{\partial \bar{w}}{\partial x} \mathbf{U} + \frac{\partial \bar{w}}{\partial y} \mathbf{V} \right) \right\rangle \mathbf{B}^T \\
&\quad + \left\langle \mathbf{W}^T, \mu g \frac{\partial \theta}{\partial x} \right\rangle \mathbf{A}^T - \left[\left\langle w_k, u_i \frac{\partial w_j}{\partial x} + v_i \frac{\partial w_j}{\partial y} \right\rangle B_{\alpha i} B_{\beta j} \mathfrak{T}_{\alpha \beta \gamma}^{(\mathbf{H})} \right]_{k \times \gamma} \\
&\quad + \langle \mathbf{W}^T, \mathbf{F} \rangle \\
&= \left\langle \mathbf{W}^T, \nu \Delta \mathbf{W} - \left(\bar{u} \frac{\partial \mathbf{W}}{\partial x} + \bar{v} \frac{\partial \mathbf{W}}{\partial y} \right) - \left(\frac{\partial \bar{w}}{\partial x} \mathbf{U} + \frac{\partial \bar{w}}{\partial y} \mathbf{V} \right) \right\rangle \mathbf{B}^T \\
&\quad + \left\langle \mathbf{W}^T, \mu g \frac{\partial \theta}{\partial x} \right\rangle \mathbf{A}^T - \left[\mathfrak{T}_{ijk}^{(\mathbf{W})} B_{\alpha i} B_{\beta j} \mathfrak{T}_{\alpha \beta \gamma}^{(\mathbf{H})} \right]_{k \times \gamma} + \langle \mathbf{W}^T, \mathbf{F} \rangle,
\end{aligned}$$

where the third-order m_w -by- m_w -by- m_w tensor

$$\mathfrak{T}^{(\mathbf{W})} = \left(\left\langle w_k, u_i \frac{\partial w_j}{\partial x} + v_i \frac{\partial w_j}{\partial y} \right\rangle \right)_{ijk}. \quad (\text{E.2})$$

From eqn. (6.5), we have the DyBO-gPC formulation for the vorticity component,

$$\begin{aligned}
\frac{\partial \bar{w}}{\partial t} &= \left(\nu \Delta \bar{w} - \bar{u} \frac{\partial \bar{w}}{\partial x} - \bar{v} \frac{\partial \bar{w}}{\partial y} + \mu g \frac{\partial \bar{\theta}}{\partial x} \right) - \mathbf{U} \frac{\partial \mathbf{W}^T}{\partial x} - \mathbf{V} \frac{\partial \mathbf{W}^T}{\partial y}, \\
\frac{\partial \mathbf{W}}{\partial t} &= -\mathbf{W} \mathbf{D}_w^T + \left[\nu \Delta \mathbf{W} - \left(\bar{u} \frac{\partial \mathbf{W}}{\partial x} + \bar{v} \frac{\partial \mathbf{W}}{\partial y} \right) - \left(\frac{\partial \bar{w}}{\partial x} \mathbf{U} + \frac{\partial \bar{w}}{\partial y} \mathbf{V} \right) \right] + \mu g \frac{\partial \theta}{\partial x} \mathbf{A}^T \mathbf{B} \\
&\quad \left[\left(u_i \frac{\partial w_j}{\partial x} + v_i \frac{\partial w_j}{\partial y} \right) B_{\alpha i} B_{\beta j} B_{\gamma k} \mathfrak{T}_{\alpha \beta \gamma}^{(\mathbf{H})} \right]_{1 \times k} + \mathbf{F} \mathbf{B}, \\
\frac{d\mathbf{B}}{dt} &= \mathbf{B} \left(-\mathbf{C}_w^T + \left\langle \nu \Delta \mathbf{W}^T - \left(\bar{u} \frac{\partial \mathbf{W}^T}{\partial x} + \bar{v} \frac{\partial \mathbf{W}^T}{\partial y} \right) - \left(\frac{\partial \bar{w}}{\partial x} \mathbf{U}^T + \frac{\partial \bar{w}}{\partial y} \mathbf{V}^T \right), \mathbf{W} \right\rangle \Lambda_{\mathbf{W}}^{-1} \right) \\
&\quad + \mu g \mathbf{A} \left\langle \frac{\partial \theta^T}{\partial x}, \mathbf{W} \right\rangle \Lambda_{\mathbf{W}}^{-1} - \left[\mathfrak{T}_{ijk}^{(\mathbf{W})} B_{\alpha i} B_{\beta j} \mathfrak{T}_{\alpha \beta \gamma}^{(\mathbf{H})} \right]_{\gamma \times k} \Lambda_{\mathbf{W}}^{-1} + \langle \mathbf{F}^T, \mathbf{W} \rangle \Lambda_{\mathbf{W}}^{-1},
\end{aligned}$$

