Adaptive Sparse Grid Approaches to Polynomial Chaos Expansions for Uncertainty Quantification

by

Justin Gregory Winokur

Department of Mechanical Engineering & Materials Science
Duke University

Date: _____

Approved:

_____
Omar M. Knio, Supervisor

_____
Olivier Le Maître

_____
Mohamed Iskandarani

_____
Guglielmo Scovazzi

_____
Henri Gavin

Dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Mechanical Engineering & Materials Science in the Graduate School of Duke University
2015

Abstract

Adaptive Sparse Grid Approaches to Polynomial Chaos Expansions for Uncertainty Quantification

by

Justin Gregory Winokur

Department of Mechanical Engineering & Materials Science
Duke University

Date: _____

Approved:

_____
Omar M. Knio, Supervisor

_____
Olivier Le Maître

_____
Mohamed Iskandarani

_____
Guglielmo Scovazzi

_____
Henri Gavin

An abstract of a dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Mechanical Engineering & Materials Science in the Graduate School of Duke University
2015

Abstract

Polynomial chaos expansions provide an efficient and robust framework to analyze and quantify uncertainty in computational models. This dissertation explores the use of adaptive sparse grids to reduce the computational cost of determining a polynomial model surrogate while examining and implementing new adaptive techniques.

Determination of chaos coefficients using traditional tensor product quadrature suffers the so-called curse of dimensionality, where the number of model evaluations scales exponentially with dimension. Previous work used a sparse Smolyak quadrature to temper this dimensional scaling, and was applied successfully to an expensive Ocean General Circulation Model, HYCOM during the September 2004 passing of Hurricane Ivan through the Gulf of Mexico. Results from this investigation suggested that adaptivity could yield great gains in efficiency. However, efforts at adaptivity are hampered by quadrature accuracy requirements.

We explore the implementation of a novel adaptive strategy to design sparse ensembles of oceanic simulations suitable for constructing polynomial chaos surrogates. We use a recently developed adaptive pseudo-spectral projection (aPSP) algorithm that is based on a direct application of Smolyak's sparse grid formula, and that allows for the use of arbitrary admissible sparse grids. Such a construction ameliorates the severe restrictions posed by insufficient quadrature accuracy. The adaptive algorithm is tested using an existing simulation database of the HYCOM model during Hurricane Ivan. The *a priori* tests demonstrate that sparse and adaptive pseudo-spectral

constructions lead to substantial savings over isotropic sparse sampling.

In order to provide a finer degree of resolution control along two distinct subsets of model parameters, we investigate two methods to build polynomial approximations. The two approaches are based with pseudo-spectral projection (PSP) methods on adaptively constructed sparse grids. The control of the error along different subsets of parameters may be needed in the case of a model depending on uncertain parameters and deterministic design variables. We first consider a nested approach where an independent adaptive sparse grid pseudo-spectral projection is performed along the first set of directions only, and at each point a sparse grid is constructed adaptively in the second set of directions. We then consider the application of aPSP in the space of all parameters, and introduce directional refinement criteria to provide a tighter control of the projection error along individual dimensions. Specifically, we use a Sobol decomposition of the projection surpluses to tune the sparse grid adaptation. The behavior and performance of the two approaches are compared for a simple two-dimensional test problem and for a shock-tube ignition model involving 22 uncertain parameters and 3 design parameters. The numerical experiments indicate that whereas both methods provide effective means for tuning the quality of the representation along distinct subsets of parameters, adaptive PSP in the global parameter space generally requires fewer model evaluations than the nested approach to achieve similar projection error.

In order to increase efficiency even further, a subsampling technique is developed to allow for local adaptivity within the aPSP algorithm. The local refinement is achieved by exploiting the hierarchical nature of nested quadrature grids to determine regions of estimated convergence. In order to achieve global representations with local refinement, synthesized model data from a lower order projection is used for the final projection. The final subsampled grid was also tested with two more robust, sparse projection techniques including compressed sensing and hybrid least-angle-

regression. These methods are evaluated on two sample test functions and then as an *a priori* analysis of the HYCOM simulations and the shock-tube ignition model investigated earlier. Small but non-trivial efficiency gains were found in some cases and in others, a large reduction in model evaluations with only a small loss of model fidelity was realized. Further extensions and capabilities are recommended for future investigations.

To my Family and Friends

Contents

List of Tables

List of Figures

List of Algorithms

List of Abbreviations and Symbols

Symbols

$\boldsymbol{\xi}$	Vector of independent random variables, parameterizing uncertainty in $F(\boldsymbol{\xi})$.
$\rho(\boldsymbol{\xi})$	Integral measure
$F(\boldsymbol{\xi})$	General model QoI
$F'(\boldsymbol{\xi})$	Synthesized or estimate model value of $\boldsymbol{\xi} \in \mathcal{X}^{\boldsymbol{l}}$ of the QoI from $\mathbf{P}_{\mathscr{L}=\mathscr{L}_{\boldsymbol{l}}^{\mathtt{FT}}\setminus \boldsymbol{l}}(\boldsymbol{\xi})$
$\widetilde{F}(\boldsymbol{\xi})$	Mixture of model data and F' used in the subsampled PSP in Chapter 5. PCE solution in Chapter 4 (used with respect to $Z(\boldsymbol{\xi})$)
$F^{\mathcal{L}}(\xi)$	One dimensional Lagrange interpolant
$\mathcal{L}$	One dimensional Lagrange interpolating polynomial basis function
$\mathscr{L}$	Quadrature/Projection *level* index for Smolyak quadrature or PSP construction
$\mathscr{L}_{\boldsymbol{l}}^{\mathtt{FT}}$	Full-Tensor quadrature/projection index
$\mathcal{A}$	"Active" Set for aPSP construction
$\mathcal{O}$	"Old" Set for aPSP Construction
$\mathcal{K}$	General Truncated index. Often PCE index. Unless otherwise stated, $\mathcal{K} \in \mathbb{N}_0^d$
$\mathcal{K}_{\boldsymbol{l}}^{\mathtt{FT}}$	Full-Tensor ("rectangular") multi-index
$\mathcal{K}^{*\boldsymbol{l}}$	*Maximal* PCE index (and half-exactness set) of $\mathbf{Q}_{\boldsymbol{l}}^{\mathtt{FT}}$ (note: full-tensor)

$\mathcal{K}^{*\mathscr{L}}$	*Maximal* PCE index (and half-exactness set) of Smolyak $\mathbf{Q}_{\mathscr{L}}$
$\widetilde{\mathcal{K}}_{s,q}(\mathcal{K})$	Index-set $\mathcal{K}$, enlarged by factor s with additional mixed terms for hyperbolic parameter $q \geqslant 0$
$\mathcal{K}_{\beta}^{\boldsymbol{p},q}$	Hyperbolic cross index-Set, starting at β, with hyperbolic mixed term paramater $q \geqslant 0$, truncated by order $\boldsymbol{p}$. $\mathcal{K}^{\boldsymbol{p},q}$ assumes $\beta = 0$
$\mathcal{K}^{\mathcal{I}}$	Input basis for Compressed Sensing or Hybrid-LAR
$\mathcal{K}^{\mathcal{A}}$	$\mathcal{K}^{\mathcal{A}} \subseteq \mathcal{K}^{\mathcal{I}}$, Active basis for Hybrid-LAR
$\mathcal{K}^{S,\epsilon}$	$\mathcal{K}^{S,\epsilon} \subseteq \mathcal{K}$. Subset of the basis $\mathcal{K}$ above a threshold ϵ.
$\mathcal{K}^{\mathcal{L}}$	Index-Set where for $n_{\boldsymbol{l}}$ points in any direction, polynomials up to order $p_{\boldsymbol{l}} - \mathbf{1}$. Solution can be factored into the Lagrange-Interpolation basis

$\hat{\mathsf{e}}_n$	Canonical index unit vector in the n^{th} direction
$\mathbf{Q}$	Quadrature Operator
$\mathbf{Q}_{\boldsymbol{l}}^{\mathtt{FT}}$	Full-Tensor quadrature with level $\boldsymbol{l}$
$\mathbf{Q}_{\mathscr{L}}$	Smolyak quadrature with index $\mathscr{L}$
$\mathbf{P}$	Projection Operator
$\mathbf{P}_{\boldsymbol{l}}^{\mathtt{FT}}$	Projection with full-tensor quadrature of level $\boldsymbol{l}$
$\mathbf{P}_{\mathscr{L}}$	Smolyak Pseudo-Spectral projection with index $\mathscr{L}$
$\mathscr{A}_{\boldsymbol{l}}$	Polynomial exactness set of full-tensor quadrature $\mathbf{Q}_{\boldsymbol{l}}^{\mathtt{FT}}$
$\mathscr{A}_{\mathscr{L}}$	Exactness set of Smolyak quadrature $\mathbf{Q}_{\mathscr{L}}$

Nested Projection Notation:

$G^{(i)}(\boldsymbol{\xi}_{\mathtt{u}})$	Inner-space Model evaluation, $F(\boldsymbol{\xi}_{\mathtt{p}}{}^{(i)}, \boldsymbol{\xi}_{\mathtt{u}})$
$H(\boldsymbol{\xi}_{\mathtt{p}}{}^{(i)})$	Outer-space QoI of inner space: $\|G^{(i)}\|$

$\tilde{g}_{\boldsymbol{k}}^{(i)}$	Inner-space PCE coeficient (including zeros for $\boldsymbol{k} \notin \mathcal{K}_{\mathtt{u}}$)
$C_{\mathtt{u}}, C_{\mathtt{p}}$	Combined product-space η values.
$\mathcal{F}(\boldsymbol{l})$	Forward neighborhood of index $\boldsymbol{l}$
$Z(\boldsymbol{\xi})$	Error function between the model and the surrogate
$\epsilon(\boldsymbol{l})$	Normalized local estimated projection surplus due to index $\boldsymbol{l}$
η^2	Total estimate of available projection surplus
$\mathtt{Tol}_{\eta}$	aPSP projection surplus tolerance
$\mathcal{G}(\mathscr{L})$	*Full* set of nodes in a grid from index $\mathscr{L}$
$\mathcal{Y}^{\boldsymbol{l}}$	Set of nodes in a particular *full-tensor* quadrature rule, $\mathbf{Q}_{\boldsymbol{l}}^{\mathtt{FT}}$. $\mathcal{Y}^{\boldsymbol{l}} \equiv \mathcal{G}(\mathscr{L}_{\boldsymbol{l}}^{\mathtt{FT}})$.
$\mathcal{X}^{\boldsymbol{l}}$	$\mathcal{X}^{\boldsymbol{l}} \subseteq \mathcal{Y}$ that are *unique* to the particular full-tensor quadrature rule. $\mathcal{X}^{\boldsymbol{l}} \cap \mathcal{X}^{\boldsymbol{l}' \neq \boldsymbol{l}} = \varnothing$
$p^{j}_{\boldsymbol{\xi}^{\boldsymbol{l},(i)}}$	Axial parent of $\boldsymbol{\xi}^{\boldsymbol{l},(i)} \in \mathcal{X}^{\boldsymbol{l}}$ in the j^{th} direction.
$\mathcal{A}^{+}$	$\mathcal{O}$ admissible forward neighbors of all $\mathcal{A}$. $\mathcal{A}^{+} \notin \mathscr{L}$
$\mathcal{C}$	Backward neighbors of all $\mathcal{A}^{+}$. Represents the collection of candidate indices for enrichment. $\mathcal{C} \in \mathscr{L}$
$\mathscr{K}^{\boldsymbol{l}}$	Kept and evaluated nodes in $\mathcal{X}^{\boldsymbol{l}}$
$\mathscr{S}^{\boldsymbol{l}}$	Skipped (and synthesized) nodes in $\mathcal{X}^{\boldsymbol{l}}$
$\mathcal{H}(\mathscr{L})$	Final subsampled grid from $\mathscr{L}$ with local tolerance (or α-based dynamic local tolerance)
$\lambda\left(\boldsymbol{\xi}^{\boldsymbol{l},(i)}\right)$	Local estimated error for $\boldsymbol{\xi}^{\boldsymbol{l},(i)}$
$\mathtt{Tol}_{\lambda}^{k}$	Local tolerance at iteration k
α	In Chapter 5, dynamic local tolerance aggressiveness parameter. In Chapter 4, mixing parameter of monomial and mixed interaction estimates projection surplus
n_{l_i}	Number of 1D quadrature nodes for level l_i
$n_{\boldsymbol{l}}$	Number of quadrature nodes for full-tensor $\mathbf{Q}_{\boldsymbol{l}}^{\mathtt{FT}}$. $n_{\boldsymbol{l}} = \prod_{i=1}^{d} n_{l_i}$

$\boldsymbol{\mathcal{Z}}$	Vandermonde Matrix (also known as Information Matrix) in matrix formation of the PCE solution
Λ	Diagonal weight matrix for BP and BPDN
h_n	n-th diagonal term of $\boldsymbol{\mathcal{Z}}\left(\boldsymbol{\mathcal{Z}}^T\boldsymbol{\mathcal{Z}}\right)^{-1}\boldsymbol{\mathcal{Z}}$
$\gamma^{(n)}$	Error of regression solution built without $\boldsymbol{\xi}^{(n)}$ evaluated at $\boldsymbol{\xi}^{(n)}$. Part of LOO CV
E_{LOO}, E^*_{LOO}	Regular and corrected (respectively) Leave-one-out (LOO) error
$T(P,N)$	Correction factor for regression error
ζ	Relative L_2 error. (Measured or estimated)
$\mathbb{E}\{F\}$	Expectation Value of F
$\mathbb{V}\{F\}$	Variance of F

Abbreviations

PC	Polynomial Chaos
PCE	Polynomial Chaos Expansion
RV	Random Variable
QoI	Quantity of Interest
MC	Monte Carlo
MCMC	Markov-Chain Monte Carlo
LHS	Latin Hypercube
NISP	Non-Intrusive Spectral Projection
	DSP — Direct Spectral Projection
	PSP — Smolyak Pseudo-Spectral Projections
	aPSP — Adaptive PSP
PDF	Probability Density Function
FT	Full-Tensor
CS	Compressed Sensing

BP	Basis Pursuit
BPDN	Basis Pursuit DeNoising
LAR	Least-Angle Regression
CV	Cross-Validation
LOO	Leave-One-Out cross-validation
GPK	Gauss-Patterson-Kronrod [73]
RMS	Root-Mean-Square
HYCOM	HYbrid Coordinate Ocean Model
OGCM	Ocean General Circulation Model
GOM	Gulf-of-Mexico
MLD	Mixed-Layer Depth
SST	Sea-Surface Temperature
SSH	Sea-Surface Height

Acknowledgements

If I have seen further it is by standing on the shoulders of giants.
–*Sir Isaac Newton*, February 15, 1676

In his 1676 letter to Robert Hooke, Newton was discussing his scientific advancements. Such sentiment rings particularly true to me both in the context of the scientific work presented here and in my own personal life.

The work in this dissertation would not have been possible if it were not for the continued support and diligent guidance of my advisor, Dr. Omar Knio. His ability to direct me, both academically and professionally, was invaluable through the years of completing this work. Often times, after a long struggle to understand the nuances of a particular issue, I would present it to him and, within seconds, he would fully comprehend and be ready to assist. After his trademark "big cup of coffee", he would offer valuable and useful direction. Through the many logistical peculiarities of both of our situations, Dr. Knio has been more than accommodating and always sought out solutions with my best interest in mind.

I would also like to thank Dr. Olivier Le Maître and Dr. Mohamed Iskandarani for the honor and the great pleasure of co-authoring papers and for the benefit of their guidance. Their input has always been of the utmost importance in not only refining the methodologies presented here but also the style by which it has been conveyed. I have learned a great deal by mimicking, adapting, and merging their unique writing style with my own voice. Alen Alexanderian, who began his post-doc with Dr. Knio around the same time I arrived, has been an incredible resource for

understanding and conquering new methods as well as a great source of professional advice. I would also like to acknowledge the other co-authors of [133, 134] for their assistance and contributions, and to the authors of [2, 120, 121, 130] for the honor of being able to contribute to their work.

I have not stood on the shoulders of these academic giants alone. The many discussions, insights, and questions of my lab mates, Manav Vohra and Guotu Li have been valuable for both better understanding my own methods (explaining to others is, after all, the best way to ensure you undersand it yourself) and for their guidance. Camaraderie in the trenches of research resulted in better methods and a more complete understanding. I would also like to acknowledge my friends who were both a support in my academic pursuits and also a constant guide to life outside of work.

I wouldn't be where I am today if it weren't for my ever-supportive family whose shoulders have supported me through thick and thin. First of all, I am honored to be continuing my family's tradition of advanced academic pursuit. Mine is only one in a long lineage of scholastic achievements. My maternal grandfather, Richard Coren (A"H) was a professor of physics at Drexel University and he, along with my grandmother, Elaine Coren, were always quick to provide advice and guidance from their own hard-earned experience. My paternal grandparents, Manny (A"H) & Joan Winokur were unwavering in their encouragement, comfort and appreciation!

I would like to particularly mention my parents, Harriet Winokur and Steven Winokur (and Peter Adels). Not only have they been a constant reservoir of support, encouragement, and inspiration, but they have served as an invaluable source of advice and input on both a personal and professional level. I would be remiss if I did not also acknowledge my father's earnest effort to understand and comprehend my research as well as his help in copy editing this dissertation. Of course, my siblings Adam and Emily have always been there and have been an asset to me and my

worldview. Then there is Gracie (my dog), an energizer bunny of humor, comfort, and (sometimes) distraction.

Finally, I do not think I would be the man or the scholar I am today if it weren't for my best friend and wife, Meredith Winokur (née Huffsmith) always by my side. She has been an unwavering source of comfort and motivation. Her patience, as a countless "five minutes" has turned into thirty, and her determination to lift my spirits when I was down has unequivocally been a major reason I have accomplished all that I have. I look forward to many more accomplishments made possible by our teamwork and everlasting love.

This work was supported by the US Department of Energy, Office of Advanced Scientific Computing Research, Awards DE-SC0008789 and DE-SC0007020, the Office of Naval Research Award N00014-101-0498, and by the Gulf of Mexico Research Initiative, contract numbers SA1207GOMRI005 (CARTHE), SA12GOMRI008 (DEEP-C), and SA1515 (CARTHE-II).

1

Introduction

1.1 Uncertainty Quantification and Goals

Uncertainty Quantification (UQ) is a growing field of research that focuses on understanding, quantifying and mitigating uncertainty in computational simulations. Recent advancements in computational power have allowed for more complex models, but inherent in many of these models is uncertainty, necessitating additional analysis and new classes of methods.

There are many sources of uncertainty in computer modeling. Some sources are inherent in the model itself (e.g. due to discretization and grid size), while others may be due to insufficient experimental inputs. In some models, inherent stochasticity yields a probabilistic and non-deterministic output further increasing uncertainty. A particularly difficult type of uncertainty stems from the model governing equations themselves, which are often the result of careful scale analysis and starting assumptions (e.g. continuum assumption, incompressibility, elasticity, etc). Increasing resolution will never resolve physics that are not in the model. Another common type of uncertainty is due to input parameters and initial conditions. Specific model

parameters may not be known exactly such as those appearing in phenomenological closure relations, or actual physical parameters. The various sources of uncertainty are often grouped into two broad categories: aleatoric and epistemic. Aleatoric uncertainties are inherent to the model itself and are *not* due to insufficient knowledge. Epistemic uncertainties are those introduced by model assumptions, parameterizations, and/or discretization. Epistemic uncertainties, in principle, may be reduced given sufficient knowledge of the problem or computational resources. On the other hand, aleatoric uncertainties are intrinsically irreducible [115].

UQ encompasses broad classes of tools and analyses useful both for obtaining a better understanding of a particular model and for improving and validating model response. For example, a practitioner may wish to calibrate a model from experimental data and make use of MCMC sampling on surrogates. Or, the importance of particular uncertainties may not known. Such pursuits lead to the fundamental goal of this work: **Understand and quantify how uncertainty in a model input leads to uncertainty in the resulting model output** (i.e. how do uncertainties propagate?).

In this work, we will focus exclusively on parametric uncertainty, whether due to physical parameters, or parameters of closure relations. We are particularly motivated by high-dimensional examples (i.e. many sources of uncertainties), making analysis particularly difficult due to the complex and often mixed interaction of the uncertain parameters. Traditionally, such analyses scale exponentially with dimension (number of uncertainties) introducing the pervasive curse of dimensionality.

Our goal is to ascertain the dependance and variability of the model on these sources of parametric uncertainty. One common analysis is to calculate the statistical moments (e.g. mean, variance) or to develop a probability density function (PDF) of the model from the stochastic inputs. These quantities are assessable via a Monte-Carlo (MC) experiment however, we are most often interested in large, complex, and

computationally expensive models. The required number of model runs (realizations) required are prohibitively expensive for all but the simplest of models. We therefore must seek an alternative approach and aim to **develop a faithful model surrogate to allow for rapid approximation of model response**. A model surrogate is a functional representation of the model output which we may sample or integrated at very low cost to understand the affects of uncertainty in our model.

In particular, we work in a stochastic paramater space where we parameterize d sources of uncertainty in a model by the vector $\boldsymbol{\xi}$. For a given $\boldsymbol{\xi}$ we represent a model's quantity of interest (QoI) as $F(\boldsymbol{\xi})$. Surrogate modeling may be local, such as splines [74, 44], or global, such as polynomial approximations. We will focus exclusively on global polynomial approximations and in particular Polynomial Chaos Expansions (PCEs) [132]. PCEs are a spectral, Fourier-like expansions that seek to approximate a model's response as a finite series of orthonormal polynomials. We explore them at great length in Chapter 2 but will briefly summarize them here. We approximate $F(\boldsymbol{\xi})$ as a series of the form

$$F(\boldsymbol{\xi}) \approx \sum_{\boldsymbol{k}\in\mathcal{K}} c_{\boldsymbol{k}} \Psi_{\boldsymbol{k}}(\boldsymbol{\xi}), \tag{1.1}$$

where $\mathcal{K}$ is a suitable finite multi-index set and the $\Psi_{\boldsymbol{k}}$'s are the appropriate class of multi-dimensional orthonormal polynomials. The coefficients, $c_{\boldsymbol{k}}$ are referred to as the "chaos-coefficients". We refer the reader to Section 2.1 for a more thorough and complete exposition of PCEs. The main computational work is to determine the chaos coefficients. Once the PCE coefficients have been determined, it is trivial to calculate useful quantities such as the mean, variance and model sensitivities (globally or locally) from the structure of the PCE itself [76, 3].

We are now in a position to further describe the objective of this work. We aim to **develop, implement, and validate novel techniques and algorithms to more efficiently create high-fidelity PCE** (1.1) **surrogates**. There are two general

approaches to determining the coefficients: the so-called intrusive approach, where the PCE is inserted into the governing equations and solved via Galerkin projection, and the non-intrusive approach, where the model is treated as a "black-box" and coefficients are determined from an ensemble of model evaluations.

Much of the initial PCE-based UQ methodology was developed with the intrusive method [49] and had been applied to many areas including heat transfer [55, 87], fluid dynamics and the Navier Stokes equations [71, 81, 94] (including electro-chemical flow [29]), as well as solid/structural mechanics [91, 48]. While intrusive methods remain an active area of research [68, 60, 37, 78], we wish to use large, legacy solvers wherein we do not need to modify the underlying code and will therefore focus exclusively on non-intrusive methods.

Various existing techniques for non-intrusive projections are explored in Chapter 2, Sections 2.3 through 2.7, where the determination of chaos-coefficients often reduces to high-dimensional integration. Traditional multi-dimensional integration again suffers the curse of dimensionality, where the number of model evaluations scale exponentially with dimension (number of uncertainties). We will therefore focus exclusively on Smolyak sparse grids [116], which mitigate the exponential scaling by considering only certain multi-dimensional quadratures. Furthermore, we aim to use the structure of the model itself to adaptively reduce computation. We are finally in a position to state out final goal in this dissertation: **develop and implement efficient adaptive sparse grid approaches to determine high-fidelity polynomial chaos expansions with as few model evaluations as possible while mainting sufficient control over the adaptation**.

1.2 Motivating Problems

The general motivation for this work is to develop, implement, and test novel sparse grid approaches to UQ for PCE surrogates. We are particularly interested in ways

to use the structure of the problem to improve efficiency (i.e. reduce the number of model evaluations) while providing increased control over the adaptation. The PCE surrogate can then be used for a broad range of analyses from inversion [120, 61, 105], forward propagation of uncertainty [104], and sensitivity analysis [117, 26, 3, 2, 7]. These methods are developed to be model-agnostic and may be applied to wide classes of problems. As such, we treat each model as a "black-box" wherein we obtain a deterministic output for a set of random inputs.

The development of these adaptive sparse grid techniques is motivated by the results of *Alexanderian, et al.* [2] and *Thacker, et al.* [124], which studied the uncertainty in the Hybrid Coordinate Ocean Model (HYCOM) [19, 138, 52, 10], an open-source Ocean General Circulation Model (OGCM). Specifically, *Alexanderian, et al.* [2] applied non-adapted sparse grids to studying the affects of a stochastic wind-drag parameter [66] and subgrid-scale parameters due to the KPP [75, 20, 83] turbulence closure relations (See Ref. [2, Section 2.3] for more detail) on the passage of Hurricane Ivan through the Gulf of Mexico (GOM) in September 2004. The results indicated a greater sensitivity to certain parameters suggesting an adaptive approach may be appropriate.

We are also motivated by the desire to both represent the uncertainty in reaction rates and to characterize the dependence on experimental parameters for a methane shock-tube ignition model. We've identified 22 relevant reaction rates and wish to investigate the forward proportion of uncertainty in addition to the dependence on three experimental parameters. Since we expect the variability due to the experimental parameters to be dominant over the stochastic reaction rates we are therefore motivated to better control the convergence due to each type of uncertainty separately.

1.3 Contributions & Layout of This Work

The primary contribution of this work is the development and implementation of novel sparse grid approaches to UQ and is laid out as follows:

In Chapter 2 we formally introduce PCEs and their associated mathematical formalism. We briefly discuss intrusive methods and then we proceed to introduce non-intrusive techniques. We particularly focus on sparse grid projection methods and introduce potential hurdles to adaptive approaches. We also provide an overview of novel sparse techniques for PCEs as well as collocation methods.

In Chapter 3 (adapted from [134, 133]), we use the results of *Alexanderian, et al.* [2] to motivate the need for adaptive methods. We introduce a novel Smolyak pseudo spectral projection method (aPSP) [25, 22] to perform an adapted projection and improve efficiency. We use these methods on an existing database of HYCOM realizations to demonstrate, *a priori*, the ability to realize considerable savings. We examine the various facets of the aPSP method on the HYCOM problem and possible extensions to multiple quantities of interest (QoIs).

In Chapter 4 (adapted from [133]), we use a high dimensional shock-tube combustion model with 22 stochastic reaction rates and three design variables to motivate the need for additional directional control of the aPSP algorithm. We introduce two techniques to better control convergence and validate these approaches on a simple 2D test problem before applying it to the shock-tube model. We analyse the two methods and discuss further extensions.

We again aim to further reduce the computation burden of determining PCEs in Chapter 5, where we introduce a locally-adaptive subsampling technique similar to that of [63]. We develop a method to perform projections without full model data and we introduce the use of two more robust PCE solvers. We again validate these methods on model test problems and then perform an *a priori* analysis of

the HYCOM model of Chapter 3 (and [2, 134]) along with the shock-tube model of Chapter 4 (and [133]) to demonstrate a small but non-trivial improvement in efficiency.

Finally, in Chapter 6 we summarize and conclude the work presented here and examine future extensions. We also present Smolyak quadrature [116, 47] in a simple, easy-to-understand manner that eschews the mathematical formalism for an intuitive and simple understanding in Appendix A. Appendices B and C introduce concepts used elseware in this dissertation.

2

Polynomial Chaos Expansions

In this chapter we set notation and formally introduce PCEs [132]. We also explore different techniques to determine the PCE coefficients for a given model. PCEs for UQ were initially developed in the seminal work of *Ghanem & Spanos* [49] and focused on functional representations expressed in terms of Gaussian random variables (RVs). PCEs are well suited to represent smooth QoIs with rapidly converging expansions [76], as well as QoIs exhibiting steep or even discontinuous dependence on the uncertain parameter [79, 80]. PCEs have since been generalized to wide classes of basis functions corresponding to particular distributions of input uncertainties [135, 5]. Furthermore, they allow us to readily determine statistical moments [76], such as the mean and variance of QoIs, to compute model output sensitivity to input parameters [26, 2], and to substantially speed up the solution of inverse problems [90, 89, 120]. Other bases, such as wavelets, are particularly well suited for a hierarchical multiresolution analysis and allow for discontinuities where a global polynomial basis may not be suitable [76].

2.1 PCE Formulation

In this work, we restrict ourselves to PCEs based on a suitable parameterization of model inputs in terms of a $d-$dimensional real-valued random vector $\boldsymbol{\xi} \doteq (\xi_1 \cdots \xi_d)$. The random vector $\boldsymbol{\xi}$ is assumed to have independent components; we denote $\Xi \doteq \Xi_1 \times \cdots \times \Xi_d \subseteq \mathbb{R}^d$ the range of $\boldsymbol{\xi}$ and $\rho : \Xi \mapsto \mathbb{R}_+$ its probability density function,

$$\rho(\boldsymbol{\xi}) \doteq \rho_1(\xi_1) \cdots \rho_d(\xi_d), \quad \int_{\Xi_i} \rho_i(\xi_i) d\xi_i = 1 \text{ for } i = 1, \cdots, d. \tag{2.1}$$

For given ρ and Ξ, we introduce $L_2(\Xi, \rho)$ the space of real-valued, second-order random variables in $\boldsymbol{\xi}$, such that

$$F(\boldsymbol{\xi}) \in L_2(\Xi, \rho) \Leftrightarrow \|F\|_{L_2(\Xi,\rho)} < \infty, \quad \|F\|^2_{L_2(\Xi)} \doteq \langle F, F \rangle, \tag{2.2}$$

where $\langle \cdot, \cdot \rangle$ denotes the natural inner product in $L_2(\Xi, \rho)$:

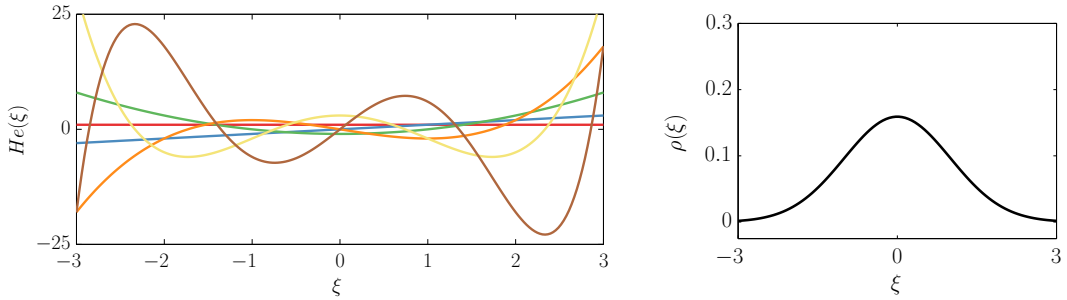
$$\forall F, G \in L_2(\Xi, \rho)^2, \ \langle F, G \rangle \doteq \int_{\Xi} F(\boldsymbol{\xi}) G(\boldsymbol{\xi}) \rho(\boldsymbol{\xi}) d\boldsymbol{\xi}. \tag{2.3}$$

For $i = 1, \cdots, d$, let $\{\psi_0^i, \psi_1^i, \dots\}$ be the set of orthonormal polynomials based on the measure ρ_i, where the subscript k, refers to the polynomial order, such that $\psi_0^i = 1$. For $\boldsymbol{k} = (k_1 \cdots k_d) \in \mathbb{N}_0^d$, we define

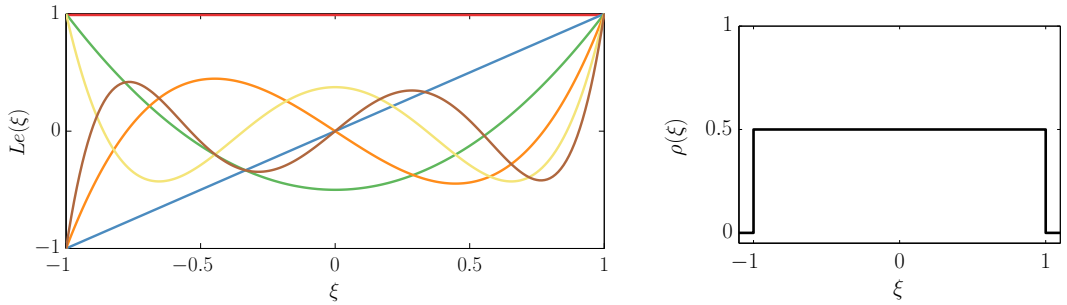
$$\Psi_{\boldsymbol{k}}(\boldsymbol{\xi}) \doteq \prod_{i=1}^{d} \psi_{k_i}^i(\xi_i), \tag{2.4}$$

the d-variate polynomial in $\boldsymbol{\xi}$ whose total degree is $|\boldsymbol{k}| \doteq \sum_{i=1}^d k_i$. It is immediate to verify that $\{\Psi_{\boldsymbol{k}}, \boldsymbol{k} \in \mathbb{N}^d\}$ is an orthonormal polynomial basis of $L_2(\Xi, \rho)$.

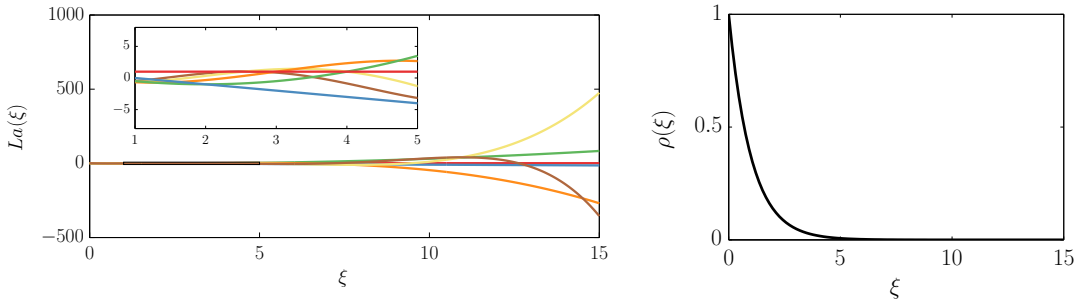
PCEs were first developed with Gaussian measure [132] (corresponding to Hermite polynomials) [49] though any arbitrary measure may be used. In this work, we will focus on a uniform measure corresponding to Legendre polynomials. Other families of discrete polynomials are presented in Table. 2.1. See [76] for discrete



(a) Hermite polynomials with a Gaussian distribution



(b) Legendre polynomials with a uniform distribution



(c) Laguerre polynomials with a gamma distribution

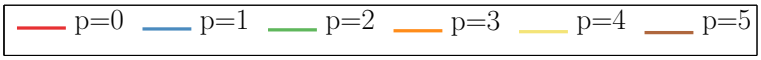


FIGURE 2.1: Examples of three classes of orthogonal polynomials (left) and their corresponding measure (right). Note that what is plotted is the *non-normalized* forms of the polynomials.

Table 2.1: Table of continuous polynomial families for various measures. Note that a uniform measure is a special case of the Beta distribution. Adapted from [76].

Distribution ($\rho(\xi)$)	Polynomial Family	Support
Gaussian	Hermite	$(-\infty, \infty)$
Gamma, γ	Laguerre	$[0, \infty]$
Beta, β	Jacobi	$[a, b]$
Uniform	Legendre	$[a, b]$

families. Plots of the Hermite, Legendre and Laguerre polynomials and measures are provided in Fig. 2.1.

The Polynomial Chaos (PC) approximation of $F \in L_2(\Xi, \rho)$ is

$$F(\boldsymbol{\xi}) \approx \sum_{\boldsymbol{k} \in \mathcal{K}} c_{\boldsymbol{k}} \Psi_{\boldsymbol{k}}(\boldsymbol{\xi}), \tag{2.5}$$

where $\mathcal{K} \subset \mathbb{N}^d$ is a prescribed set of multi-indices defining the truncated PC basis and $c_{\boldsymbol{k}}$ are deterministic coefficients. In this work we restrict ourselves to the case of uniform measures ρ_i corresponding to the tensorization of translated and rescaled Legendre polynomials ψ_k^i, and we shall omit hereafter, when not necessary, references to specific dimensions $i = 1, \cdots, d$.

2.2 Intrusive Methods

A class of solution methodologies for determine the PCE coefficients is the so-called intrusive methods. Intrusive methods inject the PCE (2.5) into the model governing equations and solve for all coefficients simultaneously via a Galerkin Projection [49, 76].

Intrusive methods provide immediate access to the coefficients through a single (though much larger) model solution but the structure of the stochastic problem may require different solvers from those of the deterministic model [76]. Non-linear governing equations are especially challenging, often requiring special treatment of

the Galerkin projection [30].

Because they require a complete reformulation of the model governing parameters, intrusive methods are usually limited to simpler models. While useful to gain insight into the response of a stochastic model, such reformulation makes intrusive projection prohibitive for large, legacy solvers. Instead, for the remainder of the discussion, we will focus on non-intrusive methods where the model is treated as a so-called "black-box" and does not require modification.

2.3 Non-Intrusive Spectral Projection

To determine the coefficients in (2.5), we consider so-called non-intrusive methods where the coefficients of a quantity of interest F are determined from a finite ensemble of deterministic model evaluations (or realizations) for different values of $\boldsymbol{\xi} \in \Xi$ [76]. We focus on the Non-Intrusive Spectral Projection (NISP) [102, 81] approach, which require the approximation error $F(\boldsymbol{\xi}) - \sum_{\boldsymbol{k}\in\mathcal{K}} c_{\boldsymbol{k}}\Psi_{\boldsymbol{k}}(\boldsymbol{\xi})$ to be orthogonal to the span of the polynomial basis. Using the orthonormal character of the basis, this leads to

$$\left\langle F - \sum_{\boldsymbol{k}'\in\mathcal{K}} c_{\boldsymbol{k}'}\Psi_{\boldsymbol{k}'}, \Psi_{\boldsymbol{k}} \right\rangle = 0 \Rightarrow c_{\boldsymbol{k}} = \langle F, \Psi_{\boldsymbol{k}} \rangle$$

$$= \int_{\Xi} F(\boldsymbol{\xi})\Psi_{\boldsymbol{k}}(\boldsymbol{\xi})\rho(\boldsymbol{\xi})d\boldsymbol{\xi} \quad \forall\ \boldsymbol{k} \in \boldsymbol{\mathcal{K}}. \tag{2.6}$$

Classically, the integrals in the right-hand side of (2.6) are approximated using an N_q point quadrature rule of the form

$$c_{\boldsymbol{k}} = \int_{\Xi} F(\boldsymbol{\xi})\Psi_{\boldsymbol{k}}(\boldsymbol{\xi})\rho(\boldsymbol{\xi})d\boldsymbol{\xi} \approx \mathbf{Q}(F\Psi_{\boldsymbol{k}}) = \sum_{q=1}^{N_q} F(\boldsymbol{\xi}^{(q)})\Psi_{\boldsymbol{k}}(\boldsymbol{\xi}^{(q)})w^{(q)}, \tag{2.7}$$

where the $\{\boldsymbol{\xi}^{(q)}, q = 1, \cdots, N_q\}$ is the set of quadrature points having associated weights $w^{(q)}$ for quadrature rule $\mathbf{Q}$. Equation (2.7) shows that the main computational burden of NISP is to compute N_q model evaluations $F(\boldsymbol{\xi}^{(q)})$, so the complexity

of the method is $\mathcal{O}(N_q)$. In traditional tensor-product quadratures, N_q scales exponentially with the number of dimensions d (the so-called curse of dimensionality) and is intractable for all but very low dimensional problems [76]. This issue has motivated the use of sparse grid methods [67], which mitigate the computation burden by reducing the number of model evaluations to construct a suitable PC approximation of F.

2.3.1 Direct Spectral Projection

Consider $\mathrm{Q}_1, \mathrm{Q}_2, \ldots$ a sequence of 1-D quadrature formulas having increasing polynomial exactness, denoted p_l for Q_l, that is

$$\int G(\xi)\rho(\xi)d\xi = \mathrm{Q}_l G = \sum_{q=1}^{N(l)} G\left(\xi^{(q,l)}\right) w_{1D}^{(q,l)}, \quad \forall G \in \pi_{p_l},$$

where π_{p_l} the set of polynomials of degree less or equal to p_l and $N(l)$ is the number of points in the formula Q_l. We call l the level of the formula Q_l. Prescribing the multi-index $\boldsymbol{l} = (l_1 \cdots l_d) \in \mathbb{N}_+^d$, the full-tensor-product (FT) quadrature of an integrable function, G, can be written as follows:

$$\begin{aligned}
\mathbf{Q}_{\boldsymbol{l}}^{\mathtt{FT}} G &= \left(\mathrm{Q}_{l_1} \otimes \cdots \otimes \mathrm{Q}_{l_d}\right) G \\
&= \sum_{q_1=1}^{N(l_1)} \cdots \sum_{q_d=1}^{N(l_q)} G\left(\xi_1^{(q_1,l_1)}, \cdots, \xi_d^{(q_d,l_d)}\right) w_{1D}^{(q_1,l_1)} \cdots w_{1D}^{(q_d,l_d)} \\
&= \sum_{q=1}^{N(\boldsymbol{l})} G(\boldsymbol{\xi}^{(q,\boldsymbol{l})}) w_{\boldsymbol{l}}^{(q,\boldsymbol{l})}. \qquad (2.8)
\end{aligned}$$

The (FT) quadrature rule can be used in Equation (2.7) to compute the NISP coefficients, $c_{\boldsymbol{k}}$, using $G \equiv F\Psi_{\boldsymbol{k}}$, $\boldsymbol{k} \in \mathcal{K}$. We call using the same quadrature rule to compute all the coefficients $c_{\boldsymbol{k}}\ \forall\ \boldsymbol{k} \in \mathcal{K}$, the Direct Spectral Projection (DSP).

Note that in the case of uniform measures, the use of *nested sequences*, for instance based on the Clenshaw-Curtis [21, 131], Fejer, [43] and Gauss-Patterson-

Kronrod (GPK) [73, 98] rules, is often preferred. The complexity of $\mathbf{Q}_{\boldsymbol{l}}^{\mathtt{FT}}$ for such nested sequences is $N(\boldsymbol{l}) = \prod_{i=1}^{d} N(l_i)$, and so it increases exponentially with d.

Smolyak quadrature [116, 47] uses sparse grids to mitigate this complexity by first introducing the 1-D difference formulas between two successive level, $\Delta_l^{\mathrm{Q}} = \mathrm{Q}_l - \mathrm{Q}_{l-1}$, $\Delta_1^{\mathrm{Q}} = \mathrm{Q}_1$, such that Equation (2.8) can be recast as

$$\begin{aligned}
\mathbf{Q}_{\boldsymbol{l}}^{\mathtt{FT}} G &= \left(\mathrm{Q}_{l_1} \otimes \cdots \otimes \mathrm{Q}_{l_d}\right) G \\
&= \left(\left(\sum_{i_1=1}^{l_1} \Delta_{i_1}^{\mathrm{Q}}\right) \otimes \cdots \otimes \left(\sum_{i_d=1}^{l_d} \Delta_{i_d}^{\mathrm{Q}}\right)\right) G \\
&= \sum_{i_1=1}^{l_1} \cdots \sum_{i_d=1}^{l_d} \left(\Delta_{i_1}^{\mathrm{Q}} \otimes \cdots \otimes \Delta_{i_d}^{\mathrm{Q}}\right) G \\
&= \sum_{\boldsymbol{i} \in \mathscr{L}_{\boldsymbol{l}}^{\mathtt{FT}}} \left(\Delta_{i_1}^{\mathrm{Q}} \otimes \cdots \otimes \Delta_{i_d}^{\mathrm{Q}}\right) G = \sum_{\boldsymbol{i} \in \mathscr{L}_{\boldsymbol{l}}^{\mathtt{FT}}} \boldsymbol{\Delta}_{\boldsymbol{i}}^{\mathrm{Q}} G,
\end{aligned} \tag{2.9}$$

where

$$\mathscr{L}_{\boldsymbol{l}}^{\mathtt{FT}} = \{\boldsymbol{i} \in \mathbb{N}_+^d, i_j \leqslant l_j \text{ for } j = 1, \cdots, d\}$$

is the ($\mathtt{FT}$) multi-index set of tensorizations. The sparse quadrature rule $\mathbf{Q}_{\mathscr{L}}$ is finally constructed by considering the summation over a subset $\mathscr{L}$ of tensorized quadrature differences:

$$\mathbf{Q}_{\mathscr{L}} G = \sum_{\boldsymbol{i} \in \mathscr{L}} \boldsymbol{\Delta}_{\boldsymbol{i}}^{\mathrm{Q}} G, \quad \mathscr{L} \subset \mathscr{L}_{\boldsymbol{l}}^{\mathtt{FT}}. \tag{2.10}$$

Traditionally, $\mathscr{L}$ is truncated with a total-level truncation to level ℓ as [116, 47]

$$\mathscr{L}_\ell = \left\{ \boldsymbol{i} \in \mathbb{N}_+^d \Big| \sum_{j=1}^{d} i_j \leqslant \ell + d - 1 \right\}.$$

When $\mathscr{L} = \mathscr{L}_{\boldsymbol{l}}^{\mathtt{FT}}$, $\mathbf{Q}_l = \mathbf{Q}_{\boldsymbol{l}}^{\mathtt{FT}}$ is also identically recovered for *nested* quadratures. However, any arbitrary set of tensorizations $\mathscr{L}$ may be used as long as it is admissible in the sense that the following condition holds [46, 47]

$$\forall\ \boldsymbol{l} = (l_1 \cdots l_d) \in \mathscr{L} : l_{1 \leqslant j \leqslant d} > 1 \Rightarrow \boldsymbol{l} - \hat{\mathsf{e}}_j \in \mathscr{L},$$

where $\{\hat{\mathsf{e}}_j, j = 1, \cdots, d\}$ is the canonical unit vectors of $\mathbb{N}^d$. This admissibility condition is necessary to preserve the telescopic property of the sum of quadrature differences.

For a simpler but more in-depth introduction to Smolyak quadratures and their construction, see Appendix A.

In the following, we denote by $\mathcal{G}(\mathscr{L})$ the set of nodes in the sparse grid,

$$\mathcal{G}(\mathscr{L}) \doteq \bigcup_{\boldsymbol{l} \in \mathscr{L}} \left\{ \left(\xi^{(q_1, l_1)} \cdots \xi^{(q_d, l_d)} \right), 1 \leqslant q_j \leqslant N(l_j), 1 \leqslant j \leqslant d \right\} \tag{2.11}$$

and let $N(\mathscr{L}) = |\mathcal{G}(\mathscr{L})|$, where $|\cdot|$ is the cardinality of a set.

Furthermore, if $\mathscr{A}_{\boldsymbol{l}}$ represents the polynomial exactness set of $\mathbf{Q}^{\mathtt{FT}}$, then the exactness set of the Smolyak quadrature rule, $\mathbf{Q}_{\mathscr{L}}$ is given by [47, 46, 22]

$$\mathscr{A}_{\mathscr{L}} = \bigcup_{\boldsymbol{l} \in \mathscr{L}} \mathscr{A}_{\boldsymbol{l}}.$$

A quadrature rule $\mathbf{Q}_{\mathscr{L}}$ is said to be *sufficiently exact* with respect to a polynomial multi-index set $\mathcal{K}$ when, for any couple $(\boldsymbol{k}, \boldsymbol{k}') \in \mathcal{K} \times \mathcal{K}$, the basis orthonormality condition is recovered using the discrete inner product:

$$\langle U, V \rangle_{\mathbf{Q}_{\mathscr{L}}} \doteq \mathbf{Q}_{\mathscr{L}}(UV) \tag{2.12}$$

If $\mathbf{Q}_{\mathscr{L}}$ is not sufficiently exact, severe and detrimental "internal aliasing" effects will ensue when performing DSP [134, 22]. In the case of `FT` quadratures, the largest admissible set $\mathcal{K}^{*\boldsymbol{l}}$ can be easily determined from the degrees of polynomial exactness of the 1-D sequence, using the so-called half-accuracy set. It follows that when Smolyak quadrature rule $\mathbf{Q}_{\mathscr{L}}$ has an exactness $\mathscr{A}_{\mathscr{L}}$, $\mathcal{K}^{*\mathscr{L}}$ is given by

$$\mathcal{K}^{*\mathscr{L}} = \bigcup_{\boldsymbol{l} \in \mathscr{L}} \left\lfloor \frac{\mathscr{A}_{\boldsymbol{l}}}{2} \right\rfloor, \tag{2.13}$$

the half-exactness set of the Smolyak quadrature, where $\lfloor \cdot \rfloor$ is the floor operator.

Schematic examples are presented in Fig. 2.2 of three $\mathscr{L}$ indices and the resulting full-accuracy sets, $\mathscr{A}_{\mathscr{L}}$ and the half-accuracy sets, $\mathcal{K}^{*\mathscr{L}}$. Note that for Fig. 2.2(a) and Fig. 2.2(b), the $\mathscr{L}$ is based on a hyperbolic-cross (see Appendix C, Equation C.4) with Fig. 2.2(b) representing the traditional total-level truncation ($q = 1$). Fig. 2.2(c) represents $\mathscr{L}^{\mathtt{FT}}$ and the resulting $\mathcal{K}^{*\mathscr{L}}$.

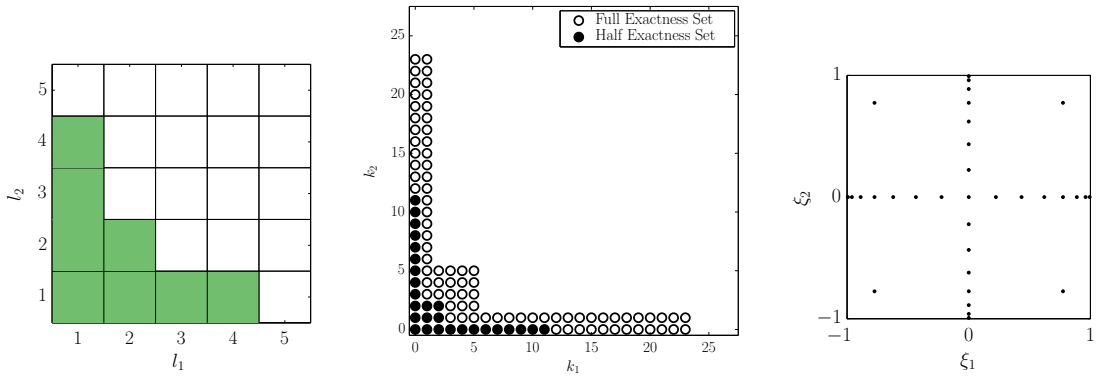
We shall denote $\mathbf{P}_{\boldsymbol{l}}^{\mathtt{FT}} F$ the ($\mathtt{FT}$) projection operator of $F \in L_2(\Xi, \rho)$ onto the span of $\{\Psi_{\boldsymbol{k}}, \boldsymbol{k} \in \mathcal{K}^{*\boldsymbol{l}}\}$, namely

$$\mathbf{P}_{\boldsymbol{l}}^{\mathtt{FT}} F \doteq \sum_{\boldsymbol{k} \in \mathcal{K}^{*\boldsymbol{l}}} c_{\boldsymbol{k}} \Psi_{\boldsymbol{k}}(\boldsymbol{\xi}), \quad c_{\boldsymbol{k}} = \mathbf{Q}_{\boldsymbol{l}}^{\mathtt{FT}}(F \Psi_{\boldsymbol{k}}). \tag{2.14}$$

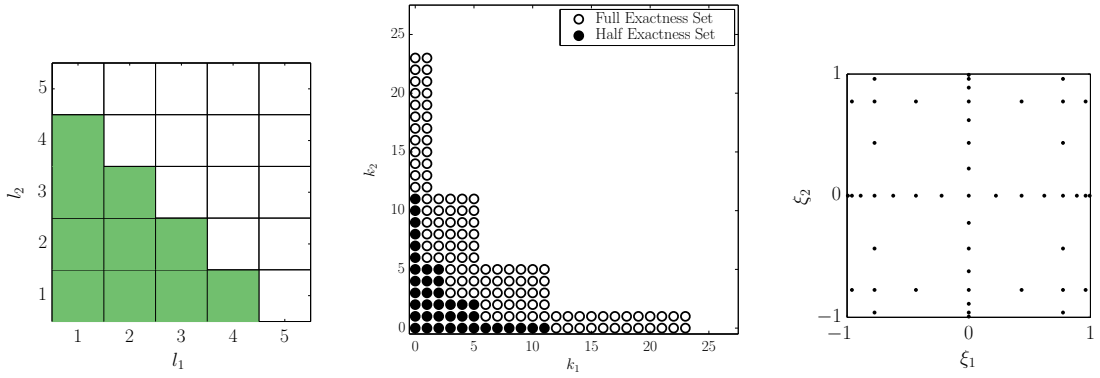
where $\mathscr{L}_{\boldsymbol{l}}$ has exactness set $\mathscr{A}_{\boldsymbol{l}}$ and half-exactness set $\mathcal{K}^{*\boldsymbol{l}}$.

An immediate consequence of using the sparse quadrature formula with $\mathscr{L} \subset \mathscr{L}_{\boldsymbol{l}}^{\mathtt{FT}}$, is that $\mathbf{Q}_{\mathscr{L}}$ is not necessarily sufficiently exact with respect to $\mathcal{K}^{*\mathscr{L}}$, as the sparse formula is not able to exactly integrate the product of high-order monomials. Such a limitation is readily visible in Fig. 2.2(b) for the case of a "traditional" Smolyak quadrature. The inner product of the monomials $\Psi_{\boldsymbol{k}=\{11,0\}}$ and $\Psi_{\boldsymbol{k}'=\{0,11\}}$ require integrating a polynomial of highest-order $\xi_1^{11}\xi_2^{11}$ but $\{11, 11\} \notin \mathcal{A}_{\mathscr{L}}$. A DSP performed with $\mathbf{Q}_{\mathscr{L}}$ and $\mathcal{K}^{*\mathscr{L}}$ would introduce internal aliasing [22].

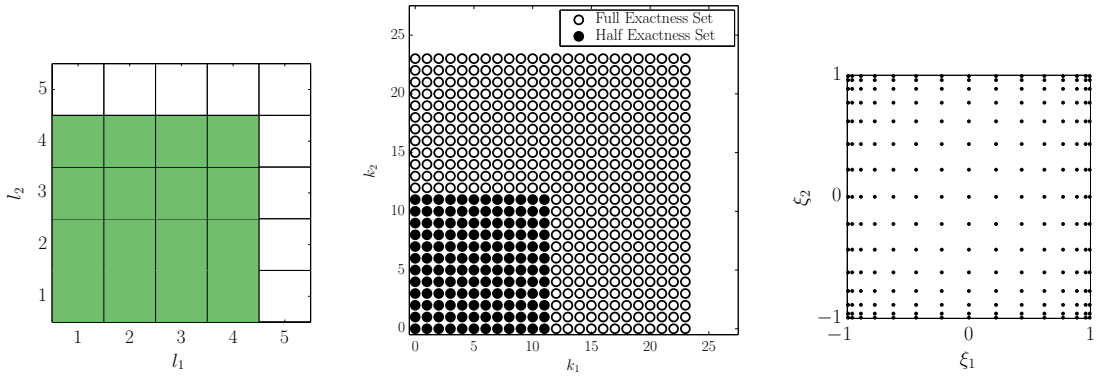
We can still define the largest sets $\mathcal{K}^*(\mathscr{L})$ for which the sparse quadrature is sufficiently exact, but these sets are usually non-unique, leading to ambiguous maximal internal aliasing-free projection spaces. In any case, we observe that the use of sparse quadrature in place of $\mathtt{FT}$ ones, while reducing the complexity ($|\mathcal{G}(\mathscr{L})| \ll \prod_{j=1}^{d} N(l_j)$), leads also to a significant reduction of the polynomial space allowing for internal aliasing-free DSP ($|\mathcal{K}^*(\mathscr{L})| \ll |\mathcal{K}^{*\boldsymbol{l}}|$). The sparse pseudo-spectral projection (PSP) method presented in Chapter 3, Section 3.3 allows for polynomial spaces larger than defined $\mathcal{K}^*(\mathscr{L})$, for the same sparse grid $\mathcal{G}(\mathscr{L})$ and without introducing internal aliasing.



(a) Arbitrary $\mathscr{L}$



(b) "Traditional" total-order $\mathscr{L}$



(c) Full-Tensor $\mathscr{L}$

FIGURE 2.2: Schematic examples of an index-set $\mathscr{L}$ (right), the resulting accuracy sets $\mathcal{A}_{\mathscr{L}}$ and the half-accuracy set $\mathcal{K}^{*\mathscr{L}}$ (the *maximal* PCE index) (center), and the quadrature stencil (right). Accuracy and quadrature stencil correspond to the nested GPK rules [73, 98].

2.4 Regression

NISP and quadrature based methods rely on the use of a tensor-product structured grid of specific quadrature points and associated weights. Another method to estimate the coefficients of Equation (2.5) is to use regression techniques [6, 8] such as least-squares linear fitting.

It can be shown that the NISP methods minimuze the L_2 distance between the surrogate and the actual model with a quadrature-based estimate. For a set of N sample points $\boldsymbol{\xi}$, regression methods seek to minimize the *discrete* norm

$$\left\| F - \sum_{\boldsymbol{k}\in\mathcal{K}} c_{\boldsymbol{k}}\Psi_{\boldsymbol{k}} \right\|^2 = \frac{1}{N}\sum_{i=1}^{N}\left(F\left(\boldsymbol{\xi}^{(i)}\right) - \sum_{\boldsymbol{k}\in\mathcal{K}} c_{\boldsymbol{k}}\Psi_{\boldsymbol{k}}\left(\boldsymbol{\xi}^{(i)}\right)\right)^2$$

on a given basis $\mathcal{K}$ with $P+1 = |\mathcal{K}|$. For a given set of sample poins, $\boldsymbol{\xi}^{(i)}$ and correspond realizations $F(\boldsymbol{\xi}^{(i)}),\ i = 1, ..., N$, the matrix form of the PCE formulation is:

$$\boldsymbol{\mathcal{Z}c} = \boldsymbol{f} \tag{2.15}$$

where $\boldsymbol{\mathcal{Z}} \in \mathbb{R}^{N\times(P+1)}$ is an *information* or generalized Vandermonde-type matrix defined by

$$\boldsymbol{\mathcal{Z}}_{ik} \doteq \Psi_k(\boldsymbol{\xi_i}) \qquad i = 1, ..., N,\ k = 0...P, \quad P+1 = |\mathcal{K}|, \tag{2.16}$$

$\boldsymbol{f} \in \mathbb{R}^{N\times 1} = \left\{F(\boldsymbol{\xi^{(1)}}), ..., F(\boldsymbol{\xi^{(N)}})\right\}^T$, and $\boldsymbol{c} \in \mathbb{R}^{(P+1)\times 1} = \{c_0, ..., c_P\}^T$. We note that since $N > P+1$, Equation (2.15) is overdetermined and no solution exists, Instead, we seek the least-squares solution.