where matrices \mathbf{C}_w and \mathbf{D}_w can be solved via eqn. (2.22) from G_{*w} ,

$$\begin{aligned} \Lambda_{\mathbf{W}} G_{*w} = & \left\langle \mathbf{W}^T, \nu \Delta \mathbf{W} - \left(\bar{u} \frac{\partial \mathbf{W}}{\partial x} + \bar{v} \frac{\partial \mathbf{W}}{\partial y} \right) - \left(\frac{\partial \bar{w}}{\partial x} \mathbf{U} + \frac{\partial \bar{w}}{\partial y} \mathbf{V} \right) \right\rangle \\ & + \left\langle \mathbf{W}^T, \mu g \frac{\partial \boldsymbol{\theta}}{\partial x} \right\rangle \mathbf{A}^T \mathbf{B} - \left[\mathfrak{T}_{ijk}^{(\mathbf{W})} B_{\alpha i} B_{\beta j} B_{\gamma l} \mathfrak{T}_{\alpha \beta \gamma}^{(\mathbf{H})} \right]_{k \times l} + \langle \mathbf{W}^T, \mathbf{F} \rangle \mathbf{B}. \end{aligned}$$

Combining the above discussion and using the generalized material derivative (D.9), we arrive at the DyBO-gPC formulation for SNSE (6.9),

$$\frac{\partial \bar{\boldsymbol{\theta}}}{\partial t} = \kappa \Delta \bar{\boldsymbol{\theta}} - \mathfrak{D}_{(\bar{u}, \bar{v})} \bar{\boldsymbol{\theta}} - \mathfrak{D}_{(\mathbf{U}, \mathbf{V})} (\boldsymbol{\theta} \mathbf{A}^T \mathbf{B}), \quad (\text{E.3a})$$

$$\begin{aligned} \frac{\partial \boldsymbol{\theta}}{\partial t} = & -\boldsymbol{\theta} \mathbf{D}_{\boldsymbol{\theta}}^T + \kappa \Delta \boldsymbol{\theta} - \mathfrak{D}_{(\bar{u}, \bar{v})} \boldsymbol{\theta} - \mathfrak{D}_{(\mathbf{U}, \mathbf{V})} \bar{\boldsymbol{\theta}} \mathbf{B}^T \mathbf{A} \\ & - \left[\mathfrak{D}_{(u_i, v_i)} \theta_j B_{\alpha i} A_{\beta j} A_{\gamma k} \mathfrak{T}_{\alpha \beta \gamma}^{(\mathbf{H})} \right]_{1 \times k}, \end{aligned} \quad (\text{E.3b})$$

$$\begin{aligned} \frac{d\mathbf{A}}{dt} = & \mathbf{A} \left(-\mathbf{C}_{\boldsymbol{\theta}}^T + \kappa \langle \Delta \boldsymbol{\theta}^T, \boldsymbol{\theta} \rangle \Lambda_{\boldsymbol{\theta}}^{-1} - \langle (\mathfrak{D}_{(\bar{u}, \bar{v})} \boldsymbol{\theta})^T, \boldsymbol{\theta} \rangle \Lambda_{\boldsymbol{\theta}}^{-1} \right) \\ & - \mathbf{B} \langle (\mathfrak{D}_{(\mathbf{U}, \mathbf{V})} \bar{\boldsymbol{\theta}})^T, \boldsymbol{\theta} \rangle \Lambda_{\boldsymbol{\theta}}^{-1} - \left[\mathfrak{T}_{ijk}^{(\boldsymbol{\theta})} B_{\alpha i} A_{\beta j} \mathfrak{T}_{\alpha \beta \gamma}^{(\mathbf{H})} \right]_{\gamma \times k} \Lambda_{\boldsymbol{\theta}}^{-1}, \end{aligned} \quad (\text{E.3c})$$

$$\frac{\partial \bar{w}}{\partial t} = \left(\nu \Delta \bar{w} - \mathfrak{D}_{(\bar{u}, \bar{v})} \bar{w} + \mu g \frac{\partial \bar{\boldsymbol{\theta}}}{\partial x} \right) - \mathfrak{D}_{(\mathbf{U}, \mathbf{V})} \mathbf{W}, \quad (\text{E.3d})$$

$$\begin{aligned} \frac{\partial \mathbf{W}}{\partial t} = & -\mathbf{W} \mathbf{D}_w^T + \left[\nu \Delta \mathbf{W} - \mathfrak{D}_{(\bar{u}, \bar{v})} \mathbf{W} - \mathfrak{D}_{(\mathbf{U}, \mathbf{V})} \bar{w} \right] + \mu g \frac{\partial \boldsymbol{\theta}}{\partial x} \mathbf{A}^T \mathbf{B} \\ & - \left[\mathfrak{D}_{(u_i, v_i)} w_j B_{\alpha i} B_{\beta j} B_{\gamma k} \mathfrak{T}_{\alpha \beta \gamma}^{(\mathbf{H})} \right]_{1 \times k} + \mathbf{F} \mathbf{B}, \end{aligned} \quad (\text{E.3e})$$

$$\begin{aligned} \frac{d\mathbf{B}}{dt} = & \mathbf{B} \left(-\mathbf{C}_w^T + \langle \nu \Delta \mathbf{W}^T - (\mathfrak{D}_{(\bar{u}, \bar{v})} \mathbf{W})^T - (\mathfrak{D}_{(\mathbf{U}, \mathbf{V})} \bar{w})^T, \mathbf{W} \rangle \Lambda_{\mathbf{W}}^{-1} \right) \\ & + \mu g \mathbf{A} \left\langle \frac{\partial \boldsymbol{\theta}^T}{\partial x}, \mathbf{W} \right\rangle \Lambda_{\mathbf{W}}^{-1} - \left[\mathfrak{T}_{ijk}^{(\mathbf{W})} B_{\alpha i} B_{\beta j} \mathfrak{T}_{\alpha \beta \gamma}^{(\mathbf{H})} \right]_{\gamma \times k} \Lambda_{\mathbf{W}}^{-1} + \langle \mathbf{F}^T, \mathbf{W} \rangle \Lambda_{\mathbf{W}}^{-1}, \end{aligned} \quad (\text{E.3f})$$

where matrices $\mathbf{C}_{\boldsymbol{\theta}}$ and $\mathbf{D}_{\boldsymbol{\theta}}$ can be solved via eqn. (2.22) from $G_{*\boldsymbol{\theta}}$,

$$\begin{aligned} \Lambda_{\boldsymbol{\theta}} G_{*\boldsymbol{\theta}} = & \kappa \langle \boldsymbol{\theta}^T, \Delta \boldsymbol{\theta} \rangle - \langle \boldsymbol{\theta}^T, \mathfrak{D}_{(\bar{u}, \bar{v})} \boldsymbol{\theta} \rangle - \langle \boldsymbol{\theta}^T, \mathfrak{D}_{(\mathbf{U}, \mathbf{V})} \bar{\boldsymbol{\theta}} \rangle \mathbf{B}^T \mathbf{A} \\ & - \left[\mathfrak{T}_{ijk}^{(\boldsymbol{\theta})} B_{\alpha i} A_{\beta j} A_{\gamma l} \mathfrak{T}_{\alpha \beta \gamma}^{(\mathbf{H})} \right]_{k \times l}, \end{aligned} \quad (\text{E.4})$$

and matrices \mathbf{C}_w and \mathbf{D}_w can be solved via eqn. (2.22) from G_{*w} ,

$$\begin{aligned} \mathbf{\Lambda}_w G_{*w} = & \langle \mathbf{W}^T, \nu \Delta \mathbf{W} - \mathcal{D}_{(\bar{u}, \bar{v})} \mathbf{W} - \mathcal{D}_{(\mathbf{U}, \mathbf{V})} \bar{w} \rangle \\ & + \left\langle \mathbf{W}^T, \mu g \frac{\partial \theta}{\partial x} \right\rangle \mathbf{A}^T \mathbf{B} - \left[\tilde{\mathfrak{z}}_{ijk}^{(\mathbf{W})} B_{\alpha i} B_{\beta j} B_{\gamma l} \tilde{\mathfrak{z}}_{\alpha \beta \gamma}^{(\mathbf{H})} \right]_{k \times l} + \langle \mathbf{W}^T, \mathbf{F} \rangle \mathbf{B}. \end{aligned}$$