It can be shown [76, 9] that the discrete L_2 distance between $\boldsymbol{f}$ and $\boldsymbol{\mathcal{Z}c}$ is minimized when

$$\boldsymbol{c} = \left(\boldsymbol{\mathcal{Z}}^T\boldsymbol{\mathcal{Z}}\right)^{-1}\boldsymbol{\mathcal{Z}}^T\boldsymbol{f} \tag{2.17}$$

Unlike the NISP approach, there is no requirement that the realizations are structured; however, a necessary and sufficient condition for the problem to be well-posed

is that $\left(\boldsymbol{\mathcal{Z}}^T\boldsymbol{\mathcal{Z}}\right)$ must be full-rank [76]. The selection of minimization points will affect the properties of $\left(\boldsymbol{\mathcal{Z}}^T\boldsymbol{\mathcal{Z}}\right)$ and, as such, there exists optimal conditions for the selection of $\boldsymbol{\xi}$. See [76, p. 67] for more detail.

The point-wise error of the PCE may be ascertained empirically on the sample points as

$$Err^2_{emp} = \frac{1}{N}\sum_{i=1}^{N}\left(F\left(\boldsymbol{\xi}^{(i)}\right) - \sum_{\boldsymbol{k}\in\mathcal{K}} c_{\boldsymbol{k}}\Psi_{\boldsymbol{k}}\left(\boldsymbol{\xi}^{(i)}\right)\right)^2$$

but this is the exact quantity that Equation (2.17) is constructed to minimize and is therefore a poor measure. Furthermore, if $(P+1) = N$, $Err_{emp} = 0$ despite potentially over-fitting the data. Additional correction parameters have been investigated [18, 112] but still do not fully account for over-fitting.

Rather than measure the empirical error of all of the sample points, an alternative approach is to use cross-validation (CV) techniques such as Leave-One-Out (LOO) error. A LOO error estimate is computed by estimating the model with all points but one $\left(\boldsymbol{\xi}^{(n)}\right)$, then calculating the error of the PCE built without $\boldsymbol{\xi}^{(n)}$ at $\boldsymbol{\xi}^{(n)}$. The full algorithm is presented in Algorithm 2.1. The calculation of E_{LOO} may be

Algorithm 2.1 LOO CV Error Estimate.

for $n = 1, \ldots, N$ **do**
- $\boldsymbol{f}^{(\sim n)} = \left\{F(\boldsymbol{\xi}^{(1)}), ..., F(\boldsymbol{\xi}^{(n-1)}), F(\boldsymbol{\xi}^{(n+1)}), ..., F(\boldsymbol{\xi}^{(N)})\right\}^T$ ▷ $\boldsymbol{f}$ without $\boldsymbol{\xi}^{(n)}$
- $\boldsymbol{\mathcal{Z}}^{\sim n}_{ik} = \Psi_k(\boldsymbol{\xi}^{(\boldsymbol{i})}) \quad i = 1, ..., (n-1), (n+1), ..., N \; k = 0...P$ ▷ $\boldsymbol{\mathcal{Z}}$ without $\boldsymbol{\xi}^{(n)}$
- $\boldsymbol{c}^{(\sim n)} = \left((\boldsymbol{\mathcal{Z}}^{\sim n})^T\boldsymbol{\mathcal{Z}}^{\sim n}\right)^{-1}(\boldsymbol{\mathcal{Z}}^{\sim n})^T\boldsymbol{f}^{(\sim n)}$ ▷ Solution without $\boldsymbol{\xi}^{(n)}$
- $\gamma^{(n)} = F\left(\boldsymbol{\xi}^{(n)}\right) - \sum_{\boldsymbol{k}\in\mathcal{K}} \boldsymbol{c}_{\boldsymbol{k}}^{(\sim n)}\Psi_{\boldsymbol{k}}\left(\boldsymbol{\xi}^{(n)}\right)$ ▷ Error at $\boldsymbol{\xi}^{(n)}$

end for

$E^2_{LOO} = \frac{1}{N}\sum_{n=1}^{N}\left(\gamma^{(n)}\right)^2$

dramatically accelerated with an analytical form for $\gamma^{(n)}$ [11, 9]. Let h_n be the n-th diagonal term of $\boldsymbol{\mathcal{Z}}\left(\boldsymbol{\mathcal{Z}}^T\boldsymbol{\mathcal{Z}}\right)^{-1}\boldsymbol{\mathcal{Z}}$, then

$$\gamma^{(n)} = \frac{F\left(\boldsymbol{\xi}^{(n)}\right) - \sum_{\boldsymbol{k}\in\mathcal{K}} \boldsymbol{c}_{\boldsymbol{k}}\Psi_{\boldsymbol{k}}\left(\boldsymbol{\xi}^{(n)}\right)}{1 - h_n}$$

where $\boldsymbol{c_k}$ are the coefficients calculated with all $\boldsymbol{\xi}^{(i)}$. Since $\left(\boldsymbol{\mathcal{Z}}^T\boldsymbol{\mathcal{Z}}\right)^{-1}$ must be calculated in Equation (2.17), it is very inexpensive to also calculate E_{LOO}. Leave-One-Out error is also equivalent to K-Fold Cross-Validation with $K = N$ [72]. The E_{LOO} is still potentially sensitive to over-filling (as $(P+1) \to N$) and therefore the final error may be corrected as follows:

$$E^*_{LOO} = T(P,N)E_{LOO} \tag{2.18}$$

with possible forms for $T(P,N)$ such as [9]

$$T(P,N) = \frac{N-1}{N-P-1}$$

and [9, 18]

$$T(P,N) = \frac{N}{N-P}\left(1 + \frac{\text{trace}\left(\left(\frac{1}{N}\boldsymbol{\mathcal{Z}}^T\boldsymbol{\mathcal{Z}}\right)^{-1}\right)}{N}\right).$$

Regression approaches do not natively suggest an optimal PCE basis, $\mathcal{K}$. If the basis is chosen such that $(P+1) = N$ (see Equation (2.26) below), one recovers the Lagrange Interpolant [39] and there is no empirical error. However, it is generally suggested to choose a basis such that $(P+1) \approx \frac{N}{2}$ [59, 39]. Alternatively, the basis may be selected using the E_{LOO} and other cross-validation procedures [111, 72] by iteratively growing the basis with Forward Selection [56].

Rather than perform regression with the full basis, $\mathcal{K}$, alternative approaches can also be used to select the basis polynomials as a subset of the initial basis. Two such approaches are least absolute shrinkage and selection operator (LASSO) [125] and least-angle-regression (LAR) [36]. We will specifically focus on the LAR in the following section.

2.5 Hybrid Least-Angle-Regression

Least-Angle-Regression [36] (LAR) is an iterative technique to estimate the solution of a linear system. Again consider the linear system for the PC expansion from Equation (2.15)

$$\boldsymbol{\mathcal{Z}}\boldsymbol{c} = \boldsymbol{f}$$

where the columns of $\boldsymbol{\mathcal{Z}}$ represent the basis functions in $\mathcal{K}$. LAR begins with an input basis $\mathcal{K}^{\mathcal{I}}$ and sets $c_{\boldsymbol{k}} = 0 \ \forall \ \boldsymbol{k} \in \mathcal{K}^{\mathcal{I}}$. It then iteratively builds the basis by successively activating basis functions. The procedure is briefly described as follows. See [137, 9, 36] for more detail.

- Set all coefficients to zero and calculate the residual, $\boldsymbol{r} = \boldsymbol{f} - \boldsymbol{\mathcal{Z}}\boldsymbol{c}$.
- Choose the basis ($m = 1$) that is most correlated with the residual and set the corresponding coefficient such that the residual is equally correlated with another basis ($m = 2$).
- Set the two basis coefficients such that they are equally correlated with a third ($m = 3$) basis.
- Continue to set $m - 1$ basis coefficients to be equally correlated with an m^{th} basis until $m = \min(P, N - 1)$ basis coefficients are "activated".

The LAR procedure results in m possible active bases of $\mathcal{K}^{\mathcal{I}}$ but the constant basis term, $\mathcal{K}_{\varnothing} = \{0, ..., 0\}$ is never activated. Instead of using the LAR coefficients, we follow a similar Hybrid-LAR procedure to that of [9] which is shown in [36] to improve performance. The Hybrid-LAR procedure is detailed below [9].

- Run the LAR procedure on $\mathcal{K}^{\mathcal{I}}$. Let $\mathcal{K}_i^{\mathcal{A}} \subset \mathcal{K}^{\mathcal{I}}$ be the active basis set prescribed by the LAR procedure at step i. Note that $|\mathcal{K}_i^{\mathcal{A}}| = i$ where $|\cdot|$ is the cardinality.

- For all $i = 1, ..., m$, perform the regular least-squares regression (2.17) on $\mathcal{K}_i^{\mathcal{A}} \cup \mathcal{K}_{\varnothing}$. Estimate error for basis $\mathcal{K}_i^{\mathcal{A}} \cup \mathcal{K}_{\varnothing}$ with E_{LOO}^* (2.18)
- Choose final PCE basis as $\mathcal{K} = \mathcal{K}_i^{\mathcal{A}} \cup \mathcal{K}_{\varnothing}$ that minimizes E_{LOO}^*

The Hybrid-LAR approach essentially uses LAR as a basis selection technique and then performs regression (from Section 2.4) on the active basis.

Unlike regular regression, LAR does not require that $(P_{in} + 1) = |\mathcal{K}^{\mathcal{I}}| \leqslant N$ since the final active basis will always be $\mathcal{K} \leqslant m = \min(P, N - 1)$. However, the choice of $\mathcal{K}^{\mathcal{I}}$ does affect the resulting $\mathcal{K}_i^{\mathcal{A}}$ and the final PCE fidelity. See Ref. [9, Section, 5.3.1] for an input basis selection procedure that enriches the input bases until an optimal one is found.

As with other non-NISP methods, there is not an exact experimental design associated with a basis. Again, see Ref. [9, Section, 5.3.2] for a suggested procedure based on random sampling. As we will see in Chapter 5, we may omit both the input-basis selection and experimental design as we will use alternative (NISP-based) approaches in combination with Hybrid-LAR.

There exist a few direct implementations of the LAR algorithm [70] for Matlab. Unless otherwise noted, all LAR solutions in this work were generated with a slightly modified version of the `SpaSM` [113] Toolbox. The modifications were to be able to easily recover computed results after the problem became unstable (due to rank-deficiency) and to perform the regression at each LAR step.

Inherently, the LAR procedure creates a sparser basis than the input-basis. Another solution methodology that exploits sparsity is compressed sensing.

2.6 Compressed Sensing

Compressed Sensing (CS) was developed in the signal processing, electrical and computer engineering, image processing, and medical imaging community [127, 12, 41].

The presumption is that on the appropriate basis, the spectrum of most signals are sparse. For example, image compression such as JPEG and JPEG-2000 exploit such sparsity on a discrete cosine transform, or discrete wavelet transform (respectively) [12]. Knowing the signal is sparse, one can use very few samples, a large "dictionary" of basis functions, and a solver designed to exploit/enforce sparsity, to reliably reconstruct the signal. The implications are wide ranging from compression to drastically improved medical imaging [41].

The compressed sensing for PCEs is again posed in matrix form:

$$\boldsymbol{\mathcal{Z}}\boldsymbol{c} = \boldsymbol{f}$$

where $\boldsymbol{\mathcal{Z}}$ (2.15) is the Vandermonde-type matrix for the respective basis and realizations locations. However, unlike regression, a larger basis, $\mathcal{K}$ (with $P+1 = |\mathcal{K}|$) is used to generate $\boldsymbol{\mathcal{Z}}$ such that $P+1 \gg N$. That is, the matrix problem is underdetermined and must be constrained in order to be well-posed. A simple and intuitive contraint is to minimize the energy in the system as [12]

$$\hat{\boldsymbol{c}} = \arg\min_{\boldsymbol{c}} ||\boldsymbol{c}||_{L_2} \quad \text{such that} \quad \boldsymbol{f} = \boldsymbol{\mathcal{Z}}\boldsymbol{c}. \tag{2.19}$$

The solution to (2.19) is given explicitly as [12]

$$\hat{\boldsymbol{c}} = \boldsymbol{\mathcal{Z}}^T \left(\boldsymbol{\mathcal{Z}}\boldsymbol{\mathcal{Z}}^T\right)^{-1} f.$$

However, the minimum-energy solution is often a poor choice; especially when $\boldsymbol{c}$ is sparse (that is the number of non-zero entries is much fewer than P). Changing the basis (and $\boldsymbol{\mathcal{Z}}$) also will fundamentally change the solution. The minimum-energy solution will be exact at each collocation point but will suffer over-fitting away from the collocation points.

Instead, one could enforce maximal sparsity of the system [15, 34, 127] by minimizing the ℓ_0 "norm"

$$||\boldsymbol{c}||_0 = \text{Number of non-zero } \boldsymbol{c} \text{ entries}$$

Note that the ℓ_0 norm is not a true norm [12] in the mathematical sense but is still useful in this context. The sparsity maximization problem is then posed as

$$\hat{\boldsymbol{c}} = \arg\min_{\boldsymbol{c}} ||\boldsymbol{c}||_0 \quad \text{such that} \quad \boldsymbol{f} = \boldsymbol{\mathcal{Z}}\boldsymbol{c} \tag{2.20}$$

While the solution of the sparse maximization problem is ideal, it also requires a combinatorial search and is computationally intractable for all but the sparsest of solutions [12, 34].

It has been shown, most notably by Donoho et al. [32, 31], that the solution of the ℓ_1 minimization is equivalent to the "ℓ_0-norm" under certain conditions (See Refs. [12, 31, 35, 127]). Specifically, the problem can be recast as

$$(P_1): \quad \min ||\Lambda\boldsymbol{c}||_1 \;\; s.t \;\; \boldsymbol{\mathcal{Z}}\boldsymbol{c} = \boldsymbol{f} \tag{2.21}$$

In the case of noisy data, an alternative formulation of (2.21) is:

$$(P_{1,\delta}): \quad \min ||\Lambda\boldsymbol{c}||_1 \;\; s.t \;\; ||\boldsymbol{\mathcal{Z}}\boldsymbol{c} - \boldsymbol{u}||_{L_2} \leqslant \delta \tag{2.22}$$

where $||\boldsymbol{c}||_{\ell_1} \equiv \sum_{\boldsymbol{k}\in\mathcal{K}} |c_{\boldsymbol{k}}|$. Equations (2.21) and (2.22) are known as Basis-Pursuit (BP) and Basis-Pursuit-Denoising (BPDN) respectively. In addition, (2.22) may be reformulated as

$$\hat{\boldsymbol{c}} = \arg\min_{\boldsymbol{c}} \left(\frac{1}{2}||\boldsymbol{\mathcal{Z}}\boldsymbol{c} - \boldsymbol{f}||_2 + \lambda||\Lambda\boldsymbol{c}||_{\ell_1} \right)$$

where λ is related to a Lagrange multiplier. The relation between λ and δ is not known *a priori* except in very specific cases [129]. The ℓ_1-regularization problem is convex and easily solvable with linear programming [12].

In (2.21) and (2.22), Λ is a diagonal weight matrix. The choice of the weight matrix has great importance [12] and is an area of active research [53, 99]. The authors of [35] suggest using Λ_{kk} as the L_2 norm of the k^{th} column of $\boldsymbol{\mathcal{Z}}$. This approximates the L_2 norm of the Ψ_k with the chosen colocation points. Recall that

we use a *normalized* basis for the PCE (2.5) and therefore $\Lambda = 1$. However, a choice of $\Lambda = 1$ also means that the CS algorithm will add energy to high-order modes without penalty. Our numerical tests suggest that this results in severe over-fitting. Instead, we choose Λ such that

$$\max_{\boldsymbol{\xi} \in \Xi} |\Lambda_{kk} \Psi_k(\boldsymbol{\xi}) \rho(\boldsymbol{\xi})| = 1 \quad \forall \quad k = 0, ..., P \tag{2.23}$$

Equation (2.23) restricts the basis functions to the range $[-1, 1]$ with the appropriate integral measure of the PCE function space. The solution of (P_1) and $(P_{1,\delta})$ with a given Λ is equivalent to setting $\boldsymbol{\mathcal{Z}}' = \Lambda \boldsymbol{\mathcal{Z}}$ and solving (P_1) and $(P_{1,\delta})$ without Λ. As such, for the case of uniform measure and Legendre Polynomials, the choice of Λ from Equation (2.23) is equivalent to projecting on the "Traditional" (non-normalized) Legendre Polynomials like those plotted in Fig. 2.1 and discussed in [76, Appendix B.1.1]. In the Legendre case, this weighting penalizes higher-order basis functions and pushes the solution towards low-order terms. This choice of weighing is similar to that of [53].

The choice of δ in Equation (2.22) is also an important consideration. A δ too small may cause over-fitting of the data and a δ too large will reduce the final PCE fidelity [35]. We use a non-exhaustive, random-subsampling cross-validation routine, adapted from [35], to estimate the error of the PCE for a given δ presented in Algorithm 2.2. In addition to providing a means to select δ, the CV error is useful for additional analysis. We note that different authors have suggested using the mean of each estimated error instead of the RMS [35, 111].

Much theoretical research has been performed on the convergence of Equation (2.22). In traditional CS literature, the Vandermonde Matrix, $\boldsymbol{\mathcal{Z}}$, termed the "measurement matrix" is often fixed [27]. Regardless, certain properties of $\boldsymbol{\mathcal{Z}}$ must be fulfilled. These may be measured via the spark [32, 12], the restricted-isometry-property [136, 123], or the mutual-coherence [64, 35, 12]. All measures essentially

Algorithm 2.2 BPDN CV Error Estimate $(\boldsymbol{\xi}, f, \mathcal{K}, N_r, N_v, \delta, n)$.

$N = N^r + N^v$
for $i = 1, \ldots, n$ **do** ▷ n permutations of $(\boldsymbol{\xi}^r, f^r)$ and $(\boldsymbol{\xi}^v, f^v)$
  Randomly divide $\boldsymbol{\xi}, f$ into $(\boldsymbol{\xi}^r, f^r)$ and $(\boldsymbol{\xi}^v, f^v)$
  $\delta_r = \sqrt{\frac{N^r}{N}}\delta$
  $\boldsymbol{\mathcal{Z}}^r_{j\boldsymbol{k}} = \Psi_{\boldsymbol{k}}(\boldsymbol{\xi}_j) \quad \forall\ \boldsymbol{\xi}_j \in \boldsymbol{\xi}^r,\ \boldsymbol{k} \in \mathcal{K}$
  $\boldsymbol{\mathcal{Z}}^v_{j\boldsymbol{k}} = \Psi_{\boldsymbol{k}}(\boldsymbol{\xi}_j) \quad \forall\ \boldsymbol{\xi}_j \in \boldsymbol{\xi}^v,\ \boldsymbol{k} \in \mathcal{K}$
  $\hat{\boldsymbol{c}} = \arg\min_{\boldsymbol{c}} \|\Lambda \boldsymbol{c}\|_1 \quad s.t. \quad \|\boldsymbol{\mathcal{Z}}^r \boldsymbol{c} - f^r\|_2 \leqslant \delta_r$ ▷ Estimated BPDN Solution
  $E_i = \sqrt{\frac{N}{N_v}} \|\boldsymbol{\mathcal{Z}}^v \hat{\boldsymbol{c}} - f^v\|$ ▷ Scaled error
end for
Error= RMS($\boldsymbol{E}$) ▷ RelativeError= $\frac{\text{Error}}{\|f\|}$

seek to quantify the uniqueness of each column of $\boldsymbol{\mathcal{Z}}$. For example, mutual coherence measures the largest (normalized) inner-product between any two columns of the matrix. These measured quantify the probability of the BP(DN) solution converging to the sparsest (2.20) solution. For a discussion of sparsity requirements for BPDN see [35, 136, 123].

There are a wide variety of MATLAB packages to solve the BPDN problem including `SPGL1` [128, 129], `l1-Magic` [16] and `SparseLab` [33]. Unless otherwise noted, all BP and BPDN solutions in this work were recovered with `SPGL1` though often validated against the other solvers.

In addition, the basis pursuit optimization problem may be solved with a Bayesian formalism. Bayesian methods have been used for PCE reconstruction [105], however the method can be expanded for CS reconstruction. The Bayesian formulation for CS uses a sparseness prior, such as the Laplace Density Function, or other hierarchical priors [65]. The coefficients are then chosen as the maximum *a posteriori* value. The key advantage to the Bayesian approach is the ability to determine an error estimate on the spectral coefficients which is not possible with BP and BPDN solvers [54, 65].

2.7 Collocation and Interpolation

Collocation methods seek to develop an interpolant that, unlike projection methods, are exact at the nodal (collocation) points. Polynomial interpolation is carried out through the tensor product of 1D Lagrange Polynomials. However, unlike PCE methods where a basis is specified, for collocation methods, the points themselves specify the basis.

For $\{\xi_i, f_i = F(\xi_i)\}_{i=1}^n$ collocation points and function evaluations in 1D, there are n basis functions of the form [101, 76]

$$\mathcal{L}_i = \prod_{\substack{j=1 \\ j \neq i}}^{n} \frac{\xi - \xi_j}{\xi_i - \xi_j}, \tag{2.24}$$

and then the interpolating function is

$$F^{\mathcal{L}}(\xi) = \sum_{i=1}^{n} f_i \mathcal{L}_i(\xi). \tag{2.25}$$

It is also immediate to verify that

$$\mathcal{L}_i(\xi_j) = \delta_{ij}$$

where δ_{ij} is the Kronecker delta function. Therefore $F^{\mathcal{L}}(\xi_i) \equiv f_i$ and Equation (2.25) is an exact interpolant at the collocation points.

The interpolation may be extended into multiple dimensions with a tensor product construction [109]. Additionally, multiple dimensional rules may be combined with sparse grids (similar to Equation (2.8)) [95, 96, 63]. Note that there is no projection step to a Lagrange interpolant; the coefficient is exactly the functional value.

For a *structured*, colinear grid (such as those from full-tensor or sparse quadratures), the PCE solution on the following basis, may be factored into the Lagrangian

basis [39]. For every full-tensor $\mathbf{P}_{\boldsymbol{l}}^{\mathtt{FT}}$ $\left(\text{or } \mathbf{Q}_{\boldsymbol{l}}^{\mathtt{FT}}\right)$ $\forall\ \boldsymbol{l} \in \mathscr{L}$, there are $n_{\boldsymbol{l}} = \prod_{i=1}^{d} n_{l_i}$ quadrature nodes where n_{l_i} is the number of 1D quadrature nodes for level l_i with the given quadrature rule. We then define

$$\mathcal{K}^{\mathcal{L}} = \bigcup_{\boldsymbol{l} \in \mathscr{L}} \{n_{l_1}, ..., n_{l_d}\} - \mathbf{1} \tag{2.26}$$

where $\mathbf{1} \doteq \{1, ..., 1\}$. The final result is that for $N = |\mathcal{G}(\mathscr{L})|$ quadrature nodes, $(P+1) = |\mathcal{K}^{\mathcal{L}}| \equiv N$. The resulting Vandermonde matrix $\boldsymbol{\mathcal{Z}}$ (2.16) is square and the matrix-formulation of the PCE problem (2.15) is directly invertible. The resulting polynomial is unique and can be factored into a Lagrange basis.

Collocation methods are particularly sensitive to the selection of collocation points. A poor choice of points will result in severe Runge Phenomenon [106, 42], where the interpolant oscillates severely near the bounds of the domain and is particularly burdensome for equally-distant points [103]. The susceptibility of a distribution of colocation points for Lagrangian interpolation may be quantified through the Lebesgue constant [13, 50] which measures the maximum amplitude of the interpolant basis. Gauss-Legendre and Chebyshev nodes are particularly well suited to interpolation as they have more collocation points near the boundaries and smaller Lebesgue constants [76]. Chebyshev nodes are also nested making them amenable to sparse grids and the reuse of prior model evaluations. An alternative approach is to use Leja Sequences [82] which have the property of being nested, have a small Lebesgue constant, and being amendable to arbitrary domains [28]. Leja sequences may also be defined for any n and not follow a typical $n_\ell = 2^\ell - 1$ scale.

2.8 Conclusion

In this chapter, we introduced PCEs and noted a few common non-intrusive methods to determine the PCE coefficients. Non-intrusive methods are particularly attractive

for the investigation of real-world models with complex inputs and allow the use of legacy solvers. These methods are agnostic to the model itself and only require a deterministic output for a set of inputs (PCEs for non-deterministic model solutions are the basis of other work [105]). In the following chapters, we will implement and extend the methods presented here to more efficiently produce high-fidelity PCE surrogates of interesting models including ocean-circulation and chemical kinetics on sparse grids.

3

A Priori Testing of Sparse Adaptive PCEs on an OGCM Database

3.1 Introduction

In prior work by *Alexanderian, et al.* [2], we relied on a classical Smolyak sparse grid tensorization [116] (2.9) and DSP (see Section 2.3.1) to study the HYCOM simulated sea surface temperature's (SST) dependence on the combined uncertainties in subgrid scale mixing and wind drag parameters. To obtain accurate statistics, we [2] relied on a systematic isotropic refinement of the sparse grid, namely by successively increasing the resolution level along all dimensions of the random parameter space. As further described below, this necessitated the generation of a relatively large database involving hundreds of realizations.

The present study is motivated by the desire to explore, in the context of short-duration Ocean General Circulation Model (OGCM) predictions under extreme forcing events, whether adaptive sparse quadrature can afford the construction of PCE surrogates with equal or improved fidelity to isotropic refinement, based on a substantially smaller number of realizations, and consequently at a small fraction of

the computational cost. We are also motivated by ongoing investigations [120] of the oceanic circulation in larger domains than was considered in our previous GOM analysis [2], and for longer simulation times. Since the cost of such extended analyses effectively limits the number of realizations that can be simulated, it proved highly desirable to analyze the performance of adaptive sampling schemes prior to launching them to dynamically sample large-scale computations. Consequently, in this chapter we exploit the pre-existing database to investigate the properties of a recent pseudo-spectral algorithm [25, 22] that is based on a direct application of the Smolyak's sparse grid formulas. The advantage of this generalized approach is that it affords the use of arbitrary admissible sparse grids while maximizing the number of polynomial coefficients that can be computed without internal aliasing.

The layout of this chapter is as follows. In Section 3.2 we discuss the computational model and the previously computed database. We also discuss the enrichment of the pre-existing database in light of the global sensitivity analysis in [2]. In Section 3.3 we introduce the Smolyak pseudo-spectral projection method and apply it to the realization database. In Section 3.4, we analyze the performance of the adaptive algorithms and demonstrate an order-of-magnitude reduction in required realizations. Finally, in Section 3.5 we discuss potential improvements on our methodology and applications to other problems.

3.2 HYCOM Database

The interest in improving the quadrature sampling of PCEs came about during a global sensitivity analysis of the uncertainty in subgrid mixing and wind drag parameters of an ocean general circulation model (Hybrid Coordinate Ocean Model, HYCOM) [2] . Here we summarize this prior work, and motivate the need for more effective sampling strategies by highlighting the drawbacks of isotropic refinement.

The work in [2] revolved around studying the impact of four uncertain param-

Table 3.1: The random input parameters for HYCOM. Here $\mathcal{U}[a_i, b_i]$ designates a uniformly distributed probability density function over the interval $a_i \leqslant \theta \leqslant b_i$; the canonical stochastic variables $|\xi_i| \leqslant 1$ are then defined as $\theta_i(\xi_i) = \frac{a_i+b_i}{2} + \frac{b_i-a_i}{2}\xi_i$.

Parameter	Description	Distribution
θ_1	critical Richardson number	$\mathcal{U}(0.25, 0.7)$
θ_2	background viscosity (m^2/s)	$\mathcal{U}(10^{-4}, 10^{-3})$
θ_3	background diffusivity (m^2/s)	$\mathcal{U}(10^{-5}, 10^{-4})$
θ_4	stochastic wind drag coefficient	$\mathcal{U}(0.2, 1.0)$

eters on the simulated oceanic response to Hurricane Ivan 2004. Three parameters were connected with the mixed-layer parameterization and one with the wind drag coefficient; the uncertainty of these four parameters are listed in Table 3.1. The main output uncertainties, or quantities of interest (QoIs), correspond to averaged value of sea surface height (SSH), sea surface temperature (SST) and mixed-layer depth (MLD), within a geographic region whose bottom-left and upper right corners are at 90W/20N and at 82W/30N, respectively; in addition, the heat-flux (Q_{tot}) in a 150-km region surrounding the observed center of Ivan was also investigated. The computational grid and the averaging region are plotted in Fig. 3.1. The HYCOM

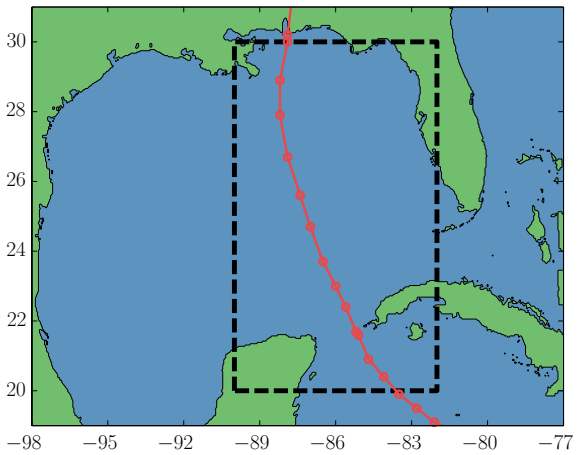


FIGURE 3.1: HYCOM GOM grid area and averaging region with bottom-left and upper right corners are at 90W/20N and at 82W/30N respectively. Plotted in red is the storm-track of Hurricane Ivan.

grid covered the GOM and a large portion of the Caribbean Sea at a resolution of $1/25^\circ \approx 4km$ and 20 vertical layers; the simulation spanned the entire transit of Ivan through the Gulf of Mexico from September 9 to September 14, 2004. Each realization required 2.5 hours on an 8-core node and required 764 megabytes of storage. This large computational burden is the primary motivation behind the present work.

The PCE basis consisted of tensor products of Legendre polynomials and the series was truncated isotropically at fifth order in each direction. The numerical quadrature consisted of a classical Smolyak sparse grid based on the one-dimensional Gauss Kronrod Patterson rule [73]. An ensemble of 385 realizations was required to perform DSP and recover polynomials of total-order $\leqslant 5$. This represented a substantial amount of computational time and storage given the grid's spatial resolution and extent. The adequacy of the PCE representation was checked via a number of error metrics including a point-wise error and the convergence of the PDF for increased quadrature levels [2]. Global sensitivity analysis revealed that the uncertainty in the QoIs was dominated by the third and fourth parameters (the background diffusivity and wind drag coefficient respectively) whereas the first two contributed little; see Fig. 3.2 for a comparison of total-sensitivity and sensitivity of each grouping of input parameters and time. We note that the variance due to ξ_3 is relatively constant, however as the variance due to ξ_4 grows, the fraction of variance associated with ξ_3 diminishes.

Additional insight into the response of the oceanic circulation to the uncertain parameters can be gained from Fig. 3.3, which depicts projections of the box-averaged SST onto the axes of the random parameter space. The plots reveal an almost linear dependence of SST on ξ_1, ξ_2 and ξ_3 (panels a, b and c, respectively), which suggests that a lower-order PCE along these dimensions would be sufficient. In contrast, Fig. 3.3(d) reveals a highly curved response for higher values of ξ_4, which indicates that a higher order expansion is required to suitably represent it.

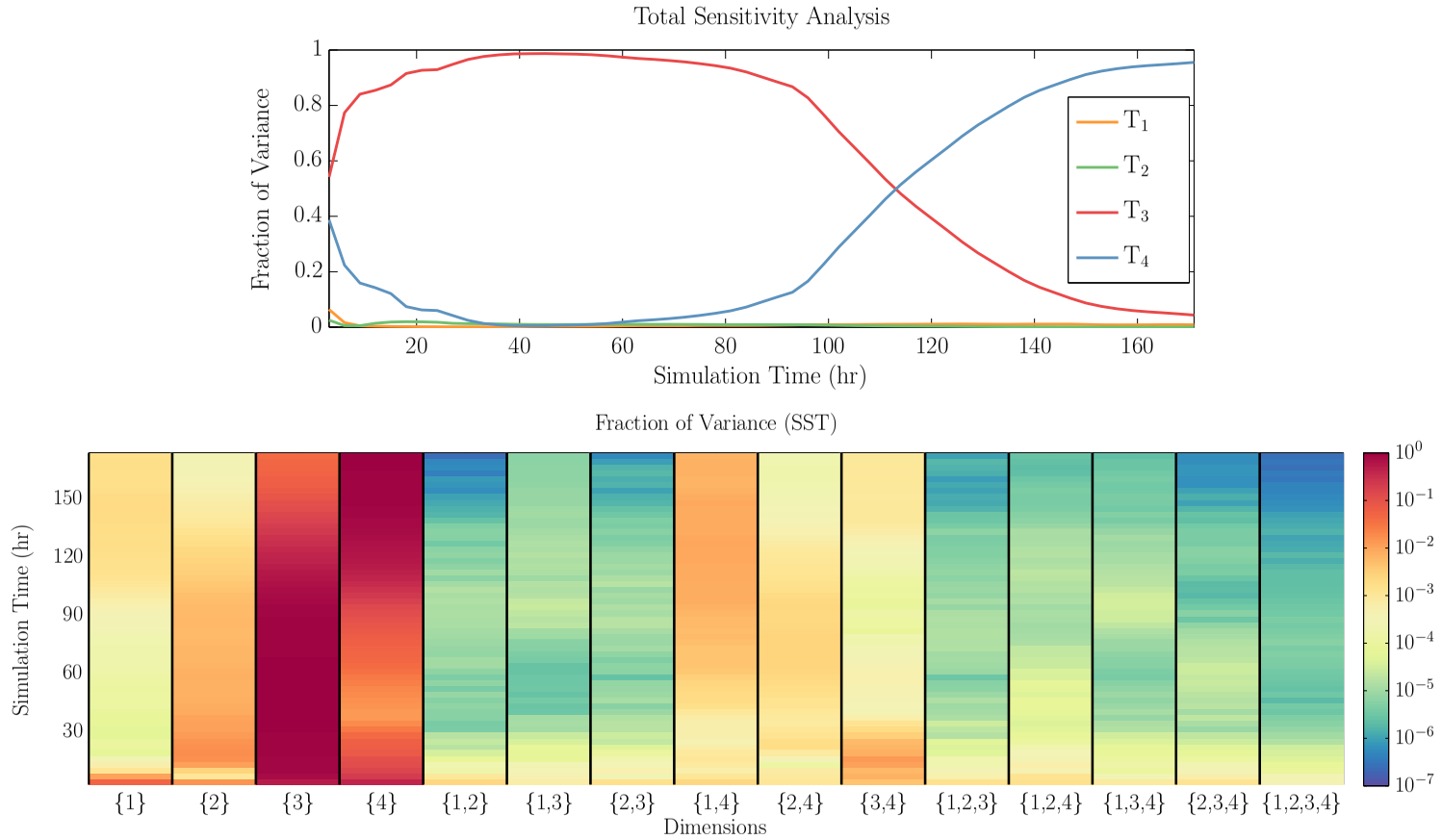


FIGURE 3.2: Time evolution of sea surface temperature global sensitivity indices. The background diffusivity dominates until about hour 80, when Ivan enters the averaging box, whereas the wind drag coefficient becomes the dominant contributor at later times. (top) Total sensitivities, (bottom) contribution of each grouping of dimension.

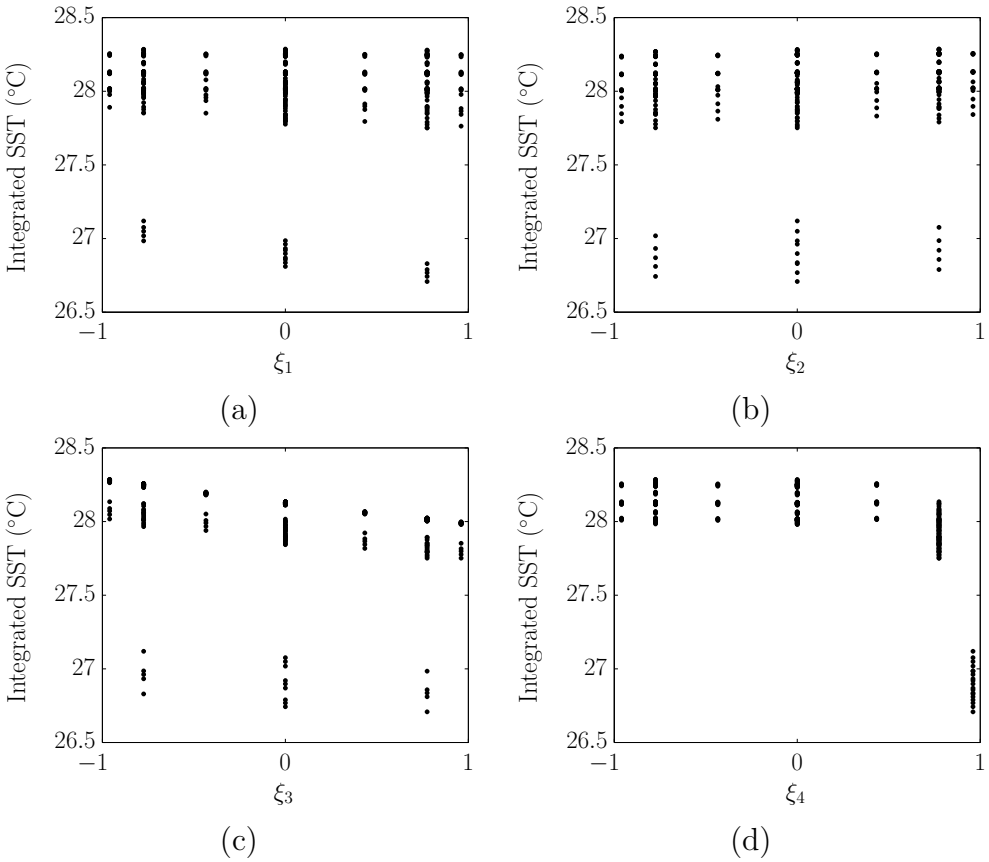


FIGURE 3.3: Projections of the box-averaged SST at $t = 150$ hr. on canonical axes of the random parameter space: (a) ξ_1, (b) ξ_2, (c) ξ_3, and (d) ξ_4.

Figure 3.4 shows the coefficients, c_k, labeled by the corresponding multi-index; the zeroth coefficient is excluded. As expected from Fig. 3.3, the only significant coefficient involving ξ_3 is the linear term. Apart from this term, the coefficients having significant amplitude involve monomials of ξ_4. The slow decay of coefficients involving ξ_4 also highlights the need for increased resolution along that axis.

In preparation for the analysis of adaptive expansions, additional computations were performed using the same model and stochastic setting considered in [2]. We also generated a separate database consisting of 256 member Latin Hypercube Sampling (LHS) [92] ensemble. This was deemed desirable for the purpose of evaluating the quality of the PCE representations based on a set of realizations that is indepen-

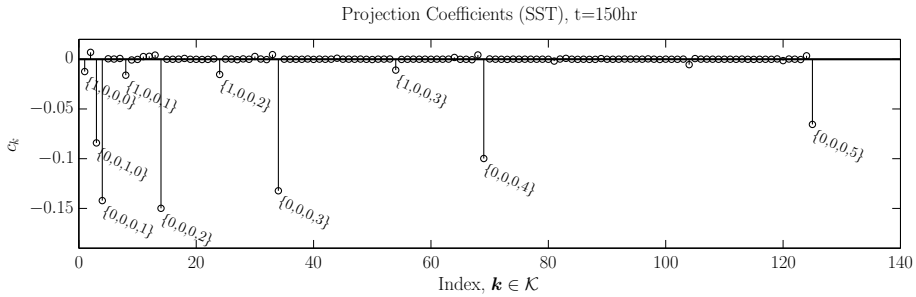


FIGURE 3.4: Coefficents of the box-averaged SST at at $t = 150$ hr. An isotropic fifth total order expansion is used. The coefficients with the largest amplitudes are labeled with the corresponding multi-index representing the polynomial order along each stochastic direction. The zeroth term is omitted for clarity.

Table 3.2: Table of anisotropic series experiments. In the first column, $\boldsymbol{p}$ is a vector representing the polynomial order of accuracy along the individual dimensions, the second column represents the size of the resulting basis, and the third column indicates the number of realizations in the Smolyak sparse grid. The polynomial basis is described using the multi-index set in Appendix C, Equation (C.2).

Simple Truncation	P	Number of realizations
$\boldsymbol{p} = (5, 5, 5, 5)$	126	385
$\boldsymbol{p} = (5, 5, 7, 7)$	168	513
$\boldsymbol{p} = (2, 2, 5, 5)$	36	73
$\boldsymbol{p} = (2, 2, 7, 7)$	59	169

dent from that used for their construction.

Motivated in large part by the results of the global sensitivity analysis, a preliminary assessment of adaptive refinement was conducted, namely by enriching the isotropic database generated in [2] preferentially along the third and fourth dimensions. Specifically, the database was enriched by including additional levels in the Smolyak quadrature, in such a way as to enable us to accurately capture polynomials of order $\leqslant 5$ in the first two variables, and $\leqslant 7$ in the third and fourth variables.

Using the resulting anisotropic data, we analyzed the behavior of the solution for the different anisotropic truncations listed in Table 3.2. In particular, we used the relative L_2 error between the corresponding PC representations and the LHS

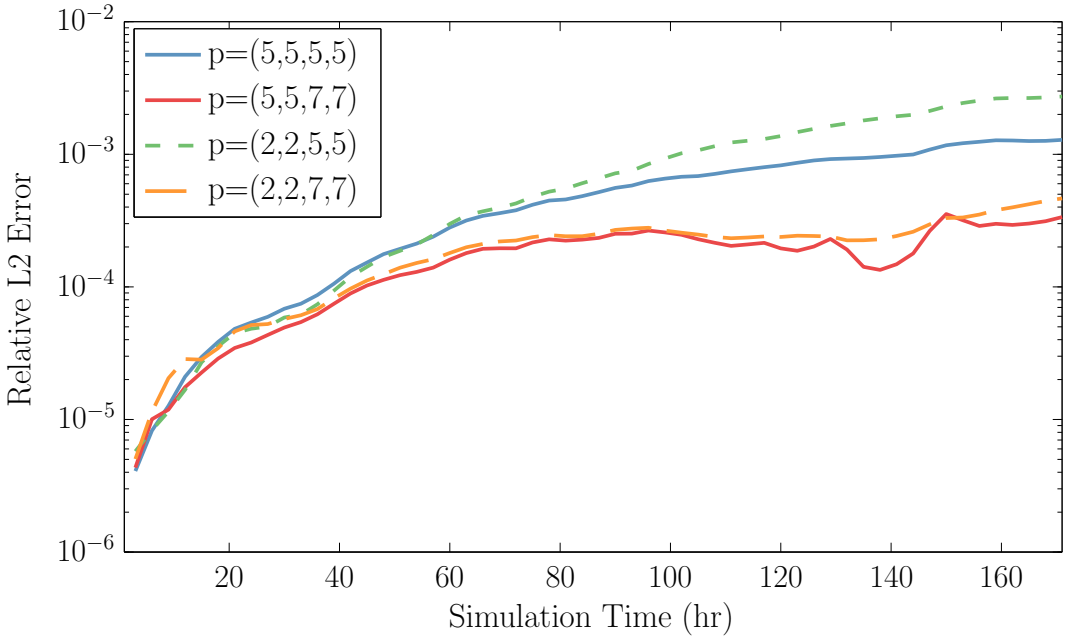


FIGURE 3.5: Relative L_2 error between the area-averaged SST and the LHS sample. Curves are generated for different anisotropic refinement levels, as indicated.

ensemble as a fidelity metric. Since the relative importance of individual components of $\boldsymbol{\xi}$ may be time-dependent, we plot in Fig. 3.5 the time evolution of this metric for different refinement levels. The results indicate that the PC representations which include higher order terms in the first two dimensions perform marginally better than those with a lower order truncation, whereas a substantial reduction in the error can be observed at later times when the PCE is enriched to seventh order in the third and fourth directions. This suggests that a coarse resolution in the first two directions can be safely adopted, and that using adaptive refinement, a suitable representation can be obtained at a substantially reduced cost compared to isotropic refinement.

In the following, we will investigate how the adaptive pseudo-spectral algorithm can be used to systematically refine the sampling of the stochastic parameter space and the corresponding PCE representation, and to quantify the performance gains thus achieved. The enriched Smolyak set with $\boldsymbol{p} = (5, 5, 7, 7)$ and 513 realizations

will be specifically used as a database for the *a priori* tests.

3.3 Smolyak Pseudo-Spectral Projections

A disadvantage of the dimensional refinement scheme above is that it is relatively inefficient, potentially leading to the inclusion of a large number of realizations with every refinement step. Another disadvantage concerns the associated truncation scheme which, in order to avoid internal aliasing severely limits the number of polynomials included in the truncated basis. Internal aliasing occurs when the quadrature is not sufficiently accurate to maintain discrete orthogonality between all basis functions. As discussed in Fig. 2.2(b), plotted is the full-exactness and half-exactness set for a "traditional" Smolyak quadrature. Notice if $\boldsymbol{k} = \{0, 11\}$ and $\boldsymbol{k}' = \{11, 0\}$, the inner-product

$$\langle \Psi_{\boldsymbol{k}}, \Psi_{\boldsymbol{k}'} \rangle \approx \langle \Psi_{\boldsymbol{k}}, \Psi_{\boldsymbol{k}'} \rangle_{\mathbf{Q}_{\mathscr{L}}} = \sum_{i=1}^{N} \Psi_{\boldsymbol{k}} \left(\boldsymbol{\xi}^{(i)} \right) \Psi_{\boldsymbol{k}'} \left(\boldsymbol{\xi}^{(i)} \right) w_i$$

will not integrate exactly to zero since $\Psi_{\boldsymbol{k}=\{0,11\}} \Psi_{\boldsymbol{k}'=\{11,0\}}$ has the highest order term $\boldsymbol{\xi}_1^{11} \boldsymbol{\xi}_2^{11}$ and is outside the polynomial exactness of the quadrature.

An attractive avenue to overcome these drawbacks is provided by the adaptive pseudo-spectral algorithm recently proposed in [22]. In this section, we will first outline its construction and then evaluate its performance based on a priori tests that exploit the HYCOM database described in the previous section.

3.3.1 Sparse Pseudo-Spectral Projection

The sparse pseudo-spectral projection (PSP) method relies on the definition of a sequence of *1-D* projection operators [25, 22]. Let $\{\mathrm{P}_{l \geqslant 1}\}$ denote the sequence of 1-D

projection operators associated to the sequence $\{\mathrm{Q}_{l\geqslant 1}\}$ of 1-D quadrature, where

$$\mathrm{P}_l : F(\xi) \mapsto \mathrm{P}_l F(\xi) \equiv \sum_{k\in\mathcal{K}^l} c_k \psi_k(\xi)$$

$$c_k = \mathrm{Q}_l(F\psi_k) = \sum_{q=1}^{N(l)} F(\xi^{(q)})\psi_k(\xi^{(q,l)}) w_{1D}^{(q,l)}. \tag{3.1}$$

Where in 1D, $\mathcal{K}^l = \mathcal{K}^{*\boldsymbol{l}}$ (2.13) uniquely defines the PCE multi-index. Note that $\mathrm{P}_{l\geqslant 1}$ is free of internal aliasing, owing to the dependence with l of the projection spaces, which ensures that the quadrature is sufficiently exact.

Consider for $l \geqslant 1$ the difference of successive 1-D projection operators,

$$\Delta^{\mathrm{P}}_{l=1} = \mathrm{P}_1, \quad \Delta^{\mathrm{P}}_{l>1} = \mathrm{P}_l - \mathrm{P}_{l-1},$$

such that $\mathrm{P}_l = \sum_{i=1}^{l} \Delta^{\mathrm{P}}_i$. The Full-Tensor (`FT`) projection operator in (2.14) can be recast in terms of a sum of tensorized difference projection operators as follows

$$\begin{aligned}
\mathbf{P}^{\mathtt{FT}}_{\boldsymbol{l}} F &= \left(\mathrm{P}_{l_1} \otimes \cdots \otimes \mathrm{P}_{l_d}\right) F \\
&= \left(\left(\sum_{i_1=1}^{l_1} \Delta^{\mathrm{P}}_{i_1}\right) \otimes \cdots \otimes \left(\sum_{i_d=1}^{l_d} \Delta^{\mathrm{P}}_{i_d}\right)\right) F \\
&= \sum_{i_1=1}^{l_1} \cdots \sum_{i_d=1}^{l_d} \left(\Delta^{\mathrm{P}}_{i_1} \otimes \cdots \otimes \Delta^{\mathrm{P}}_{i_d}\right) F \\
&= \sum_{\boldsymbol{i}\in\mathscr{L}^{\mathtt{FT}}_{\boldsymbol{l}}} \left(\Delta^{\mathrm{P}}_{i_1} \otimes \cdots \otimes \Delta^{\mathrm{P}}_{i_d}\right) F.
\end{aligned} \tag{3.2}$$

The sparse PSP operator $\mathbf{P}_{\mathscr{L}}$ is finally obtained by considering a summation over an admissible subset $\mathscr{L} \subset \mathscr{L}^{\mathtt{FT}}_{\boldsymbol{l}}$ of tensorized difference projection operators. This results in

$$\mathbf{P}_{\mathscr{L}} F = \sum_{\boldsymbol{i}\in\mathscr{L}} \boldsymbol{\Delta}^{\mathrm{P}}_{\boldsymbol{i}} F, \quad \boldsymbol{\Delta}^{\mathrm{P}}_{\boldsymbol{i}} F \doteq \left(\Delta^{\mathrm{P}}_{i_1} \otimes \cdots \otimes \Delta^{\mathrm{P}}_{i_d}\right) F. \tag{3.3}$$

The key point here is that the sparse PSP operator in (3.3) involves a telescopic sum of differences in the projection of F onto subspaces of increasing dimensions, where

each individual (FT) tensorized quadrature projection is evaluated *without internal aliasing*. As a result, the sparse PSP in (3.3) is free of internal aliasing. Further, $\mathbf{P}_{\mathscr{L}} F$ belongs to the span of $\{\Psi_{\boldsymbol{k}}, \boldsymbol{k} \in \mathcal{K}(\mathscr{L})\}$, where

$$\mathcal{K}(\mathscr{L}) = \bigcup_{\boldsymbol{l} \in \mathscr{L}} \mathcal{K}^{*\boldsymbol{l}} = \mathcal{K}^{*\mathscr{L}}$$

Unlike DSP, PSP is able to project onto the *largest* admissible set of the final quadrature (the so-called half-accuracy set). We have $\mathcal{K}(\mathscr{L}) \supseteq \mathcal{K}^*(\mathscr{L})$ and the projection space of the sparse PSP is usually significantly larger than that of the DSP, while the two methods have the same complexity relying both on the same sparse grid $\mathcal{G}(\mathscr{L})$. Note, however, that the global sparse quadrature rule $\mathbf{Q}_{\mathscr{L}}$ is generally not sufficiently accurate with respect to $\mathcal{K}(\mathscr{L})$. The inclusion of the sets $\mathcal{K}(\mathscr{L})$ and $\mathcal{K}^*(\mathscr{L})$ is illustrated in Fig. 3.6. We note that $\mathcal{K}(\mathscr{L})$ (Direct Projection Polynomials) is *not* uniquely defined.

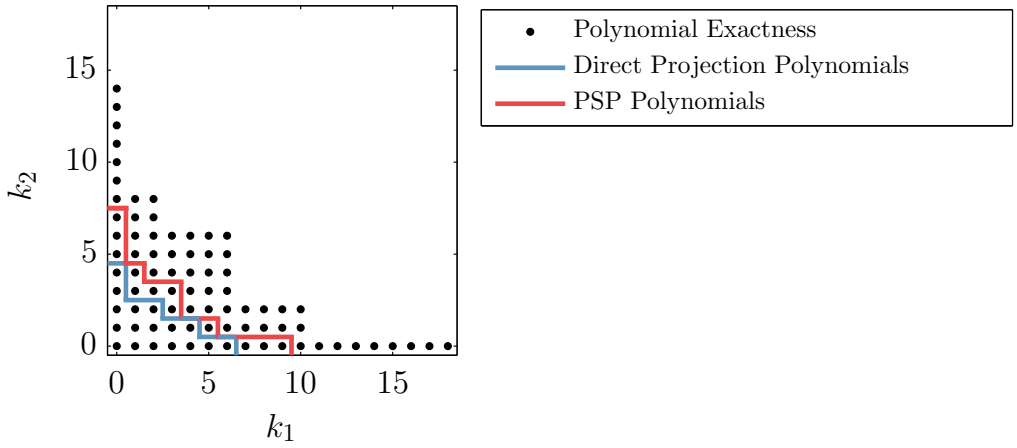


FIGURE 3.6: Comparison of polynomials multi-index sets for DSP and PSP using a sparse grid $\mathcal{G}(\mathscr{L})$ in $d = 2$ dimensions. Points are the exactness set; multi-indices (k_1, k_2) such $\Psi_{\boldsymbol{k}}(\boldsymbol{\xi}) = \psi_{k_1}(\xi_1)\psi_{k_2}(\xi_2)$ is exactly integrated by $\mathbf{Q}_{\mathscr{L}}$. Multi-indices of the DSP set $\mathcal{K}^*(\mathscr{L})$ are below the blue line, while elements of the PSP set $\mathcal{K}(\mathscr{L})$ are below the red-line. The DSP (blue) polynomial set is not unique.

3.3.2 Adaptive Pseudo-Spectral Projection

The PSP method does not, in itself, guarantee a low projection error if the projection space is not rich enough (external aliasing). To address this issue, we adapt the method proposed in [46] to the context of the sparse PSP. The adaptive PSP (aPSP) method was first introduced in [22, 134] and subsequently used in [120, 61] in the context of Bayesian inference and experimental design. The algorithm is explained in detail in [22, 61] and is briefly outlined below and also presented in Algorithm 3.1.

We begin by defining two distinct multi-index sets, the set $\mathcal{O}$ of old multi-indices and the set $\mathcal{A}$ of active multi-indices, such that $\mathscr{L} = \mathcal{O} \cup \mathcal{A}$ is admissible, and $\mathcal{A}$ is $\mathcal{O}$-admissible, that is

$$\forall \boldsymbol{l} \in \mathcal{A}, \quad l_i > 1 \Rightarrow \boldsymbol{l} - \hat{\mathsf{e}}_i \in \mathcal{O}, \quad \text{for } i = 1, \cdots, d.$$

The set $\mathcal{A}$ contains the multi-indices available for adaption. Associated with each $\boldsymbol{l} \in \mathcal{A}$ is a refinement indicator $\epsilon(\boldsymbol{l}) \geqslant 0$. In view of (3.3), a natural measure for the contribution of a tensorization $\boldsymbol{l} \in \mathscr{L}$ to the projection is the $L_2(\Xi, \rho)$ norm of the associated tensor-product-delta-operator; we take

$$\epsilon(\boldsymbol{l}) \doteq \frac{1}{\beta} \left\| \left(\Delta_{l_1}^{\mathrm{P}} \otimes \cdots \otimes \Delta_{l_d}^{\mathrm{P}} \right) F \right\|_{L_2(\Xi,\rho)}. \tag{3.4}$$

where β is an appropriately chosen normalization constant. The indicator $\epsilon(\boldsymbol{l})$ is then the L_2-norm of the projection surplus associated to $\boldsymbol{l}$. At each iteration of the adaptive algorithm, the *critical* multi-index $\boldsymbol{l}^* \in \mathcal{A}$ with the highest indicator $\epsilon(\boldsymbol{l})$ is selected. $\boldsymbol{l}^*$ is removed from $\mathcal{A}$ and added to $\mathcal{O}$. Then, each of the d multi-indices of the forward neighborhood of $\boldsymbol{l}^*$,

$$\mathcal{F}(\boldsymbol{l}^*) \doteq \{\boldsymbol{l}_i^+ = \boldsymbol{l}^* + \hat{\mathsf{e}}_i, i = 1, \cdots, d\}, \tag{3.5}$$

is added to $\mathcal{A}$ provided that $\mathscr{L} \cup \{\boldsymbol{l}_i^+\}$ remains $\mathcal{O}$-admissible. $\boldsymbol{l}^+$ is the candidate index and $\hat{\mathsf{e}}_i$ is the canonical unit vector in the i^{th} direction. Note that none of the

forward neighbors of $\boldsymbol{l}^*$ may be admissible to complete $\mathcal{A}$. When new multi-indices are added to $\mathcal{A}$, the sparse grid $\mathcal{G}(\mathcal{O} \cup \mathcal{A})$ and projection $\mathbf{P}_{\mathcal{O}\cup\mathcal{A}}F$ are updated (involving model evaluations at the new grid points only), and the indicators $\epsilon(\boldsymbol{l})$ of the new multi-indices are computed. The adaptive procedure can then be repeated.

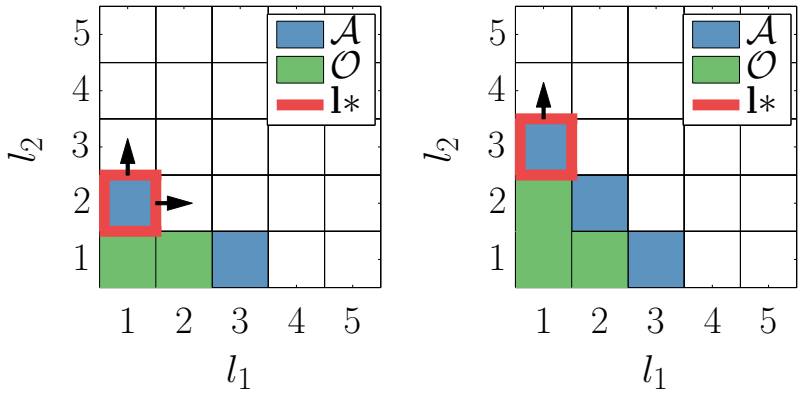
A schematic example in presented in Fig. 3.7 where the critical index, $\boldsymbol{l}^*$ is highlighted in red and the black arrows note the forward neighbors to be added. Notice that from Fig. 3.7(b) to Fig. 3.7(c), $\boldsymbol{l}^+ = \{2, 3\}$ is not added since it is not $\mathcal{O}$-admissible at that step however the index is later added in Fig. 3.7(d). In Fig. 3.7(d), the critical index is now $\boldsymbol{l}^* = \{2, 3\}$ but does not admit any forward neighbors and is still moved to $\mathcal{O}$. The $\mathscr{L}$ is equivalent between Fig. 3.7(d) (where there are no $\mathcal{O}$-admissible forward neighbors of $\boldsymbol{l}^*$) and Fig. 3.7(e) but η^2 will be reduced and the iteration count is still incremented.

Also associated with the adaptive procedure is a convergence estimator, $\eta(\mathcal{A})$, defined as

$$\eta^2(\mathcal{A}) = \sum_{\boldsymbol{l}\in\mathcal{A}} \epsilon^2(\boldsymbol{l}). \tag{3.6}$$

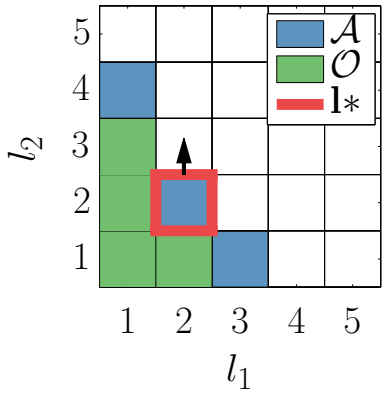
As discussed in [22], $\eta(\mathcal{A})$ is usually an acceptable surrogate for the projection error. Therefore, the adaptivity continues until $\eta < \texttt{Tol}$ or $\mathcal{A} = \varnothing$, where $\texttt{Tol} > 0$ is a user defined error tolerance.

An alternative is to require $\mathscr{L}$ admissibility of $\boldsymbol{l}^+$ instead of $\mathcal{O}$ admissibility. Such a requirement maintains the final admissibility of $\mathscr{L}$ however the adaptation is more aggressive by adding indices with higher-order mixed dependence to $\mathscr{L}$ sooner in the adaptation. In particular, such a strategy greatly inhibits the addition of purely monomial terms without mixed interaction. As will be discussed later in Section 3.4, this was used successfully on the HYCOM data however, we do not, in general, recommend such an aggressive approach. The slightly-modified algorithm is presented in Algorithm 3.2.

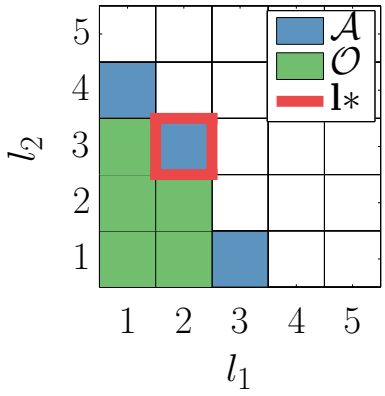


(a) Iteration 3

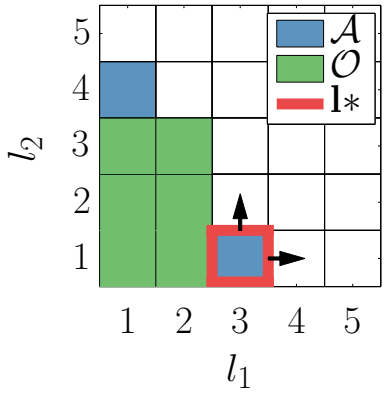
(b) Iteration 4



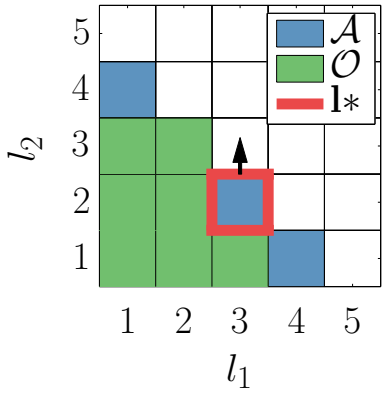
(c) Iteration 5



(d) Iteration 6



(e) Iteration 7



(f) Iteration 8

FIGURE 3.7: Schematic example of the aPSP algorithm in 2D. Notice that only $\mathcal{O}$-admissible forward neighbors are added and an index may be removed from $\mathcal{A}$ without admitting forward neighbors.

Algorithm 3.1 Adaptive Smolyak Pseudo-Spectral Projection (aPSP).

$\mathcal{A} \doteq \{1, \ldots, 1\}$
$\mathcal{O} \doteq \varnothing$
while $\eta^2 \doteq \sum_{\boldsymbol{l} \in \mathcal{A}} \epsilon(\boldsymbol{l})^2 \geqslant \mathtt{Tol}_\eta^2$ **do**
  Select $\boldsymbol{l}^* \in \mathcal{A}$ with highest $\epsilon(\boldsymbol{l})$
  $\mathcal{O} \leftarrow \mathcal{O} \cup \boldsymbol{l}^*$
  $\mathcal{A} \leftarrow \mathcal{A} \backslash \boldsymbol{l}^*$
  for $n = 1, \ldots, d$ **do**
    $\boldsymbol{l}^+ = \boldsymbol{l}^* + \hat{\mathtt{e}}_n$ ▷ Candidate Index
    if $\boldsymbol{l}^+ - \hat{\mathtt{e}}_k \in \mathcal{O} \ \forall \ k \in \left[1, \ldots, d, l_k^+ > 1\right]$ **then** ▷ $\mathcal{O}$-Admissibility
      $\mathcal{A} \leftarrow \mathcal{A} \cup \boldsymbol{l}^+$
      Project $\mathbf{P}_{\boldsymbol{l}^+}^{\mathtt{FT}}$ with F ▷ Full Tensor projection
      Calculate $\boldsymbol{\Delta}_{\boldsymbol{l}^+}^{\mathbf{P}}$
      $\epsilon(\boldsymbol{l}^+) = (1/\beta)\|\boldsymbol{\Delta}_{\boldsymbol{l}^+}^{\mathbf{P}}\|$
    end if
  end for
end while
Evaluate $\mathbf{P}_{\mathscr{L}} = \sum_{\boldsymbol{l} \in \mathscr{L}} \boldsymbol{\Delta}_{\boldsymbol{l}}^{\mathbf{P}}$

Algorithm 3.2 Adaptive Smolyak Pseudo-Spectral Projection [Aggressive Form].

$\mathcal{A} \doteq \{1, \ldots, 1\}$
$\mathcal{O} \doteq \varnothing$
while $\eta^2 \doteq \sum_{\boldsymbol{l} \in \mathcal{A}} \epsilon(\boldsymbol{l})^2 \geqslant \mathtt{Tol}_\eta^2$ **do**
  Select $\boldsymbol{l}^* \in \mathcal{A}$ with highest $\epsilon(\boldsymbol{l})$
  $\mathcal{O} \leftarrow \mathcal{O} \cup \boldsymbol{l}^*$
  $\mathcal{A} \leftarrow \mathcal{A} \backslash \boldsymbol{l}^*$
  for $n = 1, \ldots, d$ **do**
    if $\boldsymbol{l}^+ = \boldsymbol{l}^* + \hat{\mathtt{e}}_n \notin \mathscr{L}$ **then** ▷ Candidate Index Not in $\mathscr{L}$
      if $\boldsymbol{l}^+ - \hat{\mathtt{e}}_k \in \mathscr{L} \ \forall \ k \in \left[1, \ldots, d, l_k^+ > 1\right]$ **then** ▷ $\mathscr{L}$ - Admissibility
        $\mathcal{A} \leftarrow \mathcal{A} \cup \boldsymbol{l}^+$
        Project $\mathbf{P}_{\boldsymbol{l}^+}^{\mathtt{FT}}$ with F ▷ Full Tensor projection
        Calculate $\boldsymbol{\Delta}_{\boldsymbol{l}^+}^{\mathbf{P}}$
        $\epsilon(\boldsymbol{l}^+) = (1/\beta)\|\boldsymbol{\Delta}_{\boldsymbol{l}^+}^{\mathbf{P}}\|$
      end if
    end if
  end for
end while
Evaluate $\mathbf{P}_{\mathscr{L}} = \sum_{\boldsymbol{l} \in \mathscr{L}} \boldsymbol{\Delta}_{\boldsymbol{l}}^{\mathbf{P}}$

3.4 Results on HYCOM Database

The aggressive form of the adaptive algorithm described in Section 3.3 is first tested using the area-averaged SST as the QoI, and the algorithm's performance is analyzed by computing the relative error between its PCE and the LHS ensemble. The study is performed for $t = 60$ hr and $t = 120$ hr, i.e. at times when ξ_3 is dominant and when both ξ_3 and ξ_4 contribute substantially to the total variance (see Fig. 3.2). From Fig. 3.3 we expect SST to exhibit a steep dependence on ξ_4 at later times, whereas the dependence on ξ_3 is expected to remain essentially linear. Note that at $t = 120$ hr, the hurricane is located within the analysis region.

As previously discussed, the adaptive pseudo-spectral algorithm is anticipated to lead to performance gains due to two features: (a) judicious selection of stencils so as to maximize the ability to capture the variance of the selected QoI, and (b) maximizing the PC basis ($\mathcal{K}$), namely by including the *maximal* polynomial basis, $\mathcal{K} = \mathcal{K}^{*\mathscr{L}}$ (2.13). Figure 3.8 illustrates the impact of these features by comparing the errors in the area averaged SST for different basis refinement strategies:

S1. the (aggressive) adaptive pseudo-spectral algorithm;

S2. the pseudo-spectral PCE construction algorithm applied to an isotropically-refined sparse grid; and

S3. the isotropic Smolyak quadrature with direct projection.

For S2 and S3, the final point includes the entire database. Thus, the key difference between the second and third constructions concerns the inclusion of a larger basis in the former. In all cases, both the quadrature accuracy *and* the PCE stencil are refined.

Figure 3.8 shows the error of the adaptive pseudo-spectral algorithm decreasing rapidly, and essentially saturating after the fifth iteration. At this stage, the adaptive

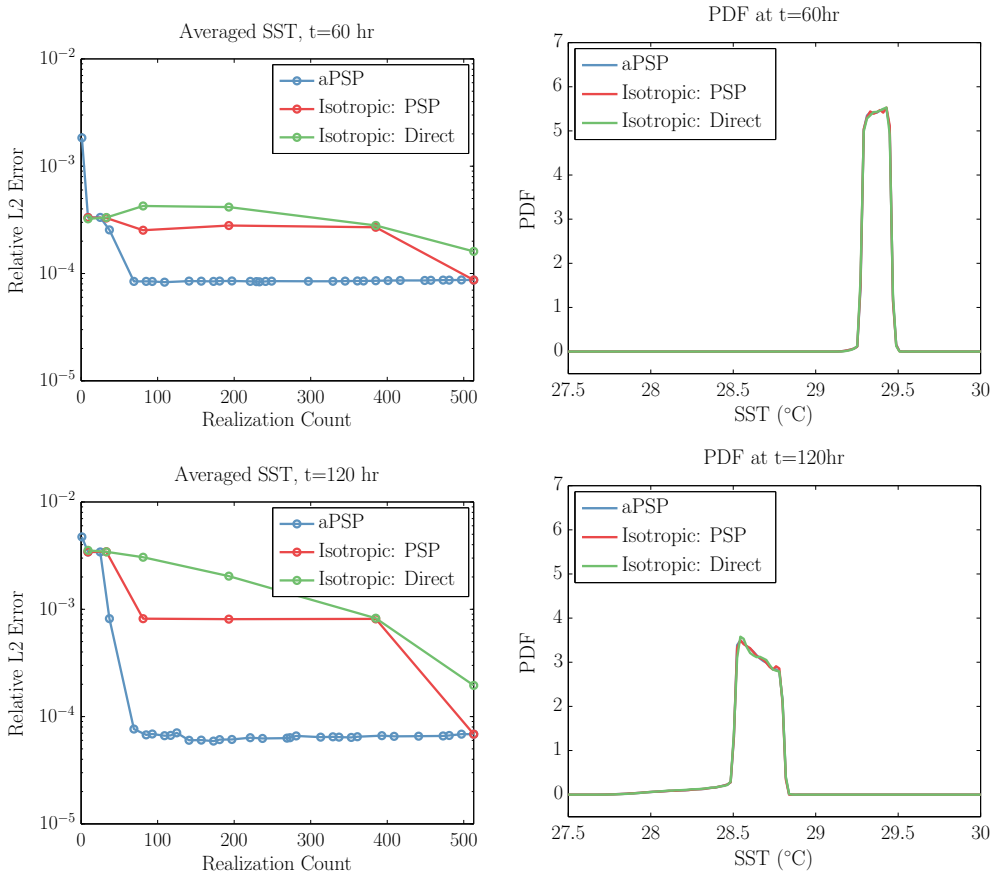


FIGURE 3.8: Left: LHS error versus realization count for the three refinement strategies: (S1) the adaptive pseudo-spectral algorithm, (S2) the pseudo-spectral construction applied in combination with isotropic refinement, except for the last point where the full database is used, and (S3) Smolyak sparse quadrature, except for the last iteration for which dimension truncation is used in conjunction with the full database. Right: PDFs of SST based on (S1) the PCE corresponding to the adaptive pseudo-spectral algorithm at iteration 5 (69 realizations and 59 polynomials), (S2) the pseudo-spectral PCE using the full database, and (S3) the PCE constructed using Smolyak quadrature using the full database. In all cases, the PDFs are generated using 10^6 samples drawn from the corresponding PCEs. Top row: $t = 60$ hr; bottom row: $t = 120$ hr.

pseudo-spectral PCE (S1) uses 69 realizations, and the corresponding basis has 59 polynomials. In contrast, Fig. 3.8 indicates that the decay of the relative error is substantially slower with isotropic refinement (S3), though noticeable improvement over Smolyak quadrature is observed when the basis is constructed pseudo-spectrally (S2).

The curves also indicate that the error levels experienced with isotropically refined grids remain substantially larger than with the adaptive grid, at least until the entire database is used. Combined, the results indicate that the adaptive pseudo-spectral algorithm can lead to substantial gains, which arise due to judicious refinement of the realization stencil and of the PCE basis. An illustration of the savings afforded by the adaptive pseudo-spectral construction is provided in the right panel of Fig. 3.8, which contrasts PDFs of the SST obtained based on the PCE generated with the adaptive scheme at iteration 5 (69 realizations), with results obtained similarly but using the full database. The close agreement between all three curves highlights the computational savings obtained using the adaptive pseudo-spectral algorithm as well as the validity of the corresponding representation.

It is interesting to note that with isotropic sampling, a large number of realizations are added as the refinement level increases, whereas in adaptive sampling the number of realizations increases at a much slower rate with the iteration count. It is also interesting to note the plateau in the relative L_2 (RMS) error is small, about 1%. This is consistent with the earlier analysis in [2], which indicated that a level 5 isotropic refinement leads to a suitable representation of the solution response.

In order to investigate the origin of the plateau in the RMS error in Fig. 3.8, we have examined the evolution of the realization stencil in the adaptive refinement scheme. This exercise revealed that the error plateau arises when the refinement reaches the limits of the database, i.e. when forward neighbors of the critical multi-index are not available in the database. An illustration of the evolution of the adaptive refinement is provided in Fig. 3.9, which depicts projections of the stencil on the ξ_3–ξ_4 plane. Recall that the first two iterations are always identical, and reflect an isotropic refinement step in all directions. Consistent with the results of the sensitivity analysis [2], and the projections shown in Fig. 3.3, the refinement first proceeds along ξ_3. Because a linear approximation in ξ_3 effectively captures the

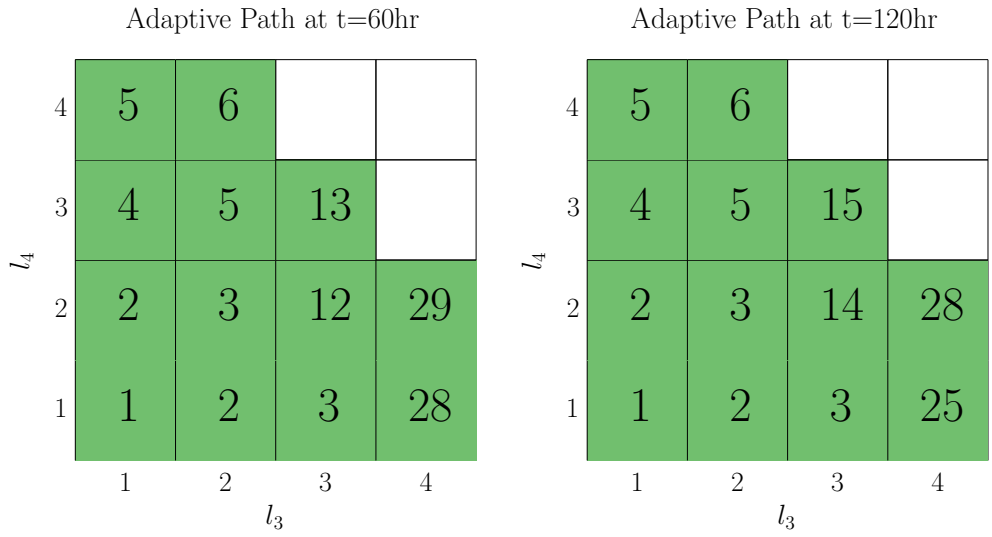


FIGURE 3.9: Schematic illustration of the evolution of the adaptive refinement algorithm. Plotted are projections of the stencils on the ξ_3–ξ_4 plane. The boxes indicate the refinement levels, whereas the numbers within each box refer to the iteration number at which the box was added to the stencil. Recall that iterations 1 and 2 are always isotropic. Note that refinement occurs first along ξ_3, but then progresses along ξ_4, where the limits of the database are reached. Left: $t = 60$ hr; right: $t = 120$ hr.

corresponding dependence, refinement then proceeds along ξ_4. The steep variation of SST at the higher values of ξ_4 (see Fig. 3.3) requires high-order polynomial representation, refinement proceeds along ξ_4 until the limits of the database are reached (iteration 5). This then forces the algorithm to sequentially select the remaining stencils afforded by the database, including the mixed terms that appear at the higher iteration counts.

As noted, the refinement became limited by the realization database at iteration 5 for both $t = 60$ hr and $t = 120$ hr. Without such limitation, the refinement would have more monomial terms in ξ_4. Once the database was exhausted, the relative L_2 error plateaued in Fig 3.8 and all additional terms had little impact.

Another interesting observation in Figs. 3.8 and 3.9 is that the adaptive refinement proceeds in a similar fashion for the area-averaged SST at $t = 60$ hr and

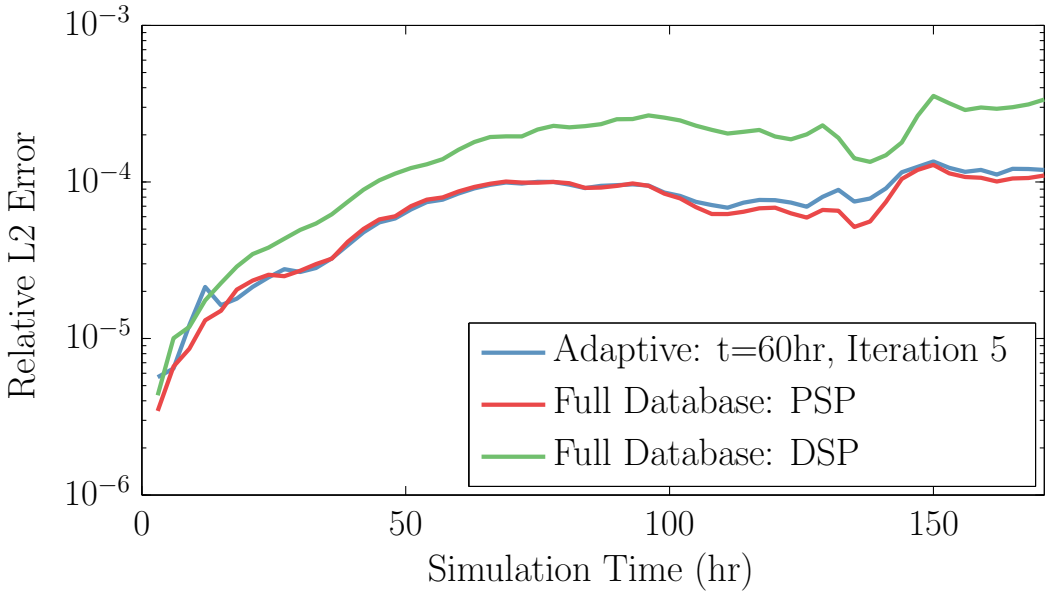


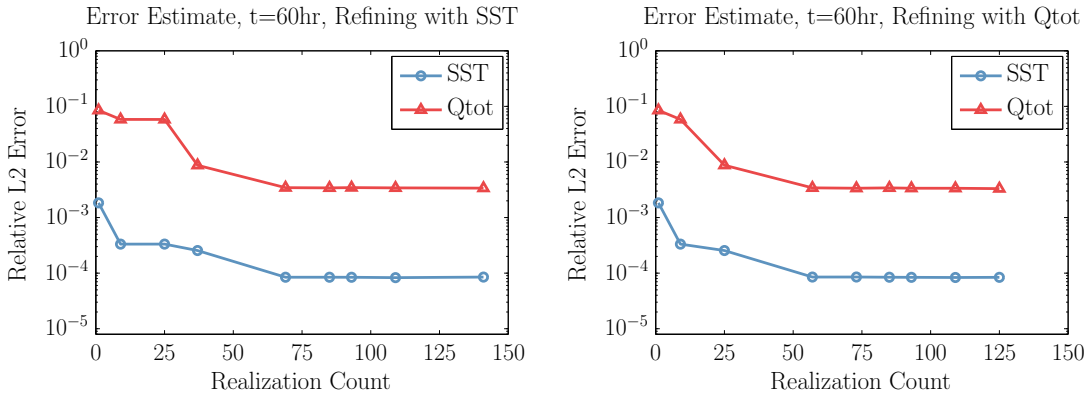
FIGURE 3.10: Relative L_2 error between the PCE of the averaged SST and the LHS sample. Plotted are curves generated with (i) the pseudo-spectral algorithm adapted to the solution at $t = 60$ hr, (ii) the pseudo-spectral algorithm using the full database, and (iii) DSP on smolyak quadrature using the full database. For the adapted solution, the refinement is stopped after iteration 5, leading to 69 realizations and a PCE with 59 polynomials. The full 513 database curves have 402 polynomials for the pseudo-spectral construction and 168 polynomials for the DSP.

$t = 120$ hr. This can be explained by noting that the dependence on ξ_3, the dominant variable at early times, is to a large extent captured by a linear term throughout the simulation. As the hurricane enters into the analysis region, the drag coefficient (ξ_4) becomes the dominant contributor to the variance. As depicted earlier, the dependence on ξ_4 is highly non-linear and evolves with time. However, it is rapidly captured with the next few refinement steps. Though time-dependent, subsequent iterations that lead to the inclusion of mixed terms take place well after the error has decreased dramatically. This should be viewed as a particular feature of the present application and may not be necessarily the case for others.

To further verify the fidelity of the adaptive PCE with time, we plot in Fig. 3.10 the evolution of the relative LHS error based on the adaptive pseudo-spectral solution

truncated at iteration 5 (69 realizations, 59 polynomials), the pseudo-spectral expansion using the full database (513 realizations, 402 polynomials), and the Smolyak quadrature using the full database (168 polynomials). The adaptive solution is based on the box-average SST at $t = 60$ hr. The results show close agreement between the adapted solution and the pseudo-spectral approximation based on the full database, throughout the computations. Also note that at later simulation times, the adaptive solution exhibits error levels that are substantially smaller than those obtained using DSP on Smolyak quadrature, though the latter involve a much larger number of realizations.

It is also instructive to examine whether a PCE resulting from adaptive refinement based on a specific QoI would also prove suitable for the purpose of representing other QoIs, and whether adaptive refinement driven by multiple QoIs simultaneously would potentially require a large number of realizations and consequently limit performance gains. Though these issues, in general, vary from one application to another, it is still worthwhile to address them in the present setting of the oceanic response to hurricane-strength winds.



(a) Driven by SST (b) Driven by Q_{tot}

FIGURE 3.11: Relative L_2 error between LHS sample and the PCEs for SST and Q_{tot}. The adaptive refinement is based on the critical multi-index for (a) the box-averaged SST and (b) Q_{tot}.

Figure 3.11 shows the relative L_2 error based on the LHS ensemble for heat flux, Q_{tot} (see Section 3.2, Page 32), and the box-averaged SST. Plotted are results obtained using PCEs that are adapted using either (a) the box-averaged SST or (b) Q_{tot} as the sole refinement QoI at $t = 60$ hr. The curves show that the decay of the error is not identical in both cases, indicating that the refinement does not follow the same pattern, and that optimal sampling and PCE may depend on the selected QoI. This is not surprising since the SST is averaged over a fixed region, whereas Q_{tot} involves an average in a circular area around the center of the hurricane. Thus, Q_{tot} is always affected by the high winds whereas the box-averaged SST experiences both weak winds and high winds. Note, however, that in both cases a plateau in the error curves is achieved at about the same number of realizations, and much earlier than with isotropic sampling. Thus, in the present examples, the performance gains do not appear to be substantially affected by the QoI selected.

An alternative to the single QoI adaptation is to adapt concurrently based on multiple QoIs, wherein the refinement consider the union of the forward neighbors corresponding to the critical multi-indices of all QoIs. This strategy potentially involves a larger number of realizations than with single-QoI adaptation, but naturally overcomes the drawback identified above. In Fig. 3.12, we plot curves of the relative L_2 LHS error against the realization count for all four QoIs considered, namely SST, SSH, MLD, and Q_{tot}. The results indicate that with the present refinement strategy, all error estimates decay rapidly. Note that, as expected, in the present case the refinements steps generally include a larger number of realizations than in single-QoI adaptation. The error curves tend to become essentially flat when the realization count exceeds 70. Thus, for the present setting, similar performance gains are achieved when considering SST alone or in combination with SSH, MLD, and Q_{tot}. Of course, in other settings, some QoIs may converge sooner than others, and it may be advantageous to consider more elaborate algorithms, for instance stopping

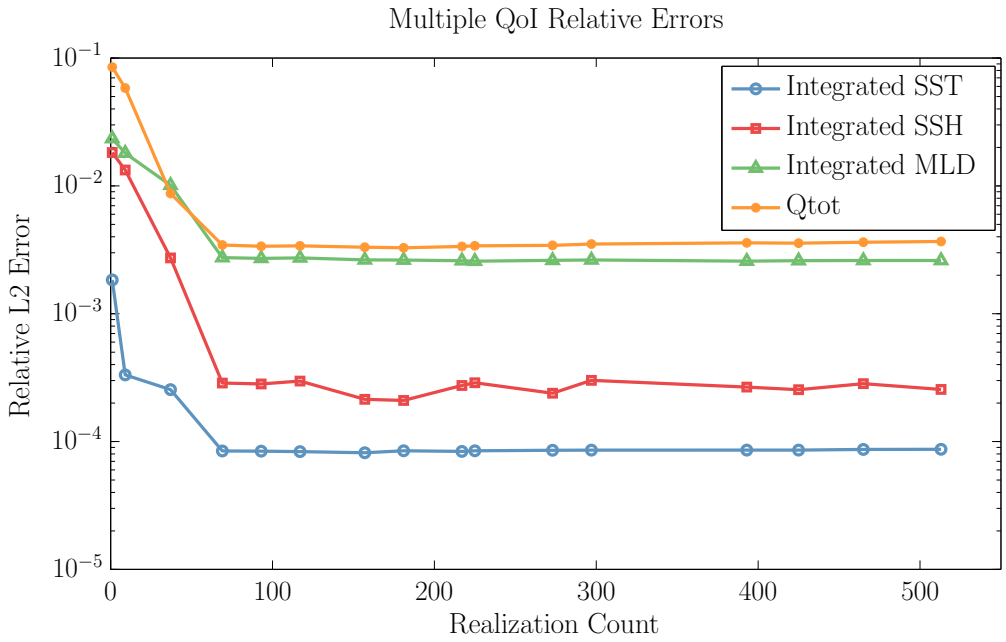


FIGURE 3.12: Relative error versus realization count. Plotted are curves SST, SSH, MLD, and Q_{tot}. The stencil is enriched by the union of all admissible forward neighbors determined for each QoI individually.

refinement based on a specific QoI when the corresponding PCE is deemed sufficiently resolved. Alternatively, one could refine on the index that has the maximum (*normalized*) $\epsilon(\boldsymbol{l})$ amongst all of the model QoIs. We also note that a single QoI may be used at different simulation times and treated as multiple QoIs.

We conclude with brief remarks concerning the scope of the present implementations and potential extensions. In the a priori analyses above, the adaptive algorithm was allowed to proceed until the entire database of pre-computed realizations was exhausted. In dynamic implementation, this is of course not possible, and suitable stopping criteria must be provided. In this regard, we point out that the adaptive sampling scheme supports a variety of approaches, for instance, based on monitoring the saturation of the variance in selected QoIs, providing global tolerance levels for the error indicators, or detecting occurrence of plateaus in the point-wise error of the sparse grid. The implementation of such criteria would naturally exploit the

variance-based error indicators associated with active multi-indices, and may also involve the corresponding number of forward neighbors. As discussed in [46], the latter may also be used for the purpose of introducing suitable cost functions, and accordingly optimize the enrichment of the realization stencil.

3.5 Conclusion

In this chapter, we exploited an existing database of HYCOM simulations of the oceanic response in the GOM to Hurricane Ivan to perform an *a priori* analysis and testing of adaptive methods. The database was adapted from prior work [2], where a sparse isotropic sampling scheme was used to propagate uncertainties in the wind drag coefficient and subgrid mixing parameterizations. Guided by the sensitivity analysis in [2], the database was enriched along the dominant directions. This resulted in an extended database comprising 513 independent realizations, with stochastic coordinates located at the quadrature points of an anisotropic sparse grid. For the purpose of characterizing representation errors, we also generated an independent Latin Hypercube Sample consisting of 256 realizations.

The adaptive pseudo-spectral algorithm recently developed in [22] was implemented and deployed on the enriched database. It accommodates anisotropic refinement on arbitrary admissible sparse grids, and enables the evaluation of a "maximal" basis set in a manner that avoids internal aliasing. Inherent in the adaptive pseudo-spectral construction is an error refinement indicator that can be readily used to drive anisotropic refinement of the sparse grid. This involves inclusion of admissible forward neighbors of the critical multi-index, or critical multi-index set.

The performance of the adaptive pseudo-spectral algorithm was also tested *a priori* using the HYCOM database. The tests included adaptive refinement based on a single QoI and on multiple QoIs. In particular, the tests indicated that suitable PCEs of the uncertain model response can be constructed at a small fraction of the

cost of isotropic sampling, and that the performance gains arise from the algorithm's ability to refine efficiently, as well as from the inclusion of all polynomials whose coefficients can be determined without internal aliasing.

The present *a priori* tests thus provide support to the deployment of the adaptive pseudo-spectral algorithm in a dynamic fashion. The adaptive pseudo-spectral algorithm has already been adopted in a study of the oceanic response to Typhoon Fanapi where it is used to build the surrogate needed for uncertainty propagation and parameter inference; results from this study will be reported elsewhere [120].

4

Directional Adaptation for Uncertainty Quantification

4.1 Introduction

Recent computational experiences [90, 89] have clearly put into evidence that the availability of accurate model surrogates can provide substantial advantages in inference or parameter problems. This stems from the fact that it is substantially more effective to evaluate the surrogate than the model itself, especially when the complexity, scale, or dimensionality of the model is large. These advantages have in fact been demonstrated in different settings, including stochastic approaches based on random sampling of the surrogate [120], or in variational approaches where the surrogate is evaluated iteratively [119].

Another situation where the availability of suitable surrogates is advantageous concerns problems of optimal experimental design. Frequently, these aim at maximizing the information gain from a set of experiments [86], with estimates involving integrals over both the space of experimental design variables and the space of random parameters [17, 61, 110]. In these and other settings, it is highly desirable to

have available *global* surrogates that provide suitable representations of quantities of interest (QoIs) as function of both deterministic design variables and of random model inputs. This topic is central to present chapter.

We explore two approaches for constructing an adapted global PCE representation of a model's QoI with the aim to independently control the adaptation in different sets of directions. The first is a nested approach, similar to that developed by *Eldred et al.* [38, 40] who relied on nested iterations between two spaces ("aleatory-epistemic" uncertainty) with the goal of interval estimation. In our work, we seek for a global representation over the product space of design and stochastic parameters. Specifically, we perform an adaptation in a first set of directions ("outer directions"), and at each of the corresponding grid points perform NISP in the remaining directions ("inner directions"). To accommodate potentially high-dimensional parametrization, we rely on an adaptive pseudo-spectral construction (aPSP) (see Section 3.3) with well-established sparse tensorization concepts [116, 47]. The aPSP method [25, 22] eliminates internal aliasing by applying a Smolyak-type sparse tensor-product approximation to the projection operator instead of the classical quadrature approximation. The aPSP also accommodates non-isotropic sparse grid adaptation wherein the tensorization set is iteratively refined using a greedy algorithm [46, 22]. The aPSP has been shown to reduce the number of model evaluation needed for the construction while *increasing* accuracy [134, & Chapter 3] and has been applied in practical settings [61, 120, 119].

An alternative approach is also explored, namely by considering the product space of two subsets of parameters (*e.g.* design variables and stochastic parameters). In this second approach, we perform a single adaptive projection in a product space allowing for increased sparsity in the quadrature grids. However, unlike the nested approach, there is no longer direct control of the fidelity of the representation in the two subsets of parameters individually. To overcome this hurdle, we develop a Sobol

decomposition [117, 58] of the projection surpluses to derive directional indicators to tune adaptation and termination criteria.

This chapter is as organized follows. In Section 4.2, we present the two approaches for obtaining a global representation of QoIs in terms of two independent sets of parameters. In Section 4.3, we employ a simple test problem to assess the performance of the two techniques and to analyze the quality of resulting representations. In Section 4.4 we perform a large-scale demonstration for a stiff chemical system involving three design variables and 22 stochastic parameters. Major conclusions are summarized in Section 4.5.

4.2 Methodology

The aPSP algorithm introduced in Chapter 3, Section 3.3 does not inherently distinguish between the individual directions, ξ_i. In many situations, however, we would be interested in a finer control of the projection error along certain directions. This is the case for instance when a set of input variables is made up of a subset of variables having a different origin or nature (uncertainty sources). To remain general, we then consider a partition of $\boldsymbol{\xi}$ into two independent sets of variables $\boldsymbol{\xi}_{\mathtt{u}}$, and $\boldsymbol{\xi}_{\mathtt{p}}$ with respective ranges $\Xi_{\mathtt{u}}$ and $\Xi_{\mathtt{p}}$, densities $\rho_{\mathtt{u}}$ and $\rho_{\mathtt{p}}$, such that

$$\Xi = \Xi_{\mathtt{u}} \times \Xi_{\mathtt{p}}, \quad \rho(\boldsymbol{\xi}) = \rho_{\mathtt{u}}(\boldsymbol{\xi}_{\mathtt{u}}) \times \rho_{\mathtt{p}}(\boldsymbol{\xi}_{\mathtt{p}}).$$

We shall denote $d_{\mathtt{u}}$ and $d_{\mathtt{p}}$ the dimensions of $\Xi_{\mathtt{u}}$ and $\Xi_{\mathtt{p}}$, and we seek to construct in an adaptive way a PCE $F(\boldsymbol{\xi} = (\boldsymbol{\xi}_{\mathtt{p}}, \boldsymbol{\xi}_{\mathtt{u}})) \approx \sum_{\boldsymbol{k}\in\mathcal{K}} c_{\boldsymbol{k}}\Psi_{\boldsymbol{k}}(\boldsymbol{\xi})$ of some quantity of interest F, while somehow controlling the errors along $\boldsymbol{\xi}_{\mathtt{u}}$ and $\boldsymbol{\xi}_{\mathtt{p}}$ independently.

4.2.1 Nested Projection

In the Nested Projection (NP), the aPSP method of Section 3.3.2 is applied in a nested fashion, allowing for independent control of the convergence in $\boldsymbol{\xi}_{\mathtt{u}}$ and $\boldsymbol{\xi}_{\mathtt{p}}$.

Nested PC Expansion

For $F(\boldsymbol{\xi}_{\mathrm{p}}, \boldsymbol{\xi}_{\mathrm{u}}) \in L_2(\Xi_{\mathrm{p}}, \rho_{\mathrm{p}}) \otimes L_2(\Xi_{\mathrm{u}}, \rho_{\mathrm{u}})$, the nested expansion consists in considering first a *truncated* spectral expansion of $F(\boldsymbol{\xi}_{\mathrm{p}}, \cdot)$ with respect to $\boldsymbol{\xi}_{\mathrm{u}}$, referred to hereafter as the inner expansion:

$$F(\boldsymbol{\xi}_{\mathrm{p}}, \boldsymbol{\xi}_{\mathrm{u}}) \approx \sum_{\boldsymbol{k} \in \mathcal{K}_{\mathrm{u}}} f^{\boldsymbol{k}}(\boldsymbol{\xi}_{\mathrm{p}}) \Psi_{\boldsymbol{k}}(\boldsymbol{\xi}_{\mathrm{u}}). \tag{4.1}$$

We then perform a PC expansion in $\boldsymbol{\xi}_{\mathrm{p}}$ (outer expansion) of the coefficients $f^{\boldsymbol{k}}(\boldsymbol{\xi}_{\mathrm{p}})$ in Equation (4.1), namely

$$f^{\boldsymbol{k}}(\boldsymbol{\xi}_{\mathrm{p}}) \approx \sum_{\boldsymbol{j} \in \mathcal{K}_{\mathrm{p}}} c_{\boldsymbol{j}}^{\boldsymbol{k}} \Psi_{\boldsymbol{j}}(\boldsymbol{\xi}_{\mathrm{p}}). \tag{4.2}$$

Substituting Equation (4.2) into Equation (4.1), we obtain the global PCE of $F(\boldsymbol{\xi}_{\mathrm{p}}, \boldsymbol{\xi}_{\mathrm{u}})$:

$$F(\boldsymbol{\xi}_{\mathrm{p}}, \boldsymbol{\xi}_{\mathrm{u}}) \approx \sum_{\boldsymbol{k} \in \mathcal{K}_{\mathrm{u}}} \sum_{\boldsymbol{j} \in \mathcal{K}_{\mathrm{p}}} c_{\boldsymbol{j}}^{\boldsymbol{k}} \Psi_{\boldsymbol{j}}(\boldsymbol{\xi}_{\mathrm{p}}) \Psi_{\boldsymbol{k}}(\boldsymbol{\xi}_{\mathrm{u}}). \tag{4.3}$$

Note that multi-indices in the two sets $\mathcal{K}_{\mathrm{u}}$ and $\mathcal{K}_{\mathrm{p}}$ have different sizes in general. We pursue this idea of nested expansion in the context of the PSP method, as outlined below.

Inner Projection

Let $\mathcal{G}(\mathscr{L}_{\mathrm{p}}) = \{{\boldsymbol{\xi}_{\mathrm{p}}}^{(i)}, i = 1, \cdots, N(\mathscr{L}_{\mathrm{p}})\}$ be an admissible sparse grid in Ξ_{p}, see (2.11). We call $\mathcal{G}(\mathscr{L}_{\mathrm{p}})$ the outer sparse grid. At each node of the outer grid the aPSP is employed to approximate $G^{(i)}(\boldsymbol{\xi}_{\mathrm{u}}) \doteq F({\boldsymbol{\xi}_{\mathrm{p}}}^{(i)}, \boldsymbol{\xi}_{\mathrm{u}})$, thus providing:

$$G^{(i)}(\boldsymbol{\xi}_{\mathrm{u}}) \approx \sum_{\boldsymbol{k} \in \mathcal{K}_{\mathrm{u}}^{(i)}} g_{\boldsymbol{k}}^{(i)} \Psi_{\boldsymbol{k}}(\boldsymbol{\xi}_{\mathrm{u}}). \tag{4.4}$$

Note that this construction is purely local at ${\boldsymbol{\xi}_{\mathrm{p}}}^{(i)}$ and involves evaluations of F at a fixed value $\boldsymbol{\xi}_{\mathrm{p}} = {\boldsymbol{\xi}_{\mathrm{p}}}^{(i)} \in \Xi_{\mathrm{p}}$. The adaptation of the inner sparse grid uses refinement

indicators ϵ_{in} based on $G^{(i)}(\boldsymbol{\xi}_{\mathtt{u}})$ in Equation (3.4). The inner adaptation is carried out until a prescribed termination criterion $\mathtt{Tol}_{\mathtt{u}}$ is reached. The same criterion $\mathtt{Tol}_{\mathtt{u}}$ is used for all nodes of the outer sparse grid. Therefore, different outer nodes ${\boldsymbol{\xi}_{\mathtt{p}}}^{(i)}$ may use different inner sparse grids to support the aPSP of their local functions $G^{(i)}$. We denote $\mathcal{G}_{\mathtt{u}}^{(i)} = \mathcal{G}(\mathscr{L}_{\mathtt{u}}^{(i)})$ the resulting inner sparse grid at node ${\boldsymbol{\xi}_{\mathtt{p}}}^{(i)}$, and $\mathcal{K}_{\mathtt{u}}^{(i)} \doteq \mathcal{K}(\mathscr{L}_{\mathtt{u}}^{(i)})$ the multi-index set of polynomial tensorizations in the approximation of $G^{(i)}$.

Outer Projection

For the outer sparse grid, we again rely on the aPSP scheme, but with refinement indicators ϵ_{out} based on $H(\boldsymbol{\xi}_{\mathtt{p}}) \equiv \|F(\boldsymbol{\xi}_{\mathtt{p}}, \cdot)\|_{L_2(\Xi_{\mathtt{u}}, \rho_{\mathtt{u}})}$. The function $H(\boldsymbol{\xi}_{\mathtt{p}})$ is unknown but needs to only be evaluated at the nodes ${\boldsymbol{\xi}_{\mathtt{p}}}^{(i)}$ of $\mathcal{G}(\mathscr{L}_{\mathtt{p}})$, where we can take as a surrogate the norm of the approximation of $G^{(i)}(\boldsymbol{\xi}_{\mathtt{u}})$. This leads to

$$H\left({\boldsymbol{\xi}_{\mathtt{p}}}^{(i)}\right) = \left\|G^{(i)}\right\|_{L_2(\Xi_{\mathtt{u}}, \rho_{\mathtt{u}})} \approx \sqrt{\sum_{\boldsymbol{k} \in \mathcal{K}_{\mathtt{u}}^{(i)}} \left(g_{\boldsymbol{k}}^{(i)}\right)^2}.$$

The adaptation of the outer sparse grid is carried out until the termination criterion $\mathtt{Tol}_{\mathtt{p}}$ is reached. Since the inner projections are performed locally, the points ${\boldsymbol{\xi}_{\mathtt{p}}}^{(i)}$ support approximations of $G^{(i)}$ defined over generally different polynomial subspaces of $L_2(\Xi_{\mathtt{u}}, \rho_u)$, as prescribed by $\mathcal{K}_{\mathtt{u}}^{(i)}$. A "global" polynomial subspace of $L_2(\Xi_{\mathtt{u}}, \rho_{\mathtt{u}})$ can be defined as the union of all the $\mathcal{K}_{\mathtt{u}}^{(i)}$,

$$\mathcal{K}_{\mathtt{u}}(\mathscr{L}_{\mathtt{p}}) \doteq \bigcup_{i=1}^{N(\mathscr{L}_{\mathtt{p}})} \mathcal{K}_{\mathtt{u}}^{(i)}, \tag{4.5}$$

and extending the inner approximations as follows

$$G^{(i)}(\boldsymbol{\xi}_{\mathtt{u}}) \approx \sum_{\boldsymbol{k} \in \mathcal{K}_{\mathtt{u}}(\mathscr{L}_{\mathtt{p}})} \tilde{g}_{\boldsymbol{k}}^{(i)} \Psi_{\boldsymbol{k}}(\boldsymbol{\xi}_{\mathtt{u}}), \quad \tilde{g}_{\boldsymbol{k}}^{(i)} = \begin{cases} g_{\boldsymbol{k}}^{(i)} & \boldsymbol{k} \in \mathcal{K}_{\mathtt{u}}^{(i)}, \\ 0 & \text{otherwise.} \end{cases} \tag{4.6}$$

The definition of $\mathcal{K}_{\mathtt{u}}(\mathscr{L}_{\mathtt{p}})$ enables us to define the global polynomial expansion of F, simply by performing the PSP of the set of inner PC coefficients $\{\tilde{g}_{\boldsymbol{k}}^{(i)}, \boldsymbol{k} \in \mathcal{K}_{\mathtt{u}}(\mathscr{L}_{\mathtt{p}})\}$, defined at each node ${\boldsymbol{\xi}_{\mathtt{p}}}^{(i)}$ of the outer sparse grid. This operation results in a vector of coefficients, $c_{\boldsymbol{k}}^{\boldsymbol{j}}$, with $\boldsymbol{k} \in \mathcal{K}_{\mathtt{u}}(\mathscr{L}_{\mathtt{p}})$ and $\boldsymbol{j} \in \mathcal{K}_{\mathtt{p}}(\mathscr{L}_{\mathtt{p}})$, where $\mathcal{K}_{\mathtt{p}}$ denotes the multi-index set of the projection subspace associated with the H-adapted outer sparse grid. This leads to the global representation:

$$F(\boldsymbol{\xi}_{\mathtt{p}}, \boldsymbol{\xi}_{\mathtt{u}}) \simeq \sum_{\boldsymbol{j} \in \mathcal{K}(\mathscr{L}_{\mathtt{p}})} \sum_{\boldsymbol{k} \in \mathcal{K}_{\mathtt{u}}(\mathscr{L}_{\mathtt{p}})} c_{\boldsymbol{k}}^{\boldsymbol{j}} \Psi_{\boldsymbol{j}}(\boldsymbol{\xi}_{\mathtt{p}}) \Psi_{\boldsymbol{k}}(\boldsymbol{\xi}_{\mathtt{u}}). \tag{4.7}$$

Note that by construction, we end up with an expansion based on a full summation over the multi-indices in $\mathcal{K}_{\mathtt{p}}(\mathscr{L}_{\mathtt{p}})$ and in $\mathcal{K}_{\mathtt{u}}(\mathscr{L}_{\mathtt{p}})$, i.e. with a structure that is reminiscent of fully tensorized constructions. As previously discussed, this is achieved by setting some (non computed) inner expansion coefficients to zero, see Equation (4.6). Though this could potentially result in errors, it is anticipated that these coefficients would primarily affect higher-order modes having small amplitudes only. Nonetheless, we will examine this issue in Section 4.3 based on results obtained for an idealized test problem.

Though the fully tensored nature of the $\boldsymbol{\xi}_{\mathtt{p}}$ and $\boldsymbol{\xi}_{\mathtt{u}}$ spaces is expected to be computationally demanding, an advantage of the nested projection is that it provides a clear handle on convergence in the two spaces separately. Note, however, this does not readily translate into estimates of convergence for the global representation in Equation (4.7). Finally, we also note that the outer and inner projections may be inverted in the nested projection depending on the nature of the problem, and that the approach could conceptually be extended to any number of parameter spaces.

4.2.2 Product Space Projection

In this alternative approach, we focus on expanding $F(\boldsymbol{\xi})$, with $\boldsymbol{\xi} = (\boldsymbol{\xi}_{\mathtt{p}}, \boldsymbol{\xi}_{\mathtt{u}})$. However, applying directly the adapted PSP method to $F(\boldsymbol{\xi})$ would treat all parameters

equally. This equal treatment may raise potential issues. For example, F may exhibit larger variations with $\boldsymbol{\xi}_{\mathtt{p}}$ than with $\boldsymbol{\xi}_{\mathtt{u}}$, and consequently the adaptation may ignore essential variations with $\boldsymbol{\xi}_{\mathtt{u}}$. Irrespective of whether such issue arises, it is desirable to incorporate means to separately assess resolution along distinct dimensions, as well as criteria that enable us to tune adaptation along these dimensions.

Theoretical Considerations

Recall that the aPSP involves two distinct sets of tensorizations, $\mathcal{O}$ and $\mathcal{A}$, which defines the projection of F through

$$\mathbf{P}_{\mathcal{O}\cup\mathcal{A}}F = \sum_{\boldsymbol{l}\in\mathcal{O}} \boldsymbol{\Delta}_{\boldsymbol{l}}^{\mathrm{P}} F + \sum_{\boldsymbol{l}\in\mathcal{A}} \boldsymbol{\Delta}_{\boldsymbol{l}}^{\mathrm{P}} F,$$

where $\boldsymbol{\Delta}_{\boldsymbol{l}}^{\mathrm{P}} F$ is called the projection surplus associated with the tensorization $\boldsymbol{l}$. Therefore, the second sum in the equation above represents the global projection surplus associated with the active set $\mathcal{A}$ of tensorizations available for adaptation. This global projection surplus, denoted $\delta\mathbf{P}_{\mathcal{A}}F$, belongs to the polynomial subspace defined by the multi-index set $\mathcal{K}(\mathcal{A})$, i.e.

$$\delta\mathbf{P}_{\mathcal{A}}F \equiv \sum_{\boldsymbol{l}\in\mathcal{A}} \boldsymbol{\Delta}_{\boldsymbol{l}}^{\mathrm{P}} F = \sum_{\boldsymbol{k}\in\mathcal{K}(\mathcal{A})} \delta f_{\boldsymbol{k}} \Psi_{\boldsymbol{k}}(\boldsymbol{\xi}), \quad \mathcal{K}(\mathcal{A}) = \bigcup_{\boldsymbol{l}\in\mathcal{A}} \mathcal{K}^*(\boldsymbol{l}).$$

We are now in position to measure the relative contributions of $\boldsymbol{\xi}_{\mathtt{p}}$ and $\boldsymbol{\xi}_{\mathtt{u}}$, by means of the orthogonal Sobol decomposition of $\delta\mathbf{P}_{\mathcal{A}}F$ (see Appendix B), that is formally

$$\delta\mathbf{P}_{\mathcal{A}}F(\boldsymbol{\xi}) = (\delta\mathbf{P}_{\mathcal{A}}F)_{\varnothing} + (\delta\mathbf{P}_{\mathcal{A}}F)_{\mathtt{p}}(\boldsymbol{\xi}_{\mathtt{p}}) + (\delta\mathbf{P}_{\mathcal{A}}F)_{\mathtt{u}}(\boldsymbol{\xi}_{\mathtt{u}}) + (\delta\mathbf{P}_{\mathcal{A}}F)_{\mathtt{p}\cup\mathtt{u}}(\boldsymbol{\xi}_{\mathtt{p}}, \boldsymbol{\xi}_{\mathtt{u}}).$$

The squared norm of the global projection surplus is then equal to the squared norms of the functions in the decomposition of $\delta\mathbf{P}_{\mathcal{A}}F$; we have

$$\|\delta\mathbf{P}_{\mathcal{A}}F\|^2_{L_2(\Xi,\rho)} = V_{\varnothing}^{\mathcal{A}} + V_{\mathtt{p}}^{\mathcal{A}} + V_{\mathtt{u}}^{\mathcal{A}} + V_{\mathtt{p}\cup\mathtt{u}}^{\mathcal{A}}, \tag{4.8}$$

where $V_{\varnothing}^{\mathcal{A}} = (\delta\mathbf{P}_{\mathcal{A}}F)^2_{\varnothing}$, while other terms can be easily computed form the PC expansion of $\delta\mathbf{P}_{\mathcal{A}}F$, as explained in Appendix B.2. Further, $V_{\varnothing}^{\mathcal{A}}$ is the squared

surplus of the projection on the mean mode and it is usually small compared to the other contributions to the norm of the global surplus. The partial variance $V_{\mathtt{p}}^{\mathcal{A}}$ (resp. $V_{\mathtt{u}}^{\mathcal{A}}$) then measures the available projection surplus along directions in $\mathtt{p}$ (resp. in $\mathtt{u}$) only, whereas $V_{\mathtt{p}\cup\mathtt{u}}^{\mathcal{A}}$ is the measure along the mixed directions. The partial estimates can be used to tune the adaptation strategy and to gauge accuracy control along the $\mathtt{p}$ and $\mathtt{u}$ directions. For instance, if $V_{\mathtt{u}}^{\mathcal{A}}$ becomes less than a certain tolerance value, we can decide to prevent the adaptive process to further enrich the sparse grid along the hyperplane $\Xi_{\mathtt{u}}$ of Ξ.

Construction of Directional Indicators

Though feasible, the decomposition of the different contributions to the (squared) norm of the available projection surplus in (4.8) induces some computational overheads, as it requires first the PC expansion of $\delta\mathbf{P}_{\mathcal{A}}F$ which involves a summation of all the projection surpluses $\boldsymbol{\Delta}_{\boldsymbol{l}}^{\mathrm{P}}F$ for $\boldsymbol{l}\in\mathcal{A}$. We would prefer to derive indicators of the available surpluses along the $\mathtt{p}$, $\mathtt{u}$ and mixed directions, through a summation of local indicators associated with each $\boldsymbol{l}\in\mathcal{A}$. A first simplification is obtained by performing *locally* the Sobol decomposition of each of the projection surpluses,

$$\boldsymbol{\Delta}_{\boldsymbol{l}}^{\mathrm{P}}F=\sum_{\boldsymbol{k}\in\mathcal{K}^*(\boldsymbol{l})}\delta f_{\boldsymbol{k}}^{\boldsymbol{k}}\Psi_{\boldsymbol{k}}(\boldsymbol{\xi})\Rightarrow\|\Delta_{\boldsymbol{l}}^{\mathrm{P}}F\|_{L_2(\Xi,\rho)}^2=V_{\varnothing}^{\boldsymbol{l}}+V_{\mathtt{p}}^{\boldsymbol{l}}+V_{\mathtt{u}}^{\boldsymbol{l}}+V_{\mathtt{p}\cup\mathtt{u}}^{\boldsymbol{l}},$$

and derive upper-bounds for the decomposition of the available surplus using

$$V_{\varnothing}^{\mathcal{A}}\leqslant\sum_{\boldsymbol{l}\in\mathcal{A}}V_{\varnothing}^{\boldsymbol{l}},\quad V_{\mathtt{p}}^{\mathcal{A}}\leqslant\sum_{\boldsymbol{l}\in\mathcal{A}}V_{\mathtt{p}}^{\boldsymbol{l}},\quad V_{\mathtt{u}}^{\mathcal{A}}\leqslant\sum_{\boldsymbol{l}\in\mathcal{A}}V_{\mathtt{u}}^{\boldsymbol{l}},\quad V_{\mathtt{p}\cup\mathtt{u}}^{\mathcal{A}}\leqslant\sum_{\boldsymbol{l}\in\mathcal{A}}V_{\mathtt{p}\cup\mathtt{u}}^{\boldsymbol{l}}.$$

We shall consider in the following the case where the polynomial exactness of the 1-D quadrature formulas at level $l=1$ is $p_1\leqslant 1$. This is a common situation in sparse grid methods that for efficiency generally use a low number of 1-D points at the first level, typically $N(1)=1$ (the first tensorization $\boldsymbol{l}=(1,\cdots,1)$ involves

$N(1)^d$ points, so a low value is mandatory for large d). In this case, the 1-D projection operators $\mathrm{P}_1 F \in \pi_0$ (see equation 3.1), corresponding to the projection onto the space of constant functions. The following observation allows for a simple definition of the surplus decomposition: the surplus $\boldsymbol{\Delta}_{\boldsymbol{l}}^{\mathrm{P}} F$ having $l_i = 1$ is a function independent of the variables ξ_i. Further, for multi-indices $\boldsymbol{l} = (\boldsymbol{l}_{\mathtt{p}}, \boldsymbol{l}_{\mathtt{u}})$, where $\boldsymbol{l}_{\mathtt{p}}$ and $\boldsymbol{l}_{\mathtt{u}}$ are sub-multi-indices related to directions $\mathtt{p}$ and $\mathtt{u}$ respectively, the projection surplus $\boldsymbol{\Delta}_{\boldsymbol{l}}^{\mathrm{P}} F$ is a function of a subset of variables in $\boldsymbol{\xi}_{\mathtt{p}}$ (resp. $\boldsymbol{\xi}_{\mathtt{u}}$) only, if $|\boldsymbol{l}_{\mathtt{p}}| > d_{\mathtt{p}}$ and $|\boldsymbol{l}_{\mathtt{u}}| = d_{\mathtt{u}}$ (resp. $|\boldsymbol{l}_{\mathtt{p}}| = d_{\mathtt{p}}$ and $|\boldsymbol{l}_{\mathtt{u}}| > d_{\mathtt{u}}$). That is, at least one index of $\boldsymbol{l}_{\mathtt{p}}$ is greater than 1 and all indices of $\boldsymbol{l}_{\mathtt{u}}$ equal 1. Since by construction the multi-index $\{1, \cdots, 1\} \notin \mathcal{A}$, this suggests the partition of $\mathcal{A}$ into three disjoint subsets

$$\mathcal{A} = \mathcal{A}_{\mathtt{p}} \cup \mathcal{A}_{\mathtt{u}} \cup \mathcal{A}_{\mathtt{p,u}},$$

where

$$\mathcal{A}_{\mathtt{p}} \doteq \{\boldsymbol{l} = (\boldsymbol{l}_{\mathtt{p}}, \boldsymbol{l}_{\mathtt{u}}) \in \mathcal{A}, |\boldsymbol{l}_{\mathtt{u}}| = d_{\mathtt{u}}\}, \quad \mathcal{A}_{\mathtt{u}} \doteq \{\boldsymbol{l} = (\boldsymbol{l}_{\mathtt{p}}, \boldsymbol{l}_{\mathtt{u}}) \in \mathcal{A}, |\boldsymbol{l}_{\mathtt{p}}| = d_{\mathtt{p}}\}, \tag{4.9}$$

and $\mathcal{A}_{\mathtt{p,u}} \doteq \mathcal{A} \backslash (A_{\mathtt{p}} \cup \mathcal{A}_{\mathtt{u}})$. The decomposition of $\mathcal{A}$ into $\mathcal{A}_{\mathtt{p}}$, $\mathcal{A}_{\mathtt{u}}$, and $\mathcal{A}_{\mathtt{p,u}}$ is illustrated for a schematic example in Fig. 4.1.