Bibliography

- [1] H. Abdi and L.J. Williams. Principle component analysis. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2(4):433–459, 2010.
- [2] M. Adams, A. Lashgari, B. Li, M. McKerns, J. Mihaly, M. Ortiz, H. Owhadi, A.J. Rosakis, M. Stalzer, and T.J. Sullivan. Rigorous model-based uncertainty quantification with application to terminal ballistics — Part I. Systems with uncontrollable inputs and large scatter. *Journal of the Mechanics and Physics of Solids*, 60(5):983–1001, 2012.
- [3] M. Adams, A. Lashgari, B. Li, M. McKerns, J. Mihaly, M. Ortiz, H. Owhadi, A.J. Rosakis, M. Stalzer, and T.J. Sullivan. Rigorous model-based uncertainty quantification with application to terminal ballistics — Part II. Systems with uncontrollable inputs and large scatter. *Journal of the Mechanics and Physics of Solids*, 60(5):1002–1019, 2012.
- [4] J. Anderson. *Computational Fluid Dynamics*. McGraw-Hill, 1st edition, 1995.
- [5] R. Askey and J. Wilson. Some Basic Hypergeometric Orthogonal Polynomials That Generalize Jacobi Polynomials. *Memoirs of the American Mathematical Society*, 54(319):1–55, 1985.
- [6] B.V. Asokan and N. Zabaras. A stochastic variational multiscale method for diffusion in heterogeneous random media. *Journal of Computational Physics*, 218(2):654–676, 2006.

- [7] I. Babuska, F. Nobile, and R. Tempone. A stochastic collocation method for elliptic partial differential equations with random input data. *SIAM Journal on Numerical Analysis*, 45(3):1005, 2007.
- [8] A. Barth, C. Schwab, and N. Zollinger. Multi-level Monte Carlo finite element method for elliptic PDEs with stochastic coefficients. *Numerische Mathematik*, pages 1–39, April 2011.
- [9] J. Bec, U. Frisch, and WEK Khanin. Kicked Burgers turbulence. *Journal of Fluid Mechanics*, 416:239–267, 2000.
- [10] J. Bec and WEK Khanin. Burgers turbulence. *Physics Reports — Review Section of Physics Letters*, 447(1–2):1–68, 2007.
- [11] Ch. Becco, N. Vandewalle, J. Delcourt, and P. Poncin. Experimental evidences of a structural and dynamical transition in fish school. *Physica A*, 367:487–493, July 2006.
- [12] G. Berkooz, P. Holmes, and J.L. Lumley. The proper orthogonal decomposition in the analysis of turbulent flows. *Annual Review of Fluid Mechanics*, 25:539–575, 1993.
- [13] L. Bertini and M. Ponsiglione. A variational approach to the stationary solutions of the Burgers equation. *SIAM Journal on Mathematical Analysis*, 44(2):682–698, 2012.
- [14] M. Bieri and C. Schwab. Sparse high order FEM for elliptic sPDEs. *Computer Methods in Applied Mechanics and Engineering*, 198(13–14):1149–1170, March 2009.
- [15] J.P. Bouchaud, M. Mezard, and G. Parisi. Scaling and intermittency in Burgers turbulence. *Physical Review E*, 52(4):3656–3674, 1995.
- [16] J.P. Boyd. *Chebyshev and Fourier Spectral Methods*. Dover Publications, 2nd edition, 2001.

- [17] P. Bratley and B.L. Fox. ALGORITHM 659: implementing Sobol's quasirandom sequence generator. *ACM Transactions on Mathematical Software*, 14(1):88–100, March 1988.
- [18] H.J. Bungartz and M. Griebel. Sparse grids. *Acta Numerica*, 13:147–269, 2004.
- [19] R.H. Cameron and W.T. Martin. The orthogonal development of non-linear functionals in series of Fourier-Hermite Functionals. *Annals of Mathematics*, 48(2):385–392, 1947.
- [20] E.J. Candes, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, 52(2):489–509, 2006.
- [21] H.D. Ceniceros and Thomas Y.Z. Hou. An efficient dynamically adaptive mesh for potentially singular solutions. *Journal of Computational Physics*, 172(2):609–639, 2001.
- [22] D.H. Chambers, R.J. Adrian, P. Moin, D.S. Stewart, and H.J. Sung. Karhunen-Loeve expansion of Burgers model of turbulence. *Physics of Fluids*, 31(9):2573–2582, 1988.
- [23] V. Chandrasekaran, S. Sanghavi, P.A. Parrilo, and A.S. Willsky. Sparse and low-rank matrix decompositions. In *47th Annual Allerton Conference on Communication, Control and Computing*, pages 962–967, Monticello, IL, 2009. IEEE.
- [24] V. Chandrasekaran, S. Sanghavi, P.A. Parrilo, and A.S. Willsky. Rank-sparsity incoherence for matrix decomposition. *SIAM Journal on Optimization*, 21(2):572–596, 2011.
- [25] C.-C. Chu, I.G. Graham, and Thomas Y.Z. Hou. A new multiscale finite element method for high-contrast elliptic interface problems. *Math. Comput.*, 79(272):1915–1955, 2010.
- [26] K.A. Cliffe, M.B. Giles, R. Scheichl, and A.L. Teckentrup. Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients. *Computing and Visualization in Science*, 14(1):3–15, 2011.