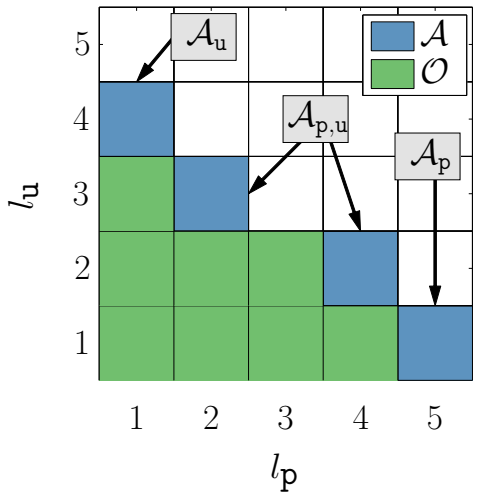


FIGURE 4.1: Schematic example of the decomposition of $\mathcal{A}$ into $\mathcal{A}_{\mathtt{p}}$, $\mathcal{A}_{\mathtt{u}}$, and $\mathcal{A}_{\mathtt{p,u}}$.

The estimator η of the available surplus norm is broken accordingly, resulting in

$$\eta^2(\mathcal{A}) = \sum_{\boldsymbol{l}\in\mathcal{A}} \epsilon^2(\boldsymbol{l}) = \eta_{\mathtt{p}}^2 + \eta_{\mathtt{u}}^2 + \eta_{\mathtt{p},\mathtt{u}}^2, \quad \eta_{\bullet}^2 = \sum_{\boldsymbol{l}\in\mathcal{A}_{\bullet}} \epsilon^2(\boldsymbol{l}). \tag{4.10}$$

Note that having $\boldsymbol{l} \in \mathcal{A}_{\mathtt{p},\mathtt{u}}$ does not imply that it is associated to a projection surplus depending necessarily on a set of variables belonging to both $\boldsymbol{\xi}_{\mathtt{p}}$ and $\boldsymbol{\xi}_{\mathtt{u}}$, nor that its Sobol decomposition yields $(\boldsymbol{\Delta}_{\boldsymbol{l}}^{\mathrm{P}} F)_{\mathtt{p}} = (\boldsymbol{\Delta}_{\boldsymbol{l}}^{\mathrm{P}} F)_{\mathtt{u}} = 0$, because of external aliasing and quadrature error. Nonetheless, the indicators $\eta_{\mathtt{p}}$, $\eta_{\mathtt{u}}$ and $\eta_{\mathtt{p},\mathtt{u}}$ can effectively be used to construct directional stopping criteria in the adaptive procedure. The main advantage of using the proposed indicators, compared to performing the actual Sobol decomposition of the global projection surplus in (4.8), is that the update of the η's during the adaptive process is immediate, because the indicators $\epsilon(\boldsymbol{l})$ are easy to compute (locally) and do not change when new tensorizations are introduced in the sparse grid.

Adaptive Strategies

The directional adaptive methods proposed below follow the usual incremental completion of the tensorizations set, adding to $\mathcal{O}$ the critical tensorization $\boldsymbol{l}^*$ with the highest indicator ϵ, and completing $\mathcal{A}\backslash\{\boldsymbol{l}^*\}$ with elements of the forward neighbors of $\boldsymbol{l}^*$ maintaining the $\mathcal{O}$ admissibility. The process is repeated until $\mathcal{A} = \varnothing$ (exhaustion of available tensorization) or $\eta^2 < \mathtt{Tol}^2$, where $\mathtt{Tol}$ is a user defined global tolerance. Here, we further restrict the set of admissible tensorizations for the completion of $\mathcal{A}$, whenever a convergence is achieved along the $\mathtt{p}$ or $\mathtt{u}$ directions. To this end, we first introduce two convergence criteria associated to the $\mathtt{p}$ and $\mathtt{u}$ dimensions,

$$C_{\mathtt{p}}(\eta_{\mathtt{p}}, \eta_{\mathtt{p},\mathtt{u}}) < \mathtt{Tol}_{\mathtt{p}}^2, \quad C_{\mathtt{u}}(\eta_{\mathtt{u}}, \eta_{\mathtt{p},\mathtt{u}}) < \mathtt{Tol}_{\mathtt{u}}^2,$$

where $C_{\mathtt{p}}$ and $C_{\mathtt{u}}$ are appropriate combinations of the directional and mixed indicators, such that $C_{\bullet}(\eta_{\bullet}, \eta_{\mathtt{p},\mathtt{u}}) \leqslant \eta^2$ for $\bullet = \mathtt{p}, \mathtt{u}$. For instance, one may consider

$$C_\bullet(\eta_\bullet, \eta_{\mathtt{p},\mathtt{u}}) = \eta_\bullet^2 + \alpha\, \eta_{\mathtt{p},\mathtt{u}}^2, \quad 0 \leqslant \alpha \leqslant 1. \tag{4.11}$$

Note that the tolerances $\mathtt{Tol}_\mathtt{p}$ and $\mathtt{Tol}_\mathtt{u}$, and the parameter α should generally be carefully selected, particularly when the estimates $C_\mathtt{p}$ and $C_\mathtt{u}$ have disparate magnitudes.

If none of the directional criteria is satisfied, we apply the classical method using all forward neighbors $\mathcal{F}(\boldsymbol{l}^*) = \mathcal{F}_\mathtt{p}(\boldsymbol{l}^*) \cup \mathcal{F}_\mathtt{u}(\boldsymbol{l}^*)$, with $\mathcal{F}_\mathtt{p}(\boldsymbol{l}^*) \doteq \{(\boldsymbol{l}_\mathtt{p}^* + \hat{\mathtt{e}}_i, \boldsymbol{l}_\mathtt{u}^*), i = 1, \cdots, d_\mathtt{p}\}$ the $\mathtt{p}$-forward neighbors of $\boldsymbol{l}^*$ and similarly for their $\mathtt{u}-$counterpart $\mathcal{F}_\mathtt{u}(\boldsymbol{l}^*)$. On the contrary, when the $\mathtt{p}$ (resp. $\mathtt{u}$) directional criterion is satisfied, we only consider $\tilde{\mathcal{F}}_\mathtt{p} \subseteq \mathcal{F}_\mathtt{p}$ (resp. $\tilde{\mathcal{F}}_\mathtt{u} \subseteq \mathcal{F}_\mathtt{u}$) forward neighbors. Different reduction strategies can be conceived. In the computational examples below, we investigate the following two alternatives:

T1: Restrict the inclusion of the forward neighbors in converged directions to only tensorizations of levels lower or equal to the highest levels reached in these directions. Specifically, denoting $\bullet = \mathtt{p}$ or $\mathtt{u}$, the restricted forward neighbors are defined by

$$\tilde{\mathcal{F}}_\bullet(\boldsymbol{l}^*) \equiv \begin{cases} \{\boldsymbol{l} \in \mathcal{F}_\bullet(\boldsymbol{l}^*), \boldsymbol{l}_\bullet \leqslant \tilde{\boldsymbol{l}}_\bullet, \} & C_\bullet < \mathtt{Tol}_\bullet^2, \\ \mathcal{F}_\bullet(\boldsymbol{l}^*) & \text{otherwise,} \end{cases}$$

where $\tilde{\boldsymbol{l}}_\bullet$ is defined component-wise as the largest $\bullet$-multi-index in $\mathcal{O} \cup \mathcal{A}$, i.e.

$$(\tilde{l}_\bullet)_i = \max_{\boldsymbol{l}_\bullet \in \mathcal{O} \cup \mathcal{A}} (l_\bullet)_i, \quad 1 \leqslant i \leqslant d_\bullet.$$

Likewise, the inequality between two multi-indices, $\boldsymbol{l} \leqslant \boldsymbol{l}'$, is understood component-wise, that is for multi-indices with length d

$$\boldsymbol{l} \leqslant \boldsymbol{l}' \Leftrightarrow l_i \leqslant l_i' \text{ for } 1 \leqslant i \leqslant d.$$

Note that in general, strategy T1 leads to the inclusion of additional forwards neighbors with either mixed $\mathtt{p}-\mathtt{u}$ dependence such that $\boldsymbol{l}^*_\bullet \leqslant \tilde{\boldsymbol{l}}_\bullet$, or dependence only in directions other than $\bullet$.

T2: The second strategy is more radical in that no tensorizations along converged directions are considered for the completion of $\mathcal{A}$. That is,

$$\tilde{F}_\bullet(\boldsymbol{l}^*) \equiv \begin{cases} \varnothing & C_\bullet < \mathtt{Tol}_\bullet^2, \\ F_\bullet(\boldsymbol{l}^*) & \text{otherwise,} \end{cases}.$$

The adaptation proceeds classically by completing $\mathcal{A}\backslash\{\boldsymbol{l}^*\}$ with the elements of $\tilde{F}_\mathtt{p}(\boldsymbol{l}^*) \cup \tilde{F}_\mathtt{u}(\boldsymbol{l}^*)$ maintaining the $\mathcal{O}$ admissibility.

The two strategies T1 and T2 are schematically illustrated in Fig. 4.2 in the case of an achieved directional convergence, where it is assumed $C_\mathtt{p} = \eta_\mathtt{p} < \mathtt{Tol}_\mathtt{p}^2$. Notice that in Fig. 4.2 (b) the tensorization $\{3,2\}$ is added, but not in Fig. 4.2 (e), where tensorization $\{2,3\}$ *is* added despite the halting line because it belongs to the $\mathtt{u}$-forward neighborhood of $\boldsymbol{l}^*$. Note also that in Fig. 4.2 (a) and (d), $\boldsymbol{l}^* = \{3,1\}$ for which no forward neighbor is added because of $\mathcal{O}$-admissibility ($\{3,2\}$ missing) and because the $\mathtt{p}$ direction has converged ($\tilde{\boldsymbol{l}}_\mathtt{p} = (3)$). However, when $\boldsymbol{l}^*$ *is* removed from $\mathcal{A}$ we have $\mathcal{A}_\mathtt{p} = \varnothing$, making $\eta_\mathtt{p} = C_\mathtt{p} = 0$. This is not a concern since this only occurs after convergence in that dimension.

Adaptation continues until either *both* $C\mathtt{p} \leqslant \mathtt{Tol}_\mathtt{p}^2$ and $C\mathtt{u} \leqslant \mathtt{Tol}_\mathtt{u}^2$ or, until the global $\eta \leqslant \mathtt{Tol}^2$. Unlike the nested adaptation, the product-space adaption has both independent convergence metrics and a global convergence metric. Also, whereas extension of the nested adaptation approach to greater than two spaces is not recommended, the product-space construction trivially lends itself to any number of spaces or groupings of dimensions.

Note that the present aPSP scheme with directionally tuned refinement naturally exploits sparse-tensorization in the product space, and is consequently expected to be

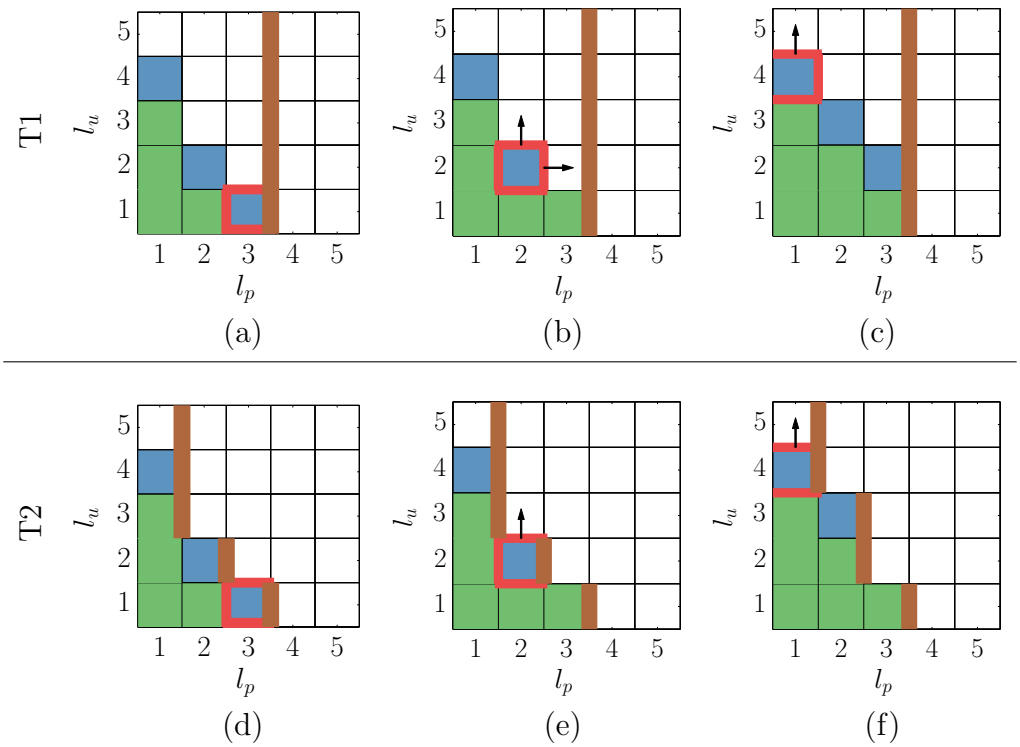


FIGURE 4.2: Schematic demonstration of strategy T1 (top plots, a-c) and T2 (bottom plots, d-f) for the case $\eta_{\mathtt{p}}^2 < \mathtt{Tol}_{\mathtt{p}}$. Shown in blue is $\mathcal{A}$, and in green is $\mathcal{O}$. The red box denotes $\boldsymbol{l}^*$ and the thick-brown line represents the halting-condition imposed by the two strategies.

more efficient than the nested algorithm. This is achieved at the potential expense of a coarser control of the resolution properties along the `p` and `u` directions. The effect of this compromise is examined in the analysis below, together with a comparative analysis of the performance of the two algorithms.

4.3 Low-Dimensional Test Problem

4.3.1 Test Problem Definition

In order to examine the behavior of the methods presented in Section 4.2, we make use of the two-dimensional ($d = 2$) test function,

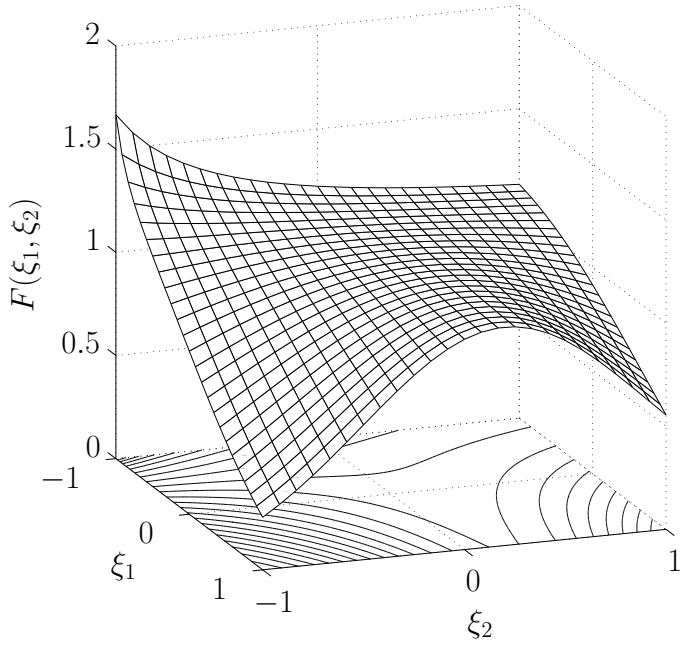
$$F(\xi_1, \xi_2) = \left(1 + \frac{1/3}{2\xi_1 + \xi_2 + 7/2}\right) \exp\left[-\left(\frac{1}{2}\left(\xi_2 - \frac{1}{5}\right)(\xi_2 + 1)\right)^2\right], \quad \boldsymbol{\xi} \in [-1, 1]^2, \tag{4.12}$$

allowing us to rapidly evaluate performance and to conduct error analyses. The function F (Fig. 4.3) was designed so as to exhibit a mixed dependence in ξ_1 and ξ_2 and a greater dependence on ξ_1 as measured by the Sobol sensitivity indices of F which are $S_{\{1\}} = 0.531$, $S_{\{2\}} = 0.199$, and $S_{\{1,2\}} = 0.270$ (see Appendix B). Because the nested method adapts differently the inner and outer directions, we shall call (1,2)-Nested method the case where the inner-loop acts on ξ_2 (that is $\mathtt{u} = \{2\}$) and the outer-loop acts on ξ_1 ($\mathtt{p} = \{1\}$), whereas for the (2,1)-Nested method the roles of these directions are exchanged. To illustrate the quantities of interest driving the adaptation in the (1,2)-Nested method, Fig. 4.3 also depicts for a set of values $\xi_1^{(i)}$ the cross-sections $F(\xi_1^{(i)}, \xi_2) = G^{(i)}(\xi_2)$ used for the inner-loop adaptation, as well as the outer-loop quantity of interest $H(\xi_1) = \|F(\xi_1, \cdot)\|_{L_2([-1,1])}$.

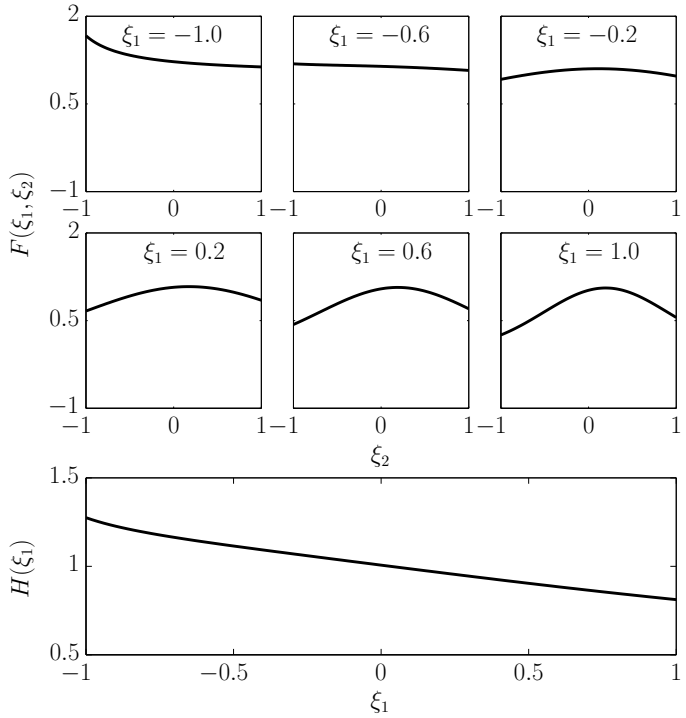
4.3.2 Error Analysis

Let us denote here $\widetilde{F}(\boldsymbol{\xi})$ the PCE surrogate of F obtained with one of the adaptive projection methods, and $Z(\boldsymbol{\xi})$ the error function:

$$Z(\boldsymbol{\xi}) \doteq F(\boldsymbol{\xi}) - \widetilde{F}(\boldsymbol{\xi}) = F(\boldsymbol{\xi}) - \sum_{\boldsymbol{k} \in \mathcal{K}} \widetilde{f}_{\boldsymbol{k}} \Psi_{\boldsymbol{k}}(\boldsymbol{\xi}). \tag{4.13}$$



(a) Function F



(b) Functions for the (1,2)-Nested methods.

FIGURE 4.3: The plot in (a) shows the 2D surface $F(\xi_1, \xi_2)$. The plots in (b) depict the functions used in the (1,2)-Nested method: cross-sections $G^{(i)}(\xi_2)$ for selected values of $\xi_1^{(i)}$, as indicated (six top plots), and the function $H(\xi_1)$ for outer-loop adaptation (bottom).

Introducing the exact (but unknown) PC expansion of F, Z can be split into two orthogonal contributions:

$$F(\boldsymbol{\xi}) = \sum_{\boldsymbol{k}\in\mathbb{N}^d} f_{\boldsymbol{k}}\Psi_{\boldsymbol{k}}(\boldsymbol{\xi}) \quad \Rightarrow \quad Z(\boldsymbol{\xi}) = \underbrace{\sum_{\boldsymbol{k}\in\mathcal{K}} \left(f_{\boldsymbol{k}} - \widetilde{f}_{\boldsymbol{k}}\right)\Psi_{\boldsymbol{k}}(\boldsymbol{\xi})}_{\text{External aliasing}} + \underbrace{\sum_{\boldsymbol{k}\in\mathbb{N}^d\setminus\mathcal{K}} f_k\Psi_k(\boldsymbol{\xi})}_{\text{Projection error}}.$$

Another decomposition, more relevant to the task of error control, consists of orthogonal Sobol decomposition (see Appendix B) of Z according to:

$$Z(\boldsymbol{\xi}) = Z_{\varnothing} + Z_{\{1\}}(\xi_1) + Z_{\{2\}}(\xi_2) + Z_{\{1,2\}}(\xi_1,\xi_2).$$

The *relative* error norm, ζ, can then be decomposed into

$$\zeta^2 \doteq \frac{\|Z\|^2_{L_2}}{\|F\|^2_{L_2}} = \zeta^2_{\varnothing} + \zeta^2_{\{1\}} + \zeta^2_{\{2\}} + \zeta^2_{\{1,2\}}, \tag{4.14}$$

where $\zeta_\bullet = \|Z_\bullet\|_{L_2}/\|F\|_{L_2}$. Thus, $\zeta_{\varnothing}$ measures the relative error on the mean, $\zeta_{\{1\}}$ (resp. $\zeta_{\{2\}}$) measure the relative error in direction ξ_1 (resp. ξ_2) only, and $\zeta_{\{1,2\}}$ measures the relative error in mixed directions. Note that in higher dimension, this decomposition of the error norm can be extended to arbitrary partitions of the dimensions.

Following the discussion in Appendix B.2, the relative errors can be easily computed from the PCE of Z and F. Since for the present example an explicit PCE for F in (4.12) is not available, we instead construct a highly accurate projection of F on a 90^{th} total-order polynomial basis using full-tensored Gauss-Legendre [1] rule having a total of 100^2 quadrature points.

4.3.3 Behavior of the Adaptive Methods

In this section, we analyze and contrast the performance of the different adaptive schemes developed above. To this end, computations are performed for the test

problem, using the tolerances reported in Table 4.1. Recall that the nested projection uses one tolerance for each direction, $\texttt{Tol}_{\{1\}}$ and $\texttt{Tol}_{\{2\}}$, that serve as stopping criteria for the inner or outer loops depending if the (1,2) or (2,1)-Nested method is considered. The aPSP uses a global tolerance $\texttt{Tol}$ and the aPSP-T1 and aPSP-T2 methods (see Section 4.2.2) have in addition directional tolerances $\texttt{Tol}_1$ and $\texttt{Tol}_2$. When $\texttt{Tol}_1 = \texttt{Tol}_2 = 0$, one recovers the original aPSP method and is then used as reference in the presented results.

Table 4.1: Tolerance values for different adaptive methods used for the test problem.

	Nested	aPSP (reference)	aPSP-T1, 2
$\texttt{Tol}$	–	10^{-5}	10^{-5}
$\texttt{Tol}_1$	10^{-7}	0	10^{-7}
$\texttt{Tol}_2$	10^{-9}	0	10^{-9}

Adaptive Product Space Projection with Directional Criteria

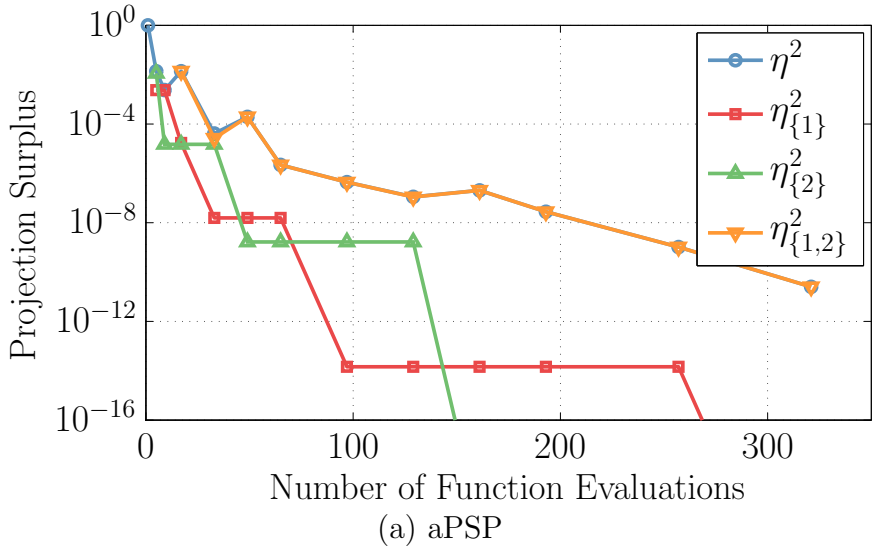
Due to the low-dimensionality of the test problem, the product-space adaptation was performed for the tolerance values in Table 4.1 with $\alpha = 0$ in (4.11). Figure 4.4 depicts the squared projection surplus norms, η^2, and their Sobol decompositions into pure and mixed contributions, during the adaptive process for the aPSP and aPSP-T1 methods. Results for aPSP-T2 are not shown as they lead to results similar to those for aPSP-T1. The evolutions of the adaptation criteria are reported, as a function of the number of evaluations of F required by the two methods. The plots show that the two methods follow a similar path but that aPSP-T1 terminates earlier as the dimensional tolerances are met. In addition, it is seen that $\eta_{\{1\}}$ in the aPSP-T1 method becomes zero after few adaptive iterations, for $\mathcal{A}_{\{1\}} = \varnothing$. However, this occurs after $\eta_{\{1\}}$ drops below the specified tolerance $\texttt{Tol}_1$, indicated by a dashed line in Fig. 4.4(b), as we would have expected. Note that for the two

methods, $\eta^2_{\{1,2\}} \gg \eta^2_{\{1\}}, \eta^2_{\{2\}}$ indicating that much of the remaining surplus is due to mixed terms. The results thus confirm that method aPSP-T1 is able to control the adaptivity for separate sets of directions and to control termination of the refinement along the corresponding dimensions.

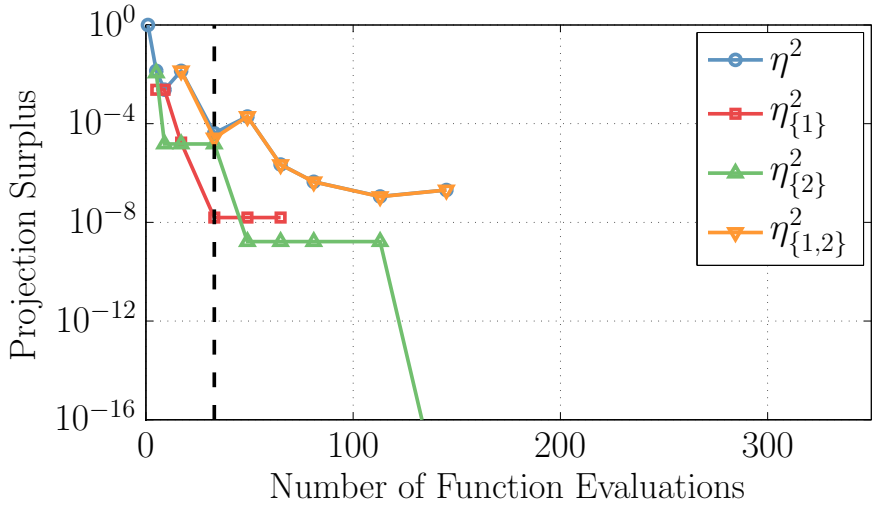
To further illustrate the behavior of the directionally-tuned adaption, we plot in Fig. 4.5 the pseudo-spectral projection multi-index sets, $\mathscr{L}$, the selection indicators $\epsilon(\boldsymbol{k}) \ \forall \ \boldsymbol{k} \in \mathscr{L}$, and the adaptive paths for (a) the aPSP, (b) the aPSP-T1 and (c) the aPSP-T2 methods. The number inside each box is the iteration at which each index is added. The results indicate that, as expected for the present settings, the reference aPSP method does include more mixed multi-indices ($\boldsymbol{l} = (l_1, l_2)$ with $l_1, l_2 > 1$) compared to the other two methods, corresponding to a richer projection space. Indeed, for aPSP-T1 and aPSP-T2 the adaptation halts refining sooner than aPSP along direction $\{1\}$, which is associated to a larger $\texttt{Tol}_1$ value, while still refining along direction $\{2\}$ as for aPSP. The plots also highlight the effect of selecting strategy T1 or T2, the later being more aggressive in the sense that it results in a coarser multi-index set $\mathscr{L}$ with fewer mixed tensorizations than for aPSP-T2.

Nested Projection

We now focus on the behavior of the nested adaptation with the tolerance values reported in Table 4.1. The adaptation was performed for both the (1,2) and (2,1)-Nested algorithms. Figure 4.6 presents the final grids of points where the function has been evaluated, for the reference aPSP method and the two nested projection alternatives. The results indicate that, in contrast to aPSP, the nested adaptations exhibit a *nearly* fully-tensorized grids of points between the two spaces and a larger number of function evaluations as a consequence. The plots also illustrate how at different outer-direction grid points, the inner-space grids are not necessarily identical.

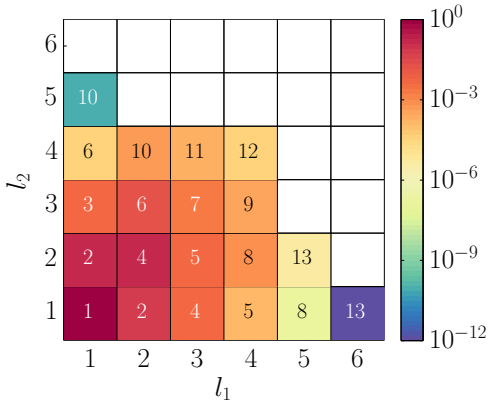


(a) aPSP

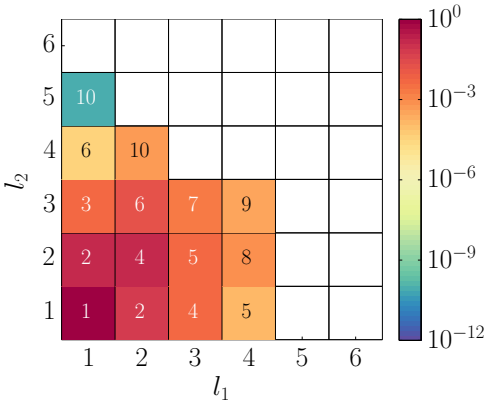


(b) aPSP-T1

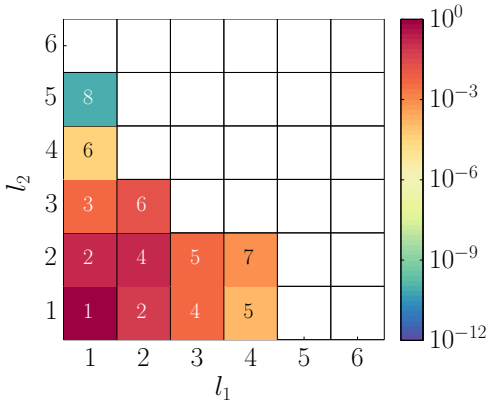
FIGURE 4.4: Relative projection surplus η^2 and their Sobol decompositions versus the number of function evaluations for the test function in Equation (4.12). Plotted in (a) are the results for aPSP (reference). Plot (b) shows the results of the aPSP-T1. The dashed line in (b) represents the point at which the aPSP-T1 converge in direction {1}.



(a) Reference



(b) Method T1



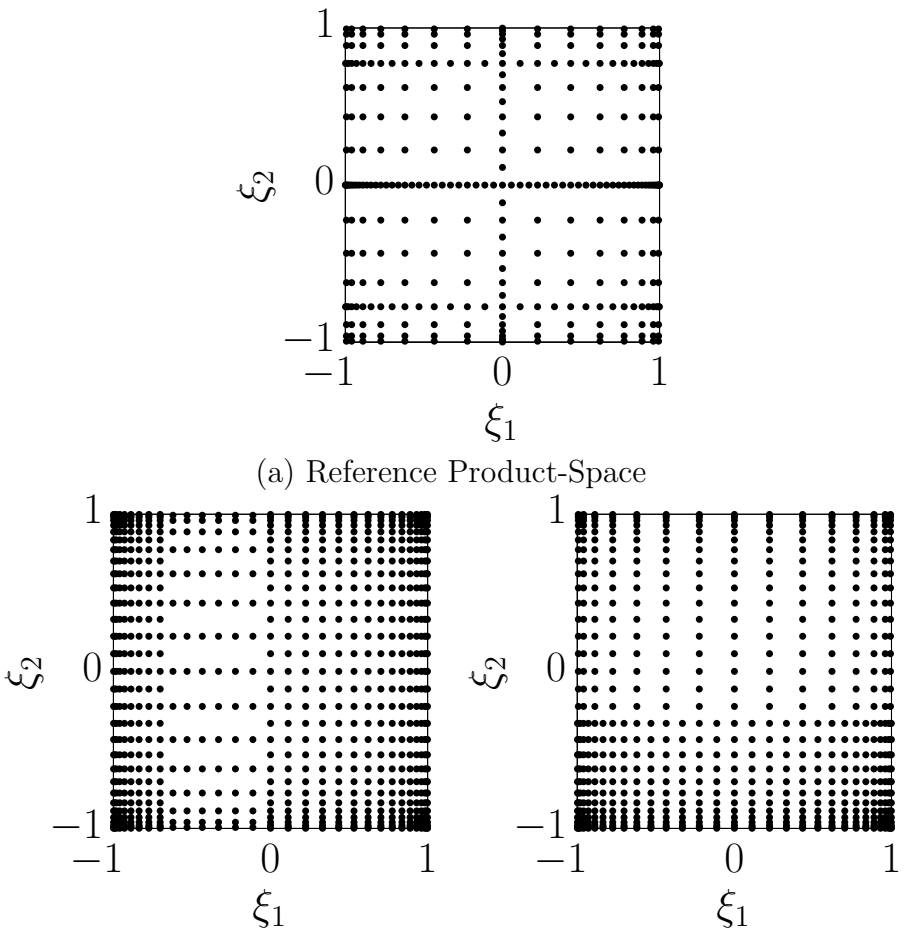
(c) Method T2

FIGURE 4.5: Final multi-index sets for the pseudo-spectral projection using (a) aPSP, (b) aPSP-T1 and (c) aPSP-T2 methods. Also shown are the color-coded values of the norm of the projection surpluses $\epsilon(\boldsymbol{l})$ (3.4) and the iteration number at which that multi-index was added to $\mathscr{L}$.

4.3.4 Error and Performance Analysis

Figure 4.7 shows the global error estimates, ζ^2, plotted against the number of function evaluations, for: (a) the (1,2)-Nested method, (b) the (2,1)-Nested Method, (c) the aPSP method, and (d) the aPSP-T1 method. Results for aPSP-T2 are similar to those of aPSP-T1, and are consequently omitted. Plotted along with ζ^2 is its decomposition based on (4.14).

The results in Fig. 4.7(a) and 4.7(b) illustrate an intrinsic feature of the nested



(a) Reference Product-Space

(b) (1,2)-Nested grid

(c) (2,1)-Nested grid

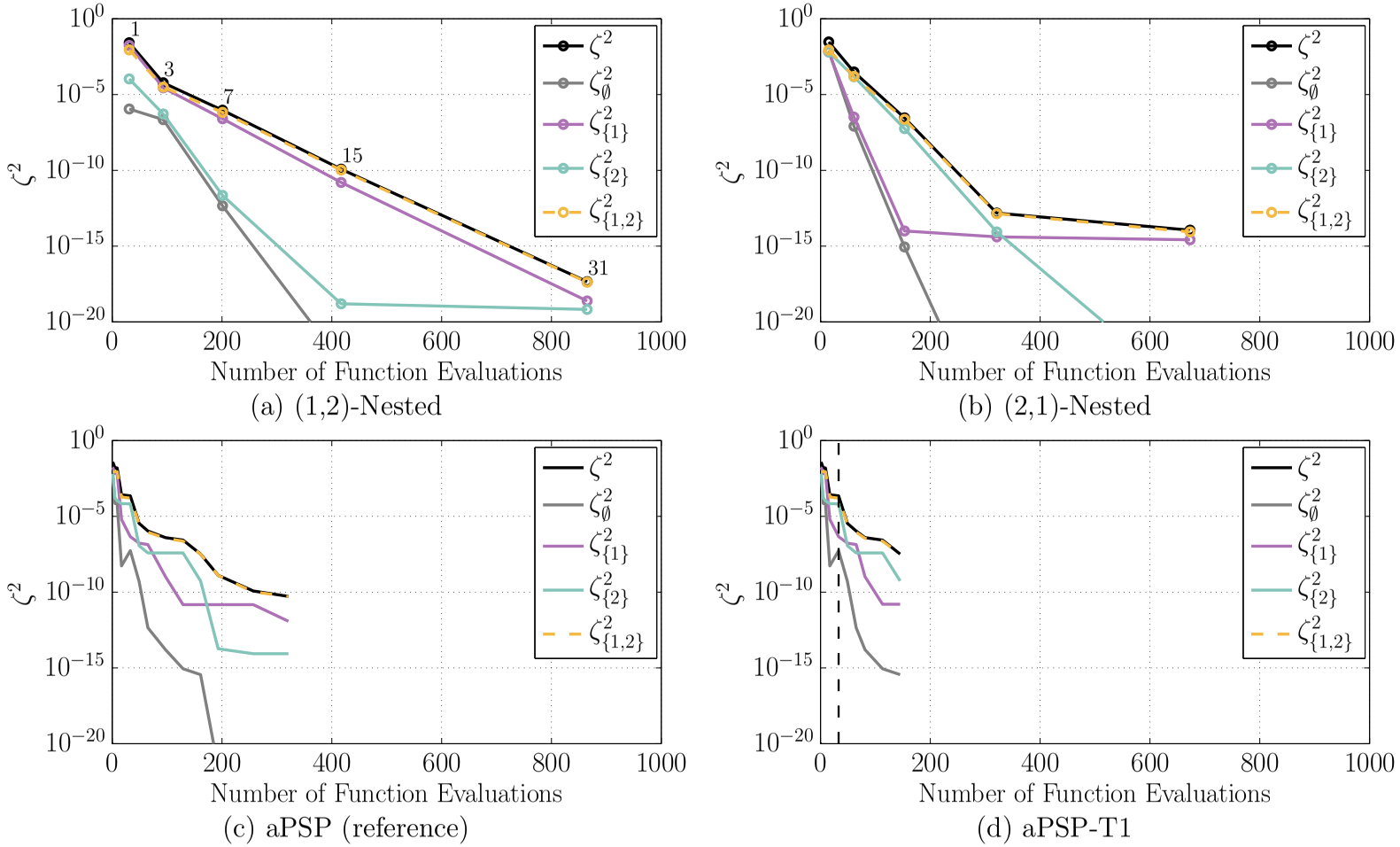
FIGURE 4.6: Comparison of the final grids $\mathcal{G}(\mathscr{L})$ for (a) aPSP method, (b) the (1,2)-Nested adaptation, and (c) (2,1)-Nested adaptation.

construction, namely that the error associated with inner dimension is reduced immediately. This can be appreciated by the fact that $\zeta^2_{\{2\}}$ in Fig. 4.7(a) and the $\zeta^2_{\{1\}}$ in Fig. 4.7(b) decrease rapidly. Meanwhile, the global error closely follows the error in the outer dimension. Note that for the present example, the mixed-term error $\zeta_{\{1,2\}}$ has a dominant contribution to the global error in both the nested, aPSP and aPSP-T1 methods. This observation, however, need not hold in general, particularly in situations when F has essentially an additive form: $F(\boldsymbol{\xi}_{\mathtt{p}}, \boldsymbol{\xi}_{\mathtt{u}}) \approx F_{\mathtt{p}}(\boldsymbol{\xi}_{\mathtt{p}}) + F_{\mathtt{u}}(\boldsymbol{\xi}_{\mathtt{u}})$.

The performance of the aPSP-T1 can be assessed by contrasting the results in Figs. 4.7(c) and 4.7(d). Specifically, the results indicate that, consistent with the behavior of the η^2 indicators in Fig. 4.4, the aPSP terminates with a lower global error ζ^2; however, this comes at the cost of more function evaluations. The results also illustrate the effectiveness of aPSP-T1 in guiding and halting the refinement in direction $\{1\}$, with a small impact on the global error. This feature will prove especially useful in a higher dimensional applications, where tighter control on directional refinement can be substantially advantageous.

Finally, contrasting the results in Figs. 4.7(a) and (b) with those in Figs. 4.7(c) and (d), the following general conclusion can be drawn from the test example. First, all adaptive methods exhibit a similar complexity, in the sense that the global errors have similar decay rates with the number of function evaluations. Second, the nested method offers an efficient means to reducen at a minimal costn the error along the inner direction. Third, the aPSP-T1, 2 methods offer flexibility to balance and halt the adaptation along specific directions.

Finally, we examine the impact of setting to zero the α_i^k coefficients in Equation (4.6) in the nested construction at outer grid points where these coefficients are not defined. To this end, we performed a computation where the missing coefficients were actually computed by means of an inner projection based on the union of the inner grids at all outer points. In other words, the nested grid was completed to a full-tensorization grid, requiring additional evaluations of the function. For the present example and other numerical tests (not shown) it was found that the resulting improvement in the global error was marginal while the completion of the grid could require a significant number of function evaluations, with an overall degradation of the efficiency. This point is illustrated for the present example in Fig. 4.8.



(a) (1,2)-Nested

(b) (2,1)-Nested

(c) aPSP (reference)

(d) aPSP-T1

FIGURE 4.7: ζ^2 and its Sobol decomposition versus number of function evaluations, for (a) the (1,2)-Nested, (b) the (2,1)-Nested, (c) the aPSP, and (d) aPSP-T1 methods.

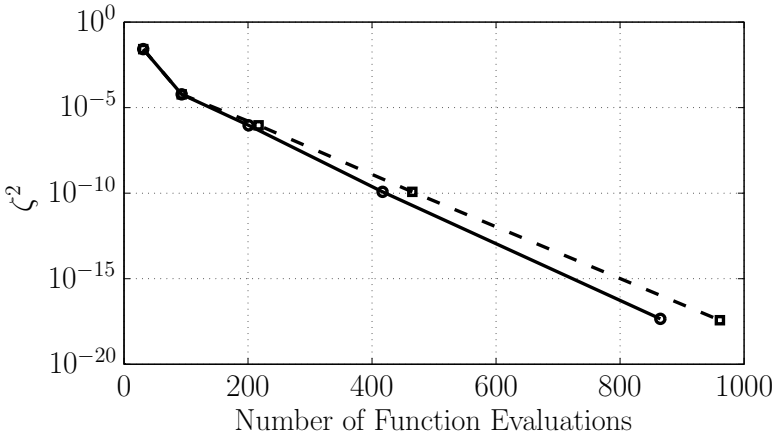


FIGURE 4.8: Comparison of the global error ζ^2 versus number of function evaluations, for the nested method (solid line) and its completion to full-tenzorization (dotted line) for the test problem.

4.4 High-Dimensional Problem

This section illustrates the behavior of the methods outlined above for a high-dimensional combustion chemistry problem that had motivated their development. The problem consists in the optimal design of shock-tube ignition experiments, and thus makes use of global surrogates that describe the dependence of key experimental observables (i.e. QoIs) on both uncertain rate parameters (to be calibrated by yet unperformed experiments) and design variables (to be optimally selected based on prior information).

The shock tube experiment presently considered focuses on methane ignition. The reaction model is based on the GRI-Mech 3.0 [114] mechanism, involving 57 neutral species and 325 neutral species reactions. The model is augmented with ion-chemistry mechanism developed by Prager *et. al* [100], including 11 charged species and 67 ionic reactions.

Since we are primarily interested in characterizing the performance of adaptive constructions, we focus on a single experimental observable, namely the peak electron

concentration that is achieved during the ignition event. The uncertain parameters concern the rate constants of 22 reactions that were deemed especially relevant based on a reaction pathway analysis. Details of this analysis are beyond the scope of the present work, and will be described elsewhere [69]. The ranges of the uncertain reaction rates are specified in terms of factors, as summarized in Table 4.2(a). Based on these factors, the uncertain rates are parametrized using independent random variables that are uniformly distributed in the interval $[-1, 1]$. This leads to a 22-dimensional germ $\boldsymbol{\xi}_{\mathtt{u}}$. We assume that the initial pressure is atmospheric, and focus on three-design variables, namely the initial temperature, and the initial concentrations of CH_4 and O_2. Thus, the design-variable vector, $\boldsymbol{\xi}_{\mathtt{p}}$, is three-dimensional, and the ranges of its components are specified in Table 4.2(b). We associate with $\boldsymbol{\xi}_{\mathtt{p}}$ a constant weight $\rho_{\mathtt{p}}$. Once design variables are selected, the evolution of the system for a specific realization of the germ $\boldsymbol{\xi}_{\mathtt{u}}$ is obtained using an extended version of the TChem software [108, 107]. The problem is then set in terms of the germ $\boldsymbol{\xi} = (\boldsymbol{\xi}_{\mathtt{u}}, \boldsymbol{\xi}_{\mathtt{p}}) \in [-1, 1]^{22+3}$ with density $\rho_{\boldsymbol{\xi}} = 2^{-25}$.

We applied the nested aPSP and aPSP-T1 methods using the the tolerance values listed in Table 4.3. These values were set so as to obtain a suitably accurate approximation of the selected QoI. Also note that in the present case, d_u is significantly larger than d_p. Consequently, we used lower tolerance values for the $\mathtt{u}$ directions than for the $\mathtt{p}$ directions.

We initially ran the aPSP-reference and found that the QoI exhibits a large sensitivity to the design variables $\boldsymbol{\xi}_{\mathtt{p}}$. This can be appreciated from Fig. 4.9, which provides the first-order and total sensitivity indices, estimated from the PCE after the algorithm has converged. The results clearly illustrate the dominant contribution of the design variables to the overall variance of the peak electron concentration. The results also indicate that for the present case there appears to be only a small contribution in the mixed term, specifically accounting for about 4% of the total

Table 4.2: (a) Uncertainty in rate parameters, and (b) ranges of design variables, for the CH_4+O_2 shock-tube ignition problem. The uncertainty in the rate parameters are specified either in terms of a percentage are relative, or as a multiplicative factor. In all cases, the distributions are represented in terms of canonical random variables uniformly distributed over $[-1, 1]$.

(a) Uncertain Parameters ($\boldsymbol{\xi}_{\mathrm{u}}$-space)

ID	Reaction	Uncertain Range
326	$CH + O \Leftrightarrow HCO^+ + E^-$	$\pm 50\%$
328	$HCO^+ + H_2O \Leftrightarrow H_3O^+ + CO$	$\pm 50\%$
330	$H_3O^+ + E^- \Leftrightarrow H_2O + H$	$\pm 25\%$
331	$H_3O^+ + E^- \Leftrightarrow OH + H + H$	$\pm 25\%$
332	$H_3O^+ + E^- \Leftrightarrow H_2 + OH$	$\pm 25\%$
333	$H_3O^+ + E^- \Leftrightarrow O + H_2 + H$	$\pm 25\%$
336	$HCO^+ + CH_3 \Leftrightarrow C_2H_3O^+ + H$	$2^{\pm}$
338	$H_3O^+ + CH_2CO \Leftrightarrow C_2H_3O^+ + H_2O$	$\pm 25\%$
340	$C_2H_3O^+ + O \Leftrightarrow HCO^+ + CH_2O$	$\pm 50\%$
349	$OH^+ + O \Leftrightarrow HO_2 + E^-$	$\pm 50\%$
350	$OH^- + H \Leftrightarrow H_2O + E^-$	$2^{\pm}$
353	$OH^- + CH_3 \Leftrightarrow CH_3OH + E^-$	$2^{\pm}$
361	$O^- + H_2O \Leftrightarrow OH^- + OH$	$2^{\pm}$
366	$O^- + H_2 \Leftrightarrow H_2O + E^-$	$2^{\pm}$
369	$O^- + CO \Leftrightarrow CO_2 + E^-$	$2^{\pm}$
006	$O + CH \Leftrightarrow H + CO$	$\pm 25\%$
049	$H + CH \Leftrightarrow C + H_2$	$1.5^{\pm}$
091	$OH + CH \Leftrightarrow H + HCO$	$\pm 25\%$
093	$OH + CH_2 \Leftrightarrow CH + H_2O$	$\pm 25\%$
125	$CH + O_2 \Leftrightarrow O + HCO$	$\pm 50\%$
126	$CH + H_2 \Leftrightarrow H + CH_2$	$\pm 50\%$
127	$CH + H_2O \Leftrightarrow H + CH_2O$	$3^{\pm}$

(b) Design Variables ($\boldsymbol{\xi}_{\mathrm{p}}$-space)

Design Parameter	Interval
Temperature (K)	$[2250, 2750]$
CH_4 Mole Fraction (%)	$[0.5, 2.0]$
O_2 Mole Fraction (%)	$[0.2, 0.75]$

Table 4.3: Tolerance values for the Nested-($\boldsymbol{\xi}_{\mathtt{u}}, \boldsymbol{\xi}_{\mathtt{p}}$), aPSP (reference), and aPSP-T1 methods with $\alpha = 0.1$ in Equation (4.11).

	Nested	aPSP (reference)	aPSP-T1
$\mathtt{Tol}$	–	2.75×10^{-3}	2.75×10^{-3}
$\mathtt{Tol}_{\mathtt{p}}$	3.50×10^{-3}	0	3.50×10^{-3}
$\mathtt{Tol}_{\mathtt{u}}$	2.50×10^{-3}	0	2.50×10^{-3}

variance; not surprisingly, the computed first-order and total sensitivity indices for the individual parameters have close values.

We examined (not shown) the behavior of the aPSP and aPSP-T1 methods during the adaptation. The aPSP-T1 method followed a similar path to the aPSP, converging early in the $\mathtt{p}$ direction while the $\mathtt{p}$ directions are subsequently refined. We also examined whether the similar adaptation paths were affected by the selected tolerance values, namely by rerunning aPSP-T1 with larger values of $\mathtt{Tol}_{\mathtt{p}}$, thus forcing the refinement along the $\mathtt{p}$ directions to terminate earlier. However, this had little impact on the grid refinement path, even after the specified $\mathtt{Tol}_{\mathtt{p}}$ value was reached. This is not surprising, in light of the fact that the contribution of the mixed terms is small.

The projection surplus, η^2, and its Sobol decomposition are plotted in Fig. 4.10. For both the aPSP and the aPSP-T1 adaptations, $\eta^2_{\mathtt{p}}$ immediately and rapidly decays as the sparse grid is adapted, whereas the mixed direction surplus is clearly dominant. The results also indicate that behavior of the projection surplus and its Sobol components are very similar for both algorithms. This is consistent with our earlier observation that the adaptation of the sparse grids evolve similar for both algorithms, though aPSP-T1 evidently terminates earlier.

Similar to what we observed with the two-dimensional test function in Section 4.3, the dimensionally guided adaptation affected the adaptive refinement path only slightly. Both aPSP and aPSP-T1 exhibit similar decay of the errors indicators

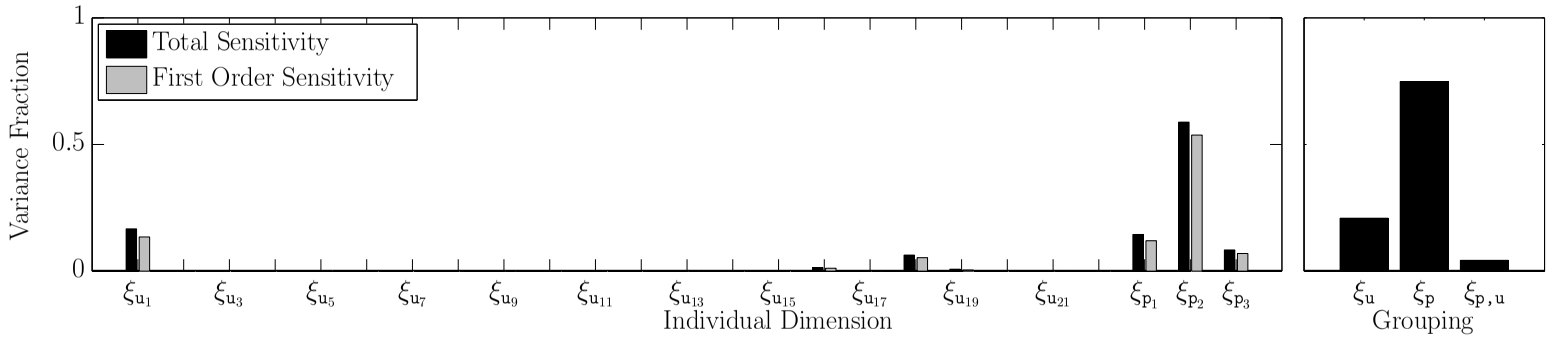
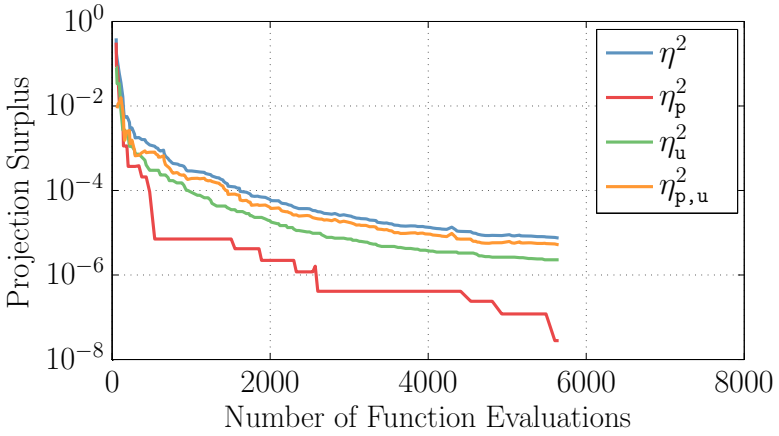
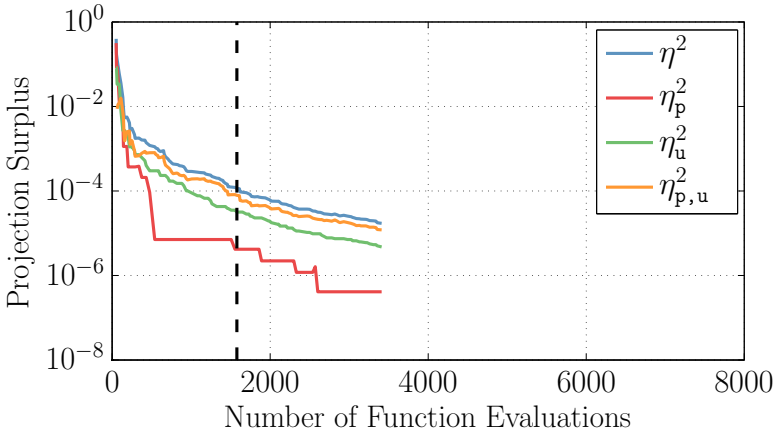


FIGURE 4.9: Sensitivity of the peak electron concentration in the shock-tube ignition problem. Plotted on the left are the first-order and and total sensitivity indices for individual parameters. The right plot depicts the sensitivity indices for the $\boldsymbol{\xi}_{\mathtt{u}}$ and $\boldsymbol{\xi}_{\mathtt{p}}$ spaces, as well as the contribution of the mixed term.



(a) aPSP (reference)



(b) aPSP-T1

FIGURE 4.10: Projection surplus, η, and its Sobol decomposition for (a) aPSP, and (b) aPSP-T1 methods. Also plotted in (b) is a dashed line representing the point at which the adaptation converged in the $\mathtt{p}$ directions.

with the number of model realizations, and as further discussed below similar behavior for errors estimates obtained from independent samples. However, with the selected tolerances, aPSP-T1 terminated with 3411 model evaluations, whereas aPSP required 5651 realizations to reach termination. This illustrates the flexibility of directional criteria in tuning adaptation (and as further illustrated below resolution), along specific dimensions.

We performed the nested adaptation with the 22 uncertain-parameters ($\boldsymbol{\xi}_{\mathtt{u}}$) as

the inner dimensions and the three design parameters ($\boldsymbol{\xi}_{\mathrm{p}}$) as the outer dimensions. The adaptation was allowed to proceed until both $\mathtt{Tol}_{\mathrm{u}}$ and $\mathtt{Tol}_{\mathrm{p}}$ values were met (recall that the nested construction does not involve a global tolerance estimate). As discussed earlier, different grid locations in the ξ_{p} space admit different adaptations into $\boldsymbol{\xi}_{\mathrm{u}}$. The dependence of the number of inner grid points at each point in $\boldsymbol{\xi}_{\mathrm{p}}$ domain is visualized in Fig. 4.11. Notice that, towards some boundaries of $\boldsymbol{\xi}_{\mathrm{p}}$ domain, inner grids are generated with many more points than in other areas. Despite the flexibility afforded in locally adapting at each outer point, at termination the nested method required a total $76,411$ model evaluations. This is over one order of magnitude larger than what was required by the aPSP methods to achieve a comparable error.

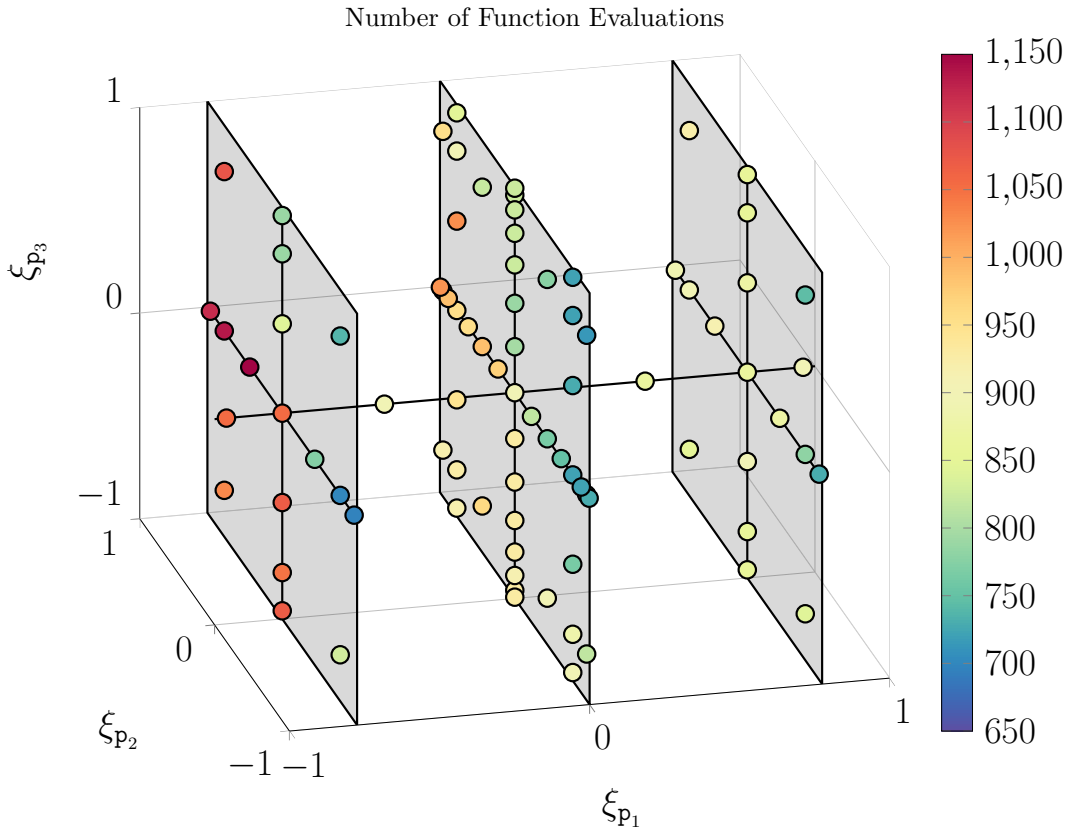


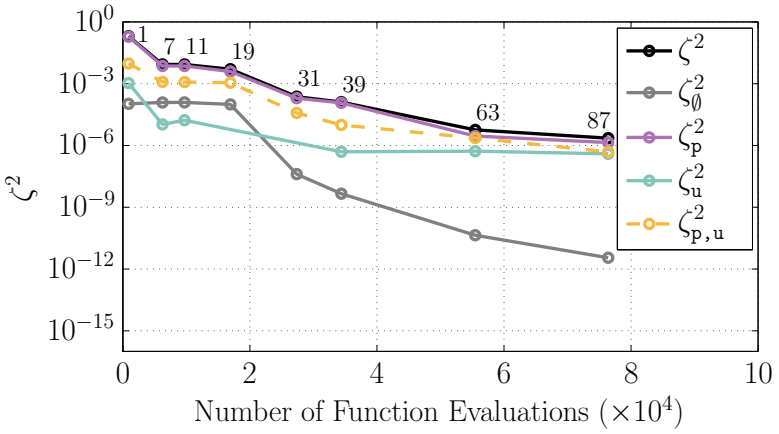
FIGURE 4.11: Number of model evaluations required at each realization of the design variables ($\boldsymbol{\xi}_{\mathrm{p}}$), needed to calculate the local $\boldsymbol{\xi}_{\mathrm{u}}$ projection. Shaded $\boldsymbol{\xi}_{\mathrm{p}_2} - \boldsymbol{\xi}_{\mathrm{p}_3}$ planes are for reference.

To validate the results we generated a Monte-Carlo ensemble consisting of 4096 realizations. An additional 8192 simulations were performed in order to obtain estimates of the Sobol decomposition of the relative error, ζ^2, between the Monte-Carlo sample and the sparse adaptive algorithm (See Appendix B.3). The ζ^2 values, and their Sobol decompositions thus estimated, are presented in Fig. 4.12.

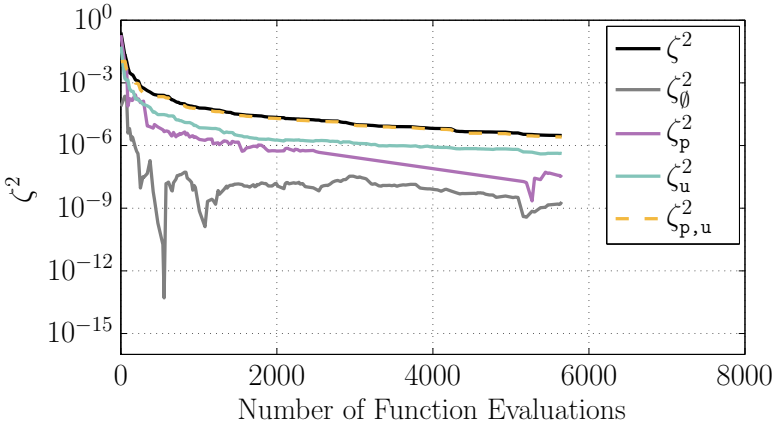
The results of Fig. 4.12(a) demonstrate how the MC estimates of the error in the inner directions are quickly reduced, whereas the global error is predominantly dictated by the refinement in outer dimensions. This is in stark contrast to the behavior seen for the aPSP methods, where the first-order contributions $\zeta_{\mathtt{p}}$ and $\zeta_{\mathtt{u}}$ *both* drop rapidly at early stages of the adaptation while the decay of the global error follows that of the mixed term. The results also indicate that the error control afforded by the nested method comes at the expense of a large computational cost, namely requiring more than an order-of-magnitude increase in the number of model realizations. Figure 4.12(c) demonstrates the effectiveness of the aPSP-T1 method in terminating adaptation along $\mathtt{u}$ directions. The results also illustrate the flexibility of specifying directional criteria, which for the presently selected values result in termination of aPSP-T1 with significantly smaller number of model evaluations than aPSP. Finally, we note that for all three methods, the error estimates obtained based on the MC sample drop to very small values as the iterations converge to their specified tolerances. This provides confidence in the validity of all three methods and in the suitability of the resulting surrogates.

4.5 Conclusion

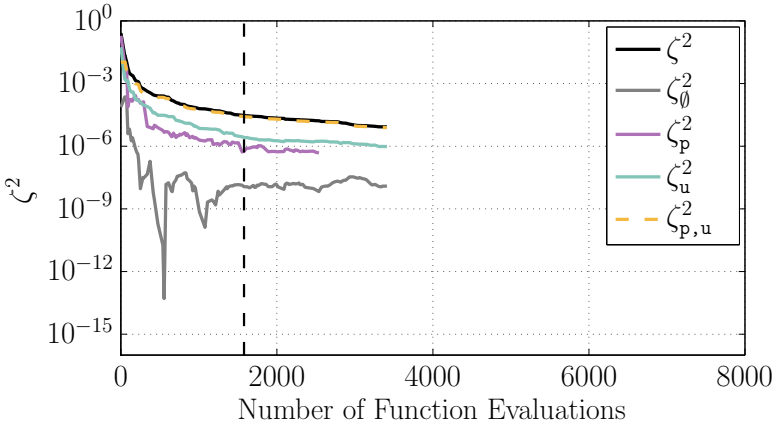
Two approaches were developed for obtaining a global surrogate of QoIs involving parameters belonging to different spaces. The first approach relied on a nested construction where the two spaces were adapted separately. Specifically, at each realization of the outer variables, an adaptive Smolyak pseudo-spectral projection



(a) Nested $(\mathtt{p}, \mathtt{u})$



(b) aPSP (reference)



(c) aPSP-T1

FIGURE 4.12: ζ^2 values for the (a) Nested $(\mathtt{p}, \mathtt{u})$, (b) aPSP, and (c) aPSP-T1 methods. In (a), the value next to individual points refer to the number of outer ($\boldsymbol{\xi}_{\mathtt{p}}$) realizations. The dotted vertical line in (c) represents the point at which $\mathtt{Tol}_{\mathtt{p}}$ is reached. Note that the scale of the x–axis differs in (a) from (b,c). We also note that the ζ values may oscillate or be omitted due to the small ζ value and size of the MC sample.

was performed in the inner space. The local PC coefficients at the outer grid points were then projected in the outer space.

The second approach was based on considering a suitable product space combining all parameters, and on performing an adaptive pseudo-spectral projection in this product space. In addition, we developed a decomposition of the convergence indicator η, which enabled us to tune adaptivity along individual dimensions. Two versions of the adaptive algorithm were developed for this purpose, allowing to terminate refinement along specific directions when specified directional tolerances are met.

The validity of the construction was analyzed, including a Sobol-decomposition of the error between the constructed surrogate-based estimates and the true model. We analyzed both approaches in light of applications on a simple test problem, as well as detailed simulations of a shock-tube ignition problem involving 3 design variables and 22 uncertain rate parameters.

Computational tests indicated that the nested approach affords great flexibility in controlling the error in individual sets of parameters, first reducing the error of the inner expansion locally at each point of the outer variables sparse grid, and then reducing the error of the outer expansion and consequently the global representation. However, this flexibility, which is gained from the tensorized construction of the nested approach, comes at the expense of larger computational burden. In contrast, the adaptive pseudo-spectral projection refinement in the product-space approach enabled the construction of suitable surrogates at a smaller cost compared to the nested algorithm. The tests also reveal that directional tuning algorithm can provide an effective means of resolution control, affording appreciable savings over unconstrained refinement.

Both the nested and product-space approaches offer many avenues for further improvement. Specifically, the nested approach is not limited to performing pseudo-

spectral projections in both the inner and outer space. The incorporation of other methods for constructing surrogates, such as collocation (Section 2.7) or regression (Section 2.4), would potentially constitute avenues for improvement. In particular, the use of regularized solvers such as compressed sensing [32, 35] (Section 2.6) appear to be promising, namely for reducing the required number of model evaluations. The product-space approach with the decomposed error indicators also offers promising areas for further extensions. For instance, the tolerances used for tuning adaptivity could be dynamically adjusted, to guide refinement towards specific hyperplanes of the parameter space. Another possibility would be to incorporate cost functions into the sparse grid refinement algorithm so as to optimally exploit computational resources. Such extensions will be considered in future work.

5

Adaptive Smolyak Pseudo-Spectral Projections on Subsampled Sparse Grids

5.1 Introduction & Background

The adaptive Smolyak pseudo-spectral projection method [22] (aPSP) used in Chapters 3 and 4 (and [134] and [133] respectively) demonstrates drastic improvements in efficiency through global, dimensional adaptivity. However, they do not address local adaptivity leading to many additional realizations in the local regions of $\boldsymbol{\xi} \in \Xi$ where a lower-order projection may be sufficient. For example, the ocean problem investigated in [2, 134] and Chapter 3 exhibited some local regions of greater functional variation as illustrated in Fig. 3.3(d) where the dependence appears exponential for $\xi_4 \geqslant 0$ (stochastic wind-drag). The regions of $\xi_4 \leqslant 0$ are relatively flat and could be approximated with a lower-order projection.

One approach is to decompose the domain and represent each subdomain with its own, independent PCE [84]. Closely related is to use a wavelet-like expansion [77, 79, 80] where a hierarchical, multi-resolution basis of orthonormal polynomials (such as Alpert Wavelets [4]) are used to represent the solution locally.

The wavelet approach is particularly well suited to represent responses that exhibit multi-resolution behavior and can be adapted to particular regions of the stochastic space [76]. However, developing a wavelet expansion non-intrusively would require a prohibitively large number of function evaluations and such expansions are poorly suited for large, legacy models.

An alternative approach is a non PCE local surrogate such as spline interpolation [93, 85, 44, 97] where low-order polynomials are fit in local regions. The local functions are scaled and fit to create a global representation of the model and can handle discontinuities. Such an approach however must be undertaken at the start of the investigation, does not lend itself to the investigation of existing data sets, and creates an essentially non-global functional representation. Additionally, high-dimensional splines are particularly difficult to use and often require a hierarchical, tensor-product approach [74].

Instead of local interpolation, a class of stochastic collocation methods [76] can be used to generate a global polynomial interpolant using Lagrange basis functions [39] (see Section 2.7). Interpolation may be performed on sparse grids [95] to reduce the curse of dimensionality, though a structured grid is not required. Alternatively, the interpolating polynomial may be constructed from piecewise hierarchal interpolating polynomials that allow for discontinuities [88]. The resulting polynomial is exact at the collocation points (nodes) and is amenable to both global and local refinement. In Ref. [88], the hierarchical polynomials are used to achieve local adaptivity in the stochastic space by identifying regions (but not dimensions) to enrich the sample. *Jakeman and Roberts* [63] combine the local grid refinement of [88] with the global dimensional adaptivity of [46] (also used in this work) to incorporate a dimensional adaptivity into the local refinement.

We build on Ref. [63] to use spectral projection instead of stochastic collocation. We use the global dimensional aPSP methods from prior chapters to select appropri-

ate sparse grids and we use a local adaptation inspired by [88, 63] to choose which nodes may be skipped. We modify the local criteria to dynamically set the tolerance based on the current state of the adaptation. Unlike hierarchical stochastic collocation, the quadratures in NISP require data at the corresponding nodes leading us to use synthesized model values in regions identified as sufficiently converged and subsequently not evaluated. Essentially, the method we propose uses low-order surrogates to synthesize data in converged regions and model data in regions that are to be enriched.

Using lower-order surrogates to synthesize data introduces noise and error into the spectral projection. We therefore suggest two alternative, more robust methods to solve for the PCE coefficients. The first is compressed sensing [31, 127, 15] (CS), with methods similar to [35] for PCE reconstruction, also presented in Section 2.6. CS uses the assumption of sparsity on a proper basis to determine the coefficients using a dictionary much larger than the number of samples. The CS problem is either solved with the full-sparse grid (including synthesized data) or, with a subsampled grid. Using subsampled (though non-sparse) quadrature grids for PCEs via CS has been explored previously in [123].

The second alternative method we introduce is hybrid least-angle regression (Hybrid-LAR). LAR [36] (and Section 2.5) is a regression technique that successively enlarges the basis by adding terms (coefficients) most correlated with the residual of an active basis. We follow [9] in using a Hybrid-LAR approach wherein the LAR procedure provides an ensemble of potential bases and regression is performed on each candidate basis. The basis with the lowest cross-validation error estimate is chosen.

The layout of this chapter is as follows. In Section 5.2 we introduce the local refinement scheme, subsampled PSP and the use of CS and Hybrid-LAR for determining the PCE coefficients. In Section 5.3 we introduce two test functions used

to evaluate and validate the subsampled sparse methods. In addition to evaluating the performance of subsampled aPSP, CS and LAR, we also investigate how CS and Hybrid-LAR performance is dependent on the input basis and we compare performance against a LHS experiment. We then proceed to test these methods on the HYCOM database and combustion chemistry model in Section 5.4. Finally, in Section 5.5, we summarize our conclusions and discuss future improvements and extensions.

5.2 Subsampling and Alternative Projection Methods

The general procedure is broken into two main steps. The first step is to generate an appropriate subsampled grid and perform aPSP with synthesized data of omitted grid points. Use of synthesized data introduces inexactness of the underlying deterministic model and therefore may potentially degrade the fidelity of the resulting PCE. As such, the second step is to use an alternative, more robust solver for the final determination of coefficients. The *estimated* coefficients from the first step may be used to greatly accelerate the solution of more robust methods and aid in improving final performance. Alternatively, the PSP coefficient estimates may be used as the *final* PCE coefficients however caution must be used to ensure the validity of the representation.

There exists wide classes of alternative solvers for the PCE system such as regression, collocation, interpolation, etc. We will focus on solvers that make use of the underlying sparsity of the system such as hybrid least-angle-regression (see Section 2.5 and [9]) and compressed sensing (see Section 2.6 and [35]).

The general outline of the subsampling framework is presented in flow-chart form in Fig. 5.1. The primary objective of subsampled PSP is the first step, i.e.. determination of the subsampled sparse grid and (initial) estimation of the coefficients.

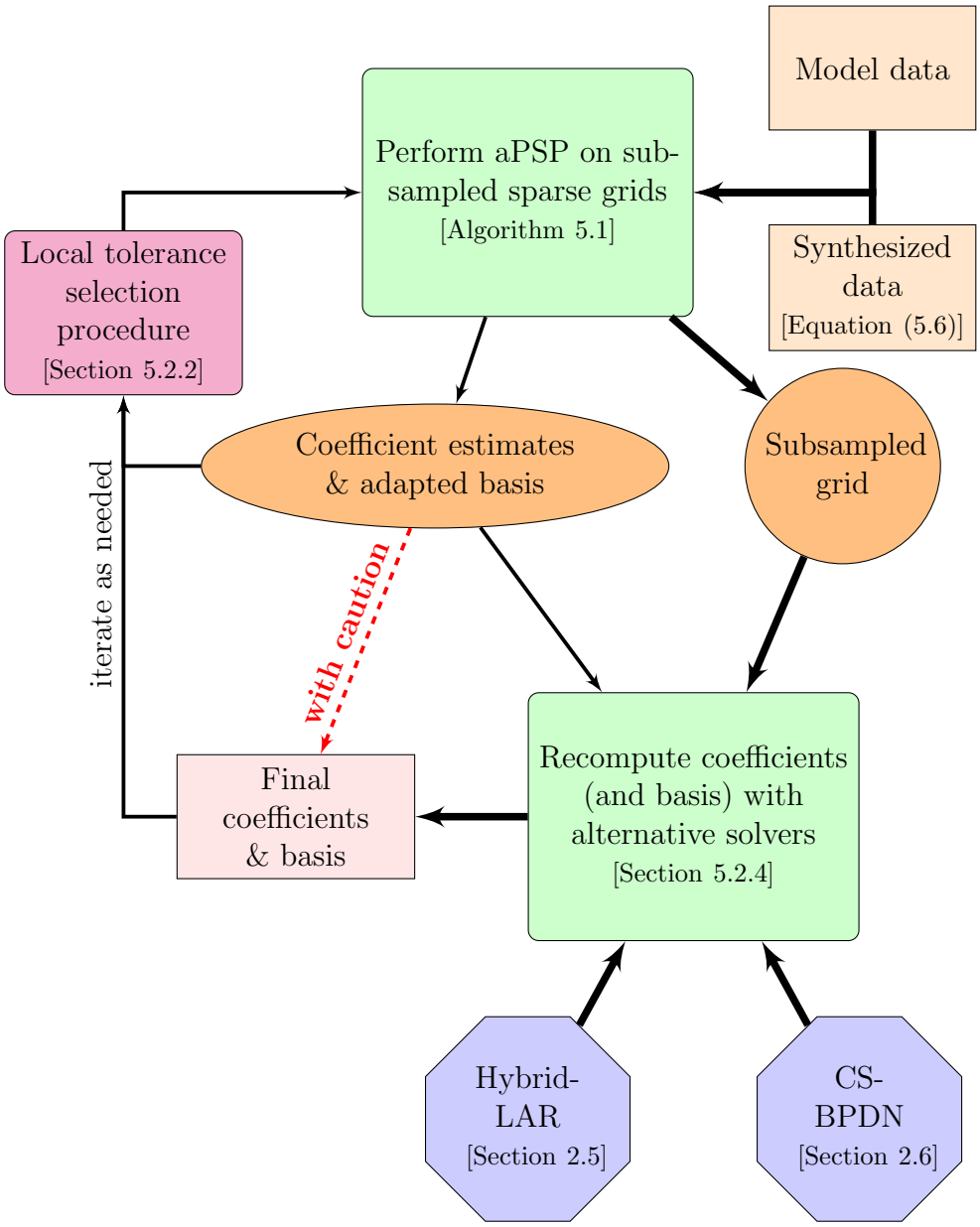


FIGURE 5.1: Flowchart of subsampling sparse grid framework. The primary objective to determine the appropriate subsampling sparse grid and estimate coefficients. The final grid (and potentially the synthesized model data) is used to determine an alternative PCE solution. The alternative PCE reconstruction may be greatly accelerated by the estimated polynomial coefficients. Noted in brackets are references to the corresponding section or equation.

The subsampled grid aPSP framework must first overcome two specific hurdles.

1. Determination of which quadrature nodes (realizations) may be skipped, and
2. Generation of synthesized model data at skipped quadrature nodes.

In the following sections, we will individually address each concern.

5.2.1 Subsampled Grid Determination

Recall the final PSP for a given $\mathscr{L}$ is in Equation (3.3) reproduced below:

$$\mathbf{P}_{\mathscr{L}} F = \sum_{\boldsymbol{l} \in \mathscr{L}} \boldsymbol{\Delta}_{\boldsymbol{l}}^{\mathrm{P}} F, \quad \boldsymbol{\Delta}_{\boldsymbol{l}}^{\mathrm{P}} F \doteq \left(\Delta_{l_1}^{\mathrm{P}} \otimes \cdots \otimes \Delta_{l_d}^{\mathrm{P}}\right) F, \quad \Delta_{l>1}^{\mathrm{P}} \doteq \mathrm{P}_{l}^{\mathtt{FT}} - \mathrm{P}_{l-1}^{\mathtt{FT}}.$$

Each $\mathrm{P}_{l}^{\mathtt{FT}}$ has the underlying *full-tensored* quadrature. The final projection grid, $\mathcal{G}(\mathscr{L})$ (Equation 2.11), is composed of the union of all full-tensor quadrature grids. Instead, we will now define

$$\mathcal{Y}^{\boldsymbol{l}} = \left\{ \left(\xi^{(q_1,l_1)} \cdots \xi^{(q_d,l_d)}\right), 1 \leqslant q_j \leqslant N(l_j), 1 \leqslant j \leqslant d \right\}, \quad N(\boldsymbol{l}) = |\mathcal{Y}^{\boldsymbol{l}}| \tag{5.1}$$

to be the full-tensor quadrature grid associated with the index $\boldsymbol{l} \in \mathscr{L}$. We note that for a full-tensor $\mathscr{L}$, $\mathcal{G}(\mathscr{L}_{\boldsymbol{l}}^{\mathtt{FT}}) \equiv \mathcal{Y}^{\boldsymbol{l}}$. In addition, each $\left(\xi^{(q_1,i_1)} \cdots \xi^{(q_d,i_d)}\right)$ has associated with it a full-tensor quadrature weight

$$w^{\boldsymbol{l},(i)} = \prod_{j=1}^{d} w^{(q_j,l_j)} \quad \forall \quad i = 1, ..., N(\boldsymbol{l}) \tag{5.2}$$

where $w^{(q_j,l_j)}$ is the weight associated with the level l_j 1D quadrature node $\xi^{(q_j,l_j)}$. We will now restrict ourselves to *nested quadratures* such that

$$\mathcal{Y}^{\boldsymbol{l}} \subseteq \mathcal{Y}^{\boldsymbol{l}'}, \quad l_i' \geqslant l_i, 1 \leqslant i \leqslant d$$

Essentially, the quadrature nodes of a lower-order rule are also present in a higher-order rule. As discussed in Section 3.3.1, nested quadratures are vital to the effectiveness of sparse grids.

We may also define a *unique* set of grid points due to each index $\boldsymbol{l}$ defined as follows:

$$\mathcal{X}^{\boldsymbol{l}} = \mathcal{Y}^{\boldsymbol{l}} \setminus \bigoplus_{n=1}^{d} \mathcal{Y}^{\boldsymbol{l}-\hat{\mathsf{e}}_n} \tag{5.3}$$

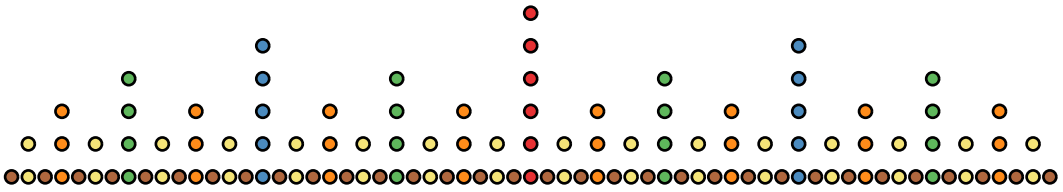
where $\hat{\mathsf{e}}_n$ is the unit vector in the n direction and $\mathcal{Y}^0 \doteq \varnothing$. The grids $\mathcal{X}^{\boldsymbol{l}}$ are unique in the sense that $\mathcal{X}^{\boldsymbol{l}} \bigcap \mathcal{X}^{\boldsymbol{l}'} = \varnothing$ for any $\boldsymbol{l} \neq \boldsymbol{l}'$.

Following the approach of [51] and [63], the nested grids have an innate hierarchical tree-structure. Each unique grid point $\boldsymbol{\xi}^{\boldsymbol{l},(i)} \in \mathcal{X}^{\boldsymbol{l}}, \quad \boldsymbol{l} \neq \{1, ..., 1\}$ has associated with it at least one and at most d parent grid points, where d is the number of stochastic dimensions. Let $p^j_{\boldsymbol{\xi}^{\boldsymbol{l},(i)}}$ represent the parent of $\boldsymbol{\xi}^{\boldsymbol{l},(i)} \in \mathcal{X}^{\boldsymbol{l}}$ in the j^{th} direction.

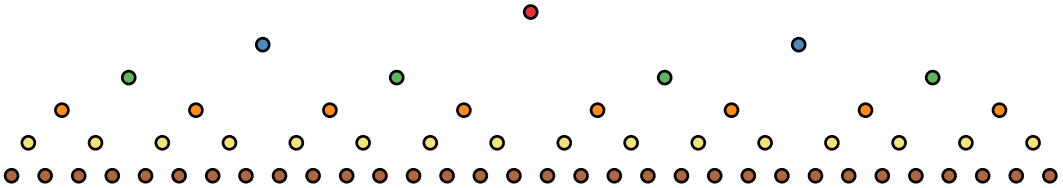
Such construction is demonstrated with an equidistant quadrature rule in Fig. 5.2 where $\mathcal{Y}$ is plotted for different quadrature levels in Fig. 5.2(a). Plotted in Fig. 5.2(b) is the unique grids $\mathcal{X}$ for each level and then in Fig. 5.2(c) with connections drawn to the parent. Note that this schematic demonstration is for the standard $2^l - 1$ scaling of an open, nested rule. A closed rule, (such as Clenshaw Curtis) has a slightly different structure and can be seen in [51].

The extension to multiple dimensions is readily available. Every unique grid point (other than the initial) has at least one and up to d parent(s). A schematic example is plotted in Fig. 5.3 where the $\mathscr{L}$ multi-index of quadrature levels is in Fig. 5.3(a) and the corresponding final grid is in Fig. 5.3(b). The grid points are all color coded to the corresponding index. In Figs. 5.3(c) & (d), the hierarchical, ancestral structure is show for the entire index and for each index to its respective parents. The construction readily extends to an arbitrary number of dimensions.

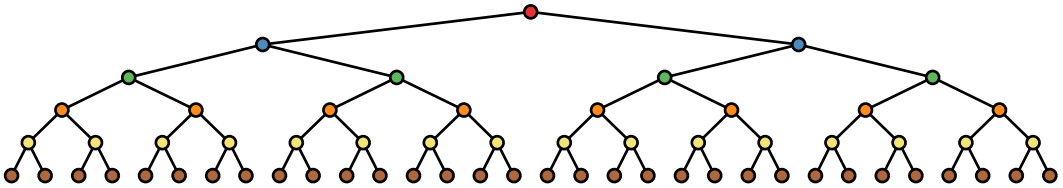
Finally, the hierarchical structure may be used to define local measures of convergence which may be used to define local refinement. Let $\lambda\left(\boldsymbol{\xi}^{\boldsymbol{l},(i)}\right)$ be a local error of $\boldsymbol{\xi}^{\boldsymbol{l},(i)}$ (defined later in Equation (5.8)) and let $\texttt{Tol}^k_\lambda$ be a local tolerance (defined later



(a) Full grid, $\mathcal{Y}^l$ for $l = 1, ..., 6$



(b) Unique grid, $\mathcal{X}^l$ for $l = 1, ..., 6$



(c) Unique grid with parental structure, $\mathcal{X}^l$ for $l = 1, ..., 6$

FIGURE 5.2: One dimensional schematic example of the hierarchical nature of nested grids demonstrated with a nested, equally-spaced open rule following an $N(l) = 2^l - 1$ scaling. In (a) is $\mathcal{Y}^l$ for $l = 1, ..., 6$ where each grid point is colored by the level at which it was added. In (b) is the unique grid $\mathcal{X}^l$ and finally in (c), the ancestral structure is added.

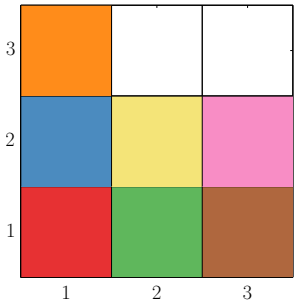
in Equation (5.9)). Let $\boldsymbol{l}^+$ be a candidate index for enrichment of $\mathscr{L}$ using the aPSP algorithm and let $\mathcal{X}^{\boldsymbol{l}^+}$ be the corresponding set of unique candidate grid points. Local subsampling is carried out by defining two subsets of $\mathcal{X}^{\boldsymbol{l}^+}$. Let $\mathscr{K}^{\boldsymbol{l}^+} \in \mathcal{X}^{\boldsymbol{l}^+}$ be the kept and evaluated set and let

$$\mathscr{S}^{\boldsymbol{l}^+} \doteq \mathcal{X}^{\boldsymbol{l}^+} \backslash \mathscr{K}^{\boldsymbol{l}^+}$$

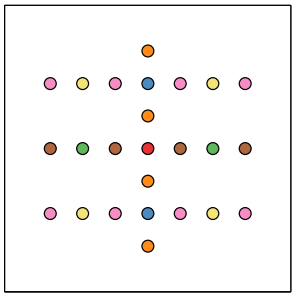
be the set of skipped nodes such that $\mathscr{K}^{\boldsymbol{l}^+} \bigcap \mathscr{S}^{\boldsymbol{l}^+} = \varnothing$. We define $\mathscr{K}^{\boldsymbol{l}^+}$ as follows:

$$\mathscr{K}^{\boldsymbol{l}^+} = \left\{ \boldsymbol{\xi}^{\boldsymbol{l}^+,(i)} \in \mathcal{X}^{\boldsymbol{l}} \middle| \; \exists j \in \{1, ..., d\} \; s.t. \; \lambda \left(p^j_{\boldsymbol{\xi}^{\boldsymbol{l},(i)}} \right) \geqslant \texttt{Tol}^k_\lambda \right\} \tag{5.4}$$

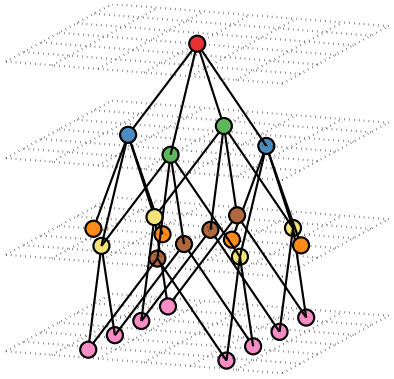
That is, $\mathscr{K}^{\boldsymbol{l}^+}$ is the set of all $\boldsymbol{\xi}^{\boldsymbol{l}^+,(i)} \in \mathcal{X}^{\boldsymbol{l}^+}$ where the candidate grid point has at



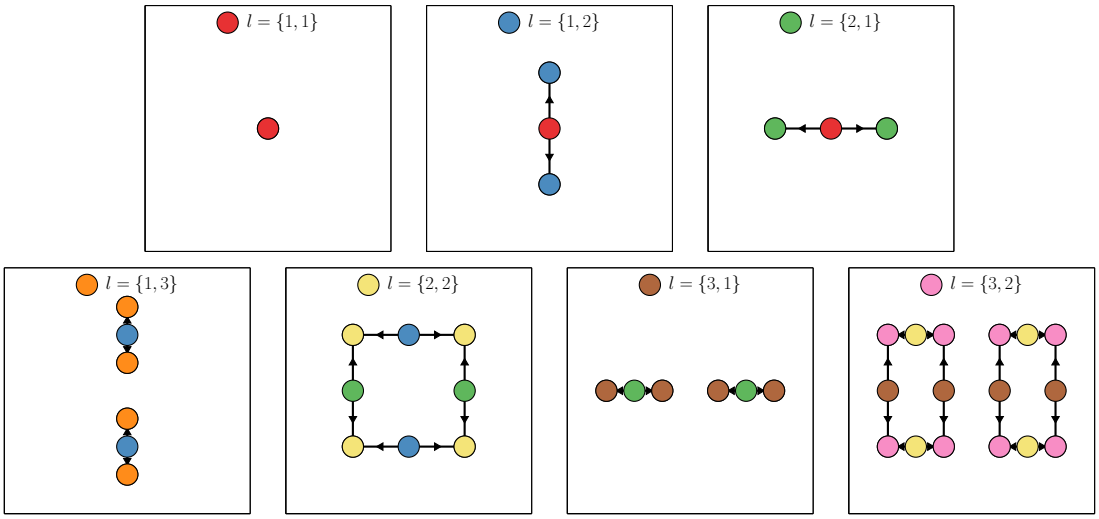
(a) Schematic $\mathscr{L}$



(b) Final grid colored by corresponding $\mathcal{X}^{\boldsymbol{l}}$



(c) 3D view of ancestry structure.



(d) 2D cuts showing ancestral structure.

FIGURE 5.3: Schematic 2D example of ancestral structure for an open, evenly-spaced grid. In (a) is the index-set of levels, $\mathscr{L}$ and in (b) is the final grid colored by $\mathcal{X}^{\boldsymbol{l}}$. In (c) and (d) are example showing the ancestry of each grid node. Notice for $\boldsymbol{l} = \{2, 2\}$ and $\boldsymbol{l} = \{3, 4\}$, each grid point has two parents.

least one parent $p^j_{\boldsymbol{\xi}^{\boldsymbol{l},(i)}}$ whose local error, $\lambda\left(p^j_{\boldsymbol{\xi}^{\boldsymbol{l},(i)}}\right)$ is greater or equal to $\texttt{Tol}^k_\lambda$. Recall $\mathcal{G}(\mathscr{L})$ is the final sparse-grid due to $\mathscr{L}$. In addion, we now define the *final* subsampled sparse grid $\mathcal{H}(\mathscr{L}) \subseteq \mathcal{G}(\mathscr{L})$ as

$$\mathcal{H}(\mathscr{L}) = \bigcup_{\boldsymbol{l}\in\mathscr{L}} \mathscr{K}^{\boldsymbol{l}}. \tag{5.5}$$

Since NISP and PSP require exact, global quadratures, synthesized model values must be used for $\boldsymbol{\xi}^{\boldsymbol{l},(i)} \in \mathscr{S}^{\boldsymbol{l}}$ and then the final projection may be performed on the combination of model and synthesized values. We also note that if $\mathcal{K}^{\boldsymbol{l}^+} = \varnothing$, then $\boldsymbol{l}^+$ is not added to $\mathcal{A}$, the active set in the aPSP algorithm. Therefore, the final $\mathscr{L}$ *may* be different from a non-subsampled adaptation.

Recall from Section 3.3.1 that $\mathbf{P}_{\mathscr{L}}(\boldsymbol{\xi})$ represents a Smolyak pseudo-spectral projection operator using the multi-index of levels $\mathscr{L}$. Furthermore, recall that $\mathbf{P}_{\mathscr{L}=\mathscr{L}^{\texttt{FT}}_{\boldsymbol{\ell}}} = \mathbf{P}_{\boldsymbol{\ell}}$ where

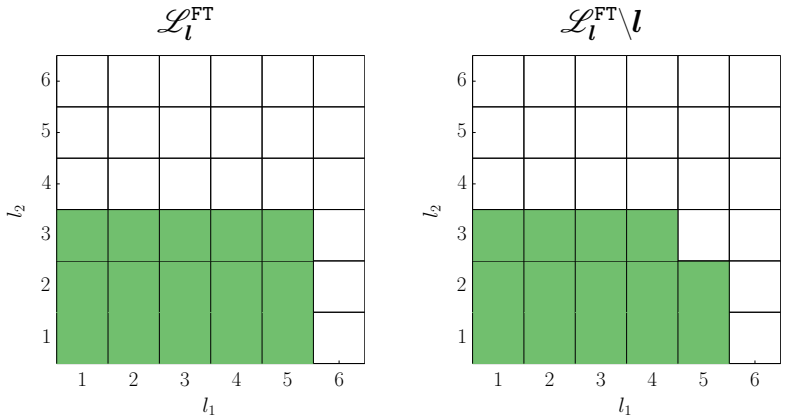
$$\mathscr{L}^{\texttt{FT}}_{\boldsymbol{\ell}} \doteq \{\boldsymbol{l} \in \mathbb{N}^d \big| l_i \leqslant \ell_i, 1 \leqslant i \leqslant d\},$$

is the full-tensor index. The projection $\mathbf{P}_{\boldsymbol{l}^+}$ requires model values for $\boldsymbol{\xi}^{\boldsymbol{l}^+,(i)} \in \mathcal{X}^{\boldsymbol{l}^+}$ and is therefore unavailable before evaluating $\boldsymbol{\xi}^{\boldsymbol{l}^+,(i)}$. However, a model estimate value $F'\left(\boldsymbol{\xi}^{\boldsymbol{l}^+,(i)}\right)$ may be synthesized with

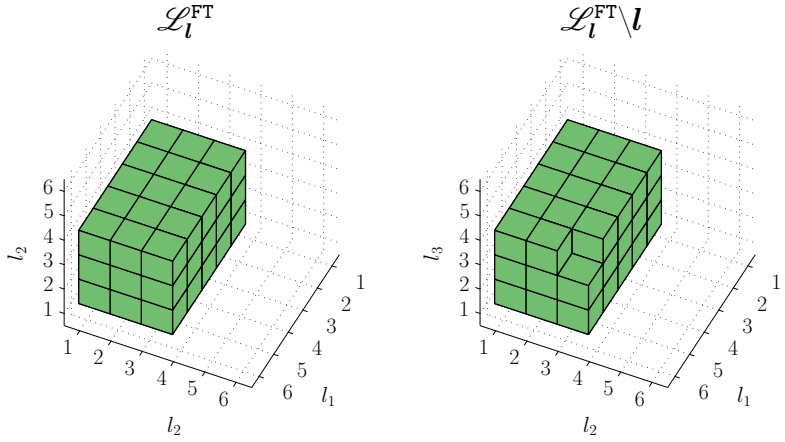
$$F'\left(\boldsymbol{\xi}^{\boldsymbol{l}^+,(i)}\right) = \mathbf{P}_{\mathscr{L}=\mathscr{L}^{\texttt{FT}}_{\boldsymbol{l}^+}\backslash\boldsymbol{l}^+}(\boldsymbol{\xi}^{\boldsymbol{l}^+,(i)}), \quad \boldsymbol{l}^+ \neq \{1,...,1\} \tag{5.6}$$

without using any of the $\boldsymbol{\xi}^{\boldsymbol{l}^+,(i)} \in \mathcal{X}^{\boldsymbol{l}^+}$ model data. $F'\left(\boldsymbol{\xi}^{\boldsymbol{l}^+,(i)}\right)$ represents the "best" estimate for the model values in $\mathcal{X}^{\boldsymbol{l}^+}$ before including those nodes. Schematic examples of $\mathscr{L}^{\texttt{FT}}_{\boldsymbol{\ell}}$ and the associated $\mathscr{L}^{\texttt{FT}}_{\boldsymbol{\ell}}\backslash\boldsymbol{l}$ are presented in Fig. 5.4.

We also define $F'\left(\boldsymbol{\xi}^{\boldsymbol{l},(i)}\right) = \infty$ for $\boldsymbol{l} = \{1,...,1\}$ since there is no parent index and to produce the desired properties later in Equation (5.8). Finally, we are now able to define our subsampled evaluations using a mixture of synthesized (F') and actual



(a) 2D schematic example for $\boldsymbol{l} = \{5, 3\}$



(b) 3D schematic example for $\boldsymbol{l} = \{5, 3, 3\}$

FIGURE 5.4: Schematic example of a full-tensor index $\mathscr{L}_{\boldsymbol{\ell}}^{\mathtt{FT}}$ and the parent index $\mathscr{L}_{\boldsymbol{\ell}}^{\mathtt{FT}} \backslash \boldsymbol{l}$. The projection with $\mathscr{L}_{\boldsymbol{\ell}}^{\mathtt{FT}} \backslash \boldsymbol{l}$ does not require function evaluations for $\boldsymbol{\xi}^{\boldsymbol{l},(i)} \in \mathcal{Y}^{\boldsymbol{l}}$ and may therefore be used as synthesized surrogate values.

(F) model data

$$\widetilde{F}\left(\boldsymbol{\xi}^{\boldsymbol{l},(i)}\right) = \begin{cases} F\left(\boldsymbol{\xi}^{\boldsymbol{l},(i)}\right) & \boldsymbol{\xi}^{\boldsymbol{l},(i)} \in \mathscr{K}^{\boldsymbol{l}} \\ F'\left(\boldsymbol{\xi}^{\boldsymbol{l},(i)}\right) & \boldsymbol{\xi}^{\boldsymbol{l},(i)} \in \mathscr{S}^{\boldsymbol{l}} \end{cases} \tag{5.7}$$

Therefore, the subsampled full-tensor PSP operator from Equation (2.7) may be redefined to

$$c_k = \mathbf{Q}\left(\widetilde{F}\Psi_k\right) = \sum_i \widetilde{F}(\boldsymbol{\xi}^{(i)})\Psi_k(\boldsymbol{\xi}^{(i)})w^{(i)}$$

If $\mathscr{S}^{\boldsymbol{l}} = \varnothing \ \forall \ \boldsymbol{l} \in \mathscr{L}$, the original full-tensor, non-subsampled coefficients are recovered.

We may also now define an error of each quadrature node as follows

$$\lambda\left(\boldsymbol{\xi}^{\boldsymbol{l},(i)}\right) = \left|F'\left(\boldsymbol{\xi}^{\boldsymbol{l},(i)}\right) - \widetilde{F}\left(\boldsymbol{\xi}^{\boldsymbol{l},(i)}\right)\right| w^{\boldsymbol{l},(i)} \tag{5.8}$$

where $w^{\boldsymbol{l},(i)}$ is from Equation (5.2), the corresponding full-tensor quadrature weight. The use of $w^{\boldsymbol{l},(i)}$ is suggested by [63] and serves to properly weight the importance of each node to the respective quadrature rule. Alternatively, it may be thought of as if $\lambda\left(\boldsymbol{\xi}^{\boldsymbol{l},(i)}\right)$ were known for all $\boldsymbol{\xi}^{\boldsymbol{l},(i)} \in \mathcal{Y}^{\boldsymbol{l}}$ (*not* $\mathcal{X}^{\boldsymbol{l}}$), and if $\mathscr{S}^{\boldsymbol{\ell}} = \varnothing \ \forall \ \boldsymbol{\ell} \in \mathscr{L}_{\boldsymbol{l}}^{\mathtt{FT}}$ then $\sum_{i \in \mathcal{Y}^{\boldsymbol{l}}} \lambda\left(\boldsymbol{\xi}^{\boldsymbol{l},(i)}\right)$ would be the L_1 error between the model and the PCE value.

Equation 5.8 also has the following additional properties:

- If $\boldsymbol{l} = \{1, ..., 1\}$, $\lambda\left(\boldsymbol{\xi}^{\boldsymbol{l},(i)}\right) \to \infty > \mathtt{Tol}_\lambda$ and as such will always be adapted, and
- If $\boldsymbol{\xi}^{\boldsymbol{l},(i)} \in \mathscr{S}^{\boldsymbol{l}}$, $\widetilde{F}\left(\boldsymbol{\xi}^{\boldsymbol{l},(i)}\right) = F'\left(\boldsymbol{\xi}^{\boldsymbol{l},(i)}\right)$ and $\lambda\left(\boldsymbol{\xi}^{\boldsymbol{l},(i)}\right) = 0 \leqslant \mathtt{Tol}_\lambda$ and will therefore never admit an axial child itself.[1]

In [63], the equivalent of the local tolerance was set at the beginning of the adaptation and was used as part of the termination estimation. In this work, we rely on the aPSP algorithm's inherent termination criterial on $\eta^2 < \mathtt{Tol}_\eta^2$ and therefore set the local tolerance dynamically.

The local tolerance is computed at *the beginning* of the adaptation (before a critical index, $\boldsymbol{l}^*$ is determined) and is based on the state of the entire projection index, $\mathscr{L}^k$ at the k^{th} iteration.

The local tolerance is set as

$$\mathtt{Tol}_\lambda^k = \alpha \max\left[\lambda\left(\boldsymbol{\xi} \in \mathcal{X}^{\mathcal{C}}\right)\right], \qquad 0 \leqslant \alpha \leqslant 1 \tag{5.9}$$

[1] An axial child may still be added if another axial parent has a local error greater than the tolerance.

where $\mathcal{C} \subseteq \mathscr{L}$ is a subset of $\mathscr{L}$ that represents all potential parent indices and $\mathcal{X}^{\mathcal{C}} \doteq \bigcup_{\boldsymbol{l}\in\mathcal{C}} \mathcal{X}^{\boldsymbol{l}}$. In order to define $\mathcal{C}$, we must first define $\mathcal{A}^+$, the collection of all $\mathcal{O}$-*admissible* forward neighbors of $\mathcal{A}$. Recall that we require $\mathcal{O}$-admissibility for forward neighbors however, we must account for $\boldsymbol{l}^*$ not yet being moved to $\mathcal{O}$.

$$\mathcal{A}^+ = \bigcup_{\boldsymbol{l}\in\mathcal{A}} \bigcup_{n=1}^{d} \widehat{\boldsymbol{l}}_n, \qquad \widehat{\boldsymbol{l}}_n = \begin{cases} \boldsymbol{l} + \hat{\mathsf{e}}_n & \text{if } \exists m \in \{1,...,d\} \ s.t. \ \boldsymbol{l} + \hat{\mathsf{e}}_n - \hat{\mathsf{e}}_m \in \mathcal{O} \cup \boldsymbol{l} \\ \varnothing & \text{otherwise} \end{cases} \tag{5.10}$$

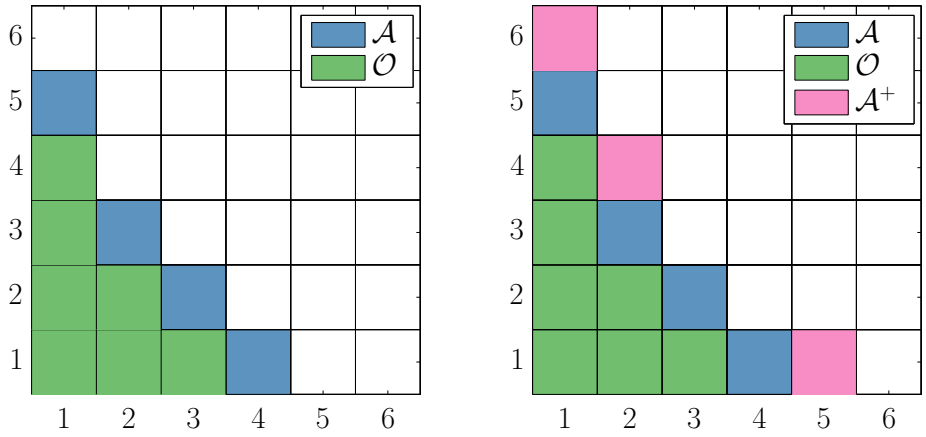
We note that $\mathcal{A}^+ \notin \mathscr{L} = \mathcal{A} \cup \mathcal{O}$. A schematic example of $\mathcal{A}^+$ is shown in Fig. 5.5 where Fig. 5.5(a) is a schematic example index and Fig. 5.5(b) is the index with $\mathcal{A}^+$ also plotted. $\mathcal{A}^+$ is only those indices that could potentially be present in the next iteration. We can now define $\mathcal{C}$ as the union of the backwards neighbors of $\mathcal{A}^+$

$$\mathcal{C} = \bigcup_{\boldsymbol{l}\in\mathcal{A}^+} \bigcup_{\substack{m=1 \\ l_m>1}}^{d} \boldsymbol{l} - \hat{\mathsf{e}}_m \tag{5.11}$$

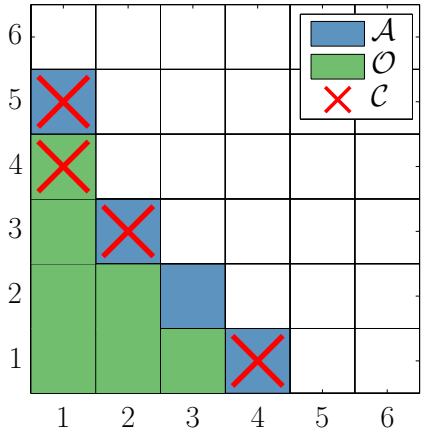
The resulting $\mathcal{C}$ index for the schematic example in Fig. 5.5(a) is plotted in Fig. 5.5(c). Notice that $\mathcal{C}$ includes terms both in $\mathcal{A}$ and $\mathcal{O}$ and there exists terms in $\mathcal{A} \notin \mathcal{C}$.

The final subsampled aPSP algorithm is presented in Algorithm 5.1. We note that the final algorithm is very similar to that of Algorithm 3.1 except $\widetilde{F}\left(\boldsymbol{\xi}^{\boldsymbol{l},(i)}\right)$ is used in place of $F\left(\boldsymbol{\xi}^{\boldsymbol{l},(i)}\right)$ and an index is not admitted if $\mathscr{K}^{\boldsymbol{l}^+} = \varnothing$.

The end result of the subsampled aPSP algorithm can be thought of as using lower-order surrogates to synthesize data for higher-order surrogates *in regions estimated to be converged*. The final coefficients and $\mathcal{K}$ may be used as the final projection coefficients, however care must be taken. The α selection procedure (see below) should inform the validity of the results (if non-converged regions are inappropriately synthesized, the lower α of the next iteration should make that clear) but alternative PCE solution methods may prove more robust in the face of errors introduced from synthesized data.



(a) Initial sample index with $\mathcal{A}$ and $\mathcal{O}$ (b) Showing $\mathcal{A}^+$



(c) Index with $\mathcal{C}$ noted with a red X

FIGURE 5.5: Schematic example of an index (a) with $\mathcal{A}^+$ plotted (b) and then showing the final $\mathcal{C}$ subset for determining $\texttt{Tol}_\lambda$ (c).

5.2.2 α Selection Criteria

A choice of $\alpha = 0$ eliminates subsampling and returns the aPSP result. Selection of α is a non-trivial task and is also problem dependent based on the specific behavior and dimensionality of the model. Numerical tests suggest that, for a reasonably low α, if $\alpha_2 < \alpha_1$, the final grid $\mathcal{G}^{\alpha_1} \subseteq \mathcal{G}^{\alpha_2}$. However, this result is also problem dependent (as we will see for one of our test functions where this isn't strictly the case).

Algorithm 5.1 Subsampled aPSP.

$\mathcal{A} \doteq \{1, \ldots, 1\}$
$\mathcal{O} \doteq \varnothing$
while $\eta^2 \doteq \sum_{\boldsymbol{l} \in \mathcal{A}} \epsilon(\boldsymbol{l})^2 \geqslant \texttt{Tol}_\eta^2$ **do**
  $\texttt{Tol}_\lambda^k = \alpha \max\left[\lambda\left(\boldsymbol{\xi} \in \mathcal{X}^{\mathcal{C}}\right)\right]$ ▷ (5.9), $\texttt{Tol}_\lambda$ evaluated *before* enrichment
  Select $\boldsymbol{l}^* \in \mathcal{A}$ with highest $\epsilon(\boldsymbol{l})$
  $\mathcal{O} \leftarrow \mathcal{O} \cup \boldsymbol{l}^*$
  $\mathcal{A} \leftarrow \mathcal{A} \backslash \boldsymbol{l}^*$
  for $n = 1, \ldots, d$ **do**
    $\boldsymbol{l}^+ = \boldsymbol{l}^* + \hat{\mathsf{e}}_n$ ▷ Candidate Index
    if $\boldsymbol{l}^+ - \hat{\mathsf{e}}_k \in \mathcal{O} \ \forall \ k \in \left[1, \ldots, d, l_k^+ > 1\right]$ **then** ▷ Admissibility
      $\mathscr{K}^{\boldsymbol{l}^+} = \left\{\boldsymbol{\xi}^{\boldsymbol{l},(i)} \in \mathcal{X}^{\boldsymbol{l}^+} \middle| \ \exists j \in \{1, ..., d\} \ s.t. \ \lambda\left(p^j_{\boldsymbol{\xi}^{\boldsymbol{l},(i)}}\right) \geqslant \texttt{Tol}_\lambda^k\right\}$ ▷ (5.4)
      $\mathscr{S}^{\boldsymbol{l}^+} = \mathcal{X}^{\boldsymbol{l}^+} \backslash \mathscr{K}^{\boldsymbol{l}^+}$
      if $\mathscr{K}^{\boldsymbol{l}^+} \neq \varnothing$ **then** ▷ only add w/ evaluated points
        $\mathcal{A} \leftarrow \mathcal{A} \cup \boldsymbol{l}^+$
        Evaluate $F'\left(\boldsymbol{\xi}^{\boldsymbol{l}^+,(i)}\right)$ ▷ (5.6)
        Evaluate $\widetilde{F}\left(\boldsymbol{\xi}^{\boldsymbol{l},(i)}\right)$ & $\lambda\left(\boldsymbol{\xi}^{\boldsymbol{l},(i)}\right)$ ▷ (5.7) & (5.8)
        Project $\mathbf{P}_{\boldsymbol{l}^+}$ with $\widetilde{F}$ ▷ Full Tensor projection
        Calculate $\boldsymbol{\Delta}^{\mathbf{P}}_{\boldsymbol{l}^+}$
        $\epsilon(\boldsymbol{l}^+) = (1/\beta)\|\boldsymbol{\Delta}^{\mathbf{P}}_{\boldsymbol{l}^+}\|$
      end if
    end if
  end for
end while
Evaluate $\mathbf{P}_{\mathscr{L}} = \sum_{\boldsymbol{l} \in \mathscr{L}} \boldsymbol{\Delta}^{\mathbf{P}}_{\boldsymbol{l}}$

We recommend an iterative procedure to determine α. We suggest beginning with a larger α value (e.g. $\alpha \approx 0.1$) and choosing very small steps to decrease α. At each step, we recalculate and observe the following quantities listed below. Let $\mathcal{K}_\alpha$, $\mathscr{L}_\alpha$ and $c_{\boldsymbol{k}_\alpha}$ be the PCE multi-index, quadrature index, and coefficients (respectively) resulting from α. Let α_n be the α value for the n^{th} adaptation and let $\alpha_{n'}$ be the lowest α tested.

For every α value, we recalculate the following with $\alpha_{n'}$ and use these metrics to select an optimal α. All of these values are presented as an absolute value but may need to be appropriatly normalized. The quantities used to select an α are as follows:

- L_2 convergence *between PCEs* for different α values:

$$\sqrt{\sum_{\boldsymbol{k}\in\mathcal{K}_{\alpha_n}\cup\mathcal{K}_{\alpha_{n'}}}\left(c_{\boldsymbol{k}_{\alpha_n}}-c_{\boldsymbol{k}_{\alpha_{n'}}}\right)^2} \quad \forall\ n=1,...,n'-1 \tag{5.12}$$

  where $c_{\boldsymbol{k}}=0\ \forall\ \boldsymbol{k}\ \notin\ \mathcal{K}$.

- Maximum point-wise error *between PCEs* using largest grid. This is particularly useful as it compares errors at grid points not previously evaluated for $\alpha_{n\neq n'}$. Also useful is to examine if the maximum error occurs at $\boldsymbol{\xi}\notin\mathcal{G}^{\alpha_{n'}}$. Compare $\forall\ n=1,...,n'-1$

$$\max_{\boldsymbol{\xi}\in\mathcal{H}(\mathscr{L}_{\alpha_{n'}})}\left|\sum_{\boldsymbol{k}\in\mathcal{K}_{\alpha_n}}c_{\boldsymbol{k}_{\alpha_n}}\Psi_{\boldsymbol{k}}(\boldsymbol{\xi})-\sum_{\boldsymbol{k}'\in\mathcal{K}_{\alpha_{n'}}}c_{\boldsymbol{k}'_{\alpha_{n'}}}\Psi_{\boldsymbol{k}'}(\boldsymbol{\xi})\right| \tag{5.13}$$

- Change in the size of the final index: $|\mathcal{K}_{\alpha_n}|$ vs. $|\mathcal{K}_{\alpha_{n'}}|\quad\forall\ n=1,...,n'-1$, and which α values critically changed the size. Or change in the size of $\mathscr{L}$ since $\mathscr{L}$ prescribes $\mathcal{K}$.

- Number of evaluated grid points for decreasing α. Compare both for common indices:

$$\sum_{\boldsymbol{l}\in\mathscr{L}_{\alpha_n}\bigcap\mathscr{L}_{\alpha_{n'}}}\left|\mathscr{K}^{\boldsymbol{l}}_{\alpha_n}\right| \quad \text{vs.} \quad \sum_{\boldsymbol{l}\in\mathscr{L}_{\alpha_n}\bigcap\mathscr{L}_{\alpha_{n'}}}\left|\mathscr{K}^{\boldsymbol{l}}_{\alpha_{n'}}\right|,$$

  and for the total number of evaluated points:

$$|\mathcal{H}(\mathscr{L}_{\alpha_n})|\doteq\sum_{\boldsymbol{l}\in\mathscr{L}_{\alpha n}}\left|\mathscr{K}^{\boldsymbol{l}}_{\alpha_n}\right| \quad \text{vs.} \quad \left|\mathcal{H}(\mathscr{L}_{\alpha_{n'}})\right|\doteq\sum_{\boldsymbol{l}\in\mathscr{L}_{\alpha_{n'}}}\left|\mathscr{K}^{\boldsymbol{l}}_{\alpha_{n'}}\right|$$

  $\forall\ n=1,...,n'-1$.

- Empirical point-wise error on the sampled quadrature grid, $\mathcal{H}(\mathscr{L}_{\alpha_n})$:

$$\sqrt{\sum_{\boldsymbol{\xi}\in\mathcal{H}(\mathscr{L}_{\alpha n})}\left(\sum_{\boldsymbol{k}\in\mathcal{K}}c_{\boldsymbol{k}_{\alpha_n}}\Psi_{\boldsymbol{k}}(\boldsymbol{\xi})-F(\boldsymbol{\xi})\right)^2} \tag{5.14}$$

We note that Equation (5.14) is the point-wise error between the PCE and the model values while Equation (5.12) examines the global difference between the PCEs.

- Saturation of estimated error with cross-validation. The CV error does not need to be recalculated at each new adaptation.
  - Non-exhaustive random-subsampling CV error for BPDN (Algorithm 2.2). Note that this value is non-deterministic.
  - Leave-one-out error for Hybrid-LAR (Algorithm 2.1).

At each iteration, we recalculate these quantities (except the CV and empirical error) for all previous iterations. In particular, we focus on the last few α_n values to examine their convergence with respect to $\alpha_{n'}$, the smallest α. If the values do not change when compared to a more enriched PCE (and realization ensemble), we have reached an optimal α. We again note that as $\alpha \to 0$, the final grid approaches that of a non-subsampled grid.