- [27] B.J. Debusschere, H.N. Najm, Pebay P.P., O.M. Knio, R.G. Ghanem, and O. Le Maitre. Numerical Challenges in the Use of Polynomial Chaos Representations for Stochastic Processes. *SIAM Journal on Scientific Computing*, 26(2):698, 2004.
- [28] J.L. Demuth, B.H. Morrow, and J.K. Lazo. Weather forecast uncertainty information, An exploratory study with Broadcast Meteorologists. *Bulletin of the American Meteorological Society*, 90(11):1614–1618, 2009.
- [29] A. Doostan and H. Owhadi. A non-adapted sparse approximation of PDEs with stochastic inputs. *Journal of Computational Physics*, 230(8):3015–3034, 2011.
- [30] W. E, K. Konstantin, A. Mazel, and Y. Sinai. Probability distribution functions for the random forced burgers equation. *Phys. Rev. Lett.*, 78(10):1904–1907, 1997.
- [31] W. E and Y. Sinai. Recent results on mathematical and statistical hydrodynamics. *Russian Mathematical Surveys*, 55(4):635–666, 2000.
- [32] Y. Efendiev, Thomas Y.Z. Hou, and W. Luo. Preconditioning Markov Chain Monte Carlo simulations using coarse-scale models. *SIAM Journal on Scientific Computing*, 28(2):776, 2006.
- [33] J. Foo, X. Wan, and G.E. Karniadakis. The multi-element probabilistic collocation method (ME-PCM): Error analysis and applications. *Journal of Computational Physics*, 227(22):9572–9595, 2008.
- [34] J.N. Franklin. *Matrix Theory (Dover Books on Mathematics)*. Dover Publications, 1 edition, 2000.
- [35] J. Galvis and Y. Efendiev. Domain decomposition preconditioners for multiscale flows in high-contrast media. *Multiscale Modeling & Simulation*, 8(4):1461–1483, 2010.

- [36] J. Galvis and Y. Efendiev. Domain decomposition preconditioners for multiscale flows in high-contrast media: Reduced dimension coarse spaces. *Multiscale Modeling & Simulation*, 8(5):1621–1644, 2010.
- [37] B. Ganapathysubramanian and N. Zabaras. A stochastic multiscale framework for modeling flow through random heterogeneous porous media. *Journal of Computational Physics*, 228(2):591–618, 2009.
- [38] W. Gautschi. On generating orthogonal polynomials. *SIAM Journal on Scientific and Statistical Computing*, 3(3):289–317, 1982.
- [39] A. Genz and B.D. Keister. Fully symmetric interpolatory rules for multiple integrals over infinite regions with Gaussian weight. *Journal of Computational and Applied Mathematics*, 71(2):299–309, 1996.
- [40] T. Gerstner and M. Griebel. Numerical integration using sparse grids. *Numerical Algorithms*, 18:209–232, 1998.
- [41] R.G. Ghanem and P.D. Spanos. *Stochastic Finite Elements: A Spectral Approach*. Dover Publications, 1st edition, 2003.
- [42] M.B. Giles. Multilevel Monte Carlo Path Simulation. *Operations Research*, 56(3):607–617, May 2008.
- [43] M.B. Giles. Multilevel Monte Carlo for Basket options. In *Proceedings of the 2009 Winter Simulation Conference*, pages 1263–1270, 2009.
- [44] G.H. Golub. *Matrix Computations (Johns Hopkins Studies in Mathematical Sciences)*. Johns Hopkins University Press, 3rd edition, 1996.