5.2.3 Alternative $\texttt{Tol}_\lambda$ Selection Strategies

In this work, we focused on the α-based $\texttt{Tol}_\lambda$ selection criteria but we briefly want to mention other techniques. Perhaps the most straight-forward is to set $\texttt{Tol}_\lambda$ once and for all at the onset of the subsampled aPSP. Such a strategy is simple but also may cause the adaptation to terminate too soon.

Another strategy is to set the tolerance again with an α-like scale but instead of choosing the maximum from $\mathcal{C}$ (5.11), the maximum could be selected from the parent indices of that particular index $\boldsymbol{l}$. Recall the parents of $\boldsymbol{l}$ are $\bigcup_{n=1,l_n>1}^{d} \boldsymbol{l} - \hat{\mathsf{e}}_n$. This approach has the advantage that it guarantees $\mathscr{K}^{\boldsymbol{l}} \neq \varnothing$ and therefore, the final $\mathscr{L}$ for $\alpha > 0$ will more closely resemble that of $\alpha = 0$. Numerical tests consistently

suggests that such an approach sets $\texttt{Tol}_\lambda$ to unrealistically low values and is not sufficiently aggressive.

These approaches may be implemented for specific problems in future work.

5.2.4 Alternative Solution Strategies

The coefficients estimated with Algorithm 5.1 may be used as the final coefficients in the expansions, however, the use of synthesized data for $\alpha > 0$ introduces error into the projection. Even though by construction of the adaptive algorithm, we expect the synthesized value to be close to the actual model data. Furthermore the iterative procedure to identify α should help identify important new data points where the previous higher α adaptation skipped. Regardless, we present two, more robust methods to determine the PCE coefficients; compressed sensing (CS) and hybrid least-angle-regression (LAR).

In a purely CS or Hybrid-LAR approach, there is no built-in method to determine an optimal $\mathcal{K}$. Basis selection for LAR and compressed sensing is non-trivial and is an area of active research [64, 53, 9]. However, for recomputing coefficients from the subsampled aPSP algorithm, we can use the estimates of those coefficients and $\mathcal{K}$ from PSP to inform the basis selection for CS and LAR. For CS, it is particularly important to select a basis based on the sparsity of the solution. We can estimate the sparsity of the PCE with a threshold. For example,

$$\mathcal{K}^{S,\epsilon} = \left\{ \boldsymbol{k} \in \mathcal{K} \,\middle|\, \frac{c_{\boldsymbol{k}}^2}{\sum_{\boldsymbol{k}' \in \mathcal{K} \setminus \mathcal{K}_\varnothing} c_{\boldsymbol{k}'}^2} \leqslant \epsilon \right\} \tag{5.15}$$

where $\frac{c_{\boldsymbol{k}}^2}{\sum_{\boldsymbol{k}' \in \mathcal{K} \setminus \mathcal{K}_\varnothing} c_{\boldsymbol{k}'}^2}$ represents each coefficient's contribution to the variance of the problem. The sparsity can then be *estimated* as $|\mathcal{K}^{S,\epsilon}|$.

Based on the sparsity of the basis and the $\mathcal{K}$ prescribed by aPSP, we can enrich $\mathcal{K}$ to be sufficiently large and meet BPDN convergence requirements (see Section 2.6).

Let $\mathcal{K}^{\boldsymbol{p},q}$ from Equation (C.3) be the hyperbolic-cross multi-index staring at zero up the order $\boldsymbol{p}$ and with hyperbolic tuning paramater q. Let $\boldsymbol{p}_{\max}(\mathcal{K})$ be such that $\boldsymbol{p}_{\max,i}(\mathcal{K})$ is the highest order of index $\mathcal{K}$ in the i^{th} direction. For a given sizing factor s, the basis is enriched as follows:

$$\widetilde{\mathcal{K}}_{s,q}(\mathcal{K}) = \underbrace{\mathcal{K}^{\boldsymbol{p}=\lceil s\cdot \boldsymbol{p}_{\max}(\mathcal{K})\rceil,q}}_{\substack{\text{Hyperbolic Cross}\\ \text{with } q \text{ up to highest}\\ \text{scaled order}}} \bigcup \underbrace{\left[\bigcup_{\substack{\boldsymbol{l}\in\mathcal{K}\\ i\in\{1,\dots,d\}}} \lceil s\cdot \mathcal{K}_{l_i} \rceil \right]}_{\text{Scaled basis of } \mathcal{K}} \tag{5.16}$$

Where $\lceil\cdot\rceil$ is the ceiling operator. Despite being an active area of research, we found very little sensitivity to the input CS basis provided it was sufficiently large. This will be investigated in Section 5.3.3.

The CS approach does not explicitly require the use of the synthesized data. Using the data will ensure better coverage of the stochastic space but also introduces errors due to the inherent inexactness of the synthesized data. As such, it is up to the user to decide whether or not to use this data. Prior work in [123] investigated subsampling a full-tensor quadrature with CS and did not use any surrogate data in the final computation.

The Hybrid-LAR coefficient estimates are sensitive to the size of the input LAR basis, $\mathcal{K}^{\mathcal{I}}$. In [9], the authors suggested different basis scaling algorithms and were not restricted by $(P+1) \leqslant N$ since the LAR basis selection procedure will terminate when the *active* basis, $|\mathcal{K}^{\mathcal{A}}| = \min(P, N-1)$. However, those methods were developed for a non-structured grid (eg, LHS). When applied to a structured (and colinear) grid, the $\boldsymbol{\mathcal{Z}}$ matrix becomes singular for $(P+1) > N$ and the LAR procedure becomes unstable.

We investigated using a scaling procedure like that of [9] in Section 5.3.3 (restricted to $(P+1) \leqslant N$) but consistently found it unnecessary. The structured nature of the grid actually prescribes the best input basis. Numerical tests suggest

the best *input* basis for Hybrid-LAR is $\mathcal{K}^{\mathcal{I}} = \mathcal{K}^{\mathcal{L}}$ (2.26); the interpolating basis that may be factored into a Lagrange interpolant basis. However, the active basis $\mathcal{K}^{\mathcal{A}} \subset \mathcal{K}^{\mathcal{I}} \equiv \mathcal{K}^{\mathcal{L}}$, and therefore the final regression step (after LAR basis selection), does *not* solve the interpolation problem.

In order for the information matrix $\boldsymbol{\mathcal{Z}}$ (2.16) for Hybrid-LAR to be well-posed, we may not omit $\boldsymbol{\xi} \in \mathscr{S}^{\boldsymbol{l}} \; \forall \; \boldsymbol{l} \in \mathscr{L}$. We therefore determine the LAR coefficients on the full sparse grid $\mathcal{G}(\mathscr{L})$ with $\widetilde{F}$ (5.7); the combination of model and synthesized data.

5.3 Subsampling on Model Test Problems

In order to examine and assess the performance of the algorithm presented in Section 5.2, we again made use of model test problems. We used the test problem from Section 4.12 reproduced below and plotted in Fig. 4.3(a).

$$\text{[TF1:]} \quad F(\boldsymbol{\xi}) = \left(1 + \frac{1/3}{2\xi_1 + \xi_2 + 7/2}\right) \exp\left[-\left(\frac{1}{2}\left(\xi_2 - \frac{1}{5}\right)(\xi_2 + 1)\right)^2\right] \qquad (5.17)$$

$$\boldsymbol{\xi} \in [-1, 1]^2.$$

In addition, we defined a second, higher-dimensional test problem inspired by the Genz functions [45]. We choose a shifted and scaled multi-dimensional Gaussian of the following form:

$$\text{[TF2:]} \quad F(\boldsymbol{\xi}) = \exp\left[-\left(\boldsymbol{c_0}^2 \cdot (\boldsymbol{\xi} - \boldsymbol{\xi}_0)^2\right)\right], \quad \boldsymbol{\xi} \in [-1, 1]^5 \qquad (5.18)$$

$$\boldsymbol{c_0} = [0.10, 0.20, 0.75, 1.0, 1.5]$$

$$\boldsymbol{\xi}_0 = [-1.0, -0.95, 0, 0.95, 1].$$

TF2 is a smooth function that is designed to have larger sensitivities to specific dimensions; exhibit mixed-term sensitivities; and have regions of higher variation focused both at the center and the edges of the domain. The full-decomposition of the sensitivity is depicted in Fig. 5.6.

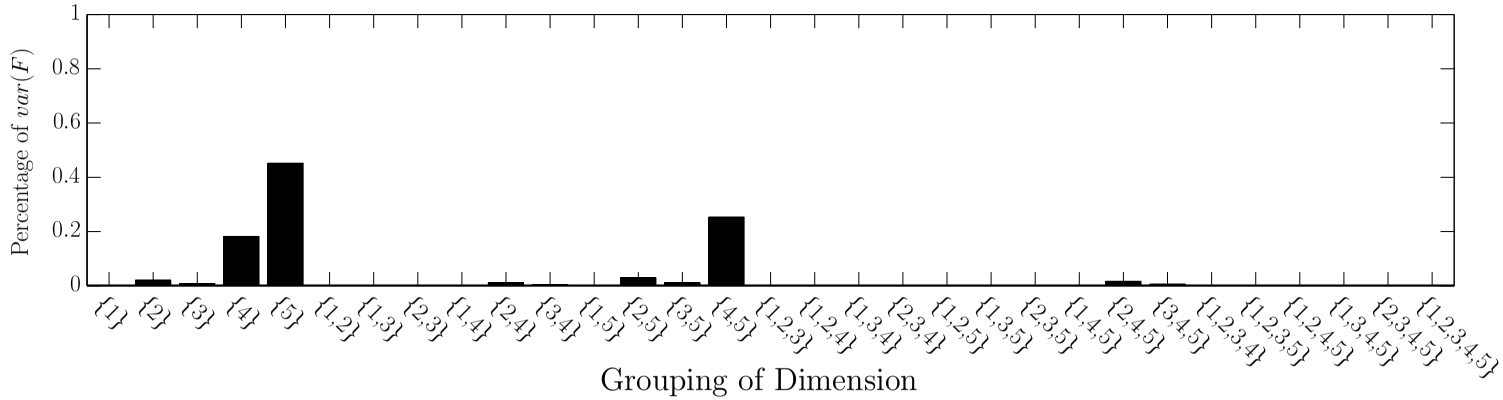


FIGURE 5.6: Sensitivity (contribution to variance) of all groupings of dimensions for TF2 (5.18). Substantial sensitivity is due to directions 4, 5 and their mixed-interaction.

5.3.1 Primary Results for Selected α Values

We performed the subsampled aPSP on various choices of α (including $\alpha = 0$, i.e. no subsampling) and then recomputed the coefficients based on the final grid as discussed in Section 5.2. For the Hybrid-LAR computation, we used the full sparse-grid with synthesized and model data and for the CS computation, we followed [123] and just used the model data.

For TF1, we computed the model error using an $N = 200^2$-point Gauss-Legendre quadrature rule. The relative error, ζ, defined according to

$$\zeta^2 = \frac{\|F - \sum_{\boldsymbol{k}\in\mathcal{K}} c_{\boldsymbol{k}}\Psi_{\boldsymbol{k}}\|}{\|F\|} \approx \frac{\sum_{i=1}^{N}\left[F\left(\boldsymbol{\xi}^{(i)}\right) - \sum_{\boldsymbol{k}\in\mathcal{K}} c_{\boldsymbol{k}}\Psi_{\boldsymbol{k}}\left(\boldsymbol{\xi}^{(i)}\right)\right]^2 w^{(i)}}{\sum_{i=1}^{N} F\left(\boldsymbol{\xi}^{(i)}\right)^2 w^{(i)}}$$

where $\boldsymbol{\xi}^{(i)}$ and $w^{(i)}$ are the multi-dimensional nodes and weights of the quadrature rule. For TF2, we used an $N = 10^5$ point LHS ensemble to estimate the relative error using

$$\zeta^2 = \frac{\|F - \sum_{\boldsymbol{k}\in\mathcal{K}} c_{\boldsymbol{k}}\Psi_{\boldsymbol{k}}\|}{\|F\|} \approx \frac{\frac{1}{N}\sum_{i=1}^{N}\left[F\left(\boldsymbol{\xi}^{(i)}\right) - \sum_{\boldsymbol{k}\in\mathcal{K}} c_{\boldsymbol{k}}\Psi_{\boldsymbol{k}}\left(\boldsymbol{\xi}^{(i)}\right)\right]^2}{\frac{1}{N}\sum_{i=1}^{N} F\left(\boldsymbol{\xi}^{(i)}\right)^2}$$

where $\boldsymbol{\xi}^{(i)}$ are the nodes of the LHS ensemble.

The BP and BPDN formulation was solved with the Matlab Toolbox `SPGL1` [128, 129] and the results were validated against other solvers, notably `SparseLab` [33] and `l1-magic` [16]. For TF1, The BPDN cross-validation procedure to determine δ yielded a value on the order of the `SPGL1` tolerance and therefore, only a BP solution is presented. The cross-validation procedure and BPDN was still used for TF2. The *input* basis for the Hybrid-LAR, $\mathcal{K}^{\mathcal{I}}$, was chosen to be $\mathcal{K}^{\mathcal{L}}$, a basis such that if solved directly would be an exact interpolatant factorable into the Lagrange interpolation basis. The basis for the CS investigation was chosen with a sizing factor s and hyperbolic parameter q chosen for Equation (5.16) to ensure $(P + 1) \approx 4N$

(though we could also use the sparsity from the aPSP coefficients to inform the basis selection). We will later examine the sensitivity of the input bases for both methods in Section 5.3.3. The tolerances used for both functions are in Table 5.1.

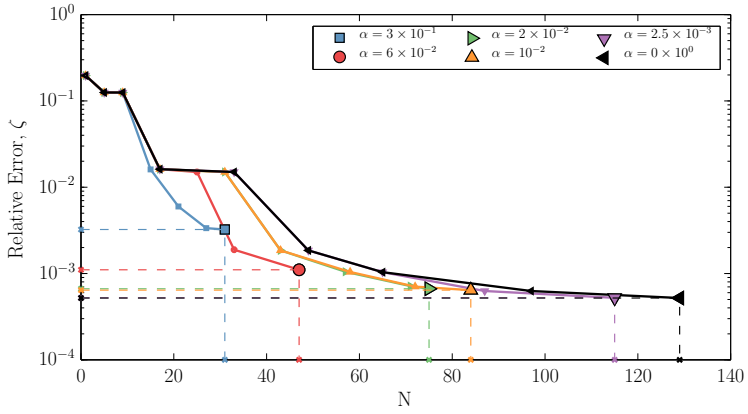
Table 5.1: $\texttt{Tol}_\eta$ values for TF1 and TF2 subsampled PSP.

Function	$\texttt{Tol}_\eta$
TF1 (5.17)	5×10^{-5}
TF2 (5.18)	1×10^{-2}

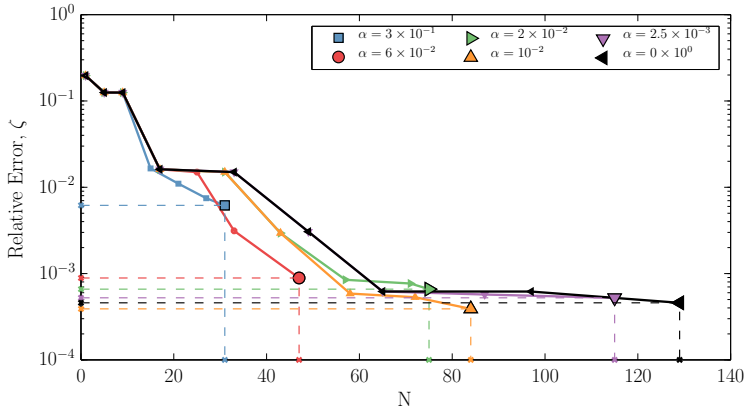
The performance of the subsampled aPSP for TF1 is presented in Fig. 5.7 where the evolution of the adaptation is presented together with the relative error and the required number of function evaluations. Using just the aPSP coefficients, Fig. 5.7(a), $\alpha > 0$ demonstrates considerable savings over the $\alpha = 0$ case. The cost is reduced by subsamping and the final surrogate fidelity is only trivially impacted.

The CS and Hybrid-LAR PCEs in Fig. 5.7(b) and (c) are also particularly interesting. For all α values, the error saturates sooner with these methods and the final surrogate fidelity is the same as or better than that of the PSP. Particularly interesting are the results for $\alpha \leqslant 2.5 \times 10^{-3}$ in Fig. 5.7(c). The Hybrid-LAR outperformed CS and PSP by nearly an order-of-magnitude. This result, for $\alpha = 0$ which removes subsampling, is notable unto itself and merits further investigation to be considered in future work.

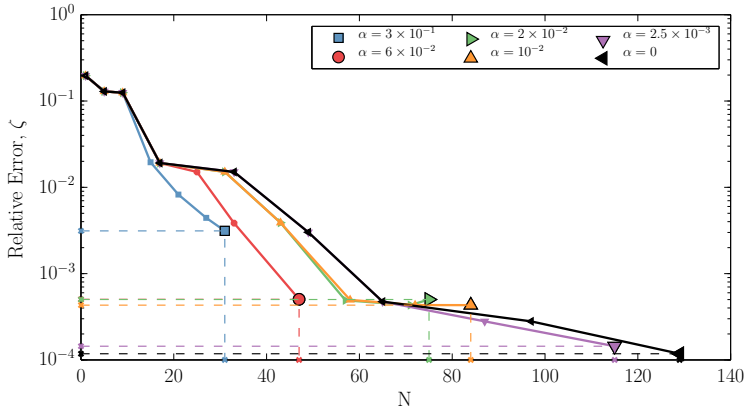
The nature in which subsampled aPSP reduces computation can be easily visualized for the simple 2D problem. Figure 5.8 visually presents the final $\mathscr{L}$ for three different α values. On the top row is the projection surplus, $\epsilon(\boldsymbol{l})$ (3.4) for each $\boldsymbol{l} \in \mathscr{L}$ and on the bottom is the fraction of nodes evaluated $\left(\frac{|\mathscr{X}^{\boldsymbol{l}}|}{|\mathcal{X}^{\boldsymbol{l}}|}\right)$. A smaller $\epsilon(\boldsymbol{l})$ suggests that the index contributes less to the overall variance and is of less importance. $\alpha = 2.5 \times 10^{-3}$ had very little subsampling on $\boldsymbol{l} = \{4, 3\}$ and skipped about 60% of



(a) aPSP



(b) CS BP



(c) Hybrid-LAR

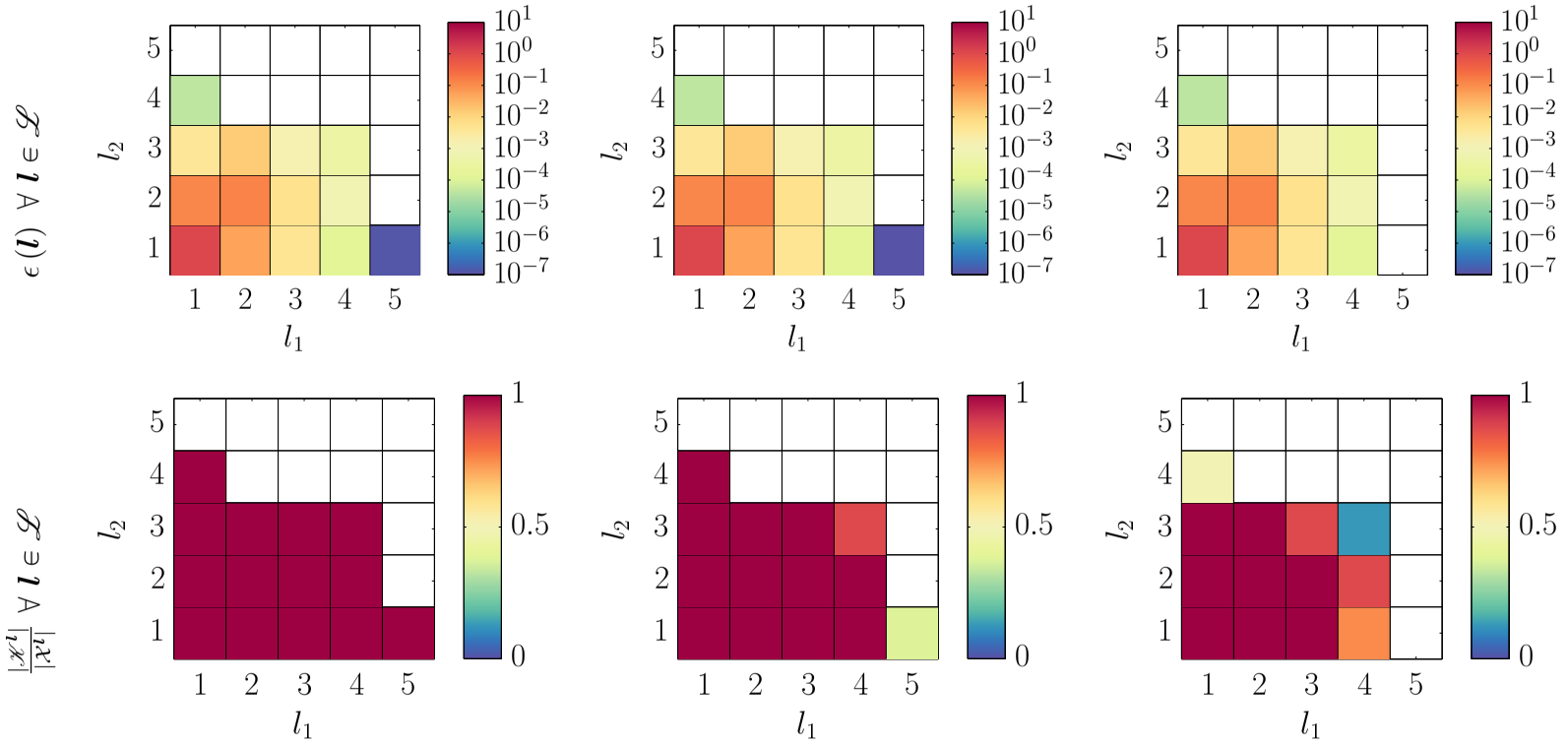
FIGURE 5.7: Relative error for TF1 computed with the subsampled grids for various α values. In (a) is the error using the aPSP estimated coefficients. In (b) is error using the CS BP computed coefficients using only the model data. In (c) is the relative error using the Hybrid-LAR coefficients computed using both the model and synthesized data. Dotted lines help visualize the final terminal error and number of model evaluation.

the evaluations for $\boldsymbol{l} = \{5, 1\}$. From the ϵ values, we see that $\boldsymbol{l} = \{5, 1\}$ is of little importance. We note that all other indices are fully sampled. The adaptation for $\alpha = 2 \times 10^{-2}$ reduced computation by both subsampling the higher-level indices and by omitting $\boldsymbol{l} = \{5, 1\}$ entirely since $\mathscr{K}^{\boldsymbol{l}=\{5,1\}} = \varnothing$.

The subsampled adaptations were also tested for the higher-dimensional TF2 with results presented in Fig. 5.9. Unlike the results of TF1 in Fig. 5.7, there is some loss of fidelity with the subsampled adaptivity at $\alpha = 2.5 \times 10^{-3}$. None-the-less, for $\alpha = 2.5 \times 10^{-3}$ case, there is a 48% savings over the fully-sampled sparse grid. The savings comes in the form of both subsampling and the omission of indices with $\mathscr{K}^{\boldsymbol{l}} = \varnothing$. If the grid for the $\mathscr{L}$ with $\alpha = 2.5 \times 10^{-3}$ were fully sampled, it would require 687 realizations but the subsampling method only evaluated 538; approximately a 22% savings.

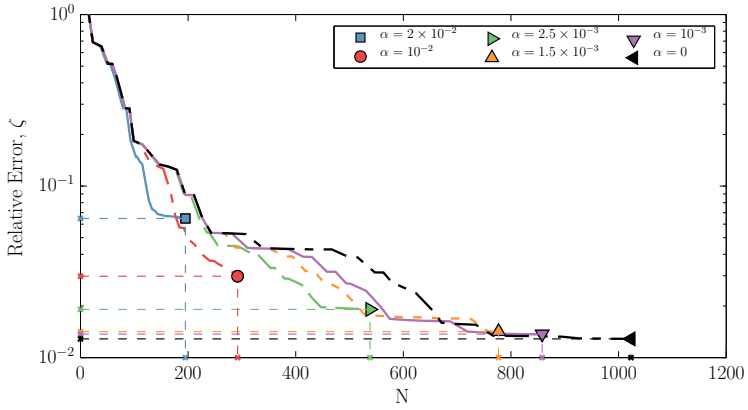
Using α values lower than that used for TF1, we reduced the error while still yielding large savings. Specifically, with $\alpha = 1.5 \times 10^{-3}$, we effectively reduced the error to that of the $\alpha = 0$ adaptation while still realizing a 24% savings. Peculiarly, for $\alpha = 1.5 \times 10^{-3}$, the adaptation did use 8 realizations that were not required in the $\alpha = 0$ case but still resulted in substantial savings. The CS-BPDN error for $\alpha = 1 \times 10^{-3}$ being greater is likely due to the cross-validation procedure not having sufficient permutations to accurately determine δ.

In order to study if the loss of fidelity for some $\alpha > 0$ is due to subsampling (and using synthesized data) or due to the different $\mathscr{L}$, we performed the same analyses but used model data for all $\mathscr{X}^{\boldsymbol{l}}$ instead of just $\mathscr{K}^{\boldsymbol{l}}$. The results for all three methods are plotted in Fig. 5.10. Comparing the results presented in Fig. 5.9 with those of Fig. 5.10, it is clear that the smaller $\mathscr{L}$ (and some of those realizations associated with it) were the cause for the loss of fidelity for $\alpha \geqslant 2.5 \times 10^{-3}$ with CS and LAR. The $\alpha = 1.5 \times 10^{-3}$ and $\alpha = 10^{-3}$ adaptation caused *extra* indices in $\mathscr{L}$ and more realizations than $\alpha = 0$ but, with subsampling, only eight additional realizations

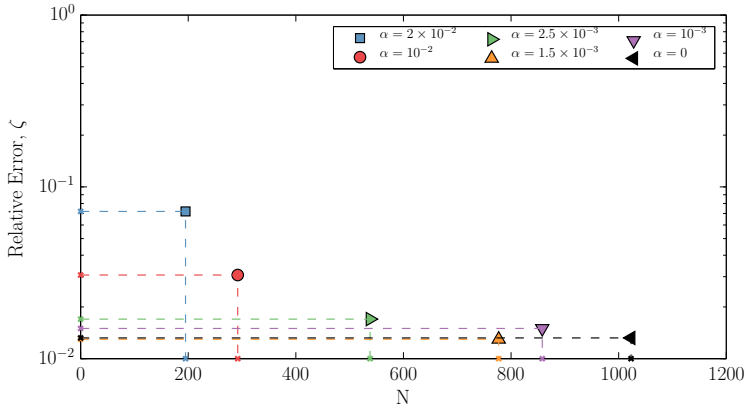


(a) $\alpha = 0,\ N = 129$ (b) $\alpha = 2.5 \times 10^{-3},\ N = 115$ (c) $\alpha = 2 \times 10^{-2},\ N = 75$

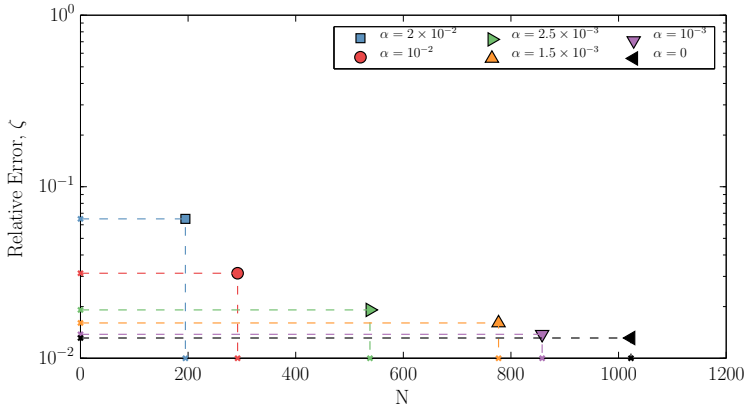
FIGURE 5.8: Visualizations of the final $\mathscr{L}$ for different α values. (top): Indices colored by $\epsilon(\boldsymbol{l})$ (log-scale). (bottom): Indices colored by the percentage of nodes evaluated. $|\cdot|$ is the cardinality of the set. Savings of function evaluations in (b) and (c) come from subsampling sampling. Additional savings in (c) are from the omission of $\boldsymbol{l} = \{5, 1\}$ since $\mathscr{H}^{\boldsymbol{l}=\{5,1\}} = \varnothing$.



(a) aPSP

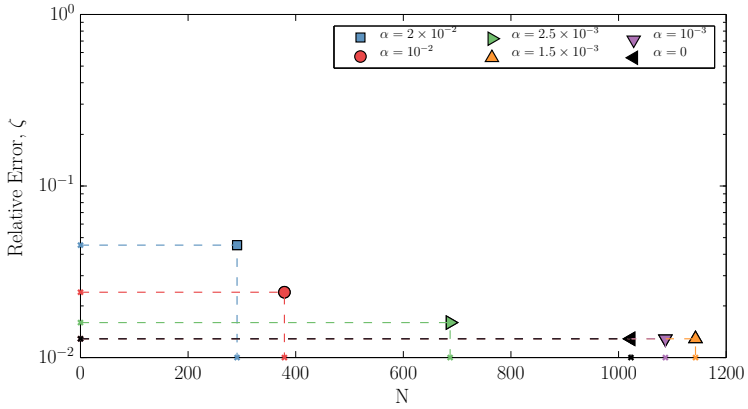


(b) CS BPDN

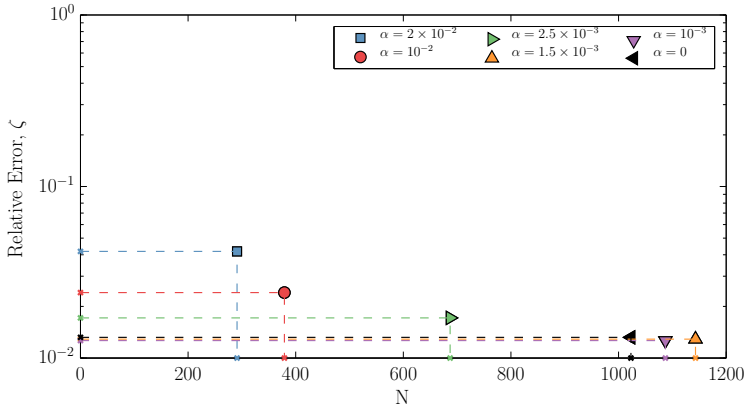


(c) Hybrid-LAR

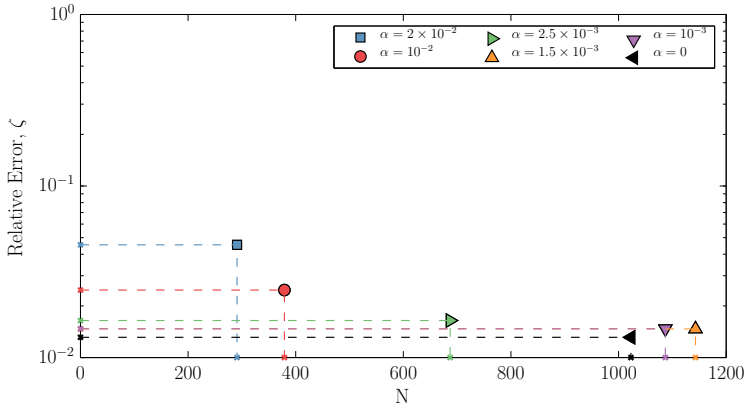
FIGURE 5.9: Relative error for TF2 computed with the subsampled grids for various α values. In (a) is the error using the aPSP estimated coefficients. In (b) is error using the CS BPDN computed coefficients using only the model data. In (c) is the relative error using the Hybrid-LAR coefficients computed using both the model and synthesized data. The error for the full adaptive path is presented in (a) but the CS and Hybrid-LAR coefficients were not computed for the full path. Note that these α values are different than Fig. 5.7.



(a) aPSP



(b) CS BP



(c) Hybrid-LAR

FIGURE 5.10: Relative Error for TF2 computed with the **non**-subsampled grids for various α. Unlike Fig. 5.9, all error measures are computed by using a full sparse grid.

were needed. The final results with subsampling still exhibited savings with these extra realizations.

5.3.2 Determination of α

As we noted in Section 5.2.2, determination of α is often problem dependent and non-trivial. While our focus is primarily proof-of-concept, we demonstrate an example α selection criterion for TF2 in Fig. 5.11. Recall that $\alpha_{n'}$ is the lowest α tested as we decrease α and all quantities should be recomputed using the $\alpha_{n'}$ adaptation. For the CV estimated error, we choose the Leave-One-Out error because it is deterministic. Note that the CV error does not need to be recomputed for all prior α.

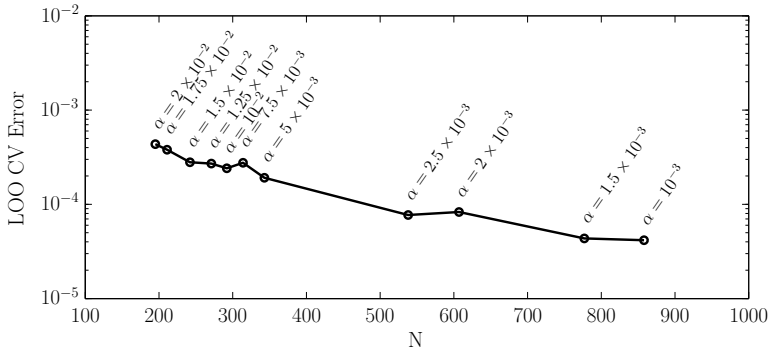
We do not look at a single quantity to determine α and instead consider all quantities together. The plots in Fig. 5.11 may be interpreted to suggest we set $\alpha = 10^{-3}$. We set this because the values recomputed for $\alpha = 1.5 \times 10^{-3}$ and $\alpha = 2 \times 10^{-3}$ change very little; the change in α is causing large changes in N for the previous step (indicating a critical α value); and, finally, the CV LOO error has also stabilized.

Based on the results in Fig. 5.9, a choice of $\alpha = 10^{-3}$ is actually conservative and we may have chosen a higher α. We also note that the L_2 convergence between PCEs closely follows that of the empirical error on the sample points but that latter may also be computed for $\alpha_{n'}$.

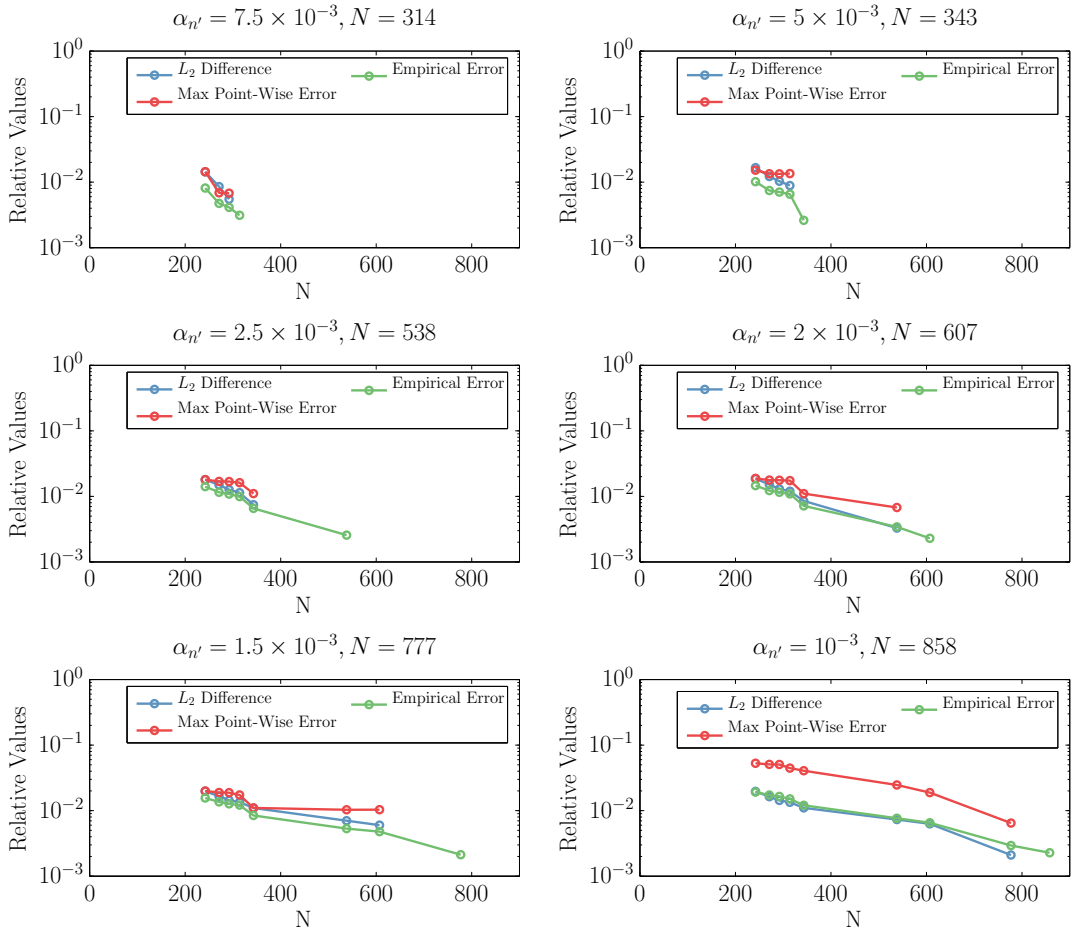
5.3.3 Analysis of CS and Hybrid-LAR Basis Selection

We now wish to analyze the sensitivity of the CS and Hybrid-LAR methods to the size of the input basis. Basis selection for compressed sensing is, in general, non-trivial [127, 35], and in particular, recent work has focused extensively on this issue [64].

In order to investigate the dependence of the basis size on the final result, we



(a) CV LOO error for decreasing α



(b) α convergence metrics recomputed at each α'

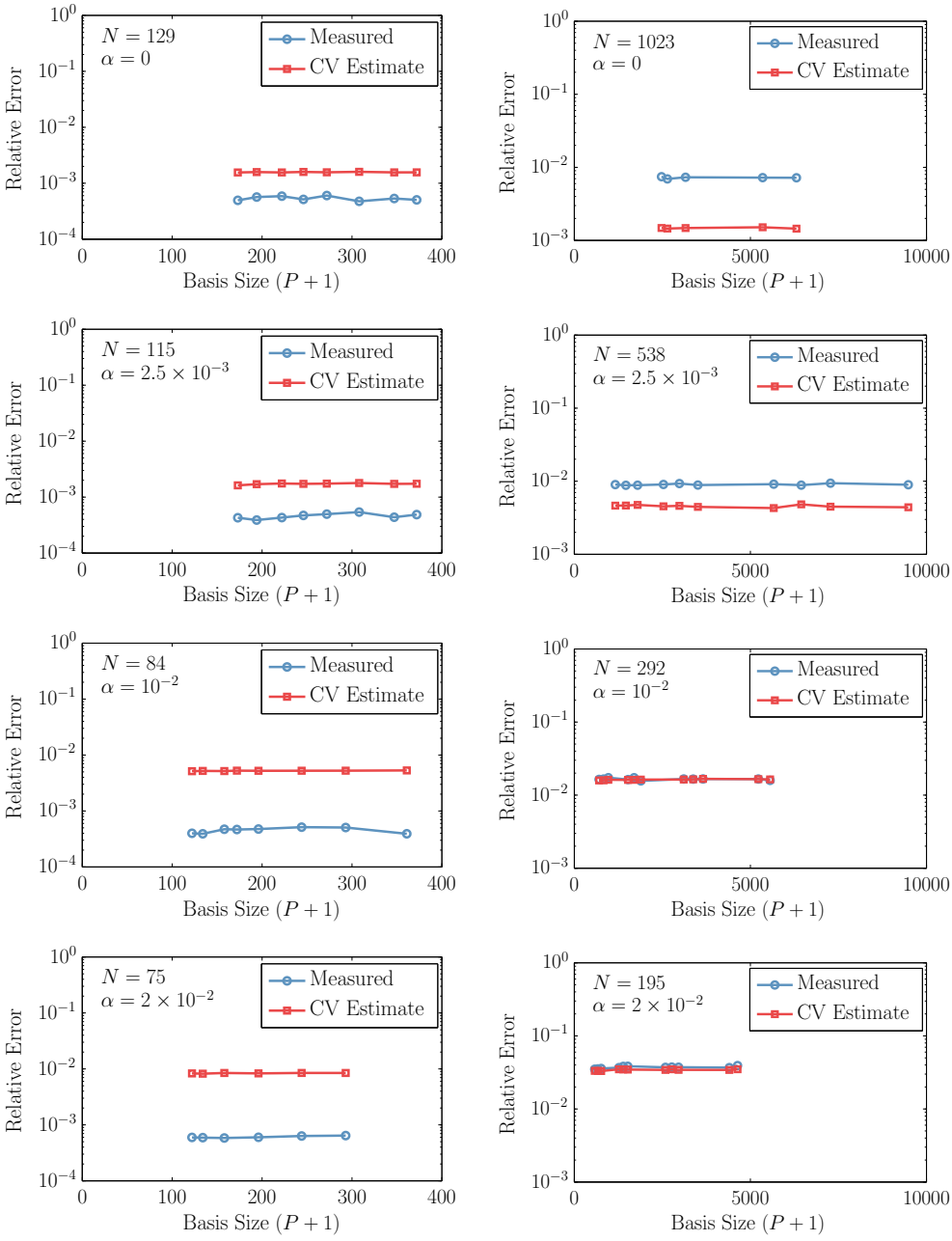
FIGURE 5.11: Example α determination procedure for TF2. In (a) is the LOO CV Error estimate from Hybrid-LAR (2.18). The blue curves in (b) are the relative L_2 difference between $\alpha_{n<n'}$ and $\alpha_{n'}$ (5.12). In red is the maximum point-wise error between $\alpha_{n<n'}$ and $\alpha_{n'}$ on $\mathcal{H}(\mathscr{L}_{\alpha_{n'}}, \alpha_{n'})$ (5.13). In green is the relative empirical error of all $\alpha_{n \leqslant n'}$ (5.14). The blue and red curve represent a comparison to a most enriched PCE ($\alpha_{n'}$) while the green curve is a measure from the model data and is also plotted for $\alpha_{n'}$.

performed numerical experiments on the respective test functions by calculating the error from successively enriched bases. We enlarged the basis suggested by the aPSP, $\mathcal{K}^{*\mathcal{L}}$ (recall that the $\mathcal{K}^{*\mathcal{L}}$ is the maximal PCE basis as suggested by the half-accuracy set and that the PSP method uses this basis), using Equation (5.16) with $q = 0.6$ and increasing s. Given the computational complexity of BPDN for a particularly large basis and that the estimated δ's were small, we used BP for TF2 in this investigation as well as TF1.

Since access to error measures are not available in a production environment, we also looked at the cross-validation error. This is similar to [64] where a K-Fold Cross-Validation Procedure was used for CS basis selection. We examined the behavior for a variety of α values and saw the same insensitivity to basis size. The results of the CS basis selection experiment are presented in Fig. 5.12.

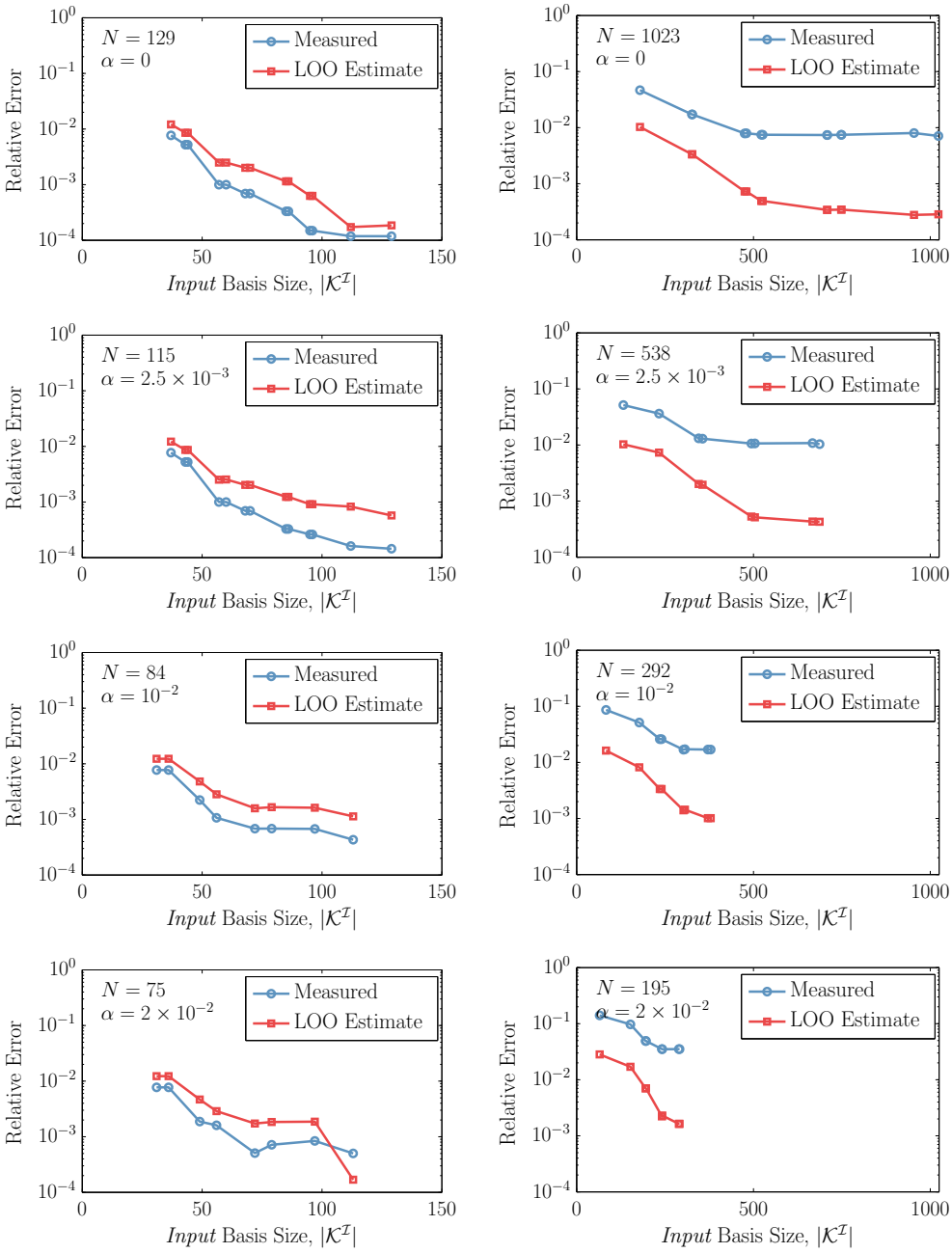
Unlike CS, which requires $(P+1) \gg N$, LAR does not, in itself, have a specific requirement of the input basis $\mathcal{K}^{\mathcal{I}}$. The final *active* basis, $|\mathcal{K}^{\mathcal{A}}| \leqslant \min[|\mathcal{K}^{\mathcal{I}}|-1, N-1]$ such that the final regression step will be well-posed. As such, prior experiments with Hybrid-LAR [9] have developed a basis-selection procedure. However, those experiments were focused on performing Hybrid-LAR on a non-structured grid (e.g. LHS). When building the information matrix $\boldsymbol{\mathcal{Z}}$ (2.16), with $|\mathcal{K}^{\mathcal{I}}| = P+1 > N$ the resulting matrix becomes singular and the LAR procedure fails. As such, the Hybrid-LAR basis investigation was performed by basis selection with Equation (5.16) and $q = 0.6$ but requiring that $(P+1) \leqslant N$. Furthermore, from the addition of the hyperbolic-cross basis, linearly-dependent columns (representing basis functions that are linearly dependent *on these sample points*) were removed with a procedure outlined in [62] using a QR decomposition with rank-estimation. In addition, the $\mathcal{K}^{\mathcal{L}}$ (2.26) basis was also used and is the final data point. The measured and CV LOO estimated error for Hybrid-LAR are presented in Fig. 5.13.

Whereas the CS method showed little to no sensitivity to the input basis (assum-



(a) TF1 (b) TF2

FIGURE 5.12: Relative and CV estimated error for CS with increasing input basis size. In all cases $P + 1 = |\mathcal{K}^{\mathcal{I}}| \gg N$. Plots show low sensitivity to basis size.



(a) TF1 (b) TF2

FIGURE 5.13: Relative and CV LOO estimated error for Hydrid-LAR with increasing input basis size. In (a) is TF1 and in (b) is TF2. The final data point in each plot is the input basis being the full interpolation basis for that grid. Note that the LOO error accurately identifies the optimal basis with the exception of $\alpha = 2 \times 10^{-2}$ with TF1.

ing sufficient size), the Hybrid-LAR coefficients are sensitive to the input. However, for this example with a structured grid of realizations, it appears that the full interpolation basis with $(P+1) = N$ has the best performance. The only exception appears to be the $\alpha = 2 \times 10^{-2}, N = 75$ case for TF1 where a smaller input basis *slightly* outperformed the full input basis. However, the LOO error did not show this and without access to *a posteriori* analyses, one would not choose such a basis. In addition, reduction in error is trivial.

5.3.4 Comparison of Subsampling to LHS

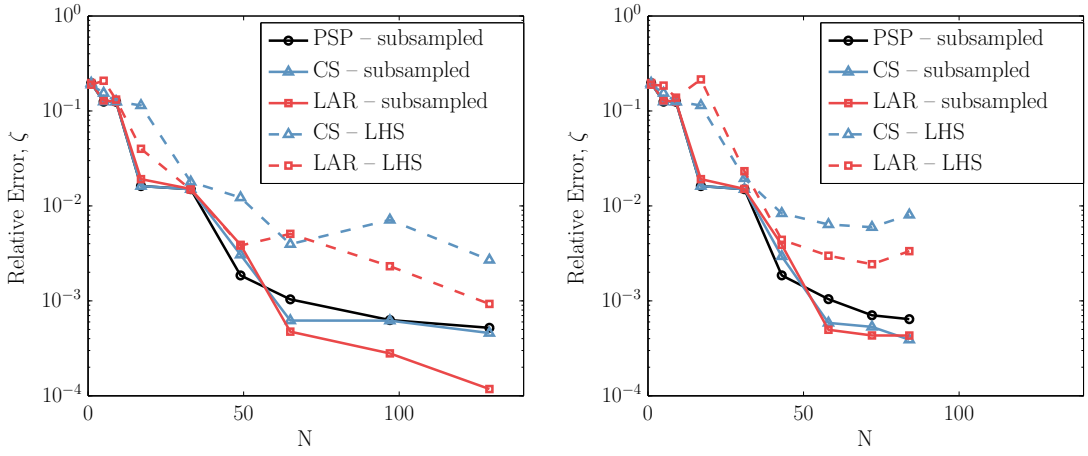
We compared the results on the subsampled grids to a Latin-Hypercube Sample (LHS) of the same size for BPDN and Hybrid-LAR. We used the same polynomial dictionary as that of the respective (subsampled) sparse grids for BPDN and followed a basis-selection procedure like that of [9] for determining the Hybrid-LAR input basis, $\mathcal{K}^{\mathcal{I}}$. Because LHS is not (natively) nested, we simply regenerated the LHS sample for each new number of realizations.

This comparison is not "fair" in the sense that the aPSP results informed the LHS experiment. A purely random-sample based experiment would need to use other methods to both inform the basis selection and convergence criteria [64].

For TF1, we focused on the non-subsampled $\alpha = 0$ run and the $\alpha = 10^{-2}$ adaptation. We compared the subsampled grid coefficients to those from LHS at all stages of the adaptation. Note that we do not seek a projection (PSP) with the LHS grids.

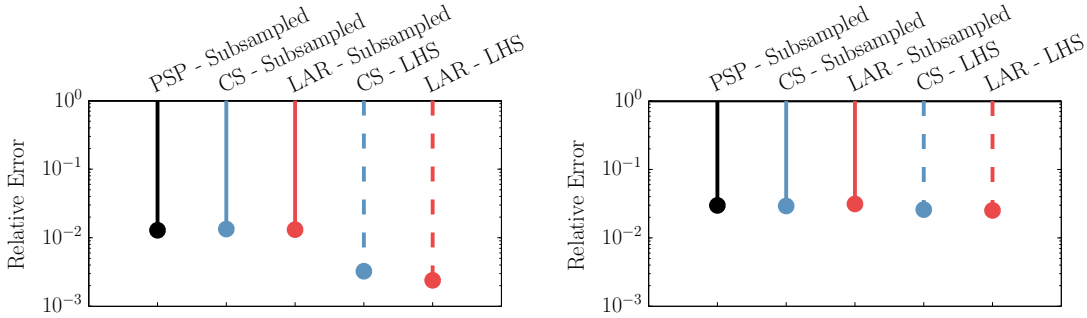
Figure 5.14 shows that sparse and subsampled grids greatly outperform a LHS experiment for the low-dimensional TF1. Such a result is not surprising as LHS experiments are particularly well suited for higher dimensions. As such, we also analysed the performance of LHS on the five-dimensional TF2 at the final projection. These results are presented in Fig. 5.15.

In Fig. 5.15 we see that for $N = 1023$, LHS outperformed the sparse grid while



(a) $\alpha = 0$ (b) $\alpha = 10^{-2}$

FIGURE 5.14: Comparison of an LHS sample to the (subsampled) sparse grids for TF1. In black is aPSP on the sparse grids. In blue is CS BP on the sparse grids (solid) and the LHS ensemble (dotted). In red is Hybrid-LAR on the sparse grids (solid) and the LHS data (dotted).



(a) $\alpha = 0$, $N = 1023$ (b) $\alpha = 10^{-2}$, $N = 292$

FIGURE 5.15: Comparison of LHS to sparse grids (a) and a subsampled grid (b) for aPSP (black), CS-BPDN (blue) and Hybrid-LAR (red). Solid lines are the structured grid and dotted are the LHS sample.

at lower N values, the subsampled structured grid performed similarly to the LHS sample.

We expect that in higher dimensions, LHS may outperform a (subsampled) sparse grid however we note that an LHS-based experimental design lacks many features of the subsampled grids. This includes adapted basis from the aPSP algorithm (recall we used the adapted basis for LHS in this experiment) and a native convergence indicator. Furthermore, sparse grids yield a deterministic experimental design and are nested. Finally, the aPSP approach allows for investigation of existing databases.

5.4 Subsampling on Production Models

Having examining the behavior of the subsampling framework on model test problems, we now perform an *a priori* analysis on the models considered earlier in this work. We first address the low-dimensional HYCOM model (Chapter 3) of The Gulf of Mexico during Hurricane Ivan in September 2004. We will then examine the 25 dimensional shock-tube model from Chapter 4 to examine the behavior of subsampling in a high-dimensional production setting.

In the prior work with the HYCOM model, we only terminated when the database was exhausted. In order to better simulate a production environment, we instead specify a $\texttt{Tol}_\eta = 8 \times 10^{-6}$ for the HYCOM model and use the non-aggressive form of (subsampled) aPSP with our QoI being the area-averaged sea-surface temperature. We expect the adaptation to terminate early as the model database is mostly non-adapted. We again make use of an $N = 256$ LHS ensemble of realizations for estimating convergence. In the shock-tube model, we consider the full 25-dimensional model (including the design parameters) and set $\texttt{Tol}_\eta = 2 \times 10^{-3}$ with $\texttt{Tol}_{\eta,\boldsymbol{\xi}_\text{u}} = \texttt{Tol}_{\eta,\boldsymbol{\xi}_\text{p}} = 0$. We again use an $N = 4096$ LHS ensemble for measuring the projection errors.

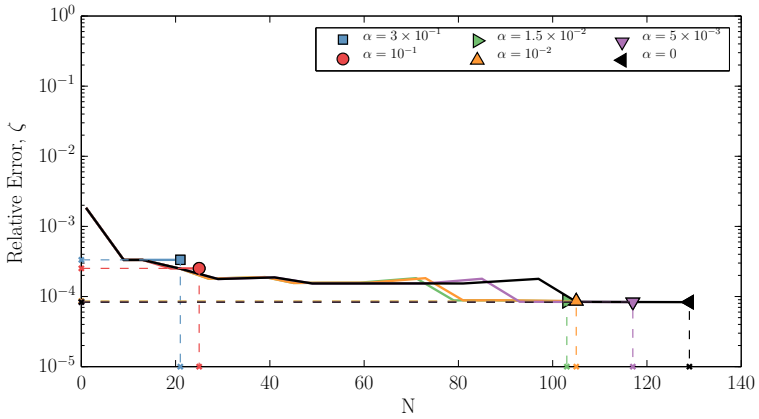
The results for subsampled aPSP, CS-BPDN (without synthesized data) and

Hybrid-LAR (with synthesized data) are presented in Fig. 5.16 for the HYCOM model. Note that the selected α values are different from those of the previous section. In general, we expect that the ideal α values are problem dependent, but this example is particularly different owing to the restriction of the database. Also note that the error performance of aPSP on HYCOM initially appears to be more poor than that of Fig. 3.10. Figure 3.10 was generated with the more aggressive form of aPSP (Algorithm 3.2) while all results in Fig. 5.16 were generated with the less-aggressive form (Algorithm 3.1). In general, especially in high-dimensions, the aggressive form is expected to perform more poorly leading us to choose the less aggressive form for this analysis.

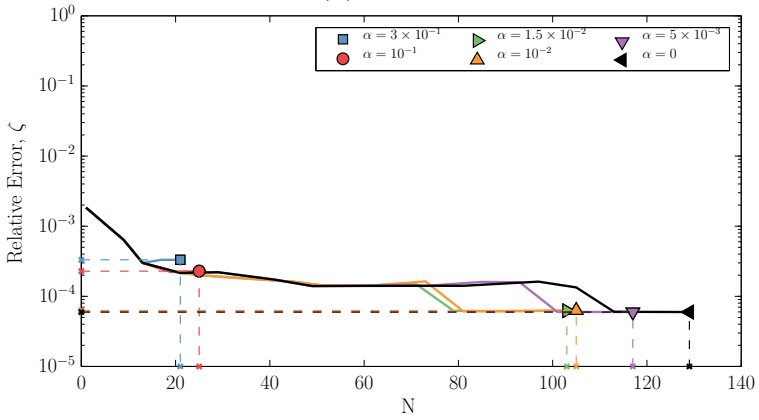
Figure 5.16 again shows savings due to subsampling. An α value as high as 1.5×10^{-2} yielded near-identical error as the $\alpha = 0$ (no subsampling) adaptation. The subsampled adaptation reduced the error and later terminated with fewer model evaluations. Also evident from Fig. 5.16(b) is that for $\alpha \leqslant 1.5 \times 10^{-2}$, CS-BPDN slightly out-performed Hybrid-LAR and PSP with the same model evaluations. This, again, is a particularly interesting result for future analysis.

We also investigated the performance of the subsampling framework on the combustion kinetics problem examined in Chapter 4 (and [133]). In particular, we considered the product-space of the $\boldsymbol{\xi}_{\mathtt{u}}$ and $\boldsymbol{\xi}_{\mathtt{p}}$ variables and treated the problem as a single 25-dimensional problem with $\mathtt{Tol}_{\eta} = 2 \times 10^{-3}$. We plotted the relative ζ values for a few specific α values in Fig. 5.17.

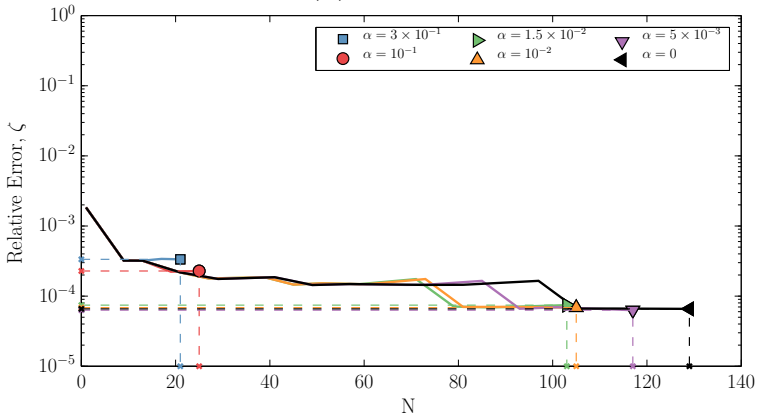
We made use of the PSP coefficients to accelerate the computation; especially for the CS recomputation of the coefficients. While CS requires a large dictionary, we are actually most interested in ensuring we have sufficient samples with respect to the (assumed) sparsity. As such, we may use the coefficients estimated with PSP above a theshold (5.15) to estimate sparsity. An example is plotted in Fig. 5.18 with the PSP coefficients using $\alpha = 5 \times 10^{-4}$. With an $\epsilon = 10^{-9}$, we capture all but 10^{-9} of



(a) aPSP

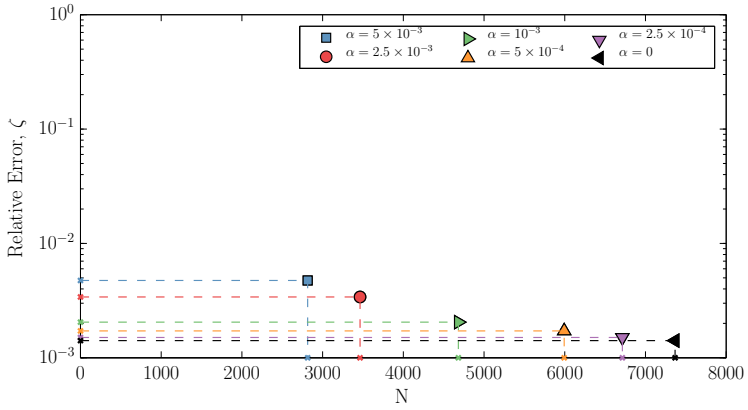


(b) CS BP

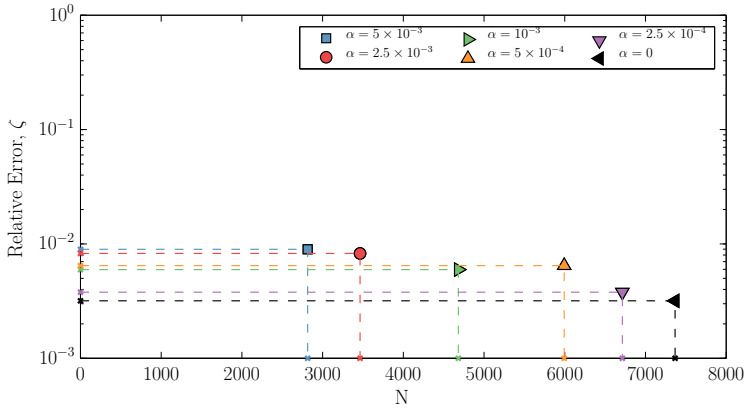


(c) Hybrid-LAR

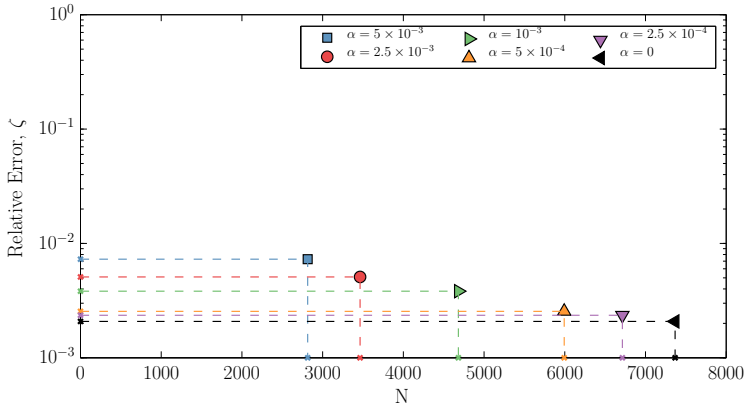
FIGURE 5.16: Relative error for HYCOM computed with the subsampled grids for various α values. In (a) is the error using the aPSP estimated coefficients. In (b) is error using the CS BP computed coefficients using only the model data. In (c) us the relative error using the Hybrid-LAR coefficients computed using both the model and synthesized data.



(a) aPSP



(b) CS BPDN



(c) Hybrid-LAR

FIGURE 5.17: Relative error for the combustion model computed with the sub-sampled grids for various α values. In (a) is the error using the aPSP estimated coefficients. In (b) is error using the CS BPDN computed coefficients using only the model data. In (c) us the relative error using the Hybrid-LAR coefficients computed using both the model and synthesized data.

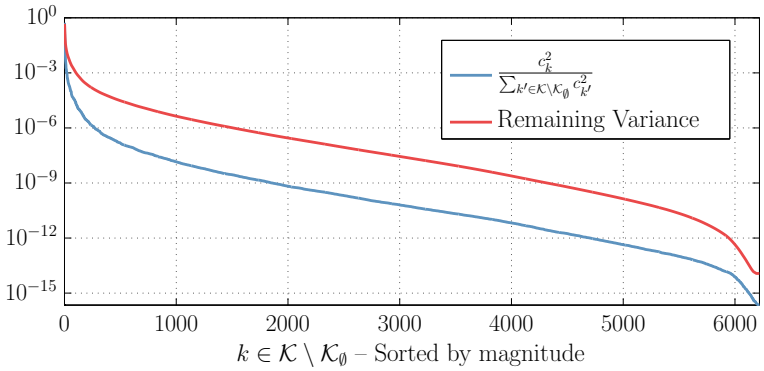


FIGURE 5.18: Sorted, normalized PSP coefficients for the combustion chemistry (blue) from the subsampled PSP with $\alpha = 5 \times 10^{-5}$. Also plotted is the remaining variance. Remaining variance is the cumulative sum of the blue curve subtracted from one. The remaining variance plateaus when the normalized coefficient reaches machine precision.

the variance with only about 66% of the (already adapted) aPSP basis. We therefore know before the CS investigation that the sparse basis of size $\sim$ 4000 is sufficient and we can therefore set the dictionary to be smaller than we may otherwise.

We again found that we could save a small but non-trivial number of realizations with subsampling. While the final PCE fidelity was only slightly reduced, $\alpha = 2.5 \times 10^{-4}$ yielded a 9% savings of model evaluations. Given the high dimensionality of this problem and the large number of realizations required, even a 9% savings is useful. Unlike other examples, we did not find Hybrid-LAR and BPDN to reduce the error over PSP; even with synthesized data. We note that the BPDN procedure is not deterministic in selection of δ and, given the size of this problem, we were restricted in the number of permutations of the cross-validation procedure (Algorithm 2.2).

We also reprocessed the results with CS-BP on the full sparse-grid with synthesized values. Due to the size of the CS problem, the increased computational cost of estimating a denoising δ, and the non-deterministic nature of the δ estimation procedure, we performed this investigation with Basis Pursuit (as opposed to BPDN). While the CS without synthesized data proved to work well in prior investigations

we wished to examine if this is the case for this particular high-dimensional problem. The results are plotted in Fig. 5.19. Figure 5.19 demonstrates that with BP and

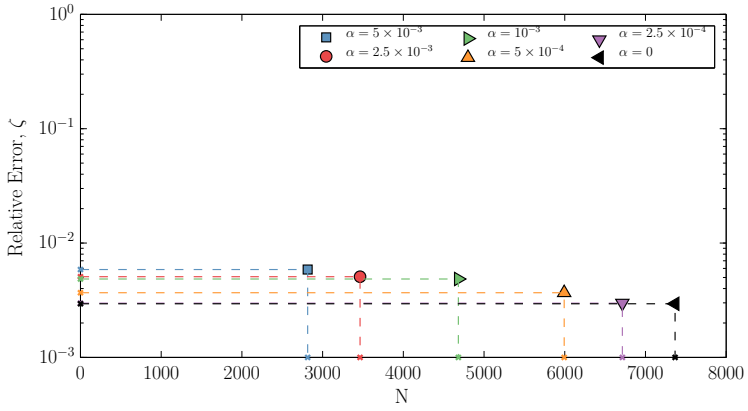


FIGURE 5.19: Reprocessing of the combustion model with BP and using synthesized data. See Fig. 5.17(b) for CS-BPDN without synthesized data.

the full sparse-grid (with synthesized values) we find nearly identical performance of $\alpha = 2.5 \times 10^{-4}$ to that of $\alpha = 0$ but PSP outperforms both CS-BP and CS-BPDN.

Following the same analysis of TF2 in the previous section, we again examined the behavior of the adaptation as we decrease α *without considering* $\alpha = 0$. The quantities described in Section 5.2.2 to aide in determining the ideal α are plotted in Fig. 5.20. Again, there is not an exact definition and convergence may be subjective. We note from Fig. 5.20(a) that for each α, nearly 90% of points are evaluated and the basis grows at each new α. However, the basis only grows slightly for $\alpha \leqslant 5 \times 10^{-4}$ but more nodes are evaluated. This suggests that savings is due both to subsampling and the omission of specific indices. In particular, it also shows that a very small change in α is causing a large change in the quantities. In a production environment, we would have suggested terminating at $\alpha = 5 \times 10^{-4}$ due to the large increase in N and from the maximum point-wise error estimate. We note how for maximum point-wise error for $\alpha = 7.5 \times 10^{-4}$ has plateaued and then that plateau drops for $\alpha = 5 \times 10^{-4}$ suggesting we that just included a set of critical realizations. We also

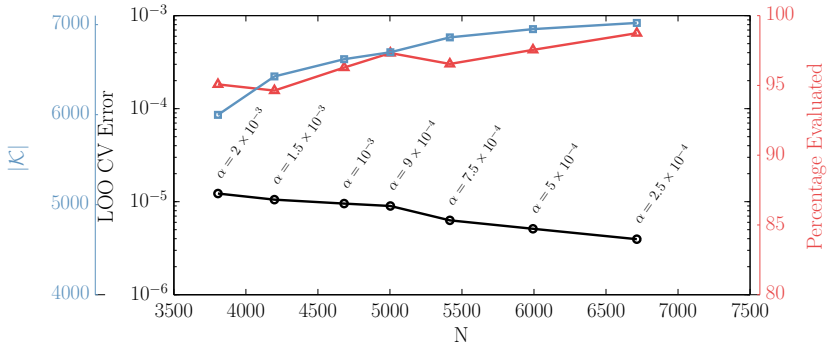
note that the LOO error estimate, while not plateaued, has decreased only slightly between $\alpha = 7.5 \times 10^{-4}$ and $\alpha = 5 \times 10^{-4}$.

The *a priori* analysis and validation of the subsampling framework on production models used in prior chapters further verifies the potential savings of subsampling. We note that there is a lot of room for improvement of α (and $\texttt{Tol}_\lambda$) selection criteria as well as for deciding when to reprocess with CS or LAR (or another solver). However, this section has demonstrated that small but non-trivial savings may be realized.

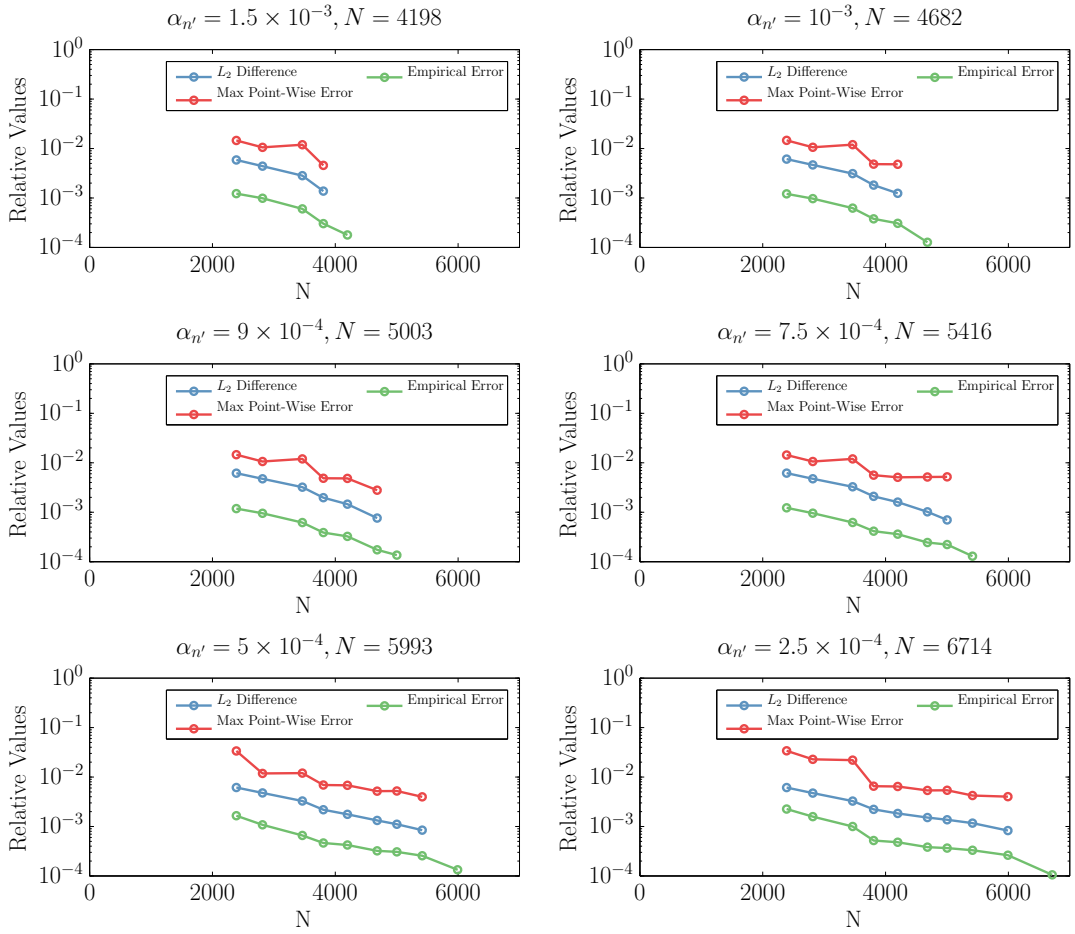
5.5 Conclusion

In this chapter we introduced subsampling on sparse grids. We exploited the nested-grid hierarchical structure of sparse grids to identify areas of stochastic space where a low-order PCE is sufficiently converged. Our goal was to supplement, but not supplant, the aPSP algorithm which already used global dimensional adaptation to greatly reduce computation. We used the local convergence estimates, along with a dynamically-set local tolerance, to effectively reduced the number of realizations for a final projection by synthesizing "skipped" model evaluations.

In order to still perform NISP and PSP, we introduced an effective means to synthesize model data in the so-called skipped regions. Using synthesized data introduced error into our spectral projection and, as such, we used CS-BP/BPDN and Hybrid-LAR, two examples of more robust solvers, to determine the coefficients. The CS method relies on a sufficiently large polynomial dictionary and exploits the sparse nature of the problem to recover a solution. It does not inherently require the use of the synthesized data. Hybrid-LAR combines least-angle-regression with regular least-squares regression to improve performance. LAR successively enriches an active set of input basis terms which we then combined with regression to determine the coefficients. When computing the regression coefficients, we utilized the



(a) CV LOO error, percentage of nodes evaluated, and the size of the basis for decreasing α



(b) α convergence metrics recomputed at each α'

FIGURE 5.20: Example α determination procedure for the 25-dimensional combustion chemistry model. In (a) is the LOO CV Error estimate from Hybrid-LAR (2.18), the size of the basis (blue) and the percentage of evaluated nodes (red). In (b) are α convergence quantities recomputed for each $\alpha_{n'}$.

inherent leave-one-out error to estimate performance and chose the basis with the lowest error. In order to maintain a full-rank matrix, Hybrid-LAR uses both model and synthesized data.

We also made use of the estimated coefficients and the aPSP-prescribed sparse grid to accelerate Hybrid-LAR and CS. The aPSP basis is enriched yielding an adapted CS basis. In addition, the estimated coefficients may be used to accelerate CS performance and to determine necessary conditions for recovery by providing a sparsity estimate. Numerical tests suggest insensitivity to the CS basis provided it is sufficiently large. The structured nature of the aPSP sparse grids allowed for rapid determination of an optimal Hybrid-LAR input basis. Numerical tests suggested that using the full interpolant basis as the *input* basis yielded superior performance.

We also introduced a suggested set of metrics by which to determine an optimal α. These metrics are designed to help identify regions where the synthesized data may be insufficient. We examined how one could use these metrics on the TF2 test problem and the high-dimensional chemical kinetics model.

We first evaluated the performance of these methods on two model test functions. We then performed an *a priori* analysis on HYCOM and the chemical combustion model from previous chapters. In all cases, we found a small but non-trivial savings were realized with little loss of PCE fidelity. We examined the direct ways in with the subsampling improved performance for the 2D model test problem.

The problem-dependence of α represents the most promising avenue for future improvements. We implemented a local-tolerance selection technique based on the current state of the entire projection. Alternative and improved selection techniques are our first priority for improvement on these methods. We also note that a wide class of problems exist that may be ill-suited to local (as opposed to global) adaptivity and we want to ensure our tolerance-selection criteria may identify these cases.

There exists many additional promising uses for the techniques presented in this

chapter; especially in production environments. When computational resources are particularly constrained, the local error indicators may be used to prioritize resources. Synthesized data may be used in place of model data until sufficient computations may be performed and appropriate analyses may be performed in parallel as new data is computed. In addition, current efforts to quantitfy more complex sources of uncertainty (to be published elseware) is encountering model instabilities in certain regions of the paramater space. The methodology presented here may be of particular use in estimating the effect of insufficient model data and to synthesize the data when appropriate. Such uses will be presented in future publications.

6

Summary & Future Extensions

6.1 Summary

Polynomial chaos expansions (1.1) are a useful tool in quantifying and understanding uncertainty. They can serve as a high-fidelity surrogate, and are inexpensive to evaluate while also providing immediate access to important statistical properties of the underlying model. We focused exclusively on non-intrusive methods wherein the PCE coefficients are estimated through an ensemble of model realizations. This approach allowed us to consider large, legacy solvers without modifying the underlying code. In particular, we focused on the Hybrid Coordinate Ocean Model, a large OGCM and on the subgrid scale parameterizations used in the model. We also investigated a high-dimensional shock-tube ignition model where we considered experimental design variables in addition to uncertain parameters.

The primary goal of the work presented in this dissertation has been to develop, implement and test novel sparse grid algorithms to efficiently and accurately create a PCE model surrogate. We primarily focused on Non-Intrusive Spectral Projection which used a sufficiently exact quadrature to project the model response onto the

span of our PCE basis. Traditional tensor-product quadratures suffer the so-called curse of dimensionality wherein the number of evaluation nodes scales exponentially with increasing dimension.

Efforts at adaptive Smolyak quadratures for direct spectral projections were hampered by accuracy requirements to avoid internal aliasing which occurs when the quadrature is not sufficiently exact to evaluate the inner-product of *all* polynomial pairs. This requirement is especially restrictive in an adaptive setting where high-order monomial terms in one direction would alias low-order terms in another direction. To avoid aliasing, significantly more quadrature nodes (model evaluations) would be required and/or the resulting basis would be severely constrained.

An alternative to avoid internal aliasing is the Smolyak pseudo-spectral projection (PSP) introduced in [25]. PSP makes use of a direct application of Smolyak tensor products to the projection operator instead of the quadrature operator. PSP effectively allow for the use of arbitrary sparse grids by making the final projection a combination of aliasing-free, full-tensor, sub-projections. In addition, for any given sparse grid, PSP allows access to the theoretically polynomial basis, further improving projection fidelity at no additional computation cost.

Following the work of *Conrad and Marzouk* [22], we combined the PSP method with an adaptive Smolyak quadrature algorithm [46] to develop a general adaptive Smolyak pseudo-spectral projection, aPSP. We enriched our HYCOM realization database and investigated using aPSP. In addition, we generated a LHS ensemble of realizations for verification. We utilized an aggressive form of the aPSP algorithm and found significant gain in efficiency. In addition, through access to many more polynomial terms in our basis (especially monomial terms), we achieved better accuracy than direct spectral projection on the same realizations.

We next turned our attention to a high-dimensional shock-tube model with 22 uncertain reaction rate constants. In addition to the rate constants, we wished to

develop a PCE with regard to three design parameters while maintaining control over convergence in the respective directions. We developed and implemented two approaches and validated them on a model test function. The first was a nested approach wherein we performed aPSP for the design parameters (the "outer" space), and at each realization of that aPSP model, we ran aPSP in the 22 uncertain reaction rates (the "inner" space). We then proceeded to project the coefficients of the inner-space into the outer-space yielding a single functional form for the dependence of the model on both types of variables. The nested approach yielded superior control of error performance in each disparate space but was inefficient and required many additional function evaluations. We therefore developed an alternative approach that considers the inner and outer spaces as a single product space. We performed aPSP in this product space but such an application lacked control of convergence in each particular direction. We developed a Sobol-based decomposition of the aPSP termination criteria and two methods by which to halt adaptation in a certain group of directions. The product-space construction allowed for a more sparse representation and required significantly fewer realizations than the nested algorithm. The additional directional control and termination allowed us to better control the final termination and further reduce the required number of model evaluations.

We finally turned our attention to adding local adaptivity to the globally adaptive aPSP algorithm. The underlying quadratures of the aPSP method do not inherently allow for any local adaptivity and therefore may have more model evaluations than necessary in certain areas of stochastic space. As such, we followed the methods of [88] to exploit the hierarchical nature of the nested quadrature grids and, following [63], incorporated it into our dimensionally adaptive procedure. In order to perform the required quadratures for NISP, we synthesized model data with lower-order projections. We also developed a dynamic tolerance formula to control local adaptivity that changes tolerance based on the state of the adaptation.

The use of synthesized data introduces error into our otherwise deterministic models and may reduce the projection fidelity. As such, we recommend and implement two more robust solvers to determine the PCE coefficients: compressed sensing (with basis pursuit denoising) and hybrid least-angle regression. Compressed sensing seeks to exploit the inherent (assumed) sparsity of the signal to determine the PCE coefficients using a large polynomial dictionary. We performed compressed sensing with a modified approach to that of [35, 53] to ascertain the PCE coefficients with or without the use of synthesized data. We followed [9] in using a hybrid approach of least-angle-regression, a basis identification algorithm, with standard least-squares regression to determine the PCE coefficients. Numerical tests suggested that we may exploit the structured nature of our realization stencil to achieve an optimal *input* basis for the Hybrid-LAR algorithm.

We applied the subsampled aPSP along with CS-BPDN and Hybrid-LAR to two test problems to validate and verify their behavior. We then performed an *a priori* analysis of the existing HYCOM (with a less aggressive form of the aPSP than that used earlier) and the shock-tube experiment. We found that for the optimal local tolerance, we could realize a small but non-trivial reduction in computation costs while maintaining sufficient PCE fidelity. We developed and presented a few quantities to help select the optimal local tolerances. We also investigated if loss of fidelity was due to subsampling (local adaptivity) or a side-affect of the modified final sparse grid.

In addition, we found that CS-BPDN and Hybrid-LAR performed well for some test problems. With the exception of the high-dimensional shock-tube experiment, the alternative PCE solvers were able to further reduce the error on the subsampled grids. In the shock-tube experiment, we found the PSP with synthesized data to perform most accurately.

6.2 Future work

The methods presented in this document leave open many avenues for future work and extension. In Chapters 4 and 5 we extended the aPSP algorithm with directional control and local adaptivity. We added directional control by developing techniques to decompose the error estimators and we added local adaptivity in part by implementing methods to estimate local errors. In the work presented, we used these new quantities *prognostically*; whether to terminate adaptation in a specific direction or to indicate that we may synthesize data in certain regions. Perhaps the most immediate avenue for extensions of the work presented is to instead incorporate these quantities for *diagnostic* purposes. These quantities can be used to validate the final results or to perform analyses on types of uncertainties.

As we noted in Chapter 5, the subsampled adaptation may deteriorate the quality of the PCE surrogate slightly but that is due to the smaller projection index and not necessarily the use of synthesized data (when reprocessed with CS-BPDN or Hybrid-LAR). Alternative local tolerance selection methodology that does not restrict the quadrature index ($\mathscr{L}$) but still allows for appropriate subsampling may improve performance. This is an active area of future extension.

Future extensions of these methods may come from the combination of those presented here along with newer dimensional reduction techniques. An active and exciting area of research is active subspaces where the coordinate system may be transformed to align with directions of greatest change and can effectively reduce dimension. Combining the transformed coordinates with these adaptive methods may further reduce computational costs. See [24, 23] for more details on active-subspace methods.

A particularly interesting result we found during the subsampling investigation (Chapter 5) is the performance of Hybrid-LAR and BPDN for regular (non-

subsampled) sparse grids. It would be particularly interesting to generalize this result to wide classes of problems; not simply those presented here. Furthermore, the numerical evidence of the insensitivity to polynomial dictionary for BPDN on sparse grids is particularly interesting as the choice of optimal polynomial dictionaries has been a difficult challenge in the past [64]. Extending this result to other polynomial families and other types of grids may prove particularly useful for both accelerating the computation of coefficients and improving performance.

Appendix A

Elementary Introduction to Smolyak Quadrature

A.1 Introduction

This is a tutorial designed to demonstrate how Smolyak quadrature [116] works and to augment Section 2.3.1. It is designed to provide an intuitive feel and is not intended to be a full treatise on the matter. It is assumed, though not required, that the reader have some understanding of quadrature in one-dimension and an idea of quadrature in multiple dimensions.

We will attempt to give an intuitive feel for Smolyak quadrature based on any one-dimensional rule. We will also include brief asides describing certain facets of the construction. We will try to emphasize ideas with examples rather than mathematical formalisms whenever possible.

We are also intentionally omitting a discusion of 1D quadrature rules. It is up to the reader to select an appropriate rule. The only caveat is that selecting a *nested* rule (see Section A.2.2) will greatly reduce the number of nodes.

Most other Smolyak quadrature tutorials only introduce the "traditional" form, but in this work we introduce Smolyak quadrature as a flexible construction that

may be tailored to a particular problem. Much of this material is adopted from [47] and [22].

A.2 1D Quadrature

Quadrature seeks to approximate a weighted definite integral as a weighted sum of function evaluations as follows:

$$\int_a^b f(x)w(x)dx \approx \sum_{i=1}^{N} f(x_i)w_i$$

where $f(x)$ is the function which we are integrating and $w(x)$ is the integral measure (usually constant). The integral is approximated as a sum of N functional evaluations at the x-location *nodes*, x_i, multiplied by an associated weight w_i. The location of the nodes and value of the weights will depend on the choice of 1D rule. The bounds (a and b) and the integral weight, $w(x)$, dictate the appropriate choices of 1D rule. See [47] for more details on choosing a 1D rule.

An example is the level three *Gauss-Patterson-Kronrod* rule is presented in Figure A.1

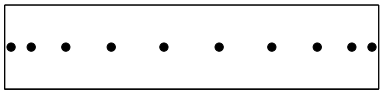


FIGURE A.1: Level three Gauss-Patterson-Kronrod node locations. In this case, a and b are finite and $w(x)$ is constant.

We call the 1D rule Q_ℓ. Therefore, what is plotted in Fig. A.1 is Q_3.

A.2.1 A Note on Exactness

All 1D rules have an associated *polynomial exactness* – The order of a polynomial for which the integral will be exact to machine precision.

For example, a level three Gauss-Patterson-Kronrod rule has seven points and an exactness of 11. Therefore, it will integrate $\int_a^b (c_0 + c_1 x + ... + c_{10} x^{10} + c_{11} x^{11})\, dx$ **exactly** to machine precision.

In this tutorial, we will use Gauss-Patterson-Kronrod 1D rules when appropriate. See Table A.1 for a description of the exactness.

Table A.1: Level, polynomial exactness and number of nodes for one-dimensional Gauss-Patterson-Kronrod quadrature [98, 14]. We note that, unlike traditional Gaussian quadratures, the nodes are nested but the resulting rule is less exact.

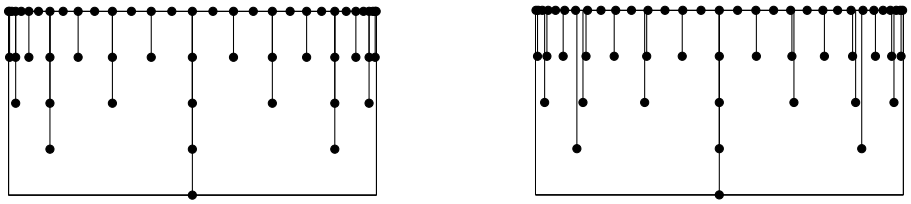
Level, ℓ	Number of points	Polynomial Exactness
1	1	1
2	3	5
3	7	11
4	15	23
5	31	47
6	63	95
7	127	191
8	255	383
9	511	767

We note that polynomial exactness is not the sole measure of a quadrature's accuracy. Some quadrature rules such as Clenshaw-Curtis [21] exhibit lower polynomial exactness but often perform better for a *non-polynomial* function. See [126] for more discussion.

A.2.2 Nested Rules

Nested 1D rules are ones in which increasing the level of the quadrature reuses previously computed points. See Figure A.2 for an example of nested rules and non-nested rules.

There is a cost to using a nested rule. A 15-point Gauss-Patterson-Kronrod rule has polynomial exactness of 23 whereas a 15-point non-nested Gauss-Legendre rule has a polynomial exactness of 29.



(a) Nested (Gauss-Patterson-Kronrod) (b) Non-Nested (Gauss-Legendre)

FIGURE A.2: Example showing how nodes overlap when increasing quadrature level. In (a), a nested rule (Gauss-Patterson-Kronrod). In (b), a non-nested rule (Gauss-Legendre). Notice how most nodes in the Gauss-Legendre rule do not align. Note that Gauss-Patterson-Kronrod rules only exist for certain number of points (see Table A.1) but Gauss-Legendre rules can be define for any N. Plotted in (b) is Gauss-Legendre rules for the same N as the corresponding Gauss-Patterson-Kronrod rule.

It will become clear in Section A.4 why it is important in Smolyak quadrature to use a nested rule.

A.3 Multi-Dimensional Quadrature (full-tensor)

Multi-dimensional quadrature is traditionally created with a *tensor-product*. Mathematically, this looks like

$$\mathbf{Q}_{\boldsymbol{\ell}}^{d} = \mathrm{Q}_{\ell_1} \otimes ... \otimes \mathrm{Q}_{\ell_d}$$

It can be thought of as combining each point in both x and y. The weights are then the product of the 1D weights. For example, consider a 2 point rule:

$$\begin{aligned} x &= \{x_1, x_2\} \\ w &= \{w_1, w_2\} \end{aligned}$$

The resulting 2D *quadrature stencil* has 4-points:

$$\begin{aligned}(x, y) &= \{(x_1, x_1), (x_1, x_2), (x_2, x_1), (x_2, x_2)\} \\ w &= \{(w_1 * w_1), (w_1 * w_2), (w_2 * w_1), (w_2 * w_2)\}\end{aligned}$$

Graphically, this is seen in Figure A.3, again for Gauss-Patterson-Kronrod 1D rules.

Note that levels can combine *anisotropically* – different levels in different directions.

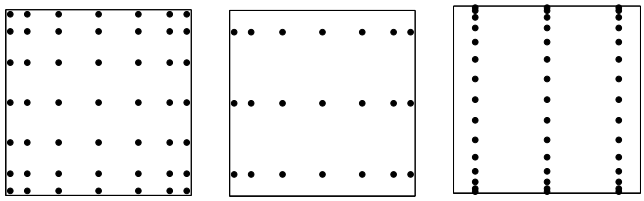


FIGURE A.3: Schematic example of 2D quadrature realizations from the tensor-product of 1D Gauss-Patterson-Kronrod rules. The number of nodes along a particular direction are based on the level, ℓ_i and the corresponding number of 1D nodes from Table A.1. The resulting quadrature stencil may be anisotropic.

A.3.1 Multi-Dimensional Accuracy

The multi-dimensional accuracy of a full-tensor quadrature is based on the 1D rules.

For example, in 2D, the accuracy of $\boldsymbol{\ell} = (3, 3)$ is up to 11^{th} order for Gauss-Patterson-

Kronrod in both x and y. That means that it will integrate

$$\int_a^b\int(\quad c_{0,0} + c_{1,0}x + c_{0,1}y + c_{1,1}xy + ... +$$

$$c_{10,0}x^{10} + c_{11,0}x^{11} + ... +$$

$$c_{0,10}y^{10} + c_{0,11}y^{11} + ... +$$

$$c_{10,10}x^{10}y^{10} + c_{11,11}x^{11}y^{11} \quad)dxdy$$

correctly to machine round-off precision (usually $O(10^{-16})$).

Figure A.4 graphically plots the *exactness set* of the rules in Fig. A.3.

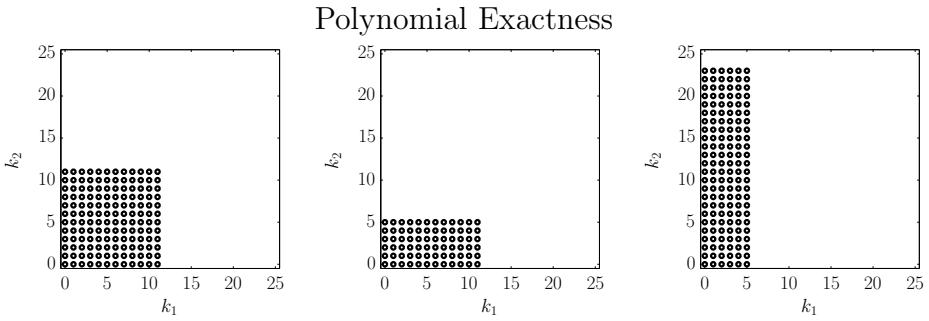


FIGURE A.4: Schematic example of 2D tensor-product quadrature accuracy corresponding to Figure A.3. The circles represent polynomials that can be integrated exactly.

A.3.2 Scaling Issues

The reason for considering alternative quadratures such as Smolyak, is the scaling issues with multi-dimensional quadrature. For a 1D rule with n points, the corresponding full-tensorized rule in d dimensions has

$$N = n^d$$

points. This curse of dimensionality is prohibitive for all but the simplest systems. Furthermore, in many applications, high monomial accuracy is needed however the high *mixed-term* accuracy is not necessary.

This motivates the need for an alternative quadrature; one that scales well and has a more appropriate accuracy profile.

See Figure A.5 for an example of how full-tensor quadrature scales compared to Smolyak quadrature. Note that both curves exhibit the same *monomial accuracy*. That is, ability to, for example, integrate high powers of x or y but not xy.

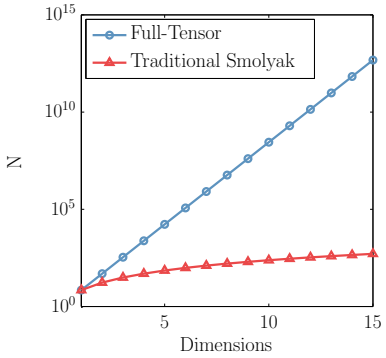


FIGURE A.5: Dimensional scaling for full-tensor quadrature (blue) and Smolyak quadrature (red). Note that both quadratures have the same accuracy for monomial terms.

A.4 Smolyak Quadrature

In this section, we will outline Smolyak quadrature. We will begin with a mathematical overview and then discuss the quadrature multi-index. We will then provide an alternative formulation and explain the notion of a *Smolyak coefficient*. Finally, we will provide an example that aims to make the algorithm more intuitive.

A.4.1 Formulation

The basis of Smolyak Quadrature is to define the quadrature operator as a telescoping sum. Since quadrature is linear, we can write the following:

$$\mathrm{Q}_\ell = \sum_{l=1}^{\ell} \mathrm{Q}_l - \mathrm{Q}_{l-1}$$

where we define $Q_0 \equiv \varnothing$. Alternatively, let

$$\Delta_\ell = \mathrm{Q}_l - \mathrm{Q}_{l-1}$$

leading to

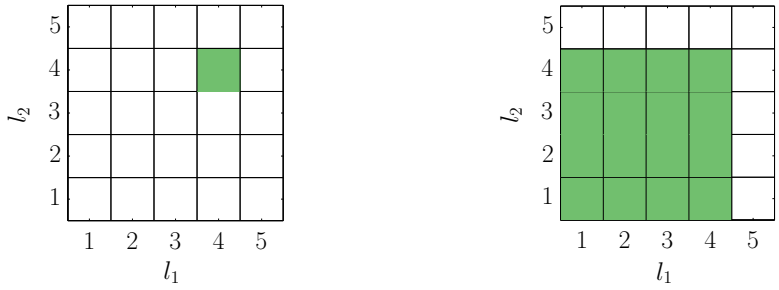
$$\mathrm{Q}_\ell = \sum_{l=1}^{\ell} \Delta_l$$

Now we look at multi-dimensional quadrature. The following demonstrates the key aspect of Smolyak Quadrature.

$$\begin{aligned}
\mathbf{Q}_{\boldsymbol{\ell}}^d &= \mathrm{Q}_{\ell_1} \otimes \ldots \otimes \mathrm{Q}_{\ell_d} \\
&= \sum_{k=1}^{\ell_1} \Delta_k \otimes \ldots \otimes \sum_{j=1}^{\ell_d} \Delta_j \\
&= \sum_{k=1}^{\ell_1} \ldots \sum_{j=1}^{\ell_d} (\Delta_k \otimes \ldots \otimes \Delta_j) = \sum_{\boldsymbol{l} \in \mathscr{L}_{\boldsymbol{\ell}}^{\mathrm{FT}}} (\Delta_{l_1} \otimes \ldots \otimes \Delta_{l_d})
\end{aligned}$$

The last line of the equation is the key formulation to Smolyak Quadrature where the ordering of the tensor-product and the summation operations are exchanged. The summation operation is also represented with a multi-index (see Section A.4.2).

Thus, we reduce the single full-tensor quadrature to a summation of full-tensor rules. This is demonstrated in Fig A.6.



(a) Single quadrature, $\boldsymbol{l} = (4, 4)$ (b) Full-tensor Smolyak quadrature

FIGURE A.6: Visual example of a single quadrature and the Smolyak quadratures. When built on nested quadratures, the end result is identical.

An advantage of Smolyak quadrature is that we now have the flexibility to *truncate* the sum and include fewer mixed quadrature rules. This may be more clear when we discuss the *multi-index* – a summation index that replaces the multiple summations.

Since the Smolyak algorithm relies on the nested sum of different quadrature levels, using nested rules allows the reuse of function evaluations from lower-level quadratures. Using Smolyak quadrature on non-nested rules will often mitigate any savings.

A.4.2 Aside: Multi-Index

The multi-index is a convenient notation for the multiple summations and it is key to "tuning" the quadrature. For example, if we have two summations as follows:

$$\sum_{i=1}^{3}\sum_{j=1}^{3} a_{ij} = a_{11} + a_{21} + a_{31} + a_{12} + a_{22} + a_{32} + a_{13} + a_{23} + a_{33}$$

We can replace the notation of the multiple sums as follows

$$\sum_{i=1}^{3}\sum_{j=1}^{3} a_{ij} = \sum_{\boldsymbol{l}\in\mathscr{L}} a_{\boldsymbol{l}}$$

Where $\mathscr{L}$ is the *multi-index*. In this case, we define $\mathscr{L}$ as follows:

$$\mathscr{L} = \{\{1,1\},\{2,1\},\{3,1\},\{1,2\},\{2,2\},\{3,2\},\{1,3\},\{2,3\},\{3,3\}\}$$

In 2D, this is easy to plot graphically as in Fig. A.7.

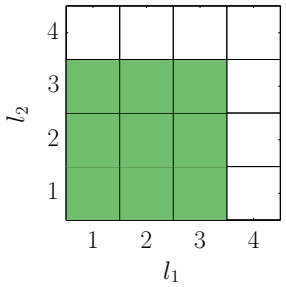


FIGURE A.7: Graphical illustration of a full-tensor multi-index, $\mathscr{L}$.

The key is that we can rearrange this multi-index into any order, and we can *truncate* it as long as it remains *admissible* (explained below). For example. If we rearrange $\mathscr{L}$ to be sorted by *total level* and we require the total level be $\leqslant 4$, we have the following. Take note of the order of the indices.

$$\mathscr{L} = \{\{1,1\},\{2,1\},\{1,2\},\{3,1\},\{2,2\},\{1,3\}\}$$

Graphically presented in Fig A.8. This multi-index is *truncated* by total order. We

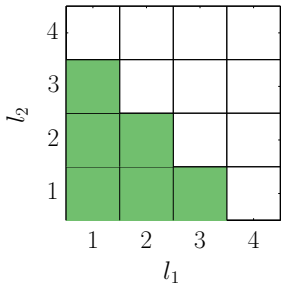


FIGURE A.8: Graphical illustration of multi-index $\mathscr{L}$ truncated by total level 3.

can plug this truncated $\mathscr{L}$ into the Smolyak summation to get a Smolyak Quadrature.

Admissible Indices

Admissibility is the only requirement for a Smolyak multi-index. Mathematically, it is the requirement that

$$\boldsymbol{l} - \hat{\mathsf{e}}_j \in \mathscr{L} \quad \text{for} \;\; 1 \leqslant j \leqslant d, \;\; l_j > 1$$

Where $\hat{\mathsf{e}}_j$ is a unit vector in the j^{th} direction[1] and d is the dimension. In words, this means that each index be "supported" in all directions. This is demonstrated graphically in Fig. A.9.

A.4.3 Multi-Index Truncations

We have establish the form of a Smolyak Quadrature:

$$Q_{\boldsymbol{\ell}}^d = \sum_{\boldsymbol{l} \in \mathscr{L}} (\Delta_{l_1} \otimes ... \otimes \Delta_{l_d})$$

where $\mathscr{L}$ is a multi-index that serves as the key tuning parameter for the quadrature. By selecting different types of multi-indices, one can set the accuracy. (This will be demonstrated later). We will now give an overview of a few common $\mathscr{L}$ options in Table A.2.

Important Note: The exact definitions here assume the lowest order is $\beta = 1$. Care must be taken for certain truncation definitions if $\beta \neq 1$.

A.4.4 Smolyak Coefficient

Recall the basic form of Smolyak Quadrature is

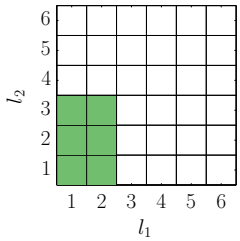
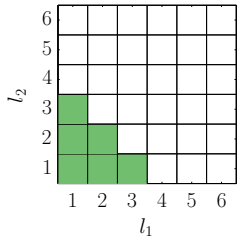
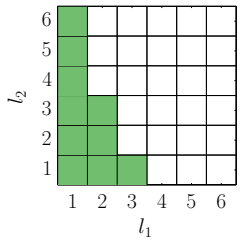
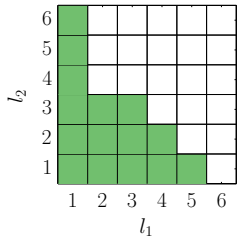
$$\mathbf{Q}_{\boldsymbol{\ell}}^d = \sum_{\boldsymbol{l} \in \mathscr{L}} (\Delta_{l_1} \otimes ... \otimes \Delta_{l_d})$$

However, we can rewrite this as follows:

$$\mathbf{Q}_{\boldsymbol{\ell}}^d = \sum_{\boldsymbol{l} \in \mathscr{L}} c_{\boldsymbol{l}} \, (\mathrm{Q}_{l_1} \otimes ... \otimes \mathrm{Q}_{l_d})$$

[1] Example: For 4 dimensions, $\hat{\mathsf{e}}_3 = \{0, 0, 1, 0\}$ and $\boldsymbol{l} = \{2, 1, 5, 3\} - \hat{\mathsf{e}}_3 = \{2, 1, 4, 3\}$

Table A.2: Examples of different multi-index truncations. Plotted are examples in 2D with the lowest level $\beta = 1$. See Appendix C for further discussion of hyperbolic multi-index truncations.

Multi-Index	Description	Example
Full Tensor	Equivalent to not using Smolyak quadrature (see Section A.4.4).	 $\boldsymbol{\ell} = (2, 3)$
Classical Smolyak	Traditional total-order multi-index of Smolyak quadratures. It is *isotropic* – the same in all directions.	 $\ell = 3$
Anisotropic & Hyperbolic	Allow for anisotropy (with $\boldsymbol{\ell}$) and number of mixed terms (q), starting at β. See Appendix C for more detail. Note, if all $\ell_i = \ell$, $q = 1$, and $\beta = 1$, it reduces to Classical Smolyak	 $\boldsymbol{\ell} = (3, 6)$
Arbitrary and/or Adaptive	Any multi-index that is admissible (described elsewhere) may be used. The multi-index may be built adaptively. See [46] for an adaptive algorithm or Section 3.3 (applied to an operator other than quadrature)	 Arbitrary

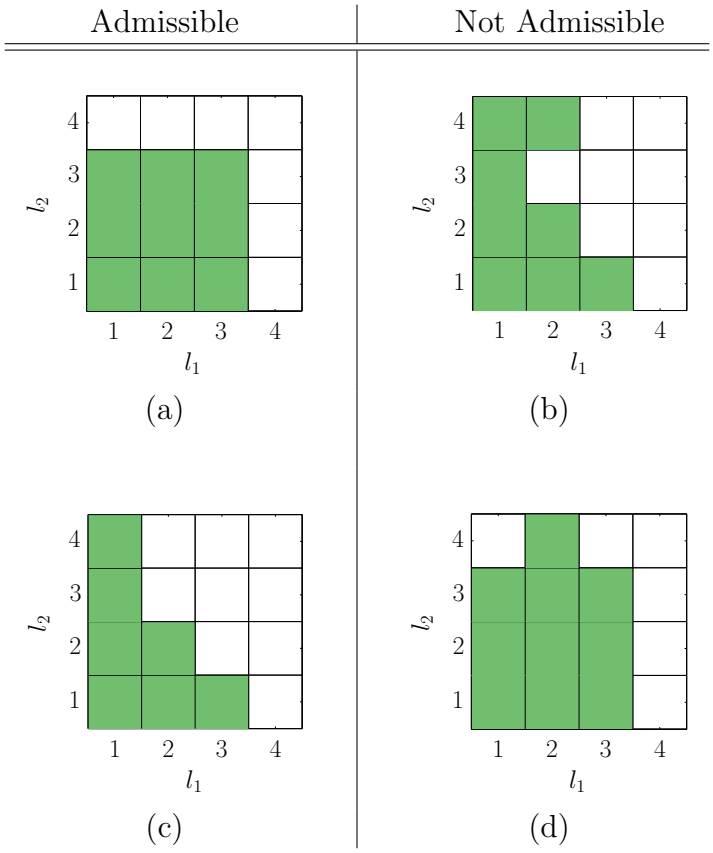
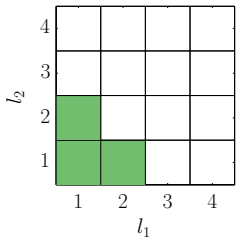


FIGURE A.9: Examples of admissible and non-admissible multi-indices. In (b), the $\{2, 4\}$ index is not supported by a $\{2, 3\}$ index and in (d), it is not supported by a $\{1, 4\}$ index. Figure adapted from [76].

where $c_{\boldsymbol{l}}$ is a *Smolyak coefficient*. The Smolyak coefficient is found by tracking which tensor products are incremented and decremented in the Δ-type construction [22].

It is best seen with a simple example. Consider

$$\mathscr{L} = \{\{1, 1\}, \{2, 1\}, \{1, 2\}\}$$



Therefore:

$$\sum_{\boldsymbol{l}\in\mathscr{L}} (\Delta_{l_1} \otimes ... \otimes \Delta_{l_d}) = (\Delta_1 \otimes \Delta_1) + (\Delta_2 \otimes \Delta_1) + (\Delta_1 \otimes \Delta_2)$$

Now look at each tensor product. Recall $\Delta_1 = \mathrm{Q}_1 - \mathrm{Q}_0 = Q_1 - \varnothing = Q_1$

$$\begin{aligned}
\boldsymbol{l_1} : \Delta_1 \otimes \Delta_1 &= \boxed{\mathrm{Q}_1 \otimes \mathrm{Q}_1} \\
\boldsymbol{l_2} : \Delta_2 \otimes \Delta_1 &= (\mathrm{Q}_2 - Q_1) \otimes \mathrm{Q}_1 \\
&= \boxed{\mathrm{Q}_2 \otimes \mathrm{Q}_1 - \mathrm{Q}_1 \otimes \mathrm{Q}_1} \\
\boldsymbol{l_3} : \Delta_1 \otimes \Delta_2 &= \mathrm{Q}_1 \otimes (\mathrm{Q}_2 - Q_1) \\
&= \boxed{\mathrm{Q}_1 \otimes \mathrm{Q}_2 - \mathrm{Q}_1 \otimes \mathrm{Q}_1}
\end{aligned}$$

Combining all of the boxed tensor-products and simplifying, we find

$$\sum_{\boldsymbol{l}\in\mathscr{L}} (\Delta_{k_1} \otimes ... \otimes \Delta_{k_d}) = -(\mathrm{Q}_1 \otimes \mathrm{Q}_1) + (\mathrm{Q}_2 \otimes \mathrm{Q}_1) + (\mathrm{Q}_1 \otimes \mathrm{Q}_2)$$

Therefore the Smolyak coefficients are as follows:

$$\begin{aligned}
c_1 &= -1 \\
c_2 &= +1 \\
c_3 &= +1
\end{aligned}$$

This is illustrated in Figure A.10.

One note of particular importance, $\sum_{\boldsymbol{l}} c_{\boldsymbol{l}} = 1$ for the telescoping to work properly. If the sum does not equal one, it may suggest an inadmissible multi-index.

2D examples of Smolyak Coefficient

Figure A.11 graphically plots the Smolyak coefficients for 2D multi-indices. Notice that for Fig. A.11(a), all of the coefficients are zero except for the largest multi-

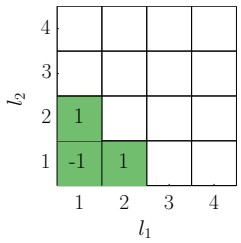
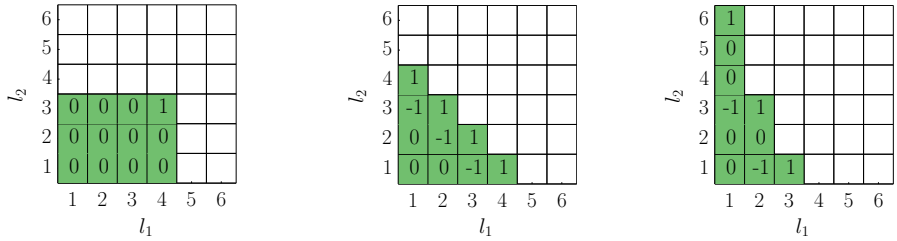


FIGURE A.10: Graphical illustration of the Smolyak coefficient. The number in the box is the corresponding coefficient.

index. This demonstrates that a full-tensor multi-index is equivalent to a normal, tensor-product-based quadrature.

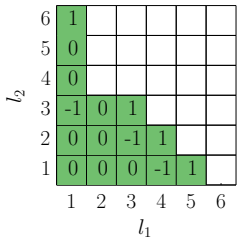
In more than two dimensions, the values will not simply be $c_{\boldsymbol{l}} = 0, \pm 1$. Also, note that all -1 indices fall on the intersections of $+1$ [22].



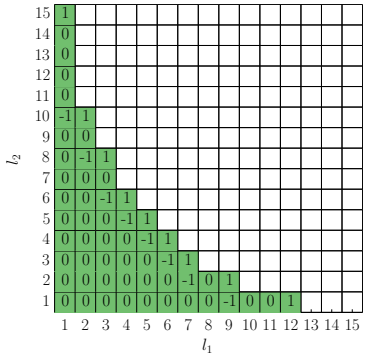
(a) Full Tensor

(b) Traditional Smolyak

(c) $\boldsymbol{\ell} = (3, 6)$



(d) Arbitrary (small)



(e) Arbitrary (large)

FIGURE A.11: Graphical illustrations of the Smolyak coefficients for various truncation in 2D. The values inside the box represent the Smolyak coefficient.

We also note that many $c_{\boldsymbol{l}} = 0$. This may be used to accelerate the computation

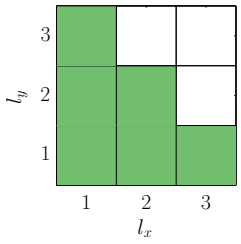
of a Smolyak quadrature rule.

A.4.5 Visual Example of Smolyak in 2D

We now want to visualize the combination of full-tensor rules in Smolyak quadrature. As demonstrated, it is a weighted sum of full-tensor (those discussed in Section A.3) quadratures. The weights are the Smolyak coefficients

For example, consider:

$$\mathscr{L} = \{\{1,1\},\{2,1\},\{1,2\},\{3,1\},\{2,2\},\{1,3\}\}$$



The resulting realization-stencils and accuracy sets are the union of those for each multi-index. This is visualized in Figure A.12.

A.5 Conclusion & Further Reading

This tutorial hopefully provided a clear understanding of Smolyak quadrature in an intuitive way. At its core, the Smolyak algorithm allows one to represent a tensor-product as a summation that can then be truncated as needed or desired.

We direct the reader to [47] for a particularly formal but easy-to-understand introduction to sparse grids including a discussion of error bounds. For adaptive Smolyak quadrature, we direct the reader to [46] where an algorithm is presented to build $\mathscr{L}$ based on the surplus of each full-tensor quadrature. A similar algorithm for other uses is employed elsewere in this document. For an in-depth discussion on the choice of 1D quadrature rules, we direct the reader to [126].

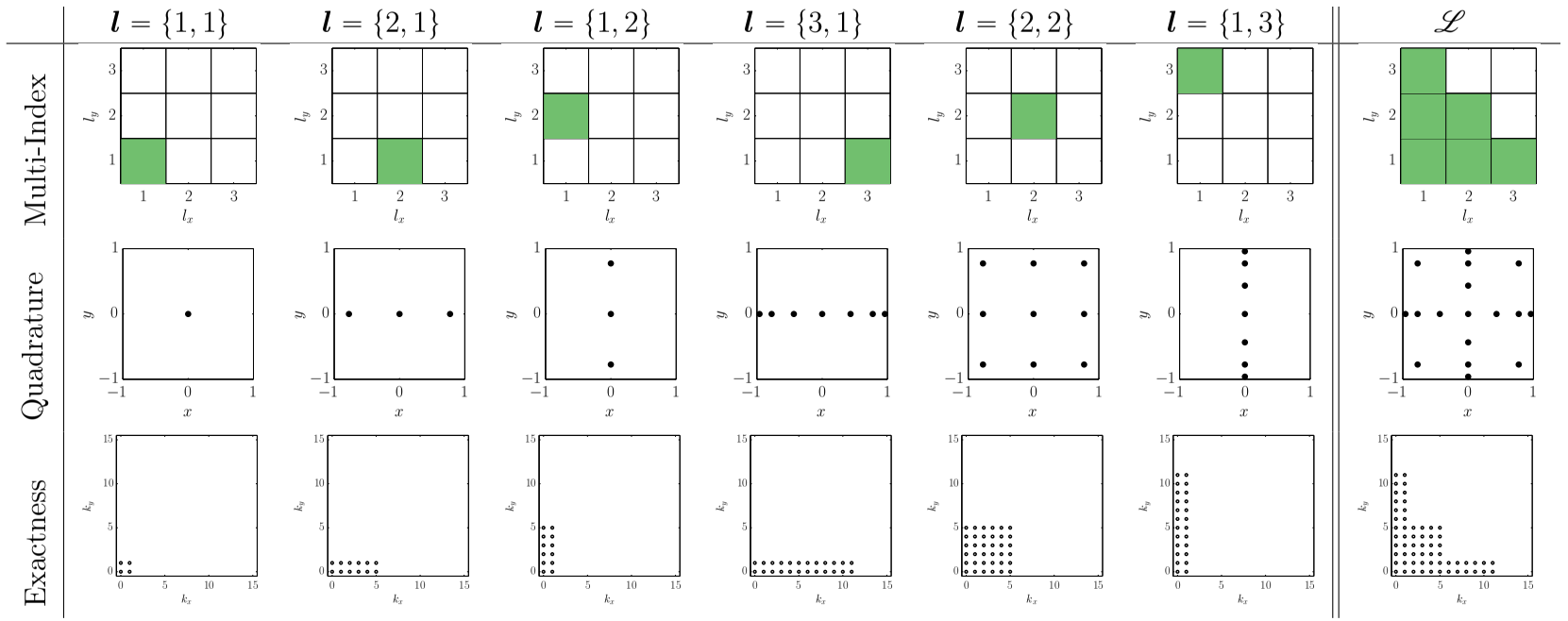


FIGURE A.12: Demonstration showing the individual full-tensor quadratures (top), realization stencils (middle), and accuracy sets (bottom) for a given $\mathscr{L}$. The final (far right) quadrature index, realization stencil and exactness set are the union of those from each $\boldsymbol{l}$ in $\mathscr{L}$.