- [45] I.G. Graham, F.Y. Kuo, D. Nuyens, R. Scheichl, and I.H. Sloan. Quasi-Monte Carlo methods for elliptic PDEs with random coefficients and applications. *Journal of Computational Physics*, 230(10):3694–3668, February 2011.
- [46] M. Griebel. Adaptive sparse grid multilevel methods for elliptic PDEs based on finite differences. *Computing*, 61(2):151–179, June 1998.
- [47] W. Hackbusch. A Sparse Matrix Arithmetic Based on H-Matrices. Part I: Introduction to H-Matrices. *Computing*, 62(2):89–108, April 1999.
- [48] W. Hackbusch and B.N. Khoromskij. A sparse H-matrix arithmetic. Part II: Application to multi-dimensional problems. *Computing*, 64(1):21–47, 2000.
- [49] E. Hairer, C. Lubich, and G. Wanner. *Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations (Springer Series in Computational Mathematics)*. Springer, 1st edition, 2006.
- [50] F. Heiss and V. Winschel. Likelihood approximation by numerical integration on sparse grids. *Journal of Econometrics*, 144(1):62–80, 2008.
- [51] W.D. Henshaw, H.O. Kreiss, and J. Ystrom. Numerical experiments on the interaction between the large- and small-scale motions of the Navier-Stokes equations. *Multiscale Modeling & Simulation*, 1(1):119–149, 2003.
- [52] H. Holden, B. Øksendal, J. Ubøe, and T. Zhang. *Stochastic Partial Differential Equations: A Modeling, White Noise Functional Approach (Universitext)*. Springer, 2009.
- [53] I. Hosokawa and K. Yamamoto. Turbulence in randomly force, one-dimensional Burgers flow. *Journal of Statistical Physics*, 13(3):245–272, 1975.

- [54] Thomas Y.Z. Hou and R. Li. Blowup or non-blowup? The interplay between theory and numerics. *Physica A*, 237(14–17):1937–1944, 2008.
- [55] Thomas Y.Z. Hou, W. Luo, B. Rozovskii, and H. Zhou. Wiener Chaos expansions and numerical solutions of randomly forced equations of fluid mechanics. *Journal of Computational Physics*, 216(2):687–706, 2006.
- [56] Thomas Y.Z. Hou and Z.Q. Shi. Adaptive data analysis via sparse time-frequency representation. *Advances in Adaptive Data Analysis*, 2(1–2):1–28, 2011.
- [57] Thomas Y.Z. Hou and X.H. Wu. A multiscale finite element method for elliptic problems in composite materials and porous media. *Journal of Computational Physics*, 134(1):169–189, 1997.
- [58] Thomas Y.Z. Hou, D.P. Yang, and H.Y. Ran. Multiscale Analysis and Computation for the Three-Dimensional Incompressible Navier-Stokes Equations. *Multiscale Modeling & Simulation*, 6(4):1317, 2008.
- [59] S. Joe and F.Y. Kuo. Remark on algorithm 659. *ACM Transactions on Mathematical Software*, 29(1):49–57, March 2003.
- [60] S. Joe and F.Y. Kuo. Constructing Sobol Sequences with Better Two-Dimensional Projections. *SIAM Journal on Scientific Computing*, 30(5):2635, August 2008.
- [61] M. Kardar, G. Parisi, and Y.C. Zhang. Dynamic scaling of growing interfaces. *Phys. Rev. Lett.*, 56(9):889–892, 1986.
- [62] M.C. Kennedy and A.O. Hagan. Bayesian calibration of computer models. *Journal of the Royal Statistical Society Series B — Statistical Methodology*, 63(3):425–464, 2001.

- [63] WEK Khanin, A. Mazel, and Y. Sinai. Invariant measures for Burgers equation with stochastic forcing. *Annals of Mathematics*, 151(3):877–960, 2000.
- [64] L.D. Kiel. *Chaos Theory in the Social Sciences, Foundations and Applications*. University of Michigan Press, 1996.
- [65] M. Kirby. Application of the Karhunen-Loeve procedure for the characterization of human face. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(1):103–108, 1990.
- [66] P.E. Kloeden and E. Platen. *Numerical Solution of Stochastic Differential Equations (Stochastic Modelling and Applied Probability)*. Springer, 2000.
- [67] P.E. Kloeden, E. Platen, and H. Schurz. *Numerical Solution of SDE Through Computer Experiments (Universitext)*. Springer, 2003.
- [68] D.J. Knezevic and A.T. Patera. A certified reduced basis method for the Fokker-Planck equation of dilute polymeric fluids: FENE Dumbbells in extensional flow. *SIAM Journal on Scientific Computing*, 32(2):793, 2010.
- [69] O. Le Maitre. A Stochastic Projection Method for Fluid Flow I. Basic Formulation. *Journal of Computational Physics*, 173(2):481–511, November 2001.
- [70] O. Le Maitre. A stochastic projection method for fluid flow II. random process. *Journal of Computational Physics*, 181(1):9–44, September 2002.
- [71] O. Le Maitre. Multi-resolution analysis of Wiener-type uncertainty propagation schemes. *Journal of Computational Physics*, 197(2):502–531, 2004.
- [72] O. Le Maitre. Uncertainty propagation using Wiener-Haar expansions. *Journal of Computational Physics*, 197(1):28–57, 2004.