Appendix B

Global Sensitivity Analysis

B.1 Sensitivity Indices

Consider $F : \boldsymbol{\xi} \in \Xi \mapsto F(\Xi) \in L_2(\Xi)$, where $\boldsymbol{\xi} = (\xi_1 \cdots \xi_d)$ is a vector of d independent real-valued random variable with joined density ρ. Let $\mathcal{D} = \{1, \cdots, d\}$ and for $\mathfrak{u} \in \mathcal{D}$ denote $|\mathfrak{u}| = \mathrm{Card}(\mathfrak{u})$ and $\mathfrak{u}_\sim = \mathcal{D} \backslash \mathfrak{u}$, such that $\mathfrak{u} \cup \mathfrak{u}_\sim = \mathcal{D}$, $\mathfrak{u} \cap \mathfrak{u}_\sim = \varnothing$. Given $\mathfrak{u} \in \mathcal{D}$ we denote $\boldsymbol{\xi}_\mathfrak{u}$ the sub-vector of $\boldsymbol{\xi}$ with components $(\xi_{\mathfrak{u}_1} \cdots \xi_{\mathfrak{u}_{|\mathfrak{u}|}})$, so $\boldsymbol{\xi} = (\boldsymbol{\xi}_\mathfrak{u} \boldsymbol{\xi}_{\mathfrak{u}_\sim})$. Under the stated assumptions, the function $F(\boldsymbol{\xi})$ has a *unique* orthogonal decomposition of the form [57]

$$F(\boldsymbol{\xi}) = \sum_{\mathfrak{u} \in \mathcal{D}} f_\mathfrak{u}(\boldsymbol{\xi}_\mathfrak{u}), \quad \langle f_\mathfrak{u}, f_\mathfrak{v} \rangle = 0 \text{ if } \mathfrak{u} \neq \mathfrak{v}. \tag{B.1}$$

The functions $f_\mathfrak{u}$ can be recursively expressed as [117]

$$f_\mathfrak{u}(\boldsymbol{\xi}_\mathfrak{u}) = \mathbb{E}\{F \mid \boldsymbol{\xi}_\mathfrak{u}\} - \sum_{\substack{\mathfrak{v} \in \mathcal{D} \\ \mathfrak{v} \subsetneq \mathfrak{u}}} f_\mathfrak{v}(\boldsymbol{\xi}_\mathfrak{v}),$$

where $\mathbb{E}\{F \mid \boldsymbol{\xi}_\mathfrak{u}\}$ is the conditional expectation of F given $\boldsymbol{\xi}_\mathfrak{u}$, namely

$$\mathbb{E}\{F \mid \boldsymbol{\xi}_\mathfrak{u}\} = \int F(\boldsymbol{\xi}_\mathfrak{u} \boldsymbol{\xi}_{\mathfrak{u}_\sim}) \rho(\boldsymbol{\xi}_{\mathfrak{u}_\sim}) d\boldsymbol{\xi}_{\sim}.$$

The decomposition (B.1) being orthogonal, the variance of F, $\mathbb{V}\{F\}$, is decomposed into

$$\mathbb{V}\{F\} = \sum_{\substack{\mathfrak{u}\in\mathcal{D}\\ \mathfrak{u}\neq\emptyset}} V_{\mathfrak{u}}, \quad V_{\mathfrak{u}} = \mathbb{V}\{f_{\mathfrak{u}}\}. \tag{B.2}$$

The partial variance $V_{\mathfrak{u}}$ measures the contribution to $\mathbb{V}\{F\}$ of the interactions between the variables $\boldsymbol{\xi}_{\mathfrak{u}}$. Since there are 2^d such partial variances, the sensitivity analysis is usually reduced to a simpler characterization, based on first and total-order sensitivity indices associated to individual variables ξ_i or group of variables $\boldsymbol{\xi}_{\mathfrak{u}}$. The first-order, $\mathtt{S}_{\mathfrak{u}}$, and total-order, $\mathtt{T}_{\mathfrak{u}}$, sensitivity indices associated to $\boldsymbol{\xi}_{\mathfrak{u}}$ are given by [58]

$$\mathbb{V}\{F\}\,\mathtt{S}_{\mathfrak{u}} = \sum_{\substack{\mathfrak{v}\in\mathcal{D}\\ \mathfrak{v}\subseteq\mathfrak{u}}} V_{\mathfrak{v}} = \mathbb{V}\{\mathbb{E}\{F \mid \boldsymbol{\xi}_{\mathfrak{u}}\}\}, \tag{B.3}$$

$$\mathbb{V}\{F\}\,\mathtt{T}_{\mathfrak{u}} = \sum_{\substack{\mathfrak{v}\in\mathcal{D}\\ \mathfrak{v}\cap\mathfrak{u}\neq\emptyset}} V_{\mathfrak{v}} = \mathbb{V}\{F\} - \mathbb{V}\{\mathbb{E}\{F \mid \boldsymbol{\xi}_{\mathfrak{u}_\sim}\}\}. \tag{B.4}$$

The first-order index $\mathtt{S}_{\mathfrak{u}}$ is then the fraction of variance that arises due to the individual variables in $\boldsymbol{\xi}_{\mathfrak{u}}$ and their mutual interactions, only; the total-order sensitivity index $\mathtt{T}_{\mathfrak{u}}$ is the fraction of the variance arising due to the variables in $\boldsymbol{\xi}_{\mathfrak{u}}$, their mutual interactions and all their interactions with all other variables in $\boldsymbol{\xi}_{\mathfrak{u}_\sim}$. Clearly, $\mathtt{S}_{\mathfrak{u}} \leqslant \mathtt{T}_{\mathfrak{u}}$, and $\mathtt{T}_{\mathfrak{u}} = 1 - \mathtt{S}_{\mathfrak{u}_\sim}$ [58].

B.2 Sensitivity Indices of PC Expansion

The partial variances and sensitivity indices of F can be easily computed from the PC expansion of F,

$$F(\boldsymbol{\xi}) \approx \sum_{\boldsymbol{k}\in\mathcal{K}} f_{\boldsymbol{k}}\Psi_{\boldsymbol{k}}(\boldsymbol{\xi}),$$

where $\mathcal{K}$ is the multi-index set of polynomial tensorizations. Owing to the linear structure and polynomial form of the PC expansion, one can easily partition $\mathcal{K}$ into

distinct subsets $\mathcal{K}_{\mathfrak{u}}$ contributing to the PC expansion of $f_{\mathfrak{u}}(\boldsymbol{\xi}_{\mathfrak{u}})$ [122, 26]. Specifically, the PC approximation of $f_{\mathfrak{u}}$ is

$$f_{\mathfrak{u}}(\boldsymbol{\xi}_{\mathfrak{u}}) \approx \sum_{\boldsymbol{k}\in\mathcal{K}_{\mathfrak{u}}} f_{\boldsymbol{k}}\Psi_{\boldsymbol{k}}(\boldsymbol{\xi}), \quad \mathcal{K}_{\mathfrak{u}} \doteq \{\boldsymbol{k}\in\mathcal{K}; k_{1\leqslant i\leqslant d} > 0 \text{ if } i\in\mathfrak{u}, k_{1\leqslant i\leqslant d} = 0 \text{ if } i\notin\mathfrak{u}\}.$$

Observe that $\mathcal{K}_{\varnothing} = \{0\cdots 0\}$. Then, $\forall \mathfrak{u}\in\mathcal{D}, \mathfrak{u}\neq\varnothing$, we have $V_{\mathfrak{u}} \approx \sum_{\boldsymbol{k}\in\mathcal{K}_{\mathfrak{u}}} f_{\boldsymbol{k}}^2$, while the variance of F is approximated by

$$\mathbb{V}\{F\} \approx \sum_{\boldsymbol{k}\in\mathcal{K}\setminus\mathcal{K}_{\varnothing}} f_{\boldsymbol{k}}^2.$$

The approximations for the first and total-order sensitivity indices $\mathtt{S}_{\mathfrak{u}}$ and $\mathtt{T}_{\mathfrak{u}}$ can be easily derived through (B.4), by taking the corresponding unions of subsets $\mathcal{K}_{\mathfrak{v}}$. For instance, in the case of singleton $\mathfrak{u} = \{i\}$, we have

$$\mathtt{S}_{\{i\}} \approx \frac{1}{\mathbb{V}\{F\}} \sum_{\boldsymbol{k}\in\mathcal{K}^{\mathtt{S}}_{\{i\}}} f_{\boldsymbol{k}}^2, \qquad \mathcal{K}^{\mathtt{S}}_{\{i\}} \doteq \{\boldsymbol{k}\in\mathcal{K}; k_i > 0, k_{j\neq i} = 0\}, \tag{B.5}$$

$$\mathtt{T}_{\{i\}} \approx \frac{1}{\mathbb{V}\{F\}} \sum_{\boldsymbol{k}\in\mathcal{K}^{\mathtt{T}}_{\{i\}}} f_{\boldsymbol{k}}^2, \qquad \mathcal{K}^{\mathtt{T}}_{\{i\}} \doteq \{\boldsymbol{k}\in\mathcal{K}; k_i > 0\}. \tag{B.6}$$

The subsets of PC multi-indices for first and total-order sensitivity indices are illustrated in Fig. B.1 for a simple example with $d = 2$.

B.3 Monte Carlo Estimates of Sobol Indices

For validation purposes, we need to estimate the sensitivity indices of functions F without an available PC expansion. For these estimations we rely on the Monte Carlo method proposed in [118].

Consider two independent random sample sets $\boldsymbol{x}^I$ and $\boldsymbol{x}^{II}$ of M realizations of $\boldsymbol{\xi}$:

$$\boldsymbol{x}^I = \{\boldsymbol{\xi}^{I,(i)}, i = 1,\cdots,M\}, \quad \boldsymbol{x}^{II} = \{\boldsymbol{\xi}^{II,(i)}, i = 1,\cdots,M\}.$$

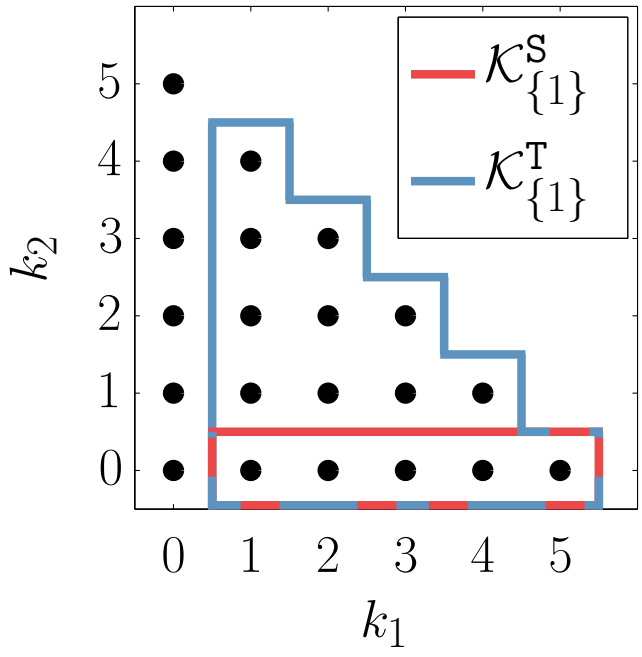


FIGURE B.1: Schematic example in $d = 2$ dimensions of the subsets of PC multi-indices $\mathcal{K}^{\mathtt{S}}_{\{i\}}$ and $\mathcal{K}^{\mathtt{T}}_{\{1\}}$ for the first-order (red) and total-order (blue) Sobol-sensitivities of ξ_1. The points represent the muti-index $\boldsymbol{k} \in \mathcal{K}$.

The computation of the first-order sensitivity index $\mathtt{S}_{\mathfrak{u}}$ amounts to the evaluation of the conditional variance $\mathbb{V}\{F \mid \boldsymbol{\xi}_{\mathfrak{u}}\}$, which can be estimated from the following average [118]

$$\mathbb{V}\{F \mid \boldsymbol{\xi}_{\mathfrak{u}}\} + \mathbb{E}\{F\}^2 = \lim_{M\to\infty} \frac{1}{M} \sum_{i=1}^{M} F(\boldsymbol{\xi}_{\mathfrak{u}}^{I,(i)} \boldsymbol{\xi}_{\mathfrak{u}_\sim}^{I,(i)}) F(\boldsymbol{\xi}_{\mathfrak{u}}^{I,(i)} \boldsymbol{\xi}_{\mathfrak{u}_\sim}^{II,(i)}). \tag{B.7}$$

Introducing the classical ($\boldsymbol{x}^I$-)sample estimator for the mean and variance of F,

$$\widehat{\mathbb{E}\{F\}} \approx \frac{1}{M} \sum_{i=1}^{M} F(\boldsymbol{\xi}^{I,(i)}), \quad \widehat{\mathbb{V}\{F\}} = \frac{1}{M-1} \sum_{i=1}^{M} \left(F(\boldsymbol{\xi}^{I,(i)}) - \widehat{\mathbb{E}\{F\}} \right)^2,$$

the Monte Carlo estimator of the first-order sensitivity index is

$$\widehat{\mathtt{S}}_{\mathfrak{u}} = \frac{1}{\widehat{\mathbb{V}\{F\}}} \left(\frac{1}{M} \sum_{i=1}^{M} F(\boldsymbol{\xi}_{\mathfrak{u}}^{I,(i)} \boldsymbol{\xi}_{\mathfrak{u}_\sim}^{I,(i)}) F(\boldsymbol{\xi}_{\mathfrak{u}}^{I,(i)} \boldsymbol{\xi}_{\mathfrak{u}_\sim}^{II,(i)}) - \widehat{\mathbb{E}\{F\}}^2 \right).$$

A similar expression can be derived for the estimation of the total-order sensitivity index $\mathtt{T}_{\mathbf{u}}$, by using the relation $\mathtt{T}_{\mathbf{u}} = 1 - \mathtt{S}_{\mathbf{u}_\sim}$ [58]. This results in

$$\widehat{\mathtt{T}}_{\mathbf{u}} = 1 - \frac{1}{\widehat{\mathbb{V}\{F\}}} \left(\frac{1}{M} \sum_{i=1}^{M} F(\boldsymbol{\xi}_{\mathbf{u}}^{I,(i)} \boldsymbol{\xi}_{\mathbf{u}_\sim}^{I,(i)}) F(\boldsymbol{\xi}_{\mathbf{u}}^{II,(i)} \boldsymbol{\xi}_{\mathbf{u}_\sim}^{I,(i)}) - \widehat{\mathbb{E}\{F\}}^2 \right).$$

It is seen that the MC evaluation of the sensitivity indices requires M evaluations of F (for the elements of $\boldsymbol{x}^I$), and M new function evaluations for each $\mathtt{S}_{\mathbf{u}}$ or $\mathtt{T}_{\mathbf{u}}$. In particular, the MC estimation of the first and total-order sensitivity indices of the d input variable requires a total of $(2d+1)M$ model evaluations.

Appendix C

Hyperbolic Index Sets

C.1 Anisotropic Index Sets

Traditionally a PCE is truncated by the *total order* of the expansion [49, 76] such that, the polynomial multi-index $\mathcal{K}$, with d dimensions, may be represented by

$$\mathcal{K}^p = \left\{ \boldsymbol{k} \in \mathbb{N}_0^d \middle| \sum_{i=1}^{d} k_i \leqslant p \right\}.$$

For such a truncation, the number of terms in the expansion is given by [49]

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

The total order truncation is isotropic and includes all mixed terms such that the *total* order of the expansion is $\leqslant p$.

Anisotropy may be induced through the use of a weighting summation of the form

$$\mathcal{K}_{\boldsymbol{v}}^p = \left\{ \boldsymbol{k} \in \mathbb{N}_0^d \middle| \sum_{i=1}^{d} v_i k_i \leqslant p \right\} \tag{C.1}$$

where $v_i > 0$. However, such a formulation removes the notion of the total-order for particular $v_i \geqslant 1$ or $v_i < 1$. An alternative is based on the following index sets:

$$\mathcal{K}^{\boldsymbol{p}} = \left\{ \boldsymbol{k} \in \mathbb{N}_0^d | \sum_{i=1}^{d} \frac{k_i}{p_i + \epsilon} \leqslant 1 \right\} \tag{C.2}$$

where $\epsilon \sim 10^{-5}$ allows for index-sets with $p_i = 0$. When all $p_i = p$, the form of $\mathcal{K}^{\boldsymbol{p}}$ immediately reduces to the isotropic, total-order truncation. In each direction p_i represents the maximal term in the i^{th} direction. For a PCE, p_i represents the highest monomial term in the i^{th} direction.

Schematic examples of isotropic and anisotropic index-sets are in Fig. C.1(top row) and Fig. C.2(a).

By construction, $\mathcal{K}^{\boldsymbol{p}}$ begins at zero (representing the constant term of a PCE) however, it may be extended to begin at any integer β as follows:

$$\mathcal{K}_{\beta}^{\boldsymbol{p}} = \left\{ \boldsymbol{k} \in \mathbb{N}_{\beta}^d | \sum_{i=1}^{d} \frac{k_i - \beta}{p_i + \epsilon - \beta} \leqslant 1 \right\}$$

Such a representation may be used to define the index set for Smolyak quadrature (see Section 2.3.1) or aPSP (see Section 3.3.1).

C.2 Hyperbolic Anisotropic Index Sets

For isotropic index-sets, mixed terms up to a total order are all included. However, it may be advantageous to limit the inclusion of mixed terms; especially when high-order monomial terms are desired.

In [7] an anisotropic, hyperbolic truncation is suggested again using a weighted summation like that of Equation (C.1). However, we again prefer a form that is weighted by the maximal order of the index set as in Equation (C.2). We implement

an anisotropic truncation strategy with the *q-norm* of the index as follows:

$$\mathcal{K}^{\boldsymbol{p},q} = \left\{ \boldsymbol{k} \in \mathbb{N}_0^d \middle| \left(\sum_{i=1}^{d} \left(\frac{k_i}{p_i + \epsilon} \right)^q \right)^{1/q} \leqslant 1 \right\} \tag{C.3}$$

Again, $\mathcal{K}^{\boldsymbol{p},q}$ may be extended to begin at integer β

$$\mathcal{K}_\beta^{\boldsymbol{p},q} = \left\{ \boldsymbol{k} \in \mathbb{N}_\beta^d \middle| \left(\sum_{i=1}^{d} \left(\frac{k_i - \beta}{p_i + \epsilon - \beta} \right)^q \right)^{1/q} \leqslant 1 \right\}. \tag{C.4}$$

When $q = 1$, the original form of $\mathcal{K}^{\boldsymbol{p}}$ is recovered and when $q = 1$ and all $p_i = p$, the traditional total-order truncation is again recovered.

Furthermore, a $q > 1$ value may be used (though no longer "hyperbolic") to admit many more mixed terms. As $q \to \infty$, a "full-tensor" index-set is recovered.

Schematic examples of isotropic and anisotropic index-sets with different q values are in Fig. C.1 for two dimensions. In Fig. C.2 are *isotropic* hyperbolic sets with different q values for three dimensions. Figure C.2(d) demonstrates how a low q value may severely restrict mixed terms as no index with $k > 0$ in all directions is added.

The exclusion of mixed terms greatly reduces the final size of the basis as seen in Fig. C.3 where, especially for higher dimensions, the cardinality of the basis becomes prohibitively large.

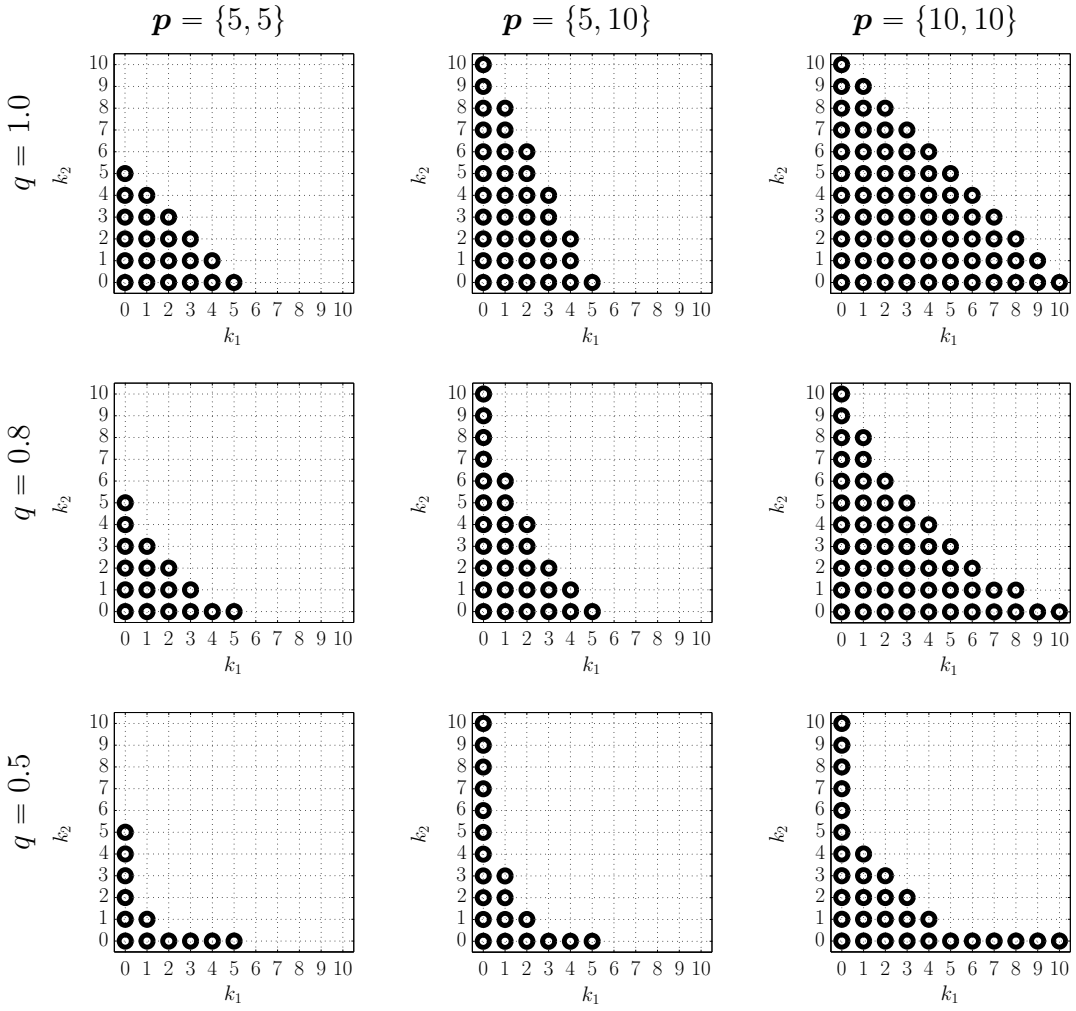
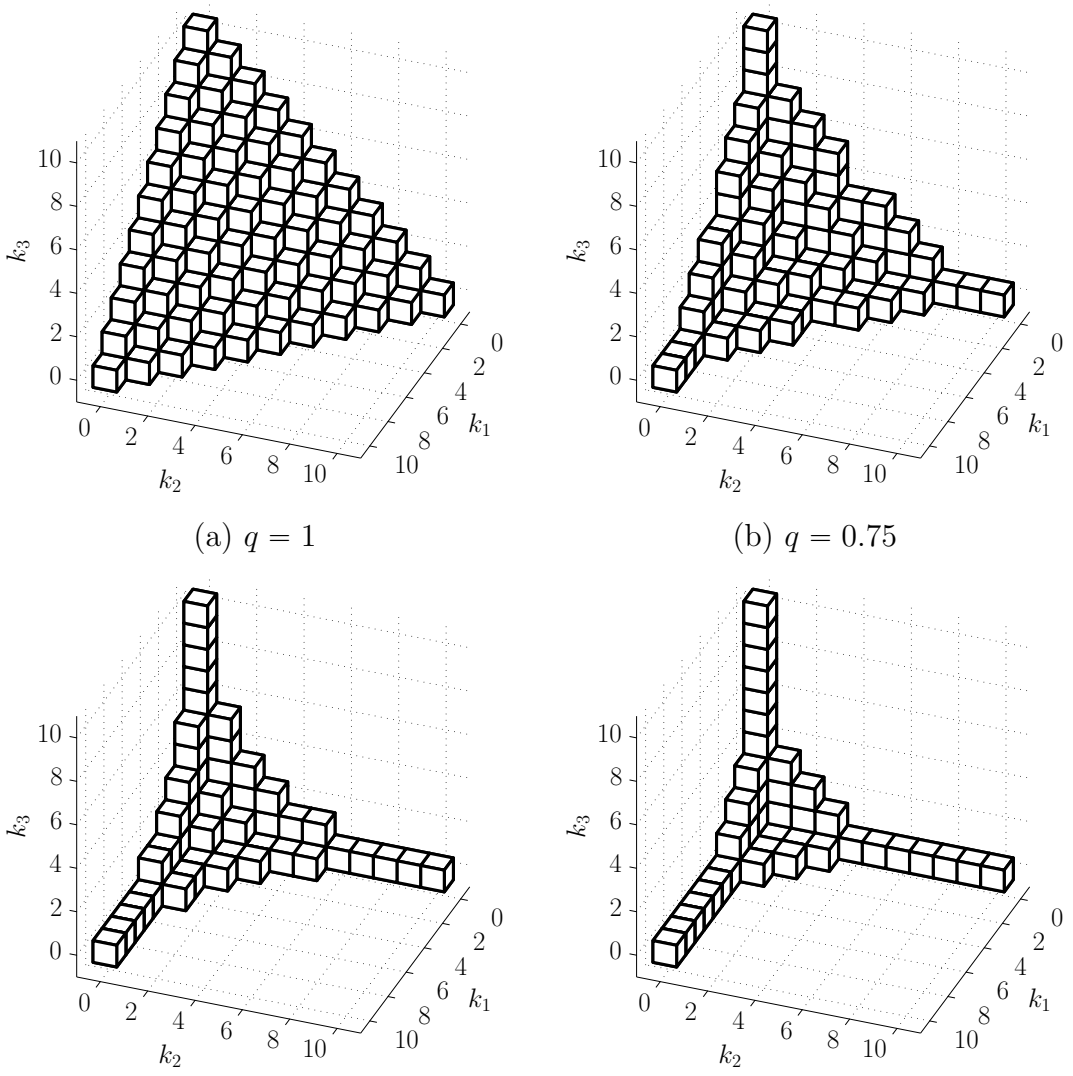


FIGURE C.1: Schematic example of hyperbolic truncated index-sets. Top row with $q = 1.0$ represents traditional, non-hyperbolic truncations. The first and third column represent isotropic index-sets. The middle column represents anisotropic truncations.



(a) $q = 1$ (b) $q = 0.75$

(c) $q = 0.55$ (d) $q = 0.45$

FIGURE C.2: Schematic demonstration of hyperbolic index-set truncation for $\boldsymbol{p} = \{10, 10, 10\}$ and $d = 3$. Notice in (d), mixed-indices are admitted between any two sets of inputs but are not present for a combination of all three directions.

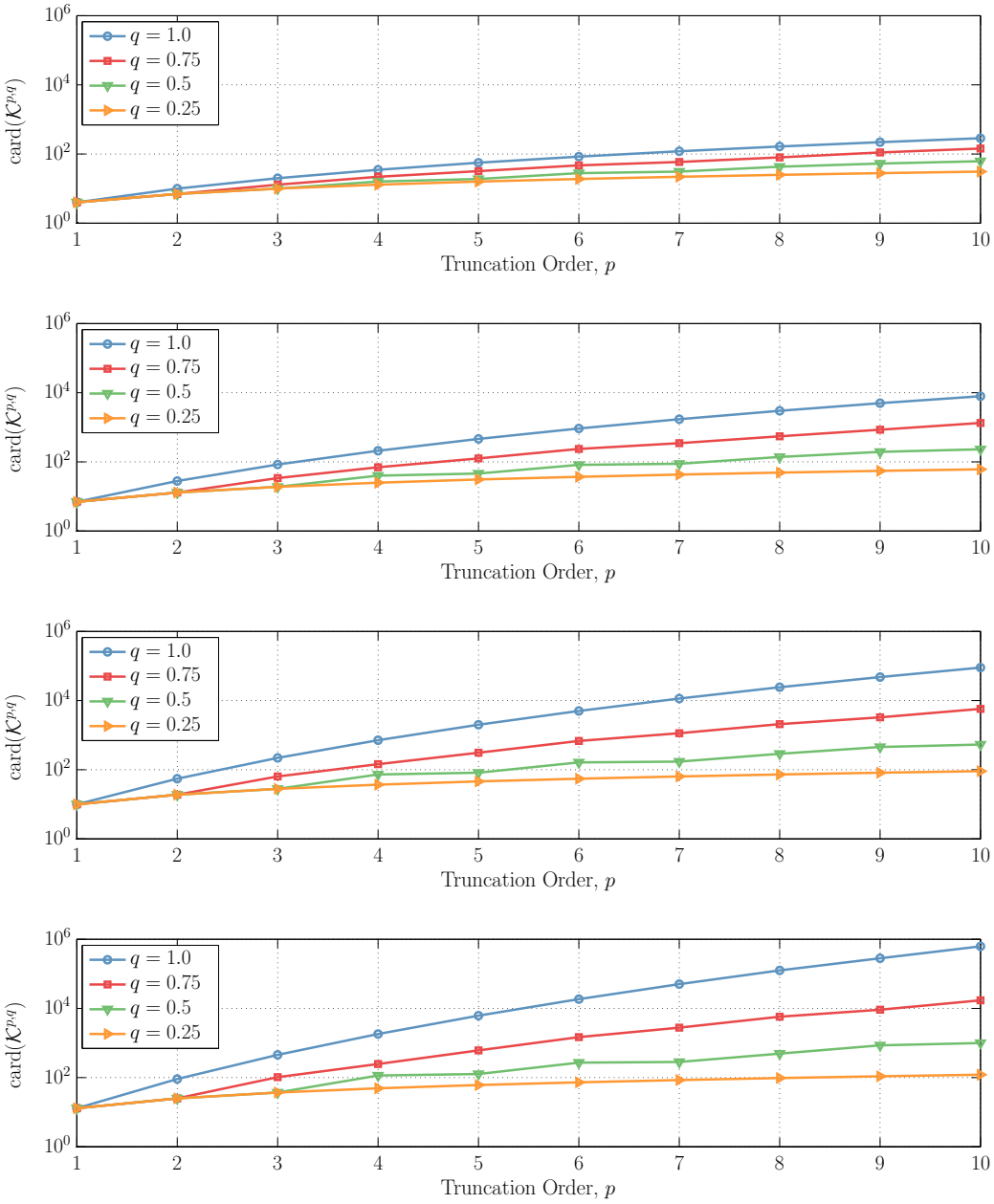


FIGURE C.3: Cardinality of resulting hyperbolic index-set for varying (isotropic) order p, and $q-norm$ values. Presented on a semi-logarithmic scale.

Bibliography

[1] Milton Abramowitz and Irene A Stegun. Handbook of Mathematical Functions: With Formulas, Graphs, and Mathematical Tables. 55. Courier Dover Publications, 1972.

[2] A. Alexanderian, J. Winokur, I. Sraj, A. Srinivasan, M. Iskandarani, W.C. Thacker, and O.M. Knio. Global sensitivity analysis in an ocean general circulation model: a sparse spectral projection approach. Computational Geosciences, 16(3):1–22, 2012.

[3] Alen Alexanderian. On spectral methods for variance based sensitivity analysis. Probability Surveys, 10:51–68, 2013.

[4] B.K. Alpert. A Class of Bases in L^2 for the Sparse Representation of Integral Operators. SIAM journal on Mathematical Analysis, 24(1):246–262, 1993.

[5] Richard Askey and James Arthur Wilson. Some Basic Hypergeometric Orthogonal Polynomials That Generalize Jacobi Polynomials, volume 319. American Mathematical Soc., 1985.

[6] Marc Berveiller, Bruno Sudret, and Maurice Lemaire. Stochastic finite element: a non intrusive approach by regression. European Journal of Computational Mechanics/Revue Européenne de Mécanique Numérique, 15(1-3):81–92, 2006.

[7] Géraud Blatman. Adaptive sparse polynomial chaos expansions for uncertainty propagation and sensitivity analysis. PhD thesis, Clermont-Ferrand 2, 2009.

[8] Géraud Blatman and Bruno Sudret. Sparse Polynomial Chaos Expansions and Adaptive Stochastic Finite Elements Using a Regression Approach. Comptes Rendus Mécanique, 336(6):518–523, 2008.

[9] Géraud Blatman and Bruno Sudret. Adaptive sparse polynomial chaos expansion based on *least angle regression*. Journal of Computational Physics, 230(6):2345–2367, 2011.

[10] Rainer Bleck. An oceanic general circulation model framed in hybrid isopycnic-cartesian coordinates. Ocean modelling, 4(1):55–88, 2002.

[11] Leo Breiman and Philip Spector. Submodel Selection and Evaluation in Regression. the X-Random Case. International statistical review/revue internationale de Statistique, pages 291–319, 1992.

[12] A.M. Bruckstein, D.L. Donoho, and M. Elad. From Sparse Solutions of Systems of Equations to Sparse Modeling of Signals and Images. SIAM review, 51(1):34, 2009.

[13] L. Brutman. On the Lebesgue Function for Polynomial Interpolation. SIAM Journal on Numerical Analysis, 15(4):pp. 694–704, 1978.

[14] John Burkardt. Gauss Patterson Quadrature Rules. `http://people.sc.fsu.edu/~jburkardt/f_src/patterson_rule/patterson_rule.html`, February 2010.

[15] E.J. Candès. Compressive sampling. In Proceedings oh the International Congress of Mathematicians: Madrid, August 22-30, 2006: invited lectures, pages 1433–1452, 2006.

[16] Emmanuel Candes and Justin Romberg. ℓ_1-Magic: Recovery of Sparse Signals via Convex Programming. URL: www. acm. caltech. edu/l1magic/downloads/l1magic. pdf, 4, 2005.

[17] Kathryn Chaloner and Isabella Verdinelli. Bayesian Experimental Design: A Review. Statistical Science, 10(3):pp. 273–304, 1995.

[18] Olivier Chapelle, Vladimir Vapnik, and Yoshua Bengio. Model Selection for Small Sample Regression. Machine Learning, 48(1-3):9–23, 2002.

[19] Eric P Chassignet, Harley E Hurlburt, Ole Martin Smedstad, George R Halliwell, Patrick J Hogan, Alan J Wallcraft, Remy Baraille, and Rainer Bleck. The hycom (hybrid coordinate ocean model) data assimilative system. Journal of Marine Systems, 65(1):60–83, 2007.

[20] Eric P. Chassignet and Jacques Verron, editors. Ocean Modeling and Parameterization, volume 516 of C: Mathematical and Physical Sciences. Kluwer Academic Publishers, 1998.

[21] Charles W Clenshaw and Alan R Curtis. A method for numerical integration on an automatic computer. Numerische Mathematik, 2(1):197–205, 1960.

[22] Patrick R Conrad and Youssef M Marzouk. Adaptive Smolyak Pseudospectral Approximations. SIAM Journal on Scientific Computing, 35(6):A2643–A2670, 2013.

[23] Paul G Constantine. A Quick-and-Dirty Check for a One-Dimensional Active Subspace. arXiv preprint arXiv:1402.3838, 2014.

[24] Paul G. Constantine, Eric Dow, and Qiqi Wang. Active Subspace Methods in Theory and Practice: Applications to Kriging Surfaces. SIAM Journal on Scientific Computing, 36(4):A.1500–A.1524, 2014.

[25] P.G. Constantine, M.S. Eldred, and E.T. Phipps. Sparse pseudospectral approximation method. Computer Methods in Applied Mechanics and Engineering, 2012.

[26] T. Crestaux, O.P Le Maître, and J.M. Martinez. Polynomial chaos expansion for sensitivity analysis. Reliability Engineering & System Safety, 94(7):1161–1172, 2009.

[27] Mark A Davenport, Marco F Duarte, Yonina C Eldar, and Gitta Kutyniok. Introduction to Compressed Sensing. Preprint, 93, 2011.

[28] Stefano De Marchi. On Leja sequences: some results and applications. Applied mathematics and computation, 152(3):621–647, 2004.

[29] Bert J Debusschere, Habib N Najm, Alain Matta, Omar M Knio, Roger G Ghanem, and Olivier P Le Maître. Protein labeling reactions in electrochemical microchannel flow: Numerical simulation and uncertainty propagation. Physics of Fluids (1994-present), 15(8):2238–2250, 2003.

[30] Bert J Debusschere, Habib N Najm, Philippe P Pébay, Omar M Knio, Roger G Ghanem, and Olivier P Le Maître. Numerical Challenges in the Use of Polynomial Chaos Representations for Stochastic Processes. SIAM Journal on Scientific Computing, 26(2):698–719, 2004.

[31] David L Donoho. For Most Large Underdetermined Systems of Linear Equations the Minimal ℓ_1-Norm Solution Is Also the Sparsest Solution. Communications on pure and applied mathematics, 59(6):797–829, 2006.

[32] David L Donoho and Michael Elad. Optimally sparse representation in general (nonorthogonal) dictionaries via ℓ^1 minimization. Proceedings of the National Academy of Sciences, 100(5):2197–2202, 2003.

[33] David L Donoho, V Stodden, and Y Tsaig. Sparse-Lab: Sparse Solution of Underdetermined Linear Systems of Equations. Technical report, Tech. Rep., Stanford University, http://sparselab. stanford. edu, 2007.

[34] D.L. Donoho. Compressed Sensing. Information Theory, IEEE Transactions on, 52(4):1289–1306, 2006.

[35] A. Doostan and H. Owhadi. A non-adapted sparse approximation of pdes with stochastic inputs. Journal of Computational Physics, 230(8):3015–3034, 2011.

[36] Bradley Efron, Trevor Hastie, Iain Johnstone, Robert Tibshirani, et al. Least Angle Regression. The Annals of statistics, 32(2):407–499, 2004.

[37] Martin Eigel, Claude Jeffrey Gittelson, Christoph Schwab, and Elmar Zander. Adaptive stochastic Galerkin FEM. Computer Methods in Applied Mechanics and Engineering, 270:247–269, 2014.

[38] Michael S Eldred. Design Under Uncertainty Employing Stochastic Expansion Methods. International Journal for Uncertainty Quantification, 1(2), 2011.

[39] M.S. Eldred and John Burkardt. Comparison of Non-Intrusive Polynomial Chaos and Stochastic Collocation Methods for Uncertainty Quantification. AIAA paper, 976(2009):1–20, 2009.

[40] M.S. Eldred, Laura Painton Swiler, and Gary Tang. Mixed aleatory-epistemic uncertainty quantification with stochastic expansions and optimization-based interval estimation. Reliability Engineering & System Safety, 96(9):1092–1113, 2011.

[41] Jordan Ellenberg. Fill in the Blanks: Using Math to Turn Lo-Res Datasets Into Hi-Res Samples. WIRED Magazine, March 2010.

[42] James F Epperson. On the Runge Example. Amer. Math. Monthly, 94(4):329–341, 1987.

[43] Leopold Fejér. Mechanische Quadraturen mit positiven Cotesschen Zahlen. Mathematische Zeitschrift, 37(1):287–309, 1933.

[44] Jerome H Friedman. Multivariate Adaptive Regression Splines. The annals of statistics, pages 1–67, 1991.

[45] A. C. Genz and A. A. Malik. Remarks on algorithm 006: An adaptive algorithm for numerical integration over an N-dimensional rectangular region. Journal of Computational and Applied Mathematics, 6(4):295–302, 1980.

[46] T Gerstner and M Griebel. Dimension-Adaptive Tensor-Product Quadrature. Computing, 71(1):65–87, 2003.

[47] Thomas Gerstner and Michael Griebel. Numerical Integration Using Sparse Grids. Numerical Algorithms, pages 209–232, 1998.

[48] R Ghanem, J Red-Horse, and A Sarkar. Modal Properties of a Space-Frame With Localized System Uncertainties. In 8th ASCE Specialty Conference of Probabilistic Mechanics and Structural Reliability, No. PMC200-269, 2000.

[49] Roger G Ghanem and Pol D Spanos. Stochastic Finite Elements: A Spectral Approach. Springer-Verlag New York, Inc., 1991.

[50] R Günttner. Evaluation of Lebesgue Constants. SIAM Journal on Numerical Analysis, 17(4):512–520, 1980.

[51] Stuttgat H.-J. Bungartz and Schnaittenbach S. Dirnstorfer. Multivariate Quadrature on Adaptive Sparse Grids. Computing, 71(1):89–114, 2003.

[52] George R Halliwell. Evaluation of vertical coordinate and vertical mixing algorithms in the HYbrid-Coordinate Ocean Model (HYCOM). Ocean Modelling, 7(3):285–322, 2004.

[53] Jerrad Hampton and Alireza Doostan. Compressive Sampling of Polynomial Chaos Expansions: Convergence Analysis and Sampling Strategies. arXiv preprint arXiv:1408.4157, 2014.

[54] Lihan He and Lawrence Carin. Exploiting Structure in Wavelet-Based Bayesian Compressive Sensing. Signal Processing, IEEE Transactions on, 57(9):3488–3497, 2009.

[55] Tran Duong Hien and Michał Kleiber. Stochastic finite element modelling in linear transient heat transfer. Computer Methods in Applied Mechanics and Engineering, 144(1):111–124, 1997.

[56] Ronald R Hocking. A Biometrics Invited Paper. the Analysis and Selection of Variables in Linear Regression. Biometrics, pages 1–49, 1976.

[57] W. Hoeffding. A Class of Statistics With Asymptotically Normal Distribution. The annals of Mathematical Statistics, 19:293–325, 1948.

[58] Toshimitsu Homma and Andrea Saltelli. Importance Measures in Global Sensitivity Analysis of Nonlinear Models. Reliability Engineering & System Safety, 52(1):1–17, 1996.

[59] Serhat Hosder, Robert W Walters, and Michael Balch. Efficient Sampling for Non-Intrusive Polynomial Chaos Applications With Multiple Uncertain Input Variables. In Proceedings of the 48th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference, AIAA paper, volume 1939, 2007.

[60] Jingwei Hu, Shi Jin, and Dongbin Xiu. A stochastic Galerkin method for Hamilton-Jacobi equations with uncertainty. 2014.

[61] Xun Huan and Youssef M Marzouk. Simulation-based optimal Bayesian experimental design for nonlinear systems. Journal of Computational Physics, 2012.

[62] Matt J. "How to remove dependent rows in a matrix?". http://www.mathworks.com/matlabcentral/answers/49984-how-to-remove-dependent-rows-in-a-matrix.

[63] J.D Jakeman and S.G Roberts. Local and Dimension Adaptive Stochastic Collocation for Uncertainty Quantification. In Jochen Garcke and Michael Griebel, editors, Sparse Grids and Applications, number 88 in Lecture Notes in Computational Science and Engineering, pages 181–203. Springer, 2013.

[64] John D Jakeman, Michael S Eldred, and Khachik Sargsyan. Enhancing ℓ_1-minimization estimates of polynomial chaos expansions using basis selection. arXiv preprint arXiv:1407.8093, 2014.

[65] Shihao Ji, Ya Xue, and Lawrence Carin. Bayesian Compressive Sensing. Signal Processing, IEEE Transactions on, 56(6):2346–2356, 2008.

[66] A Birol Kara, Peter A Rochford, and Harley E Hurlburt. Efficient and Accurate Bulk Parameterizations of Air-Sea Fluxes for Use in General Circulation Models. Journal of Atmospheric and Oceanic Technology, 17(10):1421–1438, 2000.

[67] A. Keese and H.G. Matthies. Numerical Methods and Smolyak Quadrature for Nonlinear Stochastic Partial Differential Equations. Technical report, Institute of Scientific Computing TU Braunschweig Brunswick, 2003.

[68] V. Keshavarzzadeh, R.G. Ghanem, S.F. Masri, and O.J. Aldraihem. Convergence acceleration of polynomial chaos solutions via sequence transformation. Computer Methods in Applied Mechanics and Engineering, 271:167–184, 2014.

[69] D. Kim, F. Rizzi, K.-W. Cheng, J. Han, F. Bisetti, and O. Knio. Manuscript in preparation, 2014.

[70] Sung Soo Kim. "LARS algorithm". Matlab Central: http://www.mathworks.com/matlabcentral/fileexchange/23186-lars-algorithm, March 2009.

[71] Omar M Knio, Habib N Najm, Roger G Ghanem, et al. A Stochastic Projection Method for Fluid Flow: I. Basic Formulation. Journal of computational Physics, 173(2):481–511, 2001.

[72] Ron Kohavi et al. A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. In IJCAI, volume 14, pages 1137–1145, 1995.

[73] Aleksandr Semenovich Kronrod. Nodes and Weights of Quadrature Formulas: Sixteen-Place Tables. Consultants Bureau New York, 1965.

[74] Cristian Constantin Lalescu. Two hierarchies of spline interpolations. Practical algorithms for multivariate higher order splines. arXiv preprint arXiv:0905.3564, 2009.

[75] W. G. Large, J. C. McWilliams, and S. C. Dooney. Oceanic Vertical Mixing: A Reciew and a Model with Nonlocal Boundary Layer Parameterization. Reviews of Geophysics, 32(4):363–403, 1994.

[76] O. P. Le Maître and Omar Knio. Spectral Methods for Uncertainty Quantification: With Applications to Computational Fluid Dynamics. Scientific Computation. Springer, 2010.

[77] Olivier P Le Maître, Habib N Najm, Philippe P Pébay, Roger G Ghanem, and Omar M Knio. Multi-Resolution-Analysis Scheme for Uncertainty Quantification in Chemical Systems. SIAM Journal on Scientific Computing, 29(2):864–889, 2007.

[78] O.P. Le Maître and O.M. Knio. PC analysis of stochastic differential equations driven by Wiener noise. Reliability Engineering & System Safety, 135:107–124, 2015.

[79] O.P. Le Maître, O.M. Knio, H.N. Najm, and R.G. Ghanem. Uncertainty propagation using Wiener–Haar expansions. Journal of Computational Physics, 197(1):28–57, 2004.

[80] O.P. Le Maître, H.N. Najm, R.G. Ghanem, and O. M Knio. Multi-resolution analysis of Wiener-type uncertainty propagation schemes. Journal of Computational Physics, 197(2):502–531, 2004.

[81] O.P. Le Maître, M.T. Reagan, H.N. Najm, R.G. Ghanem, and O.M. Knio. A Stochastic Projection Method for Fluid Flow. II. Random Process. Journal of Computational Phyics, 181:9–44, 2002.

[82] F. Leja. Sur certaines suites liées aux ensembles plans er leur application à la représentation conforme. Ann. Polon. Math, 4:8–13, 1957.

[83] Xianjin Li, Yi Chao, James C McWilliams, and Lee-Lueng Fu. A Comparison of Two Vertical-Mixing Schemes in a Pacific Ocean General Circulation Model. Journal of Climate, 14(7):1377–1398, 2001.

[84] Guang Lin, Alexandre M Tartakovsky, and Daniel M Tartakovsky. Uncertainty quantification via random domain decomposition and probabilistic collocation on sparse grids. Journal of Computational Physics, 229(19):6995–7012, 2010.

[85] Yi Lin and Hao Helen Zhang. On a Measure of the Information Provided by an Experiment. Annals of Statistics, 34(5):2272–2297, 2006.

[86] Dennis V Lindley. On a Measure of the Information Provided by an Experiment. The Annals of Mathematical Statistics, pages 986–1005, 1956.

[87] Ning Liu, Bin Hu, and Zhi-Wen Yu. Stochastic finite element method for random temperature in concrete structures. International journal of solids and structures, 38(38):6965–6983, 2001.

[88] Xiang Ma and Nicholas Zabaras. An adaptive hierarchical sparse grid collocation algorithm for the solution of stochastic differential equations. Journal of Computational Physics, 228(8):3084–3113, 2009.

[89] Y.M. Marzouk and H.N. Najm. Dimensionality Reduction and Polynomial Chaos Acceleration of Bayesian Inference in Inverse Problems. Journal of Computational Physics, 228(6):1862–1902, 2009.

[90] Y.M. Marzouk, H.N. Najm, and L.A. Rahn. Stochastic spectral methods for efficient Bayesian solution of inverse problems. J. Comput. Phys., 224:560–586, 2007.

[91] Hermann G Matthies, Christoph E Brenner, Christian G Bucher, and C Guedes Soares. Uncertainties in probabilistic numerical analysis of structures and solids-stochastic finite elements. Structural safety, 19(3):283–336, 1997.

[92] Michael D McKay, Richard J Beckman, and William J Conover. Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output From a Computer Code. Technometrics, 21(2):239–245, 1979.

[93] Daniel R Millman, Paul I King, Raymond C Maple, Philip S Beran, and Lawrence K Chilton. Uncertainty Quantification With a B-Spline Stochastic Projection. AIAA journal, 44(8):1845–1853, 2006.

[94] H.N. Najm. Uncertainty Quantification and Polynomial Chaos Techniques in Computational Fluid Dynamics. Ann. Rev. Fluid Mech., 41:35–52, 2009.

[95] Fabio Nobile, Raúl Tempone, and Clayton G Webster. A Sparse Grid Stochastic Collocation Method for Partial Differential Equations With Random Input Data. SIAM Journal on Numerical Analysis, 46(5):2309–2345, 2008.

[96] Fabio Nobile, Raul Tempone, and Clayton G Webster. An Anisotropic Sparse Grid Stochastic Collocation Method for Partial Differential Equations With Random Input Data. SIAM Journal on Numerical Analysis, 46(5):2411–2442, 2008.

[97] Erich Novak and Klaus Ritter. The Curse of Dimension and a Universal Method for Numerical Integration. In Multivariate approximation and splines, pages 177–187. Springer, 1997.

[98] T.N.L. Patterson. The Optimum Addition of Points to Quadrature Formulae. Math. Comput, 22:847–856, 1968.

[99] Ji Peng, Jerrad Hampton, and Alireza Doostan. A weighted ℓ_1-minimization approach for sparse polynomial chaos expansions. Journal of Computational Physics, 267:92–111, 2014.

[100] J Prager, U Riedel, and J Warnatz. Modeling Ion Chemistry and Charged Species Diffusion in Lean Methane–Oxygen Flames. Proceedings of the Combustion Institute, 31(1):1129–1137, 2007.

[101] D.A. Quadling. Lagrange's Interpolation Formula. The Mathematical Gazette, pages 372–375, 1966.

[102] M.T. Reagan, H.N. Najm, R.G. Ghanem, and O.M. Knio. Uncertainty Quantification in Reacting Flow Simulations Through Non-Intrusive Spectral Projection. Combustion and Flame, 132:545–555, 2003.

[103] Michael Revers. Approximation Constants in Equidistant Lagrange Interpolation. Periodica Mathematica Hungarica, 40(2):167–175, 2000.

[104] F. Rizzi, H.N. Najm, B.J. Debusschere, K. Sargsyan, M. Salloum, H. Adalsteinsson, and O.M. Knio. Uncertainty Quantification in MD Simulations. Part I: Forward Propagation. Multiscale Modeling & Simulation, 10(4):1428–1459, 2012.

[105] F. Rizzi, H.N. Najm, B.J. Debusschere, K. Sargsyan, M. Salloum, H. Adalsteinsson, and O.M. Knio. Uncertainty Quantification in MD Simulations. Part II: Bayesian Inference of Force-Field Parameters. Multiscale Modeling & Simulation, 10(4):1460–1492, 2012.

[106] C. Runge. Über empirische Funktionen und die Interpolation zwischen äquidistanten Ordinaten, Zeit. Math. Physik, 246, 1901.

[107] C. Safta. private communication, 2013.

[108] Cosmin Safta, Habib Najm, and O.M. Knio. TChem. http://www.sandia.gov/tchem/.

[109] Thomas Sauer and Yuan Xu. On Multivariate Lagrange Interpolation. Mathematics of Computation, 64(211):1147–1170, 1995.

[110] Paola Sebastiani and Henry P Wynn. Maximum Entropy Sampling and Optimal Bayesian Experimental Design. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 62(1):145–157, 2000.

[111] Jun Shao. Linear Model Selection by Cross-Validation. Journal of the American statistical Association, 88(422):486–494, 1993.

[112] Ritei Shibata. An optimal selection of regression variables. Biometrika, 68(1):45–54, 1981.

[113] Karl Sjöstrand, Line Harder Clemmensen, Rasmus Larsen, and Bjarne Ersbøll. SpaSM: A Matlab Toolbox for Sparse Statistical Modeling. Journal of Statistical Software, 2012 [in press].

[114] Gregory P. Smith, David M. Golden, Michael Frenklach, Nigel W. Moriarty, Boris Eiteneer, Mikhail Goldenberg, C. Thomas Bowman, Ronald K. Hanson, Soonho Song, William C. Gardiner Jr., Vitali V. Lissianski, and Zhiwei Qin. GRI-Mech 3.0. `http://www.me.berkeley.edu/gri_mech/`, 2012.

[115] Raph C. Smith. Uncertainty Quantification Theory, Implementation, and Applications. Society for Industrial and Applied Mathematics, 2014.

[116] S.A. Smolyak. Quadrature and Interpolation Formulas for Tensor Products of Certain Classes of Functions. Dokl. Akad. Nauk SSSR, 4(240-243):123, 1963.

[117] Ilya M Sobol. Sensitivity Estimates for Nonlinear Mathematical Models. Math. Model. Comput. Exp., 1:407–414, 1993.

[118] I.M. Sobol. Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. Math. Comput. in Simul., 55:271–280, 2001.

[119] I. Sraj, M. Iskandarani, W. Carlisle Thacker, A. Srinivasan, and O.M. Knio. Drag Parameter Estimation Using Gradients and Hessian From a Polynomial Chaos Model Surrogate. Monthly Weather Review, 142(2):933–941, 2014.

[120] I. Sraj, M. Iskandarani, A. Srinivasan, W. C. Thacker, J. Winokur, A. Alexanderian, C. Y. Lee, S. S. Chen, and O. M. Knio. Bayesian Inference of Drag Parameters Using AXBT Data from Typhoon Fanapi. Monthly Weather Review, 141(7):2347–2367, 2013.

[121] Mohamed Iskandarani Ashwanth Srinivasan, Carlisle Thacker, Chia-Ying Lee, Shuyi Chen, Omar Knio, Alen Alexandrian, Justin Winokur, and Ihab Sraj. Combining HYCOM, AXBTs and Polynomial Chaos Methods to Estimate Wind Drag Parameters during Typhoon Fanapi. 2013.

[122] B. Sudret. Global Sensitivity Analysis Using Polynomial Chaos Expansions. Reliability Engineering & System Safety, 93(7):964–979, 2008.

[123] G. Tang and G. Iaccarino. Subsampled Gauss Quadrature Nodes for Estimating Polynomial Chaos Expansions. SIAM/ASA Journal on Uncertainty Quantification, 2(1):423–443, 2014.

[124] William C. Thacker, Ashwanth Srinivasan, Mohamed Iskandarani, Omar M. Knio, and Mathieu Le Henaff. Propagating Boundary Uncertainties Using Polynomial Expansions. Ocean Modelling, 43–44:52–63, 2012.

[125] Robert Tibshirani. Regression Shrinkage and Selection via the Lasso. Journal of the Royal Statistical Society. Series B (Methodological), pages 267–288, 1996.

[126] Lloyd N Trefethen. Is Gauss Quadrature Better Than Clenshaw-Curtis? SIAM review, 50(1):67–87, 2008.

[127] Y. Tsaig and D.L. Donoho. Extensions of compressed sensing. Signal processing, 86(3):549–571, 2006.

[128] E. van den Berg and M. P. Friedlander. SPGL1: A Solver for Large-Scale Sparse Reconstruction, June 2007. http://www.cs.ubc.ca/labs/scl/spgl1.

[129] E. van den Berg and M. P. Friedlander. Probing the Pareto Frontier for Basis Pursuit Solutions. SIAM Journal on Scientific Computing, 31(2):890–912, 2008.

[130] Manav Vohra, Justin Winokur, Kyle R Overdeep, Paul Marcello, Timothy P Weihs, and Omar M Knio. Development of a reduced model of formation reactions in Zr-Al nanolaminates. Journal of Applied Physics, 116(23):233501, 2014.

[131] Jörg Waldvogel. Fast Construction of the Fejér and Clenshaw–Curtis Quadrature Rules. BIT Numerical Mathematics, 46(1):195–202, 2006.

[132] N. Wiener. The Homogeneous Chaos. Amer. J. Math., 60:897–936, 1938.

[133] J. Winokur, D. Kim, F. Bisetti, O.P. Le Maître, and O.M. Knio. Sparse Pseudo Spectral Projection Methods with Directional Adaptation for Uncertainty Quantification. Journal of Scientific Computing (submitted), 2015.

[134] Justin Winokur, Patrick Conrad, Ihab Sraj, Omar Knio, Ashwanth Srinivasan, W.Carlisle Thacker, Youssef Marzouk, and Mohamed Iskandarani. A priori testing of sparse adaptive polynomial chaos expansions using an ocean general circulation model database. Computational Geosciences, 17(6):899–911, 2013.

[135] D.B. Xiu and G.E. Karniadakis. The Wiener-Askey Polynomial Chaos for stochastic differential equations. SIAM J. Sci. Comput., 24:619–644, 2002.

[136] Liang Yan, Ling Guo, and Dongbin Xiu. Stochastic Collocation Algorithms Using ℓ_1-Minimization. International Journal for Uncertainty Quantification, 2(3), 2012.

[137] Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 68(1):49–67, 2006.

[138] Luis Zamudio and Patrick J Hogan. Nesting the Gulf of Mexico in Atlantic HYCOM: oceanographic processes generated by Hurricane Ivan. Ocean Modelling, 21(3):106–125, 2008.

Biography

Justin Winokur, born October 13, 1987, is a Philadelphia, PA area native where he attended Germantown Friends School. As an extension of a one-month co-op program, Justin became a science presenter at The Franklin Institute in Philadelphia; one of the country's premier science museums.

Justin's insatiable curiosity led him to a BS (with high honors) in physics at Carnegie Mellon University in 2010. He spent Summer 2008 performing research in experimental chaotic fluid dynamics with Dr. Tom Solomon at Bucknell University as part of the NSF REU program. His experience ignited a passion for the thermal-fluids discipline and a desire to pursue more research. In 2009, he was a Summer Intern at JHU/APL developing a gamma-ray spectroscopy validation framework.

Justin began his graduate career at Johns Hopkins University under the guidance of Omar Knio where he began research in uncertainty quantification. Justin graduated from Hopkins in 2012 with an MSE in Mechanical Engineering and followed Dr. Knio to Duke University, where he will be completing his requirements for a PhD in Mechanical Engineering and Materials Science in the Spring 2015. He will be joining Sandia National Labs in Albuquerque, NM as a Senior Member of Technical Staff in the Verification & Validation, Uncertainty Quantification and Credibility Processes Department.