- [73] O. Le Maitre. A Newton method for the resolution of steady stochastic Navier-Stokes equations. *Computers & Fluids*, 38(8):1566–1579, 2009.
- [74] M. Ledoux. *The Concentration of Measure Phenomenon (Mathematical Surveys & Monographs)*. American Mathematical Society, 2005.
- [75] L.Y. Li, H.A. Tchelepi, and D.X. Zhang. Perturbation-based moment equation approach for flow in heterogeneous porous media: applicability range and analysis of high-order terms. *Journal of Computational Physics*, 188(1):296–317, June 2003.
- [76] L.J. Lucas, H. Owhadi, and M. Ortiz. Rigorous verification, validation, uncertainty quantification and certification through concentration-of-measure inequalities. *Computer Methods in Applied Mechanics and Engineering*, 197(51-52):4591–4609, October 2008.
- [77] W. Luo. *Wiener chaos expansion and numerical solutions of stochastic partial differential equations*. PhD thesis, California Institute of Technology, 2006.
- [78] X. MA and N. Zabarar. An adaptive hierarchical sparse grid collocation algorithm for the solution of stochastic differential equations. *Journal of Computational Physics*, 228(8):3084–3113, 2009.
- [79] H. Matthies and A. Keese. Galerkin methods for linear and nonlinear elliptic stochastic partial differential equations. *Computer Methods in Applied Mechanics and Engineering*, 194(12–16):1295–1331, April 2005.
- [80] S. Mishra, C. Schwab, and J. Sukysc. Multi-level Monte Carlo finite volume methods for nonlinear systems of conservation laws in multi-dimensions. *Journal of Computational Physics*, 231(8):3365–3388, 2012.

- [81] S Mohammed and T.S. Zhang. The Burgers equation with affine linear noise: Dynamics and stability. *Stochastic Processes and Their Applications*, 122(4):1887–1916, 2012.
- [82] P. Moin and K. Mahesh. Direct numerical simulation: A tool in turbulence research. *Annual Review of Fluid Mechanics*, 30(1):539–578, 1998.
- [83] W.J. Morokoff and R.E. Caflisch. Quasi-Random Sequences and Their Discrepancies. *SIAM Journal on Scientific Computing*, 15(6):1251, 1994.
- [84] B. Moskowitz and R.E. Caflisch. Smoothness and dimension reduction in quasi-Monte Carlo methods. *Mathematical and Computer Modelling*, 23(8–9):37–54, 1996.
- [85] H.N. Najm. Uncertainty quantification and polynomial chaos techniques in computational fluid dynamics. *Annual Review of Fluid Mechanics*, 41(1):35–52, 2009.
- [86] N. Nalko, P.G. Martinsson, and J.A. Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Review*, 53(2):217–288, 2011.
- [87] F. Nobile, R. Tempone, and C.G. Webster. A sparse grid stochastic collocation method for partial differential equations with random input data. *SIAM Journal on Numerical Analysis*, 46(5):2309, 2008.
- [88] E. Novak and K. Ritter. Simple cubature formulas with high polynomial exactness. *Constructive Approximation*, 15(4):499–522, 1999.
- [89] W.L. Oberkampf, F.G. Blottner, and D.P. Aeschliman. Methodology for computational fluid dynamics code verification and validation. In *26. American Institute of Aeronautics and Astronautics (AIAA) computational fluid dynamics conference, San Diego, CA (United States), 19–22 Jun 1995*, page 25, 1995.

- [90] D.S. Oliver, L.B. Cunha, and A.C. Reynolds. Markov Chain Monte Carlo methods for conditioning a permeability field to pressure data. *Mathematical Geology*, 29(1):61–91, 1997.
- [91] S.A. Orszag and L. R. Bissonnette. Dynamical Properties of Truncated Wiener-Hermite Expansions. *Physics of Fluids*, 10(12):2603, 1967.
- [92] H. Owhadi, C. Scovel, T.J. Sullivan, M. McKerns, and M. Ortiz. Optimal Uncertainty Quantification. *ArXiv e-prints:1009.0679*, September 2010.
- [93] H. Owhadi and L. Zhang. Metric-based upscaling. *Communications on Pure and Applied Mathematics*, 60(5):675–723, 2007.
- [94] A. Quarteroni, R. Sacco, and F. Saleri. *Numerical Mathematics (Texts in Applied Mathematics)*. Springer, 2000.
- [95] C. Rowley. Model reduction for compressible flows using POD and Galerkin projection. *Physica D*, 189(1–2):115–129, February 2004.
- [96] T.P. Sapsis and P.F.J. Lermusiaux. Dynamically orthogonal field equations for continuous stochastic dynamical systems. *Physica D*, 238(23–24):2347–2360, 2009.
- [97] T.P. Sapsis and P.F.J. Lermusiaux. Dynamical criteria for the evolution of the stochastic dimensionality in flows with uncertainty. *Physica D*, 241(1):60–76, 2012.
- [98] W. Schoutens. *Stochastic processes and orthogonal polynomials*. Springer, 2000.
- [99] C. Schwab and R. Todor. Sparse finite elements for elliptic problems with stochastic loading. *Numerische Mathematik*, 95(4):707–734, 2003.

- [100] C. Schwab and R. Todor. Karhunen-Loeve approximation of random fields by generalized fast multipole methods. *Journal of Computational Physics*, 217(1):100–122, September 2006.
- [101] S. Smolyak. Quadrature and interpolation formulas for tensor products of certain classes of functions. *Soviet Math. Dokl.*, 4:240–243, 1963.
- [102] G Stefanou. The stochastic finite element method: Past, present and future. *Computer Methods in Applied Mechanics and Engineering*, 198(9–12):1031–1051, 2009.
- [103] T.J. Sullivan, M. McKerns, D. Meyer, F. Theil, H. Owhadi, and M. Ortiz. Optimal uncertainty quantification for legacy data observations of Lipschitz functions. *ArXiv e-prints:1202.1928*, February 2012.
- [104] T.J. Sullivan, U. Topcu, and H. Owhadi. Uncertainty quantification via codimension-one partitioning. *International Journal for Numerical Methods in Engineering*, 85(12):1499–1521, 2011.
- [105] M. Talagan. A new look at independence. *The Annals of Probability*, 24(1):1–34, 1996.
- [106] R. Todor and C. Schwab. Convergence rates for sparse chaos approximations of elliptic problems with stochastic coefficients. *IMA Journal of Numerical Analysis*, 27(2):232–261, 2006.
- [107] U. Topcu, L.J. Lucas, H. Owhadi, and M. Ortiz. Rigorous uncertainty quantification without integral testing. *Reliability Engineering & System Safety*, 96(9):1085–1091, 2011.
- [108] D. Venturi, X. Wan, and G.E. Karniadakis. Stochastic low-dimensional modelling of a random laminar wake past a circular cylinder. *Journal of Fluid Mechanics*, 606:339–367, 2008.

- [109] D. Venturi, X. Wan, and G.E. Karniadakis. Stochastic bifurcation analysis of Rayleigh-Bénard convection. *Journal of Fluid Mechanics*, 650:391, April 2010.
- [110] X. Wan and G.E. Karniadakis. An adaptive multi-element generalized polynomial chaos method for stochastic differential equations. *Journal of Computational Physics*, 209(2):617–642, 2005.
- [111] X. Wan and G.E. Karniadakis. Multi-Element Generalized Polynomial Chaos for Arbitrary Probability Measures. *SIAM Journal on Scientific Computing*, 28(3):901, 2006.
- [112] G.L. Wang, M. Zeng, and B.L. Guo. Stochastic Burgers' equation driven by fractional Brownian motion. *Journal of Mathematical Analysis and Applications*, 371(1):210–222, 2010.
- [113] Wiener N. The homogeneous chaos. *American Journal of Mathematics*, 60:897–936, 1938.
- [114] D. Williams. *Probability with martingales (Cambridge Mathematical Textbooks)*. Cambridge University Press, 1991.
- [115] D. Xiu. Fast Numerical Methods for Stochastic Computations: A Review. *Communications in Computational Physics*, 5(2):242–272, 2009.
- [116] D. Xiu and J.S. Hesthaven. High-order collocation methods for differential equations with random inputs. *SIAM Journal on Scientific Computing*, 27(3):1118–1139, 2005.
- [117] D. Xiu and G.E. Karniadakis. The Wiener-Askey polynomial chaos for stochastic differential equations. *SIAM Journal on Scientific Computing*, 24(2):619, 2002.
- [118] D. Xiu and G.E. Karniadakis. Modeling uncertainty in flow simulations via generalized polynomial chaos. *Journal of Computational Physics*, 187(1):137–167, 2003.

[119] D. Xiu and J. Shen. Efficient stochastic Galerkin methods for random diffusion equations.

Journal of Computational Physics, 228(2):266–281, 2009.